Space Debris Collision Avoidance Manoeuvre Design and Analysis

Carer, C.P.M.; Mooij, E.

DOI
10.2514/6.2025-0984

Publication date
2025

Document Version
Final published version

Published in
Proceedings of the AIAA SCITECH 2025 Forum

AIAA SciTech Forum
6-10 January 2025, Orlando, FL
AIAA SCITECH 2025 Forum

10.2514/6.2025-0984

Space Debris Collision Avoidance Manoeuvre Design and Analysis

Chloé Carer* and Erwin Mooij†
Delft University of Technology, Faculty of Aerospace Engineering,
Kluyverweg 1, 2629 HS Delft, The Netherlands

Ensuring the sustainability of future space missions requires addressing the space-debris issue proactively. Space debris threatens future space missions, making the need for collision avoidance manoeuvres essential. This research explores robust methods to perform such manoeuvres, focusing on guidance and control systems, using the Starlink constellation as a reference. A robust guidance system integrating convex optimisation enables autonomous, fuel-efficient collision avoidance of space debris. Testing across diverse conditions shows suitability for various satellite characteristics. Attitude-control requirements are analysed with a linear quadratic regulator controller. Low-thrust manoeuvres with constant capabilities are optimal for this specific mission, minimising fuel usage and achieving desired miss distances. The outcomes of this research represent a significant advancement in space-debris mitigation and contribute to enhancing space sustainability.

I. Introduction

From the first ever satellite successfully placed in orbit, Sputnik 1, in 1957, the number of objects in space never stopped increasing and is accumulating at a more rapid rate than it ever did. To provide context, in 2024, the number of tracked objects stands at approximately 45,500, whereas this number was roughly 10,000 in 2004, just two decades ago. The accumulation of debris over the last two decades can be observed in Fig. 1. The reason for such accumulation can be attributed to two key factors: *catastrophic collisions* and *the expanding utilisation of space* [1].

To start with, a series of catastrophic events, drastically expanded the number of tracked objects in space [2]. Events, such as the 2007 Fengyun-1C Chinese anti-satellite test alone, increased the number of trackable debris by 25%[a]. Additionally, the emerging NewSpace era is also directly increasing the occupation of low-Earth orbit (LEO) [3], which will in turn boost the number of debris in space in the near future. For example, SpaceX, through their Starlink mega-constellation, has the objective of placing more than 40,000 satellites in orbit around Earth, providing a large satellite Internet-access coverage, largely contributing to space pollution.

As the number of objects in space goes up, so does the likelihood of collisions, resulting in more frequent collisions and the creation of additional debris. This phenomenon, first defined by Donald J. Kessler in 1978, indicates that once a specific critical threshold is reached, the accumulation of space debris will continue to grow in a self-sustaining manner. Collisions not only produce more debris but also trigger subsequent collisions, setting off a chain reaction, which can pose a significant threat to the future of space exploration. This phenomenon, now referred to as the Kessler syndrome, highlights the urgency to improve research on the matter and put resources into solving such problems to guarantee the future of space exploration [4]. For this reason, engineers are becoming more interested in studying conjunctions and developing ways to

*MSc Student, Section Astrodynamics and Space Missions, chloecarer@yahoo.com.
†Associate Professor, Section Astrodynamics and Space Missions, e.mooij@tudelft.nl, Associate Fellow AIAA.
[a]European Space Agency, *About space debris*, https://www.esa.int/Space_Safety/Space_Debris/About_space_debris [Last visited on 23/10/2024].
[b]Space-Track, *Satellite catalogue*, https://www.space-track.org/ [Last visited on 01/02/2024].

Figure 1. Evolution of the number of objects in the Geostationary orbit (GEO) from 2004 (left) to 2024 (right). Constructed from two-line elements (TLE) taken from Space-Track[b].

prevent them. This paper will be part of such research, having as its main objective to develop a technology capable of mitigating such conjunctions, allowing space to remain accessible.

To achieve the latter objective, conducting a comprehensive literature review is crucial for exploring this issue, bridging existing knowledge gaps, and comprehending strategies for collision avoidance manoeuvre (CAM). Such a study has been conducted and enabled to find a gap in the literature, which could be filled to improve scientific research on the matter of collision avoidance.

Based on this analysis, it can be deduced that two primary technologies have the potential to address the problem of space debris proliferation: removing the debris through active debris removal and implementing CAM to prevent collisions. The latter will be the main point of focus of this study. As the number of debris increases, so does the need for manoeuvres; on average, each of the European Space Agency's (ESA) Earth-orbiting satellites requires approximately two manoeuvres each year[c]. As human activity in space increases, manual planning of collision avoidance manoeuvres will become impractical. For this reason, a technology capable of automating such processes should be developed for the future of space exploration to be safeguarded.

This paper focuses on researching the most robust methods to perform CAMs, with an emphasis on the guidance and control of manoeuvrable vehicles. The Starlink constellation is used as a reference case, considering its popularity and extensive development, although the developed system applies to any satellite. The core of achieving successful manoeuvres lies in the creation of a robust guidance system. This system integrates a convex optimisation algorithm with a reference tracking mechanism, enabling the autonomous execution of CAMs to be both robust and fuel-efficient. By using a convex guidance approach, optimal fuel trajectories are determined, while ensuring adherence to predefined reference paths and constraints, such as keep-out-sphere and available thrust. Rigorous testing across diverse conditions showcases the system's suitability for different satellites and missions. In addition, the effect of solver choice is addressed. It is noted that the current research is only the first step to converge to a robust guidance, navigation, and control (GNC) system design. Even though the guidance problem is addressed in detail, many more mission aspects could (and should) be studied. Furthermore, the interaction with attitude control is well understood, and could degrade the guidance-system performance. Proper attention should thus be paid to the control-system design. Last, but not least, having an accurate (and preferably autonomous) navigation system is an essential part of such a GNC system, to reduce the errors due to imperfect knowledge of both the satellite's own state and potential dangerous debris that is to be avoided. To limit the scope of the research, these elements are left as future work, though.

[c]European Space Agency, *FAQ: Frequently asked questions*, https://www.esa.int/Space_Safety/Space_Debris/FAQ_Frequently_asked_questions [Last visited on 01/10/2024].

Table 1. Some of the known characteristics of SpaceX's Starlink-4039 satellite.

Orbit	Characteristics
Altitude ≈ 546 km	Total mass ≈ 260 kg
circular	(minisatellite)
Inclination ≈ 53.2194°	Body dimensions ≈ 3.2 x 1.6 x 0.2 m
Eccentricity ≈ 0.000152	Body mass ≈ 196 kg
Period ≈ 95.5 min	Solar arrays: 12 deployable panels
	(3.2 x 0.0075 x 9.6 m)
	Solar panel surface area[e]
	≈ 3.2 x 9.6 m ≈ 30.72 m^2
	Solar panel mass ≈ 64 kg
	Cross-sectional area: 31.36 m^2
	(approximated the larger side of the satellite
	by a rectangle with sides 3.2 x 9.8 m)
	Propulsion: Electric (Krypton)

To this end, the layout of the paper covering the proposed research is as follows. Section II gives an overview of the research methodology, including the reference vehicle, mission and flight dynamics details. Then, Sec. III describes both the nominal guidance and attitude control sequence. Following this, the results of the research are highlighted in Sec. IV. To finish with, conclusions and recommendations for further improvements are given in Sec. V.

II. Methodology

This chapter defines the reference vehicle and mission while also introducing the mission dynamics and the perturbations accounted for in the orbital environment.

A. Reference Vehicle and Mission

A generalised model resembling the known attributes of Starlink will be developed to serve as the basis for all simulations. Although very little information on the Starlink satellites' geometry and mass distribution is available, a deduction can be made using illustrations and other available data sources, such as the Falcon 9 user guide[d] and pictures of how the satellites are stacked in the launcher's payload bay. The approximation of the vehicle's dimensions is listed in Table 1, with an illustration of the spacecraft shown in Fig. 2. Additionally, taking the overall geometry of Starlink, the inertia of the entire satellite can be computed, assuming a uniform mass distribution and using the box approximation with a flat plate as a solar panel. The overall satellite inertia can be approximated as $\boldsymbol{I} = \text{diag}(2136.3, 2302.3, 277.3)$ kg m^2.

A nominal propulsion configuration includes a maximum thrust of $F_{T,max} = 10$ N along each axis, with a specific impulse of $I_{sp} = 292$ s. However, since the Starlink satellites possess low-thrusting capabilities, to assess the effect of low-thrust propulsion on the manoeuvring, a combination of $F_{T,max} = 0.01$ / 0.1 N and $I_{sp} = 2{,}000$ s will be analysed.

The Starlink constellation is located in the critical band of approximately 550 km altitude, 53°inclination, and close to zero-eccentricity. The scenario that is studied in this paper is one where a critical collision can occur after half an orbit. Figure 3 highlights this scenario. To fulfil the CAM objective and adjust the satellite's trajectory to an acceptable state, it is necessary to establish a KOS constraint. This sphere, centred on the debris position, is designed to prevent collisions and ensure that the satellite maintains a safe distance from the debris. To achieve this, the potential combination of a nominal and tracking guidance will be explored.

[d]SpaceX, *Falcon User's Guide*, https://www.spacex.com/media/falcon-users-guide-2021-09.pdf [Last visited 22/02/2024].

[e]Seradata, *Satellite Catalogue*, https://www.seradata.com/ [Last visited on 01/10/2024].

[f]Shutterstock, *Starlink Satellite Illustration*, https://www.shutterstock.com/image-illustration/internet-broadcast-satellite-train-around-earth-2116913408 [Last visited on 11/03/2024].

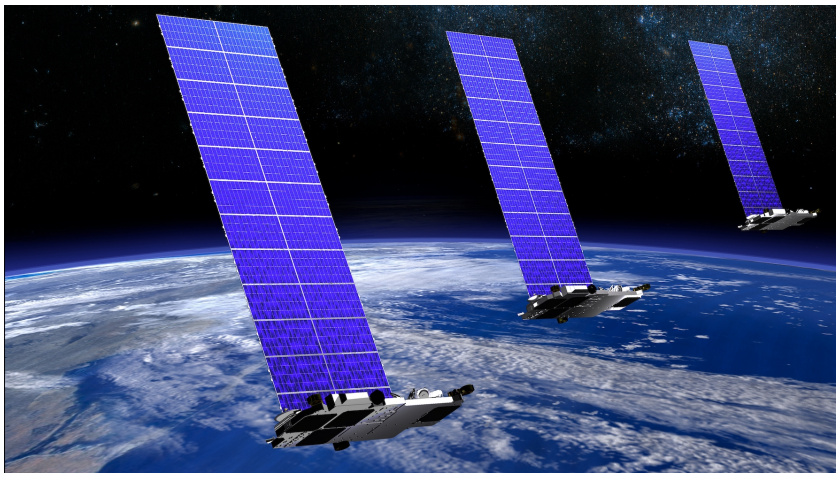

Figure 2. SpaceX's Starlink satellites[f].

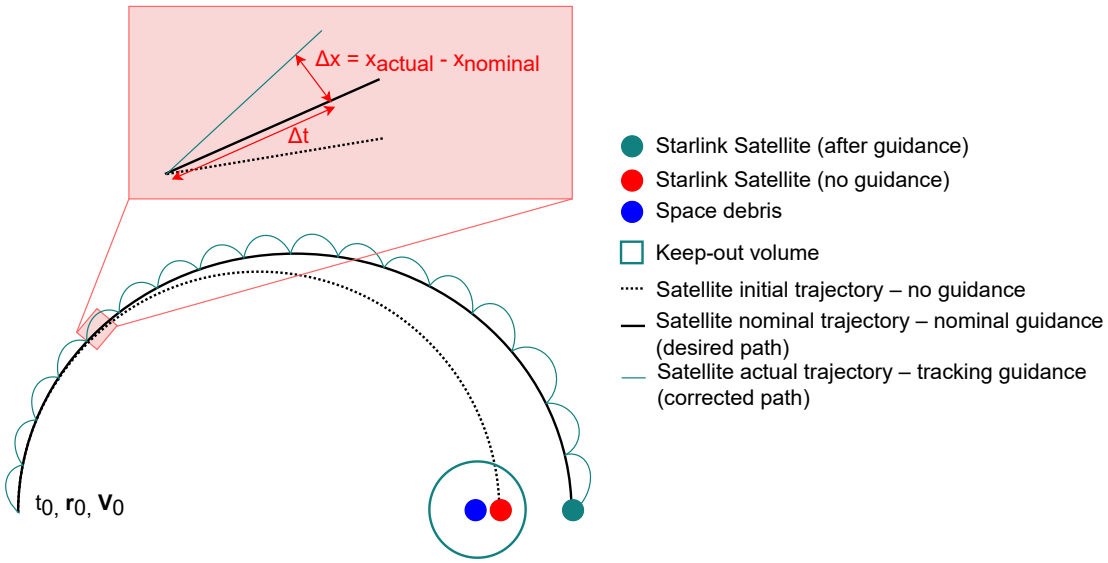

Figure 3. Overview of the guidance sequence, including the initial satellite trajectory, the nominal trajectory, and the corrected trajectory.

B. Flight Dynamics

The "real motion" of the satellite accounts for its non-linear dynamics in a perturbed environment. The perturbations taken into consideration are:

1. Earth's gravity, modelled as spherical harmonics up to degree and order 32,
2. Earth's induced gravitational and magnetic torques,
3. Moon's point mass (PM),
4. Sun's point mass,
5. Solar radiation pressure (SRP) and moment, and
6. Aerodynamic drag and moment.

However, the design of the convex-guidance system is centred around a simplified (linearised) model, such that the convex-model condition is met.

The starting point is Newton's second law of motion to describe an object's (elliptical) orbit. In this model, the influence of all forces and moments other than coming from a central gravity field are neglected, leading to a first approximation of the object's motion around Earth. Describing such motion between the object in question and the main body (Earth) in the *ECI* reference frame leads to:

$$\ddot{\boldsymbol{r}} = -\frac{\mu_E}{\|\boldsymbol{r}\|^3}\boldsymbol{r} \tag{1}$$

To accurately define the total acceleration of the space objects in orbit, all the perturbing forces ($\boldsymbol{F_{pert}}$) that are relevant should be taken into account. Adjusting Eq. (1) to accommodate the complexity of the problem at hand, gives

$$\ddot{\boldsymbol{r}} = \frac{\boldsymbol{F_{pert}}}{m} \tag{2}$$

Since mass will not be constant, because of the use of the satellite's propulsion system, an additional equation reflecting the mass change has to be added:

$$\dot{m} = -\frac{\|\boldsymbol{F_T}\|}{gI_{\text{sp}}} \tag{3}$$

In Eq. (2), the relevant perturbing forces for the problem at hand are Earth's gravitational force ($\boldsymbol{F_g}$), the aerodynamic drag ($\boldsymbol{F_{drag}}$), the third body attraction of the Moon ($\boldsymbol{F_{Moon}}$) and of the Sun ($\boldsymbol{F_{Sun}}$), as well as the solar radiation pressure ($\boldsymbol{F_{SRP}}$). In addition to the environmental perturbations, the disturbances induced by the propulsion system ($\boldsymbol{F_T}$) should be taken into account as well.

$$m\ddot{\boldsymbol{r}} = \boldsymbol{F_g} + \boldsymbol{F_{drag}} + \boldsymbol{F_{Moon}} + \boldsymbol{F_{Sun}} + \boldsymbol{F_{SRP}} + \boldsymbol{F_T} \tag{4}$$

It is noted that the acceleration due to SRP is calculated by assuming that the satellite is a box structure with a flat plate for the solar panel. This way of modelling provides a more accurate acceleration than the standard cannonball model.

III. Guidance and Control

To safely and efficiently perform collision avoidance manoeuvres, an onboard guidance and control system is essential. The guidance process determines the required ΔV (magnitude and direction), to enable the spacecraft to manoeuvre while meeting requirements for flight time, precision, fuel consumption, and risk minimisation. A robust guidance sequence, developed using convex optimisation, ensures optimal trajectories and propellant consumption.

Additionally, the attitude-control system, using a Linear Quadratic Regulator (LQR), autonomously adjusts the satellite's state to minimise the difference between desired and actual attitudes, ensuring precise and efficient manoeuvres. Both the nominal guidance and attitude control systems will be described.

A. Nominal Guidance

The nominal guidance will be introduced subsequently, with a special focus on the overall guidance framework. This will be followed by the convexification of the problem to ensure the application of convex optimisation, resulting in a well-defined convex problem.

1. *Problem Definition*

The primary aim of the guidance system is to execute a collision avoidance manoeuvre. This is achieved through convex optimisation, which ensures finding the global optimal solution. This feature makes convex optimisation particularly advantageous for nominal guidance applications. For the convex optimisation problem at hand, it has been decided, following the work of Ref. 5, to define it as a second-order cone programming (SOCP) problem. From previous applications, SOCP problems were found to be efficient to be used for this type of problem, they have low complexity, are robust, efficient, and can be solved in

Table 2. Definition of the non-convex problem, including the objective function, the constraints, and boundary conditions (BC).

Minimise:	$\int_0^{t_f} \|\boldsymbol{F_T}(t)\| \, \mathrm{d}t$	*Objective function*
subject to:	$\dot{\boldsymbol{r}}(t) = \boldsymbol{V}(t)$	*Orbital dynamics*
	$\ddot{\boldsymbol{r}}(t) = -\frac{\mu_E}{r^3(t)}\boldsymbol{r}(t) + \frac{\boldsymbol{F_T}(t)}{m(t)}$	
	$\dot{m}(t) = -\frac{\|\boldsymbol{F_T}(t)\|}{g_0 I_{\mathrm{sp}}}$	
	$\eta \, F_{T_{\min}} \leq F_{T_x}^{(-)}, \, F_{T_y}^{(-)}, \, F_{T_z}^{(-)} \leq 0$	*Thrust constraints*
	$0 \leq F_{T_x}^{(+)}, \, F_{T_y}^{(+)}, \, F_{T_z}^{(+)} \leq \eta \, F_{T_{\max}}$	
	$\|\boldsymbol{r}(t) - \boldsymbol{r_{deb}}(t)\| \geq \rho_{\mathrm{KOS}}$	*KOS constraint*
	$m(0) = m_0, \; \boldsymbol{r}(0) = \boldsymbol{r_0}, \; \dot{\boldsymbol{r}}(0) = \boldsymbol{\dot{r}_0}$	*Initial BC*
	$m(t_f) \geq m_f$	*Final BC*

polynomial time. This demonstrates their capacity to address real-world optimisation problems demanding swift convergence.

Consequently, the optimisation's objective is to determine a trajectory that minimises thrust expenditure over the entire mission duration, while still meeting mission objectives, to avoid collision with debris and move the satellite to meet the miss distance threshold. It is crucial during the optimisation process to ensure adherence to the equations of motion (orbital dynamics). These equations account for external accelerations affecting the spacecraft, namely Earth's gravity and the thrust applied as a control input, as well as the relationship governing mass flow. Aside from the dynamics, constraints are imposed on the thrust, and position vectors. Initially, limits must be established on the thrust force. This involves setting an upper and lower bound on the allowed nominal thrust, based on the engine's capabilities. Since the guidance sequence is divided into two segments, nominal and tracking, so should be the available thrust for each segment. To fulfil the CAM objective and adjust the satellite's trajectory to an acceptable state, it is necessary to establish a keep-out-sphere constraint. Furthermore, initial constraints must be imposed on the satellite's position, velocity, and mass, and a terminal constraint on the satellite mass, accounting for the satellite's available propellant. This outlines the optimisation challenge to be tackled by the satellite's guidance sequence. An overview of this guidance problem is presented in Table 2.

Upon examining the above-highlighted problem, it becomes apparent that the objective function is convex, due to the inclusion of the thrust vector's Euclidean norm. This also holds for the initial and terminal constraints, as they are affine, as well as the commanded thrust since the lower bound is zero. The remaining equations are all non-convex, which arise from the highly nonlinear dynamics of the satellite and the concave formulation of the keep-out sphere. To address this and enable the use of convex optimisation techniques, the problem must undergo a process known as convexification.

2. Problem Convexification

The current problem involves continuous-time nonlinear dynamics and non-convex control constraints. To address these complexities, a successive convex approximation is employed to efficiently solve optimal control problems featuring nonlinear dynamics and non-convex constraints. It operates on the principle of iteratively solving the original problem by successively linearising non-convex dynamics and constraints around the solution obtained in the previous iteration.

To use the dynamics model, Eqs.(2) and (3), in the design of the convex guidance system, a central field will be assumed before the system is linearised. Not doing so would overly complicate the model, which

leads to numerical issues of the solver used. The resulting state-space model is given by:

$$\Delta\dot{\boldsymbol{x}} = \boldsymbol{A_c}\Delta\boldsymbol{x} + \boldsymbol{B_c}\Delta\boldsymbol{u} \tag{5}$$

where the system matrix, $\boldsymbol{A}_c$, is obtained by taking the derivatives with respect to each state, and the control (or input) matrix, $\boldsymbol{B}_c$, can be found by taking the derivatives with respect to each control:

$$\boldsymbol{A_c} = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ -\frac{\mu_E}{\|r\|^3} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & -\frac{\mu_E}{\|r\|^3} & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -\frac{\mu_E}{\|r\|^3} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \tag{6}$$

$$\boldsymbol{B_c} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ \frac{1}{m} & 0 & 0 & \frac{1}{m} & 0 & 0 \\ 0 & \frac{1}{m} & 0 & 0 & \frac{1}{m} & 0 \\ 0 & 0 & \frac{1}{m} & 0 & 0 & \frac{1}{m} \\ \frac{1}{gI_{\text{sp}}} & \frac{1}{gI_{\text{sp}}} & \frac{1}{gI_{\text{sp}}} & -\frac{1}{gI_{\text{sp}}} & -\frac{1}{gI_{\text{sp}}} & -\frac{1}{gI_{\text{sp}}} \end{bmatrix} \tag{7}$$

where $\Delta\boldsymbol{x} = \begin{pmatrix} \Delta\boldsymbol{r}^T & \Delta\dot{\boldsymbol{r}}^T & \Delta m \end{pmatrix}^T \in \mathbb{R}^7$ and $\Delta\boldsymbol{u} = \begin{pmatrix} \Delta\boldsymbol{F}_T \end{pmatrix}^T \in \mathbb{R}^6$ are the state deviations and control increments, respectively. To simplify the problem and reduce non-linearity in the system dynamics, in the definition of $\mathbf{A_c}$, it is assumed that linearising $\mathbf{g}$ only yields a derivative in the main coordinate direction, and not any (off-diagonal) cross couplings in the other two directions. Also, the controls in negative and positive directions have been separated for ease of implementation.

Similarly, the KOS constraint is linearised about the reference trajectory point:

$$\|\boldsymbol{r_{ref}} - \boldsymbol{r_{deb}}\| + \frac{(\boldsymbol{r_{ref}} - \boldsymbol{r_{deb}})^\intercal}{\|\boldsymbol{r_{ref}} - \boldsymbol{r_{deb}}\|} \cdot (\boldsymbol{r_{opt}} - \boldsymbol{r_{ref}}) \geq \rho_{\text{KOS}} \tag{8}$$

with $\boldsymbol{r_{ref}}$ the reference trajectory position vector, $\boldsymbol{r_{deb}}$ the debris position vector and $\boldsymbol{r_{opt}}$ the optimal trajectory position vector defined by the optimiser.

Discretising the problem transforms the convex, continuous free-final-time problem with linearised dynamics and convex constraints into a convex, discrete free-final-time problem. This involves dividing the continuous time interval into discrete time steps, during which the state is iteratively computed. Consequently, a simplified approximation of the solution to the differential equations is obtained at these discrete time points. A zero-order hold sampling method is commonly used to convert a continuous-time linear system into an equivalent discrete-time system:

$$\boldsymbol{x_{k+1}} = \boldsymbol{A_d}\boldsymbol{x_k} + \boldsymbol{B_d}\boldsymbol{u_k} \tag{9}$$

Following the above process, the problem being fully convex, can be written as a proper SOCP problem and be solved. The definition of this SOCP problem is described in Table 3.

The problem can now be solved with a properly selected numerical algorithm, designed to solve SOCP problems. The selection of a generic solver is confined to those available in academic or open-source domains, which already offer a wide array of options. Following a trade-off, four were selected: Gurobi[g], ECOS[h] IPOPT[i], and fmincon[j]. In addition, a toolbox capable of interfacing with the SOCP solver in MATLAB®

[g]Gurobi Optimization, https://www.gurobi.com/, [Last Visited on 15/05/2024].
[h]ECOS, https://github.com/embotech/ecos, [Last Visited on 25/11/2024].
[i]IPOPT Documentation, GitHub Library, https://coin-or.github.io/IPOPT/, [Last Visited on 15/05/2024].
[j]fmincon documentation, MathWorkshttps://www.mathworks.com/help/optim/ug/fmincon.html, [Last Visited on 15/05/2024].

Table 3. Definition of the convex discrete problem, including the objective function, the constraints, and boundary conditions (BC).

Minimise:	$\int_0^{t_f} \|\boldsymbol{F_T}(t)\| \, \mathrm{d}t$	*Objective function*
subject to:	$\boldsymbol{x_{k+1}} = \boldsymbol{A_d x_k} + \boldsymbol{B_d u_k}$	*Orbital dynamics*
	$\eta \, F_{T_{\min}} \leq F_{T_x}^{(-)}, \, F_{T_y}^{(-)}, \, F_{T_z}^{(-)} \leq 0$	*Thrust constraints*
	$0 \leq F_{T_x}^{(+)}, \, F_{T_y}^{(+)}, \, F_{T_z}^{(+)} \leq \eta \, F_{T_{\max}}$	
	$\|\boldsymbol{r_{ref}} - \boldsymbol{r_{deb}}\| + \frac{(\boldsymbol{r_{ref}} - \boldsymbol{r_{deb}})^\intercal}{\|\boldsymbol{r_{ref}} - \boldsymbol{r_{deb}}\|} \cdot (\boldsymbol{r_{opt}} - \boldsymbol{r_{ref}}) \geq \rho_{\mathrm{KOS}}$	*KOS constraint*
	$m(0) = m_0, \quad \boldsymbol{r}(0) = \boldsymbol{r_0}, \quad \dot{\boldsymbol{r}}(0) = \dot{\boldsymbol{r}}_{\boldsymbol{0}}$	*Initial BC*
	$m(t_f) \geq m_f$	*Final BC*

is required. One such toolbox is YALMIP[k], renowned for its user-friendly interface and extensive support for various solvers. This wrapper facilitates seamless integration with a multitude of SOCP solvers and has been noted for its intuitiveness and ease of use [7].

B. Attitude Control

The guidance system introduced above produces a commanded thrust in three orthogonal directions, and has been defined in the inertial frame. The current formulation is a simplified representation of how the propulsion system is actually implemented on a satellite. It could be that there is only a single orbital engine, which requires that the complete satellite is rotated to the correct attitude to provide the manoeuvre thrust. Alternatively, the satellite may be equipped with sets of thrusters that are mounted orthogonally, allowing them to be operated independently from each other. In either case, the thrust will be defined in the satellite-fixed body frame. To prepare for a situation where the satellite has to be rotated to provide a thrust force in the right direction, in Sec. IV.C a preliminary attitude-control analysis will be discussed. The current section will provide the theoretical background for this.

The attitude control sequence uses both the angular rate and attitude error as inputs. The attitude error is determined by taking the difference between the satellite's measured state and the commanded state derived from the guidance commands. Specifically, when the guidance sequence provides a commanded thrust force in a body-fixed propulsion frame it may be converted a thrust magnitude (guidance command) and two attitude angles (ϵ and ψ). These angles relate the thrust direction to the satellite's attitude. The primary goal of the control system is to compute the commands necessary to ensure the current satellite's attitude aligns with the desired reference attitude. Specifically, the control system aims at minimising the error between the reference (desired) and actual (measured) attitude. To achieve this, a closed-loop control sequence is used, where the controller responds to the reference attitude and incorporates feedback from the system, providing real-time information about the satellite's current attitude. For attitude control, a linear state-feedback controller will be used, having the advantage of being computationally efficient, making it a great algorithm when real-time response is required. The corresponding gains are computed using optimal control theory, resulting in a controller known as LQR.

In the case of state feedback, the LQR control law can be defined as:

$$\Delta \boldsymbol{u} = -\boldsymbol{K} \Delta \boldsymbol{x} \tag{10}$$

in which $\boldsymbol{K}$ is the feedback gain matrix, $\Delta \boldsymbol{u}$ the corrective control input and $\Delta \boldsymbol{x}$ the state deviation vector.

[k]YALMIP MATLAB® toolbox, https://yalmip.github.io/, [Last Visited on 15/05/2024].

The LQR control technique takes as its objective the minimisation of a quadratic cost function, J, which simultaneously penalises the deviation from the desired object's state and the control inputs (thrust magnitude):

$$J = \int_0^\infty (\boldsymbol{x}^T \boldsymbol{Q} \boldsymbol{x} + \boldsymbol{u}^T \boldsymbol{R} \boldsymbol{u})\, \mathrm{d}t \tag{11}$$

with $\boldsymbol{x}^T \boldsymbol{Q} \boldsymbol{x}$ being the (weighted) state deviation and $\boldsymbol{u}^T \boldsymbol{R} \boldsymbol{u}$ the (weighted) control effort. $\boldsymbol{Q}$ and $\boldsymbol{R}$ can be given more or less weight depending on the desired performance. On one hand, $\boldsymbol{Q}$, the state cost matrix, will define how much importance is given to the system's specific state; some states might have to be controlled more tightly since they are more critical, they would thus have a higher weight in the cost function. On the other hand, $\boldsymbol{R}$, the control cost matrix defines the control effort used by each control input. The closed-loop system response is thus highly dependent on the selection of those parameters. Both $\boldsymbol{Q}$ and $\boldsymbol{R}$ matrices are often defined in an iterative approach but a good initial choice can be obtained using *Bryson's Rule*.[6] This rule adjusts the variables present in the LQR formulation to ensure that each term's maximum acceptable value is one. Using this rule, both matrices are defined as:

$$\boldsymbol{Q} = \mathrm{diag}\left(\frac{1}{\Delta x_{\max}^2} \quad \frac{1}{\Delta y_{\max}^2} \quad \frac{1}{\Delta z_{\max}^2}\right) \qquad \text{and} \qquad \boldsymbol{R} = \mathrm{diag}\left(\frac{1}{\Delta u_{\mathrm{x}}^2} \quad \frac{1}{\Delta u_{\mathrm{y}}^2} \quad \frac{1}{\Delta u_{\mathrm{z}}^2}\right) \tag{12}$$

Bryson's rule usually gives excellent results and is, therefore, a good first choice in defining $\boldsymbol{Q}$ and $\boldsymbol{R}$.

Using the above-defined matrices, the gain matrix $\boldsymbol{K}$ can thus be defined as follows.

$$\boldsymbol{K} = \boldsymbol{R}^{-1} \boldsymbol{B}^T \boldsymbol{P} \tag{13}$$

with $\boldsymbol{P}$ being a positive-definite matrix that follows from the solution of the Riccati equation:

$$\boldsymbol{A}^T \boldsymbol{P} + \boldsymbol{P} \boldsymbol{A} - \boldsymbol{P} \boldsymbol{B} \boldsymbol{R}^{-1} \boldsymbol{B}^T \boldsymbol{P} + \boldsymbol{Q} = \boldsymbol{0} \tag{14}$$

The $\boldsymbol{A}$ and $\boldsymbol{B}$ matrices can be defined, based on the attitude kinematics defined Sec II.B. These equations are non-linear and should therefore be linearised to be input in the model for the LQR controller. The LQR controller can be constructed in terms of quaternions or Euler angles, since both versions will be used in a trade-off in the results analysis, their descriptions will be introduced hereafter. The $\boldsymbol{A}$ and $\boldsymbol{B}$ matrices, and their respective state and control inputs, defined in terms of quaternions, are presented in Eq. (15)–(17), and in terms of Euler angles in Eq. (18)–(20), with c, s, t representing the cosine, sine, and tangent functions, respectively.

$$\boldsymbol{A_q} = \begin{bmatrix}
0 & \frac{I_{yy}-I_{zz}}{I_{xx}}\omega_{3_0} & \frac{I_{yy}-I_{zz}}{I_{xx}}\omega_{2_0} & 0 & 0 & 0 & 0 \\
\frac{I_{zz}-I_{xx}}{I_{yy}}\omega_{3_0} & 0 & \frac{I_{zz}-I_{xx}}{I_{yy}}\omega_{1_0} & 0 & 0 & 0 & 0 \\
\frac{I_{xx}-I_{yy}}{I_{zz}}\omega_{2_0} & \frac{I_{xx}-I_{yy}}{I_{zz}}\omega_{1_0} & 0 & 0 & 0 & 0 & 0 \\
\frac{q_{4_0}}{2} & \frac{-q_{3_0}}{2} & \frac{q_{2_0}}{2} & 0 & \frac{\omega_{3_0}}{2} & \frac{-\omega_{2_0}}{2} & \frac{\omega_{1_0}}{2} \\
\frac{q_{3_0}}{2} & \frac{q_{4_0}}{2} & \frac{-q_{1_0}}{2} & \frac{-\omega_{3_0}}{2} & 0 & \frac{\omega_{1_0}}{2} & \frac{\omega_{2_0}}{2} \\
\frac{-q_{2_0}}{2} & \frac{q_{1_0}}{2} & \frac{q_{4_0}}{2} & \frac{\omega_{2_0}}{2} & \frac{-\omega_{1_0}}{2} & 0 & \frac{\omega_{3_0}}{2} \\
\frac{-q_{1_0}}{2} & \frac{-q_{2_0}}{2} & \frac{-q_{3_0}}{2} & \frac{-\omega_{1_0}}{2} & \frac{-\omega_{2_0}}{2} & \frac{-\omega_{3_0}}{2} & 0
\end{bmatrix} \tag{15}$$

$$\boldsymbol{B_q} = \begin{bmatrix}
\frac{1}{I_{xx}} & 0 & 0 \\
0 & \frac{1}{I_{yy}} & 0 \\
0 & 0 & \frac{1}{I_{zz}} \\
0 & 0 & 0 \\
0 & 0 & 0 \\
0 & 0 & 0 \\
0 & 0 & 0
\end{bmatrix} \tag{16}$$

$$\Delta \boldsymbol{x} = \begin{pmatrix} \Delta\boldsymbol{\omega} & \Delta\boldsymbol{q} \end{pmatrix}^T \in \mathbb{R}^7 \qquad \Delta \boldsymbol{u} = \begin{pmatrix} \Delta \boldsymbol{M_c} \end{pmatrix}^T \in \mathbb{R}^3 \tag{17}$$

Only three of the four quaternions have to be controlled, since the fourth quaternion component automatically follows from the unit-norm constraint. For this reason, q_4 is removed from the controller design, leading to a 6x6 $\boldsymbol{A_q}$ matrix and 6x3 $\boldsymbol{B_q}$ matrix.

$$\boldsymbol{A_{Euler}} = \begin{bmatrix} 0 & \frac{I_{yy}-I_{zz}}{I_{xx}}\omega_{3_0} & \frac{I_{yy}-I_{zz}}{I_{xx}}\omega_{2_0} & 0 & 0 & 0 \\ \frac{I_{zz}-I_{xx}}{I_{yy}}\omega_{3_0} & 0 & \frac{I_{zz}-I_{xx}}{I_{yy}}\omega_{1_0} & 0 & 0 & 0 \\ \frac{I_{xx}-I_{yy}}{I_{zz}}\omega_{2_0} & \frac{I_{xx}-I_{yy}}{I_{zz}}\omega_{1_0} & 0 & 0 & 0 & 0 \\ 1 & s\phi t\theta & c\phi t\theta & c\phi t\theta\omega_{2_0} - s\phi t\theta\omega_{3_0} & \frac{s\phi\omega_{2_0}+s\phi\omega_{3_0}}{c\theta^2} & 0 \\ 0 & c\phi & -s\phi & -s\phi\omega_{2_0} - c\phi\omega_{3_0} & 0 & 0 \\ 0 & \frac{s\phi}{c\theta} & \frac{c\phi}{c\theta} & \frac{c\phi\omega_{2_0}-s\phi\omega_{3_0}}{c\theta} & \frac{s\phi\omega_{2_0}+c\phi\omega_{3_0}}{c\theta} & 0 \end{bmatrix} \tag{18}$$

$$\boldsymbol{B_{Euler}} = \begin{bmatrix} \frac{1}{I_{xx}} & 0 & 0 \\ 0 & \frac{1}{I_{yy}} & 0 \\ 0 & 0 & \frac{1}{I_{zz}} \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \tag{19}$$

$$\Delta \boldsymbol{x} = \begin{pmatrix} \Delta\boldsymbol{\omega} & \Delta\phi & \Delta\theta & \Delta\psi \end{pmatrix}^T \in \mathbb{R}^7 \qquad \Delta \boldsymbol{u} = \begin{pmatrix} \Delta \boldsymbol{M_c} \end{pmatrix}^T \in \mathbb{R}^3 \tag{20}$$

IV. Results

This chapter presents selected verification tests and results from both the guidance and control sequence resulting from various simulations and test runs. These findings provide insight into the performance of the developed systems.

A. Verification

The different verification tests that have been performed will be highlighted in this section. These include verification by comparing the current sequence with the well-known Hohmann transfer, but also verification on the different available solvers.

1. Hohmann Transfer Comparison

To verify the proper functioning of the convex guidance, the optimal solution is compared with a traditional (non-impulsive) Hohmann transfer. From theory, it is known that the Hohmann transfer is performed in the pericentre of the orbit. It is therefore not a versatile method, as it necessitates execution at least half an orbit before conjunction. However, the Hohmann transfer can serve as a baseline for verification purposes.

Both the thrust and mass profile of a non-impulsive Hohmann and of convex optimisation transfer will be compared. It has been chosen to simulate a non-impulsive Hohmann transfer since in real life impulsive manoeuvres are not possible. This allows both mass and thrust profiles, using the same satellite characteristics to be compared with the nominal convex ones. The manoeuvre will be performed for both cases with a nominal available thrust of 10 N in all directions, an I_{sp} of 292 s for a central-field, circular, and equatorial orbit, and a manoeuvre to move 10 km at apogee. The results of this analysis can be

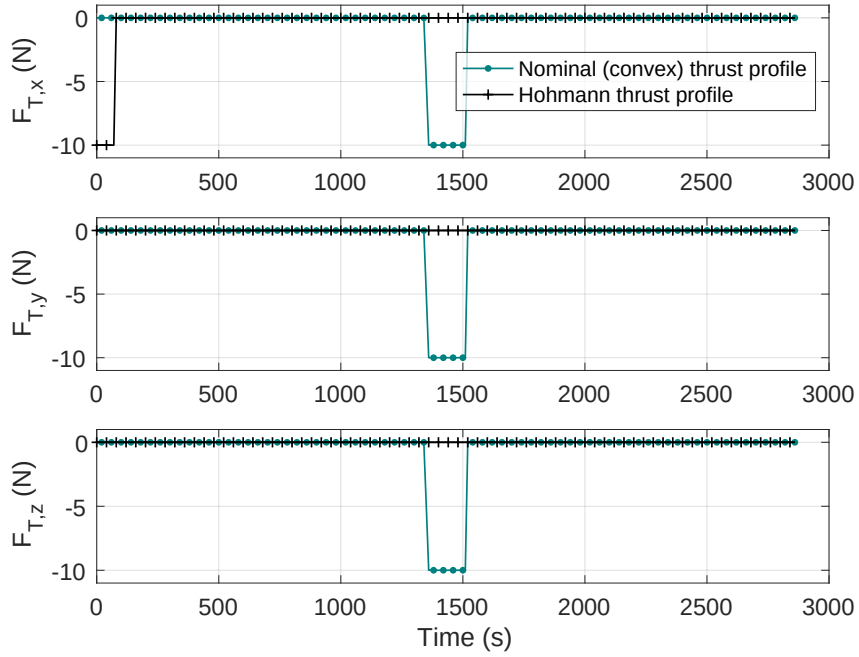

Figure 4. Thrust profile nominal (convex) vs. Hohmann manoeuvre.

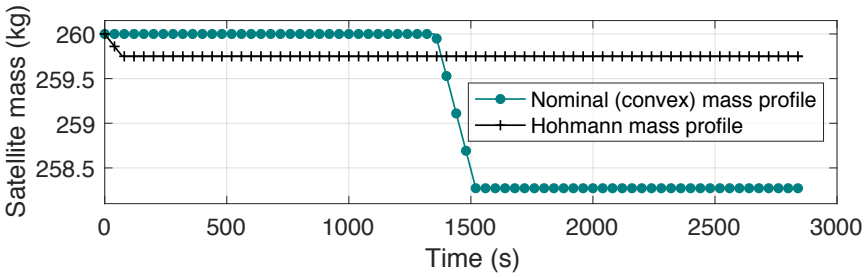

Figure 5. Propellant mass profile nominal (convex) vs. Hohmann manoeuvre.

observed in Figs. 4 and 5. Upon analysis, it becomes evident that the thrusting periods do not coincide. Examination of the mass and thrust profiles reveals that the Hohmann transfer indeed results in a more optimal profile, characterised by a shorter thrust period and reduced propellant mass consumption. Despite being larger and occurring at a different time, both the thrust and mass profiles exhibit behaviour akin to the Hohmann profile, displaying a "bang-bang" control pattern. The required mass for convex guidance amounts to 1.73 kg, surpassing that of Hohmann, yet deemed satisfactory, given the nature of the Hohmann transfer. From this analysis, the unexpected thrust in the Z-direction, despite the equatorial orbit, occurs because the optimisation makes use of 3D motion, unlike the Hohmann transfer. By allowing small out-of-plane manoeuvres, the optimiser can more effectively balance changes in orbital elements, leading to a reduction in overall ΔV. Nonetheless, it remains intriguing why the timing of the manoeuvre is different. As said, a Hohmann transfer takes place in the pericentre, which would be, in the current scenario, at t_0. However, the Hohmann transfer is an idealised, two-dimensional problem, whereas the current implementation is a three-dimensional one. It is left as future work to study the sensitivity of the optimiser to three-dimensional dynamics. This could be achieved by adding additional constraints to prevent out-of-orbital-plane excursions, and even a manoeuvre-time constraint could be added. By comparing the accuracy of the manoeuvres and the propellant mass spent, more insight into this difference can be obtained.

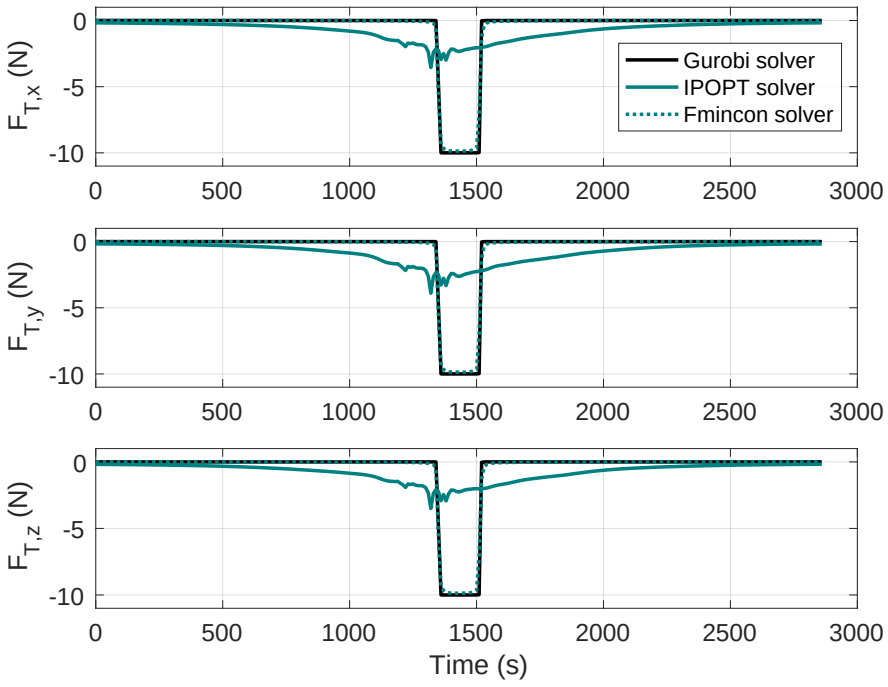

Figure 6. Variation in thrust behaviour between different solvers for an equatorial orbit.

2. Solver Comparison

Another verification test will be performed on the use of different solvers. This will guarantee that the optimum solution found through convex optimisation is not dependent on the chosen solver and that any solver would output similar results. To perform such a verification step, it has been decided to compare the behaviour of the Gurobi, IPOPT, fmincon, and ECOS solvers in the YALMIP environment. Based on this analysis, it can be inferred that all four simulations suggest that the most effective approach for executing such a manoeuvre involves initiating thrust at approximately the halfway point, exhibiting similar durations of maximum thrust behaviour. This observation ensures that the optimised trajectory is indeed the appropriate one, with a thrust boost occurring around 1,400 s into the simulation. It was observed that both the Gurobi and ECOS solvers exhibit the same behaviour (for this reason ECOS has not been plotted). However, from the analysis, some discrepancies are worth noting. To start with, the simulation time is not comparable, while the Gurobi and ECOS solvers find a solution to the problem in a few seconds, a few minutes ($\approx$ 10 min) is necessary to get an output from IPOPT, with a fixed tolerance put to 10^{-6} to allow for faster solving. For fmincon, the CPU is approximately 6 hours (tolerance put to 10^{-3}), making it a very slow solver that cannot be used onboard the satellite. In addition to this, by looking at Fig. 6, it can be observed that the thrust profile from the IPOPT solver does not exhibit "bang-bang" profiles, and, as a result, leads to a larger fuel consumption as observed in Fig. 7. This smoother behaviour could be positive as input to the control system since it will avoid overshoots, but does not meet the main objective of the CAM sequence; to be fuel efficient. Overall, the mass profiles of the fmincon, Gurobi and ECOS solvers are similar, with a difference in thrust usage of 8.2 g (fmincon vs. Gurobi). The IPOPT solver mass consumption exceeds the other three solvers by approximately 0.99 kg.

In addition to verifying and testing the behaviour of different solvers, the tolerance difference using the IPOPT solver has also been analysed. Three different tolerances have been applied for the solver, 10^{-3}, 10^{-6} and 10^{-9}. From this analysis, it was concluded that no significant differences were observed from changing the tolerances, with the effect on the mass consumption also being negligible, in the order of a few grams. The only significant difference was in the computational time, where a smaller tolerance resulted in a longer CPU time, as expected.

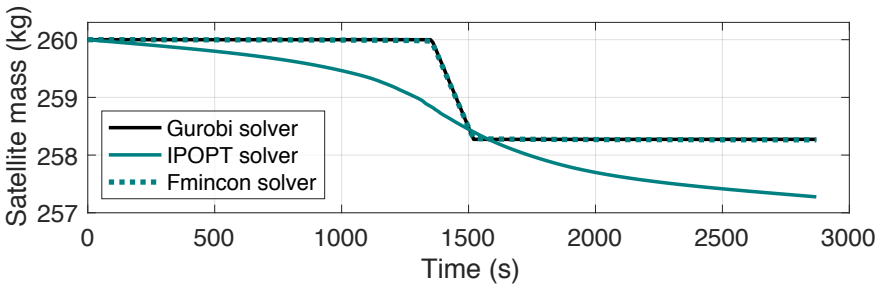

Figure 7. Variation in propellant usage between different solvers for an equatorial orbit.

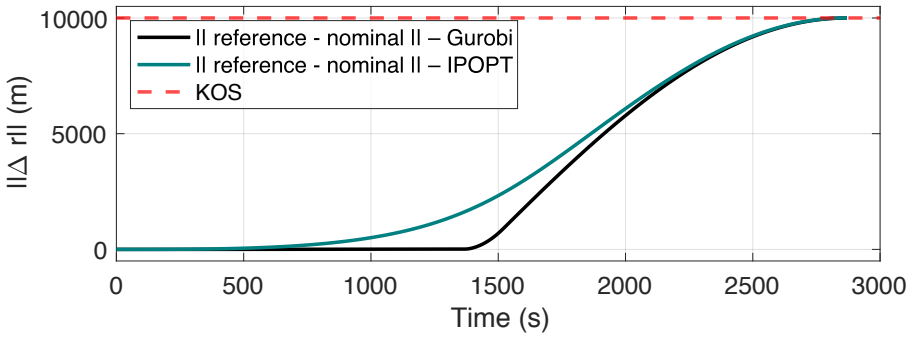

Figure 8. Overview of the position norm between the nominal and reference trajectory to evaluate how the KOS requirement is met for both IPOPT and Gurobi solvers. This analysis is performed using the following thruster's characteristics: $F_{T_{nom}} = 10$ N, $I_{sp} = 292$ s – Profile from the linearised and discretised model.

B. Perturbed Orbits

This section will highlight various results from the nominal guidance, including the nominal profile, manoeuvre period, propulsion limitations, and the effects of perturbations.

1. Nominal Profile

From the previous section, it is concluded that the Gurobi (or ECOS) solver is the most efficient to use, yielding a decent optimal solution. Nonetheless, the problem will also be solved with IPOPT, to obtain a smoother profile for future control system analysis. The previously discussed results were obtained for a circular, equatorial orbit. In the current section, the effect of several perturbations on the CAM will be analysed but now for the actual inclined Starlink orbit.

The difference between the linearised reference and optimised trajectory has been analysed to observe how the satellite deviates from its initial trajectory, to perform the manoeuvre and stay outside the KOS at the end of the trajectory, thus avoiding the debris. In Fig. 8, the difference in norm position for both IPOPT and Gurobi solver is shown. This result confirms that the KOS constraint is adhered to and that the satellite successively avoids the debris. When simulating for the optimised thrust profiles, it was observed that indeed a miss distance of 10.12 km was met after 2,867 s in the trajectory, as expected and required.

Next, the optimal thrust profile generated by the convex solver is input into the nonlinear simulation to verify that the CAM is consistent across both implementations. The results confirm this consistency within the acceptable tolerance range.

2. Manoeuvre Period Analysis

The analysis conducted examined the impact of reducing the warning time before manoeuvring and its effect on the optimised trajectory. The approach involved shortening the trajectory based on its initial true anomaly (θ_0). In the case of the initial half-orbit scenario, θ_0 was 0°. The analysis explored various θ_0 within

Table 4. Different characteristics for different manoeuvring periods.

Cases	θ_0	Simulation period (s)	Simulation time (s)
Case 1	0°	0 – 2880	2880
Case 2	45°	720 – 2880	2160
Case 3	90°	1440 – 2880	1440
Case 4	135°	2160 – 2880	720
Case 5	145°	2310 – 2880	470

the ranges of [0°, 45°, 90°, 135°, 145°], resulting in each trajectory starting with a different true anomaly and having less time to manoeuvre before encountering debris (located at $\theta_0 = 180°$).

The findings of this analysis are presented in this section and will be examined sequentially. For visualisation purposes, a uniform time frame was maintained for all trajectories. Although each trajectory theoretically starts at t = 0 with varying trajectory times, the decision was made to align everything with the longest trajectory (beginning at $\theta = 0°$) to aid visualisation and comprehension. However, it is essential to recognise that each simulation starts at a distinct time, as pointed out throughout the analysis. All five cases that have been analysed for this study have been gathered in Table 4. Optimisation for a warning time smaller than that of Case 5 results in failure under the specific constraints and objectives employed in this simulation. This failure stems from the need for either greater thrust capabilities or tighter miss distance requirements to fulfil the criteria within such abbreviated time frames. The results thus expose the limitations linked with the problem's parameters.

The convex guidance sequence has been simulated for all five cases. To start with, the thrust profiles for each case have been analysed. It must be noted that only the thrust behaviour in the X direction has been displayed for conciseness, since as has been observed earlier, the thrust profile is similar in all three directions. As depicted in Fig. 9, it is evident that all thrust profiles exhibit consistency, with the thrusting sequence increasing as the warning time decreases. This trend can be explained by the increased necessity for a larger ΔV to achieve the desired miss distance sequence when the satellite is in closer proximity to the target. This characteristic is further reflected in the corresponding mass consumption profiles (Fig. 10), where it is observable that propellant consumption rises as the warning time decreases, mirroring the thrust profile trend. Notably, the disparity between the two extreme cases, starting with a $\theta_0 = 0°$ and $\theta_0 = 145°$, results in a substantial difference in propellant mass usage of approximately 3.50 kg. This discrepancy is significant and cannot be neglected when searching for the optimum CAM strategy. Directly correlated with both the mass and thrust profiles, it becomes apparent that across all five cases, the 10 km miss distance requirement is achieved (Fig. 11). Each trajectory profile is optimised according to the constraints specific to each case. Notably, for shorter warning times, the manoeuvres appear more abrupt and direct, requiring larger resources within a constrained time frame.

3. Thrust-Capability Limitations

To ensure the robustness of the current CAM algorithm, it must demonstrate effectiveness not only within the specific propulsion system it was initially designed for but across multiple systems. To comprehensively assess the CAM algorithm's performance, testing will encompass high-thrust propulsion systems like cold gas, solid motor, mono-propellant, and bi-propellant. Additionally, evaluations will include low-thrust options like hall-thrusters. Furthermore, each system will undergo testing across various thrusting capabilities, ensuring a thorough examination of the algorithm's versatility. The different propulsion systems, together with the tested characteristics have been listed in Table 5. For each combination, the miss distance is met and the optimisation is successful. It is observed that cold gas exhibits larger mass consumption, for the tested scenario. In contrast, the low-thrust propulsion exhibits the lowest mass consumption of all, still meeting the miss distance, and has a constant thrust for the entire trajectory. With these results, this profile makes it the most optimum, leading to the conclusion that low-thrust propulsion would be the future of low-Earth orbit manoeuvring propulsion, provided no rapid (= high thrust) response is required and the propulsive manoeuvre does not interfere with the primary mission objectives.

The analysis further suggests that, despite the widespread use of chemical engines for their ability to rapidly alter a spacecraft's motion, they may not be the most optimal choice for the specific CAM under investigation. Although well-established and reliable, they necessitate significant propellant quantities and

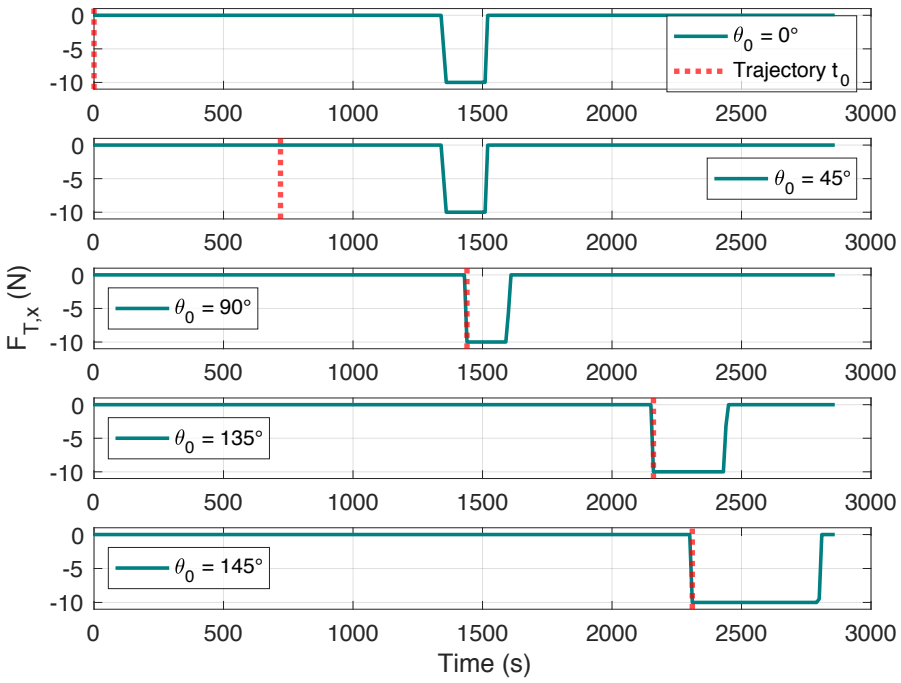

Figure 9. Thrust profile in the X direction for different manoeuvring starting periods.

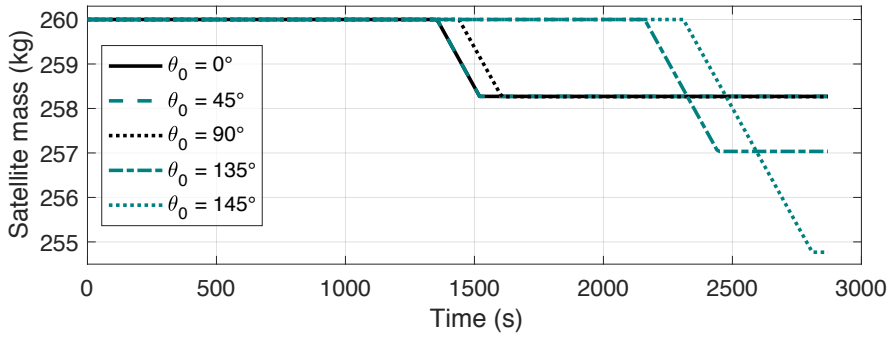

Figure 10. Satellite mass variation for different manoeuvring starting periods.

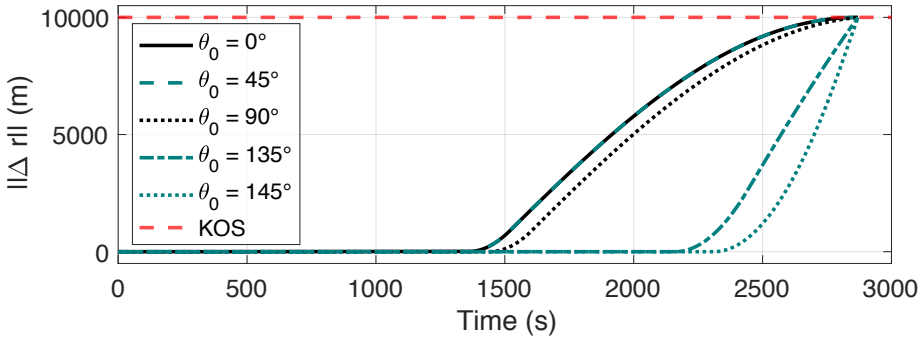

Figure 11. Position norm between the nominal and reference trajectory for different manoeuvring starting periods – Profile from the linearised and discretised model.

Table 5. Mass consumption for each propulsion type and studied characteristics.

Propulsion type	Vacuum I_{sp} (s)	Thrust (N)	Δm (kg)
Cold gas	50	1	12.6
Cold gas	50	200	10.08
Solid motor	300	50	1.68
Solid motor	300	10^6	1.68
Mono-propellant	200	0.1	0.4388
Mono-propellant	200	5	2.53
Bi-propellant	400	10	1.26
Bi-propellant	400	10^6	1.26
Hall-thruster	2000	0.01	0.0044
Hall-thruster	2000	0.1	0.042

Table 6. Effect of the environmental perturbations on the satellite trajectory by looking at its deviation from the debris, which should be 10 km in the nominal case.

Perturbations	CF	SH 32/32	Aero. drag	SRP	Moon PM	Sun PM	All
Miss distance (m)	10,129	23,447	10,134	10,129	10,131	10,130	23,455

emit substantial fuel during bursts, contradicting current technological trends towards greener, more eco-friendly space systems. An alternative propulsion method involves continuous thrusting, using more mass-efficient low-thrust systems. These systems, characterised by lower thrust outputs compared to impulsive systems, offer higher exhaust velocities while minimising propellant consumption. The Starlink constellation, which is the one used as a reference case for this study, is making use of low-thrust propulsion, through its ion thrusters (Krypton-propelled, hall-effected thrusters).

4. Perturbations Effects

The impact of environmental perturbations should be analysed next, since their influence on a LEO satellite cannot be ignored The convex guidance, which issues commands to follow a nominal trajectory, has been developed assuming a central gravity field only. Consequently, the effects of other perturbations have not been considered. From Table 6, which shows the variation in Kepler elements between a central field manoeuvre and a manoeuvre that accounts for all relevant perturbation models, it is evident that the impact is substantial, leading to a miss distance of 23.45 km.

The same analysis has been performed by switching on the relevant perturbations included in this research work one by one, and looking at the effects each one had on the final miss distance. One could debate the necessity of tracking guidance to ensure the perturbed trajectory closely follows the nominal path. An analysis with an LQR tracking guidance showed that this led to an exorbitant fuel consumption. It would be better to have onboard predictive capabilities, provided this can be confirmed with additional measurements, such that the CAM can be re-optimised. Alternatively, one could re-run the convex optimisation (on-board) to have one (or more) corrective manoeuvres to reduce the miss distance. Provided, of course, that an accurate state estimate is available. However, these two options are left as future work.

C. Preliminary Attitude Control

The LQR attitude controller can be described both in terms of quaternions and Euler angles. Both controllers have been tested, in which the required thrust angles have been converted to an attitude change. The preliminary analysis for a commanded attitude of 45° for both θ and ψ (*i.e.,* pitch and yaw Euler angles), can be visualised in Fig 12. Table 7 highlights the settings used to achieve this response. The results show that the required attitude change for correct thrust orientation must be completed before the thrust manoeuvre, as the time to achieve the commanded attitude significantly exceeds the manoeuvre duration. Reorientation takes approximately 70 seconds with a quaternion-based controller and 50 seconds with an Euler angle-based controller, while the nominal guidance manoeuvre under study ($F_{T_{nom}} = 10$ N, $I_{sp} = 292$, Gurobi solver) lasts around 100 seconds, as shown in Fig. 6. This comparison underscores a critical

Table 7. Tuning parameters for the LQR system in terms of Euler angles, for a commanded attitude change of $\phi = 0°$, $\theta = 45°$, $\psi = 45°$.

$\Delta\omega_1$	$\Delta\omega_2$	$\Delta\omega_3$	$\Delta\phi$	$\Delta\theta$	$\Delta\psi$	$M_{\mathbf{min/max}}$
5°	5°	5°	0.1	0.2	0.2	± 10 (N)

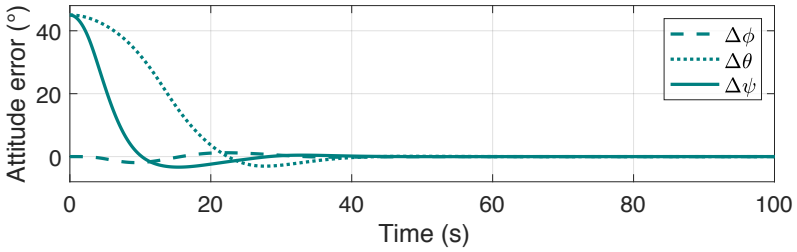

Figure 12. LQR controller response, in terms of Euler angles, for a commanded attitude change of $\phi = 0°$, $\theta = 45°$, $\psi = 45°$. Settling time ≈ 50 seconds.

limitation: if attitude control and guidance are simultaneous, the attitude adjustment phase consumes most of the manoeuvre window, making it non-viable. To address this, the satellite must reorient to the desired attitude prior to initiating the nominal guidance sequence.

V. Conclusions and Recommendations

Convex optimisation is used to determine the optimal path under various constraints. Known for its robustness, autonomy, and computational efficiency, it ensures convergence to the optimal solution within a fixed number of iterations, if a feasible solution exists. The nominal guidance system, thoroughly developed and tested, showed excellent results in several metrics. Firstly, it calculated precise trajectories, ensuring efficient collision avoidance. Secondly, it demonstrated high computational efficiency, quickly solving the optimisation sequence for real-time applications. The system is also designed to minimise fuel consumption, enhancing mission sustainability. Convex optimisation's deterministic nature guarantees consistent and dependable performance, crucial for missions where unexpected deviations can occur. Finally, the adaptable guidance sequence can be tailored to diverse mission scenarios, making it a valuable tool for the industry. Overall, this guidance system represents a significant advancement in satellite collision avoidance, enhancing space situational awareness and long-term sustainability of space operations.

Some recommendations following the research work can be highlighted. Further (theoretical) analysis is required to understand the difference between the obtained manoeuvres and the Hohmann transfer. Further development of the attitude controller seems logical, as well as the subsequent integration with a(n autonomous) navigation system. Analysing the performance of a complete guidance, navigation, and control system will improve our understanding of its robustness.

Some mission aspects have been left as future research. To start with, an analysis of the post-manoeuvring sequence could be performed. Currently, the mission scenario ends after debris avoidance. Companies would want their vehicle to return to its initial orbit to continue the mission. This is particularly relevant for constellation satellites, which need to maintain their precise formations. Therefore, it is recommended to study the post-manoeuvre phase, exploring scenarios such as continuing on the new orbit, returning to the original orbit, or transitioning to a different orbit. Each scenario's risks and benefits should be analysed. Additionally, the use of the algorithm for both the avoidance manoeuvre and return should be examined for its impact on fuel consumption, mission duration, and collision risk.

A final recommendation is to analyse the future collision probability with the same debris after an avoidance manoeuvre. This analysis should include different manoeuvring sequences and assess the impact of earlier manoeuvres. By studying various manoeuvring strategies and their timing, a better understanding of how to minimise future collision risks and optimise overall mission safety can be gained. This approach will help identify the most effective manoeuvre sequences and timings to ensure the long-term sustainability and safety of space missions.

References

[1]Buchs, R., "Collision risk from space debris: Current status, challenges and response strategies". Lausanne: EPFL International Risk Governance Center, 2021.

[2]The European Space Agency - ESA Space Debris Office, "ESA's Annual Space Environment Report", Version 8.0, 2024

[3]Zhang, J. and Cai, Y. and Xue, C. and Xue, Z. and Cai, H., "LEO Mega Constellations: Review of Development, Impact, Surveillance, and Governance", *Space: Science and Technology*, 2022.

[4]Kessler, D.J. and Johnson, N.L. and Liou, J.-C. and Matney, M., "The Kessler Syndrome: Implications to Future Space Operations", *33rd Rocky Mountain Guidance and Control Conference*, pp.47-62, 2010.

[5]Açıkmeşe, B. and Ploen, S.R., "Convex Programming Approach to Powered Descent Guidance for Mars Landing", *Journal of Guidance, Control, and Dynamics*, Vol. 30, Nr. 5, pp. 1353-1363, 2007.

[6]Bryson, A.E. and Ho, Y.C., *Applied Optimal Control: Optimization, Estimation, and Control*, Taylor and Francis Group, 1975.

[7]Lofberg, J., "YALMIP: a toolbox for modelling and optimization in MATLAB", *2004 IEEE International Conference on Robotics and Automation (IEEE Cat. No.04CH37508)*, pp.284-289, 2004