POLITECNICO
MILANO 1863

SCUOLA DI INGEGNERIA INDUSTRIALE
E DELL'INFORMAZIONE

IN COLLABORATION WITH

uc3m | Universidad Carlos III de Madrid

A stochastic orbital propagation scheme applied to re-entry analysis

Tesi di Laurea Magistrale in
Space Engineering - Ingegneria Spaziale

Author: **Aurora Saracini**

Student ID: 10709895
Advisor: Prof. Juan Luis Gonzalo Gomez
Co-advisors: Yannick Sztamfater Garcia, Manuel Sanjurjo Rivo
Academic Year: 2023-2024

Copyright

Printed in Italy

Abstract

Recent years have seen a large increase in space traffic. This fact has generated the need to produce regulations for the sustainable use of space, as well as provide services to satellite operators and other involved actors. Among these services is re-entry analysis, both to verify that the regulations are complied with and to produce notices to air traffic managers or civil protection services. The primary goal of analysing a spacecraft's re-entry into the atmosphere is to determine the time and location of decay, which is crucial for assessing ground impact risks. This is a complex challenge due to the stochastic nature of trajectory evolution in the denser layers of the atmosphere. Even minor time variations can lead to significant differences in the spacecraft's touchdown location. Additionally, fragmentation of the S/C is highly probable, further complicating predictions.

One of the challenges when analysing re-entry is the correct management of the uncertainty. Uncertainty can come from non-modelled effects, for example, how uncontrolled attitude dynamics, or insufficient knowledge of the forces involved affect the re-entry evolution. In this study, the exploration of different methods to address the previous challenges in the initial stage of a re-entry is proposed. The goal is to implement a stochastic propagator to adequately quantify the uncertainty in the state propagation of a re-entering spacecraft, and compare it with a high-fidelity deterministic propagation scheme. Therefore, from a dynamical point of view, two models are to be compared: a simple point mass model with relatively basic representation of the perturbing forces, involving a stochastic propagation scheme, and a full 6-degree-of-freedom model, including attitude dynamics and involving deterministic propagation. The 6DOF simulator could include the possibility of having active control and will be validated against the data obtained during GOCE and Roseycubesat-1 re-entry. Regarding the integration, two separate schemes are employed: for the deterministic case a RK4 method is selected, while for the second case a Stochastic Runge-Kutta 4 is proposed, originally designed to tackle stochastic differential equations. This is possible if we model the spacecraft's dynamics as a stochastic system. The performance of the two algorithms will be compared, especially on the capability of the stochastic one to have a refined prediction with a less accurate dynamical model. The proposed approach will be tested in representative scenarios to

have a clear vision on the reliability of a stochastic propagator. A Monte Carlo method will be used for statistically obtaining the re-entry window.

Keywords: Stochastic propagation, re-entry prediction, uncertainty quantification, re-entry window, diffusion factor

Abstract in lingua italiana

Negli ultimi anni si è registrato un forte aumento del traffico spaziale. Questo fatto ha generato la necessità di produrre regolamentazioni per l'uso sostenibile dello spazio, oltre a fornire servizi agli operatori satellitari e agli altri attori coinvolti. Tra questi servizi vi è l'analisi del rientro, sia per verificare il rispetto delle normative sia per produrre avvisi per i gestori del traffico aereo o i servizi di protezione civile. L'obiettivo principale dell'analisi del rientro di un veicolo spaziale nell'atmosfera è determinare il momento e la posizione del decadimento, cruciale per valutare i rischi di impatto al suolo. Questo rappresenta una sfida complessa a causa della natura stocastica dell'evoluzione della traiettoria negli strati più densi dell'atmosfera. Anche variazioni minime nel tempo possono portare a differenze significative nel punto di impatto del veicolo spaziale. Inoltre, la frammentazione del S/C è altamente probabile, complicando ulteriormente le previsioni.

Una delle sfide nell'analisi del rientro è la corretta gestione dell'incertezza. L'incertezza può derivare da effetti non modellati, ad esempio, il modo in cui la dinamica di assetto non controllata o la conoscenza insufficiente delle forze coinvolte influenzano l'evoluzione del rientro. In questo studio si propone l'esplorazione di diversi metodi per affrontare le sfide precedenti nella fase iniziale di un rientro. L'obiettivo è implementare un propagatore stocastico per quantificare adeguatamente l'incertezza nella propagazione dello stato di un veicolo spaziale in rientro e confrontarlo con uno schema di propagazione deterministico ad alta fedeltà. Da un punto di vista dinamico, si confronteranno quindi due modelli: un semplice modello di massa puntiforme con una rappresentazione relativamente basica delle forze perturbanti, che prevede uno schema di propagazione stocastico, e un modello a sei gradi di libertà, che include la dinamica di assetto e prevede una propagazione deterministica. Il simulatore 6DOF potrebbe includere la possibilità di un controllo attivo e sarà validato rispetto ai dati ottenuti durante i rientri di GOCE e Roseycubesat-1. Per quanto riguarda l'integrazione, vengono impiegati due schemi separati: per il caso deterministico si seleziona un metodo di Runge-Kutta di ordine 4, mentre per il secondo caso si propone un Runge-Kutta stocastico di ordine 4, originariamente concepito per affrontare equazioni differenziali stocastiche. Ciò è possibile se modelliamo la dinamica del veicolo spaziale come un sistema stocastico. Le prestazioni dei due algoritmi saranno confrontate,

in particolare sulla capacità di quello stocastico di fornire una previsione raffinata con un modello dinamico meno accurato. L'approccio proposto sarà testato in scenari rappresentativi per avere una chiara visione sull'affidabilità di un propagatore stocastico. Un metodo Monte Carlo sarà utilizzato per ottenere statisticamente la finestra di rientro.

Parole chiave: Propagazione stocastica, previsione di rientro, quantificazione dell'incertezza, finestra di rientro, fattore di diffusione

Contents

1 | Introduction

1.1. The Space environment: re-entering objects

Since 1957, the amount of space debris in orbit has exceeded the number of operational satellites. Space debris has evolved into a global-scale issue, posing a significant threat to the long-term sustainability of the space environment [11]. This alarming situation has led to the need for internationally accepted space debris mitigation measures.

Analysing the trend up to the present day, the number of objects, along with their cumulative mass and cross-sectional area, has been steadily increasing. This growth is primarily fuelled by the deployment of large satellite constellations and various fragmentation events, among other factors. Fragmentations currently represent the predominant source of space debris [11].

Another critical aspect is re-entry: between 78% and 98% of payloads below 1000 kg comply with regulations and naturally dispose at the end of their life cycle [11]. The number of re-entering objects is also increasing exponentially, as it can be seen in Figure 1.1.

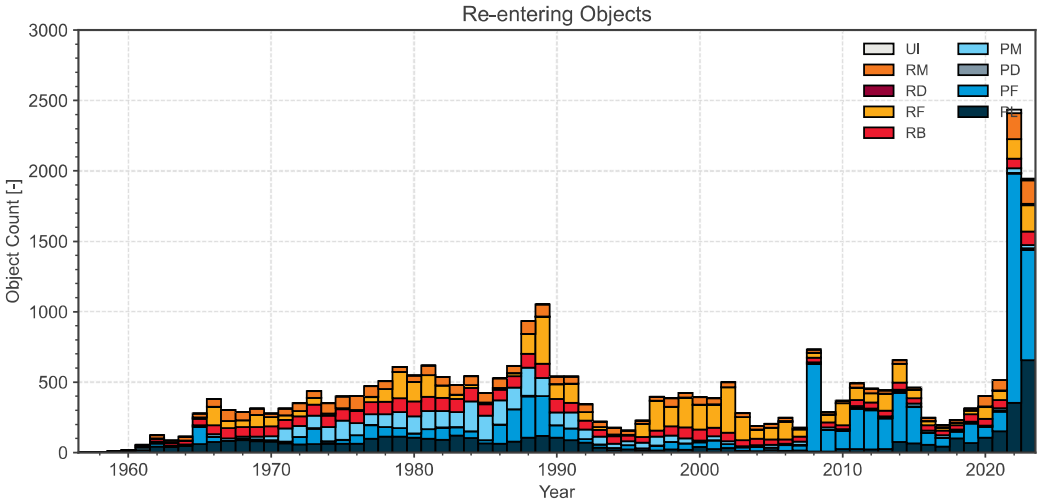


Figure 1.1: Evolution of re-entering objects in each year by object type without human spaceflight [11]

This is a positive development, as most of those satellites are now able to self-dispose, contributing to a cleaner orbital environment. Ensuring mission sustainability should be a shared goal, especially in the current era of mega-constellations. It has become crucial to have fast and reliable re-entry predictions to design optimal disposal strategies for each mission.

1.2. Uncertainty propagation methods

Predicting a re-entry remains one of the major challenges in this field due to the multitude of uncertainties involved. The most prominent and influential is the atmospheric uncertainty, which is heavily affected by solar activity. Additionally, accurately modelling the orbital dynamics, inclusive of all forces, is nearly impossible. Even the most precise models retain a degree of uncertainty and typically require significant computational time. It is therefore crucial to account for these uncertainties when predicting the re-entry of an object. Regarding uncertainty propagation, several methods are discussed in the literature. These can be broadly categorized into three main groups: linear methods, nonlinear methods, and Monte Carlo techniques. This section will outline some of these methods.

1.2.1. Linear methods

The first and simplest method presented is LinCov. This represents the most basic method for propagating uncertainty. The fundamental assumption in LinCov is the linearization of the dynamics, which is considered sufficiently accurate for describing the behavior of perturbed trajectories. The linearization is carried out by computing the STM. The objective of this method is to propagate only the mean and the covariance matrix. To achieve this, two key assumptions must be made [21]:

- a linearized model provides a sufficiently accurate approximation of the dynamics for trajectories close to the nominal one;
- the uncertainty is fully represented by a Gaussian probability distribution.

This method is analytical, offering the advantage of being extremely simple and, evidently, requiring the least computational time. However, it lacks sufficient accuracy for highly non-linear systems or when propagation periods are extended. Additionally, LinCov struggles to perform well in scenarios involving large uncertainties or non-Gaussian uncertainty propagation [21]. There is another linear method which does not linearize through the STM, but it linearizes the dynamics function in a statistical way, such that the existence of derivative of the dynamical function is not required [35]. This technique

offers a direct statistical analysis of non-linear stochastic systems exploiting linearization of the functions involved, having a much higher computational efficiency. It is particularly applicable to dynamical systems where derivatives are not defined. However, it necessitates prior knowledge of the PDF of the uncertain variable, and its accuracy diminishes significantly for strongly non-linear systems. Two other techniques that rely on linearisation are the Kalman Filter and Extended Kalman Filter [9]. Both methods process measurements of the object's state while predicting its orbital evolution. The KF assumes linear dynamics using a single reference trajectory, which is propagated by the model, estimating the actual trajectory based on the measurements. The EKF, however, continuously updates the reference trajectory with these measurements as it propagates. This makes the linearization more and more accurate.

1.2.2. Non-linear methods

This family of methods is designed to address and compensate for the issues posed by linearization.

Unscented transform

The first non-linear method introduced is the UT [9]. UT sacrifices the complete characterization of the final PDF for computational efficiency, focusing instead, on correctly computing the mean and covariance of the Gaussian distribution. This is achieved by parametrizing the initial PDF using a small number of samples. The process involves describing the initial statistics with a small set of samples, known as sigma points, to correctly capture the mean and covariance [9]. These sigma points are then propagated through the non-linear map, and the final mean and covariance are estimated from the sample mean and covariance of the propagated points. The major advantage of this method is that it requires only $2n+1$ sigma points, where n is the number of components of the vector to be propagated, to accurately propagate the uncertainty. Additionally, it provides a second-order approximation of the first two moments (mean and covariance matrix) of the mapped statistical distributions. However, the downside is that only the first two moments can be propagated non-linearly [21]. While the computational time is undoubtedly more efficient, greater accuracy can be achieved with other methods.

Polynomial chaos expansions

The PC method [37] approximates both the input and output of a system as standard random variables. By doing so, the same set of random variables represents both input

and output uncertainties. As a result, the output can be modelled as a series expansion of orthogonal polynomials. This method allows for the description of the entire PDF, including higher-order moments. PC efficiently represents solutions with finite variance, even in the case of non-Gaussian distributions. It is significantly faster than MC, requiring far fewer simulations, and is superior to UT as it can capture higher-order moments and the entire PDF. However, the key drawback is that, as the number of uncertainty variables increases, the number of polynomials required for propagation grows significantly. [21]

State transition tensors

A semi-analytic method, known as STT [25], is employed for orbit uncertainty propagation by solving for the higher-order Taylor series terms that describe localized nonlinear motion and by analytically mapping the initial uncertainties. This approach is advantageous as it does not require generating quasi-random numbers, unlike the previously mentioned methods. However, the drawback of this method is that the equations must be differentiable and continuous. In many cases, such as for high-fidelity dynamic systems, it may be inadequate due to the computational complexity involved in deriving high-order STT.[21]

Differential algebra technique

DA [15] provides a tool for computing the derivatives of functions, primarily utilized within computer programs. Instead of calculating each derivative individually, it generates Taylor series. This technique enables the computation of arbitrary-order expansions for the solution flow of a general ODE with respect to the initial conditions. For the DA-based MC simulation, as the accuracy of the Taylor expansion can be kept arbitrarily high (within its radius of convergence) by adjusting the expansion order, the approach of classical MC simulation can be enhanced by replacing thousands of integrations with evaluations of the Taylor expansion of the flow. As a result, the computational time reduces considerably without loss in accuracy. The DA-based uncertainty propagator has been widely applied to space-related missions. The great advantage of this method is that it does not require the integration of additional variational equations to obtain high-order expansions of the flow. This allows the uncertainty to be propagated non-linearly using these high-order expansions. However, its weakness is that it is quite similar to the STT method, meaning the dynamics of the problem must be continuous and differentiable. For instance, this limitation becomes evident when considering the inclusion of SRP in the model. Moreover, these techniques do not easily account for uncertainty in the dynamics, like the stochastic process does. [21]

Gaussian mixture model

The GMM method [30] is a probabilistic approach that approximates a complex, non-Gaussian PDF by breaking it down into smaller, simpler Gaussian components, which are combined using a weighted sum. Each component is characterized by its own mean and covariance. This method effectively transforms a non-Gaussian function into a Gaussian-like one, breaking a large problem into smaller, more manageable ones. It is particularly useful for handling large uncertainty propagation problems and reducing the effects of non-linearity. The major drawback of this method is that it suffers from dimensionality challenges, similar to PC. Achieving high accuracy requires a large number of Gaussian mixtures. Additionally, GMM is not recognized as a particularly fast method. In fact, optimization problems such as quadratic programming must be solved to determine the weights of the Gaussian mixtures, leading to a significant increase in runtime. [21]

1.2.3. Monte-Carlo simulation

MC simulation is a widely used technique for probabilistic analysis in engineering systems. It is a numerical simulation method employed to derive the statistical properties of a system's output, based on the known statistics of the input variables within a computational model [22]. The aim is to generate a set of random samples using the initial condition state vector exploiting the uncertainty introduced by the covariance matrix. In many cases, this method does not have a closed-form solution, so numerical methods are required to approximate it. Increasing the number of random samples improves the approximation due to the law of large numbers. Theoretically, as the number of samples approaches infinity, the approximation converges to the true posterior distribution [13]. In non-linear dynamical scenarios, accurate uncertainty propagation often relies on MC simulation to predict future probability distributions. MC simulation is particularly effective for propagating non-linear and non-Gaussian uncertainties and is well-regarded for its high accuracy and it is very easy to implement [21]. However, its primary drawback is the considerable runtime required.

Below, Table 1.1 presents a summary of all the methods.

Comparison of Uncertainty Propagation Methods

Method	Pros	Cons
Linear	Simple and fast for small uncertainties, efficient for Gaussian distributions	Low accuracy for highly nonlinear systems, struggles with non-Gaussian uncertainties
UT	Captures second-order moments, fewer sigma points required	Only propagates first two moments, less accurate for complex dynamics
PC	Represents entire PDF, higher-order moments, efficient for finite variance	Number of polynomials grows with dimensionality, complexity increases with more uncertainty variables
STT	High-order expansion without random numbers, semi-analytic solution	Requires differentiable, continuous equations, computational complexity with high-order expansions
DA	Arbitrary-order expansions, high accuracy, reduces computational time	Requires differentiable dynamics, limited to continuous systems
GMM	Approximates non-Gaussian PDFs with Gaussian components, reduces effects of non-linearity	Suffers from dimensionality challenges, optimization increases runtime
MC	High precision, well-suited for nonlinear uncertainties, easy to implement, convergence to true distribution	High computational time required, slow convergence for large sample sizes

Table 1.1: Comparison of the advantages and disadvantages of various uncertainty propagation methods described in this section.

1.3. Orbital uncertainty propagation - application

Orbital uncertainty propagation is frequently associated with SSA-related missions, such as tracking and data association, conjunction analysis, sensor resource management, and anomaly detection. Its primary focus is on tracking RSOs. Additionally, the outcomes of uncertainty propagation are utilized in trajectory optimization designs that prioritize

safety and robustness [21].

1.3.1. Trajectory estimation

Trajectory estimation is typically carried out using filtering techniques. Filters consist of two main components: the prediction and the update steps. Orbital uncertainty propagation, which usually involves propagating the covariance matrix, corresponds to the prediction phase of a filter, while the uncertain distribution of measurements and states affects the update phase. As a result, the prediction step of a filter can be directly applied to uncertainty propagation, and improvements in uncertainty propagation can, in turn, be used to better design the filter. Linear uncertainty propagation methods were initially developed based on the Kalman filter, the EKF [9], and the statistically linearized filter. These linear filters were extensively used in the early stages of human space exploration. To improve the EKF, the UKF [9] was introduced. Concurrent with the development of non-linear uncertainty propagation techniques, several non-linear filters were also proposed for space-related applications [21].

1.3.2. Sensitivity analysis

Sensitivity analysis for reliability assessment, aimed at examining the impact of various uncertainty sources on system performance, can effectively leverage uncertainty propagation. This approach is crucial for space mission design, where it is typically implemented using the MC method. For instance, it has been applied in sensitivity analysis for trajectory optimization, as well as for studying the influence and interaction of orbital uncertainties in rendezvous phasing orbital control. Moreover, it has been used to evaluate the impact of initial uncertainties on orbital conjunctions. Beyond space-related applications, sensitivity analysis using uncertainty propagation is also employed in other fields. For example, it has been utilized for global sensitivity analysis in hypersonic reentry flow computations. All these applications demonstrate the convenience and effectiveness of adopting uncertainty propagation, as it allows for identifying gaps in models or optimizing their efficiency [21].

1.3.3. Conjunction assessment

Collisions between space objects can only be determined probabilistically due to the inherent uncertainties in the parameters and motions of RSOs. As a result, conjunction assessment is typically based on uncertainty quantification results. Applying this approach to the current case is the most suitable solution. Calculating the collision probability be-

tween two objects, especially when one is debris that is difficult to track due to its small size, is a highly challenging and complex task. [12] Typically, the objects are modelled as spheres, and their positions are assumed to follow a Gaussian distribution. Additionally, the short-term evolution hypothesis can be applied, valid only for high relative velocity at encounter. Their motion is then simplified to occur along a line, where the potential collision could take place. From this foundation, the Akella and Alfriend algorithms [2] were developed, followed by contributions from Patera and others. With advancements in non-linear and non-Gaussian uncertainty propagators, more precise methods for computing collision probability P_c have been introduced. These advancements have enabled the application of various types of uncertainty propagation, significantly improving the reliability of conjunction estimation and enhancing the overall accuracy of collision predictions. [21]

1.3.4. Trajectory optimization

A less conventional application of uncertainty propagation can certainly be found in this area. Typically, the design of a nominal trajectory is carried out deterministically. However, incorporating statistical elements into this analysis could prove to be highly beneficial, as it may lead to a more accurate solution. This approach has the potential to reduce mission costs without sacrificing precision. Therefore, it is highly advisable to develop robust-optimal and stochastic-optimal manoeuvre strategies exploiting the results of orbital uncertainty propagation. This approach has been widely applied in spacecraft trajectory design. Some studies have proposed the computation of statistical mean trajectories using orbital uncertainty propagation as a substitute for the nominal trajectory. Additionally, an uncertain Lambert problem has been introduced [29], which accounts for uncertainties in both the initial and final states in Lambert's boundary-value problem. However, these methods are not universally applicable and must be carefully tailored to specific cases to ensure proper use and accuracy in trajectory design.

One particularly relevant application is the study conducted by Vasile et al. [33], which closely aligns with the current focus on re-entry. In their research, they performed a preliminary analysis of de-orbiting and re-entry dynamics, accounting for uncertainties in atmospheric density, drag coefficient, solar radiation pressure, and the area-to-mass ratio of the space object. Moreover, they also focused on the simplification from a higher-fidelity 6DOF model to a lower-fidelity 3-degree-of-freedom model, while accounting for the uncertainties introduced by this simplification. Since a full high-fidelity propagator is computationally expensive, it was interesting to observe how this approach performs with reduced complexity. This analysis provided valuable understanding of how the lower-

fidelity model behaves under uncertainty, offering a more efficient alternative while still maintaining acceptable accuracy. A Chebyshev high order polynomial approximation was used. The non-intrusive Chebyshev approach is a polynomial approximation of the function of interest [33]. From the preliminary results in this study, it can be concluded that for objects with a high area-to-mass ratio, a certain degree of stability can be expected. However, to accurately assess de-orbit time, it is essential to use a 6DOF model. In contrast, when an object is expected to tumble, a 3DOF analysis with a tumble-averaged drag coefficient offers a solid baseline for conducting uncertainty analysis. This provides a balance between computational efficiency and accuracy, depending on the object's expected behavior during re-entry.

Figure 1.2 reports a summary of what has been discussed until now.

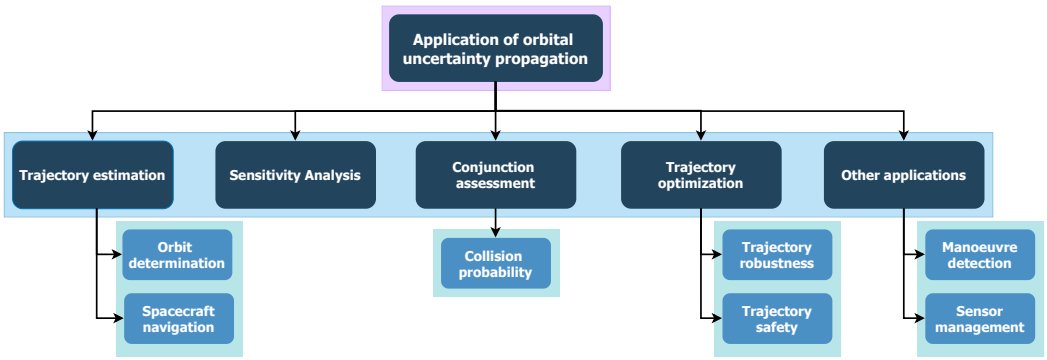


Figure 1.2: Summary of all the areas of application of orbital uncertainty propagation [21]

1.4. Re-entry prediction methods

Predicting a re-entry is undoubtedly one of the most challenging aspects of a mission. This process is highly complex due to the significant uncertainties involved. The specific challenges posed by the problem will vary depending on the type of re-entry. There are two distinct types of re-entry: controlled and uncontrolled. In a controlled re-entry, the target area can be predetermined, thus minimizing the risk of collisions. Conversely, during uncontrolled re-entry, predicting both the timing and location of descent becomes significantly more challenging [5]. Uncontrolled re-entry is particularly challenging to predict, as it is influenced solely by external forces. Various sources of uncertainty, such as physical phenomena and model approximations, impact the accuracy of predictions regarding re-entry time and location. Nonetheless, specific methodologies and procedures have been developed to improve the assessment of re-entry trajectories, offering a more reliable prediction despite the uncertainties. In this work, only the uncontrolled re-entry

will be addressed. To forecast the re-entry, the satellite's evolution must be analyzed across three distinct time periods: long-term, medium-term, and short-term evolution. In general, the orbital data of re-entering objects can be obtained from TLE data sets and propagated using various software tools. One common example is the SGP4 model [19].

1.4.1. Long- and medium-term evolution

To monitor the long- to medium-term evolution of an uncontrolled re-entry over time frames ranging from several years to a few weeks before the end of its lifetime, ESA uses a software called FOCUS-2 [18]. This software is based on principles similar to the FOCUS propagator employed in the MASTER model. FOCUS-2 integrates the combined time rates of change of singly averaged perturbation equations, taking into account factors such as the non-spherical Earth gravity potential, dynamic atmospheric effects, luni-solar gravitational perturbations, and solar radiation pressure, along with an oblate, cylindrical Earth shadow. The integration is performed using a robust fourth-order Adams–Bashforth/Adams–Moulton predictor-corrector method. Perturbations due to Earth's gravity potential are modelled using the Lagrange equations for a truncated EGM-96 gravity model up to the 23^{rd} degree. Additionally, a second-order Earth oblateness term, J22, is included. Third-body effects from the Sun and Moon are computed for point masses, using analytical ephemerides. As in the geopotential case, the perturbation equations are averaged with respect to the mean anomaly, keeping the positions of the Sun and Moon constant over the averaging time interval of one orbit. Solar radiation pressure perturbations are computed using Aksnes' theory, taking into account the area-to-mass ratio A/m and the momentum exchange coefficient c_R. These perturbation equations are averaged in closed form over the illuminated portion of the orbit, accounting for the oblate, cylindrical Earth shadow.

1.4.2. Short-term evolution

Analytical and semi-analytical orbit prediction methods operate under the assumption that perturbations remain nearly constant during the averaging interval, that the magnitudes of these perturbations are relatively small, and that their effects can be isolated without cross-coupling. However, these assumptions become invalid during the final days of a re-entry, particularly at altitudes below 200 km. At this stage, a fully numerical solution of the perturbed Newtonian equations is necessary, involving the integration of the osculating position and velocity vector, which accounts for perturbing forces and any potential orbital manoeuvres. At ESOC, the perturbation model for numerically integrating re-entry trajectories includes an Earth gravity field modelled by EGM-96, truncated

at degree and order 7, lunar and solar gravitational effects, solar radiation pressure with shadow effects, and atmospheric drag. Air densities are computed from a hybrid MSISe-90/CIRA-72 model, with drag coefficients adjusted for the flow regime. The solar and geomagnetic activity indices (F10.7, F10.7, and Ap) are used, incorporating both measurements and 27-day forecasts to calculate the air densities. The integration is carried out in a J2000 mean equatorial reference frame using an 8^{th}-order Runge–Kutta/Shanks method with adaptive step-size control. Any orbit manoeuvres are treated as impulsive velocity changes. At altitudes below 120 km, atmospheric drag becomes the dominant perturbation, with its effect increasing by an order of magnitude for every 30 km descent. Concurrently, c_D also changes significantly [18].

1.4.3. Tools developed for re-entry prediction

In recent years, with accumulated experience in re-entry prediction, various space agencies have developed software tools capable of providing reliable forecasts. These tools take into account numerous factors related to the re-entry process and often require a detailed S/C model for accurate simulation and analysis [7]. All the tools currently proposed rely on deterministic algorithms for propagation, without offering an in-depth analysis of the S/C re-entry window. Furthermore, they primarily focus on ensuring the satellite's compliance with space debris regulations, rather than concentrating on the detailed analysis of re-entry dynamics and its evolution.

DRAMA

One of the most widely used tools in the European space environment is the DRAMA software. This object-oriented tool is designed to assess mitigation strategies for both the operational and disposal phases of a mission. It evaluates the risks posed by a mission's space debris and the effectiveness of its end-of-life strategy, ensuring compliance with space debris mitigation requirements during mission design. DRAMA consists of five modules, one of which, called SARA, is specifically dedicated to re-entry prediction and risk analysis. SARA, which includes both the spacecraft entry survival analysis module and the spacecraft entry risk analysis module (SERAM), is specifically used to analyse the re-entry of objects into Earth's atmosphere. The simulation method employed by SARA is a straightforward 3DOF. SERAM performs detailed risk assessments for a given set of fragments impacting the ground, calculating casualty probabilities associated with individual fragments as well as the overall re-entry process. In general, DRAMA provides an efficient analysis of re-entry dynamics leveraging catalogued space debris data through advanced space environment modelling techniques. While this tool proves highly effective,

it would be valuable to explore whether stochastic propagation, by incorporating uncertainty to account for the lack of precise modelling, could offer a similarly straightforward yet more reliable approach.

SCARAB

SCARAB, a tool developed by ESA, has been employed in various re-entry scenarios. It is one of the first tools that included a spacecraft-oriented model. Initially, SCARAB was based on a deterministic approach, relying on a realistic physical model of the spacecraft and corresponding analytical methods. More recently, it has evolved to include finite element analyses using a triangular, panelized geometric approach to model the complete spacecraft. The re-entry trajectory is determined by computing the position and attitude of objects at each instant, derived from the 6DOF equations of motion through numerical integration. SCARAB integrates flight dynamics, aerodynamics, aerothermodynamics, and thermal-structural analyses to perform comprehensive re-entry risk assessments. The destruction and breakup phenomena are analysed at the panel level. [7] The complexity of this tool is high, making rapid and reliable re-entry predictions challenging. Additionally, it does not consider simplifying the orbital model, as proposed in this thesis.

1.5. Thesis contributions and structure

The primary aim of this thesis is to apply this stochastic propagation algorithm to satellite re-entry scenarios, enabling accurate yet highly efficient predictions. Such an approach has not been explored in the literature before, with the exception of the application of Chebyshev polynomials [33]. This work seeks to develop a completely new approach that could pave the way for increasingly precise predictions in the future. The main contribution of the thesis are the following:

- Validation of a deterministic and high-fidelity algorithm capable of predicting the re-entry of a simple satellite.
- Development of a novel stochastic propagator for satellite re-entry prediction. This propagator will operate faster than its deterministic counterpart by utilizing a simplified dynamical model, with reliability ensured through the incorporation of uncertainty to account for the simplifications.
- Performance testing of the proposed algorithm. The 3DOF stochastic algorithm will be compared against a 6DOF deterministic algorithm to evaluate how it performs under these simplifications. A simple satellite structure, such as Roseycubesat-1,

will be used in this test.

- A second performance test is conducted to analyse the stochastic propagator against a more refined deterministic propagator, such as HPOP [23]. In this case, both algorithms will operate with 3DOF, but the deterministic model will use a more sophisticated dynamical approach.

In line with the above mentioned contributions, the thesis is structured as follows.

- The theory behind the implementation of the propagators is fully described in chapter 2.
- The implementation choices for both algorithms, along with the validation of the deterministic propagator using Roseycubesat-1, are provided in chapter 3.
- The performance analysis of the stochastic against the deterministic propagator using Roseycubesat-1 and GOCE is discussed in chapter 4.
- The conclusions of the thesis are outlined in chapter 5.

2 | Dynamical model and stochastic propagation

Once the problem is properly introduced, the orbital model and the mathematical approaches for implementing both deterministic and stochastic propagators are presented. First, the perturbations and attitude dynamics equations used in the implementation are outlined. Following this, the mathematical theorems necessary for implementing the stochastic propagation are described.

2.1. Cowell's formulation

The simplest analytical method to propagate a satellite's orbit is through the Kepler's equation. However, this approach does not provide a complete representation of the orbit. The real motion of a body around the Earth must also account for the perturbative effects acting on it [31]. To achieve an accurate propagation, it is necessary to consider perturbations. One approach to account for these effects is to include the perturbing accelerations in the two-body equation, resulting in a more precise equation of motion [32]. The Cowell's formulation is the name given to a formulation of the satellite's dynamics modelled as a system of second-order differential equations for the position and velocity coordinates of the satellite [31]. The complete equation is:

$$\ddot{\vec{r}} = -\frac{\mu}{r^3}\vec{r} + \vec{a}_p \tag{2.1}$$

where $\vec{r}$ is the position vector in km, μ is the Earth gravitational parameter in $\frac{\text{km}^3}{\text{s}^2}$ and $\vec{a}_p$ the acceleration due to perturbation in $\frac{\text{km}}{\text{s}^2}$.

In this method, the equations are expressed in rectangular coordinates and integrated numerically [31]. The advantage of this method is that it allows perturbative effects to be added linearly. However, the challenge lies in accurately modelling the accelerations due to these perturbations. Retrieving these effects precisely is complex, and the resulting equation can be computationally expensive to solve [32].

There are several techniques available to solve these three second-order differential equations. However, they must first be reformulated into six first-order differential equations, as this is required to fully describe an orbit [32] using this method. All the equations are:

$$\vec{x} = \begin{bmatrix} \vec{r} \\ \vec{v} \end{bmatrix}, \dot{\vec{x}} = \begin{bmatrix} \vec{v} \\ -\frac{\mu}{r^3}\vec{r} + \vec{a}_p \end{bmatrix} \tag{2.2}$$

where $\vec{x}$ is the state vector in km and $\frac{\text{km}}{\text{s}}$ and $\vec{v}$ is the velocity vector in $\frac{\text{km}}{\text{s}}$.

2.1.1. Perturbation modelling

The acceleration $\vec{a}_p$ is influenced by various factors, and several of the most significant ones have been described.

Drag force

Undoubtedly, drag perturbation is the strongest force among all and plays a leading role in driving re-entries. It is also the most challenging to model, as it introduces the greatest uncertainty into the dynamical model. In fact, the atmosphere is highly complex to characterize, and accurately predicting its behaviour based on solar activity is nearly impossible.

Drag acts to circularize the orbit of the re-entering object, causing the eccentricity to decrease. The SMA also decreases because drag dissipates the S/C kinetic energy. The interaction between atmospheric drag and the rotating atmosphere causes the long-term oscillations and a decrease in inclination. Equation (2.3) provides the formulation of the acceleration due to drag [8].

$$\vec{a}_{\text{DRAG}} = \frac{1}{2}\rho\vec{v}_{\text{rel}}^2\left(\frac{C_D A}{m}\right) \tag{2.3}$$

where $\vec{a}_{\text{DRAG}}$ is the acceleration due to drag in $\frac{\text{m}}{\text{s}^2}$, ρ is the density of the atmosphere at the satellite altitude in $\frac{\text{kg}}{\text{m}^3}$, $\vec{v}_{\text{rel}}$ is the relative velocity of the satellite with respect to the Earth's atmosphere in $\frac{\text{m}}{\text{s}}$, C_D is the drag coefficient, A is the cross-sectional area of the satellite that faces directly the atmosphere in m^2, m is the satellite mass in kg.

J2

The disturbing perturbing potential in addition to the central gravity field can be distinguished into two contributions: zonal and tesseral harmonics. The zonal harmonics

consist of several degrees of expansion, with the second degree, known as J2, being the most relevant [8].

The J2 term causes secular variations of Ω and ω: a recession of the ascending node and an advance of the pericenter. J2 perturbation model is [8]:

$$\begin{aligned} a_{J2_x} &= -\frac{3}{2}\frac{\mu J2 R_\oplus^2}{r^5}\left(1-5\frac{z^2}{r^2}\right)x, \\ a_{J2_y} &= -\frac{3}{2}\frac{\mu J2 R_\oplus^2}{r^5}\left(1-5\frac{z^2}{r^2}\right)y, \\ a_{J2_z} &= -\frac{3}{2}\frac{\mu J2 R_\oplus^2}{r^5}\left(3-5\frac{z^2}{r^2}\right)z. \end{aligned} \tag{2.4}$$

where $\vec{a}_{J2}$ is the acceleration due to J2 decomposed in its components in ECI frame as a_{J2_x}, a_{J2_y}, a_{J2_z} in $\frac{\text{km}}{\text{s}^2}$. μ is the Earth gravitational parameter in $\frac{\text{km}^3}{\text{s}^2}$, $J2 = 0.0010826267$ is the spherical harmonic coefficient of degree two, $R_\oplus = 6378.1363$ is the Earth radius in km, r is the norm of the position vector in km, the components of the state vector are x, y and z in km.

J3

J3 represents the third-degree zonal harmonic in the Earth's gravitational potential. Qualitatively, its effects are similar to those of J2 [27], but with a lower magnitude.

The J3 perturbation is:

$$\begin{aligned} a_{J3_x} &= \frac{1}{2}\mu J3\frac{R_\oplus^3}{r^7}5x\left(7\frac{z^3}{r^3}-3\frac{z}{r}\right), \\ a_{J3_y} &= \frac{1}{2}\mu J3\frac{R_\oplus^3}{r^7}5y\left(7\frac{z^3}{r^3}-3\frac{z}{r}\right), \\ a_{J3_z} &= \frac{1}{2}\mu J3\frac{R_\oplus^3}{r^7}3\left(\frac{35}{3}\frac{z^4}{r^4}-10\frac{z^2}{r^2}+1\right)z. \end{aligned} \tag{2.5}$$

where $\vec{a}_{J3}$ is the acceleration due to J3 decomposed in its components in ECI frame as a_{J3_x}, a_{J3_y}, a_{J3_z} in $\frac{\text{km}}{\text{s}^2}$. $J3 = -0.0000025327$ is the spherical harmonic coefficient of degree three.

J4

J4 represents the fourth degree of the zonal harmonics in the gravitational potential. Its effects are somewhat similar to those of J2. However, they are significantly weaker than

those of J2 or J3. The modelling of J4 is:

$$
\begin{aligned}
a_{J4_x} &= \frac{1}{2}\mu J4 \frac{R_\oplus^4}{r^9} 35x \left(9\frac{z^4}{r^4} - 6\frac{z^2}{r^2} + \frac{3}{5}\right), \\
a_{J4_y} &= \frac{1}{2}\mu J4 \frac{R_\oplus^4}{r^9} 35y \left(9\frac{z^4}{r^4} - 6\frac{z^2}{r^2} + \frac{3}{5}\right), \\
a_{J4_z} &= \frac{1}{2}\mu J4 \frac{R_\oplus^4}{r^9} 35z \left(\frac{63}{5}\frac{z^4}{r^4} - 14\frac{z^2}{r^2} + 1\right).
\end{aligned}
\tag{2.6}
$$

where $\vec{a}_{J4}$ is the acceleration due to J4 decomposed in its components in ECI frame as a_{J4_x}, a_{J4_y}, a_{J4_z} in $\frac{\text{km}}{\text{s}^2}$. $J4 = -0.0000016196$ is the spherical harmonic coefficient of degree four.

Solar Radiation

SRP refers to the perturbation caused by direct electromagnetic radiation from the Sun, which exerts a pressure on objects in Earth orbit on the order of $10^{-6}\,\frac{N}{\text{m}^2}$. The primary effect of this force is to increase the eccentricity. However, this effect is not as pronounced in LEO [8]. The acceleration of the satellite due to SRP is:

$$
\vec{a}_{srp} = -p_{SR@1AU} AU^2 \frac{\vec{r}^2_{sc-Sun}}{||r_{sc-Sun}||^3} C_R \frac{A_{Sun}}{m}
\tag{2.7}
$$

where $\vec{a}_{srp}$ is the solar radiation pressure acceleration in $\frac{\text{m}}{\text{s}^2}$. $p_{SR@1AU}$ is the solar radiation pressure at 1 AU. $\vec{r}_{sc-Sun}$ is the position vector of the spacecraft with respect to the Sun, and $||\vec{r}_{sc-Sun}||$ is the norm of this position vector, representing the distance between the spacecraft and the Sun in km. C_R is the reflection coefficient of the spacecraft, which indicates how much solar radiation is reflected. A_{Sun} is the effective area of the spacecraft exposed to solar radiation in m^2.

Third-body perturbation

This perturbation may originate from any celestial body; however, the Moon and the Sun have the most significant influence on an Earth-orbiting object. The direct gravitational attraction of the Moon and the Sun results in secular changes to i, e, and ω. Equation (2.8) presents the expression for the acceleration caused by this perturbation on an orbiting S/C [8].

$$
\vec{a}_{3body} = \mu_{3body} \left(\frac{\vec{r}_{s/c-3body}}{r^3_{s/c-3body}} - \frac{\vec{r}_{Earth-3body}}{r^3_{Earth-3body}} \right)
\tag{2.8}
$$

where $\vec{a}_{3body}$ represents the acceleration due to the gravitational influence of a third body, in m/s^2, $\vec{r}_{s/c-3body}$ is the vector from the third body to the satellite in m, $\vec{r}_{Earth-3body}$ is the vector from the Earth to the third body in m.

2.2. Attitude dynamics

An accurate and comprehensive model requires the inclusion of the satellite's attitude dynamics. Simpler models may only implement 3DOF; however, propagation using a 6DOF method provides significantly greater accuracy compared to other algorithms. To model the dynamics of a satellite, the first step is to consider the Euler equations, which describe the evolution of a satellite's motion while orbiting the Earth. In this case, quaternions are used to represent the attitude dynamics.

Additionally, various perturbing effects are taken into account.

2.2.1. Euler equations

The Euler equations, as mentioned, describe the attitude motion of a rigid body. In this case, the rigid body is an orbiting object, which, for the purposes of this analysis, is assumed to be a simple spin-stabilized satellite with principal inertia axis. Its dynamics is [4]:

$$\begin{cases} I_x\dot{\omega}_x + (I_z - I_y)\omega_y\omega_z = T_x \\ I_y\dot{\omega}_y + (I_x - I_z)\omega_z\omega_x = T_y \\ I_z\dot{\omega}_z + (I_y - I_x)\omega_x\omega_y = T_z \end{cases} \tag{2.9}$$

The terms $\dot{\omega}_x$, $\dot{\omega}_y$, and $\dot{\omega}_z$ represent the components of the angular acceleration $\frac{\text{rad}}{\text{s}^2}$. I_x, I_y, and I_z are the moments of inertia with respect to the x, y, and z axes in kgm^2, respectively. ω_x, ω_y, and ω_z are the components of the angular velocity in $\frac{\text{rad}}{\text{s}}$. T_x, T_y and T_z are the components of the external torque acting on the satellite in Nm.

2.2.2. Drag

The interaction between satellite and upper atmosphere generates aerodynamic forces, that in turn can generate a torque about the centre of mass. Aerodynamic torques represent the major disturbance for satellites orbiting at altitudes lower than 400 km. To have an approximated mathematical model of the forces, it is assumed that air particles hit the external surface of the satellite and their kinetic energy is totally transferred to the satellite. The aerodynamic force acting on the elementary area dA defined by its

perpendicular direction $\hat{N}$ is [4]:

$$d\vec{F} = \frac{1}{2} C_D \rho \vec{v}^2 (\hat{N} \cdot \vec{v}) \vec{u}_v dA \tag{2.10}$$

where $d\vec{F}$ is the infinitesimal aerodynamic drag force acting on the surface element dA of the spacecraft in N. $\hat{N}$ is the unit vector normal to the surface element, and $\vec{u}_v$ is the unit vector in the direction of the relative velocity.

Equation (2.11) reports the drag torque expression.

$$\vec{T}_{drag} = \int_s \vec{r}_{surf} \times d\vec{F} \tag{2.11}$$

where $\vec{T}_{drag}$ is the drag torque acting on the spacecraft in Nm. The integral is taken over the spacecraft surface area s. $\vec{r}_{surf}$ is the position vector from the spacecraft's centre of mass to the surface element in m, and $d\vec{F}$ is the infinitesimal aerodynamic drag force acting on the surface element computed in Equation (2.10) in N.

2.2.3. Gravity gradient

The gravity field is not uniform, therefore there could be a torque acting on the satellite. This is mainly true for large satellites, and even if the resulting torque is small the effect can be considerable due to the long time of action. This torque is generated by the same force field that causes the orbital motion, so it is acting continuously, and it depends on the attitude of the satellite.

Equation (2.11) reports GG torque in LVLH frame.

$$\vec{T}_{gg} = \frac{3\mu}{r^3} \begin{cases} (I_z - I_y) c_2 c_3 \\ (I_x - I_z) c_1 c_3 \\ (I_y - I_x) c_1 c_2 \end{cases} \tag{2.12}$$

where $\vec{T}_{gg}$ is the gravity gradient torque in LVLH frame in Nm. The terms c_1, c_2, and c_3 are the direction cosines of the radial direction in the principal axes.

2.2.4. Solar radiation pressure

When solar radiation illuminates the outer surface of a satellite, it generates pressure, which in turn produces both a force and a torque about the satellite's centre of mass. The primary sources of electromagnetic radiation include direct solar radiation, solar radiation

reflected by the Earth, and radiation emitted directly by the Earth. The intensity of solar radiation decreases with the square of the distance from the source. Radiation forces can be modelled by assuming that a portion of the incident radiation is absorbed, another portion is reflected, and the last part is diffused. Combining these three components, the total radiation force is evaluated as in Equation (2.13) [4].

$$d\vec{F}_{srp} = -p_{SR@1AU} \int_s \left[(1 - c_a)\hat{S} + 2(c_s \cos\theta + \frac{1}{3}c_d)\hat{N} \right] \cos\theta dA \tag{2.13}$$

where $d\vec{F}_{srp}$ is the infinitesimal solar radiation pressure force acting on the spacecraft. The integral is taken over the spacecraft surface area s. c_a, c_s, and c_d are the absorption, specular reflection, and diffuse reflection coefficients, respectively. $\hat{S}$ is the unit vector in the direction of the incoming sunlight, and $\hat{N}$ is the unit vector normal to the surface element. θ is the angle between the sunlight direction and the surface normal, and dA is the surface area element.

Equation (2.14) reports the SRP torque expression.

$$\vec{T}_{srp} = \int_s \vec{r}_{surf} \times d\vec{F}_{srp} \tag{2.14}$$

where $\vec{T}_{srp}$ is the SRP torque acting on the spacecraft in Nm. The integral is taken over the spacecraft surface area s, and $d\vec{F}_{srp}$ is the infinitesimal SRP force acting on the surface element computed in Equation (2.13) in N.

2.2.5. Magnetic torque

The magnetic torque is generated according to the general law $\vec{T} = \vec{m} \times \vec{B}$, where $\vec{m}$ represents the residual magnetic induction caused by parasitic currents in the satellite, and $\vec{B}$ is the Earth's magnetic field. Typically, $\vec{m}$ is an undesired effect, whereas $\vec{B}$ is always present. Various models allow to evaluate the components of $\vec{B}$ based on the satellite's position. Therefore, magnetic torques acting on a satellite are influenced not by its inertia properties, but by its position and attitude [4].

2.3. HPOP

High Precision Orbit Propagator, as its name suggests, is a high-fidelity orbital propagator developed by Meysam Mahooti [23].

Given the satellite's position and velocity at a specific time, HPOP propagates the object's

position by calculating its acceleration and numerically integrating the resulting second-order ODE. This enables significantly more accurate orbit propagation compared to other propagators like SGP4 and SDP4, albeit with a substantially higher computational cost.

The forces and effects included in the model encompass Earth's gravitational attraction, ocean and solid tide effects, gravitational influences from the Moon, Sun, and planets, SRP, atmospheric drag, and relativistic effects. Atmospheric density is calculated using the NRLMSISE-00 model for drag estimation, and the FES2014 model is employed for Earth geopotential corrections due to ocean tides. Additionally, the central body for the frame of reference can be the Solar System Barycenter, any major planet, Earth's Moon, or Pluto [3].

2.4. Runge-Kutta 4

The RK4 method is a single-step integration scheme highly employed to solve ODEs [1]. Single-step methods combine the state at a given time with rates at several other times, based on the single-state value at time t_0. These rates are directly obtained from the equations of motion and allow for determining the state at subsequent times, $t_0 + h$, with h being the time step [32]. This explicit method provides a highly accurate solution without the need to solve any high-order derivatives.

It utilizes multiple stages: for an N-order Runge-Kutta method, the first $N - 1$ stages are defined as "predictors," while the N^{th} stage is known as the "corrector." Referring to Equation (2.15), we can see that k_0, k_1 and k_2 are the predictors and k_3 is the corrector.

$$\begin{aligned}
\vec{k}_0 &= f(\vec{x}_n, t_n) \\
\vec{k}_1 &= f\left(\vec{x}_n + \frac{1}{2}h\vec{k}_0\right) \\
\vec{k}_2 &= f\left(\vec{x}_n + \frac{1}{2}h\vec{k}_1\right) \\
\vec{k}_3 &= f\left(\vec{x}_n + \frac{1}{2}h\vec{k}_2\right)
\end{aligned} \tag{2.15}$$

where $\vec{x}$ is the state vector in km and $\frac{\text{km}}{\text{s}}$, n is the current step and h is the time-step in m. To obtain the state vector at the subsequent time step, the final formula of the RK4 method is derived from the weighted average of the four increments defined in Equation (2.15) [1]. The expression is presented in Equation (2.16).

$$\vec{x}_{n+1} = \vec{x}_n + \frac{1}{6}[\vec{k}_0 + 2\vec{k}_1 + 2\vec{k}_2 + \vec{k}_3]h \tag{2.16}$$

where all the k factors are obtained as in Equation (2.15).

2.5. Wiener process and Stochastic Runge-Kutta

A multidimensional SDE is defined as in Equation (2.17).

$$d\vec{x}_t = f_t(\vec{x}_t)dt + G(\vec{x}_t)dW_t \tag{2.17}$$

where t is the continuous time, $x_t \in \mathbb{R}^{d_x}$ is the d_x-dimensional state, $f_t : \mathbb{R}^{d_x} \times \mathbb{R} \to \mathbb{R}^{d_x}$ is the dynamics equation, G is the diffusion factor which is the scale parameter that gives the intensity of the stochastic perturbation [13] [26] and W_t is the $d_x \times 1$ vector of independent Wiener processes and considers model errors and uncertainties [13].

Every time-continuous process with independent increments and bounded individual variances is a Wiener Process. This stochastic process is characterized by having variances equal to the time elapsed between time-steps, making the elemental differential Wiener process a Gaussian distribution with zero mean, and variance of $t_2 - t_1$. It is defined as $dW_t = \sqrt{dt}$ and is the cornerstone of what is known as Itô Calculus, which lays out the foundations of SDE calculus. The equation (2.17) does not have a closed-form solution for a general non-linear function, as in this case. It must be discretized. This can be achieved using a fixed step-size $h > 0$. The discretization scheme leads to the solution presented in Equation (2.18) [26].

$$\vec{x}_{n+1} = \vec{x}_n + F(\vec{x}_n, h, G\vec{w}_n) \tag{2.18}$$

$\vec{w_n}$ is a random variable of dimension d_w higher or equal to the dimension of the state vector, which results from the integration of d_w independent Wiener processes. The function F depends on the discretization scheme chosen. The simplest scheme for the aforementioned discretization is the Euler-Maruyama method [26]. The final formula is in Equation (2.17), written in the beginning. For the fourth-order Runge-Kutta method, as in this case, $d_w = 4d_x$, meaning that F corresponds to f applied four times.

Regarding the diffusion factor G, there are various approaches to model it. The assumption made is that G remains constant with respect to both time and the state $x(\vec{t})$. A potential future development could involve modelling the diffusion factor to depend on the state or time. It can be modeled as dependent solely on acceleration, which represents the simpler approach. The models are listed below:

- $G = \sigma_G I_{3x3}$, where σ_g is a scale parameter and I_{3x3} is the $3x3$ identity matrix;

- $G = \sigma_G \sum(\lambda, s_1, s_2)$ where λ is a shape parameter, $s_i \in \pm 1$ determines the type of correlation and $\sum(\lambda, s_1, s_2)$ is:

$$\sum(\lambda, s_1, s_2) = \begin{bmatrix} 1 & s_1 \exp -\lambda & s_2 \exp -2\lambda \\ s_1 \exp -\lambda & 1 & s_1 \exp -\lambda \\ s_2 \exp -2\lambda & s_1 \exp -\lambda & 1 \end{bmatrix} \tag{2.19}$$

- $G = \text{diag}(\sigma_x, \sigma_y, \sigma_z)$, where σ_x, σ_y, σ_z are scale parameters along the three axes.
- G is modelled as:

$$G = \begin{bmatrix} G_{11} & 0 & 0 \\ G_{21} & G_{22} & 0 \\ G_{31} & G_{32} & G_{33} \end{bmatrix} \tag{2.20}$$

2.6. Monte-Carlo method

The formal definition of Monte Carlo is "representing the solution of a problem as a parameter of a hypothetical population, and using a random sequence of numbers to construct a sample of the population, from which statistical estimates of the parameter can be obtained." [16]. It is a numerical simulation method employed to derive the statistical properties of a system's output, based on the known statistics of the input variables within a computational model [22]. The Monte Carlo method is estimating the properties of the highly complex probability distribution $p(x)$, by computing the expected value as in Equation (2.21) [6].

$$E[f(x)] = \int h(x)p(x)dx \tag{2.21}$$

where h is supposed to be a useful function for the estimation. Unfortunately, this cannot be achieved analytically. The approximation problem can be addressed indirectly by generating N random samples from the distribution p, so x_i, with $i = 1, ..., N$. The function $p(x)$ is approximated by point masses, so that $\overline{h}(x)$ results as reported in Equation (2.22) [6].

$$E[f(x)] \approx E_N[f(x)] = \frac{1}{N} \sum_{i=1}^{N} \overline{h}(x_i) \tag{2.22}$$

In this study, the problem to be addressed involves the orbit dynamics of a satellite during re-entry. This implies that, in determining the trajectory, the MC method requires generating N samples based on the initial conditions of the satellite at a specific point in its orbit. Therefore, in this case, the x_i in Equation (2.22) represents the state vector. In

cartesian coordinates assuming the origin to be the centre of the Earth, the altitude over the Earth surface for the i^{th} sample is defined as:

$$H_i = r_i - R_\oplus \tag{2.23}$$

Here, H_i represents the altitude of the S/C in kilometres, r_i is the magnitude of the position vector $\vec{r}_i$, also in kilometres, and $R_\oplus$ is the Earth's radius in kilometres. By setting an altitude threshold beyond which a sample is considered to have decayed and is no longer propagated, it becomes possible to calculate key statistics, such as the re-entry time window, the variance related to this estimate, and the mean re-entry time. Naturally, increasing the number of random samples enhances the approximation, in accordance with the law of large numbers. In theory, as the number of samples tends to infinity, the approximation converges to the true posterior distribution [13]. The propagation for each individual sample is terminated when the altitude falls below the predefined threshold for re-entry, set at 115 km. In non-linear dynamical scenarios, accurate uncertainty propagation frequently depends on MC simulation to predict the future probability distribution. Consequently, this method is regarded as reliable and has been extensively used to validate both linear and non-linear approximation uncertainty propagators. MC simulation offers a method for propagating non-linear and non-Gaussian uncertainties. It is also known for its high precision and ease of implementation [21].

3 | Implementation and validation

As previously mentioned, two types of algorithms have been implemented and utilized: one deterministic and one stochastic. Once the theory has been discussed, this section will address their implementation. Following that, the deterministic algorithm will also be validated to serve as a baseline.

3.1. Reference Frames

Before proceeding with the discussion, a description of the reference frames used is provided. The main frames employed are ECI and body frames.

3.1.1. Earth-Centred Inertial reference frame

The Earth-centred inertial reference frame [36] is fixed and inertial and it is centred at the Earth's centre. The axes are defined as follows:

- The x-axis points towards the March equinox, occurring when the Sun crosses the equatorial plane from below. Its unit vector is denoted as $\hat{\mathrm{i}}_{\mathrm{eci}}$.
- The z-axis is directed towards the Earth's North Pole, with its unit vector represented by $\hat{\mathrm{k}}_{\mathrm{eci}}$.
- The y-axis is orthogonal to both the x- and z-axes, lies within the equatorial plane, and completes the right-handed coordinate system. Its unit vector is indicated as $\hat{\mathrm{j}}_{\mathrm{eci}}$.

3.1.2. Body frame

The body reference frame [4] is centred at the centre of mass of the S/C. The body frame, with unit vectors $[\hat{\imath}_b, \hat{\jmath}_b, \hat{k}_b]$, is an orthonormal frame fixed to the S/C and rotates with it. This allows the orientation of a rigid body to be represented as the relative attitude between the body-fixed frame and the ECI frame.

3.2. Cowell's formulation

The result of each model is the acceleration generated by each disturbance, and these accelerations are linearly summed to produce the total perturbation acceleration, $\vec{a}_p$, used in Cowell's formulation as detailed in section 2.1. Given that the dynamics model is intended to predict a re-entry, not all perturbations were taken into account. The disturbances that were considered for the propagation of the orbit are listed below.

- NRLMSISE-00 Drag force
- J2
- J3
- J4

J2, J3 and J4 were implemented as section 2.1 describes.

3.2.1. Drag force modelling

Modelling this disturbance is not straightforward, especially when the cross-sectional area requires accurate representation. In simpler cases, assuming a constant cross-sectional area may suffice. However, in this work, a more comprehensive approach has been adopted. The first step is to accurately represent the contribution of the ballistic coefficient to the re-entry. This begins by expressing the body axes in the ECI reference frame, as shown in Equation (3.1).

$$\begin{aligned}
\hat{i}_{b_x} &= 1 - 2\left(q_3^2 + q_4^2\right), \quad \hat{j}_{b_x} = 2\left(q_2 q_3 - q_1 q_4\right), \quad \hat{k}_{b_x} = 2\left(q_2 q_4 + q_1 q_3\right) \\
\hat{i}_{b_y} &= 2\left(q_2 q_3 + q_1 q_4\right), \quad \hat{j}_{b_y} = 1 - 2\left(q_2^2 + q_4^2\right), \quad \hat{k}_{b_y} = 2\left(q_3 q_4 - q_1 q_2\right) \\
\hat{i}_{b_z} &= 2\left(q_2 q_4 - q_1 q_3\right), \quad \hat{j}_{b_z} = 2\left(q_3 q_4 + q_1 q_2\right), \quad \hat{k}_{b_z} = 1 - 2\left(q_2^2 + q_3^2\right)
\end{aligned} \tag{3.1}$$

$\hat{i}_{b_x}$, $\hat{i}_{b_y}$, $\hat{i}_{b_z}$ are the components of the vector $\hat{i}_b$ rotated in ECI frame, $\hat{j}_{b_x}$, $\hat{j}_{b_y}$, $\hat{j}_{b_z}$ are the components of the vector $\hat{j}_b$ rotated in ECI frame, $\hat{k}_{b_x}$, $\hat{k}_{b_y}$, $\hat{k}_{b_z}$ are the components of the vector $\hat{k}$ rotated in ECI frame, q_1, q_2, q_3, q_4 are the components of the quaternion representing the attitude of the body with respect to an ECI frame.

In this case, specifically, $B = \frac{C_D A}{m}$ must be projected onto the axes expressed in Equation (3.1), allowing for a realistic contribution from each face of the body. Equation (3.2) shows how this projection is performed.

$$
\begin{aligned}
B_{ip} &= B \cdot \max\left(0, \operatorname{diag}\left(\hat{i}_b \cdot \hat{u}_v\right)\right) \\
B_{jp} &= B \cdot \max\left(0, \operatorname{diag}\left(\hat{j}_b \cdot \hat{u}_v\right)\right) \\
B_{kp} &= B \cdot \max\left(0, \operatorname{diag}\left(\hat{k}_b \cdot \hat{u}_v\right)\right) \\
B_{in} &= B \cdot \max\left(0, \operatorname{diag}\left(-\hat{i}_b \cdot \hat{u}_v\right)\right) \\
B_{jn} &= B \cdot \max\left(0, \operatorname{diag}\left(-\hat{j}_b \cdot \hat{u}_v\right)\right) \\
B_{kn} &= B \cdot \max\left(0, \operatorname{diag}\left(-\hat{k}_b \cdot \hat{u}_v\right)\right)
\end{aligned}
\tag{3.2}
$$

where B is the inverse of the ballistic coefficient, B_{ip}, B_{jp}, B_{kp} are the components of the inverse of the ballistic coefficient along the positive directions of the body-fixed reference frame in $\frac{\mathrm{m}^2}{\mathrm{kg}}$. B_{in}, B_{jn}, B_{kn} are the components of the inverse of the ballistic coefficient along the negative directions of the body-fixed reference frame in $\frac{\mathrm{m}^2}{\mathrm{kg}}$. $\hat{i}_b$, $\hat{j}_b$, $\hat{k}_b$ are the unit vectors along the axes of the body-fixed reference frame projected in ECI frame. $\hat{u}_v$ is the unit vector of the relative velocity of the satellite with respect to the atmosphere.

In this way, only three of the six coefficients resulting from Equation (3.2) will be non-zero. In fact, between the positive and negative directions of the three body axes, only one will contribute—specifically, the one with a value greater than zero. The other, being negative, is set to zero and will not influence the final calculation of B. This is to respect the real physics, since drag does not act on both faces of a panel at the same time. The value of B is then calculated simply by linearly summing the six coefficients.

The coefficient B is then used to compute $\vec{a}_{\mathrm{DRAG}}$ as reported in Equation (2.3).

3.3. Attitude dynamics

Given that the altitude is not expected to be high, only Gravity Gradient and drag torque effects on the analysed body are considered.

3.3.1. Angular acceleration from drag torque

A priori, the face(s) directly exposed to the atmosphere are not known. This uncertainty must be taken into account and the angular acceleration has to be calculated accordingly. The angular acceleration components due to the drag force acting on the satellite can be modeled as in Equation (3.3) [4].

$$\begin{aligned}\vec{\dot{\omega}}_{\text{DRAG}} = &-B_{ip}\rho\left(v_{\text{rel}}\right)^2/2\hat{i}_b + B_{in}\rho\left(v_{\text{rel}}\right)^2/2\hat{i}_b \\ &- B_{jp}\rho\left(v_{\text{rel}}\right)^2/2\hat{j}_b + B_{jn}\rho\left(v_{\text{rel}}\right)^2/2\hat{j}_b \\ &- B_{kp}\rho\left(v_{\text{rel}}\right)^2/2\hat{k}_b + B_{kn}\rho\left(v_{\text{rel}}\right)^2/2\hat{k}_b \end{aligned} \tag{3.3}$$

where $\vec{\dot{\omega}}_{\text{DRAG}}$ is the angular acceleration due to the drag perturbation acting on the satellite in $\frac{\text{rad}}{\text{s}^2}$, ρ is the atmospheric density at the satellite's altitude in $\frac{\text{kg}}{\text{m}^3}$, v_{rel} is the relative velocity of the satellite with respect to the atmosphere in $\frac{\text{m}}{\text{s}}$. B_{ip}, B_{in}, B_{jp}, B_{jn}, B_{kp}, and B_{kn} are coefficients related to the drag distribution along the satellite's surfaces. All the components of B are the same ones employed to compute the cross-sectional area exposed to drag in section 3.2.1 (see Equation (3.2)). This means only maximum 3 out 6 components will be contributing to $\vec{\dot{\omega}}_{\text{DRAG}}$, all the others will be zero.

3.3.2. Angular acceleration from gravity gradient torque

The challenge in modeling the GG torque lies in the fact that it is typically calculated in the LVLH frame. However, in this case, the moments of inertia are expressed in the body frame. One approach to derive the equations is by first expressing them in the body axes, and then rotating them back into the ECI reference frame [4].

The Direction Cosine Matrix using quaternions is reported in Equation (3.4).

$$\mathbf{A}_{bn} = \begin{bmatrix} q_1^2 - q_2^2 - q_3^2 + q_4^2 & 2(q_1q_2 + q_3q_4) & 2(q_1q_3 - q_2q_4) \\ 2(q_1q_2 - q_3q_4) & -q_1^2 + q_2^2 - q_3^2 + q_4^2 & 2(q_3q_2 + q_1q_4) \\ 2(q_1q_3 + q_2q_4) & 2(q_3q_2 - q_1q_4) & -q_1^2 - q_2^2 + q_3^2 + q_4^2 \end{bmatrix} \tag{3.4}$$

where $\mathbf{A}_{bn}$ is the DCM that represent the rotation from ECI to body frame, q_1, q_2, q_3, and q_4 are the quaternion components.

The position vector in body axes is then obtained by multiplying the DCM by the position vector in the inertial frame as in Equation (3.5) [4].

$$\vec{r}_b = \mathbf{A}_{bn}\vec{r} \tag{3.5}$$

where $\vec{r}_b$ is the position vector in m in the body frame, derived from the position vector $\vec{r}$ in ECI reference frame in m.

The GG torque in body axes is calculated using Equation (3.6) [4].

$$\vec{T}_{gg_b} = \frac{3\mu}{\vec{r}^3}\left[\frac{\vec{r}_b}{\|\vec{r}_b\|} \times \left(\mathbf{I_t}\frac{\vec{r}_b}{\|\vec{r}_b\|}\right)\right] \tag{3.6}$$

where $\vec{T}_{gg_b}$ is the GG torque in the body frame in Nm and $\mathbf{I_t}$ is the matrix of the satellite's moments of inertia in kgm^2.

In order to obtain the final angular acceleration due to GG torque, $\vec{T}_{gg_b}$ has to be rotated back in ECI by multiplying it with the transpose of $\mathbf{A}_{bn}$ and dividing it by the corresponding inertia moment.

3.4. SRK4 and diffusion factor modeling

There are several ways to implement the diffusion factor G. The simplest way is to assume an uncertainty which is independent from time, velocity and acceleration. In this way, the uncertainty is not increasing while propagating and it is easier to retrieve similar results compared to the one of the deterministic algorithms.

$$G = \sigma_i, \text{ with } \sigma_i = 1,...,6 \tag{3.7}$$

Each σ, one per state vector component, is composed of the following components:

- A part that must be adjusted according to the simplifications applied in the model. In this thesis, the focus has been on adding uncertainty solely to the acceleration. For instance, if the SRP force is not modelled and the algorithm must compensate for its absence on the S/C, it is necessary to consider uncertainty equivalent to the SRP acceleration at the orbiting altitude.
- A randomly generated number representing the randomness in the stochastic problem.

The implementation of SRK4 results then as reported in Equation (3.8) and Equation (3.9).

$$\begin{aligned}
\vec{k}_0 &= f(\vec{x}_n, t_n) \\
\vec{k}_1 &= f\left(\vec{x}_n + \frac{1}{2}h\vec{k}_0 + \frac{1}{2}G\right) \\
\vec{k}_2 &= f\left(\vec{x}_n + \frac{1}{2}h\vec{k}_1 + \frac{1}{2}G\right) \\
\vec{k}_3 &= f\left(\vec{x}_n + \frac{1}{2}h\vec{k}_2 + \frac{1}{2}G\right)
\end{aligned} \tag{3.8}$$

$$\vec{x}_{n+1} = \vec{x}_n + \frac{1}{6}[\vec{k}_0 + 2\vec{k}_1 + 2\vec{k}_2 + \vec{k}_3]h + G \tag{3.9}$$

The expressions are quite similar to those described in Equation (2.16) and Equation (2.15), with the only difference being that G is added to the formulas. The time step h is fixed and set to 10 s.

3.5. Validation

The performance analysis of the stochastic propagator has been conducted in two ways. In one approach, it is compared with an in-house deterministic propagator. The model is implemented as described in section 3.2 and is a 6DOF propagator. The uncertainty on the initial state of the satellite is propagated with MC. Before adopting it as the baseline for the performance analysis, it needs to be validated. To do so, it is compared with a real re-entry case. Roseycubesat-[20] is used as the test case. It is an educational mission with a straightforward objective: taking basic pictures, providing telemetry, and promoting radio amateur activities.

Table 3.1 reports all the data needed for the propagation.

CubeSat Specifications

Parameter	Value
Initial time of the propagation [17]	23/08/2024 02:22:51 UTC
Initial state vector in ECI [17]	[-5208, 4172, 12.3, 0.6, 0.8, 7.7] km and $\frac{\text{km}}{\text{s}}$
Estimated date of re-entry [17]	01/09/2024 at 00:00 UTC
Variance vector in RTN frame	[300, 600, 300, 10^{-4}, 10^{-4}, 10^{-4}] m and $\frac{\text{km}}{\text{s}}$
Dimensions [20]	10 cm x 10 cm x 10 cm
Mass [20]	1 kg
Inertia vector	[1, 1, 1] * 0.00167 kgm^2
C_d	1.6
Re-entry altitude	115 km
Atmospheric model	NRLMSISE-00

Table 3.1: CubeSat specifications for the propagation analysis.

The following points should be underlined:

- The initial time and state of Roseycubesat-1 are retrievable from its TLE available on the Celestrak website [17].
- Celestrak [17] does not report a precise time of the re-entry. In fact, the TLE data set contains orbital data only until the 31/08/2024 at 06:11:38 UTC, when Rosey reached 200 km of altitude. As a consequence, the validation of the algorithm is done based on this available.
- The variance vector is assumed based on the precision of the TLE format. It was in RTN frame, but it was properly rotated in ECI.
- The value of C_D is assumed according to average literature values [4].

The goal of the validation is first to confirm that the deterministic model's re-entry prediction aligns with the date reported by Celestrak [17]. Roseycubesat-1 re-entered on September 1$^{\text{st}}$, 2024, so ideally, the re-entry window should be centred around this date. To generate an accurate PDF that effectively represents the prediction, the deterministic

propagator integrates 1000 samples generated using the previously mentioned covariance matrix. Figure 3.1 displays the re-entry window resulting from this propagation.

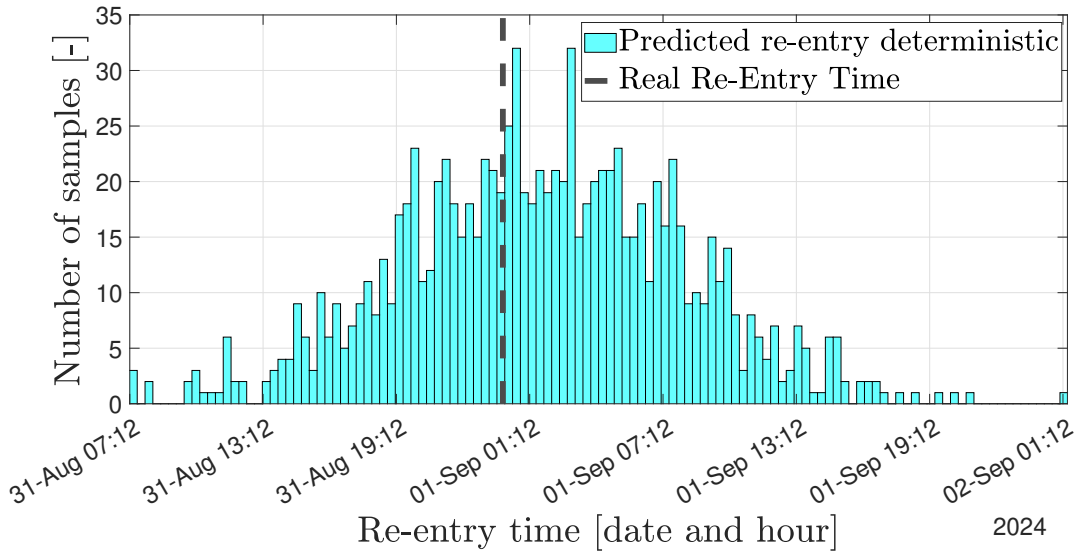


Figure 3.1: Predicted re-entry window from the deterministic algorithm compared to the actual re-entry time of Roseycubesat-1 reported in Celestrak website [17].

The first step of the validation is successful. Figure 3.1 presents both the PDF of the propagator's prediction and the actual re-entry time of the cubesat. Notably, 25 samples re-enter at exactly the same time as reported by Celestrak [17], and this date represents the mean of the PDF. The re-entry window spans ±15 hours, influenced by the uncertainty in Rosey's initial state vector.

Even though this result appears satisfactory, the validation cannot end here. The re-entry date is only an estimate, as data beyond August 31 at 6:00 a.m. is unavailable. For the validation to be acceptable, the orbital representation must also follow the same trends. To achieve this, a sample that re-enters around September 1st is selected as a baseline for comparison. The chosen sample re-enters on August 31 at 11:59:10 p.m. The aim of this analysis is for the chosen sample to reach an altitude of 200 km at the same time as the real orbit, which is on 31/08/2024 at 6:11 a.m. Figure 3.2 shows the result of the analysis.

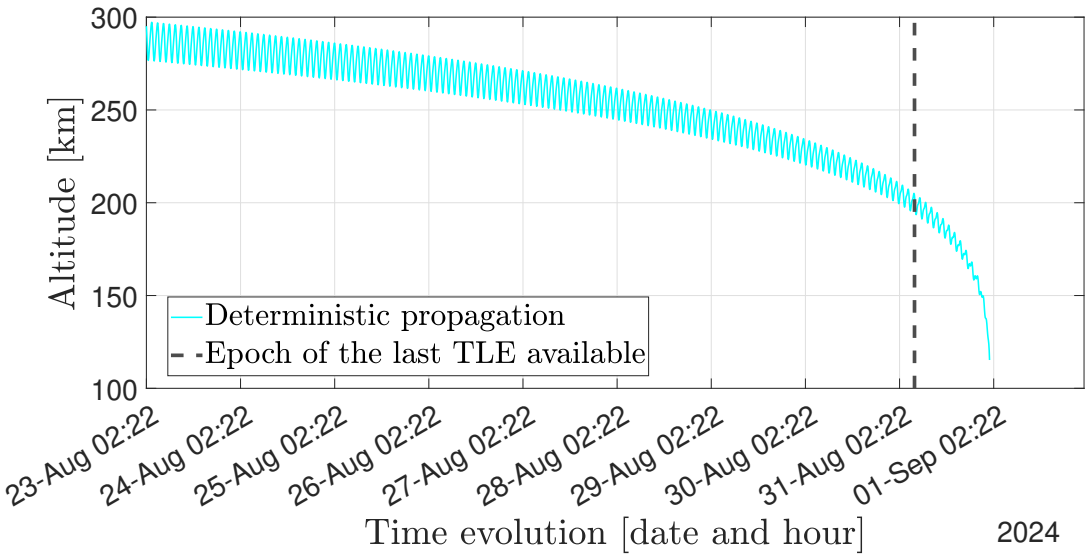


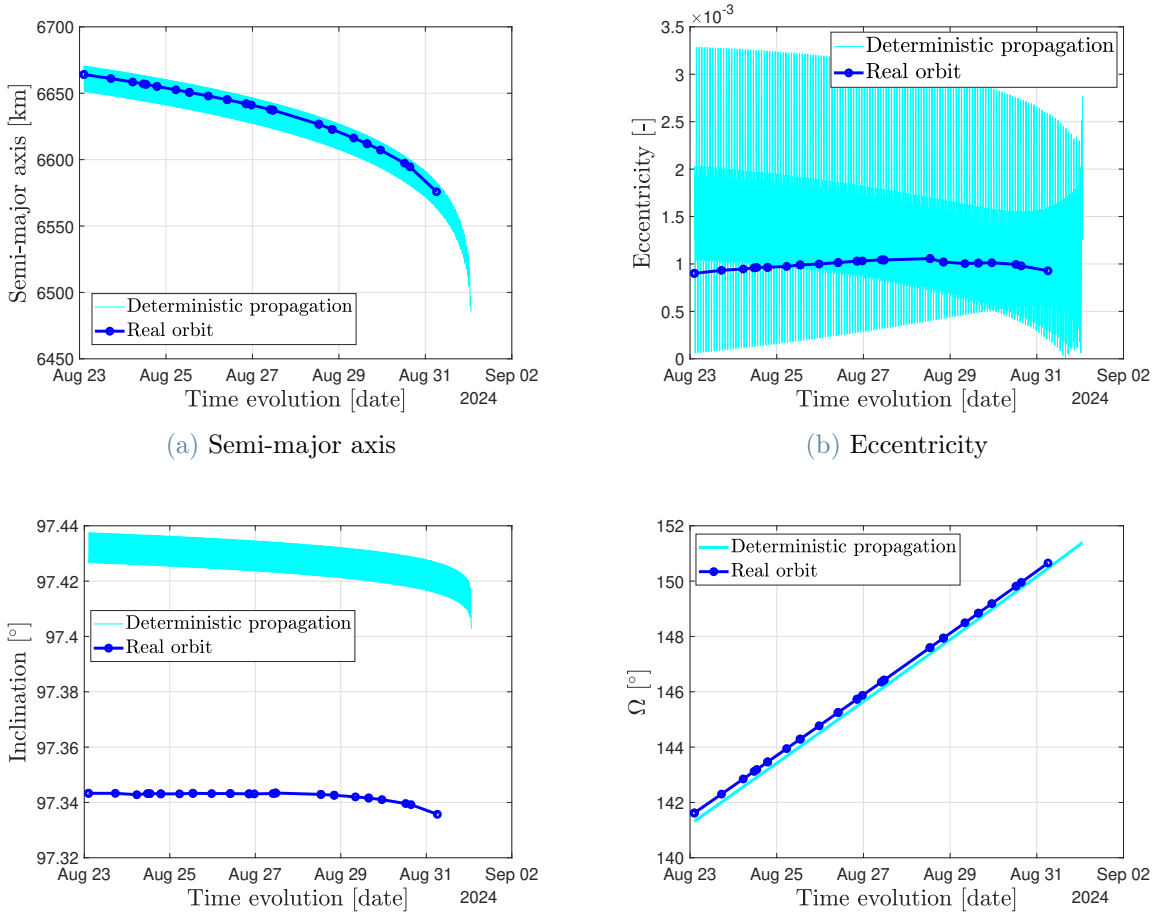
Figure 3.2: Altitude evolution of the selected sample, demonstrating its suitability for re-entry analysis.

It is immediately evident that the chosen sample is well-suited for this validation, as it re-enters very close to the hypothetical re-entry date indicated by Celestrak [17] and reaches the altitude of 200 km precisely at the moment of the last available TLE from the catalogue (dashed black line).

The final step to complete the validation is to examine the evolution of the orbit as propagated by the deterministic algorithm. This comparison verifies whether the trends in the Celestrak TLE dataset [17] align with the deterministic model and checks that the orbital behaviour resulting from perturbations matches the descriptions provided in section 2.1.1. This step is essential to ensure the algorithm's reliability, establishing a robust baseline for analysing the performance of the stochastic algorithm.

In Figure 3.3, the comparison of the semi-major axis, eccentricity, inclination, and RAAN is shown. There is no difference between the actual re-entry of Roseycubesat-1 and the one propagated using the deterministic algorithm. Minor differences do exist, but they are present from the start, indicating they are due to the MC analysis performed to propagate the uncertainty in Roseycubesat-1's initial state. It is important to note that this difference is minimal, with variations in both the inclination (Figure 3.3c) and Ω (Figure 3.3d) being on the order of a few hundredths of a degree. As a final remark, it is worth noting that the orbital parameters also follow the expected trends based on the theoretical effects of orbital perturbations on a satellite, such as the inclination remaining mostly constant while slowly decreasing, the semi-major axis diminishing, and Ω contin-

ually increasing. The eccentricity is already quite low, so the circularization effect due to drag is not noticeable.



(a) Semi-major axis

(b) Eccentricity

(c) Inclination

(d) Right Ascension of the ascending node

Figure 3.3: Keplerian elements evolution of the selected sample compared with the actual orbit evolution of Roseycubesat-1, extracted from the Celestrak TLE dataset [17].

The x-axis of Figure 3.3 represents the evolution of time by date. Each date shown corresponds to 00:00 UTC on the specified day. The validation was successful. The result demonstrates that the orbit is accurately represented and that the in-house propagator can effectively predict a satellite's re-entry. In conclusion, this in-house propagator is a good baseline for testing the performance of the novel stochastic propagator.

4 | Results

The stochastic propagator is proposed as an alternative to more detailed propagators, aiming to provide re-entry predictions for satellites that are similarly accurate but significantly faster. The stochastic propagator includes all forces acting on the re-entering object as described in section 3.2, but it only considers a 3DOF model. The uncertainty in the initial state of the re-entering object is propagated using MC. The uncertainty introduced by the model simplification is constant over time, and independent of the state. It is incorporated into the integration through SRK4 (refer to section 3.4).

To confirm the hypotheses and address the research question, the performance of the stochastic propagator must be thoroughly analysed. For this purpose, two approaches are employed to obtain a comprehensive understanding of the algorithm's behaviour:

1. Analysis of performance using the same dynamic model to observe how the stochastic propagator compensates for the absence of attitude dynamics modelling. In this case Roseycubesat-1 is employed as test case.

2. Comparison with a high-fidelity algorithm HPOP that includes all perturbative forces. In this case instead, GOCE is studied.

The objective of this analysis is clear: it aims to achieve re-entry windows from the compared algorithms that are closely aligned. Additionally, a nice-to-have is to obtain similar orbital representations, with a matching time-evolution trends in the Keplerian parameters. The expected outcome is a broader re-entry window resulting from the increased uncertainty in the stochastic algorithm. Nevertheless, this wider window will still offer valuable support for estimating the actual re-entry time. Additionally, orbital element trends are anticipated to be less smooth, with noisier and more oscillatory behaviour compared to deterministic propagation. It is also expected to achieve significantly lower computational time while using the same number of samples.

Instead, what is not considered in studying the performance of the stochastic algorithm is to maintain a similar state vector between results. Retrieving the exact same orbital evolution and achieving a perfect re-entry location is nearly impossible. This challenge

arises even with the most accurate high-fidelity propagator. In fact, prior to re-entry, a prediction campaign is conducted on the object. Various countries contribute to this effort by providing updated observations and telemetry data from the satellite. This is due to numerous uncertainties coming from man-made models, including atmospheric variations, solar activity predictions, and simplifications of the perturbations acting on the satellite.

This tool could be beneficial during the design phase. It can help to obtain an accurate re-entry window and provides predictions on orbital evolution, offering a rough estimate of where the object may ultimately fall.

4.1. 6DOF vs 3DOF - Roseycubesat-1

The first comparison involves evaluating the deterministic validated algorithm against the stochastic one, with Roseycubesat-1 as the primary focus. The initial conditions match those presented in Table 3.1. In this analysis, all potential values of σ are explored until the uncertainty becomes too high. Only a selection of these values is reported in this section, while the rest will be included in the appendix B.

The results are categorised into three main groups based on intensity level: predictions that are too late when σ is too low, accurate predictions, and early re-entries when σ is too high. They are organised in the thesis accordingly.

Table 4.1 summarises all the setups analysed in this chapter.

Roseycubesat-1 test setups

Test setup	Diffusion factor
1	0
2	$[0, 0, 0, 1, 1, 1] * 10^{-13}$
3	$[0, 0, 0, 1, 1, 1] * 10^{-6}$
4	$[0, 0, 0, 1, 1, 1] * 10^{-5}$
5	$[0, 0, 0, 1, 1, 1] * 2 * 10^{-5}$
6	$[0, 0, 0, 1, 1, 1] * 10^{-4}$
7	$[0, 0, 0, 1, 1, 1] * 10^{-2}$

Table 4.1: Configuration of test setups for the performance analysis of the stochastic propagator, in comparison with the deterministic algorithm.

Finally, all propagations use 1000 samples.

4.1.1. Setup 1 - Null diffusion factor

With this setup, the model reverts to the deterministic case. Observing the effect of the absence of the attitude dynamics modelling is important to start this performance analysis properly. The re-entry window span is anticipated to match due to the uncertainty in Roseycubesat-1's initial state. However, the re-entry window itself is expected to be completely inaccurate, causing the satellite to behave differently from its real behaviour. Figure 4.1 reports the re-entry windows resulting from both the algorithms.

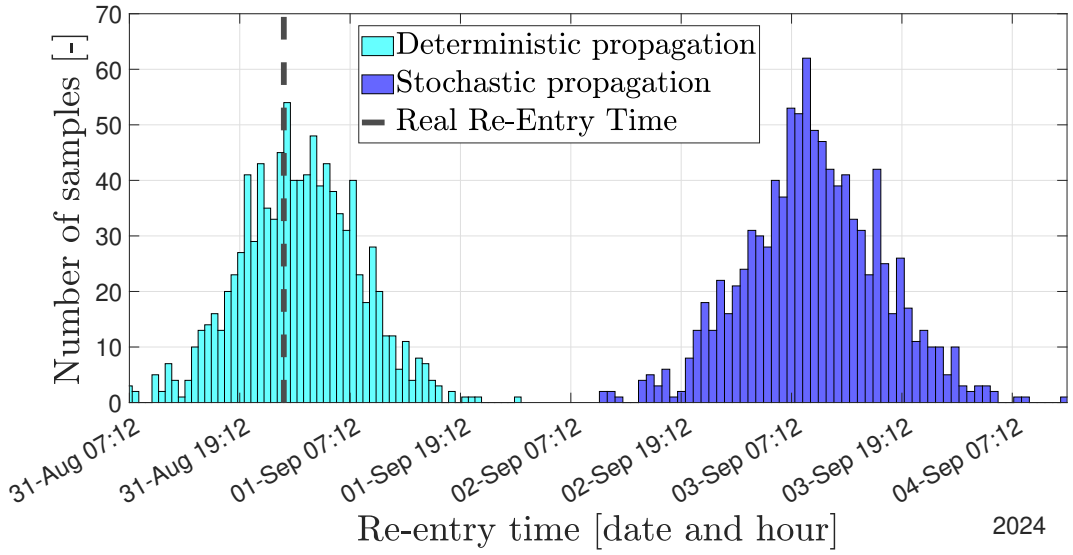


Figure 4.1: Comparison of re-entry windows between the real re-entry date [17] predicted by the deterministic algorithm and that predicted by the stochastic one, using the diffusion factor of Setup 1 in Table 4.4. This setup aims to evaluate the performance of the stochastic algorithm with a null σ against the deterministic benchmark.

The re-entry windows differ significantly due to the absence of attitude dynamics modelling, which substantially alters the re-entry evolution. Both PDFs show similarities, with distributions that resemble each other closely and span an identical range, as expected. The big difference lays on the mean value of the distributions. According to the stochastic propagation in blue, Roseycubesat-1 reaches re-entry altitude on September 3rd at around 9 a.m., which does not align with the actual re-entry date. This actual date is, however, well captured by the deterministic algorithm. This analysis confirms that using a 6DOF propagator instead of a 3DOF impacts the prediction. The objective remains to use the ideal σ value that accurately provides results without requiring attitude dynamics

implementation.

4.1.2. Setup 2 and 3 - Low diffusion factor

The stochastic algorithm requires a diffusion factor intensity that aligns with the simplifications applied to the model, besides the usual uncertainties, such as atmospheric conditions. If the level of uncertainty is set too low, its results are inaccurate. In this analysis, all values are explored to evaluate the performance of the stochastic approach.

Starting from zero, all potential values of σ are propagated. Until the correct diffusion factor for the present problem is identified, the results do not deviate significantly from those with a null diffusion factor (see section 4.1.1). Since the results are very similar, only these two cases are reported, while the rest is included in appendix B.

Figure 4.2 illustrates the re-entry window for the second setup, with a diffusion factor $G = [0, 0, 0, 1, 1, 1] * 10^{-13}$. It shows again the comparison between the deterministic and stochastic propagators.

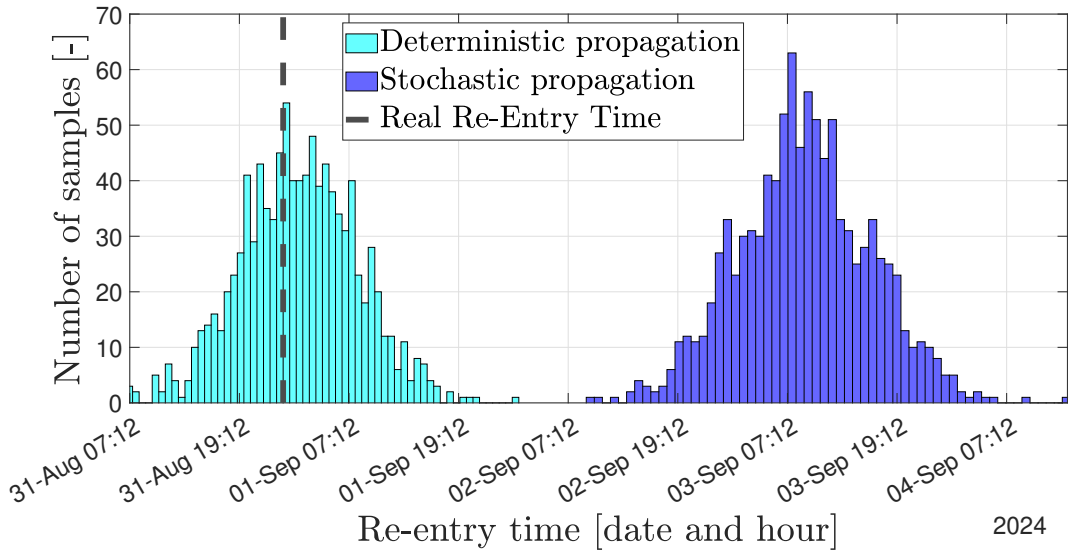


Figure 4.2: Comparison of re-entry windows between the real re-entry date [17] predicted by the deterministic algorithm and that predicted by the stochastic one, using the diffusion factor of Setup 2 in Table 4.4. This setup aims to evaluate the performance of the stochastic algorithm with a low σ against the deterministic benchmark.

This plot resembles Figure 4.1, as anticipated in the premise. The re-entry prediction remains centred around the morning of September 3rd, with a similar curve spread.

With a higher diffusion factor, though still insufficient for this problem type, the results remain unaltered from previous setups. Figure 4.3 shows no evident changes from the prior outcomes using $G = [0, 0, 0, 1, 1, 1] * 10^{-6}$.

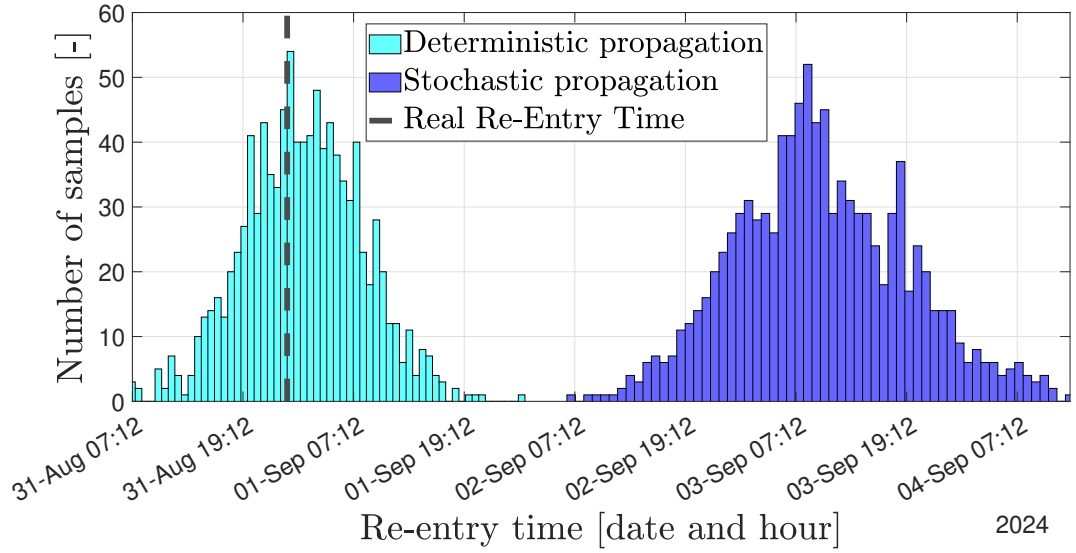
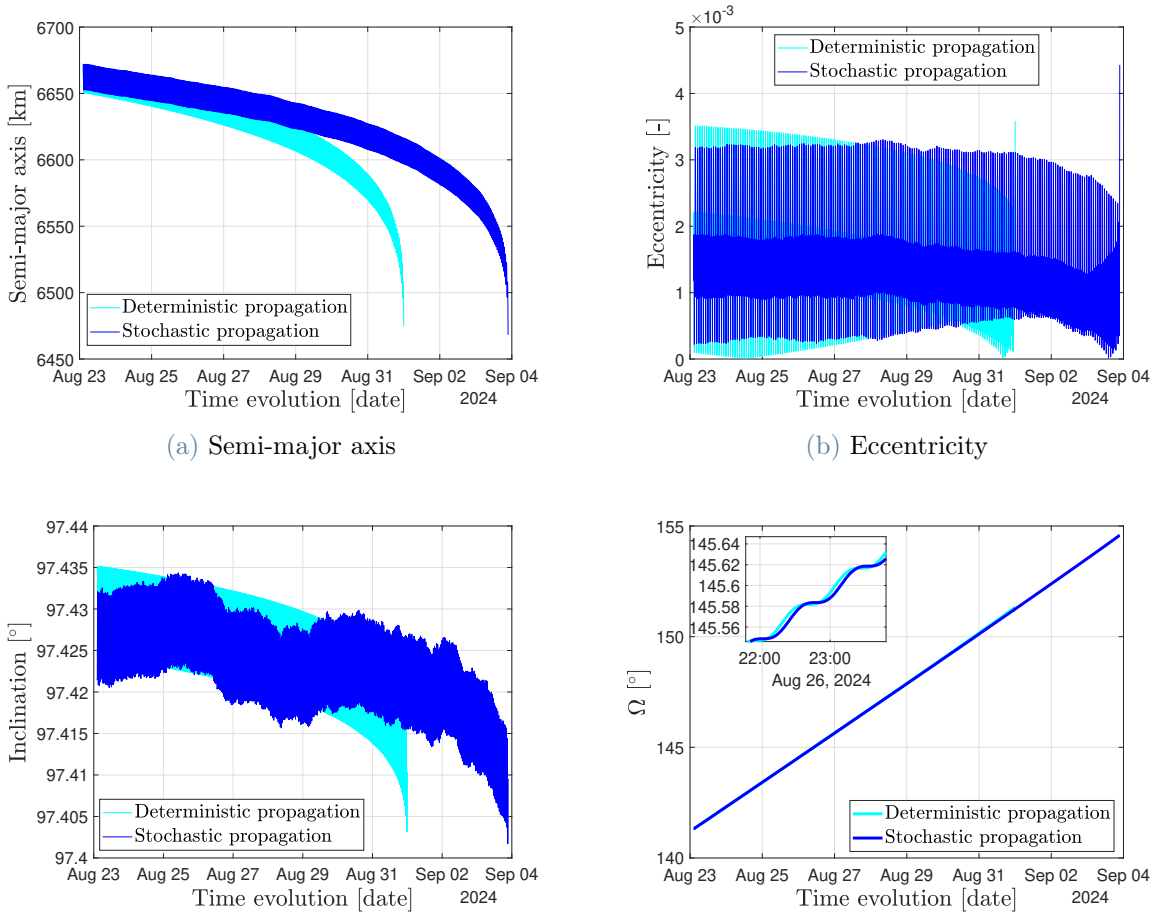


Figure 4.3: Comparison of re-entry windows between the real re-entry date [17] predicted by the deterministic algorithm and that predicted by the stochastic one, using the diffusion factor of Setup 3 in Table 4.4. This setup aims to evaluate the performance of the stochastic algorithm with a low σ against the deterministic benchmark.

However, this value of G is relatively high, meaning it does impact the propagation, introducing some noise into the Keplerian elements. The trends of these elements are presented in Figure 4.4, comparing two samples. The deterministic sample shown is the one closest to actual behaviour, also used to validate the deterministic model in section 3.5. Oscillations appear across all components except for RAAN (Figure 4.4d). The SMA (Figure 4.4a) is slightly affected, while the inclination (Figure 4.4c) and eccentricity (Figure 4.4b) show the greatest influence.



(a) Semi-major axis

(b) Eccentricity

(c) Inclination

(d) Right Ascension of the ascending node

Figure 4.4: Keplerian elements evolution of the selected sample compared with the deterministic orbit propagation of Roseycubesat-1

The x-axis of Figure 4.4 represents the evolution of time by date. Each date shown corresponds to 00:00 UTC on the specified day. To summarise, these two setups confirm that the stochastic algorithm cannot operate with a σ value that is inappropriate for the model simplifications, highlighting a limitation of this approach.

4.1.3. Setup 4 and 5 - Well-tuned diffusion factor

With a diffusion factor that is too low, the stochastic propagator fails to perform effectively. The crucial point, however, is to identify a value that aligns with the model simplifications to enable accurate predictions. With a σ with an order of magnitude of 10^{-5}, the re-entry prediction of the stochastic propagator is quite closer to the real one. Starting with $G = [0, 0, 0, 1, 1, 1] * 10^{-5}$, Figure 4.5 illustrates the re-entry windows for both algorithms. The two curves show an initial overlap, and some samples even reach

as far as September 1st. However, this configuration does not yet yield the most accurate prediction.

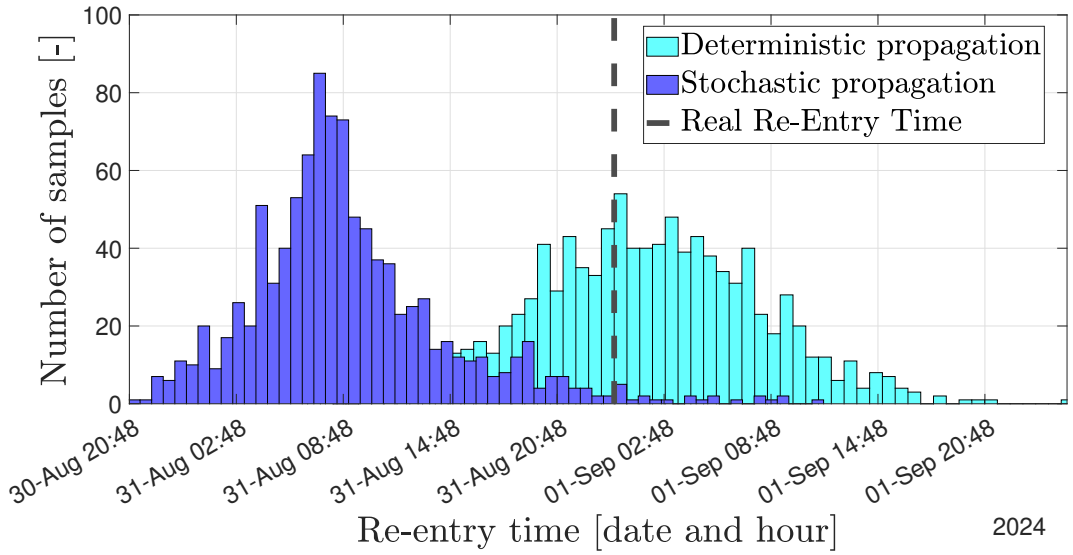


Figure 4.5: Comparison of re-entry windows between the real re-entry date [17] predicted by the deterministic algorithm and that predicted by the stochastic one, using the diffusion factor of Setup 4 in Table 4.4. This setup aims to evaluate the performance of the stochastic algorithm with a proper tuned σ against the deterministic benchmark.

The mean re-entry occurs around August 31st at 7 am, roughly 17 hours before the actual event. The re-entry window is also somewhat broader and less concentrated than the deterministic model. When the diffusion factor decreases to an order of magnitude of 10^{-6}, however, the results deviate significantly. This setup highlights that an accurate prediction is possible, but careful tuning of the diffusion factor is essential, as an inappropriate value can lead to substantial inaccuracies.

The best result is obtained when the diffusion factor is set at $G = [0, 0, 0, 1, 1, 1] * 2 * 10^{-5}$. The result is in Figure 4.6.

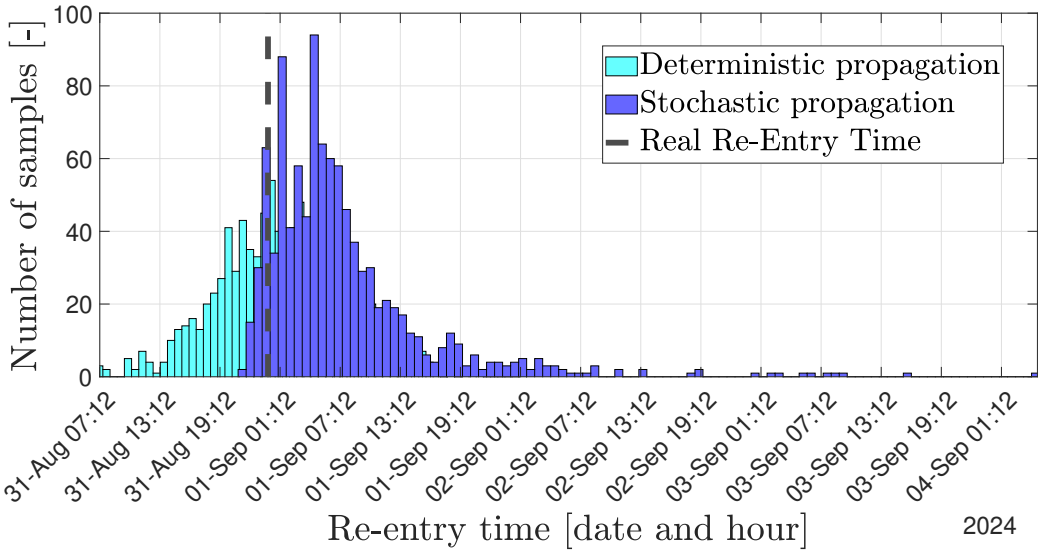
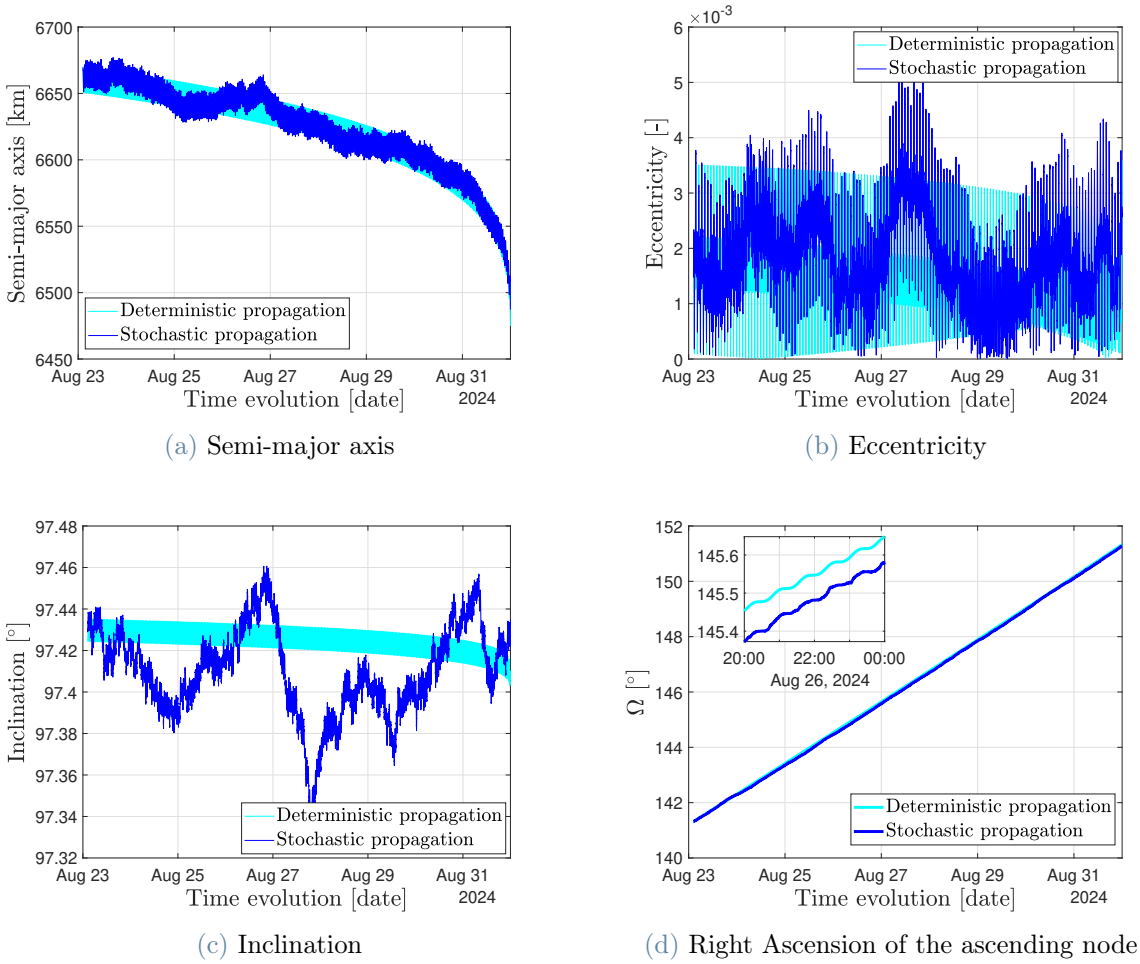


Figure 4.6: Comparison of re-entry windows between the real re-entry date [17] predicted by the deterministic algorithm and that predicted by the stochastic one, using the diffusion factor of Setup 5 in Table 4.4. This setup aims to evaluate the performance of the stochastic algorithm with a proper tuned σ against the deterministic benchmark.

Although the distribution shifts slightly to the right, 50 samples align precisely with the desired re-entry time.

What poses a greater challenge with high uncertainty is the representation of the orbit. This aspect becomes increasingly complex as noise impacts the behaviour of the parameters. Greater noise levels result in progressively more pronounced distortions. Figure 4.7 reports the trends of the main Keplerian elements. In the previous case, only two elements report significant variation, but now all parameters show fluctuations. The RAAN remains the least affected, as shown in Figure 4.7d, although even here minor oscillations are present. Both eccentricity (Figure 4.7b) and inclination (Figure 4.7c) display up-and-down variations, which, although appearing very high, reflect only minor numerical shifts. The SMA in Figure 4.7a, however, oscillates by several kilometres.

Despite these fluctuations, the selected sample aligns closely with the deterministic algorithm's results.



(a) Semi-major axis

(b) Eccentricity

(c) Inclination

(d) Right Ascension of the ascending node

Figure 4.7: Keplerian elements evolution of the selected sample compared with the deterministic orbit propagation of Roseycubesat-1.

The x-axis of Figure 4.7 represents the evolution of time by date. Each date shown corresponds to 00:00 UTC on the specified day. In summary, accurate results can be achieved with an appropriately tuned diffusion factor for the problem. However, attention must be given to orbit representation, which may prove complex.

4.1.4. Setup 6 and 7 - High diffusion factor

Now, the final aspect to address is the scenario when the diffusion factor is excessively high for the problem. When uncertainty becomes too great, the propagation can diverge significantly from the real case and initial conditions. This divergence compromises all predictions, leading to potentially random solutions. If any solution happens to align with the accurate outcome, it is merely coincidental and should not be considered as the absolute truth. To demonstrate this, values exceeding those that represent the best fit

are analysed. The expectation is to observe re-entries occurring randomly, without any pattern, resulting in orbits that lack coherence. The fluctuations present in the Keplerian elements should be far from what is feasible in reality. Figure 4.8 shows the result of setup 6, which does not differ significantly from the optimal diffusion factor setup. Here, the diffusion factor is set to $G = [0, 0, 0, 1, 1, 1] \times 10^{-4}$.

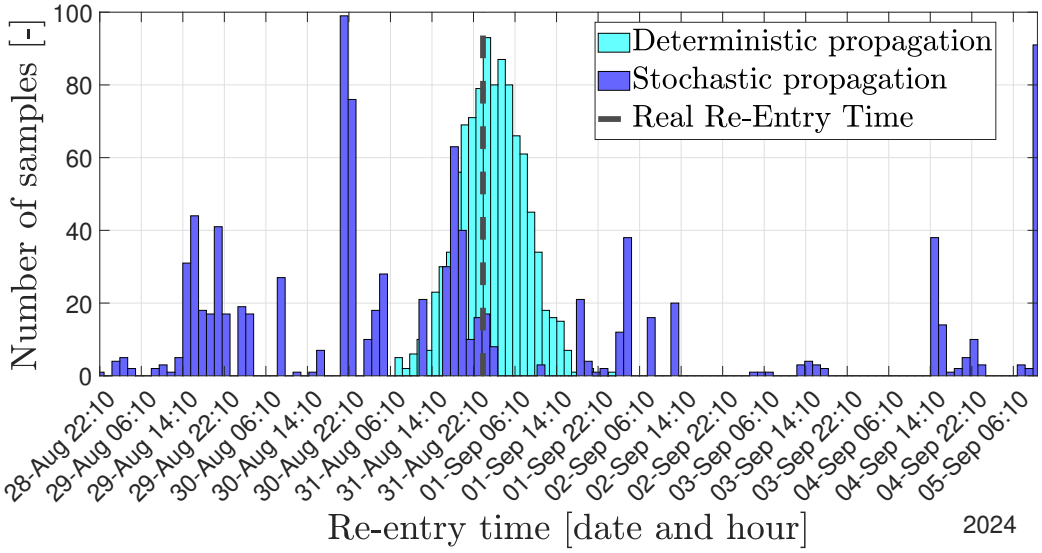
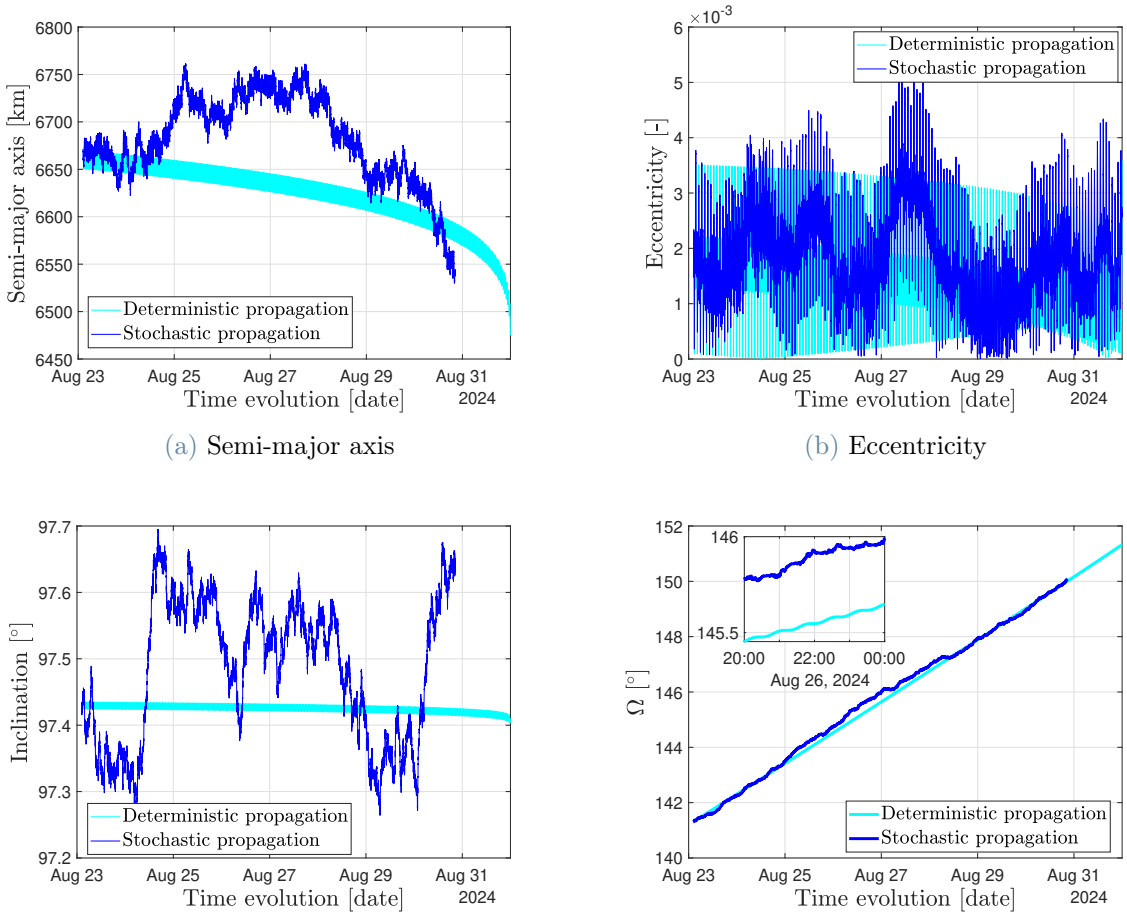


Figure 4.8: Comparison of re-entry windows between the real re-entry date [17] predicted by the deterministic algorithm and that predicted by the stochastic one, using the diffusion factor of Setup 6 in Table 4.4. This setup aims to evaluate the performance of the stochastic algorithm with a high σ against the deterministic benchmark.

The results align with expectations. Although the diffusion factor is not excessively high, most samples re-enter close to the actual date, overlapping with the deterministic prediction window. However, as anticipated, this overlap is coincidental, and the re-entry window does not represent a physically realistic scenario. The span of sample re-entries extends over more than a week, without an identifiable mean that could indicate a plausible re-entry date.

The resulting orbit from the propagation is in Figure 4.9.



(a) Semi-major axis

(b) Eccentricity

(c) Inclination

(d) Right Ascension of the ascending node

Figure 4.9: Keplerian elements evolution of the selected sample compared with the deterministic orbit propagation of Roseycubesat-1.

The x-axis of Figure 4.9 represents the evolution of time by date. Each date shown corresponds to 00:00 UTC on the specified day. None of the Keplerian elements are accurately represented. The time evolution of the orbit, affected by perturbations on the satellite, lacks physical meaning. In Figure 4.9a, the SMA oscillates by several kilometres without any apparent logic, and the inclination, shown in Figure 4.9c, exhibits similarly behaviour.

Although previous diffusion factor values are affected by noise, they still display trends that respect astrodynamics principles. Now, this coherence is lost. As a final setup, a very high diffusion factor with an order of magnitude of 10^{-2} is chosen. All samples re-enter quickly due to the high uncertainty, which causes each sample to re-enter almost immediately after the start of the propagation, as shown in Figure 4.10. Above this value, any diffusion factor leads to failure.

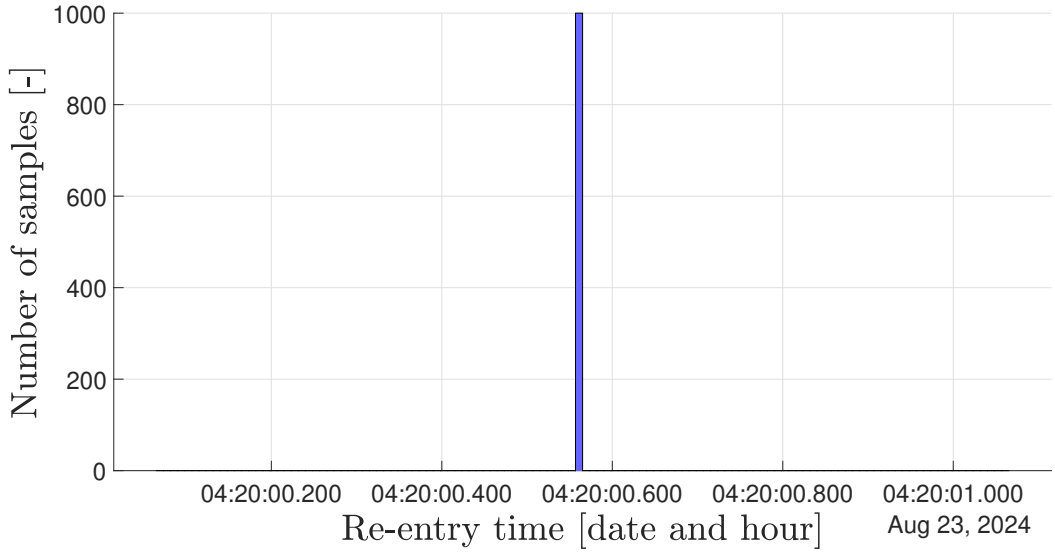


Figure 4.10: Comparison of re-entry windows between the real re-entry date [17] predicted by the deterministic algorithm and that predicted by the stochastic one, using the diffusion factor of Setup 7 in Table 4.4. This setup aims to evaluate the performance of the stochastic algorithm with a high σ against the deterministic benchmark.

4.1.5. Summary

Overall, the stochastic algorithm applied to re-entry analysis is a valuable tool. It effectively compensates for the non-modelled attitude dynamics when a 3DOF propagator is used instead of a 6DOF one. This approach has been tested through the Roseycubesat-1 re-entry analysis, where the cubesat dynamics are propagated using an in-house deterministic propagator, which serves as the baseline for the performance study of the stochastic method. The final days of the orbit were examined up to the re-entry that occurred on September 1st.

Dealing with stochastic algorithms can be challenging. It is crucial to use the correct diffusion factor for accurate results. Indeed, having a σ that is too low means that the stochastic component has no influence on the propagation, leaving it entirely at the mercy of the simplified model. This results in predictions based solely on the 3DOF algorithm. Conversely, a σ that is too high leads to a completely random or incorrect evolution, which cannot be relied upon.

When the optimal diffusion factor is achieved, the prediction is both accurate and reliable. In this case, the optimal value is around an order of magnitude of 10^{-5}, as it compensates for the satellite not always being correctly exposed to drag due to the lack of implemented

attitude dynamics. In general, the effect of drag on the acceleration of the S/C is exponential [8], but in this context, it remains valid since it does not need to compensate for non-modelled drag, only an average value for that altitude range. However, it is important to note the noise in the orbital parameters, which complicates accurate predictions of the satellite's orbital evolution. The algorithm fulfils its purpose, but it has an additional strength: computational efficiency.

The deterministic propagator, which incorporates the same perturbations as the stochastic model but includes additional attitude dynamics, has a computational time of approximately 2.8 hours. Table 4.2 presents all the computational times for the setups propagated with the stochastic method.

Computational time analysis for the stochastic propagator

Test setup	Computational time
1	1.81 h
2	1.84 h
3	1.89 h
4	1.71 h
5	1.76 h
6	1.73 h
7	1.13 h

Table 4.2: Computational times for the stochastic propagation setups, expressed in hours.

The propagations are executed on a PC equipped with a 13th-generation Intel i5-1335U processor. This is a significant strength of the model. By adding only a small amount of code, the computational time rises by more than an hour. This increase occurs because even minor operations must be repeated across 1000 samples, raising the computational load. This result highlights the value of using this algorithm for re-entry predictions.

4.2. Dynamical model accuracy - GOCE

The purpose of this analysis is to explore the use of a stochastic propagator for predicting satellite re-entry using a much simpler dynamical model. Utilizing such a propagator is expected to accelerate the prediction process. By introducing uncertainties at each propagation step and processing a large number of samples, it may be possible to achieve

results comparable to those obtained from a more accurate, high-fidelity propagation model. The goal is to estimate the re-entry date and time of GOCE, aiming to achieve a re-entry window that coincides in various setups with the theoretical prediction calculated using the high-fidelity HPOP propagator and with the recorded real re-entry time. The initial conditions considered are those in Table 4.3. The analysis window spans 21 days.

GOCE Specifications

Parameter	Value
Initial time of the propagation [10]	21/10/2013 at 01:05:56 UTC
Initial state vector in ECI [10]	[823.89, -6552.73, 3.48, -1.36, -0.16, 7.70] km and $\frac{\text{km}}{\text{s}}$
Estimated time of re-entry [34]	11/11/2013 at 00:23 UTC
Covariance matrix in ECI [10]	$P_0 = \begin{bmatrix} 6.35 & 1.58 & -0.81 & 0 & 0 & 0 \\ 1.58 & 19.06 & -0.44 & 0 & 0 & 0 \\ -0.81 & -0.44 & 1.57 & 0 & 0 & 0 \\ 0 & 0 & 0 & 56.12 & -4.01 & 27.94 \\ 0 & 0 & 0 & -4.01 & 17.56 & 2.85 \\ 0 & 0 & 0 & 27.94 & 2.85 & 223.41 \end{bmatrix} \times 10^{-7}$ km^2 and $\frac{\text{km}^2}{\text{s}^2}$
Inertia vector [14]	[173, 2750.50, 2723.20] kgm^2
Inverse of B	0.03
Re-entry altitude	115 km
Atmospheric model	NRLMSISE-00

Table 4.3: GOCE specifications for the propagation analysis.

It is important to note that the date reported in the table is the true re-entry date. The actual re-entry of GOCE is influenced by attitude dynamics control until its final hours in orbit. For this reason, it is not possible to obtain results that fully align with what actually occurred. The inverse of the ballistic coefficient has been estimated by interpolating various sources from the literature, providing an approximation rather than an exact value. A mass of 1000 kg [14] and an area of 10 m^2 [34] with $C_D = 3$ [34] were considered. For the HPOP, additional forces accounted for, beyond those in the stochastic propagation model, include third-body perturbation from the Sun and SRP.

Regarding the number of samples, 500 are propagated for the stochastic model, while 200 samples are used for HPOP. This choice is based on the computational time required for the high-fidelity algorithm, as propagating more samples would have required several days. Given the high accuracy of the algorithm, increasing the sample size would have minimal impact on the analysis. However, to allow a meaningful comparison, a visible re-entry window is necessary, making 200 samples optimal for this case.

The various setups analysed are in total 6 and are presented in Table 4.4.

GOCE test setups

Setup	Diffusion factor	Atmospheric model
1	$[0_{3x3}, 1_{3x3}] \times 10^{-8}$	nrlmsise-00
2	$[0_{3x3}, 1_{3x3}] \times 10^{-6}$	nrlmsise-00
3	$[0_{3x3}, 1_{3x3}] \times 10^{-5}$	nrlmsise-00
4	$[0_{3x3}, 1_{3x3}] \times 10^{-10}$	nrlmsise-00
5	0	nrlmsise-00
6	$[0_{3x3}, 1_{3x3}] \times 10^{-8}$	JB

Table 4.4: Setups developed to evaluate the performance of the stochastic propagator, in comparison with the deterministic algorithm.

4.2.1. Setup 1

As previously mentioned, the stochastic algorithm requires a diffusion factor intensity that corresponds to the simplifications applied in the model. With the correct level of uncertainty, accurate re-entry predictions are achievable. The performance analysis begins with the best-fitting diffusion factor, set as $G = [0, 0, 0, 1, 1, 1] * 10^{-8}$. Figure 4.11 displays the re-entry windows generated by both the HPOP and stochastic propagations.

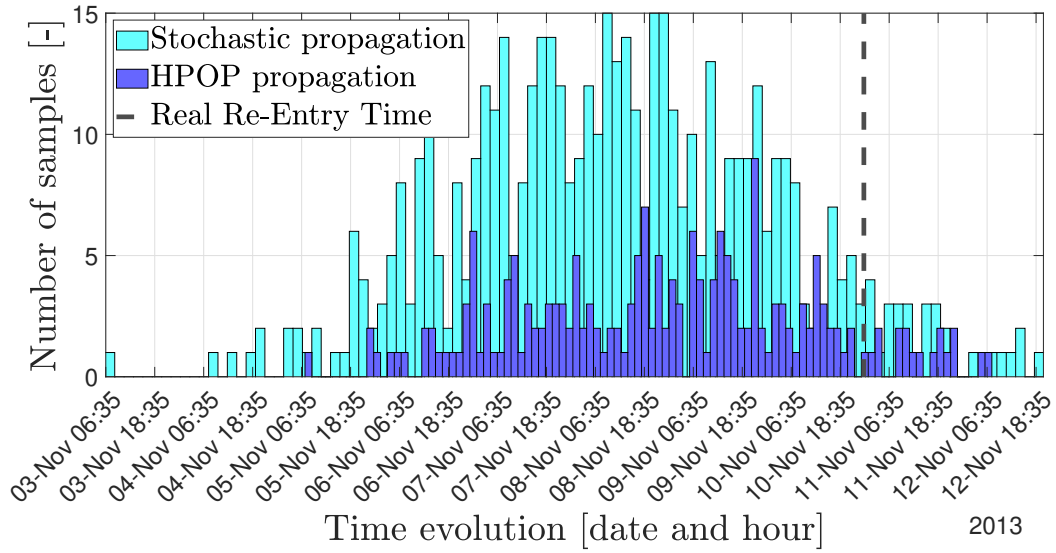


Figure 4.11: Comparison of re-entry windows between the real re-entry date predicted by the deterministic algorithm and that predicted by the stochastic one, using the diffusion factor of Setup 1 in Table 4.4. This setup aims to evaluate the performance of the stochastic algorithm with a properly-tuned σ against the deterministic benchmark.

The results are once again promising, with the two re-entry windows overlapping and displaying similar mean values. As expected, the stochastic window spans wider compared to the HPOP one. This is primarily due to the higher number of samples used in the stochastic analysis and the greater level of uncertainty, which results in a broader window.

4.2.2. Setup 2 and 3 - High diffusion factor

An ideal level of uncertainty has been identified—one that is neither too high nor too low. If the uncertainty is too high, as will be shown, the results will not be meaningful at all. If it is too low, the outcomes will not differ significantly from those of the high-fidelity propagator, essentially offering no added value. Too high, because in that case the curve will no longer resemble a Gaussian distribution, making it impossible to achieve an accurate prediction. The error would be so large that it could represent entirely incorrect results, deviating significantly from the actual re-entry scenario and undermining the purpose of the prediction model. For this reason, a setup involving higher uncertainty has been also analysed. For this case G equals to $\sigma = [0, 0, 0, 1, 1, 1] * 10^{-6}$. As can be clearly seen in Figure 4.12, the behaviour of the propagator starts to become quite unusual.

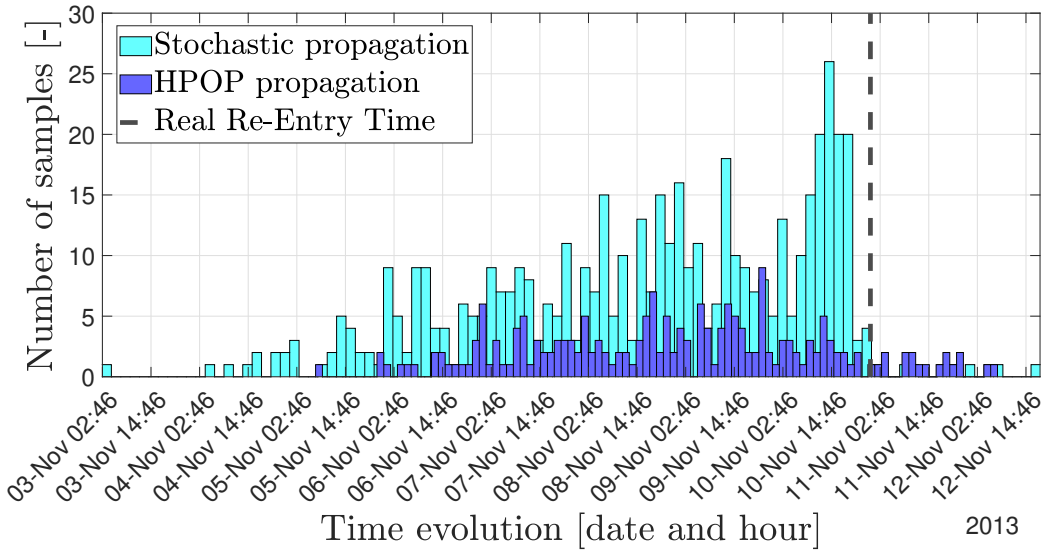


Figure 4.12: Comparison of re-entry windows between the real re-entry date predicted by the deterministic algorithm and that predicted by the stochastic one, using the diffusion factor of Setup 2 in Table 4.4. This setup aims to evaluate the performance of the stochastic algorithm with a high σ against the deterministic benchmark.

The mean prediction for re-entry is shifted by 2 days compared to the forecast made in section 4.2.1, with most of the samples re-entering after 20 days. What's particularly strange, however, is that the re-entry window remains the same, between 16 and 20 days from the initial propagation time. This highlights that the fine-tuning mentioned earlier has been entirely missed, and the propagation is now diverging significantly from the more realistic case. This underscores a limitation of the stochastic model.

It might appear from Figure 4.3 that this σ captures the realistic behaviour of GOCE, almost compensating for the stochastic propagator's 3DOF model. However, this apparent agreement is somewhat coincidental. This aspect needs further investigation, but due to the high propagation time span, it is difficult to fully appreciate its capability to tackle the issue of the lack of attitude control implementation. In this analysis, the baseline case is represented by the integration performed by HPOP, meaning that it cannot be concluded that the stochastic approach is performing well in this setup.

To support the thesis that increasing the uncertainty in the model further degrades the results of the stochastic propagator, the level of uncertainty has been increased once again. It is set at $\sigma = [0, 0, 0, 1, 1, 1] * 10^{-5}$.

Theoretically, with a higher level of uncertainty, a reliable prediction cannot be made. The objective here is to demonstrate how excessive uncertainty leads to a complete divergence

from the expected behaviour, further emphasizing the importance of proper fine-tuning to achieve reliable predictions. Since the uncertainty is higher, the number of samples has been doubled to 1000 to achieve a behaviour as closely aligned with reality as possible. The result is reported in Figure 4.13.

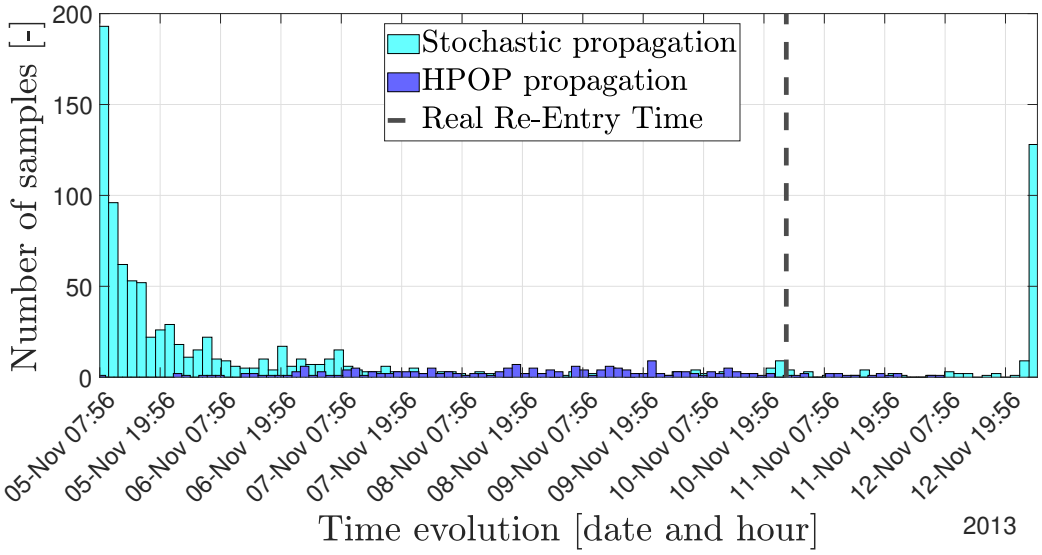


Figure 4.13: Comparison of re-entry windows between the real re-entry date predicted by the deterministic algorithm and that predicted by the stochastic one, using the diffusion factor of Setup 3 in Table 4.4. This setup aims to evaluate the performance of the stochastic algorithm with a high σ against the deterministic benchmark.

The distribution of samples re-entering the atmosphere no longer resembles a Gaussian curve. The somewhat irregular shape is due to the high level of uncertainty, which causes the orbit to diverge significantly from the actual case. This result aligns well with expectations. The majority of samples re-enter between November 5^{th}—where the peak occurs with 151 samples—and November 6^{th}. There is also a second peak, consisting of 124 samples that do not re-enter at all. This highlights that increasing the uncertainty within the model leads to a result that significantly deviates from reality, thereby completely altering the prediction. Thus, a fine-tuning process is necessary before utilizing the propagator.

4.2.3. Setup 4 and 5 - Low diffusion factor

To gain a comprehensive understanding of how the propagator behaves under different levels of uncertainty, the goal is to study the effects of reducing the uncertainty. The diffusion factor is now $\sigma = [0, 0, 0, 1, 1, 1] * 10^{-10}$. The result is reported in Figure 4.14.

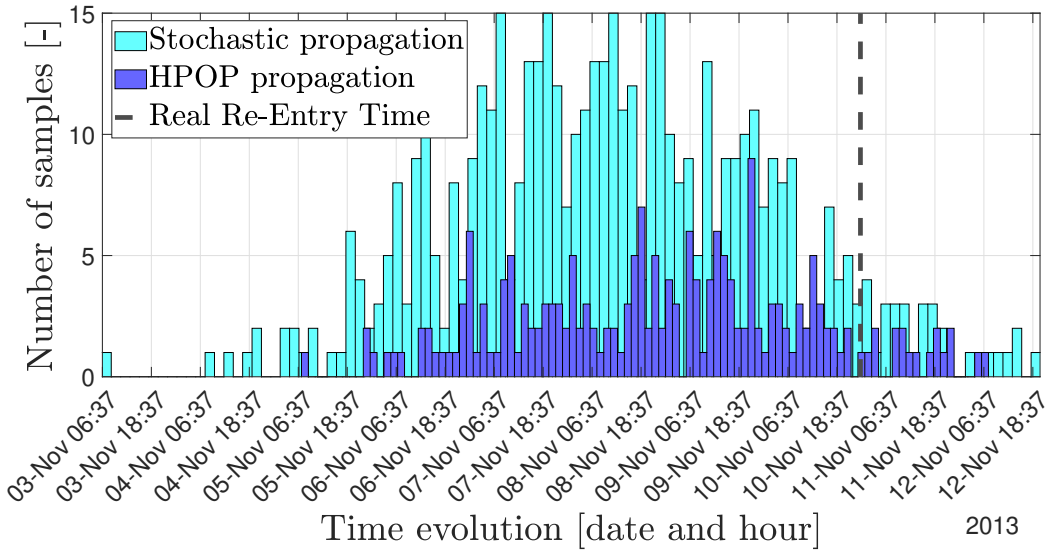


Figure 4.14: Comparison of re-entry windows between the real re-entry date predicted by the deterministic algorithm and that predicted by the stochastic one, using the diffusion factor of Setup 4 in Table 4.4. This setup aims to evaluate the performance of the stochastic algorithm with a high σ against the deterministic benchmark.

The trend does not change much from the second setup result. This is because, below a certain threshold that represents the optimal value for the scenario analysed, the performance remains similar and approaches that of the high-fidelity propagation. In fact, most of the distribution is grouped around the range from November 6^{th} and November 10^{th}. This aligns very well with the results found using HPOP, which range from 18 to 20 days.

Another interesting aspect to study is how the algorithm behaves with zero uncertainty. Figure 4.15 reports the re-entry windows of both the propagators when the diffusion factor is set to zero.

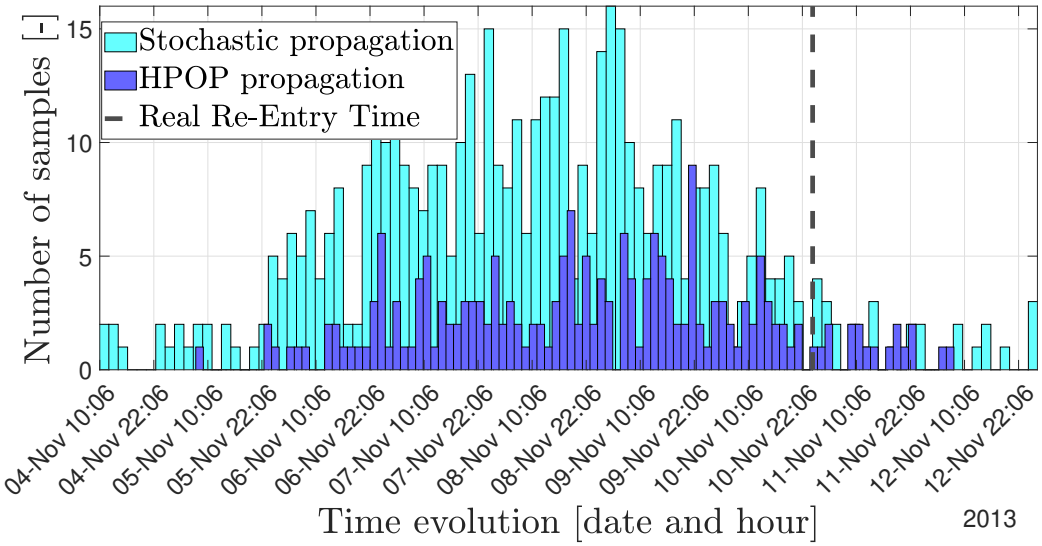


Figure 4.15: Comparison of re-entry windows between the real re-entry date predicted by the deterministic algorithm and that predicted by the stochastic one, using the diffusion factor of Setup 5 in Table 4.4. This setup aims to evaluate the performance of the stochastic algorithm with a high σ against the deterministic benchmark.

This result is quite critical. A narrower window compared to previous cases is expected. The issue lies in the fact that the uncertainty comes from two sources: one due to the initial conditions and the other from the model simplification. Using GPS data, the measurements of the satellite's state vector are highly precise. The issue arises when we propagate these conditions for 20 days. Over time, the uncertainty grows significantly, eventually overshadowing the uncertainty introduced by the model. A shorter time step would have better highlighted the latter. Further analysis would certainly help confirm this interpretation.

4.2.4. Setup 6

For the final setup, the algorithm's behaviour using the NRLMSISE-00 atmospheric model is studied with a setup using a different model. The atmospheric model in HPOP has been switched to the JB model, as outlined in section 2.3. All other settings remain consistent and the uncertainty is set to the best fit for this problem, so to $G = [0, 0, 0, 1, 1, 1] * 10^{-8}$. This case may reveal a potential limitation of the model and could provide insight into its ability to compensate when an alternative atmospheric model is employed. In this case, HPOP propagates only 25 samples and it is due to computational time reasons. It takes two days only to run 200 samples.

The re-entry windows for both propagators are displayed in Figure 4.16.

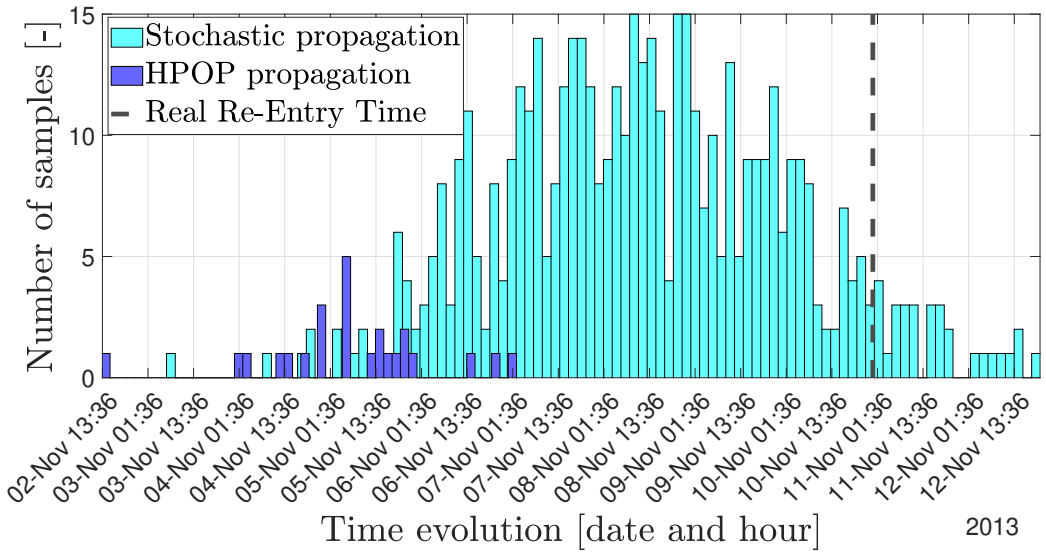


Figure 4.16: Comparison of re-entry windows between the real re-entry date predicted by the deterministic algorithm and that predicted by the stochastic one, using two different atmospheric models. This setup aims to evaluate the performance of the stochastic algorithm against the deterministic benchmark.

Using two entirely different atmospheric models leads to distinct orbital trajectories and developments, evident in the non-overlapping re-entry windows. The HPOP window, in particular, is shifted 4 days earlier than the other. This outcome highlights a further limitation of the model: it does not yield comparable results when alternative atmospheric models are applied. This is currently a weakness, but a more appropriately tuned σ could potentially account for differences in the atmospheric model as well. For example, implementing one that varies with the state.

4.2.5. Summary

This tool once again proves effective for re-entry predictions. In this test, it was evaluated for its accuracy relative to a more comprehensive model than the one applied in the stochastic analysis. Both algorithms operate as 3DOF models, allowing the impact of simplifying the dynamics model to become clearer. Although the results for this case are less promising than those for Roseycubesat-1, they highlight certain limitations of this approach and suggest areas for improvement.

First, with the appropriate diffusion factor, a reliable prediction can be achieved. In this

case, it is around 10^{-8}, as it compensates for the absence of SRP and the Sun's gravity effects [8].

As seen previously, an excessively high diffusion factor results in an unrealistic orbital evolution, leading to predictions that are physically implausible.

This second analysis highlights a limitation of the method. Specifically, with a low diffusion factor, predictions remain unaffected. This result stems from the extended propagation period—20 days in this instance—which significantly amplifies the uncertainty in GOCE's initial state, even with GPS-derived data. Consequently, this uncertainty overshadows the PDF, reducing the visible influence of the stochastic algorithm. An analysis based on a position closer to the re-entry time would be valuable for observing the stochastic uncertainty more effectively.

However, computational efficiency is a major strength of this method. HPOP requires nearly 48 hours of run time, whereas the computational time for the stochastic approach, shown in Table 4.5, is significantly lower. The difference is substantial, making this method far more practical, particularly for analyses conducted close to the re-entry period.

Computational time analysis for the stochastic propagator

Test setup	Computational time
1	2.21 h
2	2.29 h
3	2.28 h
4	2.25 h
5	2.26 h
6	2.21 h

Table 4.5: Computational times for the stochastic propagation setups, expressed in hours.

5 | Conclusion and Future work

In an era where the number of orbiting objects is increasing exponentially and mega-constellations are becoming more prevalent, a tool that offers fast and accurate predictions is essential.

This thesis presents the development and validation of a stochastic algorithm for the propagation of re-entering objects. This field is complex due to numerous uncertainties, including atmospheric variability, gravitational perturbations, and aerodynamic effects.

While deterministic propagators are designed to predict re-entries accurately, they often require extensive computational time. Even the most comprehensive propagators cannot account for all the uncertainties inherent in the problem.

The application of stochastic propagation to this type of problem addresses the issue of high computational demands while providing accurate and reliable predictions. This approach allows for the consideration of uncertainties arising from various factors. Such a tool is promising for facilitating mission designs that align with current space sustainability practices, ensuring safety for the future of space exploration.

The aim was to demonstrate that with a simpler model, applying a process noise term that accounts for the expected uncertainty can compensate for the model's lack of accuracy, resulting in outcomes comparable to those produced by a more comprehensive propagator. To assess the performance of the stochastic propagator, two real re-entries have been studied: Roseycubesat-1 and GOCE.

Roseycubesat-1 has been used to investigate how the stochastic algorithm behaves when the attitude dynamics is simplified. In this case, the deterministic and stochastic propagators share the same dynamic model; however, the stochastic one is a 3DOF model, meaning that the satellite's attitude dynamics is not explicitly implemented, whereas the deterministic model is 6DOF. The deterministic propagator was developed in-house and validated within this thesis.

GOCE re-entry, on the other hand, has been used to examine the effects when the deterministic algorithm's dynamic model is much more comprehensive than the stochastic

one. In this case, HPOP is used for the deterministic part. Both algorithms are 3DOF to better highlight the influence of the dynamic model.

Both analyses yield similar outcomes. When the diffusion factor is finely tuned to suit the specific problem, the results are promising: the re-entry windows and orbital evolutions closely align, with only minor differences, such as a broader window span or slight noise in the propagation. In this thesis, the diffusion factor is used to account for the acceleration that is not captured due to the simplifications applied to the model. However, with an excessively high diffusion factor, the results lose reliability, leading to unrealistic and inaccurate predictions. The re-entry windows become unreasonably wide, and the orbital evolution shows oscillations that are far from realistic.

On the other hand, using a diffusion factor that is too low results in a propagation almost identical to the inaccurately simplified dynamic model. These predictions lack accuracy, potentially providing a misleading information. Thus, a limitation of the method lies in the need for a well-informed estimation of the model's simplifications relative to reality. Additionally, the propagation period cannot be excessively long (20 days in this case), as the uncertainty in the initial position would overshadow the model-based uncertainty.

The greatest strength of this model is its ability to maintain accuracy while drastically reducing computational time. Even with a comparable dynamic model, running the simulation stochastically cuts down the computational time by several hours, making it a highly efficient alternative.

One outcome of this thesis is a paper published for the 75^{th} IAC 2024. It focuses exclusively on the analysis conducted with GOCE, while covering more or less the same content presented here. [28]

For future work, a deeper analysis of the GOCE re-entry will involve initial conditions closer to the re-entry date. Additionally, a diffusion factor that depends on time or state could be applied, further improving the model's accuracy to be comparable with high-fidelity propagators.

In conclusion, stochastic integration proves to be a valuable tool for analysing re-entry trajectories, especially in operational scenarios where a balance between precision and computational efficiency is crucial. The capability to use simplified but effective algorithms paves the way for advancements in future space missions and strengthens efforts to manage the risks related to space debris and satellites re-entry.

Bibliography

[1] I. Ahmadianfar, A. A. Heidari, A. H. Gandomi, X. Chu, and H. Chen. Run beyond the metaphor: An efficient optimization algorithm based on Runge Kutta method. *Expert Systems with Applications*, 181:115079, 2021.

[2] M. R. Akella and K. T. Alfriend. Probability of collision between space objects. *Journal of Guidance, Control, and Dynamics*, 23(5):769–772, 2000. doi: 10.1109/ICNC.2010.5583635.

[3] asteRisk. High-precision numerical orbital propagator. URL `https://search.r-project.org/CRAN/refmans/asteRisk/html/hpop.html`.

[4] F. Bernelli Zazzera. *Course Notes of Spacecraft Attitude Dynamics - Part 1: Attitude dynamics and kinematics*. Politecnico di Milano, 2022.

[5] F. Bernelli Zazzera, C. Colombo, and Y. Sidhoum. Re-entry predictions of space debris for collision avoidance with air traffic. *CEAS Space Journal*, 15(4):553–565, 2023.

[6] O. Cappé, S. J. Godsill, and E. Moulines. An overview of existing methods and recent advances in sequential monte carlo. *Proceedings of the IEEE*, 95(5):899–924, 2007.

[7] E. Choi, S. Cho, D. Lee, S. Kim, J. H. Jo, et al. A study on re-entry predictions of uncontrolled space objects for space situational awareness. *Journal of Astronomy and Space Sciences*, 34(4):289–302, 2017.

[8] C. Colombo. *Lecture slide of Orbital mechanics - Ch5: Orbit perturbation*. Politecnico di Milano, 2022.

[9] P. Di Lizia. *Lecture notes of Spacecraft Guidance and Navigation - Navigation part*. Politecnico di Milano, 2023.

[10] ESA. Goce re-entry dataset. `https://goce-ds.eo.esa.int/oads/access/collection/GOCE_Level_1/tree`, 2013.

[11] ESA Space Debris Office. ESA's annual space environment report. Technical Report 8, ESA, Robert-Bosch-Strasse 5, D-64293, Darmstadt, Germany, 7 2024.

[12] J. L. Foster and H. S. Estes. *A parametric analysis of orbital debris collision probability and maneuver rate for space vehicles*. NASA, National Aeronautics and Space Administration, Lyndon B. Johnson Space ..., 1992.

[13] Y. S. García. Satellite orbit determination using sequential filtering methods. Master's thesis, Universidad Carlos III de Madrid, 2021-2022.

[14] J. Geul, E. Mooij, and R. Noomen. Analysis of uncertainties and modeling in short-term reentry predictions. *Journal of Guidance, Control, and Dynamics*, 41(6):1276–1289, 2018.

[15] P. W. Hawkes. *Modern map methods in particle beam physics*. Academic Press, 1999.

[16] A. M. Johansen, L. Evers, and N. Whiteley. Monte Carlo methods. *International encyclopedia of education*, pages 296–303, 2010.

[17] T. S. Kelso. Celestrak. `https://celestrak.org/satcat/table-satcat.php?NAME=cubesat&PAYLOAD=1&MAX=500`, 1985.

[18] H. Klinkrad. *Space debris: models and risk analysis*. Springer Science & Business Media, 2006.

[19] H. Klinkrad. Methods and procedures for re-entry predictions at esa. In *6th European Conference on Space Debris, Proceedings of the conference held*, pages 22–25, 2013.

[20] G. D. Krebs. Roseycubesat 1. `https://space.skyrocket.de/doc_sdat/roseycubesat-1.htm`, Oct 2024. Accessed on: October 24, 2024.

[21] Y. Luo and Z. Yang. A review of uncertainty propagation in orbital mechanics. *Progress in Aerospace Sciences*, 89:23–39, 2017.

[22] S. Mahadevan. Monte carlo simulation. *Mechanical Engineering-New York and Basel-Marcel Dekker-*, pages 123–146, 1997.

[23] M. Mahooti. High precision orbit propagator, May 2024. URL `https://it.mathworks.com/matlabcentral/fileexchange/55167-high-precision-orbit-propagator`.

[24] P. Musgrave. N-body physics. `https://nbodyphysics.com/blog/gravity-engine-doc-1-3-2-2-2/orbit-parameters/`, 2016.

[25] R. S. Park and D. J. Scheeres. Nonlinear mapping of Gaussian statistics: theory

and applications to spacecraft trajectory design. *Journal of guidance, Control, and Dynamics*, 29(6):1367–1375, 2006.

[26] S. Pérez-Vieites, I. P. Mariño, and J. Míguez. Probabilistic scheme for joint parameter estimation and state prediction in complex dynamical systems. *Physical Review E*, 98(6):063305, 2018.

[27] N. Qi, Q. Sun, and Y. Yang. Effect of J3 perturbation on satellite position in LEO. *Aircraft Engineering and Aerospace Technology*, 90(1):74–86, 2018.

[28] A. Saracini, Y. Sztamfater Garcia, and M. Sanjurjo Rivo. Stochastic integration for re-entry analysis. In *Proceeding of the 75th International Astronautical congress*, 2024.

[29] P. W. Schumacher Jr, C. Sabol, C. C. Higginson, and K. T. Alfriend. Uncertain Lambert problem. *Journal of Guidance, Control, and Dynamics*, 38(9):1573–1584, 2015.

[30] G. Terejanu, P. Singla, T. Singh, and P. D. Scott. Uncertainty propagation for nonlinear dynamic systems using Gaussian mixture models. *Journal of guidance, control, and dynamics*, 31(6):1623–1633, 2008.

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

[32] D. A. Vallado. *Fundamentals of astrodynamics and applications*, volume 4. Microcosm Press, 2013.

[33] M. L. Vasile, E. Minisci, R. Serra, J. Beck, and I. Holbrough. Analysis of the de-orbiting and re-entry of space objects with high area to mass ratio. In *AIAA/AAS Astrodynamics Specialist Conference*, page 5678, 2016.

[34] B. B. Virgili, T. Flohrer, S. Lemmens, and H. Krag. Goce re-entry campaign. In *5th international GOCE user workshop, Paris, France*, 2014.

[35] X. Wang and H. Xie. Simulation of covariance analysis describing equation technique (cadet) in missile hit probability calculation. In *2010 Sixth International Conference on Natural Computation*, volume 8, pages 4282–4285, 2010. doi: 10.1109/ICNC.2010.5583635.

[36] B. Weber. Reference Frames. `https://orbital-mechanics.space/intro/reference-frames.html`, 2020.

[37] N. Wiener. The homogeneous chaos. *American Journal of Mathematics*, 60(4):897–936, 1938.

A | Appendix A - Orbital elements

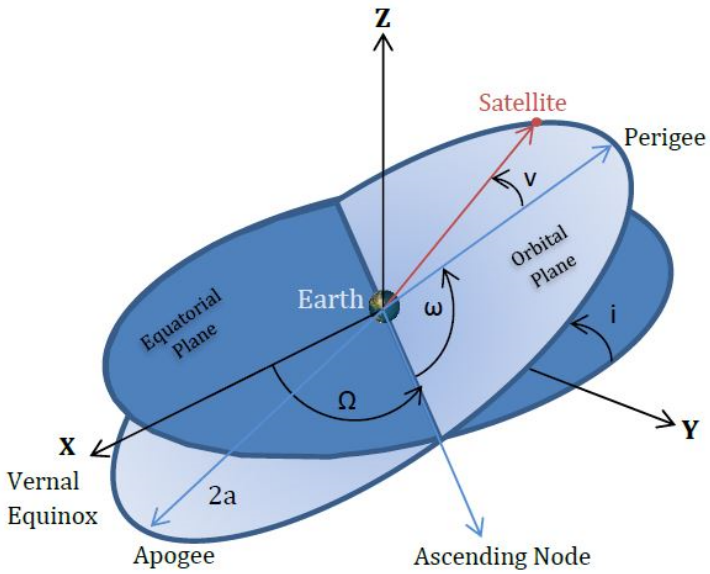


Figure A.1: Keplerian elements that fully characterise the orbit [24]

- The semi-major axis, a, represents half the distance between apogee and perigee.
- Eccentricity, e, describes the orbit's elongation compared to a circle.
- Inclination, i, indicates the tilt of the orbit relative to the Equatorial plane.
- Ω, known as Right Ascension of the ascending node, is the angle between the Vernal equinox and the ascending node, which is the point where the orbital plane crosses the equatorial plane.
- ω, referred to as Argument of perigee, is the angle between the ascending node and the line joining Earth's center to the orbit's perigee.

- θ, known as the true anomaly, is the angle between the line joining Earth's center to the perigee and the satellite's current position in orbit.

B | Appendix B

In this appendix, all test setups not included in the main text are documented. Specifically, it provides additional results from the Roseycubesat-1 analysis, as discussed in section 4.1.

All remarks made in the main text also apply to the following plots. The number of samples used for the stochastic propagation are 250, to speed up the runs and to still have a clear PDF curve.

B.1. Low Diffusion Factors

Figure B.1 shows the re-entry window when the diffusion factor is set to $G = [0, 0, 0, 1, 1, 1] * 10^{-14}$.

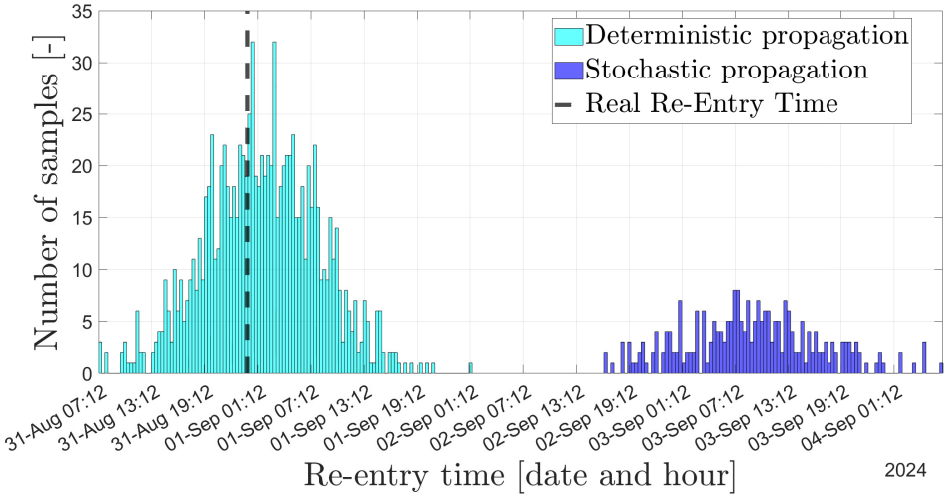


Figure B.1: Comparison of re-entry windows between the real re-entry date [17] predicted by the deterministic algorithm and that predicted by the stochastic one. This setup aims to evaluate the performance of the stochastic algorithm with a low σ against the deterministic benchmark.

Figure B.2 shows the re-entry window when the diffusion factor is set to $G = [0, 0, 0, 1, 1, 1] *$

10^{-12}.

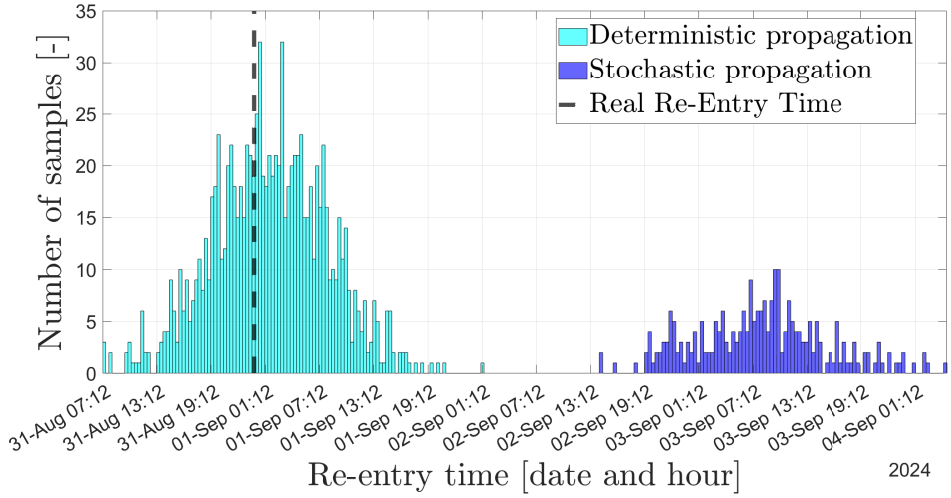


Figure B.2: Comparison of re-entry windows between the real re-entry date [17] predicted by the deterministic algorithm and that predicted by the stochastic one. This setup aims to evaluate the performance of the stochastic algorithm with a low σ against the deterministic benchmark.

Figure B.3 shows the re-entry window when the diffusion factor is set to $G = [0, 0, 0, 1, 1, 1] * 10^{-11}$.

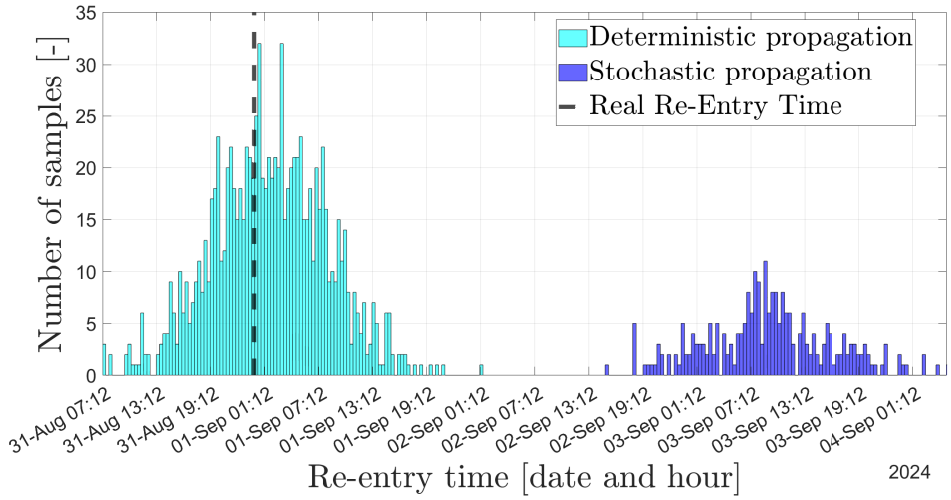


Figure B.3: Comparison of re-entry windows between the real re-entry date [17] predicted by the deterministic algorithm and that predicted by the stochastic one. This setup aims to evaluate the performance of the stochastic algorithm with a low σ against the deterministic benchmark.

Figure B.4 shows the re-entry window when the diffusion factor is set to $G = [0, 0, 0, 1, 1, 1] * 10^{-10}$.

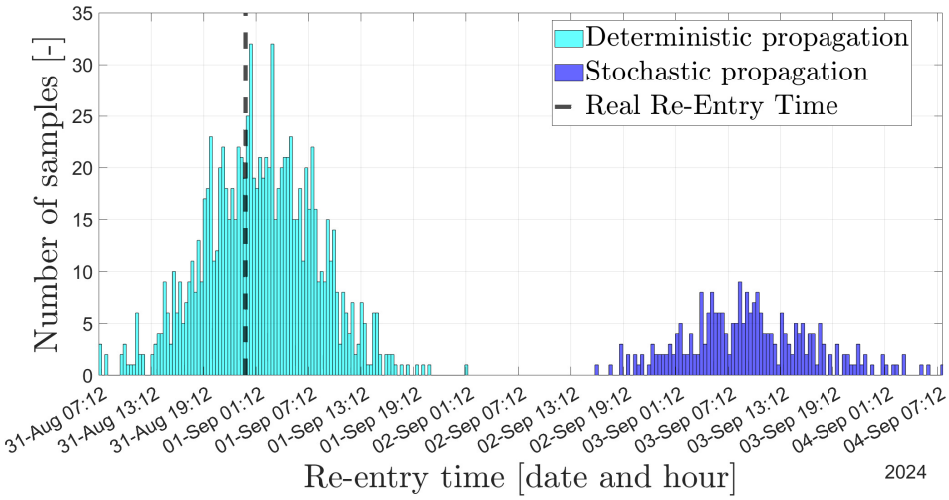


Figure B.4: Comparison of re-entry windows between the real re-entry date [17] predicted by the deterministic algorithm and that predicted by the stochastic one. This setup aims to evaluate the performance of the stochastic algorithm with a low σ against the deterministic benchmark.

Figure B.5 shows the re-entry window when the diffusion factor is set to $G = [0, 0, 0, 1, 1, 1] * 10^{-9}$.

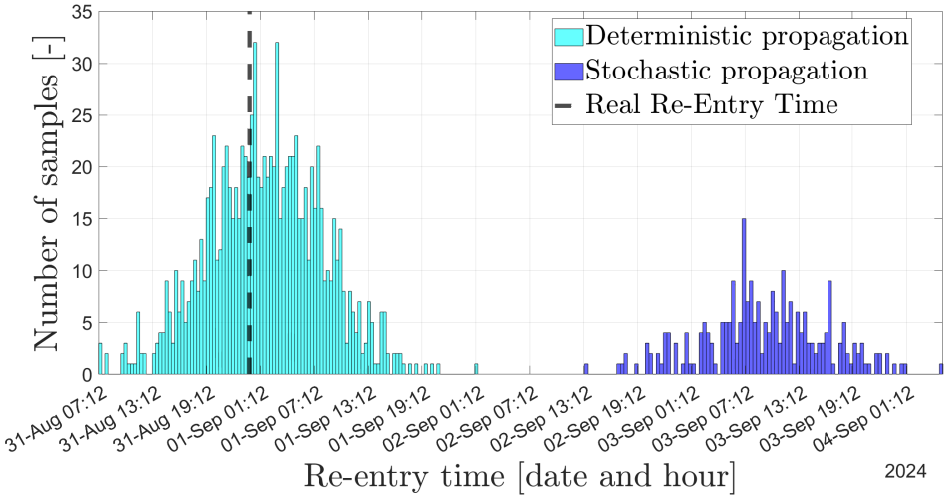


Figure B.5: Comparison of re-entry windows between the real re-entry date [17] predicted by the deterministic algorithm and that predicted by the stochastic one. This setup aims to evaluate the performance of the stochastic algorithm with a low σ against the deterministic benchmark.

Figure B.6 shows the re-entry window when the diffusion factor is set to $G = [0, 0, 0, 1, 1, 1] * 10^{-8}$.

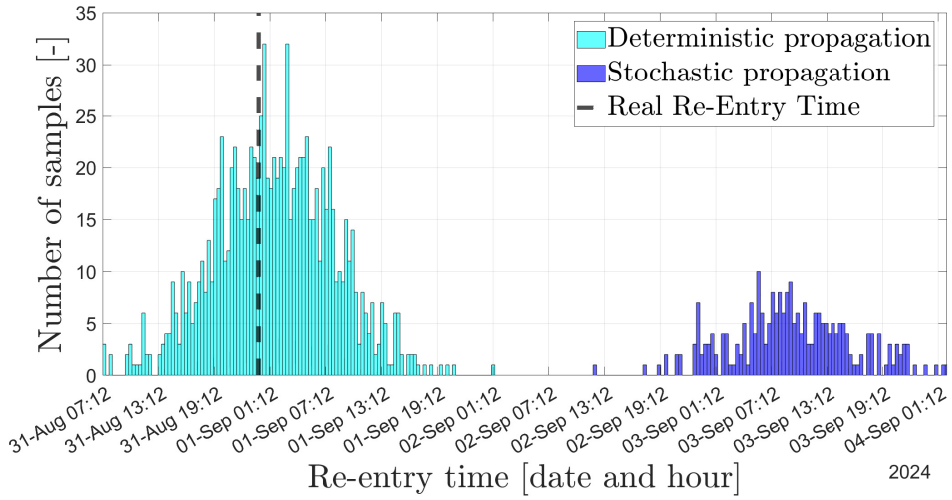


Figure B.6: Comparison of re-entry windows between the real re-entry date [17] predicted by the deterministic algorithm and that predicted by the stochastic one. This setup aims to evaluate the performance of the stochastic algorithm with a low σ against the deterministic benchmark.

Figure B.7 shows the re-entry window when the diffusion factor is set to $G = [0, 0, 0, 1, 1, 1] * 10^{-7}$.

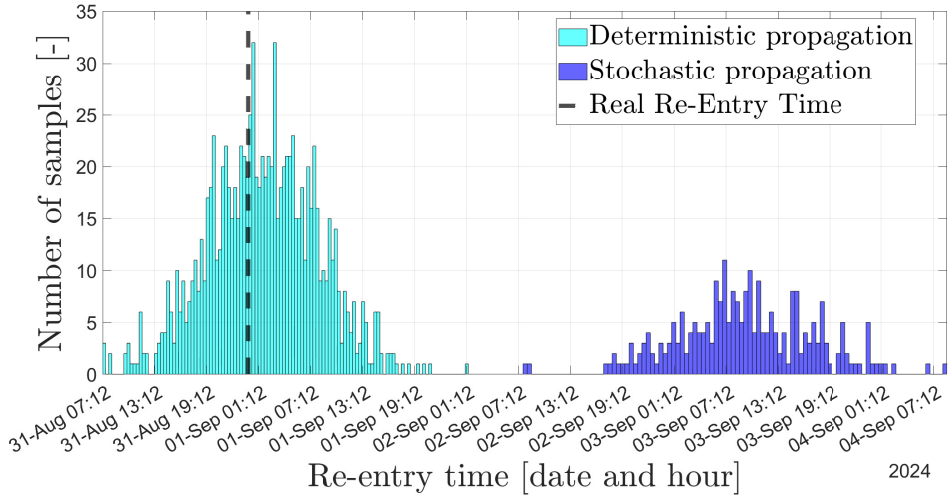


Figure B.7: Comparison of re-entry windows between the real re-entry date [17] predicted by the deterministic algorithm and that predicted by the stochastic one. This setup aims to evaluate the performance of the stochastic algorithm with a low σ against the deterministic benchmark.

B.2. High diffusion factors

The following plots display only the re-entry window predicted by the stochastic propagator. This is due to the fact that the re-entry occurs so early that including other elements in the image obscures the visibility of the data.

Figure B.8 shows the re-entry window when the diffusion factor is set to $G = [0, 0, 0, 1, 1, 1] * 3 * 10^{-5}$.

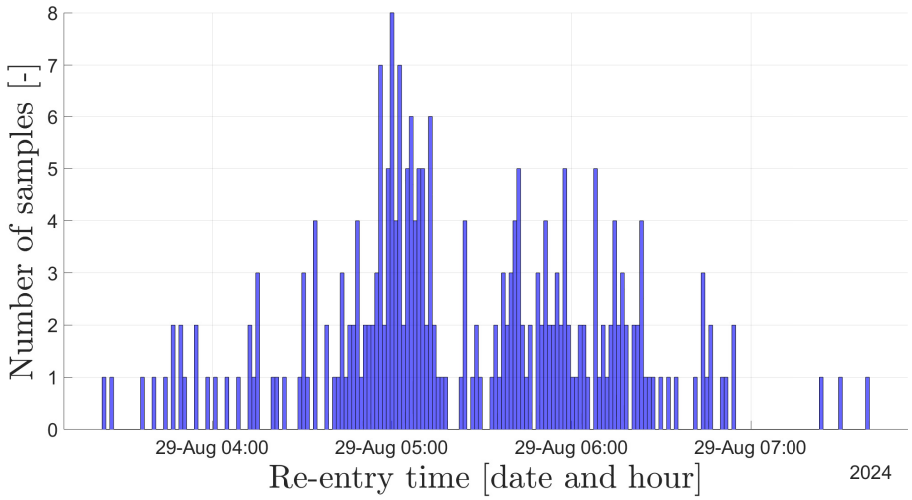


Figure B.8: Re-entry windows predicted by the stochastic algorithm. This setup aims to evaluate the performance of the stochastic algorithm with a high σ against the deterministic benchmark.

In this figure, all the samples re-enter within 30 minutes from the beginning of the propagation. Figure B.9 shows the re-entry window when the diffusion factor is set to $G = [0, 0, 0, 1, 1, 1] * 10^{-3}$.

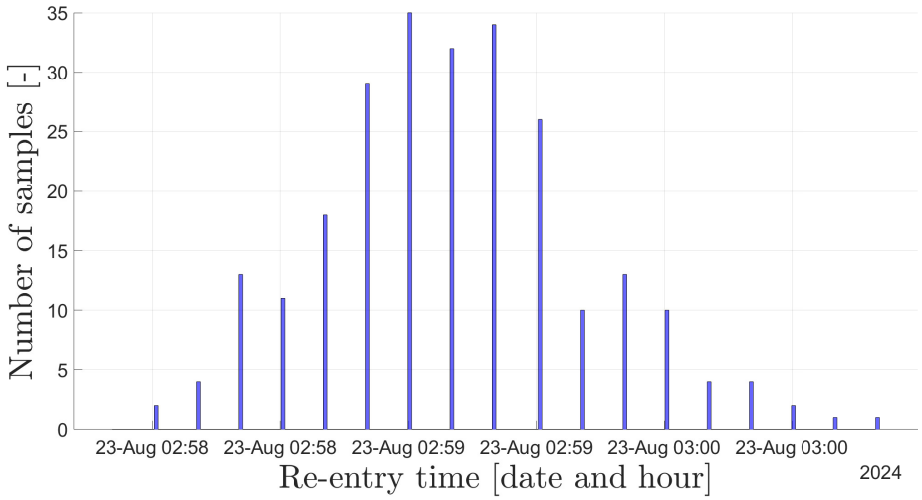


Figure B.9: Re-entry windows predicted by the stochastic algorithm. This setup aims to evaluate the performance of the stochastic algorithm with a high σ against the deterministic benchmark.

List of Figures

List of Tables

Acronyms

3DOF 3-degree-of-freedom
6DOF 6-degree-of-freedom

AoP Argument of perigee
AU Astronomical Unit

DA Differential algebra
DCM Direction Cosine Matrix
DRAMA Debris Risk Assessment and Mitigation Analysis

ECI Earth-centred inertial
EKF Extended Kalman Filter
ESA European Space Agency
ESOC European Space Operations Center

GG Gravity Gradient
GMM Gaussian mixture model
GOCE Gravity Field and Steady-State Ocean Circulation Explorer

HPOP High Precision Orbit Propagator

JB Jacchia-Bowman

KF Kalman Filter

LEO Low-Earth Orbit
LinCov Linear Covariance
LVLH Local-Vertical-Local-Horizontal

MC Monte Carlo

PC Polynomial Chaos Expansions
PDF Probability Density Function

ODE Ordinary Differential Equation

RAAN Right Ascension of the ascending node
RTN Radial-Transverse-Normal
RSO Resident Space Object
RK4 Runge-Kutta 4

SMA Semi-major axis
S/C Spacecraft
SGP4 Simplified Perturbation Models 4
SRP Solar Radiation Pressure
SDE Stochastic Differential Equation
SSA Space Situational Awareness
STM State Transition Matrix
STT State Transition Tensors
SRK4 Stochastic Runge-Kutta 4

TLE Two-Line Elements

UKF Unscented Kalman Filter
UT Unscented Transform
UTC Coordinated Universal Time

List of Symbols

Variable	Description	SI unit
H	altitude	km
$\vec{a}_p$	acceleration	$\frac{\text{km}}{\text{s}^2}$
μ	Earth gravitational parameter	398600 $\frac{\text{km}^3}{\text{s}^2}$
$\vec{r}$	position vector	km
$R_\oplus$	Earth radius	6378.1363 km
$\vec{x}$	state vector	km/$\frac{\text{km}}{\text{s}}$
$\vec{v}$	velocity	$\frac{\text{km}}{\text{s}}$
a_{DRAG}	acceleration due to drag	$\frac{\text{m}}{\text{s}^2}$
ρ	atmospheric density	$\frac{\text{kg}}{\text{m}^3}$
$\vec{v}_{\text{rel}}$	relative velocity with respect to atmosphere	$\frac{\text{m}}{\text{s}}$
C_D	drag coefficient	–
A	satellite cross-sectional area	m^2
m	satellite mass	kg
$a_{J2_x}, a_{J2_y}, a_{J2_z}$	acceleration due to $J2$	$\frac{\text{m}}{\text{s}^2}$
$J2$	spherical harmonic coefficient of degree two	0.0010826267
r	norm of the position vector	km
x, y, z	components of the state vector	km
$a_{J3_x}, a_{J3_y}, a_{J3_z}$	acceleration due to $J3$	$\frac{\text{m}}{\text{s}^2}$
$J3$	spherical harmonic coefficient of degree three	-0.0000025327
$a_{J4_x}, a_{J4_y}, a_{J4_z}$	acceleration due to $J4$	$\frac{\text{m}}{\text{s}^2}$
$J4$	spherical harmonic coefficient of degree four	-0.0000016196
$\dot{\omega}_x, \dot{\omega}_y, \dot{\omega}_z$	components of angular acceleration	$\frac{\text{rad}}{\text{s}^2}$
I_x, I_y, I_z	moments of inertia along the x, y, z axes	$\text{kg} \cdot \text{m}^2$
$\omega_x, \omega_y, \omega_z$	components of angular velocity	$\frac{\text{rad}}{\text{s}}$

Variable	Description	SI unit
$\vec{r}_b$	Position vector in the body frame	km
$\mathbf{A}_{bn}$	Direction Cosine Matrix (DCM)	–
q_1, q_2, q_3, q_4	Quaternion components	–
$\vec{T}_{gg}$	Gravity gradient torque	Nm
$\hat{i}_b, \hat{j}_b, \hat{k}_b$	Unit vectors in the body frame	–
B_{ip}, B_{jp}, B_{kp}	Components of the inverse of the ballistic coefficient along the positive directions of the body-fixed reference frame	$\frac{\mathrm{m}^2}{\mathrm{kg}}$
B_{in}, B_{jn}, B_{kn}	Components of the inverse of the ballistic coefficient along the negative directions of the body-fixed reference frame	$\frac{m^2}{\mathrm{kg}}$
$\hat{u}_v$	Unit vector representing relative velocity	–
a	Semi-major axis	km
e	Eccentricity	–
i	Inclination	°
Ω	Right ascension of the ascending node	°
ω	Argument of perigee	°
θ	True anomaly	°
$p_{SR@1AU}$	Solar radiation pressure at 1 Astronomical Unit	$\frac{N}{\mathrm{m}^2}$
AU	Astronomical Unit, the average distance between the Earth and the Sun	1.496×10^8 km
$\vec{r}_{sc-Sun}$	Position vector of the spacecraft with respect to the Sun	km
$\|\|\vec{r}_{sc-Sun}\|\|$	Norm of the position vector, representing the distance between the spacecraft and the Sun	km
C_R	Reflection coefficient of the spacecraft	–
A_{Sun}	Effective area of the spacecraft exposed to solar radiation	m^2
$\vec{T}_{drag}$	Drag torque acting on the spacecraft	Nm
$d\vec{F}$	Infinitesimal aerodynamic drag force acting on a surface element	N
ρ	Atmospheric density	$\frac{\mathrm{kg}}{\mathrm{m}^3}$
$\hat{N}$	Unit vector normal to the surface element	–

C | Acknowledgements

I would like to thank my two supervisors, Juan Luis Gonzalo Gomez and Manuel Sanjurjo Rivo, for their guidance, technical feedback, and direction in resources and algorithms throughout the work.

I would also like to thank Joaquín Míguez, who provided the numerical integration code (SRK).

Finally, I am deeply grateful to Yannick Sztamfater Garcia, whose support during the development of this thesis was invaluable. His expertise provided guidance, and his technical assistance was fundamental in bringing this work to completion.