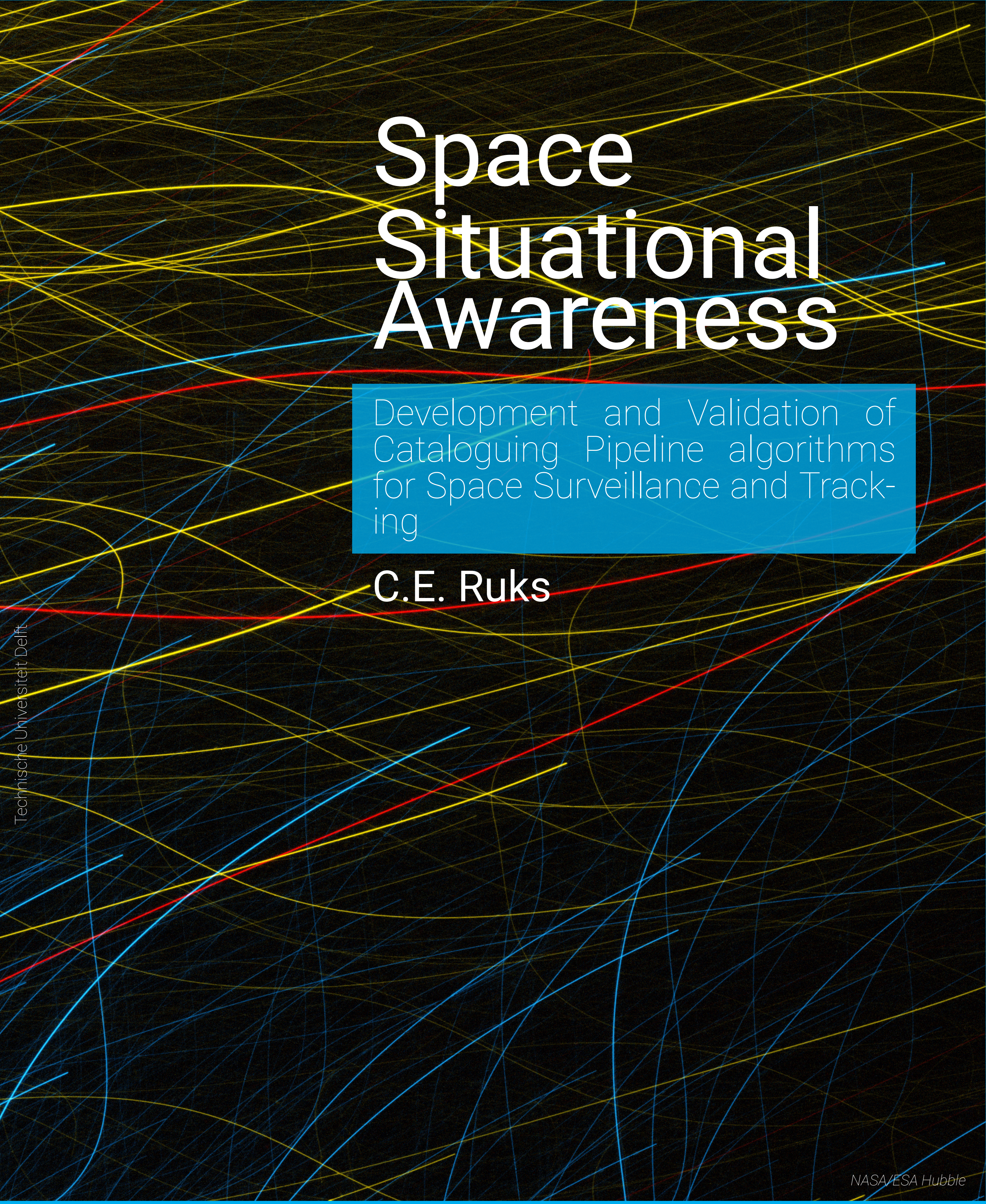


NASA/ESA Hubble

Space Situational Awareness

Development and Validation of Cataloguing Pipeline algorithms for Space Surveillance and Tracking

C.E. Ruks

Technische Universiteit Delft

TU Delft

Space Situational Awareness

Development and Validation of Cataloguing Pipeline algorithms for Space Surveillance and Tracking

by

C.E. Ruks

to obtain the degree of Master of Science
at the Delft University of Technology,
to be defended publicly on Thursday January 23, 2025 at 15:00.

Student number:	5644348	
Project duration:	May 1, 2024 – Jan 23, 2025	
Thesis committee:	ir. K.J. Cowan,	TU Delft, Chair
	Dr. A. Menicucci	TU Delft, Examiner
	Dr. S. Gehly PhD,	TU Delft, Supervisor
	Dr. M. Langbroek PhD,	TU Delft, Supervisor
	B. Kieboom,	Airbus Defence and Space, Supervisor

An electronic version of this thesis is available at `http://repository.tudelft.nl/`.

Cover: "NASA/ESA Hubble" licensed under CC BY 4.0. Acknowledgements: A. Sarajedini (University of Florida) and Judy Schmidt.

Acknowledgments

This thesis marks the end of my MSc. in Space Engineering and the research performed at Airbus Defence and Space, Friedrichshafen.

First and foremost, I would like to thank Bart Kieboom for giving me the opportunity to work on this project, for making me feel welcome at Airbus, and for his openness and willingness to assist and discuss my concerns whenever needed. I would also like to thank Steve Gehly and Marco Langbroek, for their time, patience and guidance. Your supervision and the many opportunities to engage in thoughtful discussions and exchange ideas have been invaluable, helping me to critically analyse my approach and enhance my overall understanding.

Thanks also to everyone I met along the way, both inside and outside Delft; it would be a privilege to collaborate with you again in the future.

C.E. Ruks
Delft, December 2024

Abstract

This thesis investigates methods to enhance the robustness of space object cataloguing pipelines, focusing on tracklet correlation and orbit estimation using angular measurements from short observation arcs. The cataloguing robustness is defined as achieving high true positive and negative rates for tracklet correlation to allow for the build-up of an accurate object catalogue. The study addresses the main research question: How can the robustness of the cataloguing pipeline be improved when applying orbit estimation methods to the full angle set of short observation arcs?

A baseline tracklet correlation approach, based on the Boundary Value Problem (BVP) within the Admissible Region framework, is implemented. This method uses angular observations and hypothesized ranges to estimate an object's state, with correlations evaluated via a cost function based on the Mahalanobis distance. Classical IOD methods are employed to investigate their application toward validation of tracklet correlation when reconsidering the full angle set. The considered methods include the angles-only Gauss method, a multiple angles least-squares Gauss approach, Gooding's method, as well as a Batch Least Squares (BLS) orbit determination (OD) method. The BVP and IOD methods consider two-body dynamics, and the BLS Earth's zonal harmonics and third body effects from the Sun and Moon. Simulated measurements are derived from Two Line Element sets (TLE) for initial reference states for LEO, MEO, and GEO objects, propagated with the SGP4 model accounting for Earth's atmospheric drag, zonal harmonics and third body Sun and Moon effects, providing the test data.

Results show that the BVP method performs best for GEO, achieving ~90% true positive rates with reasonable uncertainty gating. For LEO and MEO, higher thresholds and cost-function minima are required due to greater observation complexity and force-model discrepancy. Gooding's method, making use of a Lambert solver, demonstrated robust performance across multiple orbital revolutions, while Gauss' methods were less effective for large time gaps. Additionally, BLS struggled with sparse data and large time steps, offering limited state refinement despite higher computational expense.

The findings suggest gating based on chi-squared distribution thresholds for GEO and higher magnitudes for LEO and MEO to optimize true negative rates. While the BVP method provides sufficient accuracy for re-observation scenarios, classical IOD methods and BLS exhibit limitations under sparse tracklet conditions. This work highlights challenges in cataloguing lower-altitude objects, for ground-based optical observations, and suggests the application of the BVP method on lower altitudes requires inclusion of force models for the primary perturbations.

Nomenclature

Abbreviations

Abbreviation	Definition
AR	Admissible Region
ART	Airbus Robotic Telescope
AU	Astronomical Unit
BFGS	Broyden-Fletcher-Goldfarb-Shannon
BLS	Batch-Least-Squares
BVP	Boundary Value Problem
CAR	Constrained Admissible Region
ECEF	Earth-Centred-Earth-Fixed
ECI	Earth-Centred-Inertial
FP	False Positive
FN	False Negative
GEO	Geostationary Orbit
HEO	Highly Eccentric Earth Orbit
IOD	Initial Orbit Determination
IVP	Initial Value Problem
L-BFGS-B	Low-memory, bounded BFGS
LEO	Low Earth Orbit
MCC	Matthews Correlation Coefficient
MEO	Medium Earth Orbit
OD	Orbit Determination
PAR	Probabilistic Admissible Region
PAR+	Possibilisitc Admissible Region
RMSE	Root-Mean-Square-Error
RSO	Resident Space Object
SDP4	Simplified Deep Space Perturbations model
SGP4	Simplified General Perturbations model
SPOOK	SST software suite Special Perturbations Orbit determination and Orbit analysis toolKit
SSA	Space Situational Awareness
SSN	US Space Surveillance Network
SST	Space Surveillance and Tracking
TP	True Positive
TN	True Negative
TLE	Two Line Element set
VD	Virtual Debris
VP	Virtual Point

Contents

1

Introduction

A key activity for supporting a long term sustainable outer space environment is the identification and tracking of space debris, enabling to monitor and potentially mitigate or reduce collision hazards. In 2009 two communication satellites, the Iridium 33 and inactive Cosmos 2251, collided at an altitude of 790 km. This was the first unintentional collision between intact objects, as of June 22 2012 about 1881 associated orbiting fragments had been detected by the US Space Surveillance Network (SSN) [35]. The orbital perturbations caused the debris cloud to spread globally, posing an even greater risk of collision for LEO objects with a substantial fraction expected in orbit decades after impact [33]. It is clear that monitoring space activities and objects, warning and anticipating for potential collisions, is essential for safeguarding sustainable space activities in near-Earth space. Within Space Situational Awareness (SSA), Space Surveillance and Tracking (SST) is in charge of cataloguing and mapping the Space Debris population and other Resident Space Objects (RSO) in the near Earth environment [49].

To support SST research and development, the Airbus Robotic Telescope (ART) and the SST software suite Special Perturbations Orbit determination and Orbit analysis toolKit (SPOOK), allow for generation of real-world and simulated sensor data, as well as RSO catalogue maintenance [10]. Around these tools, a pipeline is being developed for cataloguing of RSOs without a priori information. This discipline is referred to as Initial Orbit Determination (IOD).

The pipeline, setting the context in which the work is to be carried out is described as follows. Starting from multiple short duration ground-based optical angle measurements, usually consisting of five angle pairs spanning about 20 seconds, called tracklets, a comparison is performed with already catalogued objects. When a tracklet is successfully associated to a catalogued object it is used to refine the catalogued object's state. Alternatively, when it is not successfully correlated to an existing catalogue object, it is grouped with other tracklets that did not correlate with catalogued objects. These tracklets are then compared with each other as to investigate whether they originate from a common newly identified object. The angle measurements obtained, however, contain noise caused by, e.g., instrument noise or atmospheric aberrations. Together with errors introduced in the correlation models, uncertainty in correct association between tracklets and objects is present. In certain cases, due to orbital geometries or uncertain data, tracklets may end up being falsely associated with each other. False associations impact the overall space surveillance and tracking activities limiting accurate revisit observations that are necessary to acquire additional measurements for accurate catalogue maintenance. Accordingly, it is beneficial for surveillance and tracking purposes to minimize incorrect associations in tracklet-to-tracklet correlation and association to existing objects. In addition to robustness, the substantial quantity of angle data and the all vs. all approach to tracklet correlation motivates the need for a low-complexity, minimum-computational-burden implementation.

1.1. Research objective & research questions

The aim of this thesis is to implement existing estimation methods and modify the cataloguing pipeline in an effort to improve its robustness at minimum computational expense. The cataloguing robustness is defined as the ability to capture truthful tracklet correlations. More specifically, it aims to achieve high true positive and true negative rates for tracklet correlation.

Estimation methods of varying nature, i.e., different use and degree of information, are implemented into the cataloguing pipeline. A multiple angle IOD method is considered to investigate its effectiveness when using the combined raw tracklet data spanning a larger time interval at an expected lower computational burden. Additionally, a Batch Least Squares (BLS) orbit determination method is considered to investigate the balance between acquired state accuracy, computational burden and sensitivity to detect correlation. A baseline for the cataloguing pipeline is established as the boundary value method as described in [59]. The process employing the baseline approach will be modified using the aforementioned estimation methods. The baseline will provide a basis for comparison within the context of this thesis and facilitate reference to other work. Performance is quantified using simulated observations according to ART's specifications, using truth-catalogued data to evaluate and validate the approach. Performance metrics include the time required to obtain the state estimate, the accuracy of the estimated state, and the robustness defined by the degree of truthful associations.

The approach will be applied to measurements from objects in different orbital regimes. This will allow the behaviour of the method to be characterised and understood in a broader context. Altogether, this work will address whether such an approach can provide a meaningful improvement on the cataloguing pipeline with respect to robustness, computational burden and state estimate accuracy. To reflect the research objective, the following main research question is formulated.

> How can the robustness of the cataloguing pipeline be improved when considering application of orbit estimation methods to the full angle set of short observation arcs?

1.2. Thesis structure

The thesis starts out by presenting an overview of the relevant literature explored for the discipline of IOD, see Chapter 2. After the literature is reviewed, a theoretical background is provided. First astrodynamic theory is introduced together with the relations used for optical measurements in Chapter 3. Then, the necessary background theory on statistics and numerical optimization is provided in Chapter 4. This allows to introduce the tracklet correlation approach and admissible region framework as part of the cataloguing pipeline in Chapter 5. After introducing the baseline method for tracklet-correlation, the considered initial orbit determination methods and the OD batch-least squares method are described in Chapter 6. The methods introduced in the previous chapter are verified in Chapter 7. The methodology and steps taken for measurement generation and the tracklet-correlation are described in Chapter 8. The results of the tracklet-correlation performed with the admissible region framework as well as the effect of the estimation methods as an additional step are presented and discussed in Chapter 9. Finally, the work is concluded in Chapter 10 and recommendations for future work are proposed.

2

Literature review & Motivation

The problem of IOD from angles only is to find the orbital state, which is defined by the six constraining parameters. A single optical measurement provides only two independent components, which is insufficient to estimate a six-dimensional orbital state. Therefore, to determine the unique solution of the orbital state, the angular measurements must be extended by the appropriate number of independent parameters. This problem has been extensively studied and remains a topic of interest due to advancements in observation technology that led to the acquisition of a significant amount of unclassified short observation arcs. Accordingly, this chapter will outline the explored literature for this thesis. It starts out by providing an overview of the classical methods and shows studies which have compared these methods in section 2.1. The problem of too-short arcs is introduced together with the relevant literature aimed to address the problem in section 2.2 and section 2.3. Last, some alternative IOD methods or approaches for tracklet correlation are presented in section 2.4.

2.1. Classical IOD Methods

Already in 1780 a method for determining the orbit from angle observations was proposed by Laplace [14]. It was originally developed for celestial objects and is, together with Gauss' method, considered part of the classical IOD methods. The classical methods extend the two independent angles by considering a set of three angle pairs at different epochs, allowing to estimate the orbit from six independent parameters. While Gauss' method performs better for near-Earth objects, it is limited by the requirement of smaller angular spacing ($< 10°$) between observations [14, 66]. Subsequent advancements like the Double r-iteration [14] and Gooding's method [20] have proven more robust and allow for larger separation in measurements, allowing data spanning multiple observation nights [60].

Several comparative studies have evaluated the aforementioned IOD methods for different orbital regimes and data conditions. Vallado, evaluated Gooding's method, finding it robust and sensitive to the selection of observation points, with improvements achieved by averaging orbital elements and using observations spaced farther apart (maximum included separation of 1-2 minutes) [65]. Schaeperkoetter, evaluated Laplace, Gauss, Double r-iteration, and Gooding on various orbits (polar, sun-synchronous, Molniya, GEO) and introduced new error quantification parameters [55]. His work concluded that Laplace and Gauss showed a decrease in accuracy as the observation interval increased, where Laplace decreased more rapidly. Furthermore, he observed Gooding's method to improve in accuracy with longer observation intervals beyond 5 degrees, about several orders of magnitude better compared to the other methods. Fadrique et al., compared Gauss and Gooding for LEO, GEO, MEO, and HEO using simulated data [16]. The main finding was the challenge of accurately estimating orbits with only angular measurements, especially when observation separations are small, suggesting to collect as many observations as possible as to sample at maximum separations. Hwang et al., compared Laplace, Gauss, Double r-iteration, and Gooding using simulated data, excluding Laplace for real-world data comparisons [28]. Hwang et al. showed Gauss to perform best with observation time intervals shorter than 10 seconds, and with increasing errors as the time interval increased. The double-r method failed to converge for intervals shorter than 30 seconds, making it suitable only for longer intervals where its performance was similar to that of the Gooding's method. They concluded Gauss' method to be most suitable for an initial guess for OD, with Gooding and double-r exhibiting a higher failure rate when applied to real observational data. Johnson, focused on Gauss, modified Gauss (assuming circular orbits), Double r-iteration, and Gooding for LEO short arc observations [32]. He concluded, Gooding and double-r to be highly dependent, yielding accurate results for some orbits while failing to converge for others. Johsnon also concluded Gauss', and Gauss' circular method to produce reliable solutions for orbital arcs below 5 degrees. Van den Abbeele, compared Gauss, modified Gauss, and Gooding for GEO short arc observations [68]. He concluded that all methods are sensitive to measurement noise when considering short observation arcs. Accuracy improved with increasing arc-length, finding the lower limit at 6% of the orbit. Especially Gauss' showed to be decreasing in accuracy for longer angle separation. Gooding's method was found to obtain higher errors than the circular assumed Gauss method. Miller and Freuh, compared Gooding and Gauss' IOD methods for simulated orbits and concluded Gooding to be more accurate in position and velocity estimates [42]. In particular, they concluded that the best case scenario for Gooding occurs when measurement spacing is greater than five degrees, but the total span of the measurements not exceeding a quarter of the target's orbit. For GEO orbits with an eccentricity in the range of 0.1 to 0.3, and large measurement intervals (>550 minutes), Gooding shows high errors in both position and velocity, however overall still lower than Gauss.

2.2. Too short arcs

When short observation arcs are considered, the classical IOD methods all show to be very sensitive to small measurement errors, as small changes in measurement errors can result in widely different orbit estimates [22]. For the initial value formulation methods such as Laplace's method, this sensitivity is caused by the small initial line-of-sight acceleration estimate which is propagated and amplified as it is inversely proportional to the range estimate. For a boundary value formulation method, e.g. Lambert's problem, it is dependent on capturing the orbit's curvature as depicted in Figure 2.1. This means that the classical methods generally require relatively long tracklet lengths for stable state estimates. For GEO objects about 15 min was found to be desired [16], or 5 to 10 minutes for LEO [66].

In addition to the requirement of longer tracklet lengths when using classical methods, the observation strategy for identifying new objects is a trade-off between coverage and accuracy. High coverage, i.e. being able to observe many different objects, suffers when longer observation is required for objects. Moreover, orbit geometry and visibility conditions do not always allow for adequate observation time.

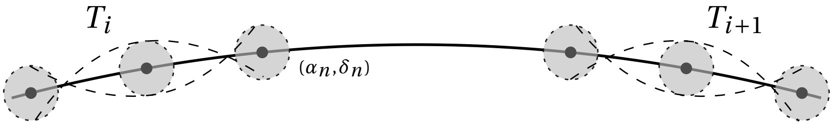


Figure 2.1: Short arc problem, showing sensitivity of small errors on orbit. The solid line represents the true orbit. The dotted circles represent the error region surrounding the true object position, with the dashed lines the estimated curvatures corresponding to the extreme case of false angle pairs. Figure is adapted from [60] and [22]

New developments took place to acquire better initial estimates from measurements that cover too little of the object's orbit for classical methods to obtain reasonable results. This ultimately lead to a new class of IOD methods suitable for the incorporation of more sophisticated models and approaches. Milani et al., introduced the Admissible Region (AR) method for asteroids, which extends four angle measurements, i.e. two angle pairs as well as two angle-rates, to six independent parameters by obtaining an admissible region in range and range-rate space [41]. It used the attributable, a composition of the mean angles and angle-rates obtained from a least squares fit to the tracklet, and constrained the admissible region in range and range-rate by considering an asteroid contained within the solar system. It was shown to effectively constrain feasible solutions for the extended parameters and sampled the region uniformly using triangulation. Each sample, defined as a virtual asteroid (VA), was then used to define the expected location on the celestial sphere.

Since then, the AR method has been applied to Earth captured objects. Tommei et al., applied the AR approach to Earth-orbiting satellites by constraining the region for orbital energy and upper and lower bounds on the range [64]. Still, the AR was uniformly using triangulation, obtaining a set of equally likely Virtual Debris (VD). DeMars and Jah, extended this framework and focused on the sampling of the AR [13]. Additionally, they further constrained the AR, i.e. the Constrained Admissible Region (CAR), by including constraints on eccentricity and semi-major axis. They assumed a two-dimensional uniform distribution, and discretized the CAR by a Gaussian mixture model representing this distribution. It was shown to effectively represent the feasible space of solutions and allowed to filter the region with additional observations. They showed that the average root sum square error of the sampled region reduces with filtering. Still developments took place toward sampling of the complete AR toward a more representative distribution. Hussein et al., extended the CAR method by considering measurement uncertainty and accounting for consistent orbit geometry constraint to the Probabilistic Admissible Region (PAR). It only samples the region which satisfies both geometry constraints simultaneously, showing to obtain a better representation over the AR [27]. Cai et al., then also extended the sampled region to incorporate epistemic uncertainty[1] through a possibility function, yielding more reliable range and range-rate points in the Possibilistic Admissible Region (PAR+) with lower error estimates compared to PAR [7].

2.3. Tracklet correlation

Instead of testing all possible samples in the admissible region as done in previous methods, Siminski et al. aimed to only extract the best fitting virtual point reducing the number of evaluations [61]. They use previous work of the CAR, and include an additional attributable as to express a cost function and search for the virtual point corresponding to the optimum solution. The CAR is thus associated with a topography corresponding to a set of attributables. This topography is expressed by the squared residuals between the measured and modelled attributable, scaled by the measurement and modelled uncertainty given an hypothesised VP. It was demonstrated to outperform the complete uniform sampling methods, obtaining the optimum in just a few evaluations using a pattern search optimization method. It considered the geostationary regime and tracklets spanning about two minutes. This allowed to correlate two tracklets based on gating the cost function and obtain an initial orbital estimate. Around the same time, Ansalone and Curti presented a similar approach based on space-based observations [1]. They use a genetic algorithm to search for the range and range-rate, minimising the error between the estimated and measured angles. They compared the method against classical methods and showed improved performance in state estimates for short arcs spanning 60 seconds.

Siminski continued to study the correlation approach, having presented the initial value method included a boundary value method that uses the data of the attributable differently, obtaining an admissible region in the outer ranges instead [59]. Based on an extensive study comparing both approaches, the BVP method was found to be more robust and efficient permitting higher simplicity for implementation especially when considering two-body dynamics [60]. The IVP approach is more complex and requires different optimization techniques compared to BVP, having less favourable topography as the loss function contains many local minima. Additionally, the representative uncertainty transformation from a measured to modelled attributable proves to be more challenging for the IVP method. To improve this representation, Cai et al. used the Euclidean distance between the estimated orbits in a non-singular canonical space using Delaunay variables as cost function. It showed higher performance in true positive and true negative rates compared to the approach using a linearly approximated uncertainty transform in the measurement space [8]. Additionally they developed a common ellipse method to exclude false positives for objects in the same constellation, showing to successfully improve on the true negative rate.

Alternatively, Huang et al., aimed to improve the IOD convergence by correlating short arcs using a Lambert solver and least squares fit [26]. The method assumes a GEO circular orbit, which allows the range to be solved directly, as the orbit is then described with four degrees of freedom (semi-major axis, inclination, right ascension of the ascending node and the true anomaly). This method assesses correlation based on the residuals between measured and estimated measurements. The approach obtained high true positive rates for simulated GEO tracklets of 3 minutes separated by 12h or less. Zhao et al., developed an IOD tracklet-to-tracklet method to determine the orbital elements from two tracklets using only Gooding's method [71]. To enable tracklet correlation of tracklets separated by multiple nights, secular effects from the J_2 and luni-solar gravitational perturbations as well as short-periodic effects due to J_2 and $J_{2,2}$ are included. Correlation is defined as a threshold on the combined root mean square between each measured tracklet and final state estimated using Gooding. The approach showed good true positive (TP) and false positive (FP) rates. The eccentric orbits (HEO and Molnyia) showed to perform worst, setting the performance boundaries for the study by ≥89.8% TP and ≤4.7% FP rates. For single-threaded computation, around 1000 tracklet to tracklet cases were performed per minute for LEO and HEO orbits while around 1500 for MEO and GEO.

2.4. Extension of IOD methods to multiple angles

Other papers investigated the effect of multiple angles and optimising angle sets. Karimi and Mortari, proposed Jacobian and Least-Squares approaches to estimate the range vector using Lagrange coefficients from

---

[1] Epistemic uncertainty comes from limited knowledge rather than the random nature of a process.

three or more angle measurements [34]. The method derives the ranges similar to Gauss' method and allows to solve the Lambert problem using the most orthogonal range estimates to obtain a midpoint range and velocity vector. This method showed to increase the state accuracy by considering larger time span angles and can handle co-planar conditions in which classical three angle pair IOD methods diverge. The method developed will be further investigated in section 7.2. Wishnek, included angle-rates and used a least-squares approach to solve for range and range-rate vectors, reformulating the method into a cost function optimized via admissible region sampling [70]. It showed to be more robust for certain cases, but at a significantly higher computational time.

2.5. Motivation

As becomes clear by the literature presented, efforts toward the short arc problem are an ongoing development toward robust object cataloguing. The main difficulty faced is the dependency on observation strategy and orbital geometry, denying an optimised universal approach for varying orbit regimes. The use of attributables for sufficiently long tracklets in the BVP correlation method has been shown as a promising candidate for GEO orbits. Still, its correlation performance is dependent on a pre-defined threshold impacting the resulting true positive and true negative rates. When extending its application outside of the GEO regime, the threshold is expected to require tuning dictated by the observation quality associated with different orbital geometries [71]. In the end, the required balance between true positive and true negative rates needs to be found as to allow for adequate cataloguing performance. As a first step towards this goal, this thesis will investigate the impact of existing estimation methods in an effort to improve its robustness at minimum computational expense. Essentially improvements in the true positive and true negative rates is sought, regardless of a required false negative rate. The performance, in addition to the obtained true positive and true negative rates, will be assessed in terms of the state accuracy and computational time. Further research can then be aimed toward understanding the impact of this approach in relation to a predefined observation strategy. Having characterised the approach's performance in this study, subsequent work could characterise its implications toward an efficient yet comprehensive object catalogue. For example, to which degree truthful correlation limit adequate object cataloguing performance.

3

Astrodynamics & Measurements

The following chapter describes the fundamental theory necessary for the thesis work. It aims to consistently introduce the concepts and terms as readily provided by textbooks [63, 9, 43] within the context of Initial Orbit Determination. It starts of by introducing common coordinate systems necessary for use in describing satellite orbit dynamics. Then the two body equations of motion are briefly discussed and expanded on with discussion of perturbations. Finally, the aspects of observations and cataloguing pipeline is presented.

3.1. Coordinate system

To describe an object in position and time, consistent coordinate systems and time notations are necessary. Additionally, the equations of motion and analyses consider an inertial system. Following the purposes of this thesis it is convenient to start with considering the observer's coordinate system and the required transformations necessary to arrive at a common system shared among all observations. Accordingly, an Earth-centred, Earth-fixed (ECEF) system is considered. It consist of orthogonal axes (X, Y, Z) whose origin coincides with Earth's centre of mass ($\oplus$). The Z-axis coincides, approximately due to Earth's polar motion, with Earth's angular velocity vector, ω_E. The X-axis then coincides with the Greenwich meridian[1] and equator, and the y-axis perpendicular to the X-Z plane. Another system, the topocentric coordinate system (x, y, z), considers the origin located on a point on the Earth's surface. Then, the location of the point on the surface determines the direction of the z-axis as the direction perpendicular to Earth's surface. The orthogonal x-axis toward the East, and the y-axis toward the North. Both of these systems provide a useful relation between the Earth-satellite-observer as shown in fig. 3.1. The ECEF system, which is rotating w.r.t. the stars, is useful for describing the observer location on Earth. However, for space objects it is more convenient to work in the Earth-centred

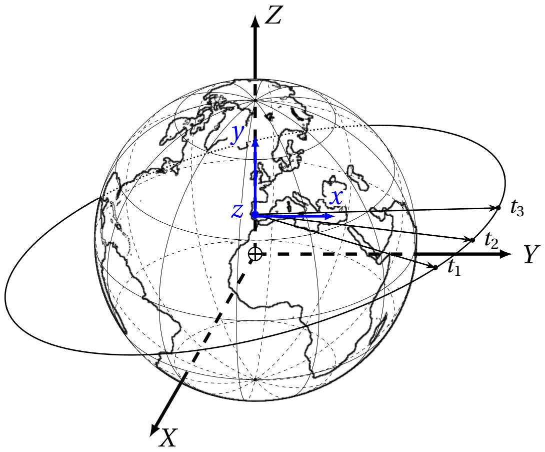


Figure 3.1: Earth-centred, Earth-fixed (ECEF) (X, Y, Z) and topocentric (x, y, z) coordinate systems with position vectors at three epochs.

---

[1] The location of a great circle crossing both the North and South poles through a standardised zero point of longitude [9].

inertial (ECI) coordinate system. This system is non-rotating w.r.t. the stars, whose reference axis is fixed to a location in space at a fixed epoch. The ECI system (X, Y, Z) can be approximated as the ECEF system rotated by an angle α_G, such that its X-axis aligns approximately with the vernal equinox. The vernal equinox is defined by the intersection line of Earth's equatorial plane and its orbital plane around the Sun, known as the ecliptic. The point where Earth crosses the ecliptic from south to north, called the ascending node, marks the vernal equinox. A common ECI system is the coordinate system aligned by the vernal equinox on the first of January, 2000 (J2000.0). The relation between the ECEF and ECI system is thus determined by the angle between the vernal equinox and Greenwich meridian, i.e. α_G. Additionally, the effects of precession, nutation and alignment with angular velocity must be accounted for in the transformation between these axes.

3.2. Equation of motion

This section will introduce the fundamental astrodynamic relations describing the space object's orbit. The parameters used to describe an object's state and its orbit are briefly outlined. The equation of motion of point mass M_1, relative to a non-rotating reference frame with point mass M_2 as the origin, can be derived by considering the gravitational force exerted by M_2 on M_1 [69]. The equation of motion for the restricted two-body dynamics is derived by considering $M_2 \gg M_1$, neglecting object's mass ($\mu = GM_2$). Additional perturbing accelerations can be included by adding a perturbing acceleration term $\mathbf{a}_p$.

$$\ddot{\mathbf{r}} = \frac{-\mu}{r^3}\mathbf{r} + \mathbf{a}_p \tag{3.1}$$

The equations of motion are valid in an inertial coordinate system with bodies considered as point masses. Perturbations on the two-body dynamics can be of gravitational and non-gravitational nature. Commonly known are the perturbing accelerations due to non-uniform mass distribution, atmospheric drag, solar radiation pressure and three-body effects. Inclusion of perturbing accelerations requires more effort in computation, as often the perturbing accelerations themselves require expensive models for accurate representation, e.g. atmospheric density. Additionally, the geometry and object properties need to be known beforehand, e.g. frontal area, mass and drag coefficient.

3.3. Object state parameters and simplified orbit geometry

Provided the object's position $\mathbf{r}$ and velocity vector $\dot{\mathbf{r}}$ are known, the orbit geometry can be described under two-body dynamics by the Keplerian elements $(a, e, i, \Omega, \omega, \nu)$. These elements allow for a simplified representation of the orbit, providing insight into its shape, orientation, and the object's position along the orbit. The relations of the state parameters and the orbital elements are briefly introduced for later sections. For a detailed derivation and description of the initial value solution and orbital elements, the reader is referred to textbooks from Wakker [69] and Tapley et al. [63].

The orbital elements are derived from the position and velocity vectors as introduced in the preceding section. The object's position and velocity vectors at a common epoch are used to derive the six orbital elements $(a, e, i, \Omega, \omega, \nu)$ describing the orbit's geometry. The first two elements, the semi-major axis a and the eccentricity e determine the size and shape of the orbit. The third, fourth and fifth parameters, the right ascension of the ascending node Ω, the argument of periapsis ω and inclination i, describe the orbit's orientation in space. Lastly, the true anomaly ν defines the location of the object along its orbit by an angle relative to the periapsis (or perigee for Earth-orbiting objects). An overview of the six elements is provided in fig. 3.2.

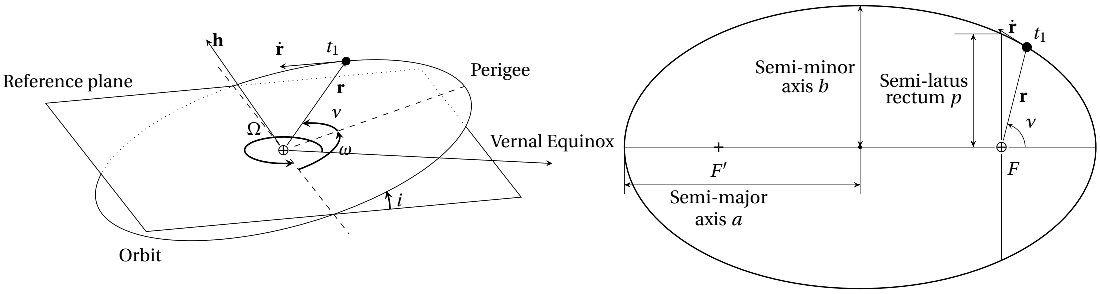


Figure 3.2: The six orbital elements. Left the 3-dimensional schematic, right the orbital plane in the perifocal coordinate system.

3.3.1. Initial value solution

From the equation of motion, the state at a future epoch can be estimated by integration of the state derivative vector, defined respectively as

$$\mathbf{X} = \begin{bmatrix} \mathbf{r} \\ \dot{\mathbf{r}} \end{bmatrix}, \quad \dot{\mathbf{X}} = \begin{bmatrix} \dot{\mathbf{r}} \\ \ddot{\mathbf{r}} \end{bmatrix}. \tag{3.2}$$

Two formulations exist, to estimate the state at a future epoch. An initial value solution and a boundary value solution. The initial value solution propagates the full initial state $\mathbf{X}(t_0)$ by single integration over a time period Δt.

$$\mathbf{X}(t_0 + \Delta t) = \mathbf{X}(t_0) + \int_{t_0}^{t_0+\Delta t} \dot{\mathbf{X}}(t) dt \tag{3.3}$$

The integration is commonly performed numerically, e.g. Runge-Kutta 4 or a multi step integration method [66].

3.3.2. Lagrange coefficients

A convenient approach in obtaining the two-body dynamics constrained initial value solution can be obtained by using the Lagrange coefficients. The Lagrange coefficients are derived by considering the perifocal frame (fig. 3.2), i.e. origin centred on the orbit's focal point F with xy-plane coinciding with the orbital plane and x axis pointing to perigee [11]. In this frame the position and velocity vector are defined in the orbital plane by

$$\begin{aligned} \mathbf{r} = x\hat{\mathbf{p}} + y\hat{\mathbf{q}} \quad &= r\cos v\, \hat{\mathbf{p}} + r \sin v\, \hat{\mathbf{q}} \\ \dot{\mathbf{r}} = \dot{x}\hat{\mathbf{p}} + \dot{y}\hat{\mathbf{q}} \quad &= (\dot{r}\cos v - r\dot{v}\sin v)\, \hat{\mathbf{p}} + (\dot{r}\sin v + r\dot{v}\cos v)\, \hat{\mathbf{q}} \end{aligned} \tag{3.4}$$

Where, v is the angle between the position vector and perigee, r the distance from F to the object and $[\hat{\mathbf{p}}, \hat{\mathbf{q}}, \hat{\mathbf{w}}]$ the axis unit vectors, with $\hat{\mathbf{w}}$ parallel to the angular velocity vector $\mathbf{h}$. Assuming unperturbed dynamics, it can be shown that the angular momentum is constant in this case [69]. Rewriting the equation of motion by its anti-derivative, and since $\dot{\mathbf{r}} \times \dot{\mathbf{r}} = 0$ and $\mathbf{r} \times \mathbf{r} = 0$, it evaluates to

$$\frac{d\mathbf{h}}{dt} = \dot{\mathbf{r}} \times \dot{\mathbf{r}} + \mathbf{r} \times \ddot{\mathbf{r}} = 0. \tag{3.5}$$

This allows to evaluate the magnitude of the angular momentum, shown constant, by

$$h = |\mathbf{r}_0 \times \dot{\mathbf{r}}_0| = x_0\dot{y}_0 - y_0\dot{x}_0. \tag{3.6}$$

Rewriting eq. (3.4) and substituting the obtained relations, eventually results in an expression for $\hat{\mathbf{p}}$ and $\hat{\mathbf{q}}$

$$\hat{\mathbf{p}} = \frac{\dot{y}_0}{h}\mathbf{r}_0 - \frac{y_0}{h}\dot{\mathbf{r}}_0, \tag{3.7}$$

$$\hat{\mathbf{q}} = \frac{\dot{x}_0}{h}\mathbf{r}_0 - \frac{x_0}{h}\dot{\mathbf{r}}_0. \tag{3.8}$$

Thus on the assumption of unperturbed motion, the position and velocity vectors at a given time can be written as functions of $\hat{\mathbf{p}}$ and $\hat{\mathbf{q}}$. Substitution of eqs. (3.7) and (3.8) into eq. (3.4) and simplifying finally yields

$$\mathbf{r} = \frac{x\dot{y}_0 - y\dot{x}_0}{h}\mathbf{r}_0 + \frac{yx_0 - xy_0}{h}\dot{\mathbf{r}}_0, \tag{3.9}$$

$$\dot{\mathbf{r}} = \frac{\dot{x}\dot{y}_0 - \dot{y}\dot{x}_0}{h}\mathbf{r}_0 + \frac{x_0\dot{y} - y_0\dot{x}}{h}\dot{\mathbf{r}}_0. \tag{3.10}$$

Where the fractional terms are known as the Lagrange coefficients $f, g, \dot{f}$ and $\dot{g}$,

$$\mathbf{r} = f\mathbf{r}_0 + g\dot{\mathbf{r}}_0, \tag{3.11}$$

$$\dot{\mathbf{r}} = \dot{f}\mathbf{r}_0 + \dot{g}\dot{\mathbf{r}}_0. \tag{3.12}$$

The Lagrange coefficients can be determined by exact functions or approximated by series, as will be discussed in section 6.1. The Lagrange coefficients will be used for the initial orbit determination method presented in chapter 6, allowing to determine the state vectors by linear combinations of the evaluated Lagrange coefficients and initial state.

3.3.3. Boundary value solution

An alternative way of determining an object's future state, requires instead two position vectors $\mathbf{r}_1$, $\mathbf{r}_2$ and a time-of-flight $t_2 - t_1$. This is referred to as the boundary value solution, or Lambert's problem. Starting from the initial position vector $\mathbf{r}$ and an initial guess for the velocity vector $\tilde{\dot{\mathbf{r}}}$, the initial augmented state is integrated similarly to the IVP solution

$$\hat{\mathbf{X}}(t_0 + \Delta t) = \tilde{\mathbf{X}}(t_0) + \int_{t_0}^{t_0+\Delta t} \hat{\dot{\mathbf{X}}}(t)dt. \tag{3.13}$$

The obtained propagated solution can then be used to correct the initial velocity vector $\dot{\mathbf{r}}$. This can be solved by several numerical root-finding approaches, and is often restricted to two-body dynamics. Lambert was the first to solve the problem of two outer position vectors and a time of flight, however several approaches have been published since. Based on the traditional approaches (Gauss, Euler [14] etc.), Lancaster and Blanchard [37] and later Gooding [21] presented robust approaches. Izzo build on the Lancaster and Blanchard approach, comparing to Gooding's numerical accuracy at significantly reduced computational effort. For the detailed discussion of Izzo's method, the reader is referred to [29, 30]. As will be discussed in section 5.3, Izzo's method will be used in the admissible region BVP method.

3.3.4. Lambert's Problem

When knowing the outer vectors $\mathbf{r}_1$ and $\mathbf{r}_2$ of the same object and the time-of-flight Δt connecting the two vectors, a well known initial orbit determination method can be applied to estimate the full orbital state. This approach, known as Lambert's problem, also allows to handle multiple revolutions between the given position vectors, specifically referred to as the multi-revolution problem. While perturbations can be included, the following constrains itself to unperturbed two-body motion. The solution to Lambert's problem is based on Lambert's theorem, which states that the time of flight between two position vectors depends only on the sum of these radial distances, the linear distance between them and the semi-major axis of the orbit [3]. This can be shown, starting from the integrated constant specific mechanical energy equation, by

$$\Delta t = \frac{1}{\sqrt{\mu}} \int_{s-c}^{s} \frac{r dr}{\sqrt{2r - r^2/a}}. \tag{3.14}$$

Here s is the semi-perimeter and c the chord given as

$$s = \frac{r_1 + r_2 + c}{2}, \quad \text{with} \quad c = |\mathbf{r}_2 - \mathbf{r}_1|. \tag{3.15}$$

The integral equation can be written in series form, which requires to be solved numerically for the semi-major axisz. For the purpose of Earth-captured objects, consider only elliptic orbits of finite semi-major axis. Assuming the semi-major axis is known, then following the relation

$$r_1 + r_2 = 2a, \tag{3.16}$$

the focus F' can be determined according to $r_1' = 2a - r_1$ and $r_2' = 2a - r_2$. The points where both these relations hold true then determines the location of F'. Altogether, having two position vectors allows to obtain the orbital plane represented by $\mathbf{h}$, i and Ω, assuming non-parallel vectors. Being able to numerically solve for the semi-major axis a allows to obtain the semi-minor axis b, eccentricity e, true anomaly ν and argument of perigee ω, given the focal distance FF'. For multiple revolutions it can result in two possible focii, F' and F''. Corresponding to the two focii two different orbit paths, a long and short period, can be obtained leading to four different orbits when including prograde or retrograde motion, as depicted in fig. 3.3. A more detailed description of the outlined steps is provided in [18].

3.4. Measurements

As part of SST, cataloguing of new RSOs is achieved through passive optical observations. Passive observations rely on an external light source to illuminate the RSO and reflect it back to the observer. In contrast, active observations (e.g. laser ranging and radar) work by emitting and receiving the reflected signal again. Active observations have the advantage that they can be used during night and day and depend less on weather conditions. Laser ranging can only be performed under good visibility conditions and is commonly used to validate other measurement techniques. Though, in contrast to active observations, passive observations require much less energy from the observer, especially for higher altitude objects. Within this thesis, only passive optical observations are considered, according to the capabilities of ART.

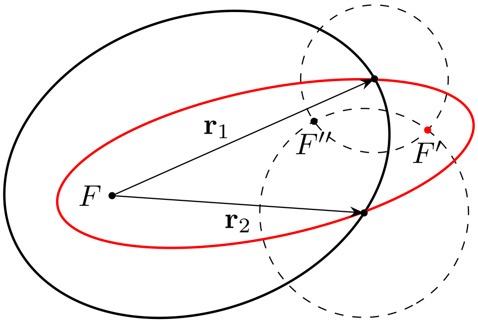
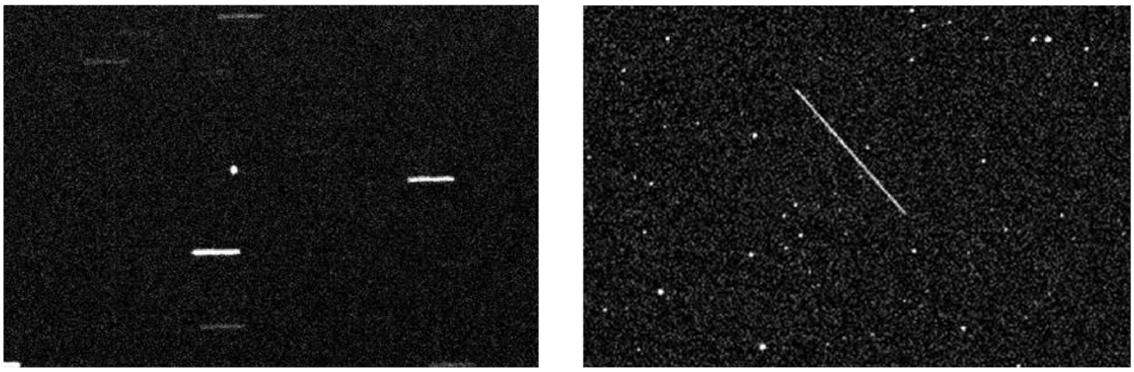


Figure 3.3: Geometry of Lambert's Problem showing the four possible orbital solutions given the outer position vectors, number of orbital revolutions, and time of flight. The four solutions include both prograde and retrograde motion, and the low (with focii F, F'') or high path (with focii F, F') solution. Figure adapted from [18].

3.4.1. Measurements

When identifying new RSOs, observations are made by surveying the sky, by means of sidereal tracking. In sidereal tracking, the background stars appear as point sources while the RSOs move at higher relative velocity and thus appear as streaks. On the contrary, when a RSOs orbit is known, the RSO may be tracked. In tracking-mode the RSO appears fixed on the image frame as a point source, while the background stars move relative to the tracked RSO and appear as streaks. This distinction is depicted in fig. 3.4.



(a) RSO tracking, stars appear as streaks.

(b) Sidereal tracking, stars appear as point sources.

Figure 3.4: RSO tracking vs. sidereal tracking [49].

Specifications of the telescope used are provided in appendix A. The telescope obtains angle data from image processing, which involves plate solving and noise filtering. The object's position consists of a pair of angles at the corresponding time t, i.e.

$$\mathbf{m}_i = (\alpha_i, \delta_i, t_i), \tag{3.17}$$

with the right ascension $\alpha \in [0, 2\pi)$ eastward along the celestial equator from the vernal equinox, and the declination $\delta \in [-\pi, \pi]$ along the hour circle positive north. These angles define the direction of the object in the sky with respect to the observer at time t, represented by the unit position vector

$$\mathbf{u}_i = \begin{bmatrix} \mathbf{u}_i \cdot \hat{\mathbf{i}} \\ \mathbf{u}_i \cdot \hat{\mathbf{j}} \\ \mathbf{u}_i \cdot \hat{\mathbf{k}} \end{bmatrix} = \begin{bmatrix} \cos(\alpha_i)\cos(\delta_i) \\ \sin(\alpha_i)\cos(\delta_i) \\ \sin(\delta_i) \end{bmatrix}. \tag{3.18}$$

And conversely

$$\begin{aligned} \alpha_i &= \arctan\left(\frac{\mathbf{u}_i \cdot \hat{\mathbf{j}}}{\mathbf{u}_i \cdot \hat{\mathbf{i}}}\right), \\ \delta_i &= \arcsin\left(\mathbf{u}_i \cdot \hat{\mathbf{k}}\right). \end{aligned} \tag{3.19}$$

Together, the range ρ and the line of sight $\mathbf{u}$ define the topocentric range vector of the object by $\boldsymbol{\rho}_T = \rho\mathbf{u}$. The geocentric range vector of the object $\mathbf{r}_i$ is obtained in combination with the geocentric range vector of the observer $\mathbf{R}$, as shown in fig. 3.5. Showing the relation between the Earth, observer and satellite as:

$$\mathbf{r}_i = \mathbf{R}_i + \rho_i \mathbf{u}_i \tag{3.20}$$

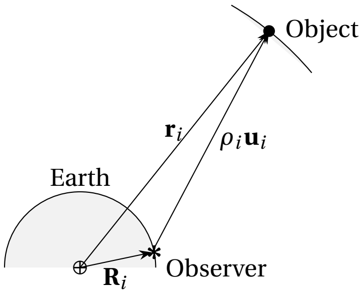


Figure 3.5: Relation of the observation position vectors.

3.4.2. Information from tracklet

Usually, multiple measurements $\mathbf{m}_{i=1}^N$ are taken shortly after each other, and readily correlated to form a tracklet

$$\mathbf{T}_j = (\mathbf{m}_i)_{i=1}^N, \tag{3.21}$$

here N denotes the number of measurements, typically in the range of 3 to 10. The measurements are obtained at time interval $\Delta t = t_e + t_r + \varepsilon$, dictated by the exposure time t_e, the readout time t_r and small errors represented by $\varepsilon \ll t_i + t_r$. When desired, apparent angular velocities may be obtained from the measurements contained in a tracklet and their time information, to form an attributable vector

$$\mathbf{A} = (\alpha_0, \delta_0, \dot{\alpha}_0, \dot{\delta}_0). \tag{3.22}$$

The attributable, which represents the angle information θ_0 and $\dot{\theta}_0$ of the tracklet at the middle epoch t_0, is obtained by a least squares fit to the tracklet. Where θ can be substituted for the measured topocentric right ascension α and declination δ. Following Maruskin's approach [40], the least squares fit assumes the angular motion to be modelled by

$$\theta(t) = \theta_0 + \dot{\theta}_0(t - t_0) + \frac{1}{2}\ddot{\theta}_0(t - t_0)^2. \tag{3.23}$$

By minimising the residual normalised by the measurement uncertainty, the corresponding angles and their rates can be found. Accordingly, the cost function is given by

$$J = \frac{1}{2\sigma_\theta^2} \sum_{i=-n}^{N} [\theta(t_i) - \theta_i]^2, \tag{3.24}$$

where $\theta(t_i)$ represents the model's estimate at time t_i and θ_i is the observed measurement at t_i. The observations are taken to be symmetrically distributed around the middle observation, $i = 0$, with constant time interval T ($t_i = t_0 + (T/n)i$). The solution to minimize the cost function is then obtained using the normal equations given by,

$$\Lambda \begin{bmatrix} \theta_0 \\ \dot{\theta}_0 \\ \ddot{\theta}_0 \end{bmatrix} = d \tag{3.25}$$

with the information matrix

$$\Lambda = \sum_{i=-n}^{N} \frac{1}{\sigma_\theta^2} \begin{bmatrix} 1 & t_i - t_0 & \frac{1}{2}(t_i - t_0)^2 \\ t_i - t_0 & (t_i - t_0)^2 & \frac{1}{2}(t_i - t_0)^3 \\ \frac{1}{2}(t_i - t_0)^2 & \frac{1}{2}(t_i - t_0)^3 & \frac{1}{4}(t_i - t_0)^4 \end{bmatrix} \tag{3.26}$$

and

$$d = \sum_{i=-n}^{N} \frac{\theta_i}{\sigma_\theta^2} \begin{bmatrix} 1 \\ t_i - t_0 \\ \frac{1}{2}(t_i - t_0)^2 \end{bmatrix}. \tag{3.27}$$

The associated covariance expressing the measurement accuracy of the angles can be obtained by inverting the information matrix, as will be presented later. An example of the modelled attributable from a set of angles in a tracklet is depicted in figs. 3.6 and 3.7. The tracklets correspond to artificially generated measurements of a 100 degree longitude GEO object. The two tracklets are separated by 3 hours, each consisting of 5 angle pairs with 20 second measurement interval.

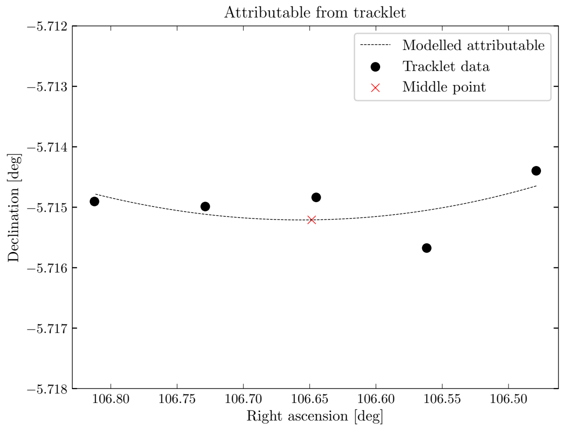


Figure 3.6: Modelled attributable from first tracklet corresponding to a GEO object, middle point indicates the angle and corresponding angle-rate used in attributable.

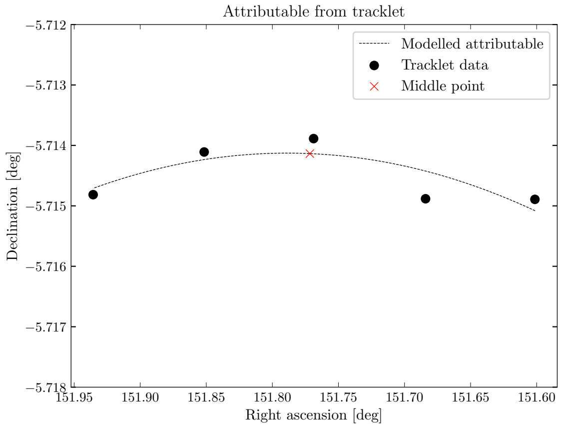


Figure 3.7: Modelled attributable from a second tracklet corresponding to the same GEO object.

The measurement model for the topocentric angle-rates are determined by the line-of-sight vector and its derivative

$$\dot{\boldsymbol{\rho}}_T = \dot{\mathbf{r}} - \dot{\mathbf{R}}. \tag{3.28}$$

From the corresponding topocentric vector components, $x, y, z, \dot{x}, \dot{y}, \dot{z}$, the angle rates are obtained by taking the first derivatives of the right ascension and declination

$$\dot{\alpha} = \frac{d\alpha}{dy}\dot{y} + \frac{d\alpha}{dx}\dot{x} = \frac{x\dot{y} - y\dot{x}}{x^2 + y^2} \tag{3.29}$$

$$\dot{\delta} = \frac{d\delta}{dx}\dot{x} + \frac{d\delta}{dy}\dot{y} + \frac{d\delta}{dz}\dot{z} = \frac{\dot{z}\left(x^2 + y^2\right) - z\left(x\dot{x} + y\dot{y}\right)}{\sqrt{x^2 + y^2}\,r^2} \tag{3.30}$$

4

Statistics & Numerical optimization

This chapter will introduce the notation used for the tracklet correlation approach used in the cataloguing pipeline. It will first cover the main theory on statistics in section 4.1, based on textbook notations found in [2, 63]. Lastly numerical optimization will be briefly introduced and elaborated on for the linear least-squares solution, as used for tracklet processing toward attributables, and the quasi-Newton optimization method for the correlation problem.

4.1. Statistics

The continuous probability distribution of a random (multivariate) variable is described by a density function $f(x)$ as the derivative of the cumulative distribution $F(x)$

$$\frac{dF(x)}{dx} = f(x) \tag{4.1}$$

The cumulative distribution quantifies the probability that the random variable X is below a given value x_u

$$F(x_u) = P(X \leq x_u) = \int_{-\infty}^{x_u} f(x)dx. \tag{4.2}$$

In many statistical applications, the random variable X can be a vector $\mathbf{x}$ of multiple random variables, known as a multivariate random variable. Its joint probability distribution is often modelled by a multivariate probability density function. The multivariate normal distribution describes the probability of a vector $\mathbf{x}$ of n random variables, given its mean $\bar{\mathbf{x}}$ and covariance matrix $C_{\mathbf{x}}$,

$$f(\mathbf{x}) = \frac{1}{2\pi^{n/2}|C_{\mathbf{x}}|^{1/2}} \exp\left(-\frac{1}{2}(\mathbf{x}-\bar{\mathbf{x}})^T C_{\mathbf{x}}^{-1}(\mathbf{x}-\bar{\mathbf{x}})\right). \tag{4.3}$$

The measurement noise, as well as the uncertainty in a state estimate, can be described by a covariance matrix, where the covariance is defined as the second moment about the mean

$$\bar{\mathbf{x}} = E[\mathbf{x}] = \int_{-\infty}^{\infty} \mathbf{x} f(\mathbf{x}) d\mathbf{x}, \tag{4.4}$$

$$C_{\mathbf{x}} = E\left[(\mathbf{x}-\bar{\mathbf{x}})(\mathbf{x}-\bar{\mathbf{x}})^T\right]. \tag{4.5}$$

4.1.1. Bayesian estimation

Bayes' theorem can be used to evaluate the posterior density that describes the probability of a variable $\mathbf{x}$ given a measurement $\mathbf{z}$ and measurement model M,

$$f(\mathbf{x}|\mathbf{z}, M) = \frac{f(\mathbf{z}|\mathbf{x}, M) f(\mathbf{x}, M)}{f(\mathbf{z}, M)}. \tag{4.6}$$

This applies directly to the classification of measurements to objects, i.e. for correlation purposes an object c_i is sought which gives the maximum likelihood for a given measurement $\mathbf{z}$. Assuming that all candidate objects are equally likely and modelled with the same probability, the likelihood is proportional to

$$f(c_i|\mathbf{z}) \propto f(\mathbf{x}|c_i) f(c_i). \tag{4.7}$$

Here, c_i represents the object, and typically its pior state is used to model the measurement $\hat{\mathbf{z}}$. This proportionality expresses the probability of an object given a measurement as the probability distribution of the difference between the reference $\hat{\mathbf{z}}$ and the measured $\mathbf{z}$ variable

$$f(\hat{\mathbf{z}}|\mathbf{z}) = f(\mathbf{z}-\hat{\mathbf{z}}). \tag{4.8}$$

Assuming both the measured and object's modelled parameters are normally distributed with their mean and covariances ($\hat{\mathbf{z}}$, $\mathbf{z}$, $C_{\hat{\mathbf{z}}}$, $C_{\mathbf{z}}$), the probability density function of the difference of the two independent normally distributed random variables is defined as [60, 62]

$$f(\mathbf{z}-\hat{\mathbf{z}}) = \frac{1}{2\pi^{n/2}|C_{\mathbf{z}}+C_{\hat{\mathbf{z}}}|^{1/2}} \exp\left(-\frac{1}{2}(\mathbf{z}-\hat{\mathbf{z}})^T (C_{\mathbf{z}}+C_{\hat{\mathbf{z}}})^{-1}(\mathbf{z}-\hat{\mathbf{z}})\right). \tag{4.9}$$

The object c_i corresponding to the maximum of this probability function denotes the object most likely associated to the measurement

$$\hat{c}_i = \max_{c_i} f(\mathbf{z}-\hat{\mathbf{z}}). \tag{4.10}$$

Equivalently, the negative logarithm can be minimized

$$\hat{c}_i = \min_{c_i} \ln\left(2\pi^{n/2}|C_{\mathbf{z}}+C_{\hat{\mathbf{z}}}|^{1/2}\right) + (\mathbf{z}-\hat{\mathbf{z}})^T (C_{\mathbf{z}}+C_{\hat{\mathbf{z}}})^{-1}(\mathbf{z}-\hat{\mathbf{z}}). \tag{4.11}$$

This function then allows to evaluate the correlation between an object and a measurement or between two measurements. The first term rejects objects and measurements with high uncertainties, the second term is the Mahalanobis distance d^2 [38] as cited in [44]. The object that yields the smallest evaluation of this function is then considered to be the most likely object associated with the measurement. In cases of high uncertainty and almost equal distance d of a measurement to two different objects, it is more appropriate to use only the second term. As the high uncertainty dominates the function in this case, it prevents the small difference between the Mahalanobis distances from being captured and the most likely object from being unambiguously evaluated.

4.1.2. Chi-squared distribution

When assuming the measurement errors are Gaussian distributed, and given the variance from the mean, or reference variable, the probability that this considered measurement $\mathbf{z}$ is associated with $\hat{\mathbf{z}}$ can be assessed. The use of the Mahalanobis distance for this assessment is motivated by its ability to account for statistical significance, particularly when aiming to gate correlations. For example, an association threshold could be expressed by the Euclidean distance of a random variable $\mathbf{z}$ from the mean and its distribution from a reference variable $\hat{\mathbf{z}}$

$$d_{\text{EUC}} = \sqrt{(\mathbf{z}-\hat{\mathbf{z}})^T(\mathbf{z}-\hat{\mathbf{z}})}. \tag{4.12}$$

This approach however does not take into account the covariance, and is in all cases for non-unity covariance not suitable. The Mahalanobis distance, which is scaled by the uncertainty and describes a statistically meaningful distance between the reference $\hat{\mathbf{z}}$ normal distribution and a variable $\mathbf{z}$ in the same space, see fig. 4.1. It is essentially a measure of how many standard deviations (in multivariate domain) the random variable is from the mean of a reference normal distribution. Additionally, the squared Mahalanobis distance has the useful property that it can be related to the multivariate normal distribution with the chi-squared distribution, as plotted in fig. 4.2. The chi-squared distribution for a scalar random variable x is given by the function

$$f(x,n) = \frac{1}{2^{n/2}\Gamma\left(\frac{n_x}{2}\right)} x^{n/2-1} e^{-x/2} \quad \text{for} \quad x \geq 0, \tag{4.13}$$

with Γ a mathematical function of n, the gamma function, and n the degrees of freedom (the amount of variables in $\mathbf{x}$). If the n dimensional random variable $\mathbf{x}$ is normally distributed, the squared Mahalanobis distance d^2 is then distributed by the chi-squared distribution χ_n^2 for n degrees of freedom [2].

$$d^2(\mathbf{x}) = (\mathbf{x}-\bar{\mathbf{x}})^T C_{\mathbf{x}}^{-1}(\mathbf{x}-\bar{\mathbf{x}}) \sim \chi_n^2, \tag{4.14}$$

The chi-squared distribution allows to gate using a single variable, i.e., the Mahalanobis distance, that is dependent on multiple degrees of freedom n. The Mahalanobis distance is used to determine whether the considered random variable is statistically relevant w.r.t. a reference distribution. The cumulative density

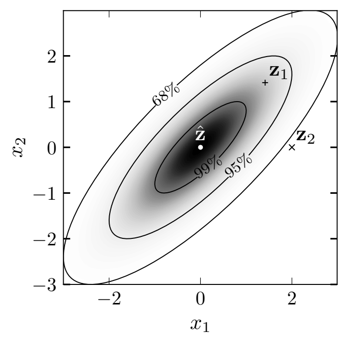


Figure 4.1: Qualitative plot of a bivariate normal distribution with reference variable $\hat{\mathbf{z}}$ and tested variables $\mathbf{z}_1$ and $\mathbf{z}_2$ at equal Euclidean distances. The iso-lines represent constant distance d to $\hat{\mathbf{z}}$.

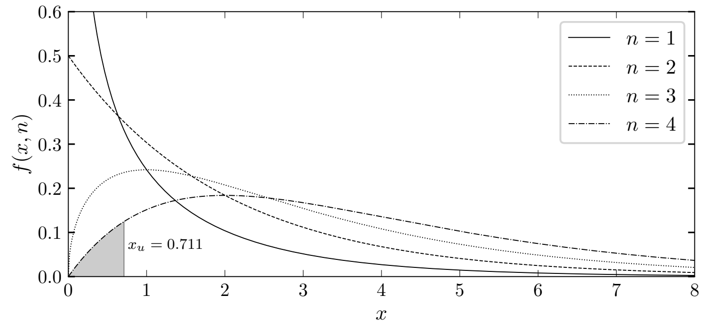


Figure 4.2: Chi-squared distribution for up to $n = 4$ degrees of freedom. Grey region for $n = 4$ shows the threshold for a 5% probability level.

function of the chi-squared distribution then yields the probability P that the squared Mahalanobis distance is less than a set threshold x_u,

$$P\left(d^2 < x_u\right) = F(x_u, n). \tag{4.15}$$

The threshold values are determined by inverting the above function. Numerical integration is used to determine the threshold values for corresponding probability of a n degree of freedom chi-squared distribution. Common probability levels are tabulated in table 4.1. If for example the threshold is set to define association of all random variables within a 5% probability (or 5% true positive rate), the squared distance d^2 should be less than $x_u = 0.711$ for $n = 4$. As will be discussed in chapter 5, the Mahalanobis distance will be used to correlate two tracklets, which is then determined by gating the distance using the desired upper bounds from the chi-squared distribution, as tabulated in table 4.1.

Table 4.1: Gating values (x_u) for different probabilities that a given n dimensional variable is below an upper value x_u of the chi-squared distribution [45].

	Probability $P(x < x_u)$							
n	0.01	0.025	0.05	0.10	0.90	0.95	0.975	0.99
1	0.000	0.001	0.004	0.016	2.706	3.841	5.024	6.635
2	0.020	0.051	0.103	0.211	4.605	5.991	7.378	9.210
3	0.115	0.216	0.352	0.584	6.251	7.815	9.348	11.345
4	0.297	0.484	0.711	1.064	7.779	9.488	11.143	13.277

4.1.3. Uncertainty transformation

When correlating an object c_i to a measurement, the variables and their uncertainty must be described in a common frame. As seen in previous section, this can be done by modelling the measurement from an object's state through some function $\hat{\mathbf{z}} = M(\mathbf{x})$ and thereby performing the association in the measurement space. Corresponding to the transformed state, also normally distributed, the probability density function of the transformed variable can be found through linear approximation of the considered model

$$M(\mathbf{x}) \approx M(\bar{\mathbf{x}}) + J(\mathbf{x} - \bar{\mathbf{x}}) \quad J = \frac{\partial M(\mathbf{x})}{\partial \bar{\mathbf{x}}} \tag{4.16}$$

using the initial mean $\bar{\mathbf{x}}$ and covariance $C_{\mathbf{x}}$

$$\bar{\mathbf{z}} = E[\mathbf{z}] = M(\bar{\mathbf{x}}), \tag{4.17}$$

$$C_{\hat{\mathbf{z}}} = E[(\mathbf{z} - \bar{\mathbf{z}})(\mathbf{z} - \bar{\mathbf{z}})^T] = J C_{\mathbf{x}} J^T. \tag{4.18}$$

This approach highlights the assumption of linear behaviour around the mean state variable. As the correlation is dependent on the statistical distance to a reference variable, and thus the accuracy of the uncertainty representation, this can be a restrictive assumption for non-linear models $M(\mathbf{x})$. If the model to transform the state is non-linear or the initial uncertainty is high, it can result in deformation of the uncertainty in the new space of the initial distribution. In this case, the mean may be well represented in the transformed distribution but the distribution for larger standard deviation is skewed. If the outer bounds are not well transformed, associations may not be realistically captured. While on the contrary, if the distribution of the inner bounds are over-estimated, a lot of false associations are more likely to occur.

Depending on the correlation approach, i.e. initial value or boundary value as will be introduced in chapter 5, the uncertainty is transformed differently and results in different accuracy of the initial representation. This will be discussed further in section 5.4.

4.2. Numerical Optimization

For orbit estimation and observation data processing numerical optimization is commonly employed, as is the case in this work. The main objective of numerical optimization is to define an objective or function $L(\mathbf{x})$ that can be expressed by dependent variables $\mathbf{x}$, which are then required to be optimized and correspond to the optimum function evaluation $L(\hat{\mathbf{x}})$ [48]. Often the function is formulated such that it is minimized

$$\min_{\mathbf{x}} L(\mathbf{x}). \tag{4.19}$$

If no constraints are considered on the function $L(\mathbf{x})$, and the function is continuous and differentiable in a closed and bounded domain, a local minimizer $\hat{\mathbf{x}}$ is said to exist if the optimality conditions are met [48]. The two (first and second order) optimality conditions are defined by

$$\nabla L(\mathbf{x}) = \mathbf{0}, \tag{4.20}$$

$$\Delta\mathbf{x}^T \mathbf{H}(\mathbf{x}) \Delta\mathbf{x} > 0, \quad \mathbf{H}(\mathbf{x}) = \nabla^2 L(\mathbf{x}). \tag{4.21}$$

Where $\mathbf{H}$ is the Hessian matrix and $\Delta\mathbf{x}$ the directional vector in all considered parameters in $\mathbf{x}$. The directional vector $\Delta\mathbf{x}$ follows from linear approximation of the derivatives by the second order Taylor series, i.e.

$$L(\mathbf{x} + \Delta\mathbf{x}) \approx L(\mathbf{x}) + (\nabla L(\mathbf{x}))^T \Delta\mathbf{x} + \frac{1}{2} \Delta\mathbf{x}^T \mathbf{H}(\mathbf{x}) \Delta\mathbf{x} = \hat{L}(\Delta\mathbf{x}). \tag{4.22}$$

4.2.1. Linear least squares

A method or application of optimization useful for data fitting, i.e. use of more than required data points for a best fit, is the least squares fit. If more data points are available then required, the additional data points can be considered as an associated fit that deviates least with the given data points. To acquire such a best fit, the solution minimizing the sum of the squares of the deviation is sought, i.e. the least-squares fit [48]. A solution to the least-squares fit can be obtained for a general linear system of equations (with independent parameters) [63]

$$\mathbf{y} = \mathbf{A}\mathbf{x} + \boldsymbol{\varepsilon}. \tag{4.23}$$

Here $\mathbf{y}$ denotes the vector containing the observed data-points, $\mathbf{A}$ the design matrix, $\mathbf{x}$ the vector containing the unknown parameters to be fitted and $\boldsymbol{\varepsilon}$ the residuals. The vector $\mathbf{x}$, is sought to obtain a minimum sum of the squared residuals

$$\min_{\mathbf{x}} \; L(\mathbf{x}) = \boldsymbol{\varepsilon}^T \boldsymbol{\varepsilon} = \left(\mathbf{y} - \mathbf{A}\mathbf{x}\right)^T \left(\mathbf{y} - \mathbf{A}\mathbf{x}\right). \tag{4.24}$$

Then following the required optimality conditions, the local minimum is said to exist. Accordingly, the first order sufficiency, also known as the normal equation, is obtained as

$$\nabla L(\mathbf{x}) = -\mathbf{A}^T \left(\mathbf{y} - \mathbf{A}\mathbf{x}\right) - \left(\mathbf{y} - \mathbf{A}\mathbf{x}\right)^T \mathbf{A} = \mathbf{0} \quad \Rightarrow \quad \left(\mathbf{A}^T \mathbf{A}\right) \mathbf{x} = \mathbf{A}^T \mathbf{y}, \tag{4.25}$$

where this $\mathbf{x}$ denotes the estimated or fitted parameters indicated as $\hat{\mathbf{x}}$. The second order sufficiency condition, eq. (4.21), is met for

$$\mathbf{H}(\mathbf{x}) = \mathbf{A}^T \mathbf{A} > 0. \tag{4.26}$$

If both conditions are met, the local minimum is found by the least-squares solution from eq. (4.25)

$$\hat{\mathbf{x}} = \left(\mathbf{A}^T \mathbf{A}\right)^{-1} \mathbf{A}^T \mathbf{y}. \tag{4.27}$$

4.2.2. Quasi-newton method

As introduced, optimization is closely related with evaluation of the gradients. Based on the gradient information used in finding the optimum solution the methods are referred to as zeroth, first or second order methods. For the second order method, the local approximation of the cost function at the current step $\mathbf{x}_k$, eq. (4.22), is used with the first order sufficiency condition to determine the step update

$$\nabla \hat{L}(\Delta\mathbf{x}) = \mathbf{0} \quad \Rightarrow \quad \nabla L + \mathbf{H}\Delta\mathbf{x} = \mathbf{0} \quad \Rightarrow \quad \Delta\mathbf{x} = -\mathbf{H}^{-1}\nabla L. \tag{4.28}$$

The cost function is then evaluated, until desired convergence, by consecutive evaluations of

$$\mathbf{x}_{k+1} = \mathbf{x}_k - \mathbf{H}^{-1}\nabla L. \tag{4.29}$$

Theoretically, the second order methods provide the most efficient approach in searching for a minimizer by using an (approximated) second order derivative of the cost function. However, as the system of equations becomes larger (i.e. multi-variable) the second order derivative (Hessian) becomes large and expensive to compute. Quasi-Newton methods offer a robust alternative, by instead approximating the second order derivatives from linear expansion of the first order derivative information

$$\nabla L(\mathbf{x}_{k+1}) = \nabla L(\mathbf{x}_k) + \mathbf{H}\left(\mathbf{x}_{k+1} - \mathbf{x}_k\right) \quad \Rightarrow \quad \hat{\mathbf{H}}\left(\mathbf{x}_{k+1} - \mathbf{x}_k\right) = \nabla L(\mathbf{x}_{k+1}) - \nabla L(\mathbf{x}_k). \tag{4.30}$$

Instead, operating on its inverse avoids solving the linear system, eq. (4.28), at each step, so $\hat{\mathbf{B}} = \hat{\mathbf{H}}^{-1}$ is used.

$$\mathbf{x}_{k+1} - \mathbf{x}_k = \hat{\mathbf{B}}_{k+1}\big(\nabla L(\mathbf{x}_{k+1}) - \nabla L(\mathbf{x}_k)\big) \tag{4.31}$$

Updating the inverse Hessian matrix is achieved as

$$\hat{\mathbf{B}}_{k+1} = \hat{\mathbf{B}}_k + \Delta\hat{\mathbf{B}}_k, \tag{4.32}$$

where the update $\Delta\hat{\mathbf{B}}_k$ can be evaluated in several ways. The tracklet-to-tracklet correlation procedure for the cataloging pipeline, as later described, uses the Broyden [5], Fletcher [17], Goldfarb [19], Shannon [58] (BFGS) update function (as cited in [48]). The BFGS method for updating the inverse Hessian matrix updates is derived and discussed in [46].

4.2.3. Constrained optimization

Often constraints are included in the optimization problem and used to find a bounded optimal solution. The problem of finding a bounded optimum is noted as

$$\min_{\mathbf{x}} L(\mathbf{x}) \quad \text{s.t.} \quad \mathcal{C} = \{\mathbf{x}_l \le \mathbf{x} \le \mathbf{x}_u\} \tag{4.33}$$

The optimality conditions for $L(\mathbf{x})$ in the bounded region $\mathcal{C}$ are given by a boundary optimum in addition to the first, eq. (4.20), and second order sufficiency, eq. (4.21), for local minimum. The boundary optimum is found if

$$\nabla L^T \Delta\mathbf{x} \ge 0, \tag{4.34}$$

i.e. there are no feasible directions $\Delta\mathbf{x}$ for which the objective function L decreases [48].

5

Admissible Region & the cataloguing pipeline

This chapter will introduce the admissible region approach by derivation, and show the developed methods for the tracklet to tracklet initial orbit determination application [64]. This application will be introduced as how it was first developed, using the Initial Value Problem (IVP) method [60]. This will show the key steps and aspects of the method as used for tracklet correlation and initial orbit determination. Having formed an understanding of the developments and approach, the disadvantages and findings of comparison with the alternative Boundary Value Problem (BVP) method are discussed. This allows to introduce and motivate the choice for the BVP method as the cataloguing pipeline, and thereby baseline method. The baseline method is then presented through derivation of the BVP method, and a brief example is shown.

5.1. Admissible Region

The Admissible Region (AR) approach originated from Milani et al. [41], who developed the method to restrict the range and range-rate solution space of asteroids by constraining the asteroid within the solar-system. Further constraints on the semimajor axis and eccentricity, i.e. the Constrained Admissible Region (CAR), were developed by DeMars and Jah for Earth-orbiting satellites [13]. As will be clear, two methods are considered for the admissible region. An initial value problem (IVP) method, where only the information of the initial observation is considered for modelling. And a boundary value problem (BVP) method, in which information of both the initial and last observation are used for modelling. Regardless, both methods require the same information, i.e. two independent attributables $\mathbf{A}_1$, $\mathbf{A}_2$. An attributable is obtained from a tracklet, the procedure is outlined below.

The derivation starts from the optical observations, which provide (see section 3.4.2) an attributable $\mathbf{A} = (\alpha_0, \delta_0, \dot{\alpha}_0, \dot{\delta}_0)$, which represents the middle angles and corresponding angular rates. The attributable is then expressed as a line-of-sight $\mathbf{u}$, eq. (3.18), and its first derivative

$$\dot{\mathbf{u}} = \frac{\partial \mathbf{u}}{\partial \alpha}\dot{\alpha} + \frac{\partial \mathbf{u}}{\partial \delta}\dot{\delta}. \tag{5.1}$$

Together with the satellite-observer-Earth relation, eq. (3.20), the velocity vector is expressed as function of the hypothesised range ρ and range-rate $\dot{\rho}$

$$\dot{\mathbf{r}} = \dot{\mathbf{R}} + \rho\dot{\mathbf{u}} + \dot{\rho}\mathbf{u}. \tag{5.2}$$

Substitution of eqs. (3.20) and (5.2) into the the two-body energy equation

$$\varepsilon = \frac{\|\dot{\mathbf{r}}\|^2}{2} - \frac{\mu_\oplus}{\|\mathbf{r}\|} = -\frac{\mu_\oplus}{2a}, \tag{5.3}$$

yields

$$2\varepsilon = \frac{\mu}{a} = \dot{\rho}^2 + 2\dot{\mathbf{R}}^T\mathbf{u}\dot{\rho} + f(\rho), \tag{5.4}$$

with

$$f(\rho) = \dot{\mathbf{u}}^T \dot{\mathbf{u}} \rho^2 + 2\dot{\mathbf{R}}^2 \dot{\mathbf{u}} \rho + \dot{\mathbf{R}}^T \dot{\mathbf{R}} + \frac{2\mu}{\|\mathbf{R} + \rho \mathbf{u}\|}. \tag{5.5}$$

The resulting equation is a quadratic equation in $\dot{\rho}$ for a given ρ. The roots can be solved given a constant orbital energy ε by

$$\dot{\rho}_{1,2} = \dot{\rho}_v \pm \sqrt{\dot{\rho}_v^2 - f(\rho) - 2\varepsilon} \quad \text{with} \quad \dot{\rho}_v = -\dot{\mathbf{R}}^T \mathbf{u}. \tag{5.6}$$

The solution for given ρ allows definition of the admissible region by a given constant orbital energy. When considering Earth-captured objects, the admissible region for a given attributable **A** and hypothesised range information $\mathbf{p} = [\rho, \dot{\rho}]^T$ is obtained by

$$\mathcal{C}_\varepsilon(\mathbf{p}) = \left\{\mathbf{p} : \varepsilon(\mathbf{p}) < 0\right\}. \tag{5.7}$$

In other words, all considered solutions for range and range-rate are constrained by the constant energy line for Earth-captured orbits $\varepsilon < 0$.

5.1.1. Constrained Admissible Region (CAR)

The admissible region can be further constrained when assuming constraints on the orbit regime such as semi-major axis and eccentricity.

$$\mathcal{C}_a(\mathbf{p}) = \left\{\mathbf{p} : a_{\min} \leq a(\mathbf{p}) \leq a_{\max}\right\} \tag{5.8}$$

Setting a value for the semi-major axis, results in an equivalent energy, as they are related through the standard gravitational parameter, $\mu = -2\varepsilon / a$. Using this equivalent energy, the roots for range-rate given a range can be solved, yielding a curve of constant semi-major axis in the admissible region. Additionally, a constraint on eccentricity can be set

$$\mathcal{C}_e(\mathbf{p}) = \left\{\mathbf{p} : e(\mathbf{p}) \leq e_{\max}\right\}. \tag{5.9}$$

The derivation follows some more rigorous rewriting as presented in [13] considering as the eccentricity can be expressed as:

$$e = \sqrt{1 + \frac{2\varepsilon \|\mathbf{h}\|^2}{\mu^2}}. \tag{5.10}$$

Ultimately, a fourth order quartic equation is obtained depending only on the range-rate, given an attributable and range

$$a_4 \dot{\rho}^4 + a_3 \dot{\rho}^3 + a_2 \dot{\rho}^2 + a_1 \dot{\rho} + a_0 = 0. \tag{5.11}$$

Solving for the real roots of eq. (5.11) yields a constant line of eccentricity in the range - range-rate space. An example following [13] is depicted in fig. 5.1 to illustrate the constrained region.

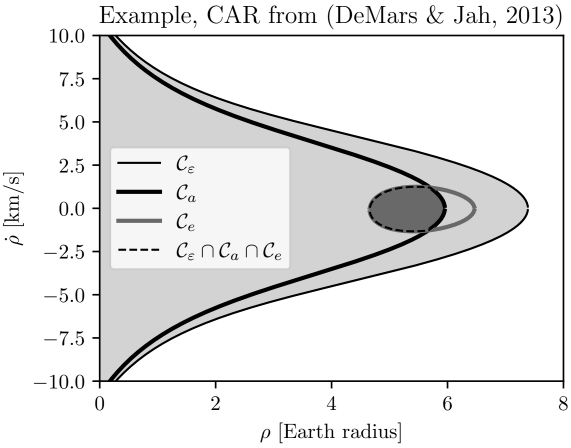


Figure 5.1: Example of a CAR following [13] ($\alpha = 10$ deg, $\delta = -2$ deg, $\dot{\alpha} = 15$ deg/hr, $\dot{\delta} = 3$ deg/hr, 0 deg lat., 60deg. long., $a \leq 50 \cdot 10^3$ km).

5.2. Tracklet-to-tracklet correlation

The origins of the AR method as derived by Tommei et al. [64], and the further constrained CAR Earth-captured objects [13], considered sampling the whole region uniformly. More recently, studies focused on sampling this region by inclusion of uncertainty and possibility. While the AR can be reduced to some degree, especially for IOD purposes it is not computationally viable to discretize an infinite region of feasible (or virtual, VP) points and assess for each sampled VP its association to a follow-up observation. Accordingly, as suggested by Tommei et al. [64], the AR method can be extended by considering two attributables. This allows to constrain the required six parameter IOD problem by using eight independent parameters. Siminski investigated such approaches based on both the IVP and BVP method [59]. Both methods use two independent attributables, of which a subset is used to model expected measurements. The expected measurements can be compared against the true measurements to assess the likelihood of the hypothesized parameters. Specifically, by including the measurement and modelled uncertainty a cost function can be evaluated for a given attributable pair $\mathbf{A}_1$, $\mathbf{A}_2$ given a set of hypothesized parameters $\mathbf{p}$. Combined with an optimization method, the best hypothesized parameters can be found at minimum function evaluations. The procedures for both formulations are depicted in fig. 5.2.

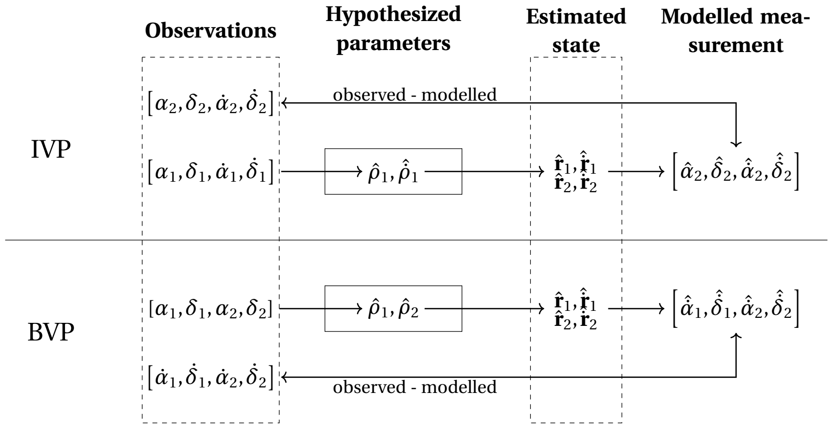


Figure 5.2: Overview of the steps in each approach for the tracklet correlation frame-work considered, figure adapted from Siminski et al. [59]

5.2.1. Initial value problem method

There are generally two approaches to solve this problem using a loss function. The IVP and BVP both make use of a discriminator to compare modelled and measured data obtained by selection of the free parameters. Siminski [60] compared and described both of these approaches in great detail. The difference between IVP and BVP method is in the state and discriminator used. The objective function to be minimised for given set of attributables in the IVP method is defined as

$$L_{\text{IVP}} = (\mathbf{z} - \hat{\mathbf{z}})^T (C_{\mathbf{z}} + C_{\hat{\mathbf{z}}})^{-1} (\mathbf{z} - \hat{\mathbf{z}}), \tag{5.12}$$

where $\mathbf{z}$ denotes the measured state, $\hat{\mathbf{z}}$ the modelled state, and C the covariance of the modelled or measured state. The subscript indicate the corresponding attributable, where $\mathbf{A}_1$ is acquired by a time ΔT earlier than $\mathbf{A}_2$. The cost function is derived in section 4.1, and effectively represents the maximum likelihood of the free parameter set given the measured angles. It represents the Mahalanobis distance from a normal distribution.

For the IVP the measured and modelled states take shape as

$$\mathbf{z} = \left[\alpha_2, \delta_2, \dot{\alpha}_2, \dot{\delta}_2\right]^T \quad \text{and} \quad \hat{\mathbf{z}} = \left[\hat{\alpha}_2, \hat{\delta}_2, \hat{\dot{\alpha}}_2, \hat{\dot{\delta}}_2\right]^T. \tag{5.13}$$

The IVP method uses the initial attributable and propagates it to the second observation epoch to model the second attributable and use it as discriminator. The resulting AR is in range ρ - range-rate $\dot{\rho}$ space, forming the six required independent parameters $\left[\alpha_1, \delta_1, \dot{\alpha}_1, \dot{\delta}_1, \rho_1, \dot{\rho}_1\right]^T$. Considering the CAR derived in previous sections, the free parameters $\mathbf{p} = \left[\rho_1, \dot{\rho}_1\right]^T$ can be sampled and used to acquire the orbital state using eqs. (3.20) and (5.2). The state vectors and observer position then allow to model the topocentric measurement angles and angle rates, $\hat{\mathbf{z}}$. As for the covariance matrices, the information matrix obtained from the least squares fit is inverted to obtain the deviations in the angles and angle rates at the middle tracklet epoch. The covariance matrix can then be populated with the diagonal terms of the inverse

$$C_{\mathbf{z}} = \begin{bmatrix} \sigma^2_{\alpha_2} & 0 & 0 & 0 \\ 0 & \sigma^2_{\delta_2} & 0 & 0 \\ 0 & 0 & \sigma^2_{\dot{\alpha}_2} & 0 \\ 0 & 0 & 0 & \sigma^2_{\dot{\delta}_2} \end{bmatrix}. \tag{5.14}$$

Still, the covariance for the modelled state $C_{\hat{\mathbf{z}}}$ is required and is obtained by transformation of $C_{\mathbf{z}}$.

$$C_{\hat{\mathbf{z}}} = \left(\frac{\partial \hat{\mathbf{z}}(\mathbf{z}, \mathbf{p})}{\partial \mathbf{z}}\right) C_{\mathbf{z}} \left(\frac{\partial \hat{\mathbf{z}}(\mathbf{z}, \mathbf{p})}{\partial \mathbf{z}}\right)^T \tag{5.15}$$

So, from two tracklets T_1 and T_2 at different observation epochs attributables are derived including the angle deviations. The IVP method then compares the modelled second attributable $\mathbf{A}_2$ with the measured second attributable $\hat{\mathbf{A}}_2 = (\hat{\alpha}_2, \hat{\delta}_2, \hat{\dot{\alpha}}_2, \hat{\dot{\delta}}_2)$ based on the initial conditions, i.e. the first measured attributable $\mathbf{A}_1$.

5.3. Cataloguing pipeline / BVP

The boundary value approach uses the angles of both attributables to model the rates and evaluate the likelihood of association between the attributables. The cost function is defined similarly as for the IVP method, as the squared residuals normalized by the uncertainty

$$L_{\text{BVP}} = (\dot{\mathbf{z}} - \hat{\dot{\mathbf{z}}})^T \left(C_{\dot{\mathbf{z}}} + C_{\hat{\dot{\mathbf{z}}}}\right)^{-1} (\dot{\mathbf{z}} - \hat{\dot{\mathbf{z}}}). \tag{5.16}$$

The measured and modelled states are composed as

$$\mathbf{z} = (\dot{\alpha}_1, \dot{\delta}_1, \dot{\alpha}_2, \dot{\delta}_2) \quad \text{and} \quad \hat{\mathbf{z}} = (\hat{\dot{\alpha}}_1, \hat{\dot{\delta}}_1, \hat{\dot{\alpha}}_2, \hat{\dot{\delta}}_2). \tag{5.17}$$

This approach of using the attributables requires to extend the attributable by the outer ranges ρ_1 and ρ_2, forming the six independent parameters as $\left[\alpha_1, \delta_1, \alpha_2, \delta_2, \rho_1, \rho_2\right]^T$. Accordingly, the CAR is in the range ρ_1 - range ρ_2 space. Then using the Earth-satellite-observer relations, the Lambert's problem can be solved given the two outer position vectors and the time of flight. However, to solve Lambert's problem completely and thus obtain the velocity vectors at both epochs $\dot{\mathbf{r}}_1$ and $\dot{\mathbf{r}}_2$, the number of completed orbital revolutions k, the direction of flight and the correct (low or high path) solution must be known. As described in section 3.3.4, at most four different orbits can be found, so each parameter and resulting orbit must be considered to define the orbit most likely to correspond to the measurements. Then, similar to the IVP method, the measured state $\mathbf{z}$ is modelled given the two outer states.

The covariance matrices $C_{\mathbf{z}}$ and $C_{\dot{\mathbf{z}}}$ are derived from the covariance matrices obtained through the least squares fit for each tracklet. These covariance matrices are computed by inverting the information matrix, $P = \Lambda^{-1}$. Once the attributables for both tracklets T_1, T_2 are determined, the covariance matrices are populated with the relevant diagonal terms corresponding to the angles and angle rates.

$$C_{\mathbf{z}} = \begin{bmatrix} \sigma^2_{\alpha_1} & 0 & 0 & 0 \\ 0 & \sigma^2_{\delta_1} & 0 & 0 \\ 0 & 0 & \sigma^2_{\alpha_2} & 0 \\ 0 & 0 & 0 & \sigma^2_{\delta_2} \end{bmatrix} \quad \text{and} \quad C_{\dot{\mathbf{z}}} = \begin{bmatrix} \sigma^2_{\dot{\alpha}_1} & 0 & 0 & 0 \\ 0 & \sigma^2_{\dot{\delta}_1} & 0 & 0 \\ 0 & 0 & \sigma^2_{\dot{\alpha}_2} & 0 \\ 0 & 0 & 0 & \sigma^2_{\dot{\delta}_2} \end{bmatrix}. \tag{5.18}$$

The deviations represent the uncertainty in the attributable, or average deviation based on the tracklet measurements. The covariance of the modelled attributable $\hat{\dot{\mathbf{z}}}$, is obtained by mapping the covariance in the measured angles $C_{\mathbf{z}}$ into the modelled angle rates covariance by linear approximation following section 4.1.3 [59].

$$C_{\hat{\dot{\mathbf{z}}}} = \left(\frac{\partial \hat{\dot{\mathbf{z}}}}{\partial \mathbf{z}}\right) C_{\mathbf{z}} \left(\frac{\partial \hat{\dot{\mathbf{z}}}}{\partial \mathbf{z}}\right)^T. \tag{5.19}$$

The Jacobian $\left(\frac{\partial \hat{\mathbf{z}}}{\partial \mathbf{z}}\right)$ is evaluated using central finite differencing with a step size $h = 10^{-8}$. Each row of the Jacobian is evaluated similarly to the first row, given by:

$$\frac{\partial \hat{\mathbf{z}}(\mathbf{z}, \mathbf{p}, k)}{\partial \alpha_1} \approx \frac{\hat{\mathbf{z}}\left([\alpha_1 + h, \delta_1, \alpha_2, \delta_2]^T, \mathbf{p}, k\right) - \hat{\mathbf{z}}\left([\alpha_1 + h, \delta_1, \alpha_2, \delta_2]^T, \mathbf{p}, k\right)}{2h}. \tag{5.20}$$

Altogether, the cost function can be evaluated to assess the likelihood two measured attributables are associated. In contrast to the IVP method, the BVP method obtains separate CARs for different number of revolutions. This is beneficial for optimization, as the topography often contains a single minimum for each k, and is characterised by smooth surfaces [60]. Finding the free parameters $\mathbf{p} = [\rho_1, \rho_2]^T$ minimizing the cost function allows to assess association of the two attributables, which is determined by a threshold on the cost function.

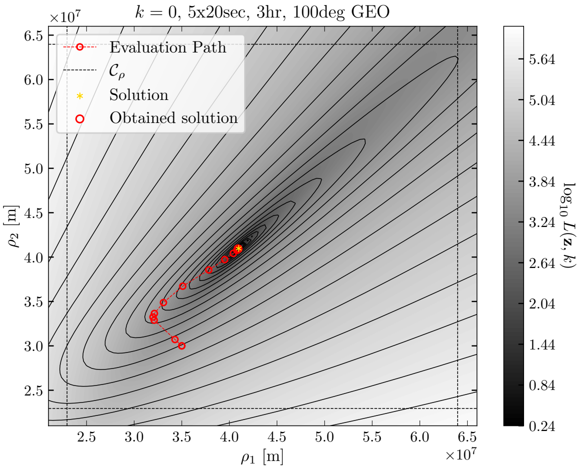


Figure 5.3: Topograhy and optimization evaluation path for an example GEO case. GEO object for simulated tracklets of 5 measurements taken at time interval of 20s with 2" noise. Revisit time of 3 hours for a 42,164 km 100 deg latitude GEO satellite. Observer located at 30 degree longitude, 40 degree latitude and 0 m altitude.

5.3.1. CAR for BVP

For the BVP approach, Schumacher et al. [56] derived the constraints in the range - range AR. The method considers constraining the feasible range space by a maximum and minimum semi-major axis and maximum eccentricity. Accordingly, maximum and minimum values can be set bounding the AR. This follows from the requirement on the position vector's magnitude to be greater than the orbit's minimum expected perigee $r_p = a(1-e)$ but smaller than its expected maximum apogee $r_a = a(1+e)$

$$(a_{\min}(1-e_{\min}))^2 \leq |\mathbf{r}|^2 \leq (a_{\max}(1+e_{\max}))^2. \tag{5.21}$$

With the Earth-satellite-observer relation and squaring the terms,

$$a_{\min}^2 (1-e_{\min})^2 \leq \mathbf{R}\cdot\mathbf{R} + 2(\mathbf{R}\cdot\mathbf{u})\rho + \rho^2 \leq a_{\max}^2 (1+e_{\max})^2 \tag{5.22}$$

the equations for the bounding ranges are obtained by

$$\rho_{\min} = -(\mathbf{R}\cdot\mathbf{u}) + \sqrt{(\mathbf{R}\cdot\mathbf{u})^2 + a_{\min}^2 (1-e_{\max})^2 - \mathbf{R}\cdot\mathbf{R}}, \tag{5.23}$$

$$\rho_{\max} = -(\mathbf{R}\cdot\mathbf{u}) + \sqrt{(\mathbf{R}\cdot\mathbf{u})^2 + a_{\max}^2 (1+e_{\max})^2 - \mathbf{R}\cdot\mathbf{R}}. \tag{5.24}$$

Which define the bounding rectangular box in range - range space as

$$\mathcal{C}_\rho(\mathbf{R}, \mathbf{u}) = \left\{\mathbf{p} : \rho_{\min} \leq \mathbf{p} \leq \rho_{\max}\right\}. \tag{5.25}$$

Additionally, the solution of the Lambert's problem is bounded to a minimum and maximum number of expected revolutions k. This is defined by assuming a circular orbit and Kepler's third law [59]

$$k_{\min/\max} = \frac{\Delta t}{P(a_{\min/\max})}. \tag{5.26}$$

Here Δt is the time between tracklets, P the orbital period and a the minimum or maximum semi-major axis. The initial guess $\boldsymbol{\rho}^*$ is based on the assumption of a circular orbit given each k, and time interval between tracklets Δt

$$\boldsymbol{\rho}^* = -(\mathbf{R}\cdot\mathbf{u}) + \sqrt{(\mathbf{R}\cdot\mathbf{u})^2 + a(k)^2 - \mathbf{R}\cdot\mathbf{R}}. \tag{5.27}$$

It should be noted that this method only works for non-zero values of k. Alternatively, a stable initial guess is to set the average semi-major axis of the corresponding bounds. For large range of semi-major axis bounds however, such as the MEO regime, a low initial guess for the ranges was seen to yield better convergence and stability using the minimum semi-major axis.

5.3.2. Optimization

The cost function can be minimized given the Lambert's solver parameters, number of orbital half-revolutions, direction of flight and low or high path solution if multiple exist (see section 3.3.4). This effectively means there can be multiple minima per correlation case. Each minimizer is acquired by utilizing the Broyden-Fletcher-Goldfarb-Shannon (BFGS) quasi-Newton method, as considered for the BVP method when presented by Siminski [60]. The BFGS allows for fast convergence given a topography close to quadratic, and is well-suited for multi-variable optimization. The limitations of this method however is shared with all other quasi-Newton methods in that it uses dense matrices for the inverse Hessian update that are expensive to store and manipulate [46]. Often this method is implemented by simplifying the matrix representation, limiting memory usage, accordingly L-BFGS-B [6], a limited-memory bounded quasi-Newton method is considered in this work. In addition, the Powell zero-order line search method is considered for flat topography, where the quasi-Newton method is expected to struggle given an approximate gradient. By searching directly along adaptive directions without relying on gradient information, the line-search method is expected to maintain robust performance and convergence, thereby equipping the pipeline with a robust fallback optimization method.

5.4. Uncertainty transformation

As introduced in section 4.1.3, the uncertainty transform is required for adequate comparison and differs per approach. Comparison of the two formulations, IVP and BVP, and their uncertainty transforms showed the transform of the boundary value method to be more accurately maintained [60]. An inaccurately transformed sample point can negatively affect estimation performance; for example, a point with high statistical significance could be misrepresented after transformation, resulting in an unrealistic distance. This, in turn, impacts the gating performance based on the Mahalanobis distance, as the theoretical distribution becomes distorted. The boundary value method transforms the uncertainty using only the measured angles and modelled angular rates, the uncertainty of the modelled angular rates is negligible compared to the angle noise and therefore better represented by a normal distribution. This in turn allows for more straightforward calibration with the theoretical chi-squared distribution for gating. As for the initial value method it requires more effort for transforming the initial uncertainty to the second tracklet epoch and uses the relatively large uncertainties in the angular-rates.

To overcome this issue for the initial value problem, either alternative transformation methods or different comparison spaces can be implemented to reduce this effect. For example DeMars and Jah implemented a Gaussian mixture model and showed that it captured the skewed uncertainty transform more accurately, leading to more truthful correlation [13]. Alternatively, it is possible to choose a different comparison space in which the transformation behaves (more) linearly. Cai et al. showed improvement in the performance by comparing the orbital states in non-singular canonical space and produced an increase in the true positive rate compared to a linearly approximated uncertainty transform in the measurement space [8].

In addition, the topography for the initial value problem method includes many local minima corresponding to the feasible number of orbital revolutions k [61]. This requires segmenting the local minima as separate domains, resulting in a less straight-forward minimization approach. The boundary value method obtains separate topography, considering the orbital revolutions in separate minimization problems [59].

While the boundary value method is less straight-forward when including perturbations, and expected to be less computationally efficient for perturbed motion, it was concluded in [60] that in most cases two-body dynamics allowed for sufficiently accurate estimates. This behaviour motivates the use of the boundary value method approach.

5.4.1. Lambert Solver

As mentioned, the BVP method relies on finding the correct hypothesized ranges, which together with the angle measurements form the position vectors. It compares the modelled and measured angle-rates, of which the latter is achieved through the full estimated states from a Lambert solver's velocity vectors. The inputs of the Lambert solver are thus two outer-position vectors $\mathbf{r}_1$ and $\mathbf{r}_2$ and the time-of-flight between the position vectors τ. Given this information, the Lambert solver solves for the corresponding velocity vectors $\dot{\mathbf{r}}_1$ and $\dot{\mathbf{r}}_2$, obtaining the full states. Additionally, as shown in section 3.3.4, there can be multiple solutions to Lambert's problem depending on the provided position vectors and time-of-flight. Accordingly, modern Lambert solvers allow to consider all possible answers and allow to handle multiple revolutions given the flight direction and the path type.

For the implementation of the BVP method, the Lambert solver of Izzo [30] is considered, obtained through Garrido 's Python implementation [18]. This implementation shows comparable accuracy at shorter evaluation times to Gooding's Lambert solver [21, 30, 18]. Modern Lambert solvers solve the time-of-flight equation using iterative schemes, typically iterating over a universal variable that describes the orbit geometry corresponding to the provided Lambert parameters, such as the time of flight, direction of motion (prograde or retrograde), solution type (high or low path), number of complete revolutions, and position vectors. The Lambert solver implementation of Izzo introduces a new Lambert invariant variable ξ, a transformation of Gooding's variable x. Following the derivation, the time-of-flight function

$$T = \frac{1}{1-x^2}\left(\frac{\psi + n_{\text{rev}}\pi}{\sqrt{|1-x^2|}} - x + \lambda y\right) \tag{5.28}$$

is solved using a Householder iterative scheme to find the corresponding roots x. This formulation, similar to that shown in section 3.3.4, presents the time of flight as function of the problems' geometry s, c and $r_1 + r_2$. Where the variables y, λ and ψ are auxiliary variables that partially contribute to obtaining the single time of flight function that is valid for both elliptic and hyperbolic orbits and are only dependent on the problem geometry. Additionally, multiple orbital revolutions are accounted for given the true number of orbital revolutions n_{rev}.

Izzo solves aforementioned time of flight function specifically for a transformed variable ξ,

$$\xi = \begin{cases} \log(1+x) & \text{for} \quad n_{\text{rev}} = 0 \\ \log\left(\frac{1+x}{1-x}\right) & \text{for} \quad n_{\text{rev}} > 0, \end{cases} \tag{5.29}$$

with the time-of-flight as $\tau = \log(T)$. The corresponding time-of-flight function can then be solved for appropriate input conditions, as shown in fig. 5.4 for the more easily interpreted untransformed variable. The figures show, for LEO, MEO and GEO orbits, the solution obtained for the iterative variable x given an orbital period of about 1 day, the corresponding outer position vectors and number of completed orbital revolutions. The LEO case considers retrograde motion, while MEO and GEO consider prograde motion. Consequently, each plot shows the two remaining possible solutions: the low or high path.

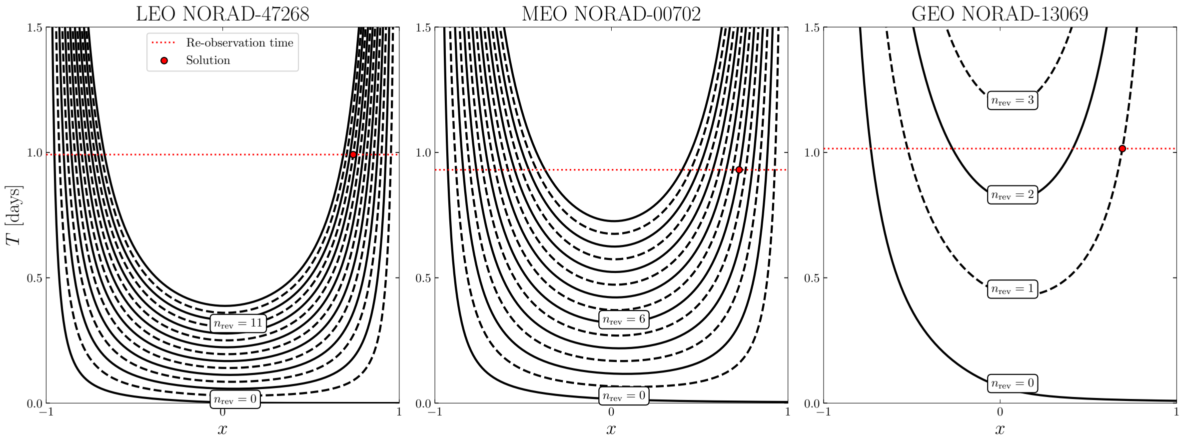


Figure 5.4: Untransformed Time-of-flight T for multiple revolutions and orbit geometry λ against x. Here showing different objects with around 1 day revisited observation. Figure adapted from poliastro's Python library [53].

6

Estimation Methods

The aim of this chapter is to provide the derivation of the (initial) orbit determination methods used for the modification of the cataloguing pipeline in addition to the aforementioned AR boundary value method. First, the three angles-only Gauss classical IOD method is derived as it is the basis for the multiple angles only Least-squares method (L_n) as will be described after. Second, the multi-revolutions three-angles Gooding method is described. Lastly, an orbit determination method is introduced known as the Batch Least Squares (BLS) method. Both the L_n and BLS method will be used to test their implication on correlation for tracklets.

6.1. Gauss angles-only

The classical method following the BVP formulation, Gauss, assumes that the satellite's orbit lies in a plane with the center of the Earth for unperturbed two-body motion. The derivation here, follows the approach from Curtis [12]. As will be seen later for the multiple angles least squares method (L_n), it is very similar to Gauss' method in the way that it uses the Lagrange coefficients. Accordingly, both methods' accuracy depend on the method used to determine the Lagrange coefficients f and g. Gauss' technique provides a robust way of estimating orbits using closely separated observations, less than 10° [66]. It starts with the assumption that the three position vectors corresponding to the observations lie in the same plane, and thus can be written as linear combinations of each other

$$c_1\mathbf{r}_1 + c_2\mathbf{r}_2 + c_3\mathbf{r}_3 = \mathbf{0}. \tag{6.1}$$

Using the Lagrange coefficients f and g, introduced in section 3.3.2, we can express the outer position vectors as a combination of the mid-point position and velocity vector

$$\mathbf{r}_i = f_i\mathbf{r}_2 + g_i\mathbf{v}_2, i = 1,3. \tag{6.2}$$

Together with the cross product of $\mathbf{r}_1$ and $\mathbf{r}_2$ with eq. (6.1),

$$\mathbf{r}_1 \times \mathbf{r}_3 \cdot c_1 = \mathbf{r}_2 \times \mathbf{r}_3 \cdot -c_2 \qquad \mathbf{r}_1 \times \mathbf{r}_3 \cdot c_3 = \mathbf{r}_1 \times \mathbf{r}_2 \cdot -c_2, \tag{6.3}$$

and letting $c_2 = -1$ to simplify the system of equations, the coefficients can be solved for:

$$c_1 = \frac{\mathbf{r}_2 \times (f_3\mathbf{r}_2 + g_3\mathbf{v}_2)}{(f_1\mathbf{r}_2 + g_1\mathbf{v}_2) \times (f_3\mathbf{r}_2 + g_3\mathbf{v}_2)} = \frac{g_3}{f_1 g_3 - f_3 g_1} \tag{6.4}$$

$$c_3 = \frac{(f_1\mathbf{r}_2 + g_1\mathbf{v}_2) \times \mathbf{r}_2}{(f_1\mathbf{r}_2 + g_1\mathbf{v}_2) \times (f_3\mathbf{r}_2 + g_3\mathbf{v}_2)} = \frac{-g_1}{f_1 g_3 - f_3 g_1}. \tag{6.5}$$

If we know the mid-point velocity and position vectors we can determine the Lagrange coefficients and vice-versa. Naturally, at this point we do not yet know either. Accordingly we may determine f and g by their series form. Written out up to the fourth-order we have

$$\begin{aligned} f_{(4)} &= 1 - \tfrac{u}{2}\tau_i^2 - \tfrac{u'}{6}\tau_i^3 - \tfrac{u''-u^2}{24}\tau_i^4 + \mathcal{O}(\tau_i^5) \\ g_{(4)} &= \tau_i - \tfrac{u}{6}\tau_i^3 - \tfrac{u'}{12}\tau_i^4 + \mathcal{O}(\tau_i^5), \qquad i = 1,3 \end{aligned} \tag{6.6}$$

where $\tau_i = t_i - t_2$, $u = \frac{\mu}{r_2^3}$ and $\mathcal{O}(\tau^5)$ denotes the truncated higher order terms. Substituting the terms of the series (only the first two are known), it is thus possible to approximate c_1 and c_3 without mid-point position and velocity vectors by

$$c_1 = \frac{g_3}{f_1 g_3 - f_3 g_1} \approx \frac{\tau_3}{\tau_3 - \tau_1} + \frac{u\tau_3\left((\tau_3-\tau_1)^2 - \tau_3^2\right)}{6(\tau_3-\tau_1)} = (a_1 + a_{1u}u) \tag{6.7}$$

$$c_3 = \frac{-g_1}{f_1 g_3 - f_3 g_1} \approx -\frac{\tau_1}{\tau_3 - \tau_1} - \frac{u\tau_1\left((\tau_3-\tau_1)^2 - \tau_1^2\right)}{6(\tau_3-\tau_1)} = (a_3 + a_{3u}u). \tag{6.8}$$

Given the (approximated) coefficients c_i, we may acquire the slant ranges ρ_i by rewriting eq. (6.1) with

$$\mathbf{r}_i = \rho_i \boldsymbol{u}_i + \mathbf{R}_i, \tag{6.9}$$

with $\mathbf{R}_i$ the observer location, ρ_i the slant range and $\boldsymbol{u}_i$ the measured unit position vector at the i^{th} epoch. This yields

$$c_1(\rho_1\boldsymbol{u}_1 + \mathbf{R}_1) + c_2(\rho_2\boldsymbol{u}_2 + \mathbf{R}_2) + c_3(\rho_3\boldsymbol{u}_3 + \mathbf{R}_3) = \mathbf{0}, \tag{6.10}$$

Rewriting to obtain

$$c_1\rho_1\boldsymbol{u}_1 + c_2\rho_2\boldsymbol{u}_2 + c_3\rho_3\boldsymbol{u}_3 = -c_1\mathbf{R}_1 - c_2\mathbf{R}_2 - c_3\mathbf{R}_3, \tag{6.11}$$

allows to write the equation in matrix form

$$\begin{bmatrix}\boldsymbol{u}_1 & \boldsymbol{u}_2 & \boldsymbol{u}_3\end{bmatrix}\begin{bmatrix}c_1\rho_1\\ c_2\rho_2\\ c_3\rho_3\end{bmatrix} = \begin{bmatrix}\mathbf{R}_1 & \mathbf{R}_2 & \mathbf{R}_3\end{bmatrix}\begin{bmatrix}-c_1\\ -c_2\\ -c_3\end{bmatrix}. \tag{6.12}$$

Solving this equation for the slant-ranges ρ_i we need to invert the unit-vector matrix. Let $\mathbf{L} = [\boldsymbol{u}_1 \;\; \boldsymbol{u}_2 \;\; \boldsymbol{u}_3]^T$, then

$$\begin{bmatrix}c_1\rho_1\\ c_2\rho_2\\ c_3\rho_3\end{bmatrix} = \mathbf{L}^{-1}\begin{bmatrix}\mathbf{R}_1 & \mathbf{R}_2 & \mathbf{R}_3\end{bmatrix}\begin{bmatrix}-c_1\\ -c_2\\ -c_3\end{bmatrix}. \tag{6.13}$$

With eqs. (6.7) and (6.8) and recalling $c_2 = -1$, obtains

$$\begin{bmatrix}c_1\rho_1\\ -\rho_2\\ c_3\rho_3\end{bmatrix} = \mathbf{M}\begin{bmatrix}-(a_1 + a_{1u}u)\\ 1\\ -(a_3 + a_{3u}u)\end{bmatrix} \tag{6.14}$$

where $\mathbf{M} = \mathbf{L}^{-1}[\mathbf{R}_1 \;\; \mathbf{R}_2 \;\; \mathbf{R}_3]$, a size [3x3] matrix. Letting each element of $\mathbf{M}$ be denoted by M_{ij} for the i^{th} row and j^{th} column, we can evaluate ρ_2 with the previous results of c_1, c_3 and remembering we set $c_2 = -1$. Accordingly, we obtain

$$\rho_2 = M_{21}c_1 - M_{22} + M_{23}c_3. \tag{6.15}$$

This can be written in a more convenient form by considering the separate terms in the coefficients c_1, c_3 and defining $d_1 = M_{21}a_1 - M_{22} + M_{23}a_3$ and $d_2 = M_{21}a_{1u} + M_{23}a_{3u}$

$$\rho_2 = d_1 + d_2 u. \tag{6.16}$$

The mid-point range r_2 can be determined by rewriting eq. (6.16) with

$$r_2^2 = \rho_2^2 + 2\rho_2\boldsymbol{\rho}_2 \cdot \mathbf{R}_2 + R_2^2 \tag{6.17}$$

and defining $C = \hat{\boldsymbol{\rho}}_2 \cdot \mathbf{R}_2$, to obtain

$$-r_2^2 + \left(d_1^2 + 2Cd_1 + \mathbf{R}_2^2\right) + 2u(Cd_2 + d_1d_2) + d_2^2u^2 = 0. \tag{6.18}$$

Finally, substituting $u = \frac{\mu}{r_2^3}$ and multiplying with $-r_2^6$ yields

$$r_2^8 - \left(d_1^2 + 2Cd_1 + R_2^2\right)r_2^6 - 2\mu(Cd_2 + d_1d_2)\,r_2^3 - \mu d_2^2 = 0. \tag{6.19}$$

The appropriate roots of the 8th order equation in r_2 allow to estimate r_2 and update u for the Lagrange coefficients f and g. Then all three position vectors are determined with the coefficients c_i and solving the full matrix equation. The acquired position vectors from the preceding steps can be considered the 'conventional' solution of Gauss' angles-only method.

6.1.1. Refined Gauss

The result obtained by Gauss, i.e. the three position vectors $\mathbf{r}_1, \mathbf{r}_2, \mathbf{r}_3$, may be refined by determining the middle velocity vector through a Lambert solver. The steps associated with this refinement will be referred to as 'refined Gauss'. Starting from the three position vectors and the previously described middle velocity vector, the Lagrange coefficients f and g are reevaluated. Here, either the series coefficients or the function coefficients can be used. The series coefficients can be reevaluated with an additional derivative term

$$u' = \frac{-3\mu \dot{r}_{2,\text{rad}}}{r_2^4}. \tag{6.20}$$

As shown by the series in eq. (6.6), f and g can now be approximated by three terms. Accordingly, the coefficients c_1 and c_3 are approximated with higher accuracy and are used to update the initial result of Gauss.

Alternatively, Prussing and Conway ([39] as cited in [50]), showed the exact expressions for the Lagrange coefficients can be used instead, avoiding the smaller convergence region associated with the series coefficients. The exact functions of f and g however, only allow to determine a more exact f and g when compared to the truncated series solution. This is because it uses the initial orbit as obtained by the series approximated Lagrange coefficients [50]. The functions for the Lagrange coefficients can be determined using the universal formulation of Kepler's equation, as is considered in this work. The universal variables allows to use common universal elements independent of orbit type. Thus when estimating the object's position, its trajectory following Kepler's equation, does not require separate approaches for hyperbolic or parabolic trajectories.

$$\begin{aligned} f &= 1 - \frac{\chi^2}{r_i} c_2\left(\alpha\chi^2\right) \\ g &= \tau_i - \frac{1}{\sqrt{\mu}} \chi^3 c_3\left(\alpha\chi^2\right) \end{aligned} \tag{6.21}$$

where χ denotes the universal anomaly, α the reciprocal of the semimajor axis a, c_2 and c_3 the Stumpff functions and μ the standard gravitational parameter. The Stumpff functions are defined by

$$c_2(z) = \begin{cases} \frac{\sqrt{z} - \sin\sqrt{z}}{(\sqrt{z})^3} & (z > 0) \\ \frac{\sinh\sqrt{-z} - \sqrt{-z}}{(\sqrt{-z})^3} & (z < 0) \quad \left(z = \alpha\chi^2\right) \\ \frac{1}{6} & (z = 0) \end{cases} \tag{6.22}$$

$$c_3(z) = \begin{cases} \frac{1 - \cos\sqrt{z}}{z} & (z > 0) \\ \frac{\cosh\sqrt{-z} - 1}{-z} & (z < 0) \quad \left(z = \alpha\chi^2\right) \\ \frac{1}{2} & (z = 0) \end{cases} \tag{6.23}$$

The refined Gauss solution is obtained by repeating either the series or function procedure until the range estimates converge.

6.1.2. Gooding

Gooding's method uses three lines of sight and their epochs to acquire the estimated orbital elements. The initial assumptions of ρ_1 and ρ_3 are updated using Newton-Raphson iteration, which continues until convergence. Specifically, Gooding makes use of an auxiliary plane perpendicular to the middle line-of-sight, $\hat{\mathbf{u}}_2$. The origin of this plane is defined by the point where $\hat{\mathbf{u}}_2$ intersects the auxiliary plane, when extended to an appropriate magnitude. The estimate of the middle position vector, $\mathbf{r}_2$, then describes the point (f, g) on this plane, see fig. 6.1. Note that the coordinates f and g are arbitrary quantities used to describe a location on the auxiliary plane, and are not related to the Lagrange coefficients. This estimate is acquired through the outer states (at t_1 and t_3) obtained by solving Lambert's problem, provided the initial assumptions are accurate (e.g. ρ_1, ρ_3 number of completed orbital revolutions & flight direction). The coordinates are defined such that $g = 0$ and $f > 0$. The coordinates are thus the target function. The target functions are used to determine the appropriate increments, $\Delta\rho_1$ and $\Delta\rho_3$, for the initial range estimates, ρ_1 and ρ_3, using the Newton-Raphson procedure:

$$\Delta\mathbf{x} = -\mathbf{J}(\mathbf{x})^{-1}\mathbf{f}(\mathbf{x}) \quad \rightarrow \quad \begin{bmatrix} \Delta\rho_1 \\ \Delta\rho_3 \end{bmatrix} = -\begin{bmatrix} \frac{\partial f}{\partial \rho_1} & \frac{\partial f}{\partial \rho_3} \\ \frac{\partial g}{\partial \rho_1} & \frac{\partial g}{\partial \rho_3} \end{bmatrix}^{-1} \begin{bmatrix} f \\ g \end{bmatrix} \tag{6.24}$$

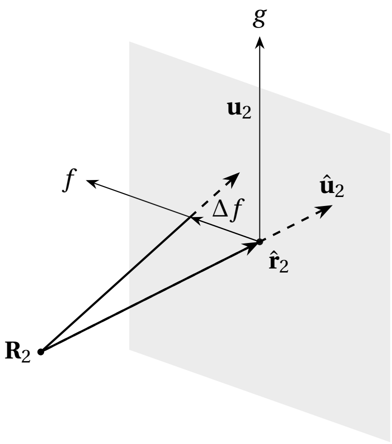


Figure 6.1: Error plane used in Gooding for the target functions [20].

Since $g = 0$ the increments are defined as:

$$\begin{aligned}\Delta\rho_1 &= -\frac{1}{\frac{\partial f}{\partial\rho_1}\frac{\partial g}{\partial\rho_3}-\frac{\partial f}{\partial\rho_3}\frac{\partial g}{\partial\rho_1}}\left(\frac{\partial g}{\partial\rho_3}\right)f\\ \Delta\rho_3 &= -\frac{1}{\frac{\partial f}{\partial\rho_1}\frac{\partial g}{\partial\rho_3}-\frac{\partial f}{\partial\rho_3}\frac{\partial g}{\partial\rho_1}}\left(-\frac{\partial g}{\partial\rho_1}\right)f\end{aligned} \tag{6.25}$$

The partial derivatives are computed numerically, and these numerical estimates have been found to be sufficiently accurate for the angles-only problem, as noted by [20]. Gooding uses the central difference scheme, preventing bias arising with a one-sided increment. While this requires two additional evaluations, it then also allows to evaluate the second-order derivatives, e.g. $f'' = (f(x+h,y)+f(x-h,y)-2f(x,y))/h^2$, and the complete set if extended to five evaluations (f_{xy}, f_{yx}). This results in Newton-Raphson achieving higher-order convergence, as the method utilizes more accurate derivative estimates. The convergence is assessed by the ratio of f to the calculated magnitude of the position vector ρ_2, which Gooding defined adequately converged if it reached $\varepsilon \leq 10^{-12}$. Using Lambert's problem instead of a linear approximate model provides greater flexibility for usable angle data. Hence, in contrast to Gauss angles-only and the Double r-iteration method, this approach allows to use a combination of optical observations from different revolutions as well. It can therefore be used to determine initial orbits using tracklets from different observation nights.

6.2. Multiple observations

Karimi and Mortari [34] proposed a least-squares approach $\mathrm{L_n}$, with $n \geq 3$ number of angle measurements. This extends Gauss' approach to n number of angles, where the Earth-satellite-observer identities are represented for each epoch i as

$$\mathbf{r}_i = \mathbf{R}_i + \rho_i \mathbf{u}_i, \quad i = 1,2,...,n. \tag{6.26}$$

Similarly, the assumption of co-planar position vectors allow to rewrite the middle vector as combination of the outer vectors

$$\mathbf{r}_i = c_i \mathbf{r}_{i-1} + d_i \mathbf{r}_{i+1}, \quad i = 2,3,...,n-1. \tag{6.27}$$

Where for convenience c and d are used to represent the coefficients c_1 and c_3 as used in section 6.1. Following the derivation from section 6.1, the coefficients c and d are generalised by

$$c_i = \frac{g_{i+1}}{f_{i-1}g_{i+1} - f_{i+1}g_{i-1}} \tag{6.28}$$

$$d_i = -\frac{g_{i-1}}{f_{i-1}g_{i+1} - f_{i+1}g_{i-1}} \tag{6.29}$$

Similar to Gauss, if the Lagrange coefficients are known the position and velocity vector at an epoch i can be determined as functions of its neighbouring vectors

$$\mathbf{r}_i = c_i \mathbf{r}_{i-1} - d_i \mathbf{r}_{i+1}, \tag{6.30}$$

$$\dot{\mathbf{r}}_i = -\left(\frac{f_{i+1}}{f_{i-1}g_{i+1} - f_{i+1}g_{i-1}}\right)\mathbf{r}_{i-1} - \left(\frac{f_{i-1}}{f_{i-1}g_{i+1} - f_{i+1}g_{i-1}}\right)\mathbf{r}_{i+1} \tag{6.31}$$

The Lagrange coefficients can be evaluated by their series based on the observation time difference $\tau_i = t_i - t_{i-1}$, following eq. (6.6). Alternatively, the universal exact Lagrange equations can be used, see eq. (6.21).

Karimi and Mortari [34] showed Gauss' three angles method to perform worse compared to L_n, refined Gauss (using exact Lagrange functions) and Double r-iteration due to the approximate Lagrange coefficients used. In general for the exact Lagrange coefficient methods larger time interval yielded lower relative error. For $\mathrm{L_n}$, more observations yielded lower relative error resulting in higher accuracy compared to Double r-iteration and refined Gauss while also having almost the same run time. Additionally it showed that different sets of angles did not influence the relative error. In general, the more measurements that are used the higher the accuracy is obtained. Additionally, up to a certain point, a longer time interval is desired between measurements. It should be noted however, that the $\mathrm{L_n}$ method utilizing multiple angles was compared against Gauss, refined Gauss and Double r-iteration that employed only the first three angles, rather than using three angles equally distributed in time.

6.2.1. Ln solution

The solution, i.e. a set of ranges corresponding to the angle observations, can be obtained by setting up a system of equations. This system of equations is based on the requirement of the Earth-satellite-observer relations eq. (6.26) and the derived co-planar relations eq. (6.27). Satisfying both relations implies

$$c_i\rho_{i-1}\boldsymbol{u}_{i-1} - \rho_i\boldsymbol{u}_i + d_i\rho_{i+1}\boldsymbol{u}_{i+1} = \mathbf{R}_i - (c_i\mathbf{R}_{i-1} + d_i\mathbf{R}_{i+1}). \tag{6.32}$$

Where, when written in matrix form, the left side ranges are denoted by a $[n \times 1]$ column vector $\boldsymbol{\rho}$. The remaining parameters and line-of-sight unit vectors are given as $[3(n-2) \times n]$ size matrix $\mathbf{M}$ and the right hand side as a $[n \times 1]$ column vector $\boldsymbol{\xi}$. Thus, given $n \geq 3$ number of measurements, $(n-2)$ vectorial identities can be written in matrix form

$$\mathbf{M}\boldsymbol{\rho} = \boldsymbol{\xi} \tag{6.33}$$

Which, given the full form takes shape as

$$\begin{bmatrix} c_2\boldsymbol{u}_1 & -\boldsymbol{u}_2 & d_2\boldsymbol{u}_3 & \mathbf{0} & \mathbf{0} & \dots & \mathbf{0} \\ \mathbf{0} & c_3\boldsymbol{u}_2 & -\boldsymbol{u}_3 & d_3\boldsymbol{u}_4 & \mathbf{0} & \dots & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & c_4\boldsymbol{u}_3 & -\boldsymbol{u}_4 & d_4\boldsymbol{u}_5 & \dots & \mathbf{0} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \dots & d_{n-1}\boldsymbol{u}_n \end{bmatrix} \begin{bmatrix} \rho_1 \\ \rho_2 \\ \rho_3 \\ \rho_4 \\ \rho_5 \\ \vdots \\ \rho_n \end{bmatrix} = \begin{bmatrix} \xi_1 \\ \xi_2 \\ \xi_3 \\ \xi_4 \\ \xi_5 \\ \vdots \\ \xi_n \end{bmatrix}. \tag{6.34}$$

Which can be solved for $\boldsymbol{\rho}$ using a least squares approach, for which the solution is obtained by

$$\boldsymbol{\rho} = \left(\mathbf{M}^T\mathbf{M}\right)^{-1}\mathbf{M}^T\boldsymbol{\xi}. \tag{6.35}$$

Once the range vector, $\boldsymbol{\rho}$, is found the radii, through eq. (6.26), are used to evaluate the orbital elements. The most orthogonal range vectors are used for a Lambert solver. Alternatively, the states can be obtained when using the Lagrange coefficients to determine the mid epoch velocity vectors, as done for Gauss' three angle approach. The latter approach is taken in this work, an overview of the implementation is presented in algorithm 1. The full code can be found in appendix B.

Algorithm 1 Implementation of L_n method

```
Require: los= {$\mathbf{u}_0,\mathbf{u}_1,\ldots,\mathbf{u}_n$}, obs_time= {$t_0, t_1,\ldots, t_n$}, sens_loc= {$\mathbf{R}_0,\mathbf{R}_1,\ldots,\mathbf{R}_n$}, tol=1e-8, k_max=500

tau1, tau3 ← obs_time
f1, g1, f3, g3 = 1, tau1, 1, tau3
rho = zeros(n, 1)

while (delta > tol) and (k < k_max) do
    k = k + 1
    r_vec ← sens_loc + rho * los
    r1_vec, r2_vec, r3_vec = r_vec[:-2], r_vec[1:-1], r_vec[2:]
    v2_vec ← r1_vec, r3_vec, f1, g1, f3, g3
    alpha = 2 / norm(r2_vec) - norm(v2_vec)**2 / GM

    for 0 to n-2 do
        x1 ← universal_kepler(tau1, r2_vec, v2_vec, alpha)
        x3 ← universal_kepler(tau3, r2_vec, v2_vec, alpha)
        f1, g1 ← Lagrange_universal(x1, tau1, r2_vec, alpha)
        f3, g3 ← Lagrange_universal(x3, tau3, r2_vec, alpha)
    end for

    ck, dk ← f1, g1, f3, g3
    M, Xi ← construct_matrices(ck, dk, sens_loc, los)
    rho = lstsq(M, Xi)

    delta = abs(rho - rho_prev)
    rho_prev = rho

end while
r_vec ← sens_loc + rho * los
r1_vec, r2_vec, r3_vec = r_vec[:-2], r_vec[1:-1], r_vec[2:]
v2_vec = (-f3*r1_vec + f1*r3_vec)/(f1*g3 - f3*g1)
```

6.3. Batch Least Squares

The batch least squares is based on two primary equations [63]. The propagation of the state using a state transition matrix

$$\mathbf{x}(t) = \mathbf{\Phi}(t, t_k)\mathbf{x}_k, \tag{6.36}$$

and the observation-state relationship

$$\mathbf{y} = H\mathbf{x}_k + \boldsymbol{\varepsilon}. \tag{6.37}$$

Here, $\mathbf{y}$ is the observation deviation vector, H the mapping matrix and $\boldsymbol{\varepsilon}$ the random vector. For the state propagation matrix, $\mathbf{x}(t)$ is the state deviation vector, $\mathbf{\Phi}(t, t_k)$ the state transition matrix and $\mathbf{x}_k$ the state estimate. Then given these relations, the state $\mathbf{x}$ is sought that maximizes the probability of obtaining the observations $\mathbf{y}$. By assuming the measurements are linearly independent and distributed by a Gaussian probability density function, the state can be found using maximum likelihood estimation and Bayesian estimation. Given observations $\mathbf{y}$ the maximum likelihood estimate $\hat{\mathbf{x}}$ maximizing the conditional density function

$$f(\mathbf{x}|\mathbf{y}) \tag{6.38}$$

is sought. Assuming all density functions to be Gaussian and all observations to be independent, the conditional density function can be rewritten as

$$f(\mathbf{x}|\mathbf{y}) = \frac{f(\mathbf{y}|\mathbf{x})f(\mathbf{x})}{f(\mathbf{y})}. \tag{6.39}$$

Where the first term term in the numerator, the likelihood of observing $\mathbf{y}$, given the state $\mathbf{x}$, is given by

$$f(\mathbf{y}|\mathbf{x}) = \frac{1}{(2\pi)^{n/2}|R|^{1/2}} \exp\left(-\frac{1}{2}\left(\mathbf{y}-H\mathbf{x}\right)^T R^{-1}\left(\mathbf{y}-H\mathbf{x}\right)\right), \tag{6.40}$$

assuming the measurement noise $\boldsymbol{\varepsilon} = \mathbf{y} - H\mathbf{x}$ follows a Gaussian distribution with zero mean and covariance R. The prior distribution of the state vector $\mathbf{x}$, the second term in the numerator, is given by

$$f(\mathbf{x}) = \frac{1}{(2\pi)^{n/2}|P|^{1/2}} \exp\left(-\frac{1}{2}\left(\mathbf{x}-\bar{\mathbf{x}}\right)^T P^{-1}\left(\mathbf{x}-\bar{\mathbf{x}}\right)\right), \tag{6.41}$$

and is assumed to be Gaussian with mean $\bar{\mathbf{x}}$ and covariance matrix P. For the denominator, its density function is defined as

$$f(\mathbf{y}) = \int f(\mathbf{y}|\mathbf{x}) f(\mathbf{x})\, d\mathbf{x}. \tag{6.42}$$

Substitution of eqs. (6.40) to (6.42) into eq. (6.39) yields the full form of the conditional density function

$$L = \frac{1}{(2\pi)^{n+p/2}|P|^{1/2}|R|^{1/2}} \exp\left(-\frac{1}{2}\left[\left(\mathbf{y}-H\mathbf{x}\right)^T R^{-1}\left(\mathbf{y}-H\mathbf{x}\right) + \left(\mathbf{x}-\bar{\mathbf{x}}\right)^T P^{-1}\left(\mathbf{x}-\bar{\mathbf{x}}\right)\right]\right)\frac{1}{f(\mathbf{y})} \tag{6.43}$$

The maximum likelihood estimate is obtained by minimizing the negative logarithm of the terms dependent on $\mathbf{x}$ given by

$$\ln L = -1/2\left[\left(\mathbf{y}-H\mathbf{x}\right)^T R^{-1}\left(\mathbf{y}-H\mathbf{x}\right) + \left(\mathbf{x}-\bar{\mathbf{x}}\right)^T P^{-1}\left(\mathbf{x}-\bar{\mathbf{x}}\right)\right]. \tag{6.44}$$

Then following the first and second order optimality conditions, the minimizer or maximum likelihood estimate $\hat{\mathbf{x}}$ is given by

$$\hat{\mathbf{x}} = \left(H^T R^{-1} H + P^{-1}\right)^{-1}\left(H^T R^{-1}\mathbf{y} + P^{-1}\bar{\mathbf{x}}\right). \tag{6.45}$$

6.3.1. Linearisation

The dynamics and the measurements involve significant non-linear relationships, complicating the direct application of linear estimation techniques. To facilitate the estimation process, the non-linear relationships are linearised around a nominal trajectory or state. The non-linear equations of motion can be denoted as

$$\dot{\mathbf{X}}(t) = F(t, \mathbf{X}(t)), \tag{6.46}$$

$$\mathbf{Y} = G(\mathbf{X}(t)) + \boldsymbol{\varepsilon}, \tag{6.47}$$

where $\mathbf{X}(t)$ is the state vector, $\mathbf{Y}$ the observations, and $F(t, \mathbf{X}(t))$ and $G(\mathbf{X}(t))$ the non-linear system dynamics and measurement model respectively. Linearisation of eqs. (6.46) and (6.47) about a nominal trajectory $\mathbf{X}_0(t)$ yields the linearised equations of motion as

$$\begin{aligned} \dot{\mathbf{x}}(t) &= A(t)\mathbf{x}(t), \\ \mathbf{y} &= \tilde{H}\mathbf{x}_i + \boldsymbol{\varepsilon}, \end{aligned} \tag{6.48}$$

where $\mathbf{x}(t) = \mathbf{X}(t) - \mathbf{X}_0(t)$ represents the deviation from the nominal trajectory. The matrices

$$A = \left[\frac{\partial F(t)}{\partial \mathbf{X}(t)}\right] \quad \text{and} \quad \tilde{H} = \left[\frac{\partial G}{\partial \mathbf{X}(t)}\right]. \tag{6.49}$$

are the Jacobian matrices of the system dynamics and measurement model. For a batch of l observations the state deviation vector $\mathbf{x}_0$ at a reference time t_0 can be found following eq. (6.45). Given an initial condition $\mathbf{x}(t_0)$, an a priori estimate $\bar{\mathbf{x}}_0$, the covariance matrix $\bar{P}_0$ the estimated state deviation $\hat{\mathbf{x}}_0$ vector can be determined as follows

$$(H^T R^{-1} H + \bar{P}_0^{-1})\hat{\mathbf{x}}_0 = H^T R^{-1}\mathbf{y} + \bar{P}_0^{-1}\bar{\mathbf{x}}_0. \tag{6.50}$$

Where the terms can be accumulated over a batch of l measurements, forming the normal equations,

$$H^T R^{-1} H = \sum_{i=1}^{l}\left[\tilde{H}_i \Phi(t_i, t_0)\right]^T R_i^{-1}\tilde{H}_i \Phi(t_i, t_0), \tag{6.51}$$

$$H^T R^{-1}\mathbf{y} = \sum_{i=1}^{l}\left[\tilde{H}_i \Phi(t_i, t_0)\right]^T R_i^{-1}\mathbf{y}_i \tag{6.52}$$

to be solved for each batch iteration, obtaining $\hat{\mathbf{x}}$. This state deviation estimate is used to adjust the a priori estimate $\bar{\mathbf{x}}$ for each batch iteration

$$(\bar{\mathbf{x}}_0)_n = (\bar{\mathbf{x}}_0)_{n-1} - (\hat{\mathbf{x}}_0)_{n-1}. \tag{6.53}$$

Where the state transition matrix and reference trajectory are determined by integration toward the current observation time:

$$\dot{\Phi}(t, t_k) = A(t)\Phi(t, t_k) \quad \text{with initial condition} \quad \Phi(t_{i-1}, t_i) = I, \tag{6.54}$$

$$\dot{\mathbf{x}} = F(\mathbf{x}, t) \quad \text{with initial condition} \quad \mathbf{x}(t_{i-1}). \tag{6.55}$$

The observation state matrix $\tilde{H}_i$ is given by the observation state relationships $G(\mathbf{x}_i, t_i)$, i.e. the measurement model,

$$\tilde{H}_i = \frac{\partial G(\mathbf{x}_i, t_i)}{\partial \mathbf{x}}. \tag{6.56}$$

Defining $\Lambda = H^T R^{-1} H + \bar{P}_0^{-1}$, $N = H^T R^{-1}\mathbf{y}$ and $\bar{\mathbf{x}}_0 = \mathbf{0}$, eq. (6.50) is expressed as

$$\Lambda \hat{\mathbf{x}}_0 = N. \tag{6.57}$$

Solving this equation obtains $\hat{\mathbf{x}}_0$, the minimum value of the performance index, eq. (6.44), more commonly written as

$$J(x) = (\hat{\mathbf{x}}_0 - \bar{\mathbf{x}}_0)^T \bar{P}^{-1} (\hat{\mathbf{x}}_0 - \bar{\mathbf{x}}_0) + \sum_{i=1}^{l} \hat{\varepsilon}_i^T R_i^{-1} \hat{\varepsilon}_i. \tag{6.58}$$

Here

$$\hat{\varepsilon}_i = \mathbf{y}_i - H_i \hat{\mathbf{x}}_0 \tag{6.59}$$

is the best estimate of the observation error, which is used for convergence assessment of the root mean square error

$$RMSE = \sqrt{\frac{\sum_{i=1}^{l} \hat{\varepsilon}_i^T R_i^{-1} \hat{\varepsilon}_i}{l \times p}} \tag{6.60}$$

with $\hat{\varepsilon}_i$ a p dimensional vector and $l \times p$ the total number of measurements, in this case $2 \times l$ right ascension declination observations.

7

Verification

This chapter presents the implementation of the estimation methods as to verify their functionality. It aims to provide the context of the method's implementations and prevent the likelihood of unintended behaviour. This allows to study the methods further beyond the mathematical operations towards application on the cataloguing pipeline as will be presented in later chapters.

7.1. Gauss

The three angle Gauss IOD method implementation follows Curtis' [12] implementation and is also verified against the provided example problem. The example problems consider a ground based observer providing three measurements about two minutes apart. As shown in table 7.1, the implementation yields nearly identical solutions, confirming its accuracy and verifying its correctness.

Table 7.1: Verification data of IOD methods.

Method		X_2 [km]	Y_2 [km]	Z_2 [km]	$\dot{X}_2$ [km/s]	$\dot{Y}_2$ [km/s]	$\dot{Z}_2$ [km/s]
Gauss [12, p. 287]	Expected	5659.032954266604	6533.744381518980	3270.154452175442	-3.8797048599380	5.1156458729787	-2.23970272593060
	Obtained	5659.032954266600	6533.744381518974	3270.154452175444	-3.8797048599381	5.1156458729785	-2.23970272593052
		$3.63797881 \times 10^{-12}$	$5.45696821 \times 10^{-12}$	$-1.81898940 \times 10^{-12}$	$1.30562228 \times 10^{-13}$	$2.69118061 \times 10^{-13}$	$-8.30446822 \times 10^{-14}$
refined-Gauss [12, p. 287]	Expected	5662.04164	6537.94991	3269.0483619037582	-3.8854231883209	5.1214081725294	-2.24339546572440
	Obtained	5662.04164	6537.94991	3269.0483619037577	-3.8854231883207	5.1214081725296	-2.24339546572445
		$0.00000000 \times 10^{00}$	$0.00000000 \times 10^{00}$	$4.54747351 \times 10^{-13}$	$-2.26485497 \times 10^{-13}$	$-2.15827356 \times 10^{-13}$	$4.75175455 \times 10^{-14}$

7.2. Ln Gauss angles

Karimi and Mortari's multiple angles method using the least squares approach is verified with the results obtained from [34]. As the paper uses generated measurements with normal distributed random noise, slight variation in results is expected. Nevertheless, comparison with the verified Gauss and refined-Gauss allows to compare the method's behaviour as well. The measurements are generated using the same procedure by considering two-body dynamics only. The procedure involves propagating the initial state for 1 hour for a desired number of consecutive measurements and time interval, representing the tracklet. The corresponding states are converted to topocentric right ascension and declination for a given ground-based observer location. Normally distributed noise with standard deviation of 5 arcseconds is added to simulate true measurements.

The two described scenarios by Karimi and Mortari [34] are used for verification. The first scenario is a co-planar observation of a LEO object (a=7780 km, e=0.1, i=0 deg, Ω=0 deg, ω=0 deg, ν=0 deg initial state) at 0 degree inclination observed at 0 degree longitude, latitude and altitude. The second scenario considers an inclined LEO orbit (a=7800 km, e=0.1, i=45 deg, Ω=345 deg, ω=15 deg, ν=0 initial state). For both scenarios 50 second interval measurements are considered. The obtained relative position errors, evaluation times and differences with Karimi and Mortari's implementation are tabulated in table 7.2.

Table 7.2: Results compared with L_n method [34]. For the implemented method measurements are generated with random noise.

Scenario I - co planar observation						
	Relative position error [%]			Evaluation time [ms]		
n	Implemented	Karimi & Mortari	Delta [%]	Implemented	Karimi & Mortari	Delta [%]
3	11.10	-	-	28	-	-
4	0.001	0.030	-95.53	34	28	21.43
5	0.003	0.023	-74.86	38	30	26.67
6	0.019	0.020	71.05	40	32	25.00
Scenario II - inclined orbit						
3	0.068	0.06	12.55	17	26	-34.62
4	0.005	0.030	-84.97	20	30	-33.33
5	0.005	0.023	-79.91	39	32	21.88
6	0.002	0.020	-90.48	49	34	44.12

The results show large differences in obtained accuracy as well as evaluation times. Remarkably, the implemented results show lower residuals and higher computational time when compared to Karimi and Mortari's implementation, which could be explained by a different approach for refining the Lagrange coefficients. Apart from the coplanar case, the behaviour is largely the same for both implementations. For the co-planar observation, the three angles show indeed to not be able to properly solve for the ranges, requiring an additional equation to solve the system. Accordingly, for more than three measurements the method is able to solve the system and provide adequate range estimates. As for the inclined orbit, the relative errors show a slight decrease with increasing number of measurements n. This behaviour is not shared on the co-planar case however.

The implemented method for the least-squares approach to Gauss' angles only method are applied for varying measurement noise, time interval and number of measurements (figs. 7.1, 7.3 and 7.5) similarly as described by Karimi and Mortari, see figs. 7.2, 7.4 and 7.6 [34]. The behaviour is largely within the same order of magnitude, and shows an increase in accuracy for increased time-interval. The least-squares approach also shows to obtain the same solution as for the three angles Gauss approach. When extending to more measurements however, the behaviour is reflected less accurately, as observed in fig. 7.5. Additionally, the extension to more measurements seem to indicate an advantage of multiple angles over the three angles Gauss, while this is true for the co-planar case, comparison with Gauss only includes the first three angles. In this comparison, Gauss considers only a fraction of the arc spanned by the multiple angles approach.

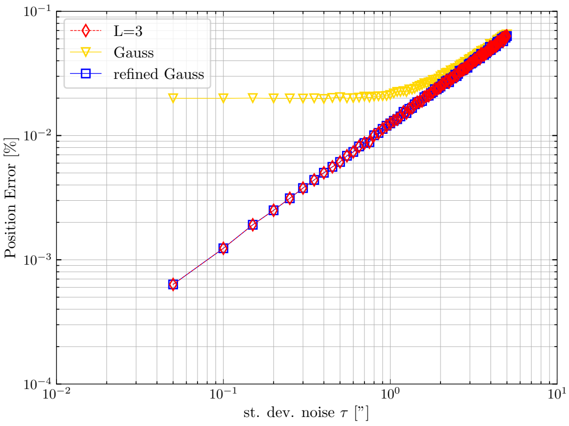


Figure 7.1: Sensitivity to noise obtained after implementation. Plot shows average results of 2000 runs with simulated noise.

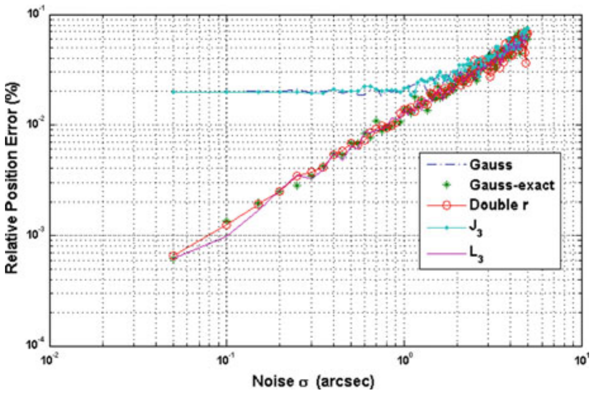


Figure 7.2: Sensitivity to noise obtained from [34]

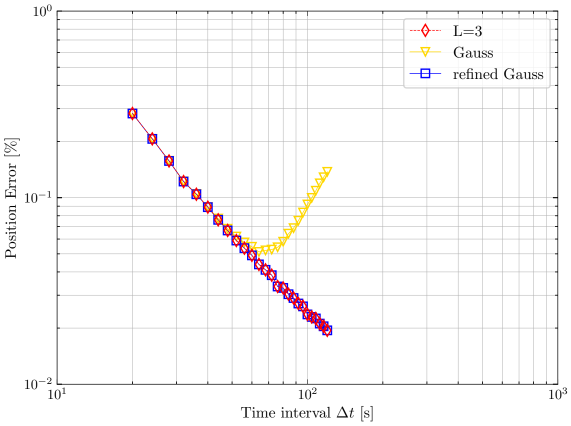


Figure 7.3: Sensitivity to time interval obtained after implementation. Plot shows average results of 2000 runs with simulated noise.

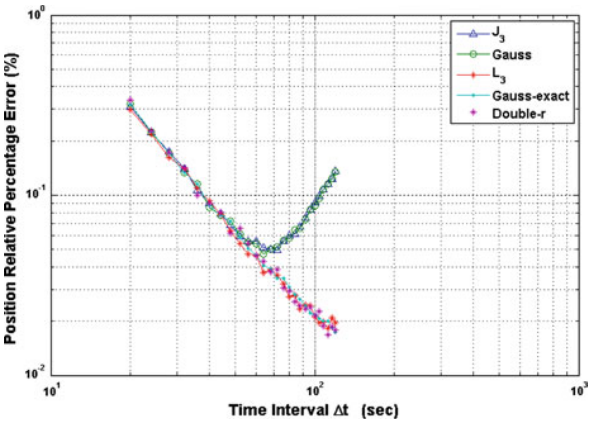


Figure 7.4: Sensitivity to time interval obtained from [34].

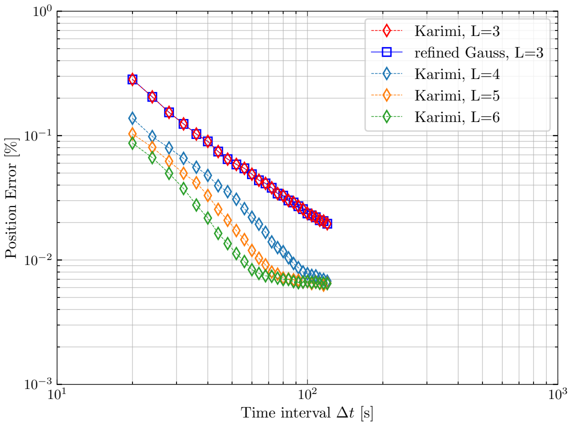


Figure 7.5: Effect of number of measurements. Plot shows average results of 2000 runs with simulated noise.

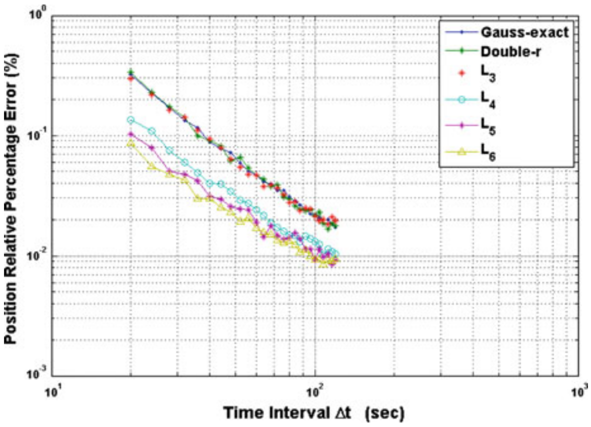


Figure 7.6: Effect of number of measurements [34].

In order to more fairly present the relative performance of the least-squares approach, the most orthogonal triplet covering the full arc is selected for Gauss. This means that if the least-squares approach uses five measurements, Gauss is applied to the first, third and fifth measurements. When extended to more angles in this way, the implemented method shows that the additional angles are not effectively used, as depicted in fig. 7.7. In general, the least squares approach gives approximately the same error as Gauss.

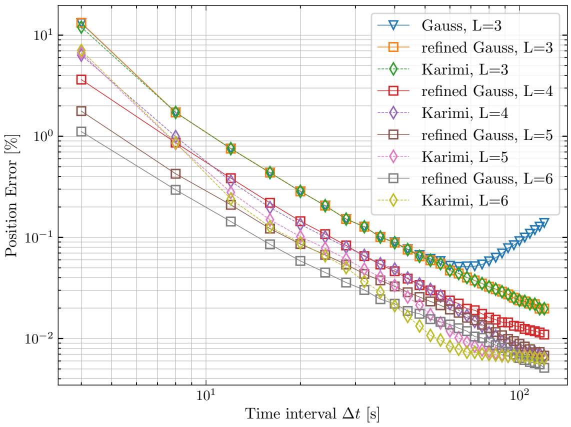


Figure 7.7: Sensitivity to number of measurements and time interval compared to using most orthogonal angles in Gauss' three angles approach.

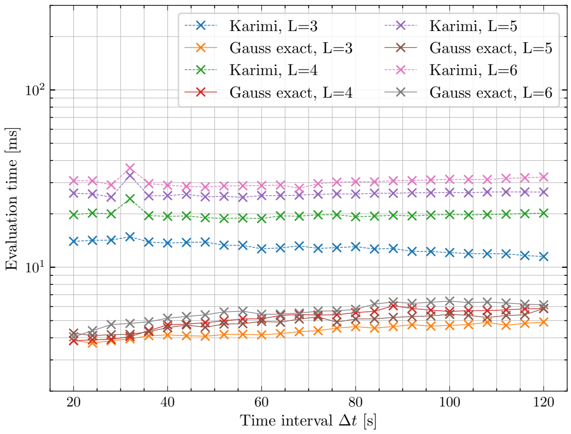


Figure 7.8: Comparison for Gauss three angles and L_n method. Gauss considers three angles equally distributed w.r.t. time. Average results over 2000 runs.

From both figs. 7.7 and 7.8 it becomes clear that when comparing the least-squares approach with Gauss three angles using orthogonally spaced measurements the difference in accuracy becomes negligible. Moreover, the shorter time span of the angles show to affect the accuracy more compared to Gauss. While there seems to be a region for longer time intervals where the least-squares approach benefits from the full angle-set, it is not consistent over the full time intervals. The computational time of the least-squares method is also less favourable as it scales with the number of measurements considered, even showing for three angles the same accuracy is obtained at higher computational cost, see fig. 7.8.

7.3. Gooding

Gooding's three angles only method is implemented from Orekit [47], it uses a Lambert solver implementation following Battin's approach. This method iterates over the semi-latus rectum to solve the time-of-flight equation in a universal approach [3, 18]. The implementation has been verified against the example presented in the work of Gooding [20]. This example, taken in reference to Herrick's work [24], considers three observations of the minor planet 683 Lanzia.

Table 7.3: Verification of Gooding's implementation by Orekit [47].

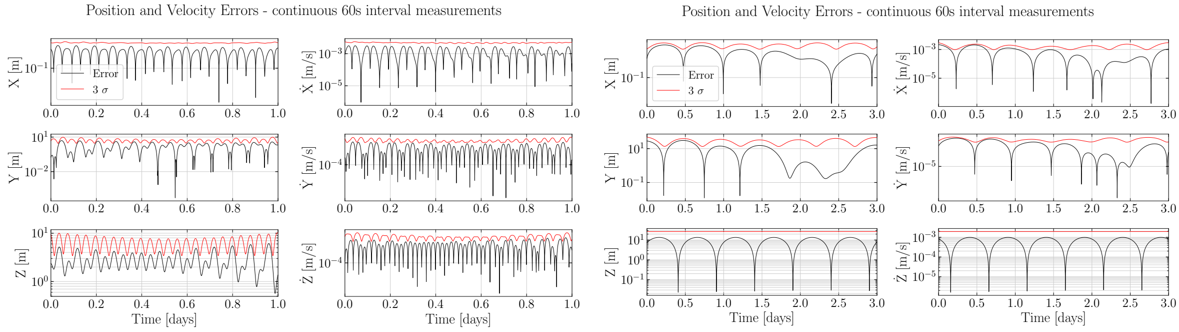
Parameter	**Expected solution [AU]**	**Obtained solution [AU]**	**Difference [AU]**
ρ_1	2.3991972	2.3991973	7.4080e-08
ρ_2	2.563703947213	2.563703947212	4.4498e-13
ρ_3	2.824544	2.824545	2.5145e-07

Note that the current implementation (Orekit 12.2 [47]) turns out to be very sensitive to the initial guess. Gooding [20] introduced the method and showed that it successfully converges to the solution with an initial guess of 5.9 in both domains. However, with the included implementation it was seen to converge to the correct answer only when the initial guess was essentially equal to the true solution. However, as Gooding's method is implemented to use initial range estimates from the tracklet correlation, the current implementation is still included and is considered to be performing adequately.

While no significant effort has been made to investigate the implementation further, it is believed that the dependency is largely due to the fact that Orekit's Gooding implementation relies on Lancaster and Blanchard's [37] implementation of the Lambert solver [23]. Gooding's IOD method is presented instead, with his own implementation of the Lambert solver using a bi-linear initial guess and Halley's iteration procedure to solve for the free parameter in the time of flight equation [21, 18]. The Lancaster and Blanchard method uses an arbitrary initial guess based on fixed values and solves for the free parameter using a secant iteration procedure. The arbitrary initial guess for the root-solving procedure is thought to cause increased sensitivity for the initial range estimates compared to a bi-linear initial guess.

7.4. Batch Least Squares

To showcase the implementation of the batch least squares estimator, a LEO and GEO orbit case are shown. For both cases only two-body dynamics is included, and the same diagonal covariance, with 10^3 km and 10^2 m/s standard deviation. For GEO a three day time span is considered, while for LEO a one day time-span is set. Over this time-span continuous measurements with 60 second time interval are generated with 2 arcseconds Gaussian white noise added to the true measurements. The observer is positioned at -6.627736 deg lon., 38.215828 deg lat. and 583.47 m altitude.

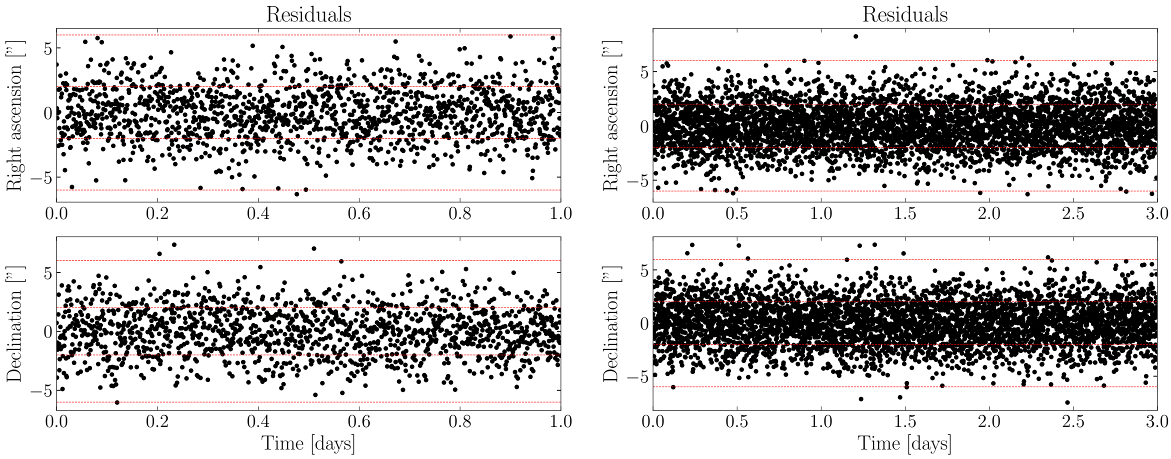


(a)

(b)

Figure 7.9: Batch least squares applied to continuous measurements of a LEO (7.9a) and GEO (7.9b) object for 1 and 3 days with 60 second interval under unperturbed dynamics. Initial state at 00:42:05.91: a=7858.39 km, e=0.0027, i=73.8977 deg, Ω=293.3976 deg, ω=110.2098 deg & ν=-85.5763 deg for LEO and a=42164 km, e=0, i=0 deg, Ω=0 deg, ω=0 deg & ν=10 deg at 2024-07-06 00:14:12.00 for GEO.

For both runs as depicted in fig. 7.9, the batch least squares properly filters out the noise for two-body dynamics modelled measurements, estimating 1.9897 and 1.9961 arcseconds standard deviation on the residuals and obtaining within a meter accuracy estimates. The residuals are presented in fig. 7.10.



(a)

(b)

Figure 7.10: Angle residuals between true measurements and modelled measurements for both LEO (7.10a) and GEO (7.10b). The red dotted lines indicate ± 1σ and ± 3σ.

8

Methodology

This chapter describes the methodology and steps taken to derive the results as presented in chapter 9. The aim is to provide a clear overview of the approach taken with the methods previously described to clearly get an idea of how the results are acquired. The simulated measurements and observation are described first along with the assumptions and methods used for visibility constraints. Second, the object population and measurements acquired are shown. Finally, the set-up used for the simulations and analyses is presented to provide context of the runtime performance later discussed.

8.1. Observations & Measurement simulation

Using an external space object catalogue as the reference for initial states, artificial objects are modelled for state propagation. The corresponding measurements are extracted from the propagated states based on a realistic measurement configuration in accordance with ART. Conditions for visibility are defined, based on which generated measurements are excluded. In the end, a dataset of measurements is obtained simulating a realistic optical observation scenario. The outline of the primary steps for the measurement simulation process is presented in fig. 8.1, each step is explained hereafter.

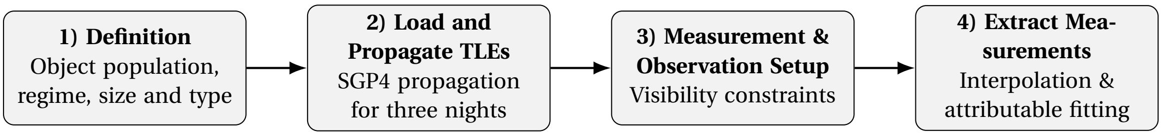


Figure 8.1: Overview of methodology for measurement simulation

8.1.1. Object population

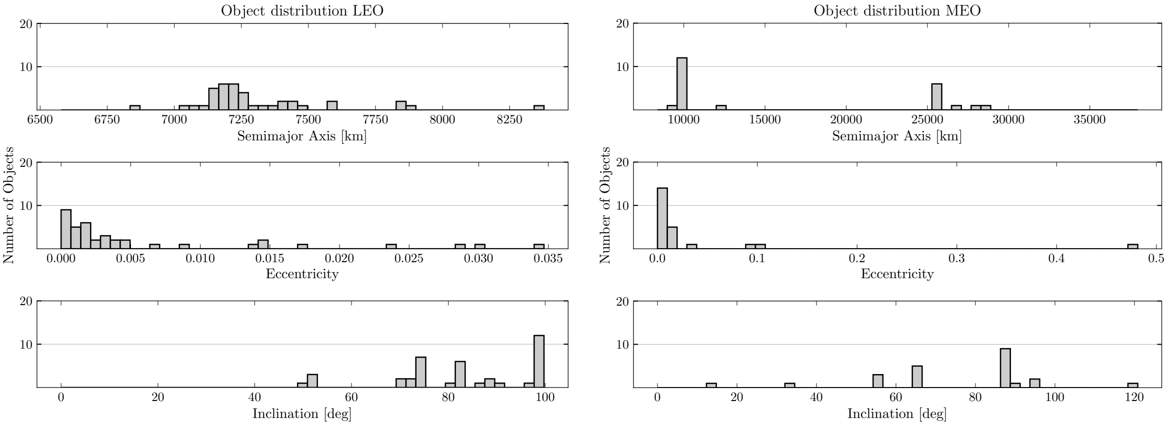
Space-track.org's database and accompanying Application Programming Interface (API) are used to construct object populations by extracting two-line element set (TLE) for objects within the desired orbital regime. Specifically, space-track's query builder is used on its 'tle' class which provide historical TLEs for a given time period. The API allows to filter the TLEs based on its contained data, such as orbital elements at epoch, object type, norad id, TLE epoch, etc. Accordingly, three separate populations were loaded to represent the LEO, MEO and GEO regimes. Due to the implementation of the measurement generation, 2000 TLEs were extracted and sorted in chronological order on TLE epoch between the 14^{th} and 15^{th} of July 2024. The TLEs are required to be sorted before extraction to limit entries, accordingly the TLE epoch was selected as to prevent sorting on properties that could impact the distribution such as orbital elements or time since launch.

Considering the tracklet-to-tracklet correlation approach, around 50 objects per regime, and two tracklets per night for three observation nights were aimed to be included as to limit excessive run-time. Additionally, the initial reference states of the objects - consisting of payloads, debris, and rocket bodies - are aimed to be distributed similarly to the object type distributions as reported in [15]. The definitions of the considered LEO, MEO, and GEO regimes follow those provided in the same report. Table 8.1 provides an overview of the objects used as a reference for the simulated measurements. However, the current approach to generating the object population is rather inefficient and has resulted in a relatively small MEO population, primarily

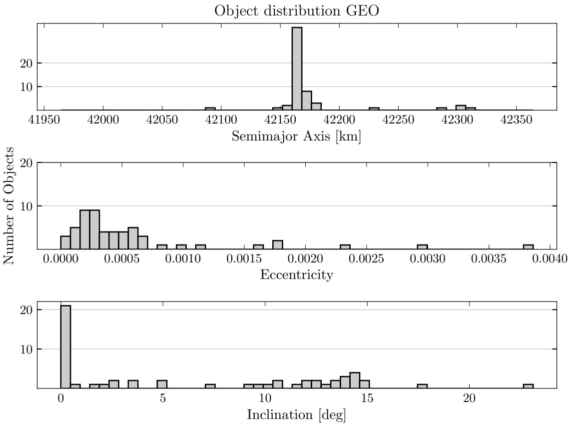
Table 8.1: Object population and orbital regimes.

Orbital regime	**No. of objects** payload	 debris	 rocket body	**Altitude, km**	**Inclination, deg**
LEO	18 (46.15%)	20 (51.28%)	1 (2.56%)	[0, 2000]	[0, 180]
MEO	13 (56.52%)	8 (34.78%)	2 (8.70%)	[2000, 31570]	[0, 180]
GEO	51 (92.73%)	0 (0.00%)	4 (7.27%)	[35586, 35986]	< 25

due to data handling difficulties rather than physical constraints. This is primarily due to the posteriori filtering of extracted measurements based on the visibility constraints. A large number of objects is considered per regime in order to filter down toward the desired object populations, which does not guarantee the desired object population when insufficiently suitable objects were considered before filtering. Nevertheless, the object types are reflected quite well with respect to [15], which shows 48% payloads, 48% debris & 5% rocket bodies for LEO, similarly 12%, 83% & 4% for MEO and 88%, 5% & 7% for GEO. The distribution is not seen to impact the results much, but still could provide meaningful discrepancies between active payloads whose orbits are often less eccentric. Regardless, this approach was seen to more realistically reflect the scenario compared to focusing on active payloads. An overview of the orbital elements of the reference TLE are presented in fig. 8.2.



(a) LEO (b) MEO



(c) GEO

Figure 8.2: Distribution of the simulated object regimes. Object types according to table 8.1.

8.1.2. Object state propagation

The loaded TLEs provide the initial state of the artificial objects contained within the orbital regime population. The TLE represents the mean orbital elements at the corresponding epoch and contains information such as the initial state and the B^* coefficient representing the object's response to atmospheric drag. This TLE descriptor provides the necessary information to obtain the object ephemeris by state propagation using the simplified perturbation models, SGP4/SDP4, which are often used in conjunction with the standardized TLE state description.

For the simplified perturbation models the Python sgp4 package is used [52], which is a wrapper around David Vallado's SGP4 implementation [67]. The propagation model accounts for both near earth, that is, objects with a period smaller than 225 minutes, and deep space objects with an orbital period greater than 225 minutes and thus contains SGP4 and SDP4, of which the merged model is commonly referred to as SGP4 [67]. Vallado [67] mentioned improvements have been made to the code since development but considered the fundamental theory as documented in [25] to be unchanged since. Hoots et al. [25] describes the following perturbations as part of the combined SGP4 model: Earth's atmospheric drag, third-body Sun and Moon and resonance effects of Earth's zonal harmonics.

This SGP4 model provides a robust and efficient method for perturbed measurement simulation, and allows to compare results with the two-body dynamics estimation methods. As a consequence, for propagation of several days, the simplified dynamics are expected to lead to state errors more dominant than the reported TLE error for MEO and GEO around a kilometer position error at epoch [51]. The TLEs are therefore propagated for a maximum of three nights, in order to prevent largely erroneous orbits while still being able to study the effect of large revisit times. Ideally, the historical TLEs should be reconsidered as often as available (instead of propagation of one initial TLE at epoch) for multiple observation nights to assure minimum discrepancy. Nevertheless, considering the restricting two-body dynamics used in the estimation method for correlation and initial orbit determination it is considered to describe the orbit dynamics with sufficient accuracy to study the approach's effectiveness on measurements for perturbed objects.

8.1.3. Measurement & Observation setup

For each object a truth ephemeris is generated containing the states for three consecutive nights at a 10 second interval, starting the 14th of July 2024. Given ART's location, 38.215828° latitude, -6.627736° longitude and 583.47 meter altitude at epoch, the topocentric right ascension and declination are determined. With a predefined visibility constraints (as discussed in section 8.1.4) and measurement configuration the ideal, zero-noise measurements are extracted from the truth ephemeris through interpolation toward the desired observation epochs.

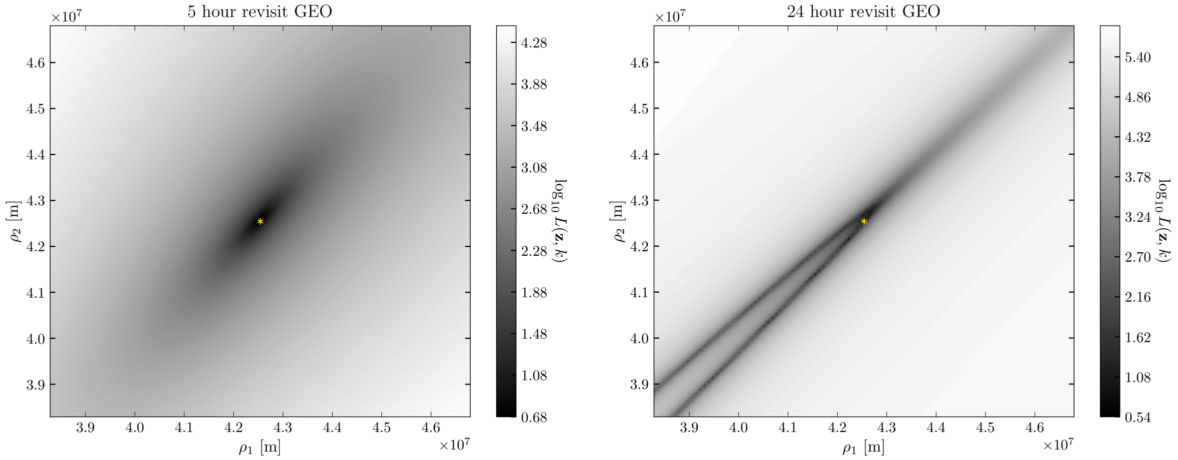
Two tracklets per object are aimed to be extracted for a single observation night. For LEO and MEO the tracklets are assumed to be successfully re-observed within the same night after 5 and 20 minutes respectively. Extraction of measurements for the GEO regime is implemented such that it reflects an observation strategy preventing observations at the same orbital location for consecutive nights, as this has been shown to degrade performance, given the Lambert solver is sensitive to revisit-times on half or full revolutions (i.e. 0 or 180 degree angle separation), see fig. 8.3 [60]. As this is seen a feasible strategy to implement in practice, it has also been included here.

The distribution of measurements for GEO is achieved by an observation strategy that alternates observing the GEO regime by half a night. This means, when the observation starts at the beginning of sundown it will re-observe the same objects half a night later, the next night, it will start observing at the middle of the night and then re-observe at the end of the night. The night is thus divided into four parts, over an interval of three nights the GEO band is observed alternating by shifting a quarter night for the observations. The difference between extraction at the start of the visible part instead of distributed extraction over several nights is shown in fig. 8.4.

The measurement configuration is defined by the tracklet composition, i.e. the number of measurements and the time interval of observation. The tracklet lengths are determined based on typical tracklet lengths from the ART telescope, dictated by the exposure and readout time of the sensor. Accordingly the number of measurements contained within a tracklet l and the measurement interval Δt are set 5, 11, 11 and 2, 4, 7 seconds for LEO, MEO and GEO orbits respectively. This means short arcs of 8 to 70 seconds are observed.

8.1.4. Visibility

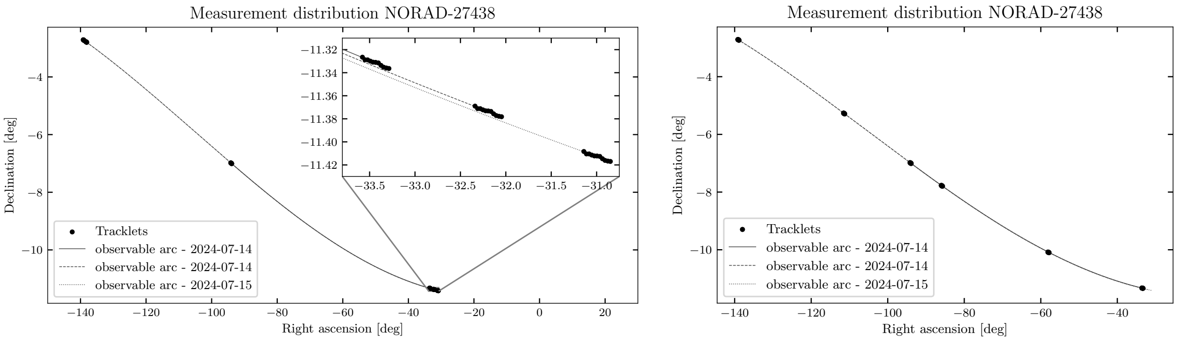
The observations are simulated assuming idealized conditions and ideal objects. This means that no limitations for the optical sensor are considered in terms of visibility, e.g. small or faint objects or bright background



(a) GEO object re-observed after 5 hours. 142 m and 236 m rmse.

(b) GEO object re-observed after 24 hours. 45 km and 45 km rmse.

Figure 8.3: Topography of the BVP solutions for a GEO object. Both cases considered measurements simulated by two-body dynamics only, 2" noise, and tracklets containing 11 measurements with 7 second interval.



(a) Observations at the start and end of each night.

(b) Observations offset by a quarter night for consecutive nights.

Figure 8.4: Topocentric right ascension and declination of measurements distributed for three nights.

stars. It is thus assumed that all objects are adequately bright for passive optical observation while still respecting the dynamics as characterized by their respective reference TLE objects. Aberrations, atmospheric effects and sensor noise is represented by adding zero-mean Gaussian noise of 2 arc seconds standard deviation to the simulated true measurements, a typical value seen for ART [54]. As for the geometric visibility, an object's visibility is defined by nautical twilight, or maximum Sun elevation of -12°, maximum object elevation of 30° and requires the object to be outside Earth's shadow. The shadow function, which determines if an object is within Earth's shadow cone, considers the intersections of the line spanning between the object's and Sun's centre, and Earth's spherical surface. Where the coordinate system is defined relative to the observed object. With the location of the Sun and Earth relative to the object the intersection between the connecting line and Earth's spherical surface is found according to [4]. The approach computes the intersections as the line-distance from the object between the Sun and the object. The first intersection's distance is denoted by x_1 and if existent, the second by x_2. Figure 8.5 shows two cases where the object is said to be within Earth's shadow cone.

If the intersections are positioned such that the Earth is in between the object and sun,

$$\Pi = \begin{cases} 1 & \text{if} \quad x_1 > 0 \quad \& \quad x_2 < r_{o-S}, \\ 0 & \text{else}, \end{cases} \tag{8.1}$$

the object is said to be in Earth's shadow ($\Pi = 1$).

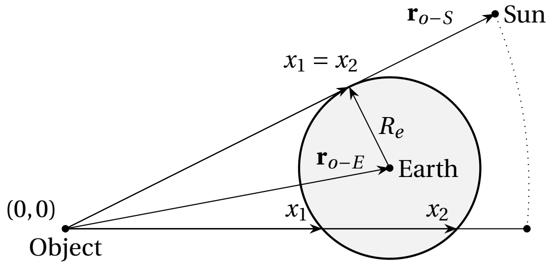


Figure 8.5: Earth shadow condition.

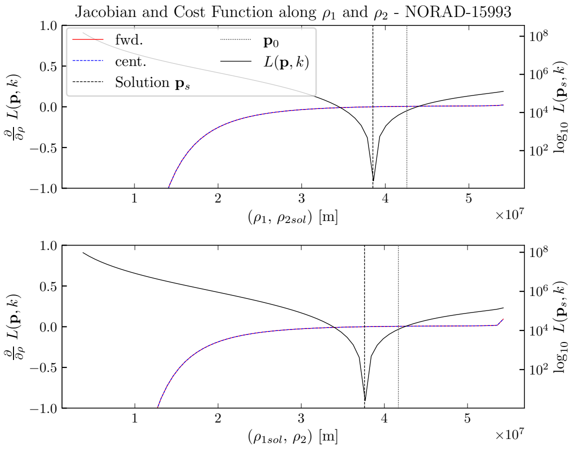


Figure 8.6: Jacobian (fwd. & cent. differencing) and cost function for correlation of 11x 7s observations of a GEO object using BFGS.

8.2. Computational set-up

The tracklet-to-tracklet correlation approach requires testing all feasible combinations of measurements. Following the outlined procedure for the measurement simulation, 115, 78 and 248 attributables were obtained for LEO, MEO and GEO. This means $n\,(n-1)\,/2$ cases are considered for correlation, i.e. 6555, 3003 & 30628 respectively. To try and minimize the run-time of the complete correlation process, parallelization is applied across all available threads. This parallelization is achieved through Joblib, a Python package optimized for efficient task parallelization and caching [31]. The correlation simulation is conducted on a Windows 10 64-bit machine equipped with an AMD Ryzen 7 3700X 8-Core processor (16 threads) at a 3.6 GHz base clock speed and 16 GB of random-access memory. Regardless, the implementation of the correlation procedure in Python can be found in appendix B or on `github.com/casruks/IOD_T2T.git`.

8.2.1. Optimization

All considered optimization methods are obtained from Scipy's (v1.14.1) optimization library [57]. As mentioned in section 5.3, the BFGS, L-BFGS-B and Powell method are considered. It should be noted however that for the larger population sizes considered the parallel iterative scheme lead to memory leaks under the current implementation for the BFGS optimization method. It was observed that iterative calls to the BFGS method with the required gradient tolerance led to memory leaks in successive correlation cases. Without success it was tested under several conditions to clear memory after each consecutive correlation case. This meant the BFGS methods could not successfully obtain the desired results over a larger population of objects. This further motivated to investigate the performance of the zeroth order as well as the low-memory bounded quasi-Newton method. Scipy's (v1.14.1) minimization functionality is used for optimization with the conditions as tabulated in table 8.2.

Table 8.2: Tolerance levels and iteration limits set for the optimization methods used. `xtol` controls the precision in the solution's position, `ftol` the tolerance in function value, and `gtol` the gradient tolerance [57].

Method	**xtol**	**ftol**	**gtol**	**Maximum Iterations**
Powell	10^{-4}	10^{-4}	N/A	2000
L-BFGS-B	N/A	10^{-4}	10^{-7}	2000

As for the quasi-Newton method, the first-order gradient is estimated using finite differences since the cost function is non-analytical due to its dependency on the Lambert solver. The forward differencing uses a default relative step size. This relative step size aligns with SciPy's implementation, which is based on the square root of the machine's precision. In this case, it is approximately ~ `(2.2204·10`$^{\texttt{-16}}$`)`$^{\texttt{0.5}}$. The difference between central differencing and forward differencing was found to be negligible in terms of accuracy, see fig. 8.6, therefore the default forward differencing is chosen. Initial guess and solution are highlighted by $\mathbf{p}_0$ and $\mathbf{p}_s$.

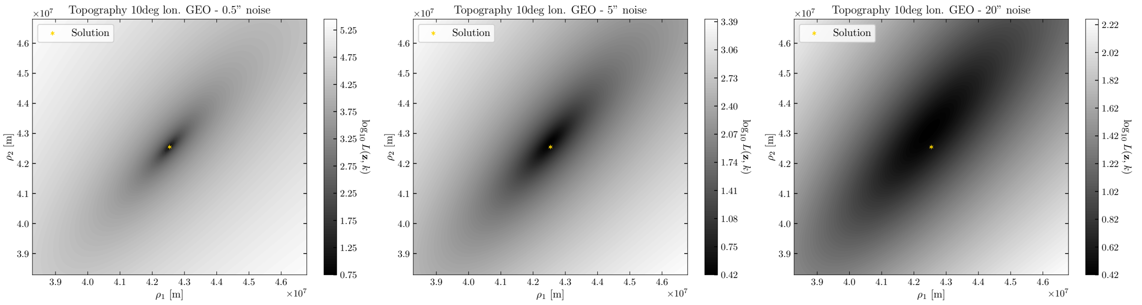
9

Results

This chapter presents the results obtained using the methodology described in the previous chapter. First the baseline correlation results are presented, where a comparison of the object populations for each regime is presented for both the Powell and L-BFGS-B optimization methods. The results are characterized by the cost-function distributions, correlation accuracy and root-mean square error between the true and estimated states. Lastly, from the BVP results, for a select gate value, the impact of an additional initial orbit estimation method and BLS OD method on aforementioned characteristics is presented.

9.1. BVP T2T

The BVP correlation method is outlined in section 5.3, its performance is directly affected by the measurement uncertainty, optimization performance and the uncertainty transformation. Accordingly, the estimation errors are dependent on the observation strategy, measurement quality, attributable fitting, Lambert solver and numerical optimization. The overall performance and primary contribution of the BVP is affected by the ability of the cost-function to correctly capture and represent the likelihood of association. If the observation strategy and measurements provide sufficient information for a suiting attributable fit and the cost-function precisely captures the correlation likelihood, the admissible region can efficiently be sampled and the true range solution will coincide with the obtained minimizer. This can be illustrated in fig. 9.1, where the impact of measurement noise on the topography is shown. In case the measurements contain larger noise values, the minimum becomes less distinct and the region around the solution flatter, as seen in fig. 9.1. This makes it more challenging for optimization methods to find a precise solution, often resulting in an estimate with larger state errors.



(a) Obtained solution for [7.33, 1.13] km and [0.758, 0.383] m/s rmse [ρ_1, ρ_2].

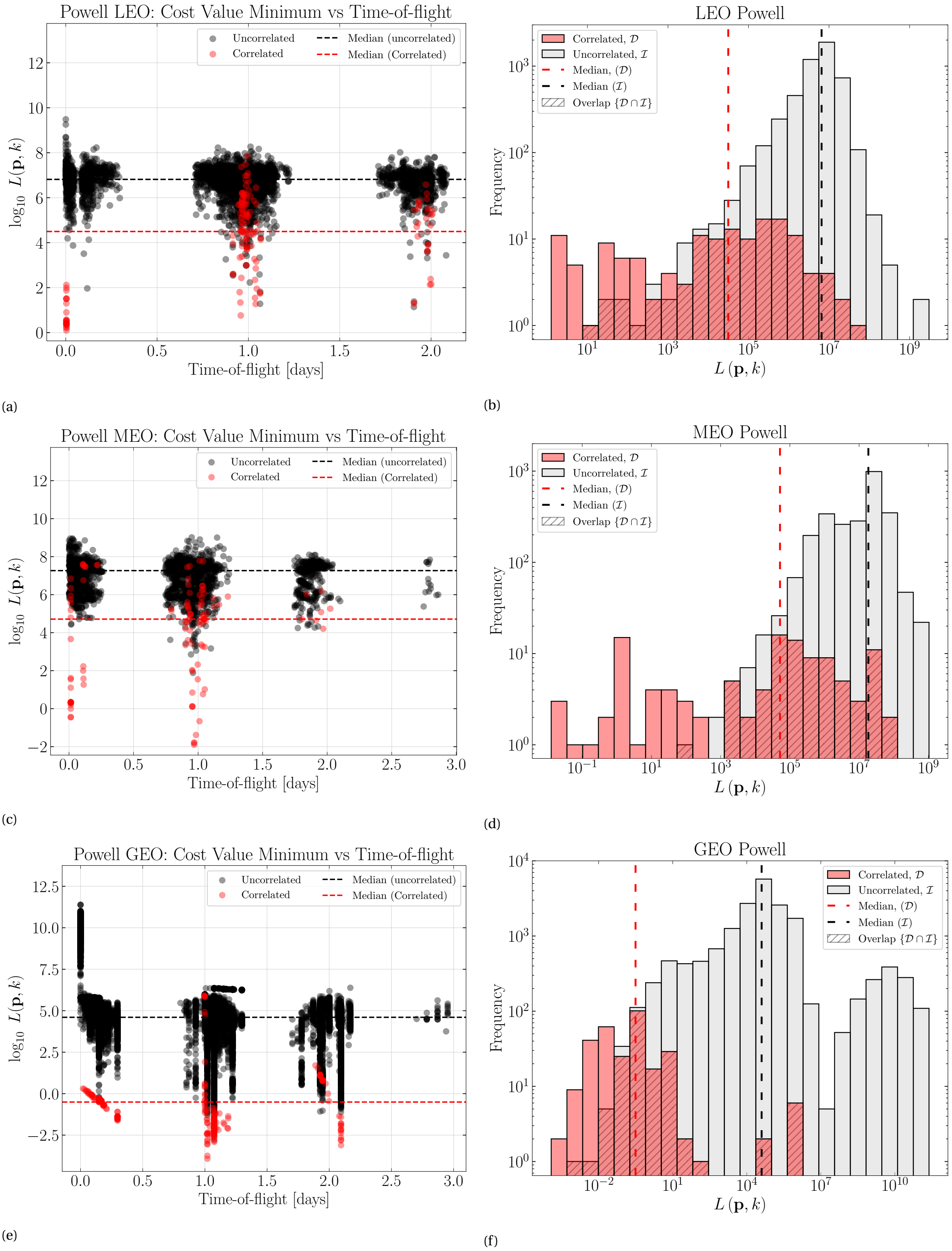
(b) Obtained solution for [74.02, 10.90] km and [7.63, 3.88] m/s rmse.

(c) Obtained solution for [155.90, 158.49] km and [18.87, 18.47] m/s rmse.

Figure 9.1: Effect on measurement noise on topography, shown for 0.5", 5" and 20" random noise. Object considered is modelled under two-body dynamics. Reference state used of [41523.434, 7321.702, 0, -0.534, 3.028, 0.] (km, km/s) at 2024-07-06 00:14:12.000, for 11 measurements in tracklet with interval of 7 seconds.

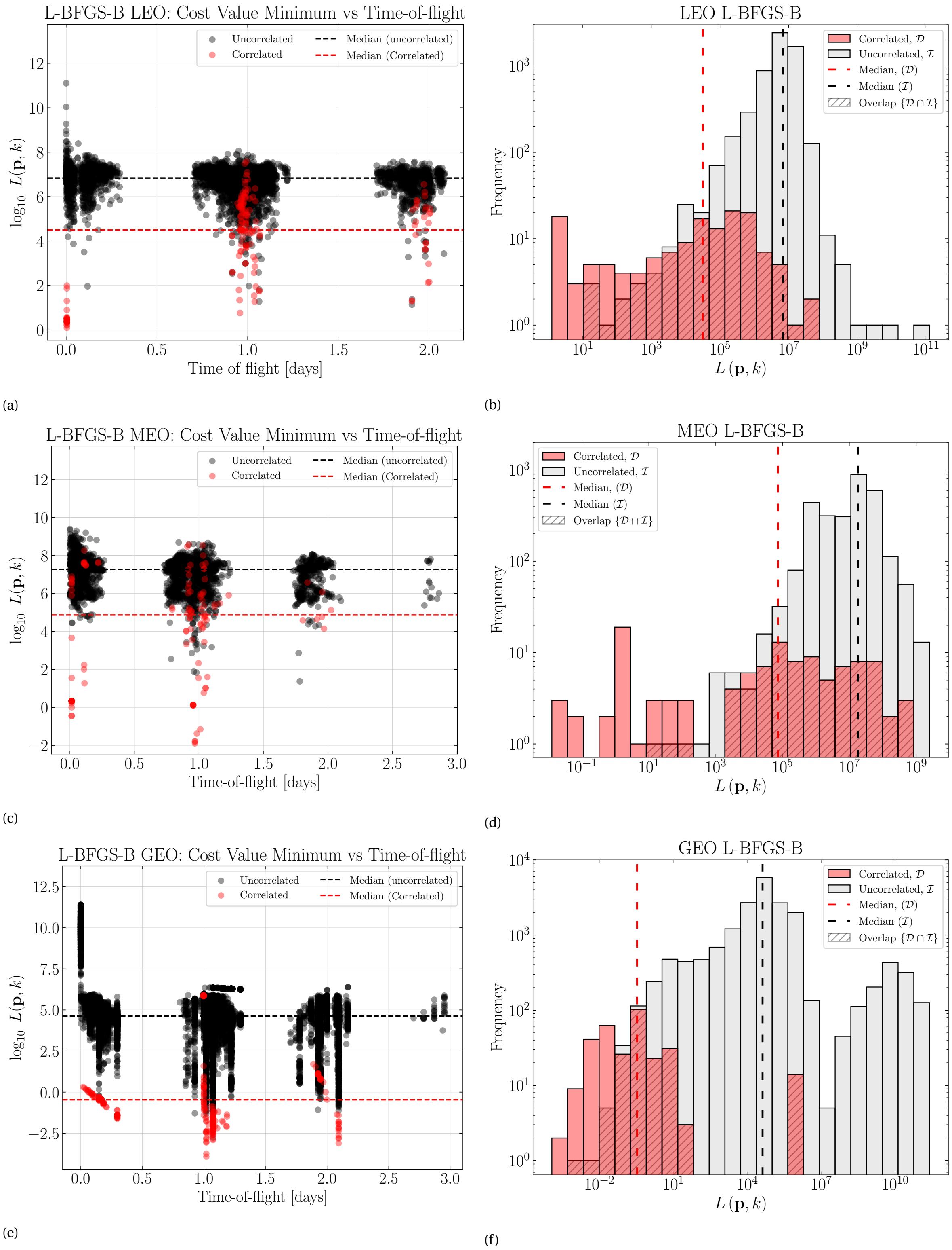
9.1.1. Correlation

Having obtained a group of attributables, all attributables are evaluated against each other for correlation. Each correlation procedure obtains the ranges minimizing the cost function, which together with the corresponding minimum cost value and estimated state $\mathbf{r}$ and $\dot{\mathbf{r}}$ form the solution. The complete set of minima, in relation to the distribution, indicates the likelihood of attributable correlation. The resulting distributions for each regime and optimization method are shown in figs. 9.2 and 9.3.



(a) (b) (c) (d) (e) (f)

Figure 9.2: Distributions of the minimum cost function obtained using Powell's line search optimization method.



(a) (b) (c) (d) (e) (f)

Figure 9.3: Distributions of the minimum cost function obtained using low memory BFGS optimization method.

In addition to the $\log_{10}$-scale cost values and frequency, the relation to the time-of-flight is plotted as well, where the transparent markers aim to show the regions of higher frequency. The distinct bands for the time-of-flight are visible around integer days time-of-flight, corresponding to the observation strategy and nautical twilight. In both plots, the true uncorrelated and correlated measurements are marked, showing there is a clear distinction between the correlated and uncorrelated minimum cost function value irrespective of orbital regime. As for the cost function distribution (right), it should be noted that the observed peaks in the distributions should be interpreted with care due to the logarithmic scale of the cost function and the bin widths.

While the distinction between correlated and uncorrelated cost values is clear in all orbital regimes, the LEO and MEO regimes show a flatter distribution at orders of magnitude higher cost function values. This seems to indicate an altitude dependency, showing the impact the relative orbital coverage or feasible revisit time has on correlation performance. In an effort to identify the sensitivity or relation of orbital regime on the cost function values, the eccentricity, inclination, completed orbital revolutions, mean motion and relative tracklet length are shown in fig. 9.4. From the data obtained, the relationship between the cost function and the orbital regime remains unclear, possibly due to the interdependence of the parameters. Changes in one orbital parameter are accompanied by variations in others, preventing the isolation of their individual effects. The minimum cost functions for varying orbital elements (fig. 9.4) seem to indicate a decrease in the lower cost values for increasing eccentricity, inclination and revisit time relative to the orbital period (nrev). The number of revolutions between observations indicate, as also seen in figs. 9.2 and 9.3, the difficulty in small angle separation, and also showing an increase in cost-function when more number of revolutions are taken between observations. Thus, higher cost functions are observed for objects with lower semi-major axes, or equivalently, for higher mean motion, and more completed orbital revolutions between observation nights.

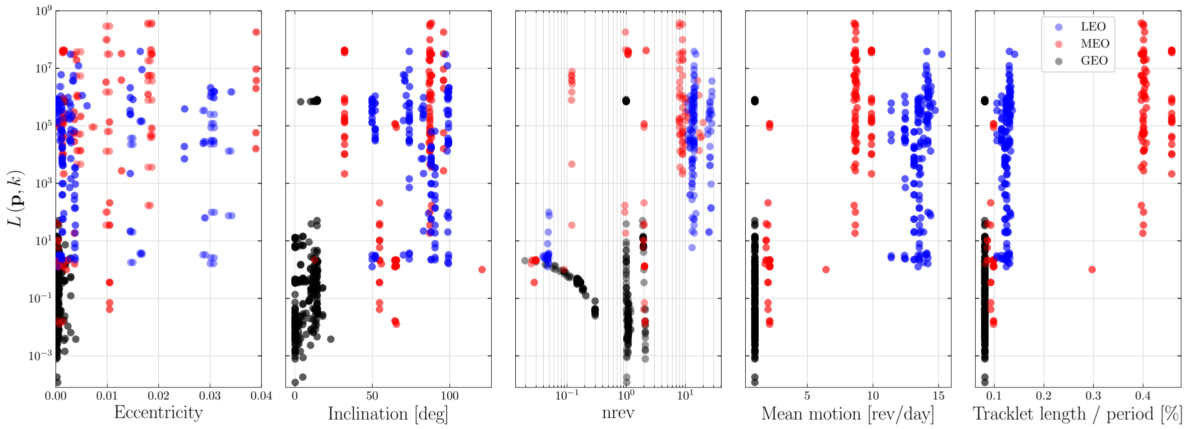


Figure 9.4: Orbital and observation parameters against obtained cost function minimum for true correlations.

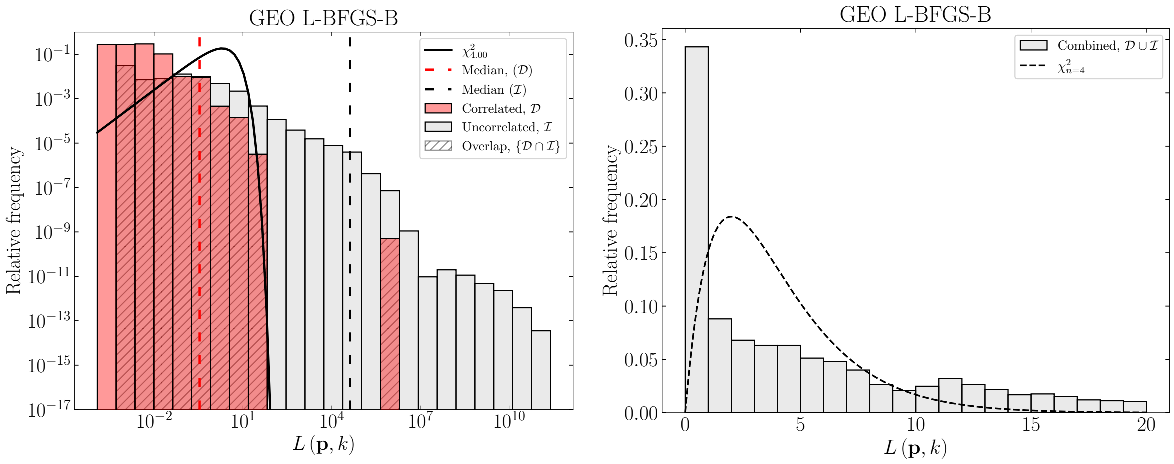
Still, the relation remains unclear as longer tracklets, which are expected to benefit correlation, do not show a clear advantage in terms of lower obtained cost functions. This can be observed for the lower altitude MEO population in fig. 9.4, showing a relatively long tracklet lengths at higher cost function minima. Additionally, a significant difference in cost function for marginal increase in tracklet length is seen between the GEO and LEO population. It may be explained because of the cost function scaling with uncertainty, in which the angles with larger deviations contribute less to the cost value, and smaller deviation angles are amplified. It seems, given the assumed observation strategy and measurements, the lower altitude objects pose a challenge for the BVP to capture accurate estimates. Whereas GEO objects are typically re-observed within two orbital-revolutions, in the order of 10 revolutions between observations are common for LEO and (low altitude) MEO objects. Furthermore, at lower altitudes, the primary perturbation of Earth's atmospheric drag and zonal harmonics are observed to result in a more pronounced force-model discrepancy when compared to GEO. This behaviour is also reflected for the higher MEO objects, see the plot of nrev versus cost function in fig. 9.4, where the cost function minima obtained are similar to the GEO cost function minima for the same 1-2 orbital revolutions. Given the sparsity of the data and the interdependence of the shape, orientation and location parameters, further investigation is recommended through sensitivity analysis. This analysis should isolate the impact of orbital parameters and observation strategies.

9.1.2. Gating

Having obtained the distribution of the minimum cost function, a threshold can be set for the corresponding correlation status. As described in section 4.1, the theoretical distribution of the Mahalanobis distance is chi-squared. Thus given the degrees of freedom, in this case four, an upper bound or threshold can be set for which the corresponding fraction of the total population should lie. Previous plots such as fig. 9.2 and fig. 9.3, must be interpreted with care, as the cost function and bins are in $\log_{10}$ scale, seeming to indicate a significant number of occurrences with high cost functions. This effect is however, exaggerated due to the non-linear bin widths covering exponentially larger ranges of cost values, making it seem like there is a concentrated distribution at high-cost function values. While the number of cases with cost function values spanning several orders of magnitude is greater than for the expected $< 10^2$ values, the distribution in reality shows a flatter tail. This can be illustrated when adjusting the frequencies by the bin width w_b and total number of occurrences N, obtaining the relative frequency as

$$f_r = \frac{f}{w_b + N}. \tag{9.1}$$

Accordingly, the higher absolute frequencies observed at the larger bin widths are scaled down to equally compare with the frequency of the lower bin widths. fig. 9.5 shows the relative distributions for the L-BFGS-B method applied to the GEO population.



(a)

(b)

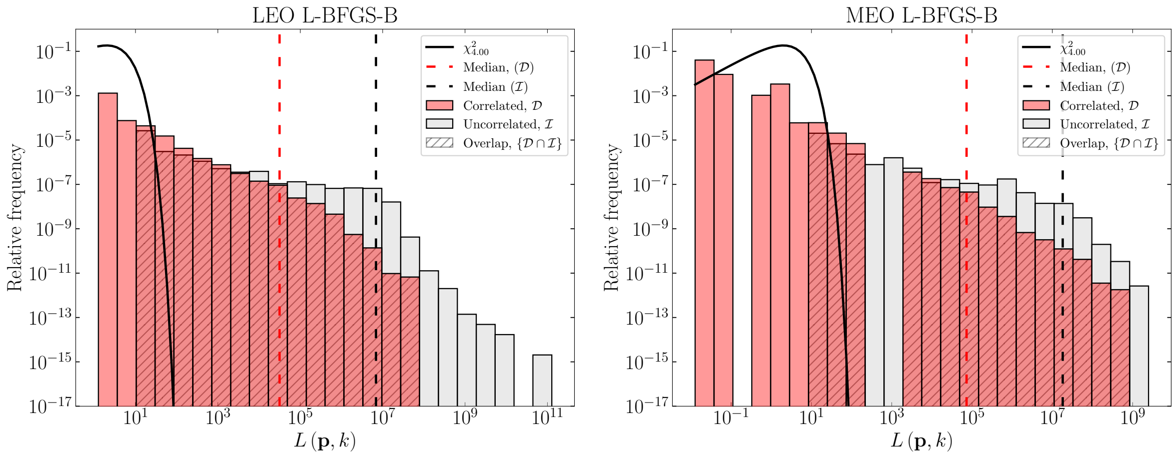
Figure 9.5: Relative frequency distribution for more truthful interpretation of the cost function distribution. Note that the relative frequency and cost function values in a) are plotted in log-scale. In b) the cost-function is linear and relative to the distribution truncated at $L(\mathbf{p}, k) = 20$.

The corresponding distribution more accurately represents the large number of occurrences at lower cost-function values. However, it does not precisely represent the chi-squared distribution, especially for the LEO and MEO regimes as can be seen in fig. 9.6. Accordingly, when defining the theoretical upper-bound containing 95% of the true correlations the true positive rates are not directly reflected in the same quantities, see table 9.1.

Table 9.1: True positive rates obtained for 95% chi-squared theoretical gate: 9.488.

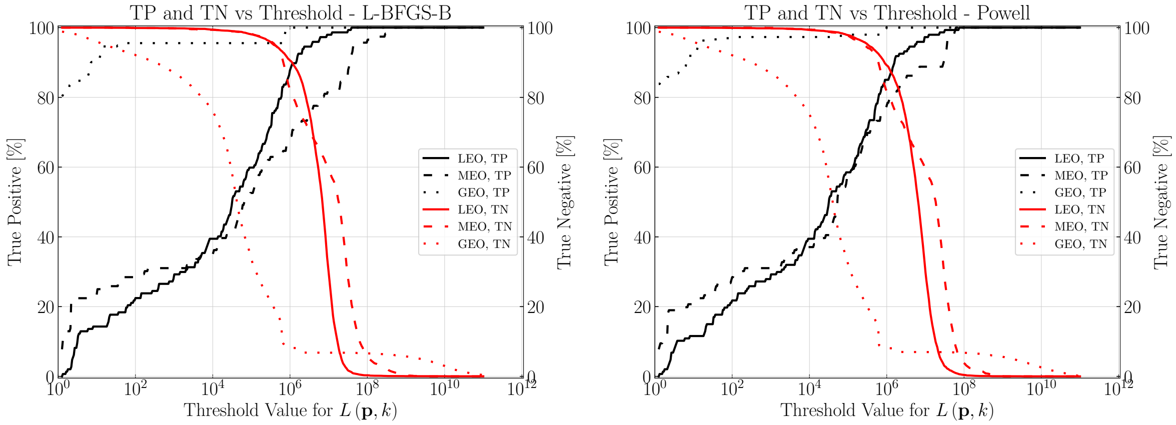
Method	**Regime**	**True positive rate [%]**
Powell	LEO	11.56
L-BFGS-B	LEO	14.29
Powell	MEO	20.69
L-BFGS-B	MEO	23.28
Powell	GEO	91.92
L-BFGS-B	GEO	90.48

The discrepancy with the theoretical chi-squared distribution is expected to be primarily due to the measurement noise of 2 arcseconds. Siminski showed the impact of measurement noise and modelled noise on the distribution, where for increasing noise levels the distribution skews to the left as observed in fig. 9.5b [60]. To see the impact of the gate, the true positive and false negative are plotted against variable threshold in fig. 9.7. Here, the impact of setting a low threshold reflects the large true negative rate, and consequently



(a) (b)

Figure 9.6: Cost function-value distribution for LEO and MEO, showing the discrepancy between a theoretical chi-squared and obtained distribution.



(a) Correlation accuracy for the quasi-Newton optimization method.

(b) Correlation accuracy for the zeroth order optimization method.

Figure 9.7: Effect of gating on the true positive and true negative correlations for different optimization methods.

low false positive rates as shown by the red curves across all regimes. As the gate is increased, more cases are determined correlated, showing the increase in true positives accompanied with a decreased true negative rate. As before, a clear difference in the cost function values is observed between LEO, MEO and GEO.

To represent the correlation performance more clearly the balanced Matthews correlation coefficient (MCC) is used. The MCC captures the correlation performance by combining the true and false positives and true and false negatives into a single number, namely

$$\mathrm{MCC} = \frac{\mathrm{TP} \cdot \mathrm{TN} - \mathrm{FP} \cdot \mathrm{FN}}{\sqrt{(\mathrm{TP}+\mathrm{FP})(\mathrm{TP}+\mathrm{FN})(\mathrm{TN}+\mathrm{FP})(\mathrm{TN}+\mathrm{FN})}}. \tag{9.2}$$

It is defined between 1 and -1, where 1 indicates a perfectly accurate prediction, 0 no better than random guessing and -1 total disagreement between the predicted and true tracklet associations. The plot (fig. 9.8) provides a similar representation as the true positive and true negative plots, in where the maximum prediction accuracy is achieved at the crossover point of maximum TN and TP. Thus for maximum prediction accuracy, i.e. maximum true negative and true positive rates, the corresponding gates are best selected for the correlation gating.

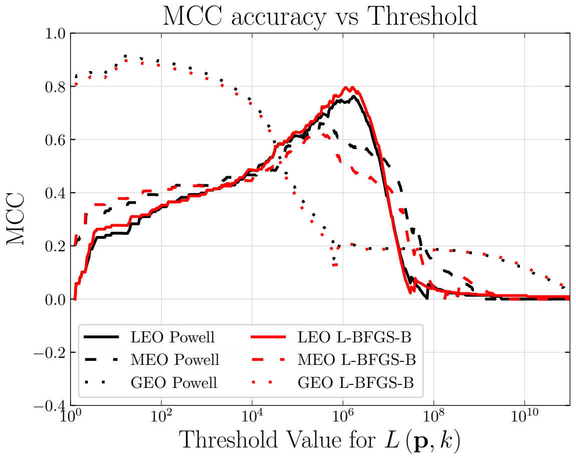


Figure 9.8: Balanced Matthews correlation coefficient (MCC), showing combined accuracy of correlation for both optimization methods.

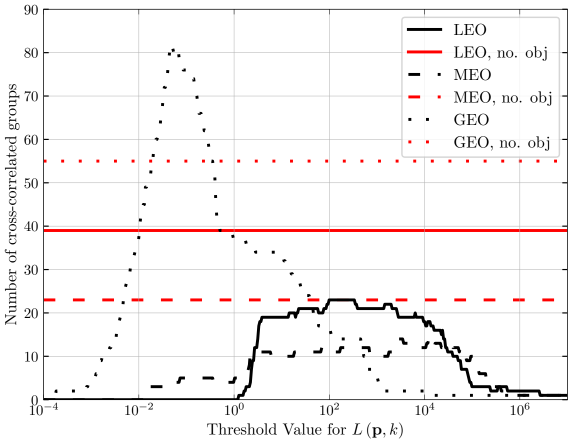


Figure 9.9: Number of cross-correlated groups for each regime from the BVP results obtained with the quasi-Newton optimization method. Horizontal lines indicate true unique number of objects in populations.

The maximum accuracy gates found are tabulated in table 9.2. The distributions share large similarities between optimization methods, minor difference are seen in maximum obtained accuracies where L-BFGS-B shows to obtain higher MCC accuracy for LEO altitude regime. As for the GEO regime, this difference is less evident, and both methods obtain a closer gate and maximum accuracy.

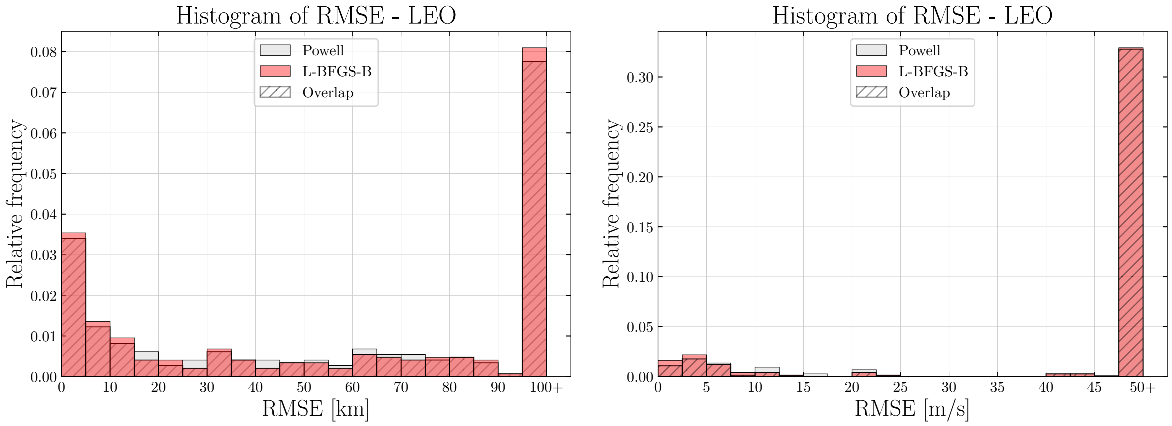
Table 9.2: Obtained gates for maximum Matthews correlation coefficient.

Method	**Regime**	**Gate [$\log_{10}$]**	**MCC accuracy [%]**
Powell	LEO	6.2262	76.60
L-BFGS-B	LEO	6.2065	79.82
Powell	MEO	5.4855	66.32
L-BFGS-B	MEO	5.4768	62.73
Powell	GEO	1.1567	91.56
L-BFGS-B	GEO	1.1567	89.91

While the gate for the maximum prediction accuracy maximizes true rates, it is more sensible from a theoretical approach to gate on more conservative values. For example around the same sigma, or cost-function, value of GEO in the order of 10. Given the current results, this lower gate obtains less false positives and consequently more true negatives and less true positives. The benefit of a more sensible distance gating can be illustrated by showing the number of cross-correlations obtained. The cross-correlations are identified when for example measurement A to B as well as B to C are estimated correlated and thus consist in a common group containing measurement A, B and C. The impact of gating on the number of cross-correlated groups can be shown in fig. 9.9. Here, the impact of a too high gate is shown by obtaining a single group containing all considered measurements. When the threshold is sufficiently high, each measurement pair is correlated and thus estimated part of the same object. On the other hand, if the gate is too low, it shows to not be able to correlate outside of the considered pairs such as those for larger revisit days or more difficult geometry obtaining higher cost function minima. Accordingly, it correlates only the pairs and obtains a large number of groups, erroneously estimating larger number of unique objects. Somewhere in between these cases is the optimal gate, in which the unique number of objects is estimated most accurately, with the false positive rate at a manageable level to not cross-correlate every other pair with each other. These considerations are especially valuable for orbit determination, in which grouped measurements could be used for orbit refinement. This choice however depends on the sensitivity of the following processing step to false positives. Regardless, as the distributions are not reflecting the theoretical chi-square distribution, there needs to be a better understanding on the impact of the orbital regimes on the distribution to allow for gating without a priori, or truth-information.

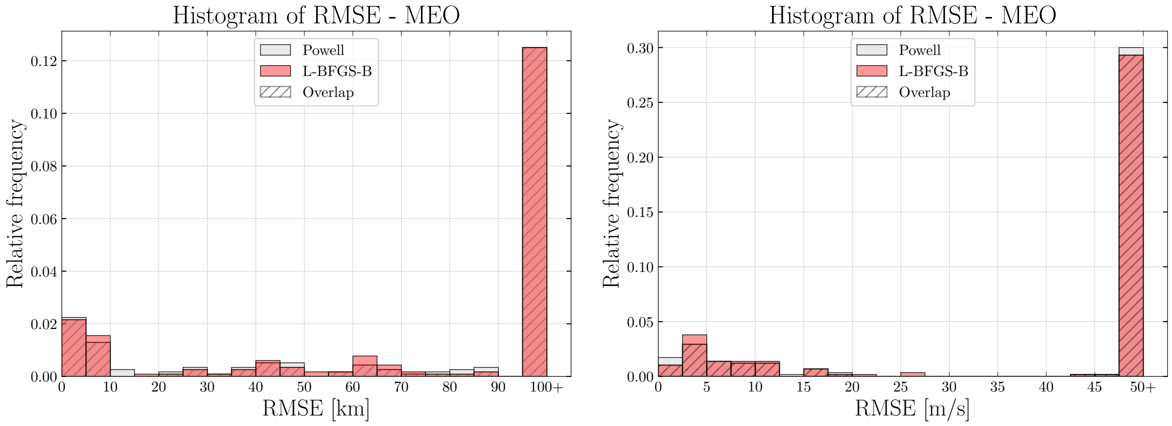
9.1.3. Estimation accuracy

The estimation accuracy of each true correlation is expressed by the root mean square error between the obtained estimate and true state. The distributions are shown in fig. 9.10 as well as function of the angular separation between observations in fig. 9.11.



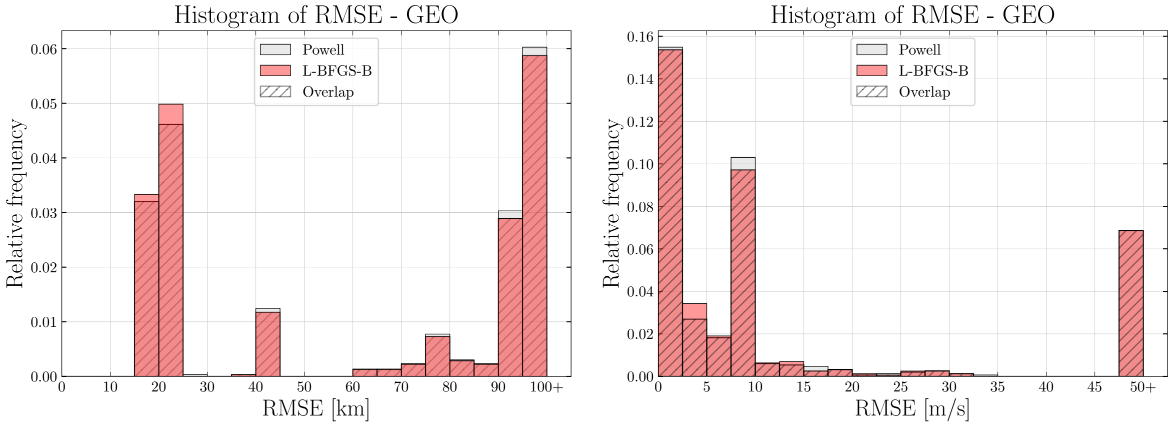
(a) RMSE in position.

(b) RMSE in velocity.



(c) RMSE in position.

(d) RMSE in velocity.



(e) RMSE in position.

(f) RMSE in velocity.

Figure 9.10: The root mean square error in the states obtained for each true correlation for LEO, MEO and GEO regime.

As can be seen in fig. 9.10, virtually no differences are observed between optimization methods, highlighting the dependence of the cost-function representation on performance. Accordingly, latter plot is presented for L-BFGS-B only and the full plots can be found in appendix C. A large fraction of the correlated cases obtain significant estimation errors, in accordance with the obtained cost function values. Again, the GEO regime shows relatively less cases with excessive errors compared to LEO and MEO. The concentration of 15 - 25 km and 0 - 2 m/s rmse for GEO is in accordance with the results of [60], highlighting the discrepancy between two-body and SGP4 dynamics. This discrepancy is the difference in force models used between the estimation method using two-body dynamics and the SGP4 model used to simulate measurements. As SGP4 accounts for Earth's zonal harmonics, Earth's atmospheric drag and third body effects of the Sun and Moon in addition to the two-body dynamics used only in the estimation methods. As the primary perturbation, the Earth's atmospheric drag, is not included, the lower altitude objects reflect significant state errors for a larger group of correlation cases, see fig. 9.10. For the GEO objects, fewer cases show large outliers of state errors, due to a less critical force model discrepancy given the orbital period and revisit times. However, the obtained distribution shows more outliers, i.e. a fraction of GEO state errors above 100 km and 50 m/s, compared to the work of Siminski [60].

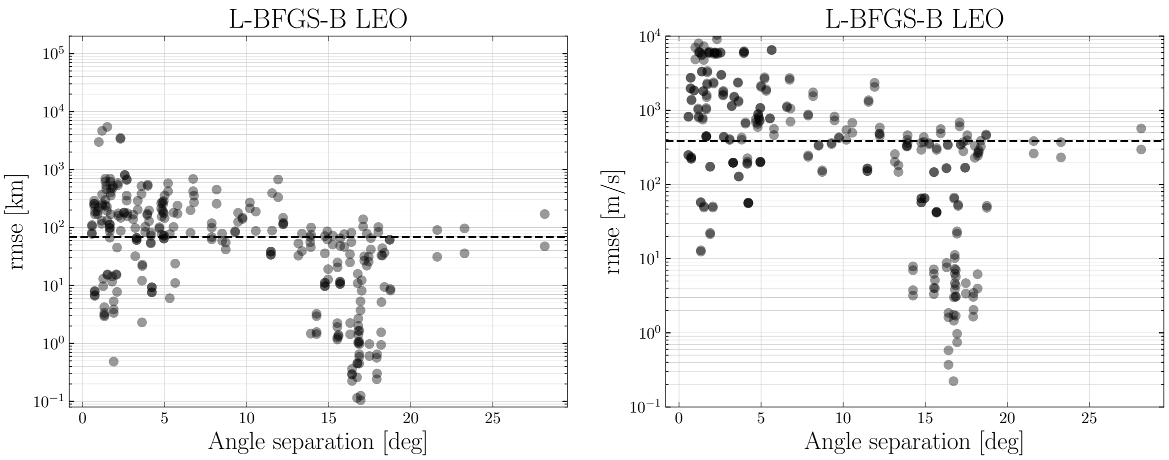
The relation with angle separation (fig. 9.11) is again visible for the GEO regime, where increased angular separation allows for better estimates in the states. And small angle separations lead to difficulties and larger errors due to the plane uncertainty of Lambert's solver. A similar trend can be observed to a lesser extent for the LEO and MEO regimes, where mostly the lower band of errors decreases for increasing angle separation.

9.1.4. Run-time performance

The run time of each population is presented in fig. 9.12 and the average run-time is compared for each orbit regime and optimization method in table 9.3. The largest impact on the run-time is expected to be the number of optimization iterations required for the feasible Lambert parameters, such as orbital half revolutions, flight direction and low or high path for multiple solutions. This would explain why for GEO objects the average run-time is lower compared to the LEO and MEO populations, as it is defined on a smaller semi-major axis interval. Additionally, the initial guess is based on the semi-major axis bounds and maximum eccentricity, for which GEO obtains a relatively precise initial guess in most cases. For MEO the semi-major axis interval is the largest and thus allows for potentially high number of iterations required due to a bad initial guess. The average run-time of LEO is the largest, however, and seems to decrease for corresponding MEO and GEO regime. It is less clear what the exact cause is for the longer runtimes in LEO, one possibility is the denser orbital population in LEO, which may lead to more feasible solutions that require additional computational effort to resolve, unlike other regimes where solutions can be more readily dismissed. The difference in computational time for Powell and L-BFGS-B regimes is primarily explained with the coverage, i.e. the number of cases that converged. For the simulation the global minimum cost function is sought for multiple discrete optimizations, where each feasible condition is tried and the overall best parameters are stored with corresponding cost function value. Whereas Powell would obtain no feasible numeric value, L-BFGS-B would obtain a numeric value and thus cover more cases. It is expected that this discrepancy can be tuned, such that the L-BFGS-B method breaks off at an earlier stage for uncorrelated cases. Currently this is not included as to allow to obtain the full distribution of numeric values for both correlated and uncorrelated cases. In the case for GEO orbits, the initial guess was typically closer to the minimum and topography is better behaved. This shows the computational time to be closer and actually better for the quasi-Newton method, benefiting from the topography. Additionally, for the more difficult regimes Powell's zeroth-order method is seen to be more efficient.

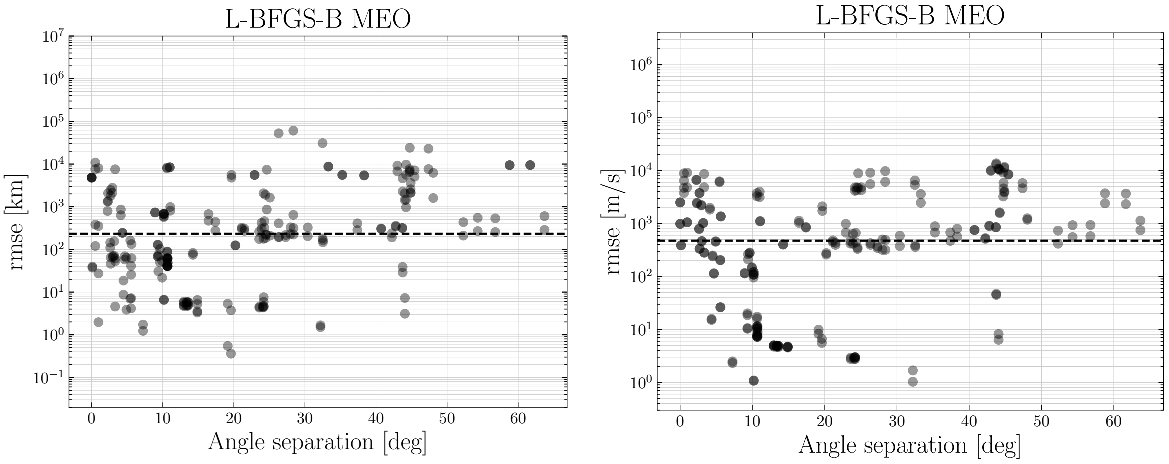
Table 9.3: Comparison of Methods and Orbits

Method	Orbit	Coverage [%]	Average Ct. time [s]	No. of cases
Powell	LEO	77.24	3.11	6555
L-BFGS-B	LEO	89.38	6.69	6555
Powell	MEO	90.91	2.21	3003
L-BFGS-B	MEO	99.90	4.42	3003
Powell	GEO	59.01	0.70	30628
L-BFGS-B	GEO	60.52	0.53	30628



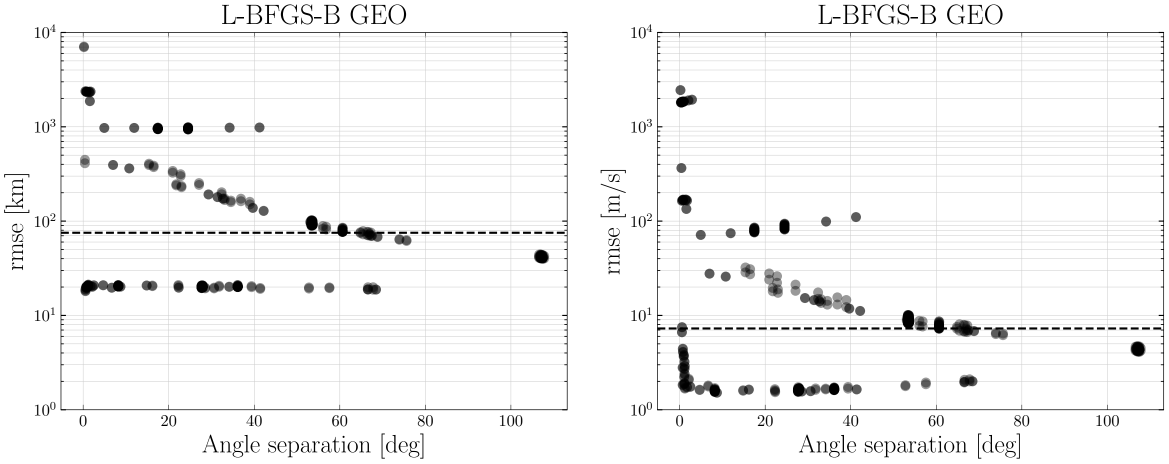
(a) RMSE in position.

(b) RMSE in velocity.



(c) RMSE in position.

(d) RMSE in velocity.



(e) RMSE in position.

(f) RMSE in velocity.

Figure 9.11: The root mean square error against angle separation for each true correlation. The dotted line shows the median value of the whole correlated population.

9.2. IOD validation step

Having obtained the estimated correlations, the classical initial orbit determination methods are applied to investigate their effectiveness on validating or reducing false positives. The data used for this is the batch of correlation cases that fall below the threshold set for maximum accuracy as shown in previous section. The

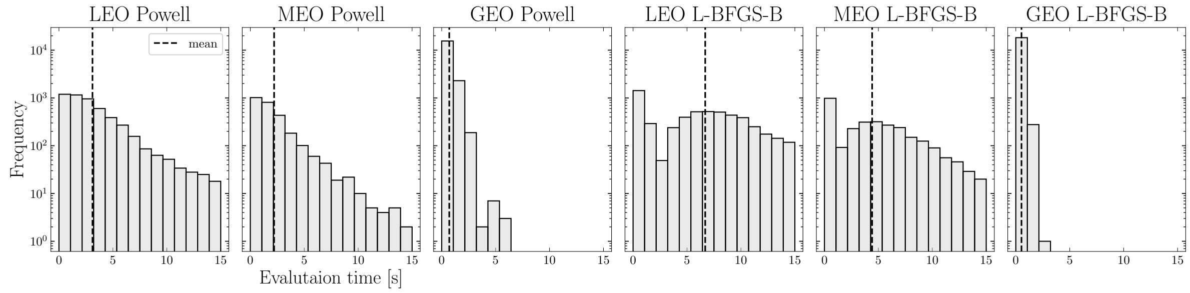


Figure 9.12: Distribution of the run-time per correlation case.

maximum accuracy gate is set for the sake of analysis, in which a more noticeable number of false positives is obtained and the effect of the following approaches may be studied to reduce false positives. Each correlation case consists of two tracklets, whose complete set is now considered for initial orbit determination based on three angles. As the tracklets are too short, the IOD methods are not applied to the individual tracklets but instead to the angles contained in the combined correlated tracklet, with the aim to increase orbital coverage. For Gauss and Gooding the first and last angles of the first tracklets are used and the last angles of the second tracklet. For $\mathrm{L_n}$ Gauss, the full angle set is used. The steps included in this approach are depicted in fig. 9.18.

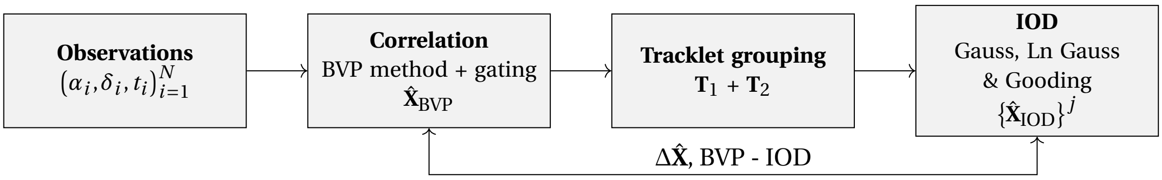


Figure 9.18: Overview of the procedure for the IOD validation step, where j denotes the number of state estimates obtained depending on a three angle only $j = 1$ or $\mathrm{L_n}$ Gauss method $j = N - 2$.

The number of pairs estimated to be correlated by the BVP method is 868, 182 & 1153 for LEO, MEO and GEO respectively. Of these estimations, still 732, 109, 855 are falsely estimated. To reduce this false positive rate, the IOD methods are applied to the same number of cases to investigate any relation between the results obtained by the BVP and IOD methods. And investigate whether re-considering the full angle-set shows any benefit in rejecting false positives.

For all IOD methods the solution is compared to the initial estimate obtained by the BVP correlation method. Specifically, the results obtained using the L-BFGS-B optimization method are used. The comparison is performed on the Keplerian elements, as the state estimates do not correspond on epoch denying direct state comparison. This way the difference in estimated orbital shape, eq. (9.3), and orientation, eqs. (9.4) and (9.5), of the estimated orbits are compared. Latter is done with the angular momentum vectors and longitude of periapsis, the sum of the right ascension of the ascending node and argument of perigee. The estimated true anomaly, or the argument of latitude are not considered for comparison as the epoch of the estimated states do not directly correspond. Accordingly, the semi-major axis

$$\Delta a = \left| \frac{p_1}{1 - e_1^2} - \frac{p_2}{1 - e_2^2} \right|, \tag{9.3}$$

the orbital plane orientation

$$\Delta\theta_h = \arccos \hat{\boldsymbol{h}}_1 \cdot \hat{\boldsymbol{h}}_2, \tag{9.4}$$

and the longitude of periapsis

$$\Delta\varpi = |\,(\Omega_1 + \omega_1) - (\Omega_2 + \omega_2)\,|, \tag{9.5}$$

are compared (figs. 9.22, 9.23 and 9.25). For the batch least squares instead the final obtained root-mean-square on the measured and modelled angle is taken as reference.

While already presented in previous section, the state estimate accuracy for the estimated correlations is again presented in the same format as will be shown for the IOD and OD methods. The same values are shown, however the state errors are shown for only the gated data set, whereas the state errors shown for BVP

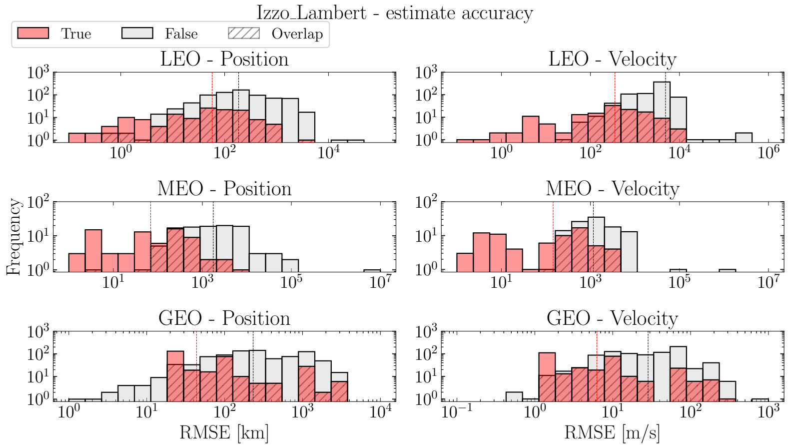


Figure 9.19: State estimate accuracy for BVP gated measurements, rmse between true (SGP4) and estimated states.

in previous section contain the full set of cases. To interpret the accuracy obtained from the BVP method, the median error are taken for the true correlations. While the distributions do not show a left skewed distribution and the median thus does not always reflect the distribution well, for cases that show skewed distributions it is still seen to more accurately reflect the relative performance of the methods.

The accuracy for each method is compared to the accuracy obtained from the BVP correlation method. table 9.4 presents the median values obtained for each method on the subset of estimated correlations, as well as the coverage. The coverage is determined by the fraction of cases obtaining a solution, indicating to which degree the method is applicable to the considered observation conditions.

Table 9.4: Median RMSE, converged cases (Cvg.) and computational time (Ct.) for each method and regime between true states (SGP4) and true positive estimated states.

Method	**LEO**				**MEO**				**GEO**			
	Cvg. [%]	**r** [km]	**v** [m/s]	**Ct. [s]**	**Cvg. [%]**	**r** [km]	**v** [m/s]	**Ct. [s]**	**Cvg. [%]**	**r** [km]	**v** [m/s]	**Ct. [s]**
Gauss	11.64	3.92	18.45	3.0	17.03	646.81	116.84	44.0	9.37	4948.32	378.93	60.0
L_n Gauss	8.29	4.9	20.4	29.0	8.79	8579.67	1859.2	289.0	0.69	19259.39	1832.96	117.0
Gooding	95.97	70.6	4055.87	2.0	96.70	95.06	1167.45	2.0	100	89.5	9.16	2.0
BVP	-	57.49	354.71	-	-	68.91	145.33	-	-	43.63	6.26	-

The comparison in accuracy helps interpret the difference in results shown between BVP and IOD methods in following sections. Large differences in orbital parameters can be concluded to indicate a worse result in general, as the BVP method shows to return overall a better estimate. Nevertheless, the main goal for the estimate comparison is toward validation purposes of the obtained BVP correlation, ideally rejecting false positives. Each method is briefly discussed for state accuracy and validation of BVP results in the following subsections.

9.2.1. Gauss

As shown in table 9.4, Gauss and L_n Gauss perform remarkably well for LEO compared to other methods, with L_n Gauss showing slightly higher errors. When looking at the distribution corresponding to the state accuracy, a concentration of true correlations is found below order 2 magnitude errors for LEO (see figs. 9.20 and 9.21). On all other regimes it obtains significant state errors, where again the three angle classical Gauss outperforms the L_n Gauss approach. The seemingly high performance at LEO is mainly attributed due to the small number of cases being handled by Gauss. The cases that it is able to estimate for are however accurate and explain the low median error. For both the Gauss and L_n Gauss this highlights their limitation on single revolution measurements and consequently revisit time. A large number of cases are characterized by large time-steps, relative to the estimated orbital motion $n \propto \sqrt{1/a^3}$. Between the tracklets the semi-major axis for the universal anomaly

$$x = \alpha\tau\sqrt{\mu}, \quad \alpha = \frac{2}{r_n} - \frac{v_n^2}{\mu} \tag{9.6}$$

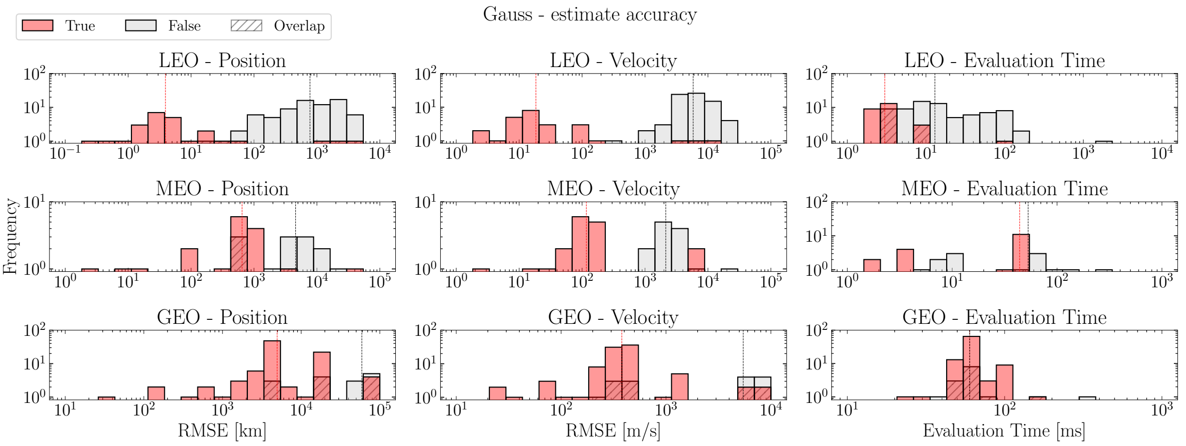


Figure 9.20: State estimate accuracy for Gauss on BVP gated measurements, rmse between true (SGP4) and estimated states. For false correlations, the true state is taken to be the state of the object with the corresponding middle epoch measurement.

computation becomes too large and denies any universal anomaly solution. Accordingly, in both approaches it is only able to handle single revolution observations with sufficient revisit-time between the tracklets, and thus handles only a few number of cases. The majority of the cases it can handle are within the same night and show large differences in elements to the BVP estimates. The average run-time for both implementations is typically below one second, where the three angle Gauss method is shown to obtain the solution faster compared to the L_n Gauss approach in most cases.

For both the Gauss and the L_n Gauss approach (see figs. 9.20 to 9.23) there is overall a distinction visible between true positives and false positives when looking at their respective median values. However, in addition to the limited application of Gauss toward single night observation, there is no clear indication on elements to reject false positives and allow to validate results with added information. As for L_n Gauss, the estimate obtaining the smallest difference in semi-latus rectum between BVP and L_n Gauss is taken from the total number of range estimates. Still, as also seen in section 7.2, it does not benefit from the full angle set, as it does not add meaningfully toward the available information to rule out false positives and thereby increase accuracy. As for the L_n Gauss method, larger errors are primarily seen due to the short angle separations for each range estimate triplet.

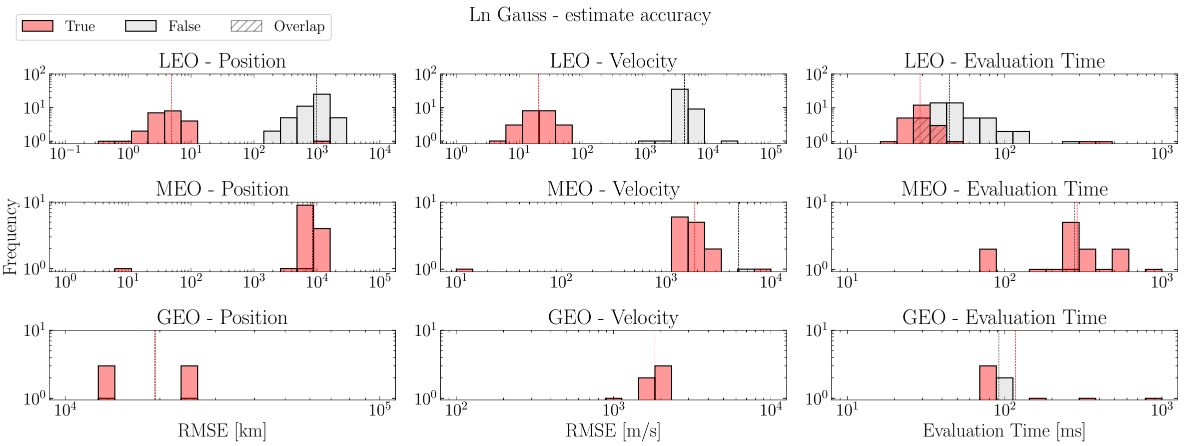


Figure 9.21: State estimate accuracy for L_n Gauss on BVP gated measurements, rmse between true (SGP4) and estimated states. For false correlations, the true state is taken to be the state of the object with the corresponding observation epoch measurement.

For all regimes, the semi-major axis and longitude of periapsis show significantly different estimates, especially for LEO and MEO considering the difference relative to the typical altitudes. For the orbital plane orientation, the differences are more contained for the correlated cases in which GEO shows higher differences compared to the lower altitude objects. This could perhaps reflect Gauss' ability to reflect the orbital plane orientation relatively well, i.e. position and velocity directions, as also concluded by Schaeperkoetter in an investigation on IOD methods on LEO orbits [55]. For Gauss, the resulting population contains 26, 18, 93 positives and 75, 13, 15 negatives for LEO, MEO, GEO respectively. For L_n Gauss, 24, 15, 6 positives and 48, 1, 2 negatives for LEO, MEO, GEO are obtained. The resulting dataset of the correlation estimates is thus significantly reduced due to the inability to handle multi-revolutions. In addition, Gauss' method does not prove useful for validation of the correlation results, as there is no clear relation for the correlated estimate differences allowing to reject false positives.

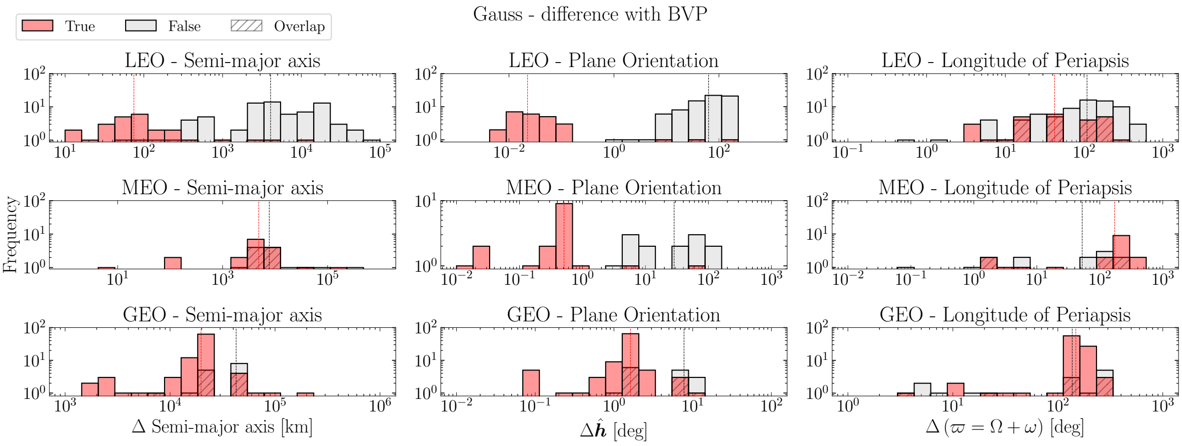


Figure 9.22: Difference in Keplerian elements between Gauss and BVP estimates.

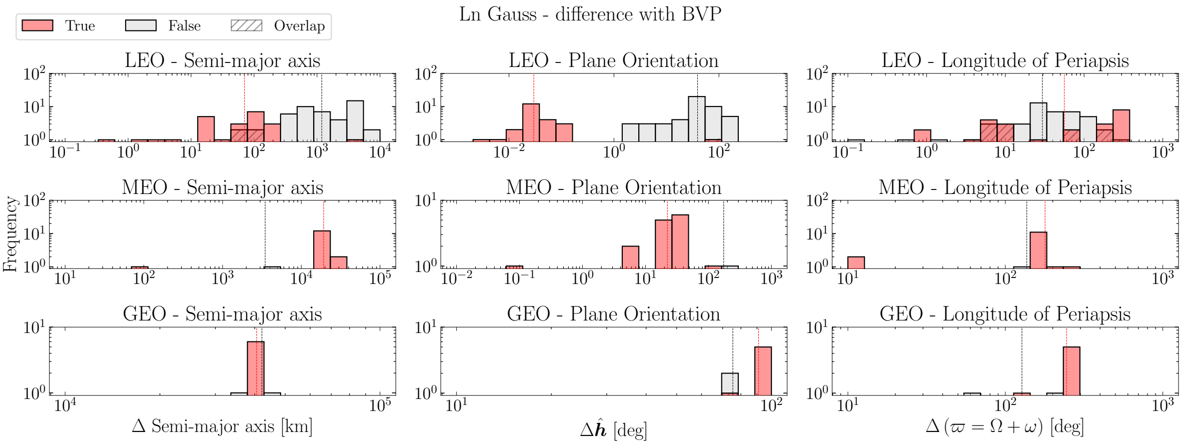


Figure 9.23: Difference in Keplerian elements between L_n Gauss and BVP estimates.

9.2.2. Gooding

Unlike Gauss' method, Gooding is able to handle multi-revolutions due to the usage of a Lamberts solver. Accordingly, the estimated results from the BVP method are directly supplied as inputs for Gooding's method. As Gooding relies on three angle pairs, with sensitivity on the initial guess of the outer ranges [65], the three most orthogonal angles are set. The initial range guess is defined by the estimate obtained from the BVP correlation as well as the flight direction and number of orbital revolutions.

Furthermore, as the nature of Gooding is based on three angles instead of the outer ranges as input, the orbital elements are still required to compare the estimated states in general sense. Before the differences are investigated however, it is observed that Gooding shows better accuracy for the position estimates compared to Gauss. Noticeable for Gooding is the large error in the velocity estimates especially for lower altitude regimes, consistent with the observations from Miller and Frueh [42].

While the median errors presented in table 9.4 seem to indicate better performance for Gauss for LEO, it is primarily due to the larger number of cases handled. Gooding is known to show reduced performance for smaller angle separation, with increased accuracy when angular separation is increased above 5 degrees up to

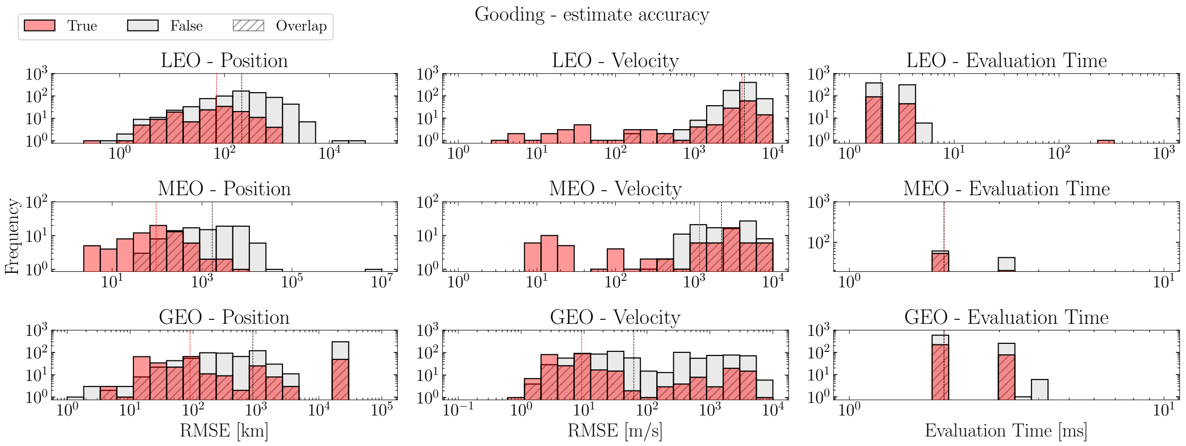


Figure 9.24: State estimate accuracy for Gooding on BVP gated measurements, rmse between true (SGP4) and estimated states. Dashed lines indicate median values. For false correlations, the true state is taken to be the state of the object with the corresponding middle epoch measurement.

90 degree angle separation [55, 42]. On the other hand, Gauss is seen to decrease performance for larger angle separations. This can be attributed to the use of refined Lagrange coefficients and a linear approximation on states with shorter time gaps. In contrast, Gooding uses an iterative boundary-value approach with Lambert's solver, which accommodates multiple revolutions and requires observation geometry outside the full and half revolutions. For all regimes Gooding's method is able to obtain a solution within one second, with typical run-times in the order of milliseconds. It shows to obtain the solution faster compared to Gauss and also remains constant for all regimes.

With the exception of a few false positives in the LEO and MEO populations, the same population is returned as by the BVP method. Thus, in contrast to Gauss, a large part of the population considered is handled due to Gooding's ability to handle multiple orbital revolutions with a Lambert solver. Gooding handled all GEO correlations, and 95.97 % and 96.70 % of the correlation cases for LEO and MEO respectively. For LEO and MEO, the cases it did not obtain consist of a large majority of false positives, 34 of the 35 solutions it did not obtain are false positives for LEO, for MEO all 6 solutions it did not obtain are false positives.

Whereas Gauss showed increasing differences for higher altitudes, Gooding shows the differences to decline for higher altitudes. Still, no clear relation or behaviour is seen that could be used for direct rejection of false positives. As it is able to handle more cases, due to multi-revolutions, it is more usable than Gauss, but

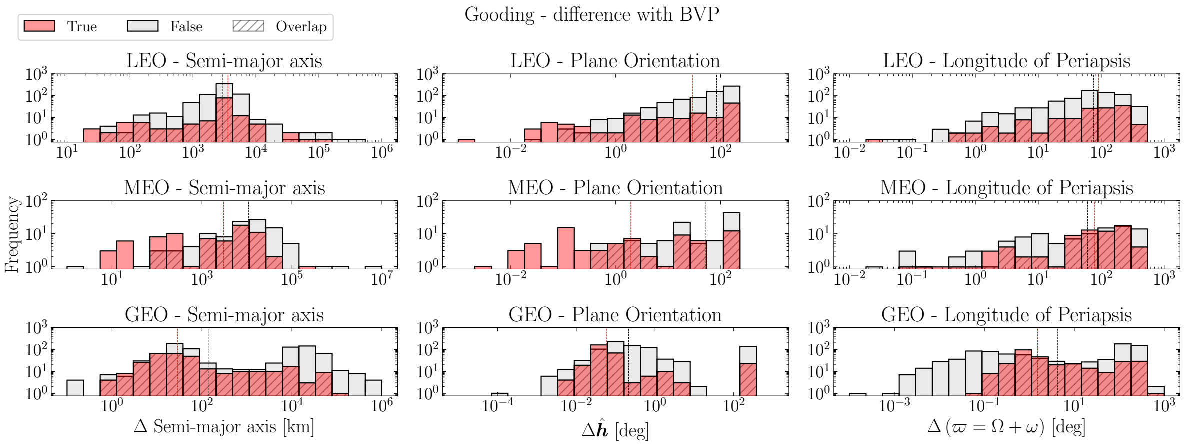


Figure 9.25: Difference in Keplerian elements between Gooding and BVP estimates.

regardless does not add more information when using the most orthogonal angles in a combined correlated set. Additionally, the Lambert solver used in Gooding is based on an older implementation and could therefore also be expected to at best obtain the same order of accuracy at higher computational burden, when purely comparing on the Lambert solver [18].

9.2.3. Alternative validation for IOD

As shown in the previous section, in a practical setting the correlation may be validated by comparing the obtained orbit estimate of the BVP correlation with the IOD method. However, the primary limitation is the assumption that the two estimates are not consistent in their errors when identifying false correlations. This approach cannot exclude cases where large, consistent state errors between the BVP and IOD methods falsely indicate a correct correlation. Therefore, an alternative approach is to use the obtained state estimate to model the measurements not used by the IOD method and compare by the root mean square error between the measured and modelled angles. The comparison with the modelled measurements from the state estimate then takes into account the accuracy of the state in addition to the consistency between methods. The steps involved in this approach are outlined in fig. 9.26.

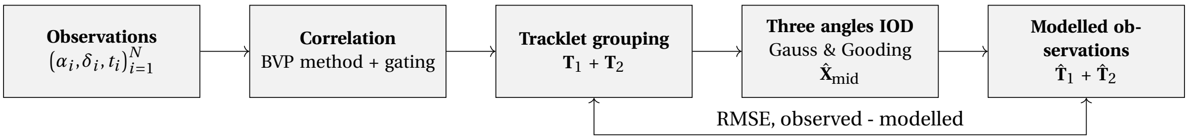


Figure 9.26: Alternative validation method for three angles only IOD methods.

As the method relies on validation of correlation by comparing against measurements, the multiple angles L_n Gauss method is not applicable and would return by definition zero root-mean-square due to the use of each measured line of sight for the system of equations. Accordingly, the alternative approach are applied only for three angle only IOD methods Gauss and Gooding. The results are presented in figs. 9.27 and 9.28.

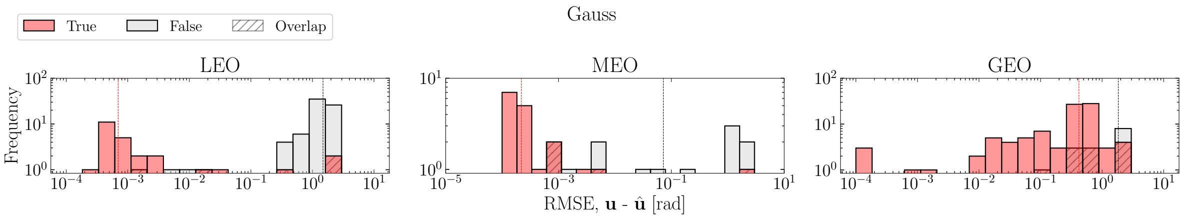


Figure 9.27: Alternative IOD step after BVP correlation for Gauss' three angle method. Dashed lines indicate respective median values.

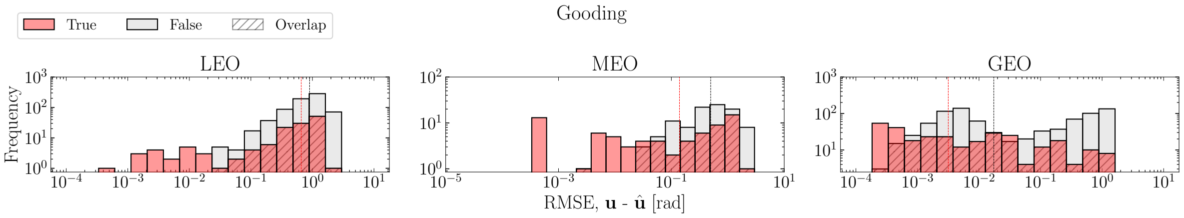


Figure 9.28: Alternative IOD step after BVP correlation for Gooding's three angle method. Dashed lines indicate respective median values.

From the distributions, the distinction between true and false positives remains, although more consistently and concisely captured by the RMSE. The medians and distributions show, similarly to the previous validation approach, a greater distinction between true and false positives for Gauss compared to Gooding in terms of accuracy, difference and RMSE values. Accordingly, similar behaviour to that shown in the previous section is observed when the IOD methods are used to validate the correlation results. This behaviour shows that the results obtained with the IOD methods applied to a correlated set of tracklets do not provide the necessary information to validate and reject false positives. Therefore, the procedure is again a balance between true positives and true negatives and it is best to rely on the BVP correlation results and use a fair uncertainty distance at around 10 maximum.

9.3. BLS validation step

For the orbit determination step, the batch least squares implementation is used. To estimate the residuals it includes third body and J2 harmonics as only perturbations. The BLS considers a maximum of 100 iterations or a difference of less than 10^{-4} in consecutive RMS for convergence. The method is applied to each correlated pair as well as to each group of measurements that are estimated to be cross-correlated. Unlike the IOD step, the BLS aims to refine the state estimates obtained by finding the deviation vector and measurement noise. As is typical for orbit determination methods, it is able to use the complete data of all collected measurements that are estimated to correspond to the same object. Cross-correlation of measurements is defined as a chained-correlation outside of the pair-wise correlations. Thus if a correlation is expected from the BVP step between A and B, and measurement B and C as well, the complete cross-correlated group then consists of A, B and C. If the cost function and selected threshold reflect the correlation accurately, the same number of groups as unique space-objects should be found.

First, BLS is applied to cross-correlated groups, with the gate value set differently from the IOD steps and the pairwise BLS approach. As shown in section 9.1.2, gating in the order of 10^6 σ for the maximum correlation accuracy for LEO and MEO yields large false cross-correlated groups. Therefore, the maximum accuracy gate for GEO is set for all regimes to reflect a more realistic uncertainty distance. The gate is considered from the L-BFGS-B method, where the maximum accuracy was found for GEO at 14.3465 $(\sim 10^{1.16})$. The correlation rates for all regimes at this threshold are tabulated in table 9.5.

Table 9.5: Correlation rates for each regime at the specified threshold.

Regime	**FNR %**	**FPR %**	**TPR %**	**TNR %**
LEO	85.71	0.02	14.29	99.98
MEO	75.00	0.00	25.00	100.00
GEO	5.40	4.69	94.60	95.31

The more conservative gating reflects a high true negative rate, obtaining for MEO a 0% false positive rate. The results of the orbit determination applied to the groups are depicted in fig. 9.29. The true correlation is marked red if the group contains measurements originating from the same unique object, and thus contain no measurements of other objects that are false cross-correlations. From the obtained refined initial state estimate, the BLS does not handle the grouped batch of measurements correctly, despite the high true negative rate. It shows extremely high RMS values for the measurement angles obtained after reaching the maximum number of iterations. Performance of the BLS seems to be limited by the force model and non-linearity of the measurements more than the impact on false positives, as seen in the population of MEO with 100% true negatives. It appears that the measurements are too sparse for the grouped orbit determination method to be refined, preventing it from properly yielding lower state error. The median runtime for each grouped batch is 0.66, 2.36 and 3.34 s for LEO, MEO and GEO respectively.

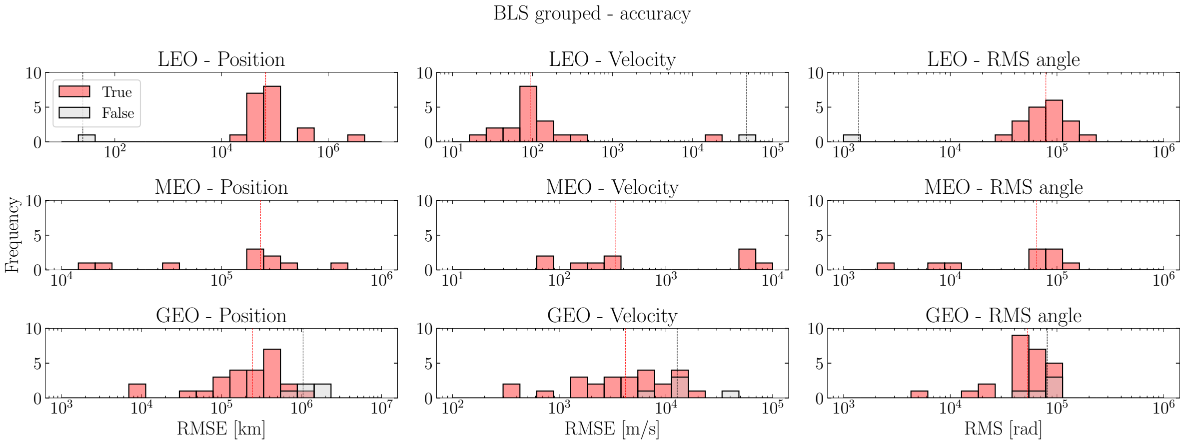


Figure 9.29: BLS applied on grouped cross-correlated measurements for each regime. Measurements are generated based on SGP4 dynamics, BLS includes J2 harmonics and Sun-Moon third body effects.

As an alternative to the grouped approach, the BLS method is also applied to each estimated correlated pair. This reduces the number of sparsely distributed measurements within a single batch for the OD method. For this approach, the gate at maximum accuracy is used as is done similarly to the IOD steps to investigate the effect on false positives. This means the correlation rates are such that the true positive and negatives are maximum for each regime, but as consequence contain significantly more false positives. The obtained state errors between the refined initial state from the BLS and the true simulated states are plotted in fig. 9.30 Still large errors are seen, and similarly to the IOD steps, no clear false positive rejection values are shown. While the state error are significantly lower compared to the grouped refinement, for GEO the median rmse is still in the same order as obtained from the BVP method and thus does not show any refinement in estimates. The lower error are believed to be due to the higher occurrence of smaller time-gaps in-between tracklets correlated, and thus the BLS converges more often. The computational time of this pair-wise approach using the BLS algorithm is significantly more expensive compared to the IOD methods, see fig. 9.31. Due to the larger number of run-cases, i.e. each pair, the median run-times are significantly higher compared to the grouped approach; 8.81, 3.12 and 0.73 s for LEO, MEO and GEO respectively.

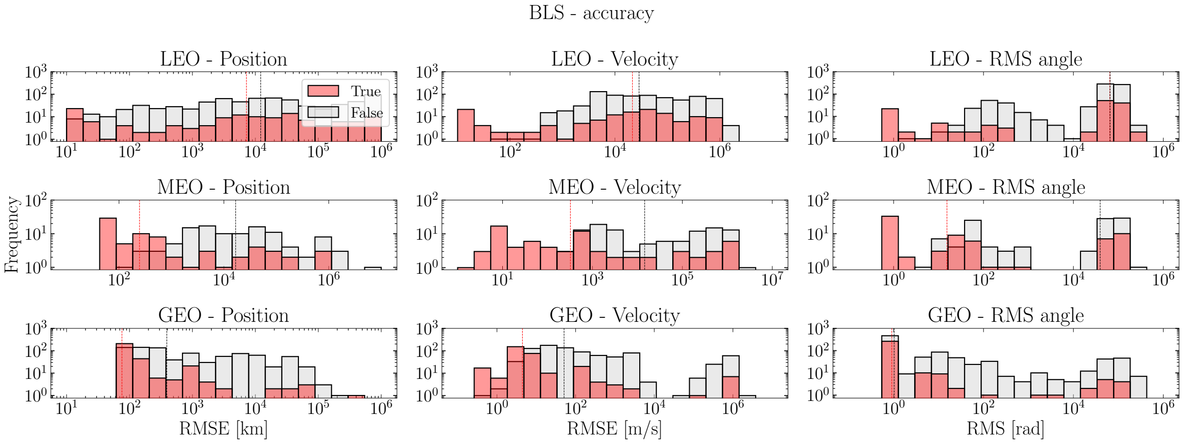


Figure 9.30: BLS applied on correlated measurement pairs for each regime. Measurements are generated based on SGP4 dynamics, BLS includes J2 harmonics and Sun-Moon third body effects.

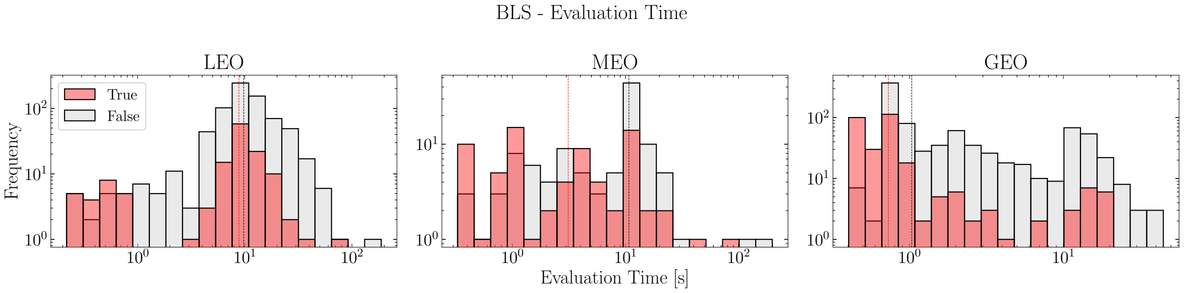


Figure 9.31: Median computational time for the pair-wise BLS approach.

Overall, the validation step is not considered a fruitful approach toward validation of the tracklet correlation results obtained by the BVP method. This is primarily due to the inability of the methods to properly use the information contained in the tracklets. It shows that the attributable approach for the too-short arcs is the best approach for initial orbit determination compared to an multiple angles approach. Gooding's method was seen to be the most promising candidate for validation of tracklet correlation results. Primarily due to its relatively good accuracy for the required computational time. The BLS implementation did not properly handle the correlated measurements, which is expected due to the force model discrepancy and sparsely distributed measurements. The impact of the correlation accuracy on the orbit determination process remains unclear. Additional work is required to investigate the impact and required estimation accuracy on the orbit determination process, investigating the impact of measurement sparsity and higher force model fidelity for the estimation method. Common among all methods is the inability to reject false positives, or outliers, by extracting more information when reconsidering the raw tracklet angles. Reducing the false positives was only achieved at the cost of lower true positives, and thus requires the same procedure as is already performed during tracklet correlation.

10

Conclusion and recommendations

The main goal of this thesis was to explore the potential of existing estimation methods and modify the cataloguing pipeline so as to improve its robustness while minimizing the associated computational expense. The cataloguing robustness is defined as the ability to capture truthful tracklet correlations for the catalogue build-up of newly identified or re-observed space-objects. In particular, by achieving high true positive and true negative rates for tracklet correlation. This thesis therefore provides an answer to the main research question:

> "How can the robustness of the cataloguing pipeline be improved when considering application of orbit estimation methods to the full angle set of short observation arcs?"

To provide an answer to the main research question, a baseline tracklet correlation approach is implemented. The approach developed by Siminski [60] is included, referred to as the Boundary Value Problem (BVP) formulation within the Admissible Region (AR) framework. This correlation method estimates an object's state using angular observations of right ascension α and declination δ at two epochs, t_1 and t_2, derived in this work from an optical ground-based sensor with noise of around 2 arcseconds. By fitting these angular measurements to a quadratic curve, mean angles and angle-rates are determined, forming an attributable. To estimate the object's six-parameter state ($\mathbf{r}$ and $\dot{\mathbf{r}}$), hypothesized ranges $\hat{\rho}_1$ and $\hat{\rho}_2$ are combined with the outer mean angles, creating a measurement vector $[\alpha_1, \delta_1, \hat{\rho}_1, \alpha_2, \delta_2, \hat{\rho}_2]$. The admissible region in ρ_1-ρ_2 is sampled, and a Lambert solver derives the corresponding velocity vectors, yielding the object's state at t_1 and t_2. A cost-function based on the Mahalanobis distance evaluates the quality of hypothesized ranges, minimized through numerical optimization to identify the best range estimates. The correlation between measurements is judged using this minimum cost-function value, with thresholds set for desired true positive rates.

To validate the correlation process and enhance cataloguing robustness, multiple initial orbit determination (IOD) methods were evaluated using full tracklet data. The multiple angles-only Gauss method, proposed by Karimi and Mortari, employs a least-squares approach to estimate ranges for all measurement epochs, utilizing the entire tracklet [34]. Its implementation was verified against the published results and compared to Gauss' three-angle method, which when compared to the most-orthogonal three angles in Gauss, demonstrated similar accuracy at higher computational effort.

Gooding's three-angle method, using a Lambert solver for multi-revolution orbits, was also tested, along with a Batch Least Squares (BLS) orbit determination method to investigate the balance between acquired state accuracy, computational burden and sensitivity to detect correlation. These methods were applied to results from the BVP correlation method and compared on estimation accuracy, orbital element differences, root-mean-square on the filtered angles as well as computational time.

Simulated measurements were generated from reference TLEs for LEO, MEO, and GEO objects, propagated over three observation nights with the SGP4 model, accounting for drag, J_2 harmonics, and third-body effects. Observations, based on Airbus' Robotic Telescope (ART), included visibility constraints and noise of 2 arcseconds, capturing tracklets between 8 to 70 seconds length for LEO and GEO.

The impact on false negative and positive rates on the tracklet-correlation is studied by correlating the complete set of tracklet pairs, obtaining $n(n-1)/2$ total number of combinations. To optimize run-time, the process was parallelized for both numerical optimization with Powell's zeroth order line-search and a quasi-Newton method. The evaluations times for each correlation case are 0.58–7 seconds on average. In general there was negligible difference in obtained results for the two considered optimization methods. The correlation procedure was seen to work best for GEO, where the cost-function reflects the theoretical chi-squared distribution best. At a 95% certainty threshold, a true positive rate of around 90% was observed. The maximum estimation accuracy, i.e., the maximum true positive and true negative rates, for GEO was achieved at reasonable uncertainty gating of 14.34 σ. For LEO and MEO, higher Mahalanobis distances and cost-function thresholds in the order of 10^6 were required for their respective maximum accuracies. At lower altitudes, higher minima of the cost function are observed, along with a generally wider distribution. The results for lower altitude orbits highlights the challenging regimes for restricting observations. Whereas GEO objects were typically re-observed within 1 day for a maximum of 1 completed revolutions, the lower altitude objects completed about 10 times more orbital revolutions in between observations. Additionally, the force-model discrepancy in the estimation methods is more significant for lower altitudes when compared to GEO objects. Regardless, for all regimes, the estimation accuracy improved with greater angular separation, and a clear distinction was found for the median cost minimum between true and false correlations.

When employing classical IOD methods for validation of the correlation results, Gauss' three-angle method is not seen as a suitable method, as it was seen to handle only a small fraction of the considered measurement pairs. The estimated correlations it was able to handle are for a large part for same-night observations, highlighting the limited applicability of Gauss to zero orbital revolutions. The Ln Gauss method, similar to Gauss's three angles, struggled with large time gaps and semi-major axis. Overall, as shown in the comparison of the two implementations, the Gauss three angle approach showed better accuracy in state estimation for shorter run times compared to the Ln Gauss implementation.

Gooding's method, which uses a Lambert solver, is able to handle multiple orbital revolutions and returned an estimate for all cases, except for some (primarily false) correlations at LEO and MEO. Noticeably, Gooding obtained high residuals in the velocity vector but lower median residual in position vector for LEO and MEO. Gooding's accuracy increased with larger angular separation and consistently delivered solutions faster than Gauss in the order of a couple milliseconds.

The Batch Least Squares (BLS) orbit determination method was applied to both the cross-correlated tracklets, consisting of larger groups of correlated measurements, and pair-wise correlated tracklets. The BLS showed to not properly handle the grouped batch of measurements, despite the use of a high true negative population. Essentially no refinement in the states was observed and convergence was not achieved. The large time-steps, and data sparsity between the tracklets seem to prevent the BLS method from refining the orbital states. The run-time for the BLS method is also more expensive in comparison to the classical IOD methods, showing median run time of 0.68 s (LEO), 1.53 s (MEO) and 3.43 s (GEO).

In the pairwise BLS application, the method achieved lower residuals compared to the grouped approach, most likely due to the smaller time gaps between tracklets. In accordance to the initial estimate accuracies obtained by the BVP method, the accuracy in the BLS pair-wise approach decreased in magnitude for higher altitudes. Still, the residuals in the GEO regime remained comparable, but worse for LEO and MEO, to those from the BVP method, showing no refinement. Furthermore, the method lacked clear rejection criteria for false positives, and the obtained RMS values on angle measurements did not provide for a clear validation metric. The pair-wise approach showed median run-times of 8.59 s (LEO), 1.42 s (MEO), and 0.72 s (GEO).

Concluding, the estimation methods did not show a clear relation for false positive rejection and consequent improvement in robustness of the cataloguing pipeline when applied to the obtained results from the tracklet-to-tracklet correlation procedure. The BVP method demonstrates reasonable performance for re-observation outside full orbital revolutions and for sufficiently small revisit times relative to the mean motion. The best approach involves gating for a high true positive rate near the upper bounds of the chi-squared distribution. Lower uncertainty distances, well within the order of 10 σ, are recommended to exclude large numbers of false positives and to avoid false cross-correlations. The impact of the obtained state and correlation accuracy on the BLS orbit determination method was not found due to the inability of the BLS implementation to handle the sparse tracklet data in both grouped cross-correlated measurements and a pair-wise approach.

10.1. Recommendations

Based on the findings in this work, some recommendations are given for future research. First of all, low-altitude objects remain a challenge for ground-based optical observations. While the use of radar capabilities at these altitudes may offer a more pragmatic approach, further investigation of the BVP method applied to these regimes in combination with optical ground-based observations is recommended to confirm the dependency on perturbed motion through inclusion of force models in BVP as to obtain higher accuracy initial estimates and distributions for gating closer to the theoretical distribution.

Second, a deeper understanding of the OD method's role in tracklet correlation is necessary. As observed for the considered IOD and OD methods, the validation methodology is not an appropriate means of efficiently enhancing the robustness of the cataloguing pipeline. Regardless, the obtained accuracy of the BVP method and initial guess, as well as the data sparsity should be characterized for the required initial state accuracy on orbit determination. Further work could assess how initial states derived from the correlation method influence an higher-fidelity force-model orbit determination. This may perhaps motivate the use of perturbed force models, at higher expense, to adequately handle lower altitude objects. Additionally, studying the impact of false positives on a higher fidelity force model orbit determination may help clarify the trade-off between the maximum accuracy true positive rate and accompanying increased false positive rate.

Third, further characterization of the BVP method is recommended through a sensitivity analysis isolating the impact of orbital elements on the estimation accuracy. This approach would enhance the understanding of BVP's behaviour under controlled conditions and aid observation strategy design, such as to avoid full or half revolutions and apply off-sets for consecutive night observations.

Practical enhancements for simulations and data handling should also be considered. The current simulated measurement populations are limited in size, primarily due the use of a rough initial implementation. Future work should focus on a more efficient approach to simulating measurements for larger population sizes. The larger and more diverse datasets could potentially provide more clarity on the obtained accuracy and cost-function distributions. The current implementation of the BVP method showed problems for the quasi-Newton optimization method when trying to handle about a large number of correlation cases (>30.000). Therefore, future work should also aim to resolve the memory leak occurring when using the BFGS optimization method. Additionally, the implementation can be significantly more efficient for large measurement correlations when programming for parallelized computations on GPUs to leverage more threads efficiently. Lastly, expanding the research to include real optical measurements is also crucial to validate the performance of the tracklet correlation approach in a real-world application.

By addressing these recommendations, future research can improve the robustness and applicability of the tracklet-correlation method explored in this work, particularly for lower-altitude regimes and large-scale optical tracking.

A

ART specifications

Table A.1: Airbus Robotic Telescope (ART) specifications [36].

Location	Extremadura, Spain	
(Lat., Long., Alt.)	(38.21607°, -6.62778°, 570 m)	
Tracking types	Surveillance, tracking	
Tasking methods	Scheduler[1]	
Data format	FITS, CCSDS TDM	
Accuracy (1-σ)	< 0.5 arcsec	
Parameter	**ASA H8**	**ASA H400**
Aperture diameter [mm]	200	400
Focal length [mm] (f-ratio)	590 (2.95)	960 (2.4)
Camera	FLI ML11002	Moravian C5-150M
Detector type	CCD, Interline	CMOS, Back-illuminated
Shutter	Global	Rolling
Detector size [px]	4008×2672	14208×10656
Pixel size [μm]	9.00	3.76
Pixel scale ["/px]	3.15	0.80
Field of view [deg]	3.50×2.34	3.18×2.39
Filters	Clear, UBVRI and empty	Empty and BVRI
Sensor peak QE [%]	50	≈ 90
Typical readout time full frame	<10 s	< 0.3 s

Figure A.1: ART’s two-telescope configuration in June 2023. Image obtained from [36].

B

Code

The following code can also be found on `github.com/casruks/IOD_T2T.git`.

B.1. Ln Gauss implementation

```
import numpy as np
from scipy.linalg import lstsq
from Constants_and_fuctions.constants import mu
import time

def Lagrange_ceff_universal(a, x_i, rn, tau_i, GM):
    ’’’
    From (Curtis, 2013), 3rd edition, algorithm D.15 Equation 3.69.
    ’’’
    phi = a*x_i**2
    c2, c3 = findc2c3(phi)

    fi = 1 - x_i**2/rn*c2
    gi = tau_i - 1/np.sqrt(GM)*x_i**3*c3
    return fi, gi

def Kepler_U(dt, rn, v_rad, a, GM):
    err = 1e-8
    k_max = 500

    x = np.sqrt(GM)*abs(a)*dt

    k = 0
    ratio = 1

    while (abs(ratio) > err) and k < k_max:
        k += 1
        c2, c3 = findc2c3(a*x**2)

        F = rn*v_rad/np.sqrt(GM)*x**2*c2 + (1 - a*rn)*x**3*c3 + rn*x - np.sqrt(GM)*dt
        dFdx = rn*v_rad/np.sqrt(GM)*x*(1 - a*x**2*c3) + (1 - a*rn)*x**2*c2 + rn

        ratio = F/dFdx
        x -= ratio

    if k > k_max:
        print(’Max iterations used for universal anomaly (k={k}).’)
    return x

def karimi_exact(tnp, pos_obs_nd, losnp, tol=1e-8, it_max=500):
    start_time = time.time()

    l = len(tnp)
    t_ = (tnp - tnp[0])
    t_diff = np.diff(t_)
```

```
    tau1,tau3 = np.array([-t_diff[:-1],t_diff[1:]])

    f1 = f3 = 1
    g1,g3 = tau1, tau3

    # non-zero initial guess
    # fg = f1*g3 - f3*g1
    # c_k, d_k = g3/fg, -g1/fg
    fg_in = f1, g1, f3, g3

    # M0, Xi0 = D_M(c_k, d_k, pos_obs_nd, losnp)
    # rho0,_resi,_rnk,_s = lstsq(M0,Xi0) #Gauss n angles

    # zero initial guess
    rho0 = np.zeros((l,1))

    # refine Gauss n angles
    rho, f1_i, f3_i, g1_i, g3_i = refine_ranges(rho0, pos_obs_nd, losnp, tau1, tau3,
                                                fg_in, tol, it_max)
    r_vec = np.array(pos_obs_nd).reshape((l,3))  + rho.reshape(l,1)*losnp

    # Determine mid-point velocity vectors
    router1_vec, rmid_vec, router3_vec = r_vec[:-2], r_vec[1:-1], r_vec[2:]
    f1, f3, g1, g3 = f1_i.reshape(l-2,1), f3_i.reshape(l-2,1), g1_i.reshape(l-2,1),
                                                g3_i.reshape(l-2,1)
    vmid2_vec = (-f3*router1_vec + f1*router3_vec)/(f1*g3 - f3*g1)

    comp_time = time.time() - start_time

    return r_vec, comp_time, vmid2_vec


def refine_ranges(rho, pos_obs_nd, los, tau1, tau3, fg_in, gtol, it_max):
    l = len(rho)
    f1, g1, f3, g3 = fg_in

    # Initial (un-refined) solution
    r_vec = pos_obs_nd + rho.reshape((l,1))*los

    # Determine initial mid-point velocity vectors. {r1, r2, r3} -> v2 etc
    router1_vec, rmid_vec, router3_vec = r_vec[:-2], r_vec[1:-1], r_vec[2:]

    # Set remaining intial values for refinement
    k = 0
    diff = 10
    rho_prev = rho
    r1_i, r2_i, r3_i = router1_vec, rmid_vec, router3_vec
    f1_i, g1_i, f3_i, g3_i = np.array([f1]*(l-2)), g1, np.array([f3]*(l-2)), g3

    while np.all(diff > gtol) and (k < it_max):
        k += 1

        # Reshape to allow matrix operation for variable tracklet length, l
        rq_shape = r1_i.shape
        g1_i = np.broadcast_to(g1_i[:, np.newaxis], rq_shape)
        f1_i = np.broadcast_to(f1_i[:, np.newaxis], rq_shape)
        g3_i = np.broadcast_to(g3_i[:, np.newaxis], rq_shape)
        f3_i = np.broadcast_to(f3_i[:, np.newaxis], rq_shape)

        # Update middle vectors with new lagrange coefficients
        vmid2_vec = (-f3_i*r1_i + f1_i*r3_i)/(f1_i*g3_i - f3_i*g1_i)
        rn = np.linalg.norm(r2_i, axis=1)
        v_rad = np.sum(vmid2_vec*r2_i, axis=1) / rn
        vn = np.linalg.norm(vmid2_vec, axis=1) # mid
        a = 2/rn - vn**2 / mu

        # Determine universal anomalies for each triplet
        x1 = [Kepler_U(tau1_, rn_, vrad_, a_, mu) for tau1_, rn_, vrad_, a_ in zip(tau1
                                                    , rn, v_rad, a)]
        x3 = [Kepler_U(tau3_, rn_, vrad_, a_, mu) for tau3_, rn_, vrad_, a_ in zip(tau3
                                                    , rn, v_rad, a)]
```

```
        # Determine Lagrange coefficients for each triplet
        fg1 = np.array([Lagrange_ceff_universal(a_, x1_, rn_, tau1_, mu) for a_, x1_,
                                                    rn_, tau1_ in zip(a, x1, rn, tau1)]
                                                    )
        fg3 = np.array([Lagrange_ceff_universal(a_, x3_, rn_, tau3_, mu) for a_, x3_,
                                                    rn_, tau3_ in zip(a, x3, rn, tau3)]
                                                    )
        # Update lagrange coefficients (same as in Gauss 3 angles)
        # Uses first elements of fg etc. out of simplicity, all elements are same to
                                                    allow for matrix mult.:
        # f1_i = [array([0.99903824, 0.99903824, 0.99903824])]
        f1_i = (f1_i[:,0] + fg1[:,0]) / 2
        g1_i = (g1_i[:,0] + fg1[:,1]) / 2
        f3_i = (f3_i[:,0] + fg3[:,0]) / 2
        g3_i = (g3_i[:,0] + fg3[:,1]) / 2

        fg = f1_i*g3_i - f3_i*g1_i
        c_k, d_k = g3_i/fg, -g1_i/fg

        # Set up design matrix and solve for updated ranges
        M, Xi = D_M(c_k, d_k, pos_obs_nd, los)
        rho, _resi, _rnk, _s = lstsq(M, Xi)

        # Quantify convergence
        diff = abs(rho - rho_prev)

        # Compute updated vectors, extract outer and middle
        r_vec = np.array(pos_obs_nd).reshape((1,3))  + rho.reshape(1,1)*los
        r1_i, r2_i, r3_i = r_vec[:-2], r_vec[1:-1], r_vec[2:]

        rho_prev = rho

    if k > it_max:
        print('Gauss (n>=3) refinement executed with max iterations (k={k}).')

    return rho, f1_i, f3_i, g1_i, g3_i

# Design matrix
def D_M(c1,c3,pos_obs,los):
    n = len(los)

    A_e,B_e = [],[]
    for i in range(n-2):
        A = np.zeros((3,n))
        los1,los2,los3 = los[i:i+3]
        A[0,i],A[1,i],A[2,i] = c1[i]*los1
        A[0,i+1],A[1,i+1],A[2,i+1] = -los2
        A[0,i+2],A[1,i+2],A[2,i+2] = c3[i]*los3

        B = pos_obs[i+1] - c1[i]*pos_obs[i] - c3[i]*pos_obs[i+2]

        A_e.append(A)
        B_e.append(B)

    A_e = np.vstack(A_e)
    B_e = np.hstack(B_e)

    return A_e,B_e
```

B.2. Parallel tracklet-to-tracklet correlation set-up

```
import ast
import time
import numpy as np
import pandas as pd
from tqdm import tqdm
from pathlib import Path
from datetime import datetime
```

```
from astropy.time import Time
from joblib import Parallel, delayed

from IOD_methods.BVP_AR import BVP, BVP_initial_val
from Constants_and_fuctions.constants import Re

method = 'L-BFGS-B' # or 'Powell'
regime_names = ['LEO', 'MEO', 'GEO']
regimes = [ 'LEO_measurement_file.txt',
            'MEO_measurement_file.txt',
            'GEO_measurement_file.txt']

bounds = [(Re, Re + 2000e3, 0.25),             # LEO
          (Re + 2000e3, Re + 31570e3, 0.25),   # MEO
          (Re + 35586e3, Re + 35986e3, 1e-2)]  # GEO

ident = 'run_id'
report_path = f'Data/T2T_results/report_{ident}.txt'

def T2T_processing(i, obj1, all_data):

        sig_theta1, sig_thetadot1 = obj1['SIG'], obj1['SIGD']
        Attr1 = obj1['ATTR']
        R1 = obj1['OBS']
        Rdot1 = obj1['OBS_V']
        fid_1 = obj1['NORAD ID']

        result_list = []

        for j in range(i + 1, len(all_data)):
            obj2 = all_data.iloc[j]
            try:
                epoch1, epoch2 = obj1['EPOCH (UTC)'], obj2['EPOCH (UTC)']
                tof = abs((epoch1 - epoch2).to_value('sec'))

                sig_theta2, sig_thetadot2 = obj2['SIG'], obj2['SIGD']

                # Construct covariance matrices
                cov_z = np.diag([sig_theta1, sig_theta1, sig_theta2, sig_theta2])
                cov_z_dot = np.diag([sig_thetadot1, sig_thetadot1, sig_thetadot2,
                                                           sig_thetadot2])

                # Combine attributes and observational data
                Attr = Attr1 + obj2['ATTR']
                R = np.hstack((R1, obj2['OBS']))
                Rdot = np.hstack((Rdot1, obj2['OBS_V']))

                # Boundary value problem (BVP) initialization
                p0, k_range, uvec = BVP_initial_val(Attr, tof, R, a_min, a_max)

                args = (k_range, Attr, R, Rdot, uvec, cov_z, cov_z_dot, tof, epoch1,
                                                           epoch2)
                result, comp_time, k, path, prograde = BVP(p0, args, method=method,
                                                           plot=(False,))

                fid_2 = obj2['NORAD ID']
                result_list.append((fid_1, fid_2, np.log10(result['fun']), tuple(result
                                                           ['x']), k, path, prograde,
                                                           tof/(24*3600), comp_time,
                                                           result['nit']))

            except Exception as e:
                fid_2 = obj2['NORAD ID']
                print(f"Error processing pair ({i}, {j}): {e}")
                result_list.append((fid_1, fid_2, np.nan, np.nan, np.nan, np.nan, np.
                                                           nan, np.nan, np.nan, np.nan
                                                           ))

        return result_list

for i, regime in enumerate(regimes):
```

```
    lst = []
    regime_time = time.time()
    start_stamp = datetime.now().strftime('%Y-%m-%d %H:%M:%S')
    all = pd.read_csv(regime, delimiter=',')
    all['EPOCH (UTC)'] = all['EPOCH (UTC)'].apply(lambda el: Time(el))
    all['OBS'] = all['OBS'].apply(lambda el: np.array(ast.literal_eval(el)))
    all['OBS_V'] = all['OBS_V'].apply(lambda el: np.array(ast.literal_eval(el)))
    all['ATTR'] = all['ATTR'].apply(lambda el: ast.literal_eval(el) if isinstance(el,
                                                str) else el)

    a_min, a_max, e_max = bounds[i]
    results = Parallel(n_jobs=-1)(delayed(T2T_processing)(i, obj1, all) for i, obj1 in
                                                tqdm(all.iterrows(), total=len(all['
                                                ATTR'])))

    # result output
    T2T_RES = [item for sublist in results for item in sublist]
    OUTPUT_T2T = pd.DataFrame(T2T_RES, columns=['OBJ_1', 'OBJ_2', 'L(P,K)', 'P', 'K', '
                                                PATH', 'PROG', 'TOF', 'CT', 'IT'])
    OUTPUT_T2T.to_csv(f"Data/T2T_results/{regime_names[i]}/T2TBVP_{method}_{ident}_{
                                                datetime.now().strftime('%H')}_{
                                                regime_names[i]}.txt", index=False)

    # report
    is_new_file = not Path(report_path).exists()

    with open(report_path, 'a') as file:
        if is_new_file:
            file.write(f"{'='*80}\n")
            file.write(f"{'METHOD:':<10} {method:<15} | {'START TIME:':<12} {
                                                        start_stamp}\n")
            file.write(f"{'='*80}\n\n")
            file.write(f"{'REGIME':<20}{'TOTAL TIME [S]':<20}{'END TIME':<30}\n")
            file.write(f"{'-'*80}\n")

        end_time = datetime.now().strftime('%Y-%m-%d %H:%M:%S')
        total_time = time.time() - regime_time
        file.write(f"{regime_names[i]:<20}{total_time:>20.5f}{end_time:>30}\n")

        if i == len(regimes) - 1:
            file.write(f"{'='*80}\n")
            file.write(f"{'END':<18} {method:<15} | {'COMPLETED ON:':<14} {datetime.now
                                                        ().strftime('%Y-%m-%d %H:%M:%S
                                                        ')}\n")
            file.write(f"{'='*80}\n\n")
```

B.3. BVP implementation and accompanying functions

```
import time
import gc
import numpy as np
from lamberthub import izzo2015, gooding1990
from scipy.optimize import minimize
from scipy.optimize._numdiff import approx_derivative
import matplotlib.pyplot as plt

from Constants_and_fuctions.constants import mu

ACCEPTABLE_ERRORS = {"No feasible solution, try lower M!", "No feasible solution, try
                                            lower M",
                     "Failed to converge", "Number of revolutions must be equal or
                                                                 greater than zero!"}

def unitvec(nrads):
    ra, dec = nrads
    rhoh = np.array([[np.cos(dec)*np.cos(ra)],
                     [np.cos(dec)*np.sin(ra)],
                     [np.sin(dec)]])
    return rhoh
```

```
def rmse(est, true):
    est = np.array(est)
    true = np.array(true)
    N = len(est)
    diff = est - true
    rss_sum = 0.
    for ii in range(N):
        rss = np.dot(diff[ii], diff[ii])
        rss_sum += rss
    rmse = np.sqrt(rss_sum/N)
    return rmse

def BVP(p0:list, args:tuple, method:str=’BFGS’, plot:tuple=(False,), Sol:tuple=None):
    start_time = time.time()
    (k_range, Attr, R, R_dot, uvec, cov_z, cov_z_dot, tof, epoch1, epoch2) = args

    result = {’fun’:  np.inf, ’x’: [0, 0], ’nit’: np.nan}
    k_res = np.nan
    res_path = []
    LOWP = None
    pr = None
    jac = None
    found_solution = False

    for prograde in [True, False]:
        for path in [True, False]:

            for k in k_range:
                evaluation_path_ = []
                def callback(xk):
                    if plot[0]:
                        evaluation_path_.append(np.copy(xk))

                if method.lower() in (’bfgs’, ’l-bfgs-b’):
                    options = {’ftol’:1e-4, ’gtol’:1e-7, ’disp’: False, ’c1’: 1e-5, ’c2
                                                                ’: 1e-4, ’maxiter’:
                                                                2000}

                    jac = ’2-point’

                    if method.lower() == ’l-bfgs-b’:
                        options = {’ftol’: 1e-4, ’maxiter’: 2000, ’gtol’:1e-7, ’disp’:
                                                                    False}

                else:
                    if method.lower() == ’nelder-mead’:
                        options = {’xatol’: 1e-4, ’fatol’: 1e-4, ’disp’: False}
                    else:
                        options = {’disp’: False}

                opt_args = (k, tof, Attr, R, R_dot, uvec, cov_z, cov_z_dot, prograde,
                                                            path)
                try:
                    result_ = minimize(Cost_fnc, p0, args=opt_args, method=method, jac=
                                                                jac, options=options,
                                                                callback=callback)

                    if (result_[’fun’] < result[’fun’]) and np.isfinite(result_[’fun’])
                                                                :
                        k_res = k
                        result = result_.copy()
                        LOWP = path
                        pr = prograde
                        found_solution = True
                        if plot[0]:
                            res_path = evaluation_path_

                        del result_
                        gc.collect()

                except Exception as e:
```

```
                    if str(e) in ACCEPTABLE_ERRORS:
                        continue
                    else:
                        print(f"Optimization error for k={k} (kmax:{k_range[-1]}),
                                                                            prograde={prograde}
                                                                            , path={path}: {e}"
                                                                            )
                        continue

    comp_time = time.time() - start_time

    if not found_solution:
        print(f"No solution ({k % k_range[-1]}) ..")

    if plot[0]:
        # Load figure style for report
        plt.rcParams['text.usetex'] = True
        style_path = "Style and Plots/paper.mplstyle"
        plt.style.use(style_path)

        a_min, a_max, e_max = plot[1:]
        args += (Sol,a_min,a_max,e_max, LOWP, comp_time, pr)
        plot_topography(p0, k_res, result, res_path, args)
        plot_jacobian(result, k_res, args, p0)
        plt.show()

    del k_range, Attr, R, R_dot, uvec, cov_z, cov_z_dot, tof, epoch1, epoch2
    gc.collect()

    return result, comp_time, k_res, LOWP, pr

def Cost_fnc(p:np.ndarray, k:float, tof:float, Attr:np.ndarray, R:np.ndarray, R_dot:np.
                                            ndarray, uvec:np.ndarray, cov_z:np.ndarray,
             cov_zdot:np.ndarray, prograde:bool, low_path:bool):

    # Extract attributable
    z = Attr[:2] + Attr[4:6]        #[a1, d1, a2, d2]
    zdot = Attr[2:4] + Attr[6:]     #[ad1, dd1, ad2, dd2]
    u1, u2 = uvec[:3], uvec[3:]

    los_vec = np.hstack((p[0] * u1, p[1] * u2))
    r = R + los_vec

    try: # Handle allowed errors lambert solver
        rdot_1, rdot_2 = izzo2015(mu, r[:3].flatten(), r[3:].flatten(), tof, M=k,
                                                    prograde=prograde, low_path=
                                                    low_path,
                                  maxiter=35, atol=1e-5, rtol=1e-7)
    except Exception as e:
        exc_str = str(e)
        if exc_str in ACCEPTABLE_ERRORS:  # Skip non-converging solution for k (or M)
                                                    orbital revolutions.
            return np.nan #return nan to stop optimizing, but must be a better way to
                                                        break optimizing process
        else:
            print('Lambert solver raised a ValueError:', exc_str)    # Print all other
                                                        value errors

    # Model attributable from orbit
    los_rate = np.hstack((rdot_1, rdot_2)) - R_dot
    zdot_hat = rates(los_vec, los_rate)

    # Transform covariance to modelled rates
    J = Jacobian(p, k, z, R, R_dot, tof, prograde, low_path)
    cov_zdot_hat = np.dot(J, np.dot(cov_z, J.T))

    # Evaluate terms of cost function, and return
    delta_zdot = zdot - zdot_hat                                                #[4 x 1]
    sum_cov = cov_zdot + cov_zdot_hat                                           #[4 x 4]
    L = np.dot(delta_zdot.T, np.dot(np.linalg.inv(sum_cov), delta_zdot))       #[1 x 1]
    return L
```

```
def Jacobian(p, k, z, R, R_dot, tof, prograde, low_path):
    '''
    Jacobian function to map the least-squares fit angle rates with C_zdot toward
                                            C_zdot_hat.

        *!  h:  absolute step-size of finite differencing, arbitrarily chosen based on
                                                    trial.
    '''

    def z_hat(z):
        u = np.hstack((unitvec(z[:2]).flatten(), unitvec(z[2:]).flatten()))
        los = np.hstack((p[0] * u[:3], p[1] * u[3:]))
        r = R + los

        rdot1, rdot2 = izzo2015(mu, r[:3], r[3:], tof, M=k, prograde=prograde, low_path
                                                    =low_path, maxiter=35, atol=1e-5,
                                                    rtol=1e-7)
        losrate = np.hstack((rdot1, rdot2)) - R_dot

        return rates(los, losrate)

    J = approx_derivative(z_hat, z, method='3-point')
    return J

def rates(xyz, xyz_dot):
        radec_dot = np.zeros(4)

        for i in range(2):
            lw = 3 * i
            up = 3 * i + 3
            x, y, z = xyz[lw:up]
            xdot, ydot, zdot = xyz_dot[lw:up]

            xy_sq = x**2 + y**2
            xyz_sq = xy_sq + z**2

            ra_dot = (x * ydot - y * xdot) / xy_sq
            dec_dot = (-z * (x * xdot + y * ydot) + zdot * xy_sq) / (np.sqrt(xy_sq) *
                                                        xyz_sq)
            radec_dot[i*2] = ra_dot
            radec_dot[i*2 + 1] = dec_dot

        return radec_dot #[alphadot1, deltadot1, alphadot2, deltadot2]

def BVP_initial_val(Attr, tof, R, a_min, a_max):

    k_range = k_int(a_min, a_max, tof)

    # Unit vectors for line of sights
    u1 = unitvec(Attr[:2]).flatten()
    u2 = unitvec(Attr[4:6]).flatten()
    uvec = np.hstack((u1, u2))

    # Calculate R_dot_u1, R_dot_u2, Rsq1, and Rsq2
    R1, R2 = R[:3], R[3:]
    R_dot_u1 = np.dot(R1, u1)
    Rsq1 = np.dot(R1, R1)
    R_dot_u2 = np.dot(R2, u2)
    Rsq2 = np.dot(R2, R2)

    # Assume circular orbit and determine initial guess
    k0 = k_range[int(len(k_range)/4)]

    if k0 != 0:
        a0 = ((tof**2 * mu) / (4*np.pi**2 * k0**2))**(1/3)
    else:
        a0 = a_min

    with np.errstate(invalid='ignore'):
        rho01 = - R_dot_u1 + np.sqrt(R_dot_u1**2 + a0**2 - Rsq1)
```

```
        rho02 = - R_dot_u2 + np.sqrt(R_dot_u2**2 + a0**2 - Rsq2)

    return [rho01, rho02], k_range, uvec

def plot_jacobian(result, k_res, args, p0_int, num_points=100):
    (k_range, Attr, R, R_dot, uvec, cov_z, cov_z_dot, tof, epoch1, epoch2, Sol, a_min,
     a_max, e_max, low_path, comp_time, prograde) = args

    # Extract true solution values
    x1_true, x2_true = Sol
    r1_true = x1_true[:3]
    r2_true = x2_true[:3]

    rho1_true = np.linalg.norm(r1_true - R[:3])
    rho2_true = np.linalg.norm(r2_true - R[3:])
    p0 = [rho1_true, rho2_true]
    p0_true = p0

    fig, axs = plt.subplots(len(p0), 1 , figsize=(6,5), sharex=True)
    labels = [r'($\rho_1$, $\rho_{2 sol}$) [m]', r'($\rho_{1 sol}$, $\rho_2$) [m]']
    for res in range(len(p0)):
        variable_range = np.linspace(p0[res]*0.1, p0[res]*2, num_points)
        jacobian_values = np.zeros(num_points)
        jacobian_values2p = np.zeros(num_points)
        Lvals = np.zeros(num_points)
        original_value = p0[res]
        for i, value in enumerate(variable_range):
            p0[res] = value
            jac = approx_derivative(Cost_fnc, p0, method='2-point', args=(k_res, tof,
                                                       Attr, R, R_dot, uvec, cov_z,
                                                                         cov_z_dot,









            jac_2p = approx_derivative(Cost_fnc, p0, method='3-point', args=(k_res, tof
                                                       , Attr, R, R_dot, uvec, cov_z,
                                                                            cov_z_dot,










            jacobian_values[i] = jac[res]
            jacobian_values2p[i] = jac_2p[res]
            Lvals[i] = Cost_fnc(p0, k_res, tof, Attr, R, R_dot, uvec, cov_z, cov_z_dot,
                                                        prograde, low_path)
        p0[res] = original_value
        ax_left = axs[res]
        line1, = ax_left.plot(variable_range, (jacobian_values), color='r', label='fwd.
                                                   ')
        line2, = ax_left.plot(variable_range, (jacobian_values2p), color='b', linestyle
                                                   ='dashed', label='cent.')
        ax_left.set_ylabel(r'$\frac{\partial}{\partial \rho} \  L(\mathbf{p},k)$')
        ax_left.tick_params(axis='y')

        ax_right = ax_left.twinx()
        line3, = ax_right.plot(variable_range, Lvals, color='k', label=r'$L(\mathbf{p},
                                                   k)$')
        ax_right.set_ylabel(r'$\log_{10} \ L(\mathbf{p}_s,k)$')
        ax_right.tick_params(axis='y')
        ax_right.set_yscale('log')

        ax_left.set_xlabel(labels[res])
        ax_left.set_ylim(-1, 1)
```

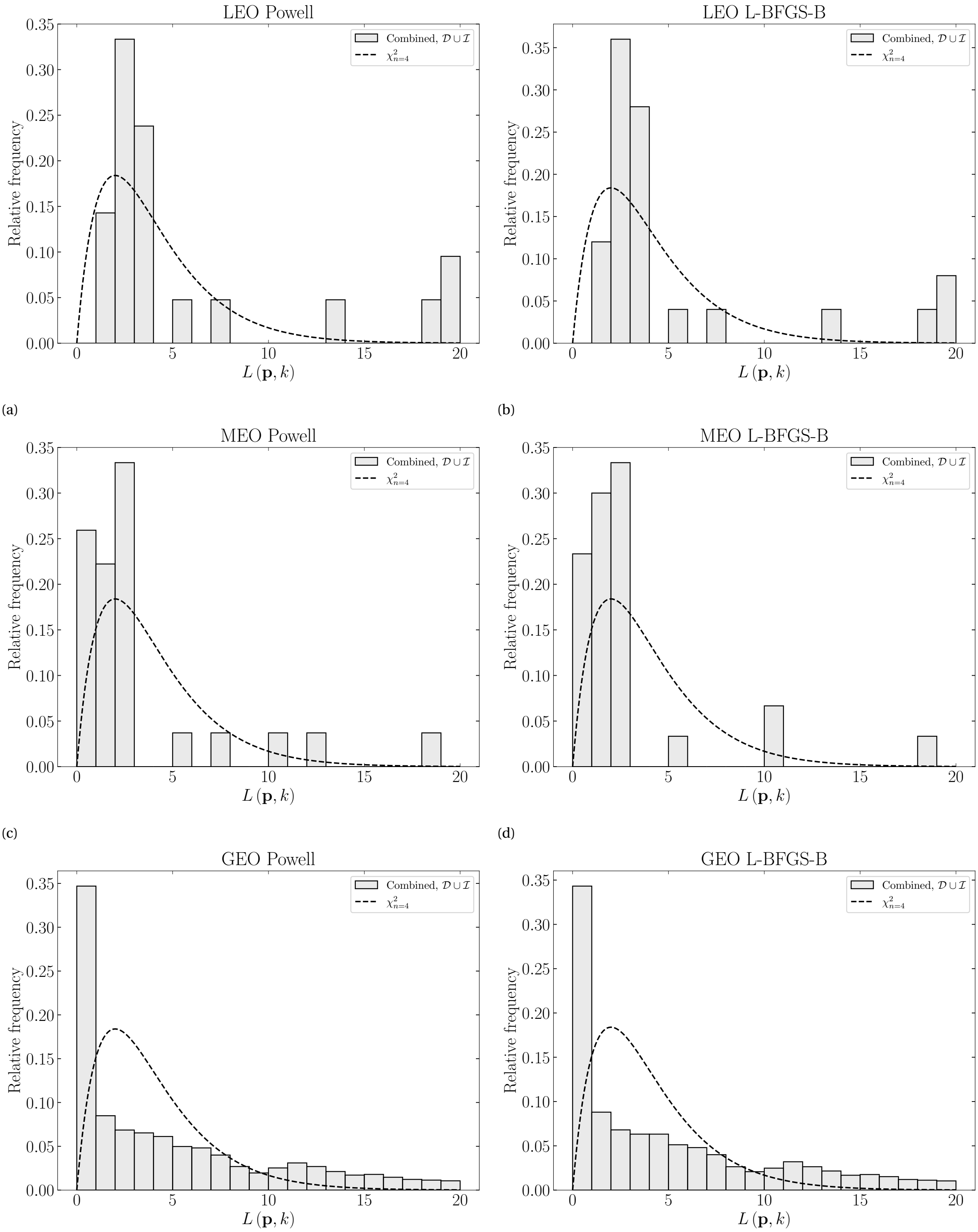
```
        vline1 = ax_left.axvline(x=p0_true[res], color='k', linestyle='dashed', label=r
                                                    'Solution $\mathbf{p}_s$')
        vline2 = ax_left.axvline(x=p0_int[res], color='k', linestyle='dotted', label=r'
                                                    $\mathbf{p}_0$')
        lines_left, labels_left = ax_left.get_legend_handles_labels()
        lines_right, labels_right = ax_right.get_legend_handles_labels()

    fig.legend(lines_left +lines_right , labels_left +labels_right +['Solution', '
                                                Initial guess'], ncol=2, loc='upper
                                                left',
               bbox_to_anchor=(0.12, 0.89))
    axs[0].set_title(r'Jacobian and Cost Function along $\rho_1$ and $\rho_2$')
    plt.subplots_adjust(hspace=0.35)
    from datetime import datetime
    timestamp = datetime.now().strftime("%m%d_%H%M%S")
    fig.savefig(f"BVP_run_{timestamp}_jacobian.pdf", bbox_inches="tight")

def plot_topography(p0, k_res, result, eval_points, args, save=False):
    import pandas as pd
    (k_range, Attr, R, R_dot, uvec, cov_z, cov_z_dot, tof, epoch1, epoch2, Sol, a_min,
                                                a_max, e_max, low_path, comp_time,
     prograde) = args

    if len(eval_points) != 0:
        rho1 = [rho[0] for rho in eval_points]
        rho2 = [rho[1] for rho in eval_points]
    else:
        rho1 = [p0[0]]
        rho2 = [p0[1]]

    R1, R2 = R[:3], R[3:]
    u1, u2 = uvec[:3], uvec[3:]

    R_dot_u1 = np.dot(R1, u1)
    Rsq1 = np.dot(R1, R1)
    R_dot_u2 = np.dot(R2, u2)
    Rsq2 = np.dot(R2, R2)

    with np.errstate(invalid='ignore'):
        rho_min1 = - R_dot_u1 + np.sqrt(R_dot_u1**2 + a_min**2 * (1 - e_max)**2 - Rsq1)
        rho_max1 = - R_dot_u1 + np.sqrt(R_dot_u1**2 + a_max**2 * (1 + e_max)**2 - Rsq1)
        rho_min2 = - R_dot_u2 + np.sqrt(R_dot_u2**2 + a_min**2 * (1 - e_max)**2 - Rsq2)
        rho_max2 = - R_dot_u2 + np.sqrt(R_dot_u2**2 + a_max**2 * (1 + e_max)**2 - Rsq2)

    if Sol is None:
        A1, B1 = rho_min1*0.9, rho_max1*1.05
        A2, B2 = rho_min2*0.9, rho_max2*1.05
    else:
        x1_true, x2_true = Sol
        r1_true = x1_true[:3]
        r2_true = x2_true[:3]

        rho1_true = np.linalg.norm(r1_true - R1)
        rho2_true = np.linalg.norm(r2_true - R2)

        margin_factor = 0.5
        A1 = rho1_true * (1 - margin_factor)
        B1 = rho1_true * (1 + margin_factor)
        A2 = rho2_true * (1 - margin_factor)
        B2 = rho2_true * (1 + margin_factor)

    x_range = B1 - A1
    y_range = B2 - A2
    max_range = max(x_range, y_range)
    x_mid = (A1 + B1) / 2
    y_mid = (A2 + B2) / 2
    A1 = x_mid - max_range / 2
    B1 = x_mid + max_range / 2
    A2 = y_mid - max_range / 2
    B2 = y_mid + max_range / 2
```

```
X = np.linspace(A1, B1, 100)
Y = np.linspace(A2, B2, 100)

Z = np.zeros((len(X), len(Y)))
for i in range(len(X)):
    for j in range(len(Y)):
        Z[i, j] = (np.log10(Cost_fnc([X[i], Y[j]], k_res, tof, Attr, R, R_dot, uvec
                                                    , cov_z, cov_z_dot, prograde,
                                                    low_path)))

fig, ax = plt.subplots()
contour_filled = ax.contourf(X, Y, Z, levels=100, cmap="gray")

cbar = fig.colorbar(contour_filled)
cbar.ax.set_ylabel(r"$\log_{10}L(\mathbf{z}, k)$", rotation=-90, va="bottom")
ax.plot(([p0[0]] + rho1), ([p0[1]] + rho2), ’r’, linestyle=’--’, marker=’o’,
                                            markerfacecolor=’none’, markersize=4,
        label=’Evaluation Path’, zorder=90)
ax.scatter(p0[0], p0[1], color=’r’, marker=’x’, s=16, label=’Start Point’, zorder=
                                            91)

if Sol != None:
    r1_true = x1_true[:3]
    r2_true = x2_true[:3]
    rdot1_true = x1_true[3:]
    rdot2_true = x2_true[3:]
    rho1_true = np.linalg.norm(r1_true - R1)
    rho2_true = np.linalg.norm(r2_true - R2)

    ax.scatter(rho1_true, rho2_true, s=16, color=’gold’, marker=(6, 2, 0), label=’
                                                Solution’, zorder=100)
    los_vec = np.hstack((rho1[-1] * u1, rho2[-1] * u2))
    r = R + los_vec
    rdot_1, rdot_2 = izzo2015(mu, r[:3].flatten(), r[3:].flatten(), tof, M=k_res,
                                                prograde=prograde, low_path=
                                                low_path,
                              maxiter=35, atol=1e-5, rtol=1e-7)

    output = {
        "Metric": ["Epoch", "Obtained RHO [km]", "TRUE RHO [km]", "RMSE P [km]", "
                                                    RMSE V [m/s]", "REL_ERR RHO [%]
                                                    ",
                   "Time taken [s]:", "K (REV):", "PROGRADE:", "LOW PATH:", "LOG10
                                                               (obtained)", "LOG10
                                                               (true)"],
        "EPOCH 1": [epoch1.iso, f"{result[’x’][0]*1e-3:.4f}", f"{rho1_true*1e-3:.4f
                                                    }", f"{rmse(r[:3], r1_true)*1e-
                                                    3:.4f}",
                    f"{rmse(rdot_1, rdot1_true):.4f}", f"{100*(rho1[-1] - rho1_true
                                                                )/rho1_true:.4f}",
                                                                f"{comp_time:.4f}",
                    f"{k_res:.4f}", f"{str(prograde)}", f"{str(low_path)}", f"{np.
                                                                log10(result[’fun
                                                                ’]):.4f}",
                    f"{np.log10(Cost_fnc([rho1_true, rho2_true], k_res, tof, Attr,
                                                                R, R_dot, uvec,
                                                                cov_z, cov_z_dot,
                                                                prograde, low_path)
                                                                ):.4f}"],
        "EPOCH 2": [epoch2.iso, f"{result[’x’][1]*1e-3:.4f}", f"{rho2_true*1e-3:.4f
                                                    }", f"{rmse(r[3:], r2_true)*1e-
                                                    3:.4f}",
                    f"{rmse(rdot_2, rdot2_true):.4f}", f"{100*(rho2[-1] - rho2_true
                                                                )/rho2_true:.4f}",
                                                                "", "", "", "", "",
                                                                 ""]
    }

    output_df = pd.DataFrame(output)
    print(output_df)
```

```
        output_df.to_clipboard(index=False, sep='\t')

    ax.scatter(rho1[-1], rho2[-1], color='r', marker='d', facecolors='none', s=25,
                                                label='Obtained solution', zorder=99)
    plt.xlabel(r'$\rho_1$ [m]')
    plt.ylabel(r'$\rho_2$ [m]')
    A1, B1, A2, B2 = [None if np.isnan(x) or np.isinf(x) else x for x in [A1, B1, A2,
                                                B2] ]
    plt.xlim(A1, B1)
    plt.ylim(A2, B2)

    plt.legend(loc='upper left')
    if Sol != None:
        from datetime import datetime
        timestamp = datetime.now().strftime("%m%d_%H%M%S")
        for c in contour_filled.collections: #to prevent white discontinuous rendered
                                                    lines in pdf
            c.set_rasterized(True)
        fig.savefig(f"BVP_run_{timestamp}.pdf", bbox_inches="tight")
    plt.show()

def k_int(a_min, a_max, tof):
    '''
    Determine interval of orbital half-revolutions.
    '''
    twopi = 2*np.pi
    P_max = twopi * np.sqrt(a_max**3 / mu)
    P_min = twopi * np.sqrt(a_min**3 / mu)

    k_min, k_max = (tof / P_max), (tof / P_min)
    step = 0.5
    k_range = np.arange((np.floor(k_min*2)/2), (np.floor(k_max*2)/2) + step, step)

    if len(k_range) == 0:
        k_range = np.array([0])

    return k_range
```

C

Complete BVP results



(a)

(b)

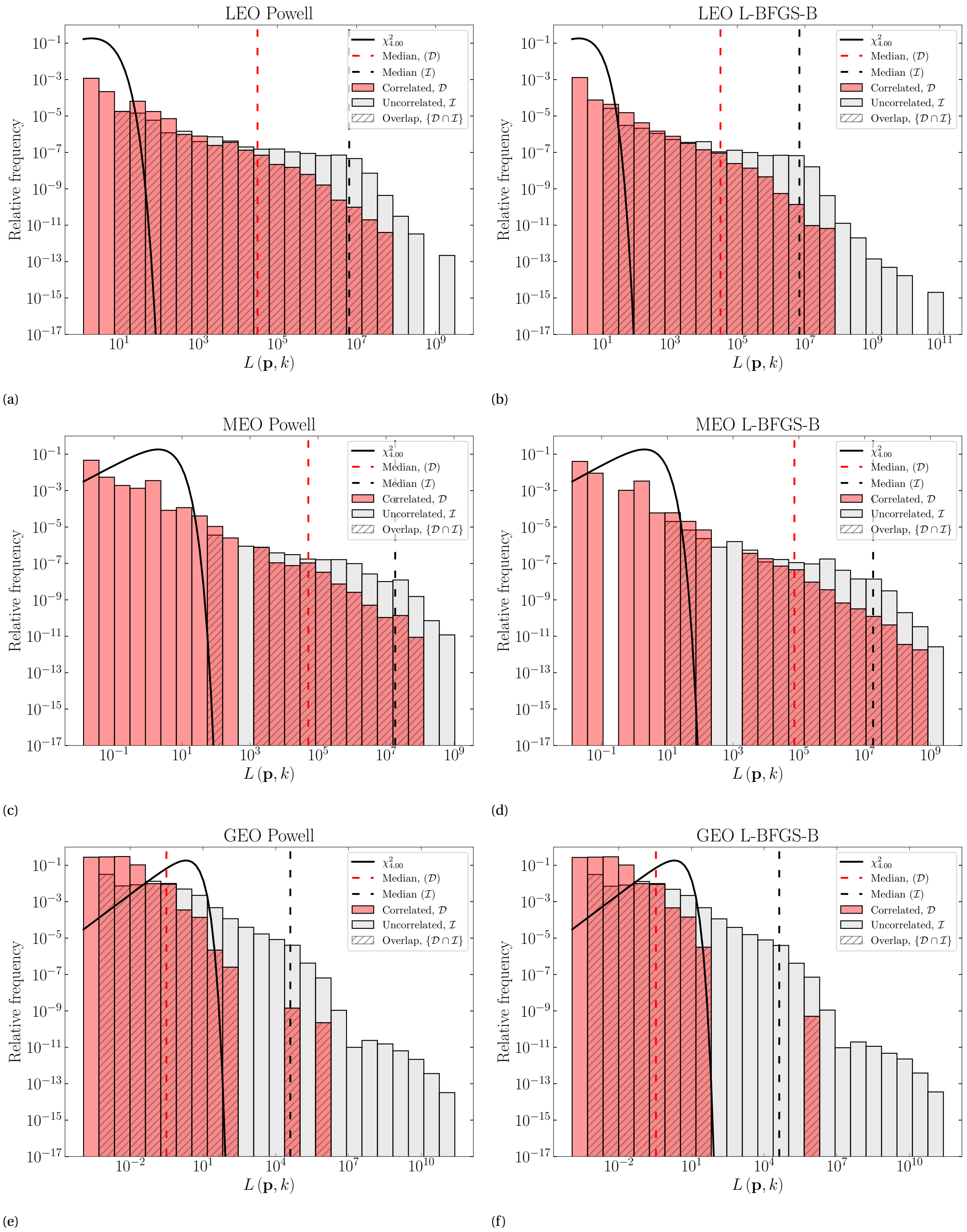
(c)

(d)

(e)

(f)

Figure C.1: The relative frequency of the obtained minimum cost function values obtained in comparison to the theoretical chi-squared distribution for each optimization method and orbital regime.



(a)

(b)

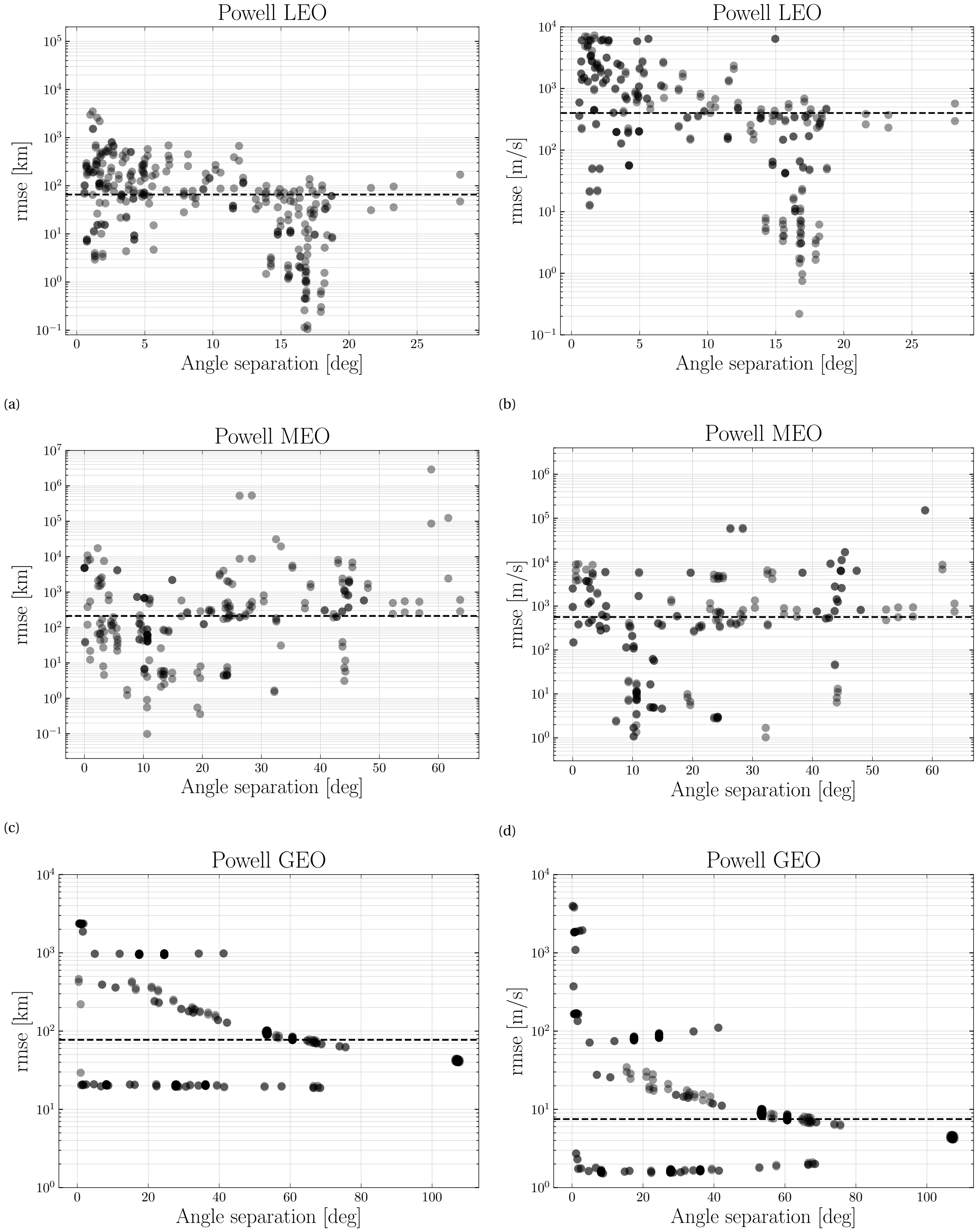
(c)

(d)

(e)

(f)

Figure C.2: The relative frequency of the obtained minimum cost function values obtained in comparison to the theoretical chi-squared distribution for each optimization method and orbital regime in log scale.



(a) (b) (c) (d) (e) (f)

Figure C.3: The root mean square error against angle separation for each true correlation. The dotted line shows the median value of the whole correlated population.

Bibliography

[1] Luigi Ansalone and Fabio Curti. “A genetic algorithm for Initial Orbit Determination from a too short arc optical observation”. In: Advances in Space Research 52.3 (Aug. 2013), pp. 477–489. ISSN: 0273-1177. DOI: `10.1016/j.asr.2013.04.004`. (Visited on 04/10/2024).

[2] Yaakov Bar-Shalom, X.-Rong Li, and Thiagalingam Kirubarajan. Estimation with Applications to Tracking and Navigation. John Wiley & Sons, Ltd, 2001, pp. 57–58. ISBN: 9780471221272.

[3] Richard H. Battin. An Introduction to the Mathematics and Methods of Astrodynamics, Revised Edition. en. Reston, VA: American Institute of Aeronautics and Astronautics, Inc., Jan. 1999, p. 276. ISBN: 978-1-60086-154-3. DOI: `10.2514/4.861543`. URL: `https://arc.aiaa.org/doi/book/10.2514/4.861543` (visited on 11/19/2024).

[4] Paul Bourke. Intersection of a Line and a Sphere (or circle). 1992. URL: `https://paulbourke.net/geometry/circlesphere/index.html#linesphere`.

[5] C. G. BROYDEN. “The Convergence of a Class of Double-rank Minimization Algorithms 1. General Considerations”. In: IMA Journal of Applied Mathematics 6.1 (Mar. 1970), pp. 76–90. ISSN: 0272-4960.

[6] Richard H. Byrd et al. “A Limited Memory Algorithm for Bound Constrained Optimization”. In: SIAM Journal on Scientific Computing 16.5 (1995), pp. 1190–1208.

[7] Han Cai, Islam Hussein, and Moriba Jah. “Possibilistic admissible region using outer probability measure theory”. en. In: Acta Astronautica 177 (Dec. 2020), pp. 246–257. ISSN: 00945765. DOI: `10.1016/j.actaastro.2020.07.041`. (Visited on 05/02/2024).

[8] Han Cai et al. “Improved tracklet association for space objects using short-arc optical measurements”. en. In: Acta Astronautica 151 (Oct. 2018), pp. 836–847. ISSN: 00945765. DOI: `10.1016/j.actaastro.2018.07.024`. (Visited on 05/28/2024).

[9] Frederick R Chromey. “To Measure the Sky: An Introduction to Observational Astronomy.” en. In: ().

[10] Giovanni Cirillo, Martin Michel, and Jens Utzmann. “Airbus Robotic Telescope and SPOOK as test-bed for space-based space surveillance”. de. In: 2023. (Visited on 04/10/2024).

[11] Howard D. Curtis. “Chapter 2 - The Two-Body Problem”. In: Orbital Mechanics for Engineering Students (Third Edition). Ed. by Howard D. Curtis. Third Edition. Boston: Butterworth-Heinemann, 2014, pp. 59–144. ISBN: 978-0-08-097747-8. DOI: `https://doi.org/10.1016/B978-0-08-097747-8.00002-5`.

[12] Howard D. Curtis. “Chapter 5 - Preliminary Orbit Determination”. In: Orbital Mechanics for Engineering Students (Third Edition). Ed. by Howard D. Curtis. Third Edition. Boston: Butterworth-Heinemann, 2014, pp. 239–298. ISBN: 978-0-08-097747-8. DOI: `https://doi.org/10.1016/B978-0-08-097747-8.00005-0`.

[13] Kyle J. DeMars and Moriba K. Jah. “Probabilistic Initial Orbit Determination Using Gaussian Mixture Models”. en. In: Journal of Guidance, Control, and Dynamics 36.5 (Sept. 2013), pp. 1324–1335. ISSN: 0731-5090, 1533-3884. DOI: `10.2514/1.59844`. (Visited on 05/02/2024).

[14] Pedro Ramon Escobal. Methods of Orbit Determination. Melbourne, FL: Krieger Publishing Company, June 1976.

[15] European Space Agency. ESA Space Environment Report 2024. European Space Agency, 2024.

[16] Fran M. Fadrique et al. “Comparison of angles only initial orbit determination algorithms for space debris cataloguing”. en. In: Journal of Aerospace Engineering, Sciences and Applications 4.1 (Jan. 2012), pp. 39–51. ISSN: 2236577X. DOI: `10.7446/jaesa.0401.04`. (Visited on 05/28/2024).

[17] R. Fletcher. “A new approach to variable metric algorithms”. In: The Computer Journal 13.3 (Jan. 1970), pp. 317–322. ISSN: 0010-4620.

[18] Jorge Martínez Garrido. Lambert’s problem algorithms: a critical review. Universidad Carlos III de Madrid, 2021.

[19] Donald Goldfarb. “A family of variable-metric methods derived by variational means”. In: Mathematics of Computation 24 (1970), pp. 23–26.

[20] R. H. Gooding. “A new procedure for orbit determination based on three lines of sight: Angles only”. In: NASA STI/Recon Technical Report N 94 (Apr. 1993). ADS Bibcode: 1993STIN...9421224G, p. 21224. (Visited on 05/15/2024).

[21] R. H. Gooding. “A procedure for the solution of Lambert’s orbital boundary-value problem”. In: Celestial Mechanics and Dynamical Astronomy 48.2 (June 1990), pp. 145–165. ISSN: 1572-9478. DOI: `10.1007/BF00049511`.

[22] Giovanni F. Gronchi. “Classical and modern orbit determination for asteroids”. In: Proceedings of the International Astronomical Union 2004.IAUC196 (2004), pp. 293–303. DOI: `10.1017/S174392130500147X`.

[23] CS GROUP and Orekit Contributors. Orekit Estimation IOD Package. `https://github.com/CS-SI/Orekit/tree/develop/src/main/java/org/orekit/estimation/iod`. 2024.

[24] S. Herrick. ASTRODYNAMICS. Vol. 1. English. New York: Van Nostrand Reinhold, 1971.

[25] Felix R. Hoots, Paul W. Schumacher, and Robert A. Glover. “History of Analytical Orbit Modeling in the U. S. Space Surveillance System”. In: Journal of Guidance, Control, and Dynamics 27.2 (2004), pp. 174–185. DOI: `10.2514/1.9161`.

[26] Jian Huang et al. “Short-Arc Association and Orbit Determination for New GEO Objects with Space-Based Optical Surveillance”. en. In: Aerospace 8.10 (Oct. 2021), p. 298. ISSN: 2226-4310. DOI: `10.3390/aerospace8100298`. (Visited on 05/28/2024).

[27] Islam I. Hussein et al. “Probabilistic Admissible Region for Multihypothesis Filter Initialization”. en. In: Journal of Guidance, Control, and Dynamics 41.3 (Mar. 2018), pp. 710–724. ISSN: 0731-5090, 1533-3884. DOI: `10.2514/1.G002788`. (Visited on 05/02/2024).

[28] Hyewon Hwang, Sang-Young Park, and Eunji Lee. “Angles-Only Initial Orbit Determination of Low Earth Orbit (LEO) Satellites Using Real Observational Data”. en. In: Journal of Astronomy and Space Sciences 36.3 (Sept. 2019), pp. 187–197. ISSN: 2093-5587, 2093-1409. DOI: `10.5140/JASS.2019.36.3.187`. (Visited on 05/28/2024).

[29] Dario Izzo. esa/pykep. Oct. 2020. DOI: `10.5281/zenodo.596739`.

[30] Dario Izzo. “Revisiting Lambert’s problem”. en. In: Celestial Mechanics and Dynamical Astronomy 121.1 (Jan. 2015), pp. 1–15. ISSN: 0923-2958, 1572-9478. DOI: `10.1007/s10569-014-9587-y`. (Visited on 05/23/2024).

[31] Joblib Development Team. Joblib: running Python functions as pipeline jobs. 2020. URL: `https://joblib.readthedocs.io/`.

[32] James P Johnson. “Limitations of Initial Orbit Determination Methods for Low Earth Orbit CubeSats with Short Arc Orbital Passes”. en. PhD thesis. San Luis Obispo, California: California Polytechnic State University, July 2020. DOI: `10.15368/theses.2020.122`. (Visited on 05/28/2024).

[33] Nicholas Johnson. “The Collision of Iridium 33 and Cosmos 2251: The Shape of Things to Come”. In: 60th International Astronautical Congress. Seoul, Korea, Republic of: NASA Johnson Space Center, 2009.

[34] Reza Raymond Karimi and Daniele Mortari. “Initial orbit determination using multiple observations”. en. In: Celestial Mechanics and Dynamical Astronomy 109.2 (Feb. 2011), pp. 167–180. ISSN: 0923-2958, 1572-9478. DOI: `10.1007/s10569-010-9321-3`. (Visited on 05/21/2024).

[35] T.S. Kelso. Iridium 33/Cosmos 2251 Collision. June 2012. URL: `https://celestrak.org/events/collision/`.

[36] Moritz Kuhn. Airbus Robotic Telescope (ART) Overview. en. Tech. rep. 0.5. Feb. 2024.

[37] E R Lancaster and R C Blanchard. “A unified form of lambert’s theorem”. en. In: (1969).

[38] Prasanta Chandra Mahalanobis. “On test and measures of group divergence: theoretical formulae”. In: (1930).

[39] B. G. Marsden. “Initial orbit determination - The pragmatist’s point of view”. In: The Astronomical Journal 90 (Aug. 1985), p. 1541. ISSN: 00046256. DOI: `10.1086/113867`.

[40] J. M. Maruskin, D. J. Scheeres, and K. T. Alfriend. “Correlation of Optical Observations of Objects in Earth Orbit”. en. In: Journal of Guidance, Control, and Dynamics 32.1 (Jan. 2009), pp. 194–209. ISSN: 0731-5090, 1533-3884. DOI: `10.2514/1.36398`. (Visited on 08/12/2024).

[41] Andrea Milani et al. “Orbit determination with very short arcs. I admissible regions”. en. In: Celestial Mechanics and Dynamical Astronomy 90.1 (Sept. 2004), pp. 57–85. ISSN: 1572-9478. DOI: `10.1007/s10569-004-6593-5`. (Visited on 05/20/2024).

[42] C.A. Miller and C. Frueh. “A COMPREHENSIVE COMPARISON AMONG CLASSICAL AND GOODING METHODS OF INITIAL ORBIT DETERMINATION IN OPTIMIZED ELECTRO- OPTICAL SENSOR NETWORKS”. en. In: vol. 8. Darmstadt, Germany: ESA Space Debris Office, 2021.

[43] Oliver Montenbruck and Eberhard Gill. Satellite Orbits. en. Berlin, Heidelberg: Springer Berlin Heidelberg, 2000. ISBN: 978-3-540-67280-7. DOI: `10.1007/978-3-642-58351-3`. (Visited on 04/22/2024).

[44] n.a. Mahalanobis distance. URL: `https://encyclopediaofmath.org/wiki/Mahalanobis_distance` (visited on 09/12/2024).

[45] NIST. 1.3.6.7.4. Critical Values of the Chi-Square Distribution. URL: `https://www.itl.nist.gov/div898/handbook/eda/section3/eda3674.htm` (visited on 09/11/2024).

[46] Jorge Nocedal and Stephen J. Wright. Numerical optimization. en. 2nd ed. Springer series in operations research. New York: Springer, 2006, p. 136. ISBN: 978-0-387-30303-1.

[47] Orekit Labs. Python Wrapper for Orekit. `https://gitlab.orekit.org/orekit-labs/python-wrapper`. 2024.

[48] Panos Y. Papalambros and Douglass J. Wilde. Principles of Optimal Design: Modeling and Computation. 3rd ed. Cambridge University Press, 2017, p. 297.

[49] Guido Pedone et al. “SPOOK: A tool for space objects catalogue creation and maintenance supporting space safety and sustainability”. In: Acta Astronautica 188 (Nov. 2021), pp. 89–98. ISSN: 0094-5765. DOI: `10.1016/j.actaastro.2021.07.026`. (Visited on 04/10/2024).

[50] John E. Prussing and Bruce A. Conway. Orbital mechanics. en. New York: Oxford University Press, 1993. ISBN: 978-0-19-507834-3.

[51] Danielle Racelis and Mathieu Joerger. “High-Integrity TLE Error Models for MEO and GEO Satellites”. en. In: 2018 AIAA SPACE and Astronautics Forum and Exposition. Orlando, FL: American Institute of Aeronautics and Astronautics, Sept. 2018. ISBN: 978-1-62410-575-3. DOI: `10.2514/6.2018-5241`. URL: `https://arc.aiaa.org/doi/10.2514/6.2018-5241` (visited on 10/19/2024).

[52] Brandon Rhodes and David A. Vallado. SGP4 Python Package. `https://pypi.org/project/sgp4/`. Version 2.23, Accessed: 2024-10-19. 2024.

[53] Juan Rodríguez and Jorge Garrido. “poliastro: a Python library for interactive astrodynamics”. In: poliastro. Ed. by Meghann Agarwal et al. Austin, Texas, 2022, pp. 136–146. DOI: `10.25080/majora-212e5952-015`.

[54] Oscar Rodriguez Fernandez, Jens Utzmann, and Urs Hugentobler. “SPOOK - A comprehensive Space Surveillance and Tracking analysis tool”. In: Acta Astronautica 158 (May 2019), pp. 178–184. ISSN: 0094-5765. DOI: `10.1016/j.actaastro.2018.02.008`. (Visited on 04/10/2024).

[55] Andrew Vernon Schaeperkoetter. “A COMPREHENSIVE COMPARISON BETWEEN ANGLES-ONLY INITIAL ORBIT DETERMINATION TECHNIQUES”. MA thesis. Texas A&M University, Dec. 2011.

[56] P Schumacher, M Wilkins, and C Roscoe. “Parallel algorithm for track initiation for optical space surveillance”. en. In: Darmstadt, Germany, 2013.

[57] SciPy Community. scipy.optimize.minimize: Minimization of scalar function of one or more variables. 2023. URL: `https://docs.scipy.org/doc/scipy/reference/generated/scipy.optimize.minimize.html%5C#scipy.optimize.minimize`.

[58] David F. Shanno. “Conditioning of Quasi-Newton Methods for Function Minimization”. In: Mathematics of Computation 24 (1970), pp. 647–656.

[59] J.A. Siminski et al. “Short-arc tracklet association for geostationary objects”. en. In: Advances in Space Research 53.8 (Apr. 2014), pp. 1184–1194. ISSN: 02731177. DOI: `10.1016/j.asr.2014.01.017`. (Visited on 08/06/2024).

[60] Jan Siminski. “Object correlation and orbit determination for geostationary satellites using optical measurements”. PhD thesis. Universiät der Bundeswehr München, Fakultät für Luft- und Raumfahrttechnik, 2016.

[61] Jan A Siminski et al. “Best hypotheses search on iso-energy-grid for initial orbit determination and track association”. In: Advances in the Astronautical Sciences 148 (2013), pp. 605–617.

[62] D. Sivia and J. Skilling. Data Analysis: A Bayesian Tutorial. Oxford science publications. OUP Oxford, 2006, pp. 72–73. ISBN: 9780198568315.

[63] Byron D. Tapley, Bob E. Schutz, and George H. Born. Statistical Orbit Determination. English. Amsterdam: Academic Press, 2004. ISBN: 978-0-12-683630-1. (Visited on 04/10/2024).

[64] G. Tommei, A. Milani, and A. Rossi. “Orbit determination of space debris: admissible regions”. en. In: Celestial Mechanics and Dynamical Astronomy 97.4 (Apr. 2007), pp. 289–304. ISSN: 1572-9478. DOI: `10.1007/s10569-007-9065-x`. (Visited on 05/20/2024).

[65] David A Vallado. “Evaluating Gooding Angles-only Orbit Determination of Space Based Space Surveillance Measurements”. en. In: (2010).

[66] David A. Vallado. Fundamentals of Astrodynamics and Applications, 4th ed. (Space Technology Library). English. Ed. by James Wertz. Paperback. Microcosm Press, Mar. 2013. ISBN: 978-1-881883-18-0.

[67] David A. Vallado et al. “Revisiting Spacetrack Report #3”. In: AIAA/AAS Astrodynamics Specialist Conference. AIAA 2006-6753. Keystone, CO: American Institute of Aeronautics and Astronautics, Aug. 2006. URL: `https://celestrak.org/publications/AIAA/2006-6753/`.

[68] Bert Van den Abbeele. “Initial Orbit Determination Using Angle Measurements”. en. MA thesis. Delft: Delft University of Technology, 2021.

[69] Karel F Wakker. Fundamentals of Astrodynamics. en. Delft: Library Delft University of Technology, 2015. ISBN: 978-94-6186-419-2.

[70] Sam Wishnek et al. “Robust Initial Orbit Determination Using Streaks and Admissible Regions”. en. In: The Journal of the Astronautical Sciences 68.2 (June 2021), pp. 349–390. ISSN: 0021-9142, 2195-0571. DOI: `10.1007/s40295-021-00264-1`. (Visited on 05/22/2024).

[71] Guangyu Zhao et al. “An orbit determination approach to associating optical tracklets of space objects”. In: Acta Astronautica 200 (Nov. 2022), pp. 506–523. ISSN: 00945765. DOI: `10.1016/j.actaastro.2022.08.044`. (Visited on 05/02/2024).