Politecnico di Torino
Department of Mechanical and Aerospace Engineering
Master's Degree in Aerospace Engineering
2024/2025

Multi-Objective Optimization of Satellite Deorbiting Maneuvers Considering Collision Risk

Supervisors:
Prof. Lorenzo CASALINO
Dr. Giuseppe DI PASQUALE
Prof. Manuel SANJURJO RIVO

Candidate:
Livia MARCIANO

Abstract

The accumulation of space objects in Low Earth Orbit (LEO) is the result of decades of space activities lacking proper regulations and accountability, as well as the recent proliferation of large multi-satellite concepts, such as mega-constellations. The presence of these objects in LEO, including both active satellites and debris, poses increasingly higher risks to space assets and their operations, due to potential collisions between them. Alongside various actions that the space industry is undertaking to mitigate this issue, the disposal of satellites plays a central role. Disposal can be actively performed employing a propulsion system, either electric or chemical. The optimal disposal strategy depends on a series of factors, creating potential trade-off opportunities between maneuvering effort, which relates to the system capabilities and its mass, and the cumulative probability of collision during the disposal phase, which is a function of the LEO regions crossed and the residence time.

This work proposes a study using a multi-objective approach to automate the search for optimal deorbiting maneuvers, determining the optimal conditions for ΔV, collision probability along the deorbiting trajectory, firing time, and decay time. The implemented method considers a non-singular orbital elements formulation for the dynamics of the satellite, including perturbations such as Earth's oblateness and atmospheric drag, which uses the NRLMSISE-00 density model. In addition, the effects of onboard electric propulsion systems are evaluated, resulting in low-thrust and impulsive maneuvering strategies, respectively. Furthermore, a study on drag uncertainties is carried out. The probability of collision is computed using DRAMA from the European Space Agency, specifically with Assessment of Risk Event Statistics (ARES). Since ARES provides the annual collision probability for a nominal orbit, this work extends its usage for a deorbiting satellite, providing an accurate estimation of the collision risk.

The result of this process is a Pareto optimal front which allows mission designers to explore optimal solutions to the problem and evaluate the best option for each specific mission. Furthermore, this work analyzes the impact of the technology used, highlighting the difference in approach between electric systems, their diverse array of performances, and their capability to fulfill increasingly stringent regulations on disposal.

The findings of this study contribute to the enhancement of post-mission disposal analysis for LEO satellites, emphasizing the critical importance of a multi-objective optimization approach in the end-of-life planning of satellite missions and ensuring the long-term sustainability of space activities.

Contents

List of Tables

List of Figures

List of Algorithms

Nomenclature

(ANN) Artificial Neural Network
(ARES) Assessment of Risk Event Statistics
(COE) Classical Orbital Elements
(DRAMA) Debris Risk Assessment and Mitigation Analysis
(DM) Decision Maker
(ECEF) Earth-Centered, Earth-Fixed
(ECI) Earth-Centered Inertial
(ESA) European Space Agency
(FEEP) Field-Emission Electric Propulsion
(GA) Genetic Algorithm
(HEP) Hall Effect Thruster
(ISS) International Space Station
(LEO) Low Earth Orbit
(MASTER) Meteoroid and Space Debris Terrestrial Environment Reference
(MEE) Modified Equinoctial Orbital Elements
(MOP) Multiobjective Optimization Problem
(ORDEM) Orbital Debris Engineering Model
(RAAN) Right Ascension of Ascending Node
(SMA) Semi Major Axis
(S/C) Spacecraft
(SSO) Sun-Synchronous Orbit

Terms specific to the present document

- **Closed loop control**: A control system that uses feedback to adjust the input in order to maintain a desired state.
- **Collision**: The event of objects colliding, resulting in fragments spreading throughout space due to perturbations.
- **Conjunction**: The event of objects coming close to each other without colliding.
- **Disposal phase**: The time between the end of a mission and the end of a spacecraft's life.
- **End of life**: Instant in which a spacecraft is permanently turned off.
- **End of mission**: Instant in which a spacecraft completes the tasks it was supposed to do.
- **Edelbaum's theory**: The theory explains low-thrust maneuvers in circular orbits with low inclinations and thrust involved.
- **Genetic Algorithm**: It is a numerical optimization algorithm applicable to a wide range of problems. These algorithms are initialized with a population of guesses.
- **Hohmann transfer**: Minimum energy transfer in two burns between circular orbits.
- **Indirect method**: It is a numerical optimization algorithm used to solve for optimal control and state variable simultaneously.
- **Re-entry**: The re-entry of a space object into Earth's atmosphere.

Chapter 1

Introduction

1.1 Background & Motivation

Since the beginning of the space age, the amount of space debris orbiting Earth has gradually increased due to routine space operations and on-orbit fragmentation. This debris includes spent upper stages, mission-related objects, but are also due to satellite explosions, and interceptions by surface-launched missiles. As of December 6, 2023, the updated information indicates the following: 36,500 space debris objects larger than 10 cm, 1,000,000 objects ranging from greater than 1 cm to 10 cm, and 130 million objects from greater than 1 mm to 1 cm [1]. Consequently, space debris significantly contributes to the particulate environment around Earth, alongside natural objects such as meteoroids.

When discussing space debris, it is essential to introduce the concept of the Kessler Syndrome [2]. This phenomenon denotes the accumulation of space debris as a chain reaction, wherein additional debris triggers more collisions, resulting in the expansion of a belt of debris around Earth. The resulting growth will be non-linear. Although we have not yet reached the point of the Kessler Syndrome, there is an expectation that we are moving in that direction. Indeed, the current debris population carries a certain probability of collision, posing a potential risk of damage to orbiting objects within this environment. Collisions have already occurred [3]: in 1996 between the Cerise satellite and the Ariane rocket stage; in 2007, a Chinese test resulted in the destruction of a Chinese satellite, creating 150,000 fragments larger than 1 cm; in 2009 between the Iridium satellite and the defunct Cosmos 2251, creating 61,000 fragments larger than 1 cm. Therefore, looking at future developments in Low Earth Orbit (LEO), we anticipate an increase in both collision fragments and the total number of objects in space. This projection is depicted in Figure 1.1, which forms part of a simulation from 2010 under the theoretical assumption of no future launches. Therefore, it is crucial to consider space debris in order to avoid jeopardizing the sustainability and safety of space

missions.

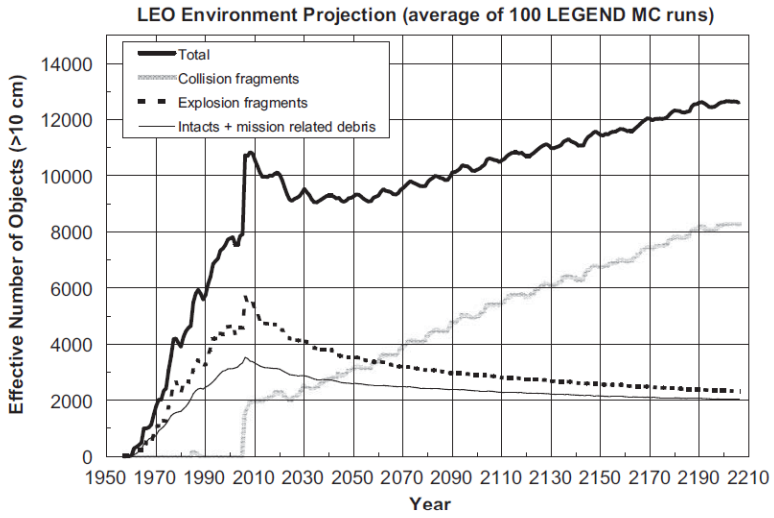


Figure 1.1: Effective number of objects VS time; LEO environment projection [4]

Putting aside future developments, the count evolution by object type as developed by ESA in 2023 shows, in Figure 1.2, a clear increase in the object count [5]:

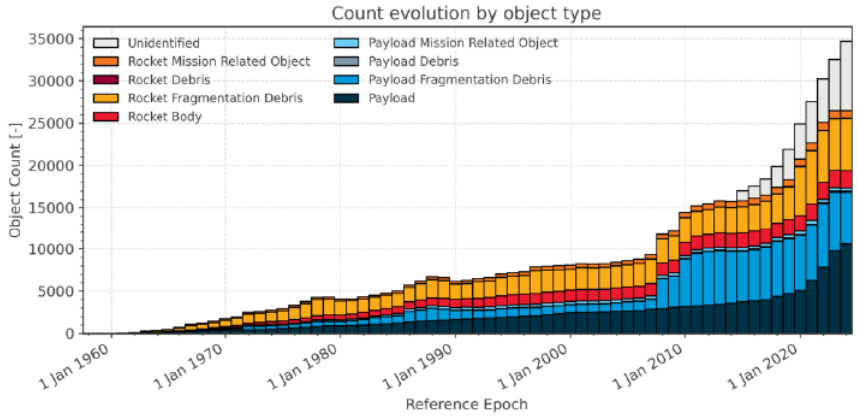


Figure 1.2: Count evolution in time by object type [5]

Finally, the amount of space debris is not a concern in all orbits. Indeed, the higher the altitude, the longer the natural deorbiting time. Even if it depends on the ballistic coefficient, which is proportional to the area-to-mass ratio ($BC = \frac{m}{C_d A}$ where m is the mass of the object $[kg]$, C_d is the drag coefficient $[-]$, and A is the frontal area of the object exposed to the airflow $[m^2]$), it mainly occurs above 800 km of altitude because, at 500 km, aerodynamic drag naturally cleans debris, resulting in a de-orbit time of approximately one year [6]. Hence, the lower region of LEO orbits stand out as the more densely populated ones, as

illustrated in Figure 1.3. Noteworthy is the Sun-synchronous orbit, which emerges as the most populated among them [7]. The widespread use of these orbits is attributed to their numerous benefits. Primarily, the consistent sunlight exposure at specific Earth locations enhances their suitability for various research and observation missions. Additionally, their relatively low altitude allows for sharp ground resolution, while their high inclination ensures nearly worldwide coverage.

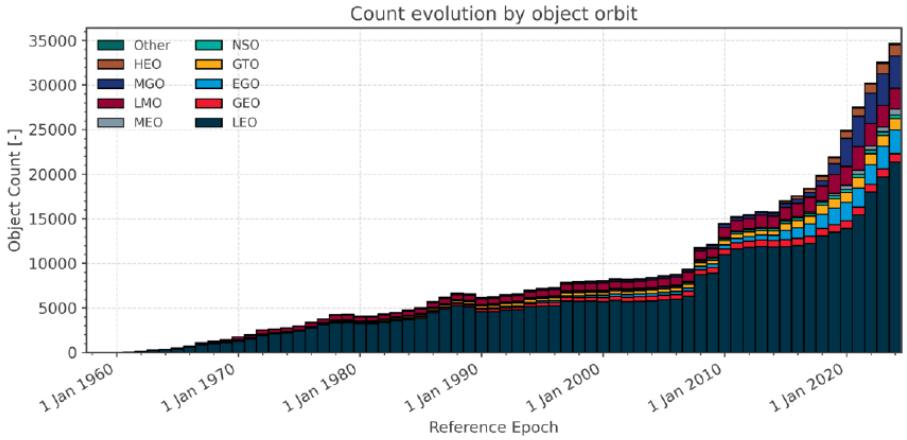


Figure 1.3: Count evolution in time by object orbit [5]

1.2 Problem Statement

To preserve the congested LEO, it is crucial to limit and control the future growth of the space debris population. This involves considering two distinct approaches. On one hand, there is a need to prevent the creation of new space debris, which underscores the importance of **post-mission disposal**. On the other hand, there is a concern about removing large cross-sectional objects already in space. Addressing this concern involves the implementation of **active debris removal**—a method designed to eliminate passive elements from their orbits using various technologies such as harpoons, nets, lasers, and more. Concerning small objects, protective measures are developed to enhance mission safety within this particulate environment.

Over the years, the ESA has developed a Space Debris Mitigation standard with the aim of enhancing the efficiency of debris removal in Earth and Lunar orbits. Recognizing that current mitigation approaches may not be sufficient to guarantee a sustainable space environment, the standard was updated in November 2023 to introduce a Zero Debris approach [8]. It contains a set of rules to be followed by every future European mission. The current debris mitigation guidelines mandate that a satellite or rocket body in low-Earth orbit (LEO)

should not remain in orbit longer than 25 years after its end of life. However, the European Space Agency (ESA) has proposed more stringent Zero Debris recommendations, which suggest reducing this timeframe to below five years. These recommendations also emphasize the importance of minimizing collision risks during the entire mission's life. Specifically, the cumulative probability of collision with any debris larger than 1 cm should be kept below 1 in 1000 until the object reenters Earth's atmosphere. Additionally, the likelihood of successfully executing the disposal must be greater than 90% (it means that almost everything will be burned in atmosphere during the re-entry phase).
At this point, discussing decommissioning, various possibilities [9] should be taken into account:

- **Uncontrolled De-orbiting:** After being placed into a specific orbit, the spacecraft naturally de-orbits within a predetermined time frame regulated by norms. This process involves initiating a deceleration maneuver to reduce the perigee altitude and the altitude. This solution is suitable for spacecraft that are completely destroyed during atmospheric re-entry.

- **Controlled De-orbiting:** It is used when the atmospheric destruction is expected to be incomplete. In this scenario, the re-entry point must be predetermined to mitigate the risk of human casualties exceeding a specified limit. Similar to uncontrolled de-orbiting, controlled de-orbiting involves initiating a reduction in perigee altitude. This one is not possible with low-thrust or passive means (too low acceleration, the deorbit system cannot change the trajectory fast enough to achieve the required re-entry point).

- **Graveyard Orbit:** Satellites above 1500 km altitude are transferred to higher-altitude orbits. This procedure is commonly practiced for GEO satellites.

Considering only the first two options, as this Master Thesis refers to lower-region of LEO orbits, it becomes evident that braking is required in both cases. This braking can be achieved through various de-orbiting technologies, categorized as either active or passive solutions. The choice of de-orbiting strategy is influenced by the debris environment. **Passive deorbit methods** require no further active control after deployment and they are propellant-free:

- **Drag augmentation devices (sails, ballons):** It is a method to enhance drag by augmenting the effective area of a satellite, with the primary goal of minimizing the de-orbit period. This solution is scalable, adaptable, and reliable, making it particularly suitable for small satellites. However, the risk of collision is influenced by various

parameters, including the area and the de-orbit time. While reducing the deorbit time is beneficial, the increase in the effective area may elevate the probability of collision. Thus, a trade-off is essential in determining the optimal balance between minimizing deorbit time and managing collision risks.

- **Electrodynamic tethers:** Electrodynamic tethers consist of conducting wires, either bare or insulated, designed to induce a voltage drop along their length by utilizing the ambient magnetic field. The current flowing through the tether interacts with the Earth's geomagnetic field, generating a Lorentz force. This force can be strategically employed to deorbit spacecrafts.

Active deorbit methods are characterized by applying thrust in the opposite direction of the object's motion to reduce orbital speed for deorbiting. Indeed, by reducing the speed of the spacecraft, the orbital energy of the spacecraft decreases, leading to a lower orbit:

- **Chemical propulsion:** Utilizing chemical propulsion involves the expulsion of high-speed gases through combustion, generating thrust.

- **Electric propulsion**: Electric propulsion is a type of spacecraft propulsion that uses electrical energy to accelerate propellant particles and generate thrust. Depending on their operating principles, we can differentiate among electrothermal propulsion (Resistojet and Arcjet), electrostatic propulsion (Ion thrusters and Hall thrusters), and electromagnetic propulsion (Magnetoplasmadynamic, Pulsed Plasma Thrusters, and Variable Specific Impulse Magnetoplasma Rocket).

Continuing the discussion about braking, two different strategies [9],[10] can be implemented:

- A high-thrust impulsive maneuver which sends the spacecraft into an elliptical orbit. This deorbiting strategy involves increasing the eccentricity of the orbit (the initial orbit is indeed circular) to decrease the altitude of the perigee.

- A low-thrust maneuver, based on continuous operation, which sends the spacecraft into a circular orbit. This deorbiting strategy involves decreasing the semimajor axis.

After providing an overview of various aspects related to deorbiting, we can now examine specific instances of deorbiting missions. For the cases described below, in order to better understand the differences between using chemical and electric propulsion, as well as other relevant factors, and to analyze the deorbiting paths of satellites, their orbital data were extracted from historic Two-Line Elements (TLEs). Historical TLE data were accessed using

a Python script that utilized the SpaceTrack library [11] and the SpaceTrackClient module. The evolution of the pericenter and apocenter radii of the orbits over time was calculated, and the results were saved into a CSV file. Subsequently, a MATLAB script was developed to read the CSV file and generate plots showcasing these trends (Figure 1.4).

- *Mir space station* [12], [13]: was a Soviet (later Russian) orbital facility, functioning in LEO from 1986 to 2001. Following 15 years of operational service, a controlled deorbit was initiated in 2001 to avert an uncontrolled descent to Earth. The deorbit process unfolded in three phases: an initial stage involving a waiting period to allow atmospheric drag to reduce Mir's orbit to 220 km, with the following docking of Progress M1-5 carrying 5900 pounds of propellant; a subsequent phase involving the adjustment of specific perigee and apogee altitudes through two braking burns of Progress's thrusters; and finally, transitioning from the descent orbit to the re-entry trajectory. All these maneuvers were executed to precisely direct Mir toward a specific re-entry point. To mitigate any potential threat to Earth, the re-entry was strategically executed above the South Pacific. Looking at this point in the Figure 1.4a, specifically referring to the initial segment of the graph, it becomes evident that the MIR space station was engaged in station-keeping maneuvers to counteract atmospheric drag. These maneuvers were essential to prevent orbital decay, and the irregular maintenance of the orbit suggests the use of chemical propulsion.

- *GOCE (Gravity Field and Steady-State Ocean Circulation Explorer)* [14]: was selected in 1999, and launched in 2009, as the first Earth Explorer Core Mission of ESA's Living Planet Programme. It operated in an extremely low Earth orbit, down to 229 km. It uses ion thrusters to execute station keeping and facing the drag, it is clear also looking at Figure 1.4b because it is very precise with respect to the Mir which was using chemical propulsion. GOCE underwent a decay of the orbit due to uncompensated atmospheric drag.

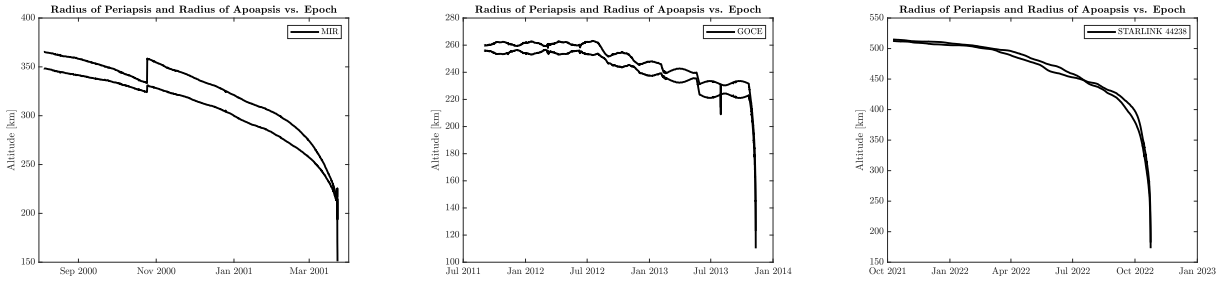
- *Starlink*: is a mega-constellation built by SpaceX, currently operating in LEO below 600 km of altitude. Its satellites are equipped with Hall thrusters. SpaceX proactively deorbits satellites that are identified as being at an elevated risk of becoming non-maneuverable. Starlink satellites burn up during reentry. Three different satellites are analyzed, representing three different cases:

  - *44238*: Based on the profiles, it is possible to interpret that the satellite is deorbited using only atmospheric drag (Figure 1.4c);

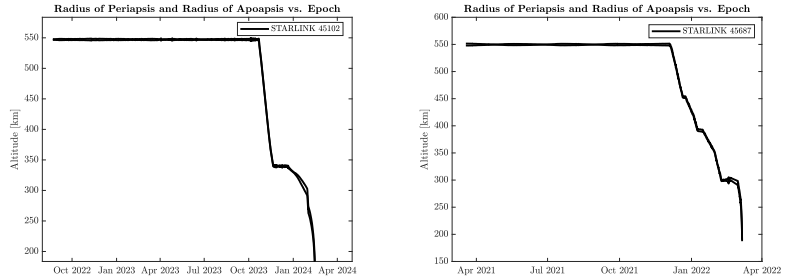
– *45102*: The deorbit process begins with a maneuver using electric propulsion, which takes longer due to the nature of this propulsion system. At a certain altitude, the maneuver is stopped, and the satellite continues deorbiting through drag alone (Figure 1.4d);

– *45687*: A series of maneuvers is executed to reduce the satellite's altitude until drag becomes sufficient for deorbiting (Figure 1.4e).

In both the last two cases, altitude is reduced through a maneuver, after which reentry is completed using atmospheric drag. For all these Starlink cases, as previously mentioned with GOCE, the initial station-keeping is very precise due to the use of electric propulsion.



(a) Mir space station (b) GOCE (c) Starlink 44238



(d) Starlink 45102 (e) Starlink 45687

Figure 1.4: The evolution over time of the pericenter and apocenter radii of the orbits, based on Two-Line Element (TLE) data from a series of missions, was analyzed.

1.3 Literature Review

The challenge of designing optimal deorbiting strategies, which include both active debris removal and post-mission disposal, involves numerous interconnected aspects, each requiring distinct mathematical approaches. To address these complexities and propose robust

solutions, the following section reviews the current state of the art in mathematical models and methodologies for deorbiting. This overview draws from a diverse range of studies and reports, with a few key examples highlighted below, showcasing a broad spectrum of strategies:

- Source: **Optimal mission planning of active space debris removal based on genetic algorithm** [15]
  - Publication date: 2019
  - Strategy, and additional information:
    - A three-stage removal strategy involves multiple spacecraft, including a main spacecraft and several small satellites [1];
    - An active debris removal approach is used;
    - This report refers to debris in low Earth orbit (LEO);
    - Means to deorbit: electric propulsion;
    - The J2000 coordinate frame is applied to the two-impulse rendezvous maneuvering model.
  - Optimal object: Minimum fuel consumption.
  - Constraint: Fuel, time and number of spacecrafts.
  - Algorithms and tools:
    - Genetic Algorithm: Used to determine the optimal sequence for space debris removal, aiming to minimize fuel consumption and improve efficiency;
    - Particle Swarm Optimization (PSO): Utilized for optimizing orbital maneuvers, specifically to find the optimal transfer orbit with the smallest impulse required, meeting time constraints;
    - Two-Impulse Optimal Rendezvous Model: This model is used for maneuvering calculations. It calculates the necessary impulses for transferring a spacecraft from one piece of debris to another, optimizing for minimal fuel consumption based on the mission's constraints.
  - Areas for Improvement: It might include the influence of perturbations.

[1] There is a main spacecraft named 'mission satellite,' which carries a series of propulsive satellites released in the vicinity of debris. These small satellites, utilizing electric propulsion, push space debris into lower orbits, with the goal of causing them to burn. Because of burning with the debris a conservation of fuel is achieved. The mission satellite moves from one target debris to another.

- Source: **Low-Thrust Many-Revolutions maneuver optimization for future active debris removal applications** [16]
  - Publication date: 2022
  - Strategy, and additional information:
    - Optimization of low-thrust, multiple-revolution transfers is achieved by comparing various initial conditions and identifying factors to minimize. A single target problem is developed;
    - An active debris removal approach is used;
    - This report refers to debris in low Earth orbit (LEO);
    - Means to deorbit: electrostatic engines;
    - It includes the presence of the J2 effect.
  - Optimal object: Minimum time solution or minimum propellant solution.
  - Constraint:
    - In the minimum time solution, the switching function is consistently positive, leading to the activation of engines. Conversely, in the minimum propellant solution, a three-phase transfer was executed, including a phase where engines are deactivated;
    - A constraint altitude was added to both types of solutions.
  - Algorithms and tools: Indirect methods (it is used to find the solution of a trajectory optimization problem): a modified version of the Edelbaum's theory is used.
  - Areas for Improvement:
    - It might include the influence of other perturbations;
    - A multiple target problem can be implemented.
- Source: **Time Optimal Drag-Based Targeted De-Orbiting for Low Earth Orbit** [17]
  - Publication date: 2023
  - Strategy, and additional information:
    - A single-stage optimization problem is executed;
    - It discusses the topic of satellite ballistic deorbiting from low Earth orbit (LEO);

    - ∗ Means to deorbit: drag modulation through the variation of the aerodynamic shape of the spacecraft;
    - ∗ Modified equinoctial orbital parameters are employed;
    - ∗ The optimization problem is addressed using the MATLAB software GPOPS-II, employing an hp adaptive Gaussian quadrature orthogonal collocation method. This method allows for precise and efficient calculation of optimal control strategies by breaking down the problem into smaller segments (phases) and solving them using high-order polynomial approximations.
    - ∗ Assumptions made: exponential model for the atmosphere; application to only circular orbits; the spacecraft is deorbited using aerodynamic drag.
  - Optimal object: Minimum time trajectories.
  - Constraint: Altitude at the de-orbit point (latitude and longitude are not directly imposed).
  - Algorithms and tools: A novel optimization algorithm has been crafted to derive minimum-time de-orbiting trajectories from diverse initial conditions, guiding spacecraft to a predetermined point at the re-entry interface.
  - Areas for Improvement: It can be combined with other technique such as ANNs to get closed loop control.

At this stage, all the key characteristics of post-mission disposal and active debris removal approaches have been outlined as solutions to address the increasingly congested state of the space environment. Additionally, the collision probability posed by the debris population has been identified as a significant concern for both current and future space missions. However, it is important to emphasize that the probability of collision is not only a challenge during a mission's operational life but also during the deorbiting phase (which can refer to the deorbiting of satellites after the end of their mission or satellites used to execute active debris removal). Therefore, the risk of collision along deorbiting trajectories becomes a critical aspect. Consequently, designing effective deorbit missions is essential.

Building on this, **the present work aims to investigate not only general post-mission disposal strategies but also optimal deorbiting maneuvers that minimize both ΔV and cumulative collision risk.**

Designing deorbiting strategies while accounting for collision probability presents a multifaceted challenge, encompassing several interconnected aspects. Below are a few key examples that illustrate a wide range of strategies:

- Source: **Drag and solar sail de-orbiting: re-entry time versus cumulative collision probability** [10]

- Publication date: 2017
- Strategy, and additional information:
  - It involves evaluating the efficiency of passive end-of-life deorbiting techniques;
  - It is applicable for altitudes up to 1000 km, as well as for high LEO altitudes up to 13,000 km;
  - Means to deorbit: solar sails and ballons;
  - It is applied to small and medium satellites weighing less than 1000 kg;
  - A comparison between two deorbiting strategies is made: decreasing the semi-major axis and increasing the eccentricity. The former method, commonly employed by drag sails, aims at gradually reducing the satellite's orbit. By increasing the cross-sectional area relative to the satellite's mass, atmospheric drag is enhanced. This drag acts against the satellite's velocity vector, gradually decreasing the semi-major axis and causing the satellite to spiral inward toward Earth. This process is direct and continuous, effectively lowering the satellite's altitude until it re-enters the atmosphere. It is particularly useful for satellites in low Earth orbit (up to about 1000 km), where atmospheric drag is significant.

    The latter method involves increasing the satellite's orbital eccentricity. By strategically deploying the sail to maximize the effect of solar radiation pressure when the satellite is moving toward the Sun, and minimizing it when moving away, the eccentricity of the orbit increases. This strategy is more suitable for satellites at higher altitudes (beyond 1000 km up to geostationary orbit levels), where solar radiation pressure can be effectively utilized;
  - A study on the collision risk is executed. The models used for this analysis include the Space Debris Model (SDM) and the CUBE algorithm, which help simulate and calculate collision probabilities dynamically as the satellite descends.
- Optimal object: Find the sail dimension that minimizes the impact on the debris population by considering the trade-off between the time required for deorbiting and the cumulative collision probability in LEO.
- Constraint: Maximum deorbit time.
- Algorithms and tools:
  - CUBE algorithm: Used for computing the collision probability at each time step of the satellite's deorbit trajectory. It integrates factors like the debris

flux and the satellite's area-to-mass ratio to provide a dynamic assessment of collision risk;

* Space Debris Model (SDM): This model simulates the background debris environment, which is crucial for assessing the collision risk faced by the satellite as it maneuvers through various orbital regions during deorbiting;
* Planetary Orbital Dynamics (PlanODyn): Employed for calculating the orbit evolution depending on the satellite's initial orbit and the deorbiting strategies applied (either drag or solar sailing). This tool is essential for determining how changes in the orbit impact the satellite's exposure to debris.

– Areas for Improvement: Exploring advanced optimization techniques to efficiently determine the minimum effective area-to-mass ratio required for deorbiting within specified time constraints.

- Source: **Assessing Collision Probability in Low-Thrust Deorbit [18]**
  - Publication Date: 2022
  - Strategy and Additional Information:
    * It focuses on evaluating collision probabilities and risks associated with low-thrust deorbiting techniques;
    * Specifically applicable to Low Earth Orbit (LEO) altitudes;
    * Means of deorbit: Laser-ablation-induced low thrust, though the techniques can be extended to other low-thrust deorbit missions;
    * It analyzes four scenarios differentiated by the percentage of the orbital period during which thrust is active;
    * Collision risk computed using the debris flux provided by the ESA MASTER-2009 model.
  - Optimal Objective: Identify the optimal thrusting strategy (50% thrust range) that minimizes the time to deorbit.
  - Constraint: Compliance with ISO guidelines for collision risk ($< 0.1\%$).
  - Algorithms and Tools:
    * **ESA MASTER-2009:** Simulates debris flux along the satellite's trajectory;
    * **Cowell's Numerical Method:** Models orbital propagation, accounting for perturbative forces such as atmospheric drag and solar radiation pressure;
    * **NASA Breakup Model:** Estimates fragments generated during collisions.

  - Areas for Improvement:
    * Designing deorbit trajectories that minimize fuel consumption.

- Source: **Interactions of the space debris environment with mega constellations—Using the example of the OneWeb constellation [19]**
  - Publication Date: 2017
  - Strategy and Additional Information:
    * OneWeb aims to minimize its impact on the LEO environment by implementing comprehensive space debris mitigation strategies throughout the satellite mission lifecycle.
    * Agreement with USSTRATCOM/JSpOC for data sharing to reduce orbit determination uncertainties.
    * Collision avoidance strategies implemented for all active mission phases.
    * Mission Lifecycle Approach: Deployment Phase; Operational Phase; Active Disposal Phase; Passive Reentry Phase.
    * Means of deorbit: electric propulsion
    * End-of-Life disposal success rate target: $\geq 90\%$.
    * Satellites equipped with grapple fixtures to allow future active debris removal.
  - Optimal Objective:
    * Minimize the risk of space debris generation.
    * Maintain a sustainable space environment for future satellite operations.
    * Ensure safe operation of the OneWeb constellation.
  - Constraint:
    * High satellite population density in LEO.
    * Uncertainty in orbital decay predictions and satellite reliability.
    * Risk of collision with background space debris.
  - Algorithms and Tools:
    * ESA-MASTER (for debris flux analysis).
    * ARES from ESA's DRAMA tool suite (for collision probability estimation).
    * NASA Breakup Model (to simulate fragmentation clouds).
    * FLORA propagator (for orbital evolution analysis).
    * NRLMSIS-2000E atmosphere model (for atmospheric density modeling).

- Areas for Improvement:
  - Develop more efficient collision avoidance strategies to reduce maneuver frequency.
  - Enhance modeling accuracy for long-term impact predictions of satellite constellations.
  - Investigate alternative orbital altitudes to reduce space debris risks.

As is apparent from the literature review presented, works aimed at optimizing decay focus on traditional objectives such as propellant consumption or decay duration. In contrast, this work explores the incorporation of collision risk within a multi-objective optimization process. Additionally, this work proposes a methodology to enhance the estimation of collision probability during the decay phase, thereby increasing the reliability of the results.

1.4 Thesis Structure

This Master's thesis focuses on the multi-objective optimization of satellite deorbiting maneuvers, balancing propellant consumption and collision probability. The study is structured into three main chapters, each addressing key aspects of the problem.

Chapter 2 is dedicated to the development of an orbit propagator, specifically designed for this study. The mathematical formulation of the problem is introduced, with a focus on the modified equinoctial parameters, which are used to describe the satellite's trajectory. The impact of perturbations, including Earth's oblateness (J2 effect) and atmospheric drag, is analyzed in detail, considering how the variations in atmospheric density influence the deorbiting process.

Once the orbital propagator is developed, its output is interpolated to serve as input for the ARES (Assessment of Risk Event Statistics) module, part of the DRAMA (Debris Risk Assessment and Mitigation Analysis) software developed by the European Space Agency (ESA). The ARES module is used to compute the probability of collision along the entire deorbiting trajectory. Given the high computational cost of these calculations, a discretization study is performed to optimize the number of intervals in which the propagator's output is divided. This ensures a balance between computational efficiency and accuracy in estimating the global probability of collision during the deorbit phase.

Chapter 3 presents numerical simulations conducted for different deorbiting scenarios, analyzing the relationship between firing time, decay time, ΔV and collision probability along the decay trajectory. The study explores how varying the thruster firing time impacts

both the deorbit duration and the cumulative probability of collision. A Pareto front is generated to illustrate all the possible optimal solutions. The implications of different deorbiting strategies using electric propulsion systems are also examined. The results of this trade-off analysis provide valuable insights into how mission designers can select optimal deorbiting configurations based on specific mission constraints. Finally, a study on drag uncertainty is conducted by varying solar activity over time using the Schatten predictions.

Chapter 4 summarizes the key findings, emphasizing the importance of a multi-objective optimization approach for satellite deorbiting maneuvers. The impact of discretization techniques, collision probability estimation, and propulsion system selection is discussed in the context of future space sustainability regulations.

Additionally, recommendations for future research directions are provided, focusing on improving collision risk assessment methodologies and enhancing computational efficiency in deorbit optimization models.

Chapter 2

Methodology

This chapter presents the methodology adopted to solve the optimization problem, as illustrated in Figure 2.1, using a Design Structure Matrix (DSM) framework. The diagram schematically represents the key inputs, outputs, and process blocks arranged in a sequential manner, each of which is detailed below.

The process begins with the optimizer (Step 1), which selects the firing duration,(t_{thr}) (Step 1). Parameters such as the drag area (A_d), thruster properties (T, I_{sp}) are inputs at this stage. The system then numerically propagates (Step 2).

The propagation step produces the state trajectory ($\boldsymbol{y}^*(t)$), propellant mass (m_p), and decay time (t_f), starting from the initial state $\boldsymbol{y}(t_0)$. These outputs are then passed to the discretizer (Step 3), which generates a discretized trajectory of n intervals ($\boldsymbol{t}_d$ and $\boldsymbol{y}_d$).

Finally, ESA's software tool Assessment of Risk Event Statistics (ARES) is executed (Step 4), provided of the collision area (A_c), which represents the size of the object, and the maximum and minimum particle size ($\boldsymbol{p}_s$) computing the global collision probability along the trajectory of decay according to the procedure detailed in the following chapter.

The optimizer iteratively loops through these steps, ultimately converging to an optimal solution in terms of the performance indicators. The final output consists of the optimized set (m_p^*, t_f^*, p_c^*), which typically maps out a Pareto front in the multi-objective design space.

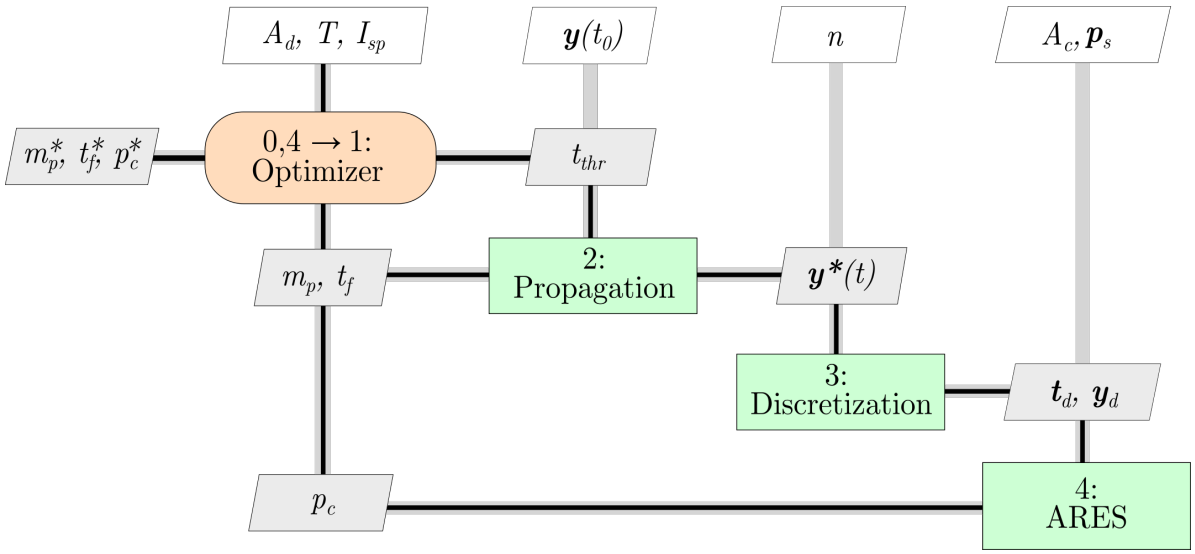


Figure 2.1: Design structure matrix of the methodology.

2.1 Mathematical Formulation of the Problem

In this section, the process of creating a propagator to model the decay of a satellite is described. The reasoning behind the chosen methods is outlined, along with all relevant formulations. The algorithms used are included in Appendix A.0.1.

2.1.1 Definition of non-singular orbital elements

An orbit propagator for a satellite is a critical tool used to model its trajectory through space. It predicts the satellite's future orbital parameters based on its current state and properties. Due to the influence of various perturbative forces, the satellite's orbital elements change over time. Different types of orbit propagators, including analytical and numerical models, offer varying trade-offs between accuracy, complexity, and computational demands. For this work, a numerical propagator was developed.

In general, determining the state of a satellite involves specifying its position and velocity at a given moment. This information can be derived from the position and velocity vectors or equivalent data. It is convenient to analyze the satellite's state using orbital parameters such as the Classical Orbital Elements (COE) (semi-major axis, orbital eccentricity, orbital inclination, argument of periapsis, right ascension of the ascending node, true anomaly), as this approach simplifies the problem. Indeed, a six-variable time-dependent problem is reduced to a one-variable time-dependent problem that can be solved analyti-

cally. In this case, only the true anomaly varies over time. Therefore, in the absence of perturbations, five of the orbital elements remain constant, with the true anomaly being the only element that changes, as it depends on the satellite's position vector.

Once the importance of using orbital elements has been explained, it is crucial to highlight an important aspect to avoid any misunderstandings: **the discussion will now proceed by using a particular set of orbital elements useful for trajectory analysis and optimization, namely the Modified Equinoctial Orbital Elements (MEE).** This choice was made because these elements are applicable to circular, elliptical, and hyperbolic orbits. The modified equinoctial equations do not exhibit singularities at zero eccentricity or at orbital inclinations of 0° and 90°, in contrast to the Classical Orbital Elements (COE). Therefore, delving into the relationship between the MEE and COE, the following equations are obtained [20] (a pseudocode for this is included in the Appendix A.0.1, named as Algorithm 1):

$$\begin{aligned} p &= a\left(1-e^2\right) \\ f &= e\cos(\omega+\Omega) \\ g &= e\sin(\omega+\Omega) \\ h &= \tan(i/2)\cos\Omega \\ k &= \tan(i/2)\sin\Omega \\ L &= \Omega+\omega+\theta \end{aligned} \tag{2.1}$$

Where:

p = semiparameter
a = semimajor axis
e = orbital eccentricity
i = orbital inclination
ω = argument of perigee
Ω = right ascension of the ascending node
θ = true anomaly
L = true longitude

Whereas the relationship [1] between the COE and MEE is as follows [20] (a pseudocode for this is included in the Appendix A.0.1, named as Algorithm 2):

[1] In these equations, the expression $\tan^{-1}(a,b)$ denotes a four-quadrant inverse tangent calculation.

$$
\begin{aligned}
a &= \frac{p}{1-f^2-g^2} \\
e &= \sqrt{f^2+g^2} \\
i &= 2\tan^{-1}\left(\sqrt{h^2+k^2}\right) = \tan^{-1}\left(2\sqrt{h^2+k^2}, 1-h^2-k^2\right) \\
\omega &= \tan^{-1}(g/f) - \tan^{-1}(k/h) = \tan^{-1}(gh-fk, fh+gk) \\
\Omega &= \tan^{-1}(k,h) \\
\theta &= L - (\Omega+\omega) = L - \tan^{-1}(g/f)
\end{aligned}
\tag{2.2}
$$

2.1.2 Perturbation models and orbit propagation

Once the type of orbital elements to be used in this work has been discussed, it is possible to focus on the concept of perturbations. To understand their effects on orbits, the impact of perturbations on the COE is first highlighted. However, all of this analysis is extendable to the MEE, as will be shown later on.

Keplerian orbits provide analytical solutions for the relative motion of two bodies, offering a simplified model of satellite motion. However, these orbits are idealized and differ from actual satellite trajectories due to various disturbances, known as perturbations. These perturbations arise from multiple sources, including the influence of a third body, the non-spherical shape of the central body, solar radiation pressure, and atmospheric drag. In particular, perturbations caused by the Earth's non-spherical shape, specifically its oblateness (where the polar radius is about 20 km less than the equatorial radius), are described by the numerical coefficient J_2. The J_2 perturbation is especially significant, as it represents the dominant effect in most orbits. Exceptions include very low orbits, where atmospheric drag is more influential, and very high orbits, where the J_2 effect becomes comparable to that of third-body influences.

When considering perturbations, the orbital elements are no longer constant due to the influence of various forces. Despite these perturbations, the orbits persist, largely due to the dominant Newtonian terms in the equations. However, all orbital elements experience either secular or non-secular variations. **In this investigation, by considering a satellite located in LEO, the most noteworthy perturbations arise from J_2 effects and atmospheric drag.**

Even though these considerations were made for the COE, the same applies to the MEE. In the absence of perturbations, 5 orbital elements (p,f,g,h,k) remain constant except for one (L). However, when considering a perturbed motion, all of them vary as follows [20]:

$$
\begin{aligned}
\dot{p} &= \frac{dp}{dt} = \frac{2p}{w}\sqrt{\frac{p}{\mu}}\Delta_t \\
\dot{f} &= \frac{df}{dt} = \sqrt{\frac{p}{\mu}}\left[\Delta_r \sin L + [(w+1)\cos L + f]\frac{\Delta_t}{w} - (h\sin L - k\cos L)\frac{g\Delta_n}{w}\right] \\
\dot{g} &= \frac{dg}{dt} = \sqrt{\frac{p}{\mu}}\left[-\Delta_r \cos L + [(w+1)\sin L + g]\frac{\Delta_t}{w} + (h\sin L - k\cos L)\frac{g\Delta_n}{w}\right] \\
\dot{h} &= \frac{dh}{dt} = \sqrt{\frac{p}{\mu}}\frac{s^2\Delta_n}{2w}\cos L \\
\dot{k} &= \frac{dk}{dt} = \sqrt{\frac{p}{\mu}}\frac{s^2\Delta_n}{2w}\sin L \\
\dot{L} &= \frac{dL}{dt} = \sqrt{\mu p}\left(\frac{w}{p}\right)^2 + \frac{1}{w}\sqrt{\frac{p}{\mu}}(h\sin L - k\cos L)\Delta_n
\end{aligned}
\tag{2.3}
$$

Where [20]:

$$
\begin{aligned}
\alpha^2 &= h^2 - k^2 \\
s^2 &= 1 + h^2 + k^2 \\
r &= \frac{p}{w} \\
w &= 1 + f\cos L + g\sin L
\end{aligned}
\tag{2.4}
$$

Moreover, Δ_r, Δ_t, and Δ_n represent non-two-body perturbations in the radial, tangential, and normal directions, respectively. The radial direction corresponds to the geocentric radius vector of the spacecraft, with positive values measured outward from the geocenter. The tangential direction is perpendicular to the radial vector, measured positive in the direction of the spacecraft's orbital motion. The normal direction is aligned with the angular momentum vector of the orbit, with positive values in the same direction. As previously mentioned, this work will focus solely on the two most significant perturbations related to the type of orbits studied: J_2 and atmospheric drag.

Therefore, for J_2 effects only, the components are as follows [20]:

$$
\begin{aligned}
\Delta_{J_{2_r}} &= -\frac{3\mu J_2 R_e^2}{2r^4}\left[1 - \frac{12(h\sin L - k\cos L)^2}{(1+h^2+k^2)^2}\right] \\
\Delta_{J_{2_t}} &= -\frac{12\mu J_2 R_e^2}{r^4}\left[\frac{(h\sin L - k\cos L)(h\cos L + k\sin L)}{(1+h^2+k^2)^2}\right] \\
\Delta_{J_{2_n}} &= -\frac{6\mu J_2 R_e^2}{r^4}\left[\frac{(1-h^2-k^2)\,(h\sin L - k\cos L)}{(1+h^2+k^2)^2}\right]
\end{aligned}
\tag{2.5}
$$

Where:

μ = gravitational constant
r = geocentric distance of the spacecraft
R_e = equatorial radius of the Earth

Whereas the radial, tangential, and normal perturbations due to aerodynamic drag are derived starting from the *ECI reference frame*[2] components, these are defined as follows:

$$\begin{aligned} \Delta_{D_i} &= -\frac{1}{2}\rho S C_D v v_i \\ \Delta_{D_j} &= -\frac{1}{2}\rho S C_D v v_j \\ \Delta_{D_k} &= -\frac{1}{2}\rho S C_D v v_k \end{aligned} \tag{2.6}$$

Where:

ρ = atmospheric density
S = aerodynamic reference area
C_d = drag coefficient
v_i, v_j, v_k = components of the relative velocity.
v = velocity magnitude

The components of the relative velocity (v_i, v_j, v_k) are computed as the difference between the velocity components in the ECI (Earth-Centered Inertial) frame and those of the Earth's rotational velocity. By considering the atmosphere as rotating with the Earth, the velocity and its components change accordingly.

Before proceeding, it is important to highlight how to compute the velocity components in the ECI (Earth-Centered Inertial) frame starting from the MEE (a pseudocode for this is included in the Appendix A.0.1, named as Algorithm 3) [20]:

$$\mathbf{v} = \begin{bmatrix} -\frac{1}{s^2}\sqrt{\frac{\mu}{p}}\left(\sin L + \alpha^2 \sin L - 2hk\cos L + g - 2fhk + \alpha^2 g\right) \\ -\frac{1}{s^2}\sqrt{\frac{\mu}{p}}\left(-\cos L + \alpha^2 \cos L + 2hk \sin L - f + 2ghk + \alpha^2 f\right) \\ \frac{2}{s^2}\sqrt{\frac{\mu}{p}}(h\cos L + k \sin L + fh + gk) \end{bmatrix} \tag{2.7}$$

[2]The Earth-Centered Inertial (ECI) frame is often used interchangeably with the geocentric frame, IJK. This system originates at the center of the Earth. The fundamental plane aligns with the Earth's equator, where the I axis points toward the vernal equinox, the J axis lies 90° east within the equatorial plane, and the K axis extends through the North Pole.

Going back to the component of the drag, once those in the ECI frame were computed, the transformation of the drag vector from the Earth-Centered Inertial (ECI) frame to the Radial-Circumferential-Normal (RCN) frame is performed (a pseudocode for this is included in the Appendix A.0.1, named as Algorithm 4):

The Rotation matrix is defined as follows:

$$R = \begin{bmatrix} R_{11} & R_{12} & R_{13} \\ R_{21} & R_{22} & R_{23} \\ R_{31} & R_{32} & R_{33} \end{bmatrix}$$

Where:

$$R_{11} = \cos(\text{AOP})\cos(\text{RAAN}) - \sin(\text{AOP})\sin(\text{RAAN})\cos(\text{INC}) \tag{2.8}$$

$$R_{12} = \cos(\text{AOP})\sin(\text{RAAN}) + \sin(\text{AOL})\cos(\text{RAAN})\cos(\text{INC}) \tag{2.9}$$

$$R_{13} = \sin(\text{AOP})\sin(\text{INC}) \tag{2.10}$$

$$R_{21} = -\sin(\text{AOP})\cos(\text{RAAN}) - \cos(\text{AOP})\sin(\text{RAAN})\cos(\text{INC}) \tag{2.11}$$

$$R_{22} = -\sin(\text{AOP})\sin(\text{RAAN}) + \cos(\text{AOP})\cos(\text{RAAN})\cos(\text{INC}) \tag{2.12}$$

$$R_{23} = \cos(\text{AOP})\sin(\text{INC}) \tag{2.13}$$

$$R_{31} = \sin(\text{RAAN})\sin(\text{INC}) \tag{2.14}$$

$$R_{32} = -\cos(\text{RAAN})\sin(\text{INC}) \tag{2.15}$$

$$R_{33} = \cos(\text{INC}) \tag{2.16}$$

Thus, the drag in the RCN reference frame is obtained:

$$D_{\text{RCN}} = RD_{\text{ECI}} \tag{2.17}$$

As the definitions of radial, tangential, and normal perturbations due to aerodynamic drag include atmospheric density, a specific choice of density model was made: **the NRLMSISE-00 model**. NRLMSISE-00 is an empirical global atmospheric model that spans from the Earth's surface to space, providing data on the temperatures and densities of atmospheric constituents. One of its key applications is predicting satellite orbital decay due to atmospheric drag. By using the *Python programming language* [21], a specific function, `nrlmsise00` [22], is included to compute the atmospheric density based on the selected model. The inputs of this function are:

- Date to compute the model;
- Altitude [km];
- Geodetic latitude [deg];
- Longitude [deg];
- 10.7-cm solar flux [sfu];
- 10.7-cm averaged solar flux, 90-day centered on input time [sfu];
- Magnetic index.

Thus, when considering a satellite deorbiting, we expect the atmospheric density to vary with altitude for several reasons, including changes in latitude and longitude. **A function to account for the variations in both latitude and longitude was required.** The steps followed to achieve this are outlined below:

- First, it was necessary to compute the position in the ECI reference frame. from the Modified Equinoctial Elements (MEE). In *Python* [21], the following equations were implemented in a function (a pseudocode for this is included in the Appendix A.0.1, named as Algorithm 5):

$$\mathbf{r} = \begin{bmatrix} \frac{r}{s^2}\left(\cos L + \alpha^2 \cos L + 2hk \sin L\right) \\ \frac{r}{s^2}\left(\sin L - \alpha^2 \sin L + 2hk \cos L\right) \\ \frac{2r}{s^2}(h \sin L - k \cos L) \end{bmatrix} \tag{2.18}$$

- A function that computes the Greenwich Mean Sidereal Time (GMST) in radians was implemented (a pseudocode for this is included in the Appendix A.0.1, named as Algorithm 6).

- A function that computes the rotation in the x-y plane to convert the position vector from the ECI (Earth-Centered Inertial) frame to the ECEF (Earth-Centered, Earth-Fixed) frame was implemented (a pseudocode for this is included in the Appendix A.0.1, named as Algorithm 7). This transformation is accomplished by rotating the ECI coordinates by the Greenwich Mean Sidereal Time (GMST) (θ_{GMST}) around the z-axis. The rotation matrix for this transformation is:

$$R_{\text{ECI to ECEF}}\ (\theta_{\text{GMST}}) = \begin{bmatrix} \cos(\theta_{\text{GMST}}) & \sin(\theta_{\text{GMST}}) & 0 \\ -\sin(\theta_{\text{GMST}}) & \cos(\theta_{\text{GMST}}) & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{2.19}$$

- At this point, a procedure was implemented to convert a position vector in the ECEF frame to the corresponding latitude and longitude (a pseudocode for this is included in the Appendix A.0.1, named as Algorithm 8). The process is illustrated in Figure 2.2.

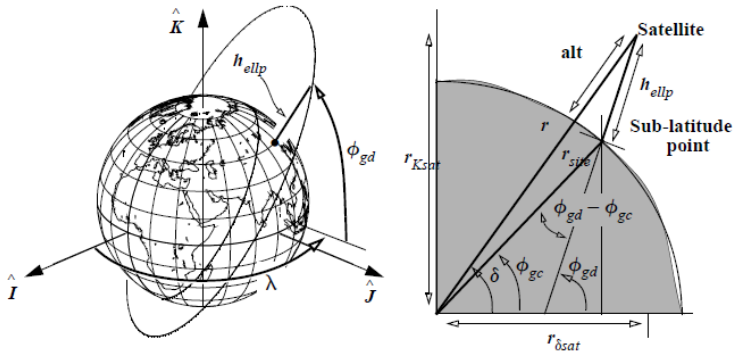


Figure 2.2: Determining a Satellite's Sub-latitude Point [23]

The method employed is based on spherical trigonometry [23]. The equatorial projection of the satellite vector is:

$$r_{\delta_{sat}} = \sqrt{r_I^2 + r_J^2} \tag{2.20}$$

Where r_I and r_J are the first and second components of the position vector expressed in ECEF.

To compute the **longitude**, it is sufficient to calculate the right ascension, as the vector is Earth-fixed:

$$\text{SIN}(\alpha) = \frac{r_J}{r_{\delta_{sat}}} \tag{2.21}$$

Concerning the **geodetic latitude**, the angle between the equatorial plane and the normal to the surface of the ellipsoid, an iterative process is typically required; the *Astronomical Almanac* method is applied. The first guess is $\phi_{gd} = \delta$:

$$\mathrm{TAN}(\delta) = \frac{r_K}{r_{\delta_{sat}}} \tag{2.22}$$

Where r_K is the third component of the position vector expressed in ECEF.

The iterative loop proceeds as follows and continues until the difference between the current and previous latitude values is smaller than a specified tolerance:

$$\begin{aligned} C &= \frac{R_{\mathrm{Earth}}}{\sqrt{1 - e^2 \sin^2(\phi_{gd})}} \\ \tan(\phi_{gd}) &= \frac{r_K + Ce^2 \sin(\phi_{gd})}{r_{\delta_{sat}}} \end{aligned} \tag{2.23}$$

Where e is the Earth's eccentricity.
At the end of this process the latitude and longitude are actually found.

2.1.3 Numerical Simulations

Once the entire process of creating the propagator has been explained, it is possible to present a result for a case that includes drag and J2 as perturbations. The studied case for this example is the following and the result of the implemented propagator is shown in Figure 2.3.

`"Initial coe"` : $[6891\mathrm{km}, 0.001, 97.45169\mathrm{deg}, 250.3635\mathrm{deg}, 120.0\mathrm{deg}, 0.0]$

Parameter	Value
mass	10.0 kg
area	0.06
Cd	2.2
F10_7a	150
F10_7	150
Ap	4
Drag	True
J2	True
Final time	5 years
Thrust	300.0 mN
ISP	200.0
Firing time	0.0 minuti

Table 2.1: Propagator Input

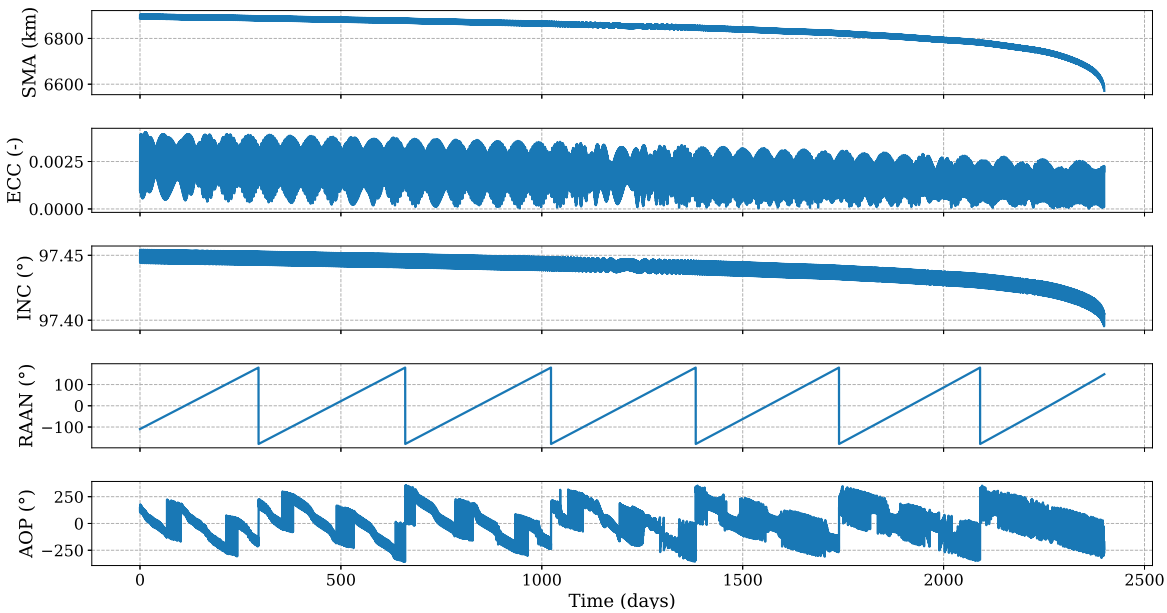


Figure 2.3: The evolution over time of the COEs for a deorbiting case run with the propagator, considering both J2 and drag perturbation.

In this graph, the firing time is set to zero. However, in Chapter 3, thrust arcs will be considered by studying three different propulsion systems in an optimization problem, where the firing time is used as a design variable.

To comprehend the graphs representing the COEs, a thorough examination of the pertur-

bations is necessary. As previously mentioned, perturbations induce variations in the COEs, and it is now possible to identify the specific types of changes occurring.

In general, the J_2 perturbation is known to induce secular changes in the right ascension of the ascending node and the argument of perigee. Conversely, eccentricity, semi-major axis, and inclination typically remain relatively constant, apart from short-periodic perturbations. Atmospheric drag, on the other hand, acts counter to the satellite's velocity vector, causing a significant reduction in the semi-major axis—a change of considerable magnitude.

Consequently, this analysis highlights the changes in orbital parameters. As expected, the semi-major axis decreases primarily due to drag, while eccentricity and inclination exhibit short-periodic perturbations resulting from J2. The right ascension of the ascending node undergoes secular variation. Finally, the peculiar behavior of the argument of periapsis is attributed to its undefined nature in a circular orbit (as this case considers a quasi-circular orbit).

A double-check of the results was performed using a tool from the company where the master's thesis was conducted.

2.2 Collision Probability Computation

Assessing the likelihood of a collision with space debris is an essential step in planning deorbiting missions.

The probability of collision, (P), between two objects can be computed by integrating the normalized Gaussian probability density function associated with the position covariance, $(f(x, y, z))$, over the collision volume:

$$P = \iiint_V f(x, y, z) dx\ dy\ dz, \tag{2.24}$$

where the objects are typically modeled as spheres [24]. This formulation provides a general framework for evaluating the likelihood of a collision between two space objects, and it is crucial for accurately estimating potential conjunction events.

However, for design applications—particularly in the context of long-duration missions with numerous and uncertain debris encounters—a flux-based approach is preferred.

The flux method approximates debris populations and their temporal evolution, which depend on factors such as launch activity, satellite operations, and the natural decay of orbital objects. One of the most widely used models in this context is MASTER [25], which provides spatial and temporal debris flux data for different object size classes.

Ultimately, the cumulative probability of collision, (p_c), is evaluated using the trajectory

history, i.e., the time evolution of the state vector $\boldsymbol{y}(t)$, by integrating the instantaneous probability over time:

$$p_c = \int_{t_0}^{t_f} P(t, \boldsymbol{y})\ dt. \tag{2.25}$$

Therefore, the probability of collision turns out to be a function of both time and trajectory.

Based on that in the context of this master's thesis, various approaches were examined to assess the probability of collision during the deorbiting phase. Initially, the use of MASTER (Meteoroid and Space Debris Terrestrial Environment Reference) [25] was considered, starting from the population flux with the intention of transitioning to the probability of collision. However, the analysis would have needed to be conducted orbit by orbit while decreasing altitude, which proved to be inefficient. Additionally, the requirement to select a discretization step for the descent orbit would have compromised the accuracy of the prediction.

Subsequently, NASA's tool ORDEM (Orbital Debris Engineering Model) [26] was attempted, but the issue persisted.

Then, it was decided to model the flux of debris, and a literature review was conducted. However, once a model applying the Petri Net Model[3][27] was identified and the algorithms were implemented, there were no means to double-check the results because some of the input data used in the paper were not explicitly outlined. Moreover, certain assumptions made in the referenced paper, such as neglecting the Right Ascension of the Ascending Node (RAAN), were not feasible for a thesis focused on a design-oriented approach.

Finally, DRAMA (Debris Risk Assessment and Mitigation Analysis) [28] was selected, with the potential use of pyDRAMA [29], focusing particularly on the ARES (Assessment of Risk Event Statistics) module [30]. This tool provides users with the annual probability of collision along the selected orbit. Consequently, the author considered the possibility of extending this functionality to include decay, as will be explained in the present chapter.

2.2.1 Overview of DRAMA Software and the ARES Module

DRAMA (Debris Risk Assessment and Mitigation Analysis) is a software developed by ESA to evaluate the compliance of space missions with international requirements related to space debris throughout a mission's operational and disposal phases. Notably, by

[3]Petri nets are both mathematical and graphical tools designed for modeling and simulating the behavior of various systems.

defining a space mission, a series of tools can be used:

- Assessment of Risk Event Statistics (ARES)
- MASTER-based Impact Flux and Damage Assessment Software (MIDAS)
- Orbital Spacecraft Active Removal (OSCAR)
- Cross-section of Complex Bodies (CROC)
- (Re-entry) Survival and Risk Analysis (SARA)

Each of these tools has a specific field of application. For the purpose of this thesis, ARES was the one of particular interest. It has four functionalities:

- Annual collision probability due to the whole population
- Avoidance Schemes Assessment
- Required Delta-V to execute the maneuvers
- Required Propellant for that Delta-V

Some key assumptions made in ARES, which outline the approach followed in all this chapter, are as follows:

- In ARES, every spacecraft is assumed to have a spherical shape, with its size defined by the parameter Spacecraft Radius (measured in metres). [31]
- The Target Orbit (Single Averaged Elements) is a simplified and averaged description of the spacecraft's orbit, which does not account for short-term oscillations or variations caused by perturbations. Moreover, this orbit is characterized by the semi-major axis, eccentricity, inclination, right ascension of the ascending node, and argument of perigee (measured in kilometres for the semi-major axis and degrees for angular components). The True Anomaly is not included, as position resolution is unnecessary for the annual statistics generated by ARES. [31]

As previously mentioned, there is an interface to the DRAMA modules from Python. This module requires inputs for performing detailed analyses. These inputs enable users to define the mission parameters and spacecraft properties necessary for computations. Below is a list of key inputs used in this Master's Thesis:

- **Minimum Particle Size**

- **Maximum Particle Size**
- **Spacecraft Radius**: Specifies the radius of the spacecraft, based on the cross-sectional area, and assumed to be spherical for simplification.
- **Semi-Major Axis**
- **Eccentricity**
- **Inclination**
- **Right Ascension of the Ascending Node**
- **Argument of Perigee**
- **Analysis Reference Date**: Specifies the reference date for the analysis.

These inputs allow users to define the spacecraft's characteristics and orbital parameters, facilitating the computation of collision probabilities, avoidance schemes, delta-V requirements, and propellant needs, as described in the previous section.

2.2.2 Cumulative probability of collision along the deorbiting trajectory

As previously mentioned, ARES provides the user with the Annual Collision Probability along the considered orbit. Notably, the target orbit is identified using Single Averaged Elements [31]. This is why the output of the propagator, which serves as the input for ARES, is not a series of osculating elements but rather **averaged elements**.
An important point is how the Annual Collision Probability (ACP) linked to the *whole population* is computed in ARES, to better understand the input required for the code, which will be explained later. The Annual Collision Probability (ACP) due to the *whole population* is computed as follows [32]:

$$ACP = \sum_{j=1}^{n} F_j \cdot \pi \cdot (R_{sc} + r_j)^2 \tag{2.26}$$

Here, F_j represents the annual flux provided by MASTER [25], corresponding to the current population group, R_{sc} is the spacecraft radius (referring to the cross-sectional area [33]), and r_j is the size of the corresponding debris element.

Therefore, in the code that computes the global collision probability along the deorbiting trajectory, the input includes not only the Semi-major Axis, Eccentricity, Inclination, Right

Ascension of the Ascending Node, Argument of Perigee, and Epoch (parameters required by the system), but also the minimum and maximum particle sizes to retrieve the flux from MASTER. Additionally, to compute the probability, another necessary input is the spacecraft radius.

An important observation is that ARES computes the annual collision probability for a nominal orbit, fundamentally assuming that this orbit is maintained (for example via station-keeping), hence does not cover evolving orbits, such as in the case of a decaying or deorbiting satellite.

Building onto the concepts and observations from the previous subsection, this work proposes an improved and efficient methodology for computing the cumulative collision probability along the deorbiting trajectory using ARES.

In this Master's Thesis, an idea was proposed based on the *assumption* that **the collision probability remains linear over time** along the same orbit. This implies that the integral of the annual collision probability can be divided uniformly over time. Based on this, using the annual collision probability, the procedure involved normalizing that value for the time spent over a specific orbit during the decay trajectory.

Due to the high computational cost of using DRAMA, it was decided to **interpolate the results** of the propagator (a discussion on the type of interpolation will be provided later). The interpolation was performed for the Classical Orbital Elements (COEs) and the epoch. As will be shown, discretization was also applied, both in terms of Semi-Major Axis (SMA) and time, to facilitate comparisons and to find the most interesting discretization approach.

The first step was to compute the annual collision probability for each orbit in the interpolated vector (ACP_k). Once these values were obtained, the Interval Collision Probability (a term coined during this work) was calculated as the annual collision probability along an orbit multiplied by the time spent at that orbit, determined by the difference in epochs (the previous and subsequent). This result was then divided by 365*24*3600. At the end of this computation, a series of Interval Probabilities was determined for each interpolated orbit of the decay trajectory. The final step was to sum all these values to obtain the Global Probability of Decay. The description of this procedure is shown in the Figure 2.4 and in the following equations (a pseudocode for this is included in the Appendix A.0.1, named as Algorithm 9) :

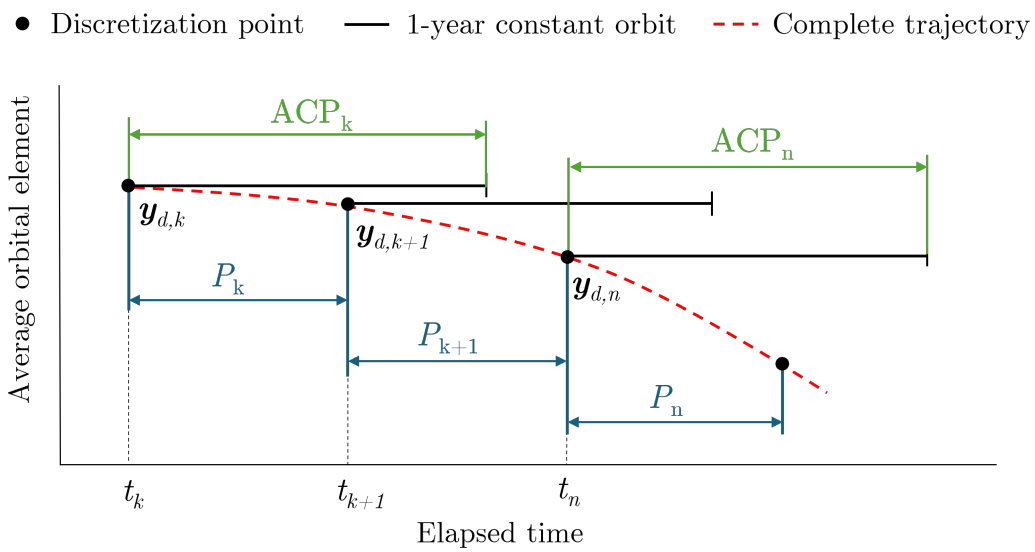


Figure 2.4: Schematic of discretization.

1. **Time between two epochs [s]:**

$$t_k = (\text{all\_epoch}_{k+1} - \text{all\_epoch}_k) \tag{2.27}$$

for each $k \in \{0, \ldots, n-1\}$, where

$$n = \text{len}(ACP_k) \tag{2.28}$$

Note: The ACP_k are defined as the annual collision probabilities computed by ARES along each orbit of the interpolated vector.

2. **Interval probabilities:**

$$\text{interval\_probability}_k = \frac{ACP_k \cdot t_k}{365 \cdot 24 \cdot 3600} \tag{2.29}$$

for each $k \in \{0, \ldots, n-1\}$.

3. **Global probability of collision (GPC) along the decay:**

$$\text{global\_probability\_decay} = \sum_{k=0}^{n-1} \text{interval\_probability}_k \tag{2.30}$$

2.2.3 An analysis of the effects of discretization

As previously mentioned, the results of the propagator were interpolated (The interpolation was performed for the Classical Orbital Elements (COEs) and the epoch) to reduce the high computational cost of using ARES: the higher the number of interpolation points, the lower the computational speed. The interpolated data, representing the discretized re-entry trajectory, serve as the input for the ARES code.

In this subchapter, the effects of discretization are studied. Specifically, by changing the discretization, the global probability along the decay also varies. This analysis is conducted to determine the optimal **type of discretization**, both in terms of *time* and *semi-major axis*, for each considered case and to identify the ideal **number of points** to be used. Once the values stabilize, the number of points is established to obtain a reliable estimation of the global probability decay.

This study is performed for different cases: a satellite released by the International Space Station (ISS) and a satellite that starts deorbiting from a Sun-Synchronous Orbit (SSO).

At this stage, the general parameters common to the described cases are presented in the following table:

Parameter	**Value**
F10_7a	150
F10_7	150
Ap	4
Drag	True
J2	True
Final time	10 years
Thrust	300.0 mN
ISP	200.0
Firing time	0.0 minuti
Minimum Particle Size	0.01 m
Maximum Particle Size	100 m

Table 2.2: General Parameters

Not only there are common and general parameters, but also there are specific data related to the studied cases. Indeed, two different types of spacecraft were analyzed in the SSO and ISS scenarios. *For these spacecraft, the drag area and the cross-sectional area are assumed to be equal by considering a simplified case without any deployable solar panel.*

Regarding the SSO case the chosen data are as follows:

Case	Type of Discretization	Satellite (CD, Area, Mass)	Orbit	Initial Epoch
DT00_SSO_	Time	2.2, 0.06 m^2, 10 kg	SSO	2000
DS00_SSO_	SMA	2.2, 0.06 m^2, 10 kg	SSO	2000

Table 2.3: Type of cases studied for the SSO

Regarding the case of the ISS orbit the chosen data are as follows:

Case	Type of Discretization	Satellite (CD, Area, Mass): 3U	Orbit	Initial Epoch
DT00_ISS_	Time	2.2, 0.03 m^2, 5 kg	ISS	2000
DS00_ISS_	SMA	2.2, 0.03 m^2, 5 kg	ISS	2000

Table 2.4: Type of cases studied for the ISS orbit

CASE: SSO

All the data used in this case are those included in Table 3.1 and Table 2.3. Additionally, in order to compute the annual collision probability, the cross-sectional area was also needed. As previously mentioned, it was taken to be equal to the drag area, which is 0.06 m^2.

While processing this case, it was decided to address this part by studying two different initial orbits, both belonging to the Sun-Synchronous Orbit (SSO) category.

The **first SSO examined was at an altitude of 350 km.** The objective was to study a case where the altitude remained almost constant over a smaller time range. This approach was motivated by the fact that a weighted sum is used in the calculations; therefore, the longer the time spent in a specific condition, the greater its impact on the results. Therefore with this first approach a faster decay was studied. The characteristics of this first SSO are: `"Initial coe"`: [6721 km, 0.001, 96.8247 deg, 250.3635 deg, 120.0 deg, 0.0]

In order to provide a clear understanding of how the discretization actually works, a plot of the time-based discretization is shown in Figure 2.5.

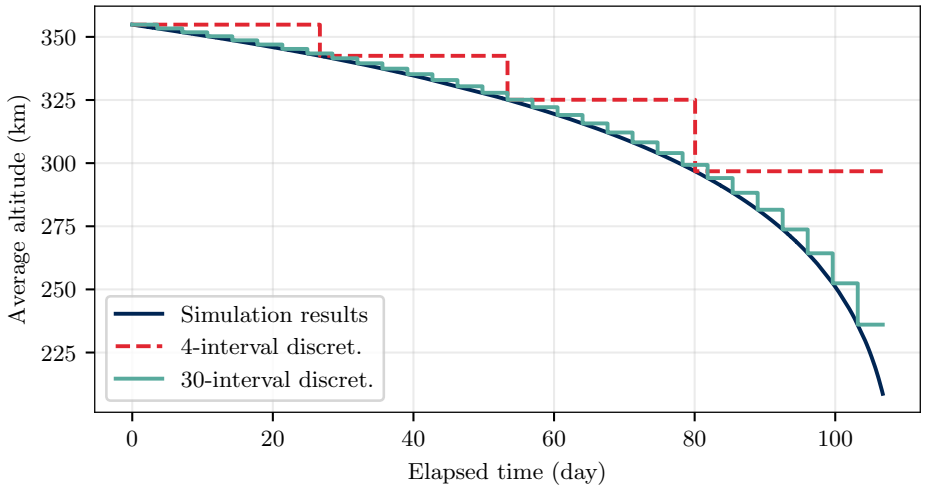


Figure 2.5: Time Discretization for an Initial Sun-Synchronous Orbit (SSO) at 350 km Altitude

The higher the number of chosen intervals, the better the decay trajectory is approximated, whereas considering fewer intervals results in assuming the COEs remain constant over a longer period of time. However, this comes at the cost of reduced computational speed (as shown in Figure 2.6), as a higher number of intervals increases the number of inputs given to ARES.

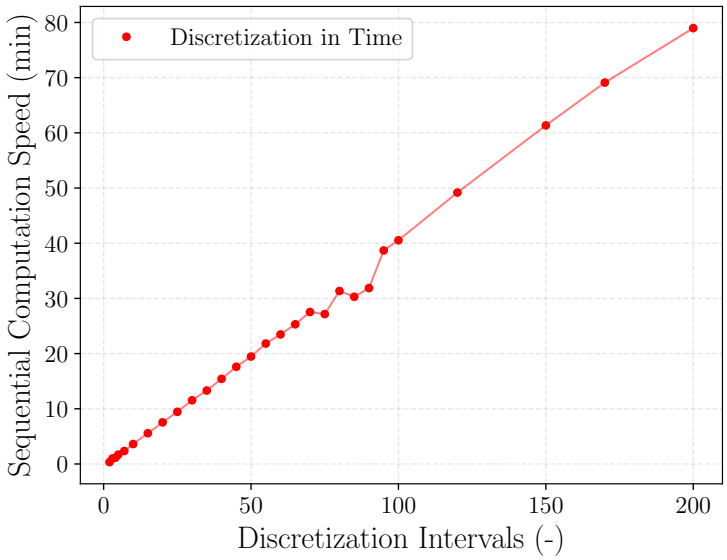


Figure 2.6: Computational Cost of Time Discretization for an Initial Sun-Synchronous Orbit (SSO) at a 350 km Altitude

Therefore, it is crucial to determine an optimal number of intervals that provides a sufficiently accurate value for the Global Probability of Collision (GPC) along the decay trajectory. This rationale underlies the following plots, which present the results in terms of GPC for different numbers of intervals and for both discretization in time and semi-major axis (SMA). Notably, Figure 2.7 represents, on the y-axis, the error with respect to a reference value obtained in a case where a very high number of intervals was considered (n=200). The goal of this approach was to identify the number of intervals at which the error stabilized. This number of intervals, which in the case of an SSO is 40, is the one used in the optimization process.

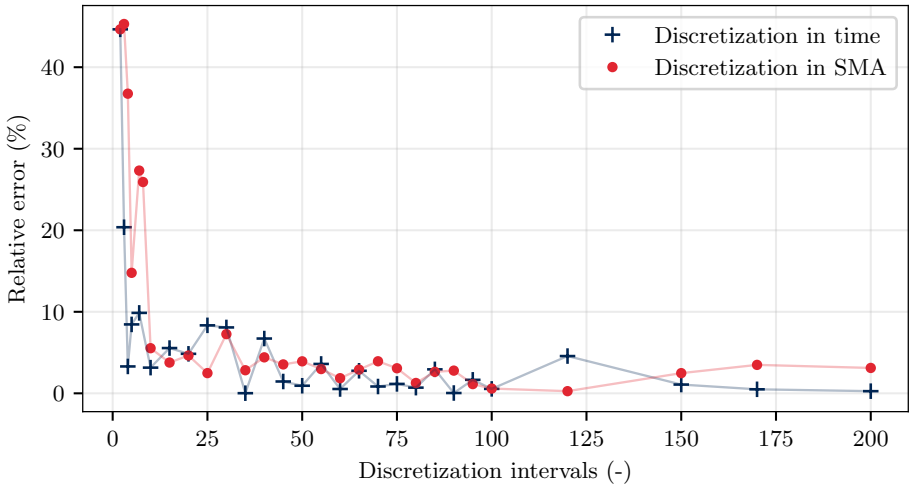


Figure 2.7: Comparison of Relative Error (with respect to a case with 200 intervals) Across Discretization Intervals for Time and SMA in the Case of an SSO at 350 km Altitude

Even though the behavior of the relative error stabilizes at around 40 intervals, small oscillations were still present. In an effort to understand this behavior, a study of the annual collision probability (which is used to compute the Global Probability of Collision (GPC), and therefore directly affects its behavior) was conducted for both the cases with 100 intervals and 4 intervals. The aim was to determine whether any oscillations in the annual collision probability could, in some way, justify the oscillations observed in the global probability of decay. Figure 2.8 illustrates the complexity of the studied field. By increasing the number of intervals, the oscillations in the values of the annual collision probability (computed for each orbit) are more closely followed. Conversely, with a smaller number of intervals, an averaging effect is observed, which smooths out the oscillations. Therefore, a sufficient number of intervals is needed to avoid losing the trend of the probability. But anyway, this result justifies the behavior of the computed GPC.

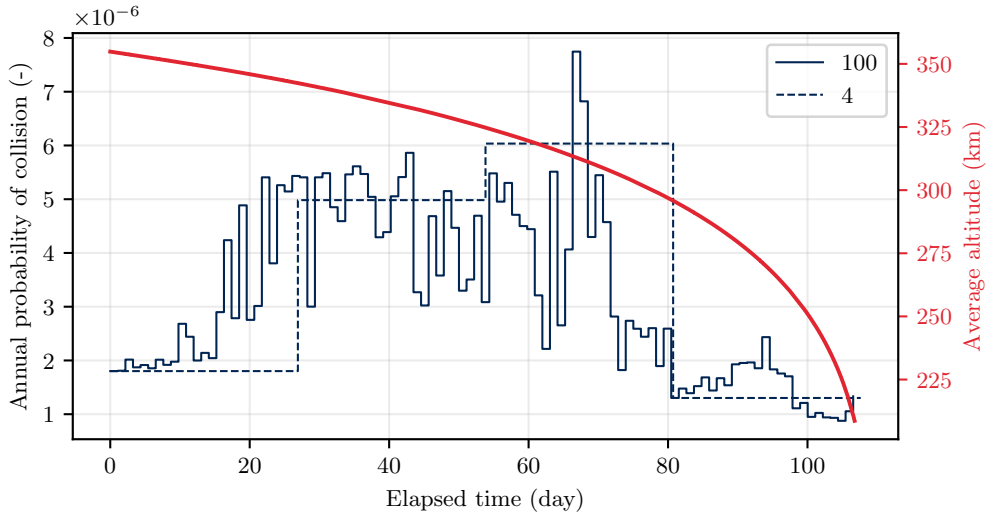


Figure 2.8: Annual Collision Probability and Average Altitude for 100 and 4 Discretization Intervals in the Case of an SSO at 350 km Altitude

The **second SSO examined was at an altitude of 520 km.** The characteristics of this second SSO are: `"Initial coe"`: [6891km, 0.001, 97.45169deg, 250.3635deg, 120.0 deg, 0.0]

Also in this case, a study of the relative error of the Global Probability of Collision (GPC) along the decay trajectory was conducted for different numbers of intervals and different type of discretization. By examining Figure 2.9, a number of 40 intervals was chosen, as this is where the relative error stabilizes.

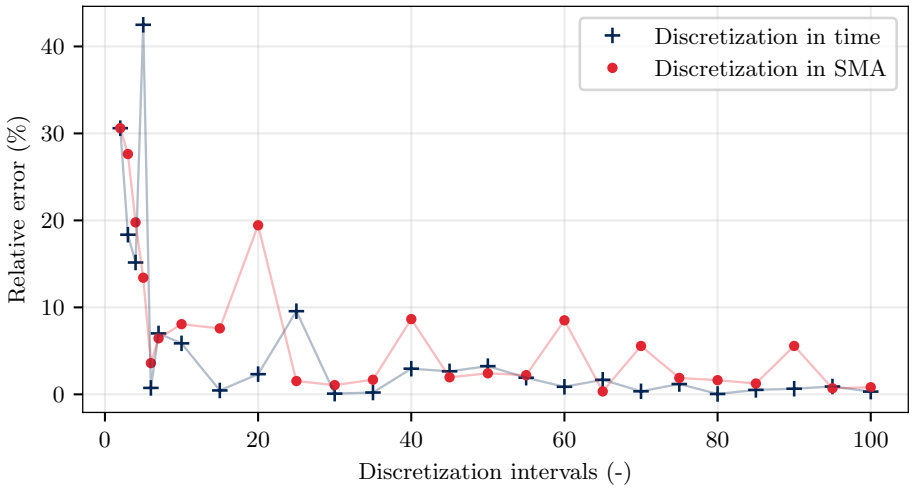


Figure 2.9: Comparison of Relative Error (with respect to a case with 200 intervals) Across Discretization Intervals for Time and SMA in the Case of an SSO at 520 km Altitude

By comparing Figure 2.7 and Figure 2.9, the difference in initial altitude becomes evident. It is clear that the higher the altitude, the greater the difference between the discretization in time and the discretization in SMA. This occurs because, when discretizing in SMA, there will be more interval points concentrated in the final part of the decay trajectory, where the altitude changes most significantly. In contrast, in the case of time-based discretization, the points are uniformly distributed along the entire decay trajectory. Therefore, for the SSO at 350 km altitude, the SMA-based discretization intervals located in the part of the decay trajectory where the altitude varies the least are more prevalent compared to the other SSO. This makes the discretization in SMA for the first case much more similar to the one based on time.

This analysis also highlights that, for a deorbiting case aimed at studying the global probability of collision along the decay trajectory, using a time-based discretization is likely more accurate. Therefore, the time-based discretization approach will be chosen for the optimization step.

Additionally, by analyzing all these results, it becomes evident that using a larger number of intervals, compared to 1 or 2 intervals, is crucial. With 40 intervals in both cases, the error decreases by approximately 30–40 %.

CASE: ISS

At this stage another case was studied. Notbaly the one of a satellite released from the ISS. The characteristics of the ISS orbit were derived from the TLE data: `"Initial coe"`: [6793.382 km, 0.00031040, 51.6377deg, 296.2827 deg, 141.8447 deg, 0.0]

Regarding the study of the relative error of the Global Probability of Collision, the result is shown in Figure 2.10. Also in this case, the results stabilize around 40 intervals.

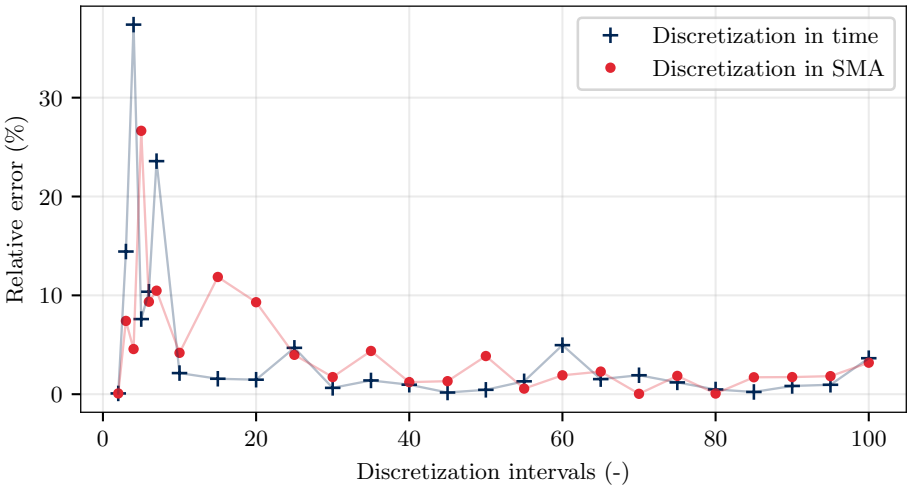


Figure 2.10: Comparison of Relative Error Across Discretization Intervals for Time and SMA in the Case of an ISS

2.3 Impact of Epoch Evolution, and Cross-Sectional Area on Global Collision Probability

Once the analysis of the discretization was completed, and after understanding that a discretization in time is better suited for the deorbiting case, which focuses on the computation of the probability of collision, an analysis of different drag values and cross-sectional areas was also conducted, along with different initial epochs.

All these studies were also carried out for the case of the SSO with an initial altitude of 520 km. Additionally, the discretization in time was set to 40 intervals, at which point the error values appeared to stabilize. The characteristics of this SSO are again: `"Initial coe"`: `"Initial coe"`: [6891km, 0.001, 97.45169deg, 250.3635deg, 120.0 deg, 0.0]

The epoch study was conducted by fixing all the other parameters as specified in Table 2.5:

Parameter	**Value**
CD	2.2
Drag Area	0.06 m^2
Cross sectional area	0.06 m^2
Mass	10 kg
F10_7a	150
F10_7	150
Ap	4
Drag	True
J2	True
Final time	10 years
Thrust	300.0 mN
ISP	200.0
Firing time	0.0 minuti
Minimum Particle Size	0.01 m
Maximum Particle Size	100 m

Table 2.5: General Parameters

The results in terms of the global probability of collision are shown in the Figure 2.11. As expected, over the years, the probability of collision along a decaying trajectory increases, primarily due to the rising amount of space debris in orbit. However, from 2020 to 2030, a decrease is observed. Since all these results are provided by an ESA tool, this trend could be attributed to a reduction in the amount of debris due to the implementation of active debris removal measures in the coming years as part of the Zero Debris approach.

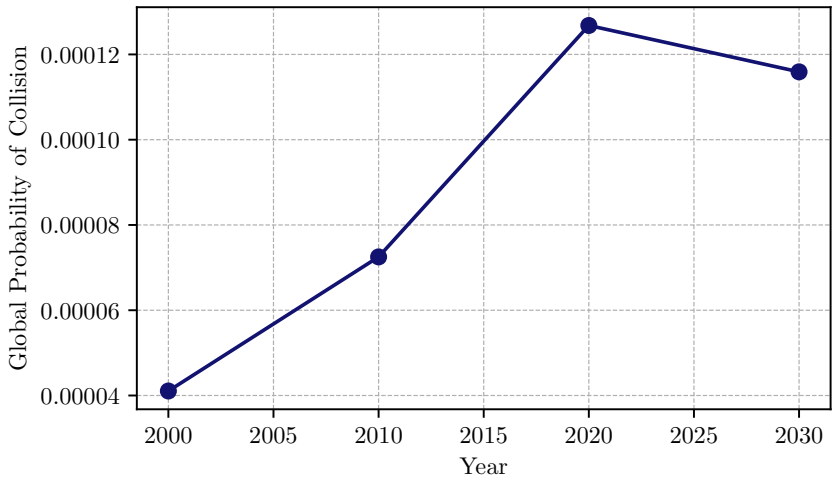


Figure 2.11: Evolution of the Global Collision Probability along years

Regarding the discussion on the impact of both the drag area and the cross-sectional area, as previously mentioned in this chapter, they are considered equal. This assumption is made because a simplified spacecraft model was used, without any deployable solar panels. Moreover, the mass is assumed varying proportionally to the area, given that for a drag area of 0.06 m^2 a mass of 10 kg was considered. In this case, the data inserted into the propagator are as in Table 2.6:

Parameter	**Value**
CD	2.2
Initial Epoch	2000
F10_7a	150
F10_7	150
Ap	4
Drag	True
J2	True
Final time	10 years
Thrust	300.0 mN
ISP	200.0
Firing time	0.0 minuti
Minimum Particle Size	0.01 m
Maximum Particle Size	100 m

Table 2.6: General Parameters

The results in terms of the global probability of collision are shown in the Figure 2.12. As expected, the higher the cross-sectional area, the greater the area exposed to possible collisions, leading to a higher GPC.

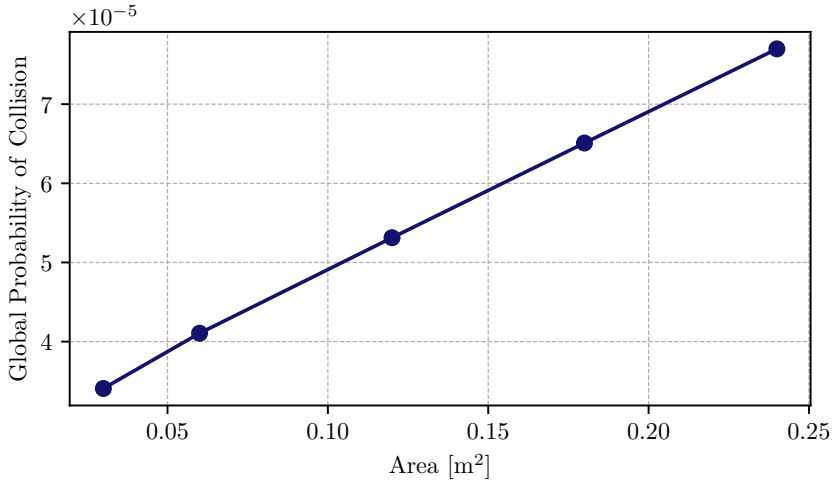


Figure 2.12: Evolution of the Global Collision Probability with area

2.4 Optimization Problem

The optimization problem, whose results are presented in the following chapter, is formulated as a control problem aimed at minimizing a set of performance indicators, while ensuring compliance with the system dynamics and boundary constraints.

Mathematically, the problem is defined as:

$$\begin{aligned} \textit{minimize} \quad & \boldsymbol{J}, \\ \textit{with respect to} \quad & \boldsymbol{u} \in U \\ \textit{subject to} \quad & \dot{\boldsymbol{y}}(t) = f_d(t, \boldsymbol{y}(t), \boldsymbol{u}(t)) \\ & \boldsymbol{y}(t_0) - \boldsymbol{\psi}_0 = 0, \\ & \boldsymbol{y}(t_f) - \boldsymbol{\psi}_f = 0, \end{aligned} \tag{2.31}$$

where $\boldsymbol{y}$ represents the state of the dynamical system, and $\boldsymbol{u}$ the controls, which are the design variables. The objective vector $\boldsymbol{J}$ includes the quantities to be minimized, which may comprise combinations of the total velocity increment (ΔV), propellant mass (m_p), decay duration (t_f) and the probability of collision throughout the decay (p_c).

The system dynamics are governed by the function f_d, while the initial and final conditions are enforced through the boundary vectors $\boldsymbol{\psi_0}$ and $\boldsymbol{\psi_f}$, respectively.

2.5 Guidance policy

The optimization problem defined in (2.31) seeks to determine the time-varying control inputs $\boldsymbol{u}(t)$, constrained within the admissible set U, that minimize a given set of objective functions. In this work, the following guidance policy is employed:

$$\alpha^* = \pi + arctan\left(\frac{e\sin(\theta)}{1+e\cos(\theta)}\right), \quad \beta^* = 0, \tag{2.32}$$

in which α^* is the optimal in-plane angle, β^* the optimal out-of-plane angle, e the orbit eccentricity and θ the current true anomaly.

This guidance policy is based on the principle that, to effectively reduce the semi-major axis, the thrust should be applied in-plane and in the direction opposite to the velocity vector. This configuration represents the optimal control strategy for minimizing the time required to achieve variations in the semi-major axis, as well as to satisfy the desired final conditions.

Given the adoption of this guidance law and the imposed constraints, the firing duration—denoted by t_{thr}—becomes the sole remaining design variable of the optimization problem.

The evaluation of the objective functions defined in (2.31) is performed through numerical integration, followed by post-processing of the resulting trajectory data. One of the key outputs is the propellant mass (m_p):

$$m_p = m(t_0) - m(t_f), \tag{2.33}$$

whereas the total velocity increment, ΔV —which is fundamentally related to m_p through the Tsiolkovsky rocket equation— can be computed accordingly. The decay time, (t_f), corresponds to the time at which the terminal conditions are satisfied, i.e., when the re-entry boundary is reached.

Chapter 3

Results and Analysis

3.1 Simulation Scenario

Once a series of cases was studied with the purpose of understanding the behavior of the global probability of collision, as well as finding a considerable number of intervals discretizing the trajectory of decay to obtain the most accurate possible approximation of the global probability of collision along the decay, the work focused on a specific case to develop numerical results in the field of the optimization problem explained in the previous chapter.

The studied case involved an initial orbit of an SSO at 520 km altitude; therefore, the number of considered intervals, as explained in Section 2.2.3, is 40. Moreover, a realistic nanosatellite [34] was chosen as shown in Figure 3.1 with the characteristics outlined in Table 3.1:

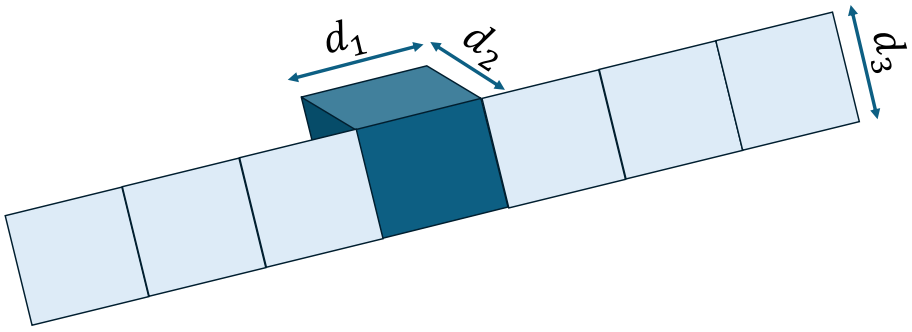


Figure 3.1: Technical schematic of the selected satellite ($d_1 = d_2 = 480$ mm; $d_3 = 620$mm)

Parameter	**Value**
Top-facing surface area (A1)	$0.48 \cdot 0.48 = 0.23$ m^2
Lateral-facing surface area (A2)	$0.48 \cdot 0.62 = 0.30$ m^2
Frontal surface area (A3)	$0.3 \cdot 7 = 2.1$ m^2
Tumbling area	$(1/3) \cdot$ (A1+A2+A3) $= 0.877$ m^2
Mass	75 kg

Table 3.1: Nanosatellite Parameters

The frontal surface area is assumed to be the cross-sectional area needed for the computation of the global probability of collision, whereas the drag area is assumed to be the tumbling area. Due to the fact that different firing times (t_{thr}) will be studied in this section, it is necessary to refer to a specific thruster.

In this work, a comparison of three different thrusters was conducted. First, a high-power system was considered, followed by a system with a high specific impulse, and finally, one with characteristics in between. The objective of this was to cover the spectrum of state-of-the-art space propulsive systems.

Initially, a **Hall Effect Thruster (HET)** , shown in Figure 3.2, [35] was studied:

Parameter	**Value**
Thrust	2.5 mN
Power	60 W
I_{sp}	850 s

Table 3.2: Hall's Thruster Parameters

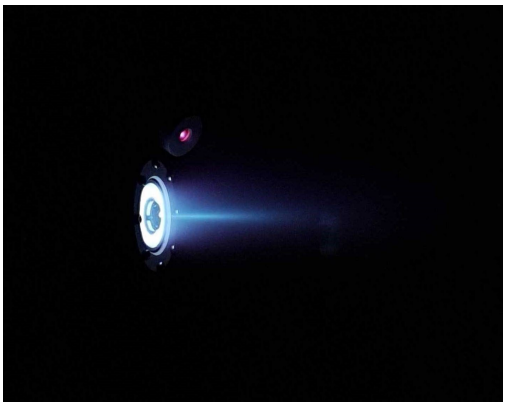


Figure 3.2: HET [36]

Once the satellite and thruster were set, a series of cases were run for different firing times, ranging from 0 days (natural decay) to 47.5 days (direct decay).

While maneuvering, the slope of the curve remains the same, as shown in Figure 3.3. Two cases are compared: one in which the firing time is 45 days and another in which it is 30 days. The former results in a shorter time in orbit, aligning with the time constraints set by the ESA. However, this approach requires a significant amount of fuel, making the spacecraft heavier at liftoff. In contrast, the latter involves a smaller ΔV applied over 30 days, leading to a longer time in orbit.

Additional simulations were conducted for intermediate cases, as presented in the following subchapter, highlighting the complexity of the studied problem. These considerations also significantly affect the overall probability of collision, as shown below.

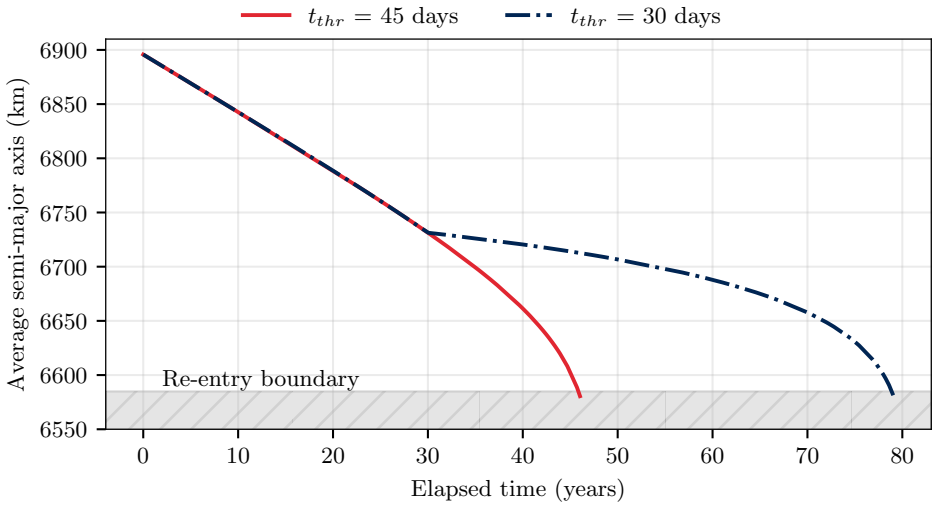


Figure 3.3: Average semi-major axis over elapsed time for different firing time (t_{thr}).

At this stage, another thruster was studied, notably the **Field-Emission Electric Propulsion (FEEP)** [37], shown in Figure 3.4. Its characteristics are as in Table 3.3:

Parameter	**Value**
Thrust	0.42 mN
Power	40 W
I_{sp}	3200 s

Table 3.3: FEEP's Thruster Parameters

Figure 3.4: FEEP thruster [38]

In the following case, two FEEP thrusters were used. Therefore, the total thrust and power will be double those listed in the table.

The final considered thruster is **ATHENA** (electrospray) [39], shown in Figure 3.5. Its characteristics are as in Table 3.4:

Parameter	**Value**
Thrust	1.75 mN
Power	60 W
I_{sp}	1500 s

Table 3.4: ATHENA's Thruster Parameters

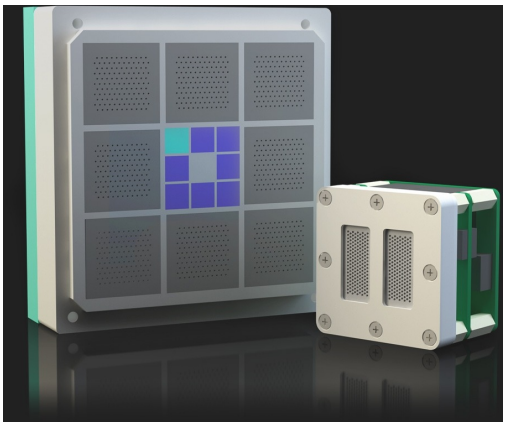


Figure 3.5: ATHENA thruster [40]

3.2 Pareto Front Analysis

In the following section, numerical results are presented for both a case in which the uncertainty in the drag was not considered and a case in which it was studied. The first study is conducted on three propulsion systems, whereas the second is performed only for one selected propulsor.

3.2.1 Collision probability as objective

The relationship between firing time and decay time is well known: the longer the firing time, the shorter the decay time. Additionally, firing time is strictly linked to propellant mass; therefore, a higher propellant mass results in a shorter decay time.

This Master's thesis aims to investigate another tradeoff based on a different design variable: the global collision probability during the decay phase. A commonly recognized threshold for collision probability, beyond which operators typically implement mitigation measures during a conjunction event, is 1E-4. Therefore, for this study, an high attention is given to this limit.

Since Multi-Objective Optimization does not yield a single optimum solution, the method adopted in this study is the Pareto dominance relation. This results in a set of alternative optimal solutions known as Pareto-optimal solutions. Their representation in the objective space forms the Pareto front, which is shown in Figure 3.6, Figure 3.7, Figure 3.8 and Figure 3.9.

A comparison of three different electric propulsion technologies is presented in this section, whose characteristics were outlined earlier. For all systems, the power consumption is compatible with the satellite's solar panel area and expected power generation at end-of-life. This analysis assumes constant solar activity, with $F10.7 = 150$ and $A_p = 16$.

For instance, starting from the **Hall thruster case**, in Figure 3.7, it is shown that a longer firing time corresponds to a greater propellant mass, a higher ΔV (as defined by Tsiolkovsky's equation. In this Master's Thesis, ΔV and propellant mass are used interchangeably.), and a lower probability of collision. Therefore, in the optimization process, a satisfactory balance among these variables must be chosen. Ultimately, the mission engineer is responsible for selecting the most suitable case based on the mission's objectives.

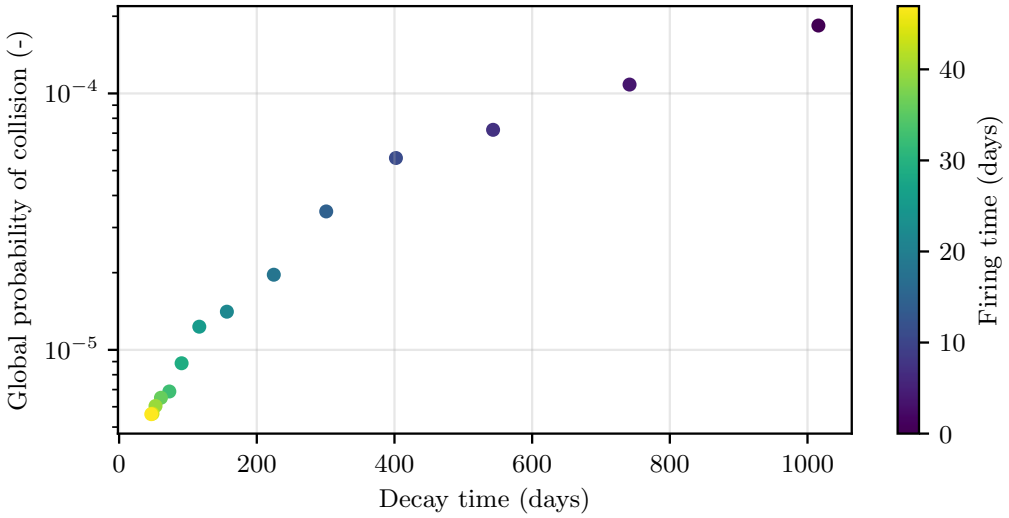


Figure 3.6: Global probability of collision, decay time, and firing time.

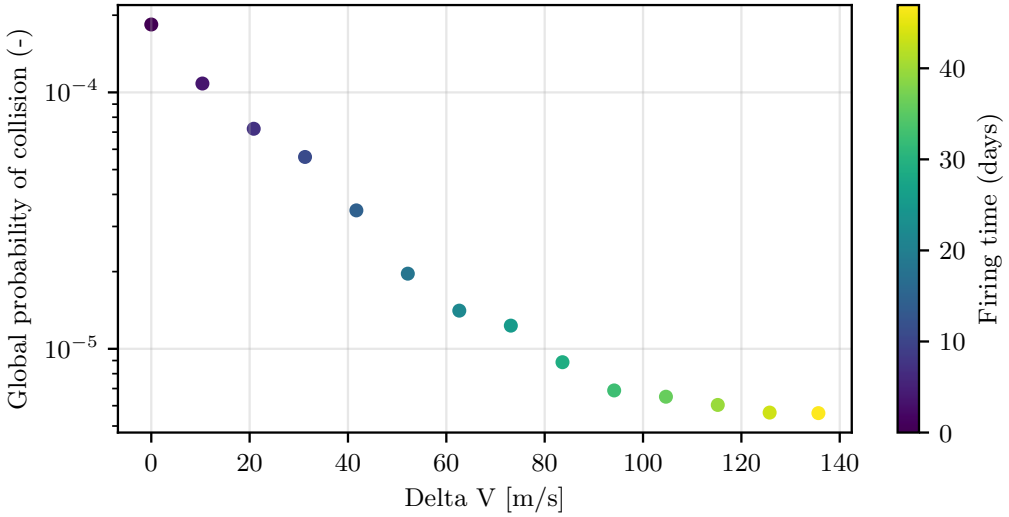


Figure 3.7: ΔV, global probability of collision and firing time.

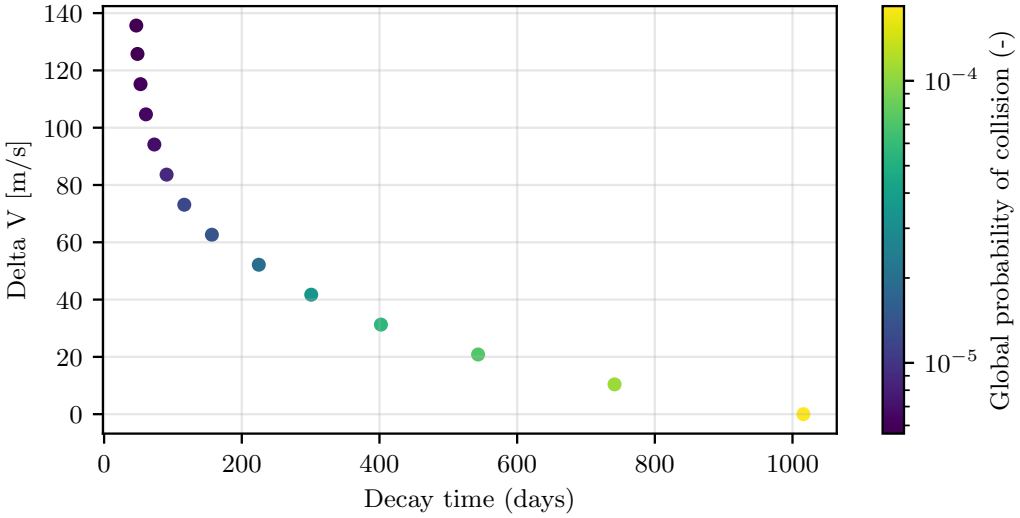


Figure 3.8: ΔV, global probability of collision and decay time.

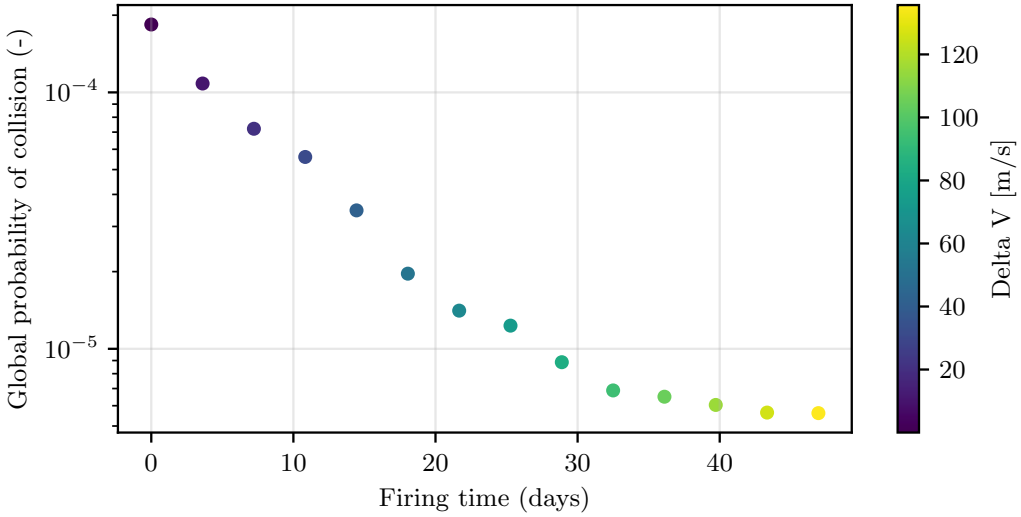


Figure 3.9: ΔV, global probability of collision and firing time.

By combining these results with those of the FEEP and ATHENA, as shown in Figure 3.13, Figure 3.10, Figure 3.11 and Figure 3.12, and introducing a limit on the usable propellant mass—since the goal is to maintain reasonable values for the actual propellant that can be used—we establish practical constraints. In the case of the FEEP, this limit is set at 440 g, while for the HET, it is 600 g due to its lower I_{sp}. Finally for ATHENA is 700 g. These limits affect the HET significantly more than the FEEP because the HET executes maneuvers much faster, resulting in higher propellant consumption.

The FEEP system can achieve the same overall collision probability as the HET, but with significantly less propellant consumption, thanks to its higher specific impulse (Figure 3.10). Although the HET could, in theory, enable the shortest deorbiting times (Figure 3.11), its performance is constrained by the available propellant mass. As a result, ATHENA emerges as the fastest viable solution under current limitations.

Figure 3.12 illustrates the evolution of collision probability along the deorbiting trajectory as a function of maneuver duration for all propulsion systems. The FEEP system is able to reduce the collision probability to approximately 1.4×10^{-5} with a firing duration of around 112 days. The HET system achieves similar results, but only up to the limit imposed by its maximum available propellant (as shown in the figure). In a hypothetical scenario with an enlarged propellant tank, the HET could further reduce the collision probability to approximately 6×10^{-6}. Finally, the ATHENA system achieves a probability as low as 7×10^{-6} with a shorter maneuver duration of 68.5 days.

Finally, a strong correlation between decay time and global probability of collision is observed for all the technologies (Figure 3.13). Therefore, when discussing the probability of collision along the decay trajectory, the variable that makes the most difference is the decay time. The higher the decay time, the higher the GPC, even though there is no linear dependence.

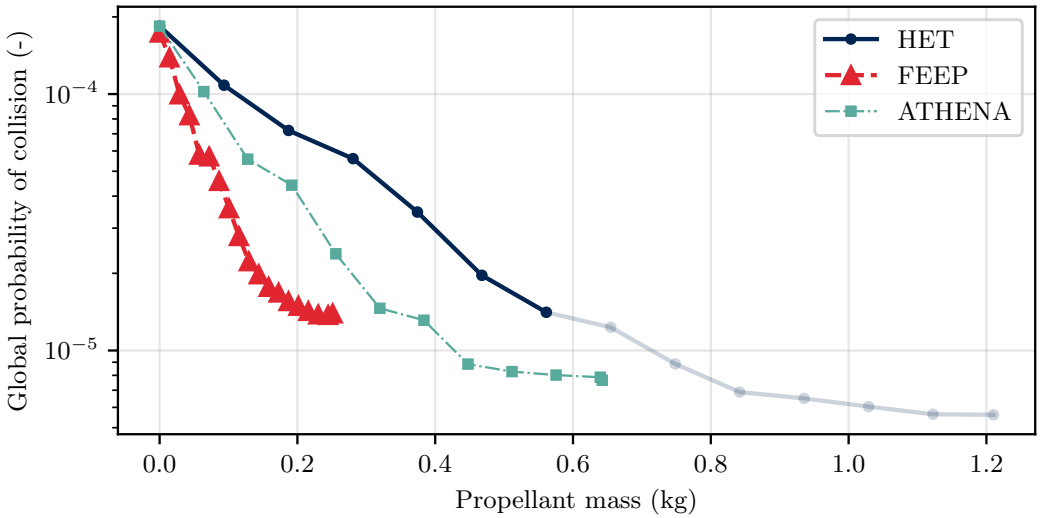


Figure 3.10: Global probability of collision and propellant mass for FEEP, HET and ATHENA.

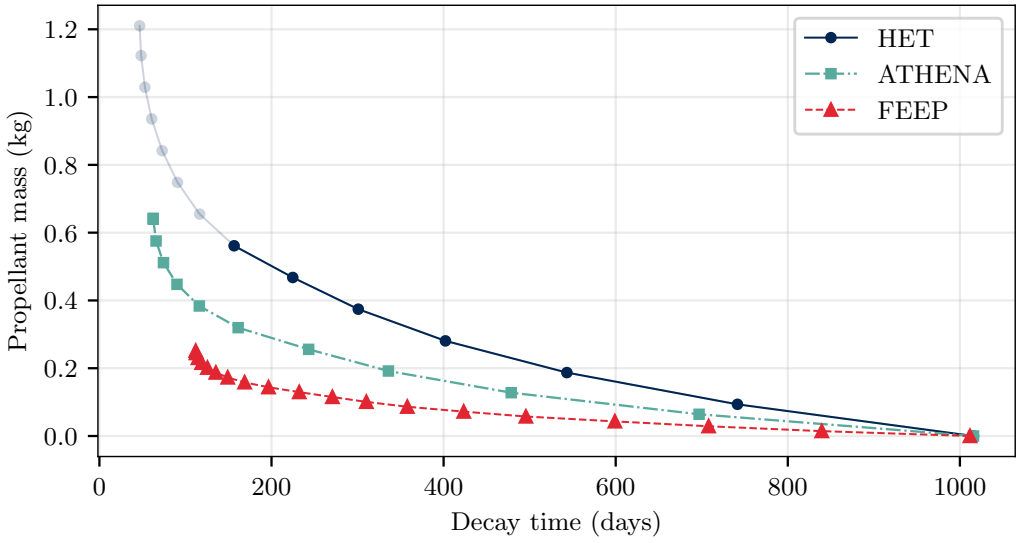


Figure 3.11: Propellant mass and decay time for FEEP, HET and ATHENA.

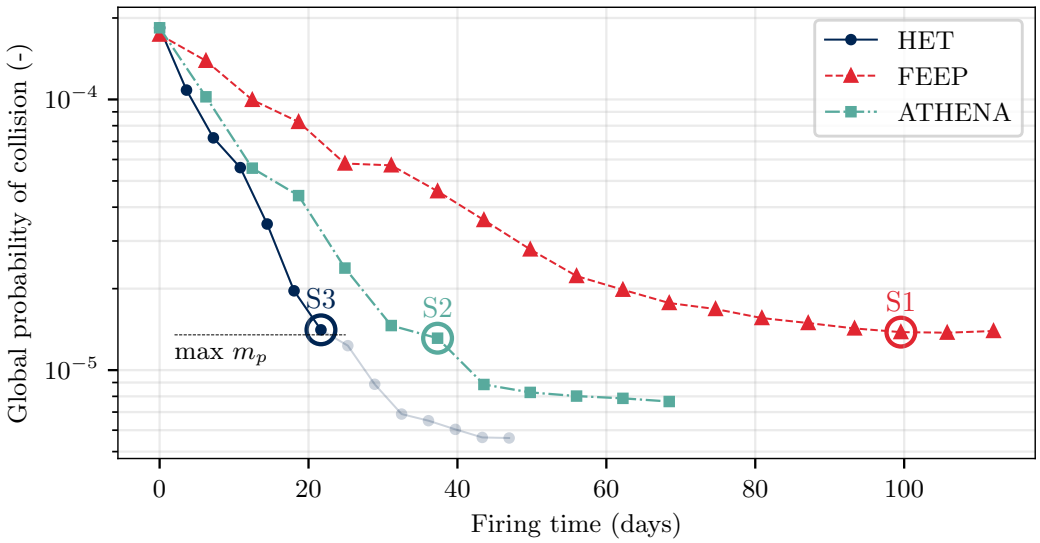


Figure 3.12: Global probability of collision and firing time for FEEP, HET and ATHENA. S1, S2, and S3 are Pareto points that will be analyzed later.

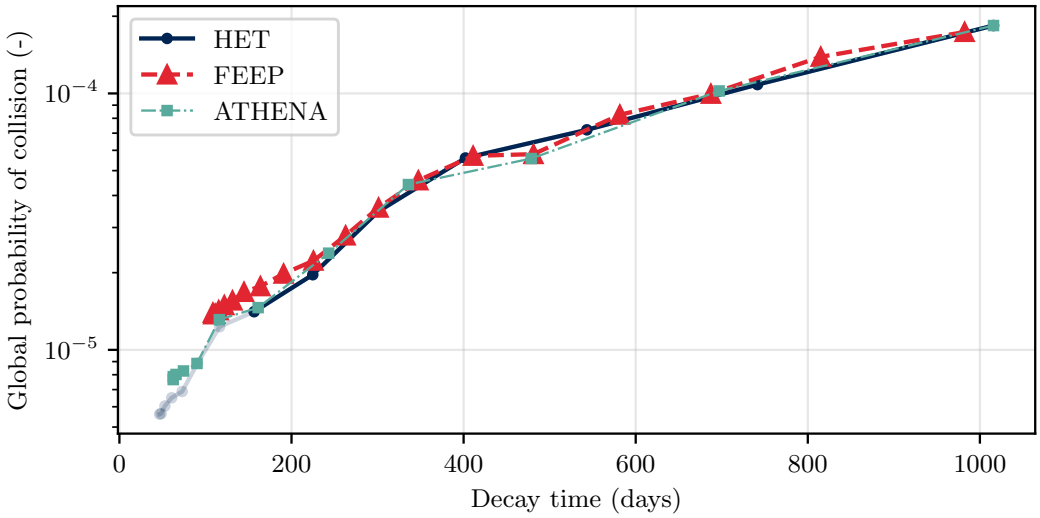


Figure 3.13: Global probability of collision and decay time for FEEP, HET and ATHENA.

To provide a clearer understanding of how each point in this Pareto front corresponds to a specific mission, a series of detailed plots are introduced at this stage. These visual representations (Figure 3.14, Figure 3.15 and Figure 3.16) allow for a more in-depth analysis, highlighting how various configurations impact mission performance. By examining these plots, it becomes possible to discern the relationship between design choices and their corresponding outcomes, offering valuable insights into the decision-making process for mission planning. The following figures represent the points named S1, S2, and S3 in Figure 3.12 and show the variation of SMA and propellant mass with respect to time. These points were chosen because, despite different firing times, they yield the same probability of collision along the decay trajectory. By looking at these plots, it is clear once again that the specific impulse of the FEEP is the most efficient, as the propellant mass consumption remains lower despite having a significantly longer firing time compared to the other cases. Additionally, it is evident that the higher the firing time, the faster the decay.

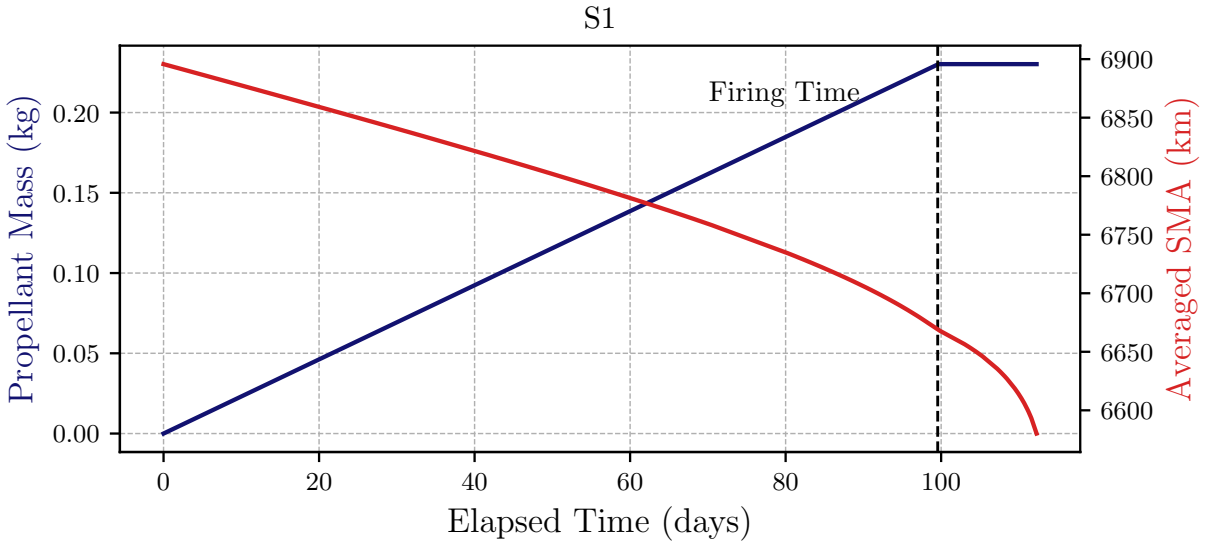


Figure 3.14: Evolution of SMA and propellant mass over time for FEEP-S1. The dashed line is the selected firing time

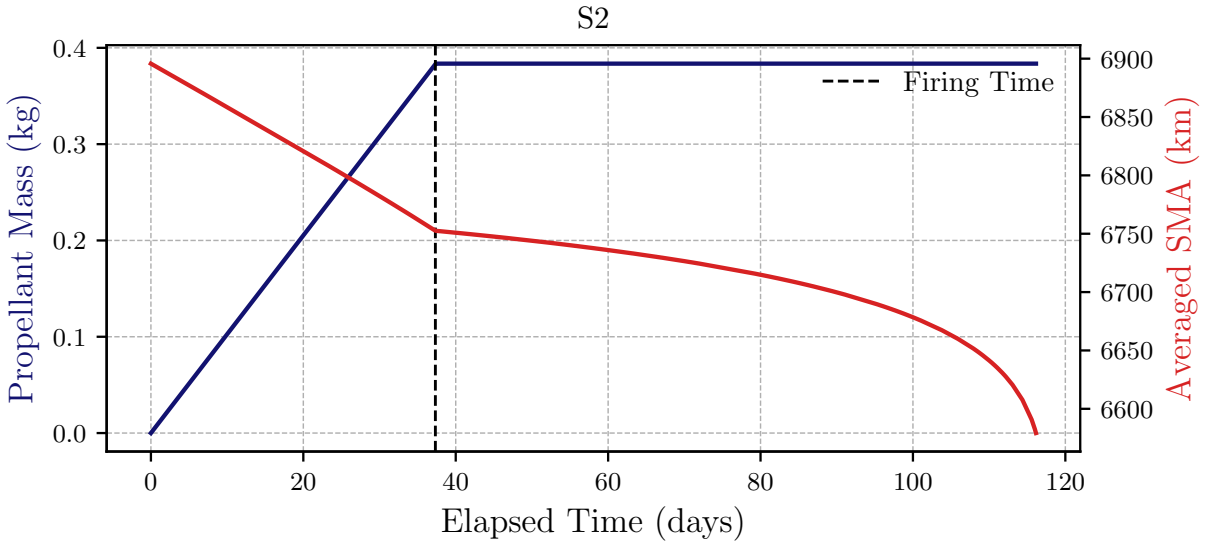


Figure 3.15: Evolution of SMA and propellant mass over time for ATHENA-S2. The dashed line is the selected firing time

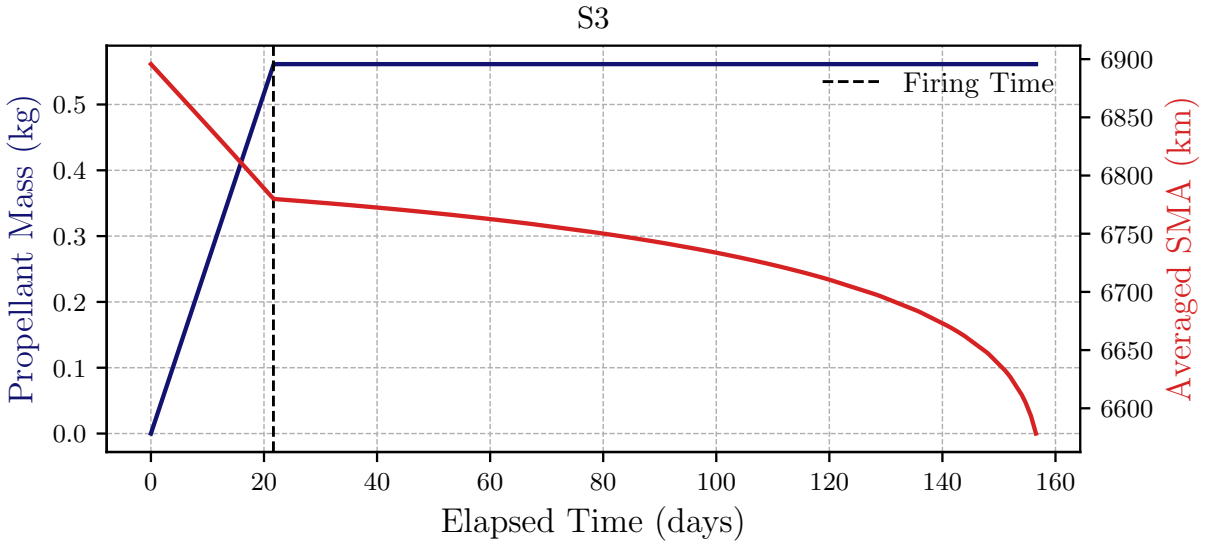


Figure 3.16: Evolution of SMA and propellant mass over time for HET-S3. The dashed line is the selected firing time

3.2.2 Atmospheric drag uncertainty analysis

This entire study was conducted with a fixed solar activity, but at this point, it is necessary to evaluate how considering a **varying solar activity** is actually affecting deorbiting, decay time, and consequently, the global probability of collision. Therefore, the uncertainty on the drag is actually taken into account. F10.7 and F10.7a are indices of solar activity. Ap is the geomagnetic index, which describes the activity of geomagnetic storms. These parameters influence atmospheric density and, consequently, the effect of aerodynamic drag, which determines the orbital decay time. The higher the solar activity, the greater the atmospheric density, and the faster the satellite's orbit will decay. In the end, in this work, three cases included in the Schatten predictions[1] were interpolated, namely -2σ (worst case: longer decay), $+2\sigma$ (best case: faster decay), and finally the *mean*. The propagator takes the time-varying values of these cases, and this set of values is interpolated over the time required by the propagator.

The studied case is that of an initial orbit being an SSO at 520 km altitude, and the discretization considered is the one in time with 40 intervals. Additionally, as a propulsor, the FEEP was selected.

At this stage, a series of contour plots were created (Figure 3.17, Figure 3.18, Figure 3.19,

[1]The Schatten files contain three cycle timing predictions: Nominal, Early, and Late. Each cycle includes a mean value along with upper and lower error bounds estimated at "+2 sigma" and "-2 sigma," respectively, resulting in a total of nine files.

Figure 3.20), illustrating the space in which all possible solutions are located between the worst-case and best-case scenarios, while also considering the mean condition. The larger the colored area, the greater the uncertainties.

By analyzing these figures, as expected, the $+2\sigma$ case allows the satellites to decay within a shorter time frame compared to the -2σ case.

An interesting observation is that by simply increasing the firing time, the uncertainties decrease along the decay trajectory (Figure 3.17), which also results in higher propellant consumption.

Figure 3.20 provides further evidence that the probability of collision is closely related to time. This study considers two different levels of solar activity, leading to distinct trajectories. However, despite these differences, the probability of collision remains nearly the same, as the uncertainty is relatively small. Nevertheless, it is not exactly identical.

The Figure 3.20 also reinforces the importance of the approach used in this Master's thesis. Indeed, ARES only provides the annual collision probability for a specified orbit (it is more used for a station keeping approach). However, if the orbit changes, as in the case of deorbiting, the probability of collision also changes. Therefore, using ARES in the conventional way would lead to an error. The magnitude of this error has not been estimated in this work until now. If a single interval is considered and the annual collision probability is simply multiplied by the entire duration of the deorbiting phase and then normalized to one year, the result is the red line in Figure 3.20. Consequently, the error would not be negligible; in fact, it does not even fall within the uncertainty interval of the drag estimate. This highlights the importance of the work performed.

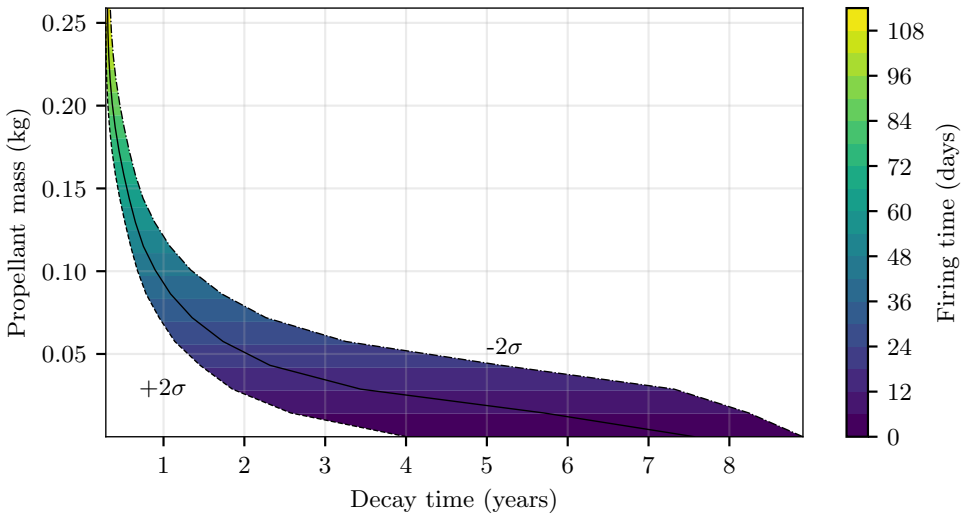


Figure 3.17: Decay time, propellant mass, and firing time with $\pm 2\sigma$ uncertainty bounds.

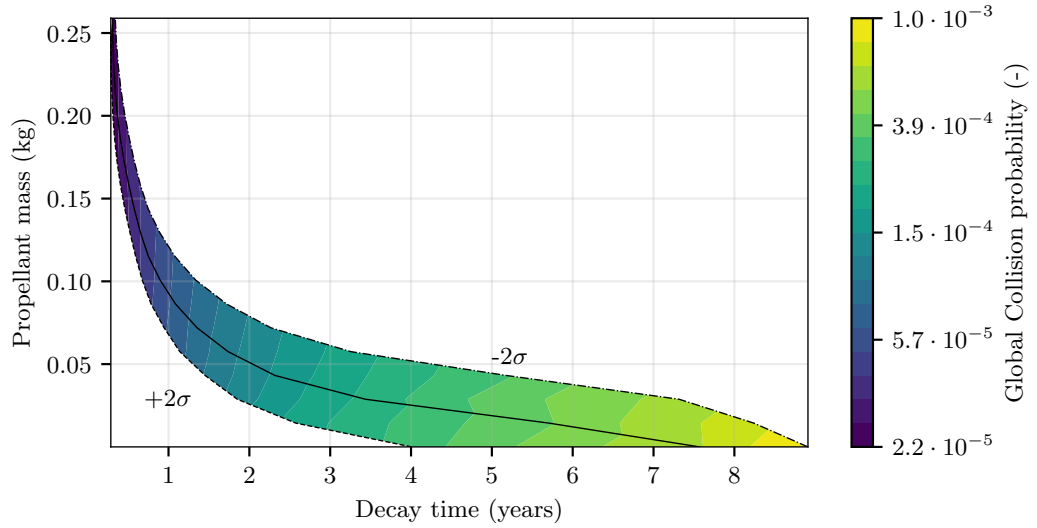


Figure 3.18: Propellant mass, collision probability and decay time with $\pm 2\sigma$ uncertainty bounds.

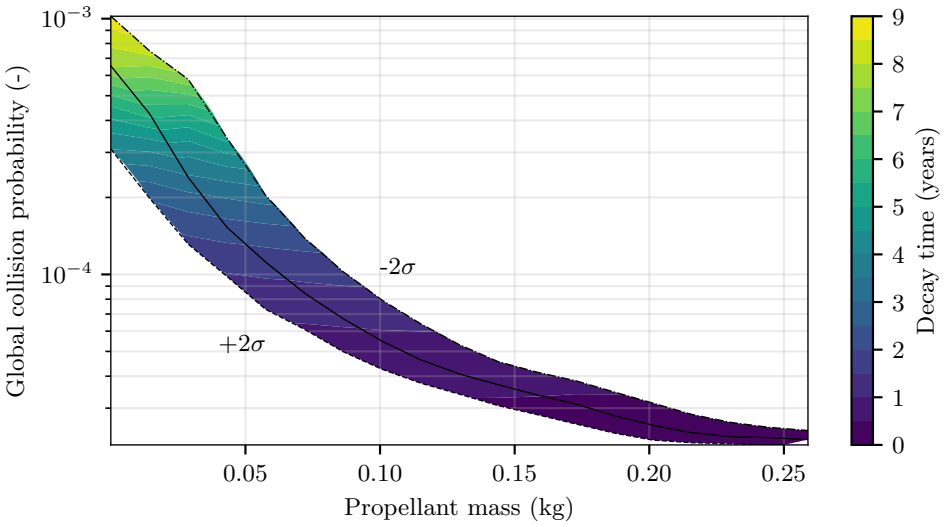


Figure 3.19: Propellant mass, decay time and collision probability with $\pm 2\sigma$ uncertainty bounds.

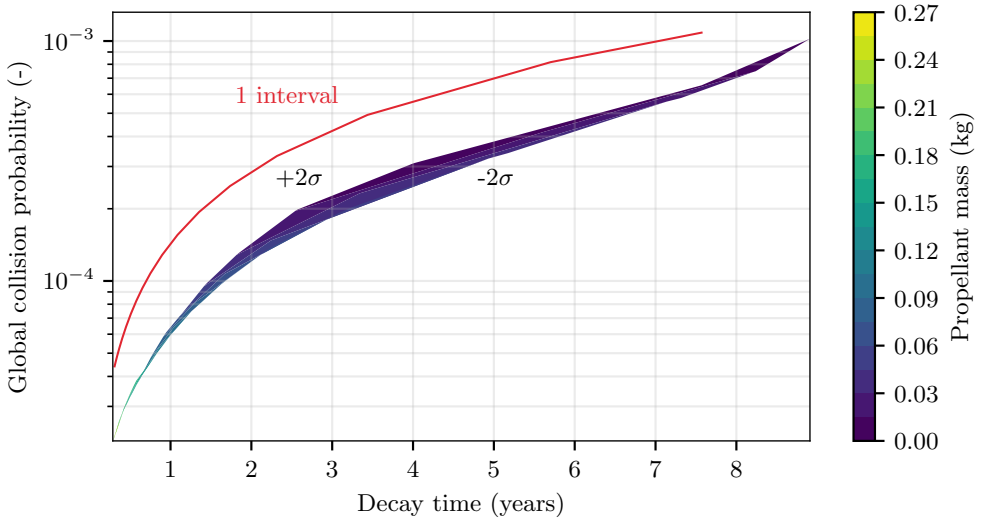


Figure 3.20: Decay time, collision probability and propellant mass with $\pm 2\sigma$ uncertainty bounds. The result for one interval shown as well.

Chapter 4

Conclusion

This thesis explores the multi-objective optimization of satellite deorbiting maneuvers, focusing on the trade-offs between ΔV, collision probability, decay time and firing time. With the increasing congestion in Low Earth Orbit (LEO) due to space debris, finding efficient and sustainable deorbiting strategies is becoming more critical. To face this challenge, the research applies a Pareto-based optimization approach, which allows mission designers to explore different trade-offs rather than committing to a single "best" solution.

The study begins with the development of an orbit propagator, using Modified Equinoctial Elements (MEE) to avoid computational singularities. It incorporates two major orbital perturbations—J2 effects (Earth's oblateness) and atmospheric drag, modeled with the NRLMSISE-00 atmospheric density model.

Once the satellite's orbital trajectory is propagated, the next challenge is to efficiently compute the Global Probability of Collision (GPC) along the decay trajectory. To reduce computational complexity when using the ARES tool (Assessment of Risk Event Statistics) for probability collision estimation, a discretization process is applied. This discretization is performed in two ways: based on semi-major axis (SMA) changes; based on time intervals. A dedicated discretization study was conducted to determine the optimal number of intervals that provide accurate results while keeping computational costs manageable. The findings indicate that for deorbiting scenarios aimed at studying the cumulative probability of collision, time-based discretization offers higher accuracy. As a result, the time-based approach was selected for the optimization phase.

Furthermore, the analysis clearly shows that increasing the number of discretization intervals significantly improves accuracy. Using 40 intervals instead of just 1 or 2 reduces the error by approximately 30–40%, making it a crucial factor in obtaining reliable collision probability estimates.

Additionally, the study examines a deorbiting case of a nanosatellite. A comparative

study of a multi-objective optimization process, generating a Pareto front for three different electric propulsion systems—Hall Effect Thruster (HET), Field-Emission Electric Propulsion (FEEP), and ATHENA electrospray—was conducted to analyze their impact on deorbiting maneuvers. The methodology enables rapid propulsion trade-offs: among the systems analyzed, FEEP is the most propellant-efficient, while the electrospray achieves the fastest decay.

Furthermore, the study investigated the influence of varying solar activity on the deorbiting process. Atmospheric density fluctuations were addressed using Schatten predictions, which account for different solar scenarios: a lower-density case (-2σ), a higher-density case ($+2\sigma$), and the nominal scenario (mean case). The main results can be summarized as follows:

- The proposed approach notably enhances the accuracy of collision probability estimation along the decay trajectory.
- Higher ΔV or propellant mass leads to reduced collision probability, primarily due to a shorter deorbiting duration.
- Under model uncertainties—such as atmospheric density—low propulsion efforts result in large variability in decay times, whereas higher efforts improve predictability and robustness.

These findings underscore the value of multi-objective optimization in space debris mitigation. By integrating discretization, dynamic collision probability estimation, and advanced orbit propagation techniques, the study demonstrates significant improvements in post-mission disposal strategies.

Future projections focus on the development and implementation of a more advanced guidance policy that not only regulates the semi-major axis but also dynamically adjusts the eccentricity to optimize orbital maneuvers. Additionally, future research will explore various deorbiting technologies, including tethers, which offer passive deorbiting solutions, and propellant-less methods, which, despite being currently less promising, may become viable with further technological advancements. These efforts aim to enhance sustainability and efficiency in satellite end-of-life strategies, ensuring more effective space debris mitigation in the years to come.

Bibliography

[1] European Space Agency (ESA). *Space Debris by the Numbers*. https://www.esa.int/Space_Safety/Space_Debris/Space_debris_by_the_numbers. Accessed: 2024-01-01. 2024.

[2] Mike Wall. *Kessler Syndrome and the Space Debris Problem*. https://www.space.com/kessler-syndrome-space-debris. Accessed: 2024-01-01. 2021.

[3] W.H. Steyn. *De-orbiting Strategies*. https://s3vi.ndc.nasa.gov/ssri-kb/static/resources/DeOrbit_whs.pdf. Accessed: 2024-01-01. 2024.

[4] J.-C. Liou, N.L. Johnson, and N.M. Hill. "Controlling the growth of future LEO debris populations with active debris removal". In: *Acta Astronautica* 66.5-6 (2010), pp. 648–653. ISSN: 0094-5765.

[5] ESA Space Debris Mitigation Working Group. *ESA Space Debris Mitigation Requirements*. https://sdup.esoc.esa.int/documents/download/ESSB-ST-U-007_Issue_1_30October2023.pdf. ESSB-ST-U-007, Issue 1, Revision 0, Accessed: 2024-01-01. 2023.

[6] Juan Ramón Sanmartín Losada and Sánchez Arriaga Gonzalo. "Seguridad espacial: una solución práctica para el problema de los residuos espaciales en órbita". In: *III Seminario sobre Actividades Espaciales y Derecho*. 2013.

[7] Maarten Schild and Ron Noomen. "Sun-synchronous spacecraft compliance with international space debris guidelines". In: *Advances in Space Research* 72 (2023), pp. 2585–2596. ISSN: 0273-1177. DOI: 10.1016/j.asr.2022.07.011. URL: https://www.sciencedirect.com/science/article/pii/S0273117722005853.

[8] European Space Agency (ESA). *ESA's Zero Debris Approach*. https://www.esa.int/Space_Safety/Clean_Space/ESA_s_Zero_Debris_approach. Accessed: 2024-09-17. 2024.

[9] Rolf Janovsky et al. "End-Of-Life De-Orbiting Strategies for Satellites". In: *Proceedings of the 4th European Conference on Space Debris* (2002), pp. 1–10. URL: https://www.researchgate.net/publication/224789858_End-Of-Life_De-Orbiting_Strategies_for_Satellites.

[10] Camilla Colombo et al. *Drag and Solar Sail Deorbiting: Re-entry Time versus Cumulative Collision Probability*. https://re.public.polimi.it/retrieve/e0c31c0c-9c62-4599-e053-1705fe0aef77/COLOC04-17.pdf. 2017.

[11] Python Package Index. *SpaceTrack*. https://pypi.org/project/spacetrack/. 2024.

[12] Harald Sack. *The Deorbit of Russian Space Station MIR*. http://scihi.org/deorbit-russian-space-station-mir/. Accessed: 2024-02-01. 2019.

[13] NASA. *20 Years Ago: Space Station Mir Reenters Earth's Atmosphere*. https://www.nasa.gov/history/20-years-ago-space-station-mir-reenters-earths-atmosphere/. Accessed: 2024-02-01. 2021.

[14] Christoph Steiger et al. *The Deorbiting of ESA's Gravity Mission GOCE – Spacecraft Operations in Extreme Drag Conditions*. https://arc.aiaa.org/doi/pdf/10.2514/6.2014-1934. 2014.

[15] Yang Chen et al. "Optimal Mission Planning of Active Space Debris Removal Based on Genetic Algorithm". In: *IOP Conference Series: Materials Science and Engineering* 715 (2020), p. 012025. DOI: 10.1088/1757-899X/715/1/012025. URL: https://iopscience.iop.org/article/10.1088/1757-899X/715/1/012025/pdf.

[16] Andrea Verri. *Low-Thrust Many-Revolutions Maneuver Optimization for Future Active Debris Removal Application*. https://webthesis.biblio.polito.it/18293/1/tesi.pdf. 2021.

[17] Emanuela Gaglio and Riccardo Bevilacqua. "Time Optimal Drag-Based Targeted De-Orbiting for Low Earth Orbit". In: *Acta Astronautica* 207 (June 2023), pp. 316–330. DOI: 10.1016/j.actaastro.2023.03.011. URL: https://doi.org/10.1016/j.actaastro.2023.03.011.

[18] Shuta Fukii et al. "Assessing Collision Probability in Low-Thrust Deorbit". In: *Journal of Space Safety Engineering* 9.1 (2022), pp. 47–55. DOI: https://doi.org/10.1016/j.jsse.2021.11.002. URL: https://www.sciencedirect.com/science/article/pii/S246889672100104X.

[19] Jonas Radtke, Christopher Kebschull, and Enrico Stoll. "Interactions of the space debris environment with mega constellations—Using the example of the OneWeb constellation". In: *Acta Astronautica* 131 (2017), pp. 55–68. DOI: 10.1016/j.actaastro.2016.11.021. URL: http://www.sciencedirect.com/science/article/pii/S0094576516306207.

[20] R. A. Brouke and P. J. Cefola. "On the Equinoctial Orbital Elements". In: *Celestial Mechanics* 5 (1972), pp. 303–310.

[21] Python Software Foundation. *Python Programming Language*. Accessed: 2024-12-20. 2024. URL: https://www.python.org/.

[22] PyPI Contributors. *nrlmsise00: Python Implementation of the NRLMSISE-00 Atmospheric Model*. Accessed: 2024-12-20. 2024. URL: https://pypi.org/project/nrlmsise00/.

[23] David A. Vallado. *Fundamentals of Astrodynamics and Applications*. Fourth Edition. Hawthorne, CA: Microcosm Press, 2013. ISBN: 978-1881883180 (pb), 978-1881883197 (hb), 978-1881883203 (ebook).

[24] Georges Krier. "Satellite Collision Probability for Long-Term Encounters and Arbitrary Primary Satellite Shape". en. In: *Proceedings of the 7th European Conference on Space Debris*. 2017.

[25] European Space Agency. *ESA's MASTER Software Tool*. https://www.esa.int/ESA_Multimedia/Images/2013/04/ESA_s_MASTER_software_tool. Accessed: 2024-09-25. 2013.

[26] NASA. *OpenMDAO - Multidisciplinary Design, Analysis, and Optimization Framework*. https://software.nasa.gov/software/MSC-25457-1. Accessed: 2024-09-25. 2024.

[27] Mohamed Torky et al. "Analyzing Space Debris Flux and Predicting Satellites Collision Probability in LEO Orbits Based on Petri Nets". In: *IEEE Access* 7 (2019), pp. 83461–83473. DOI: 10.1109/ACCESS.2019.2922835.

[28] European Space Agency. *Space Debris User Portal (SDUP)*. https://sdup.esoc.esa.int/. Accessed: 2024-09-25. 2024.

[29] European Space Agency. *DRAMA Python Package Documentation*. https://sdup.esoc.esa.int/drama/python_package_docs/. Accessed: 2024-09-25. 2024.

[30] European Space Agency. *ARES Module Documentation - DRAMA Python Package*. https://sdup.esoc.esa.int/drama/python_package_docs/modules/ares.html. Accessed: 2024-09-25. 2024.

[31] ESA/ESOC Space Debris Office (OPS-SD). *Debris Risk Assessment and Mitigation Analysis (DRAMA) Software User Manual*. Version 2.5. European Space Agency. Robert-Bosch-Strasse 5, 64293 Darmstadt, Germany, 2024.

[32] Raúl Domínguez-González et al. "Update of ESA DRAMA ARES: Comparison of Envisaged Collision Alerts with Operational Statistics and Impact of Catalogue Accuracy". In: *Proceedings of the 6th European Conference on Space Debris*. ESA SP-723, April 22–25, 2013. European Space Agency (ESA). Darmstadt, Germany: ESA Publications Division, 2013.

[33] Johannes Gelhaus et al. "Upgrade of DRAMA: ESA's Space Debris Mitigation Analysis Tool Suite". In: *Proceedings of the 6th European Conference on Space Debris*. ESA SP-723, April 22–25, 2013. European Space Agency (ESA). Darmstadt, Germany: ESA Publications Division, 2013.

[34] *MP42 Online Single Document*. Accessed: 2024-02-09. 2021. URL: https://cdn.asp.events/CLIENT_CLDD_9BDAB70C_5056_B733_4934A7872C9C46B0/sites/DSEI-2021/media/libraries/exhibitor-products/33162-MP42%202021%2006%20online%20single.pdf.

[35] Exotrail. *Spaceware - Exotrail*. Accessed: 2024-02-09. 2024. URL: https://www.exotrail.com/product/spaceware.

[36] Exotrail. *ExoMG™ Nano Thruster Performs Beyond Expectations*. Exotrail Blog. Accessed: 2025-03-22. 2025. URL: https://www.exotrail.com/blog/exomg-tm-nano-thruster-performs-beyond-expectations.

[37] ENPULSION. *ENPULSION Nano*. Accessed: 04-Mar-2025. 2025. URL: https://www.enpulsion.com/order/enpulsion-nano/.

[38] Andrei Shumeiko, Victor Telekh, and Sergei Ryzhkov. "Thrust-vectoring schemes for electric propulsion systems: A review". In: *Chinese Journal of Aeronautics* (2025). DOI: 10.1016/j.cja.2025.103401. URL: https://doi.org/10.1016/j.cja.2025.103401.

[39] David Villegas et al. "Indirect Characterization of ATHENA Performance, a Novel Externally Wetted Electrospray Propulsion System". In: *Space Propulsion Conference 2022*. 2022.

[40] Universidad Politécnica de Madrid. *Título del artículo (poni aquí el título real de la noticia)*. UPM News. Accessed: 2025-03-22. 2025. URL: https://www.upm.es/UPM/SalaPrensa/Noticias?id=9f0e57cef6f95810VgnVCM10000009c7648a____&fmt=detail&prefmt=articulo.

Appendix A

Additional Experimental Analysis

A.0.1 Codes

Algorithm 1: Transformation from Classical Orbital Elements (COE) to Modified Equinoctial Elements (MEE) (Filename: `coe2mee`)

1: **Input:** Orbital state vector $x_{coe} = [a, e, i, \omega, \Omega, \nu]$
2: **Output:** Orbital state vector $x_{mee} = [p, f, g, h, k, L]$
3: Compute the semi-latus rectum p
4: Compute f and g (eccentricity vector components)
5: Compute h and k (inclination vector components)
6: Compute the true longitude L
7: **Return** x_{mee}

Algorithm 2: Conversion from Modified Equinoctial Elements (MEE) to Classical Orbital Elements (COE)(Filename: `mee2coe`)

1: **Input:** Orbital state vector $x_{mee} = [p, f, g, h, k, L]$
2: **Output:** Orbital state vector $x_{coe} = [a, e, i, \omega, \Omega, \nu]$
3: Decompose the input vector:
$p \leftarrow x_{mee}[1], f \leftarrow x_{mee}[2], g \leftarrow x_{mee}[3], h \leftarrow x_{mee}[4], k \leftarrow x_{mee}[5], L \leftarrow x_{mee}[6]$
4: Compute the Semi-Major Axis (SMA)
5: Compute the Eccentricity (ECC)
6: Compute the Inclination (INC)
7: Compute the Argument of Periapsis (AOP)
8: Compute the Right Ascension of the Ascending Node (RAAN)
9: Compute the True Anomaly (TA)
10: Construct the classical orbital elements vector:

$$x_{coe} \leftarrow [a, e, i, \omega, \Omega, \nu]$$

11: **Return** x_{coe}

Algorithm 3: Compute Velocity Vector in ECI from MEE (Filename: `mee2velocityvector`)

1: **Input:** Orbital state vector x in MEE, standard gravitational parameter μ
2: **Output:** Velocity vector in ECI frame vel_vec_ECI
3: Decompose the MEE vector into components p, f, g, h, k, L
4: Compute trigonometric values $\cos(L)$ and $\sin(L)$
5: Compute auxiliary parameters s^2 and $\sqrt{\mu/p}$
6: Compute the components of vel_vec_ECI using s^2, $\cos(L)$, $\sin(L)$, and the MEE components
7: **Return** vel_vec_ECI

Algorithm 4: Transformation from ECI to RCN Frame (Filename: `ECI2RCN`)

1: **Input:** Force vector F_{ECI}, angles RAAN, INC, AOP
2: **Output:** Transformed force vector F_{RCN}
3: Validate that F_{ECI} is a 3D vector
4: Precompute trigonometric values for RAAN, INC, and AOP
5: Construct the rotation matrix R using the input angles
6: Multiply R with F_{ECI} to compute F_{RCN}
7: **Return** F_{RCN}

Algorithm 5: Compute Position Vector in ECI from MEE (Filename: `mee2positionvector`)

1: **Input:** Orbital state vector x in MEE
2: **Output:** Position vector in ECI frame pos_vec_ECI
3: Compute trigonometric values for $\cos(L)$ and $\sin(L)$
4: Compute auxiliary parameters s, q, r, and a_2
5: Use the computed parameters to calculate the components of pos_vec_ECI
6: **Return** pos_vec_ECI

Algorithm 6: Calculate Greenwich Mean Sidereal Time (GMST) (Filename: `calculate_gmst`)

1: **Input:** Date and time (UTC), *date*
2: **Output:** GMST angle in degrees, θ_{GMST}
3: Compute the UT1 time, t_{UT1}, from the input *date*

$$t_{UT1} \leftarrow \text{date2t\_ut1}(date)$$

4: Compute the GMST angle θ_{GMST} in seconds:

$$\theta_{GMST\_sec} \leftarrow 67310.54841 + (876600 \cdot 3600 + 8640184.812866) \cdot t_{UT1} + 0.093104 \cdot t_{UT1}^2 - 6.2 \cdot 10^{-6} \cdot t_{UT1}^3$$

5: Convert θ_{GMST} to degrees:

$$\theta_{GMST} \leftarrow \frac{\theta_{GMST\_sec}}{240} \mod 360$$

6: **Return** θ_{GMST}

Algorithm 7: Convert Vector from ECI to ECEF (Filename: `ECI2ECEF`)

Input: Vector in ECI frame v_{ECI}, date *date*
Output: Vector in ECEF frame v_{ECEF}
Compute the GMST angle from the given date
Compute trigonometric values $\cos(\theta)$ and $\sin(\theta)$
Apply the rotation matrix to transform v_{ECI} to v_{ECEF}
Return v_{ECEF}

Algorithm 8: Convert Position Vector from ECEF to Latitude and Longitude (Filename: `ECEF2LatLon`)

Input: Position vector in ECEF frame pos_vec_ECEF
Output: Geodetic latitude and longitude in degrees
Compute the equatorial projection of the position vector
Compute the initial longitude using the position vector components
Initialize the latitude computation:
    Set initial values for $distance\_E\_S\_z$, e (eccentricity), and tolerance
    Set an initial guess for latitude ϕ_{old} based on the vertical distance
while Convergence is not achieved (difference between successive latitudes > tolerance) **do**
    Compute C_m using the current latitude estimate
    Update the latitude estimate ϕ_{new}
end while
Set the final latitude to ϕ_{new}
Return latitude and longitude

Algorithm 9: Compute Collision Probabilities (Filename: `Compute Collision Probabilities`)

Input: JSON file with epochs, SMA, eccentricity, inclination, RAAN, and AOP
Output: Collision probabilities, propagation times, and global probability decay
Start total execution timer
Define dynamic JSON file path and verify its existence
Load simulation data from the JSON file:
    Extract epochs, SMA, eccentricity, inclination, RAAN, and AOP arrays
Validate completeness of the input data
Create configurations for each entry:
for each entry in the input data arrays **do**
    Construct configuration with SMA, eccentricity, inclination, RAAN, AOP, and epoch
    Append configuration to the configuration list
end for
Execute configurations sequentially with retry logic:
for each configuration in the configuration list **do**
    Attempt execution up to 5 times
    if execution succeeds **then**
        Compute collision probability and propagation time
        Append results to the results list
    end if
end for
Sort results by epoch and SMA
Compute derived metrics:
    Extract collision probabilities from results
    Compute time intervals between epochs in seconds
    Calculate instantaneous probability vector
    Compute global probability decay as the sum of the instantaneous probabilities
Output detailed results:
    Print results for each configuration, including collision probability and propagation time
    Print collision probabilities, time intervals, and global probability decay
    Print total execution time