POLITECNICO
MILANO 1863

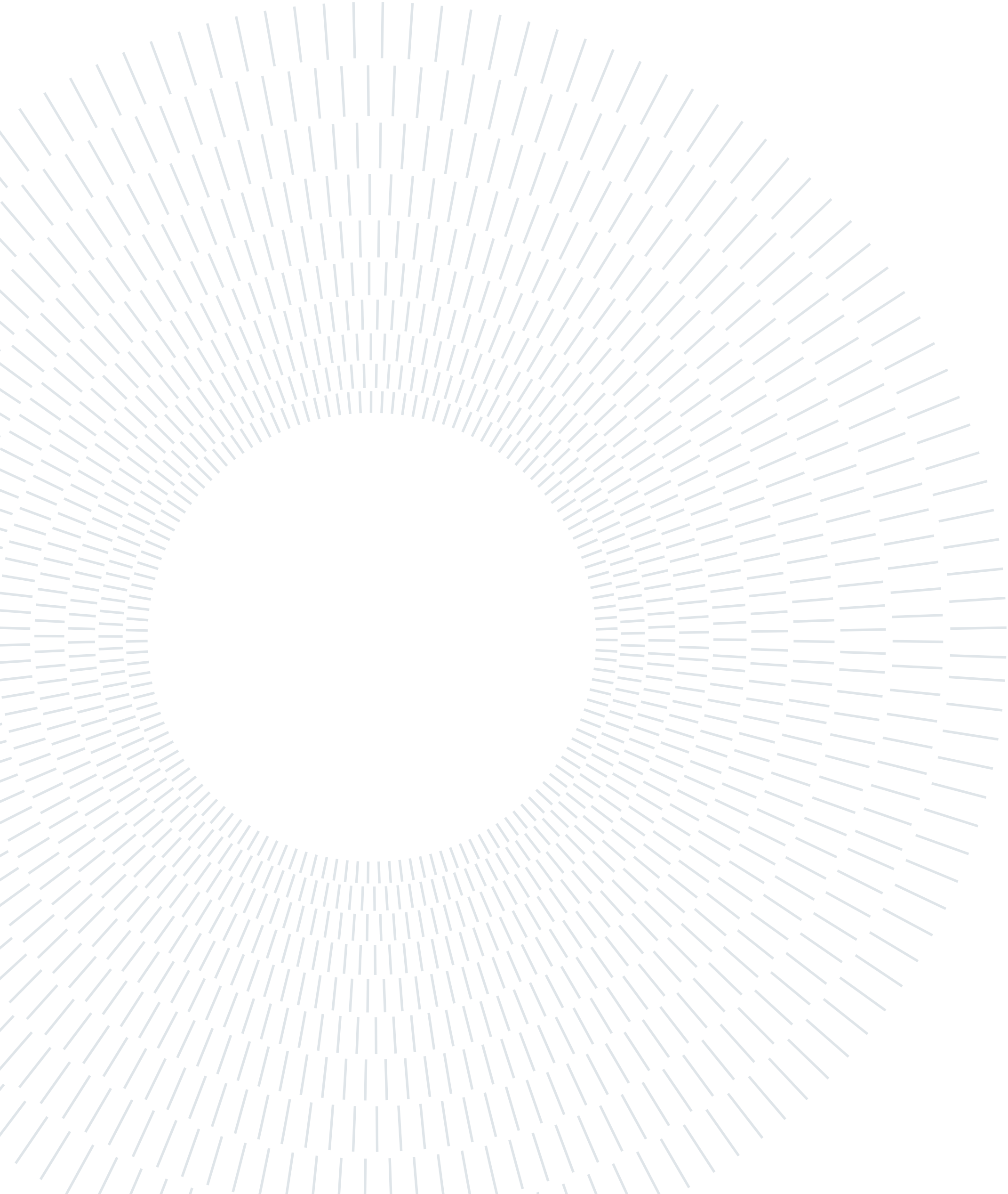
SCUOLA DI INGEGNERIA INDUSTRIALE
E DELL'INFORMAZIONE

Adaptive Guidance and Control for Formation Flying in High-Drag Low-Orbit Environment

Tesi di Laurea Magistrale in
Space Engineering - Ingegneria Spaziale

Author: **Enrico Belloni**

Student ID: 939996
Advisor: Prof. Michèle Lavagna
Co-advisors: Stefano Silvestrini, Jacopo Prinetto
Academic Year: 2021-22



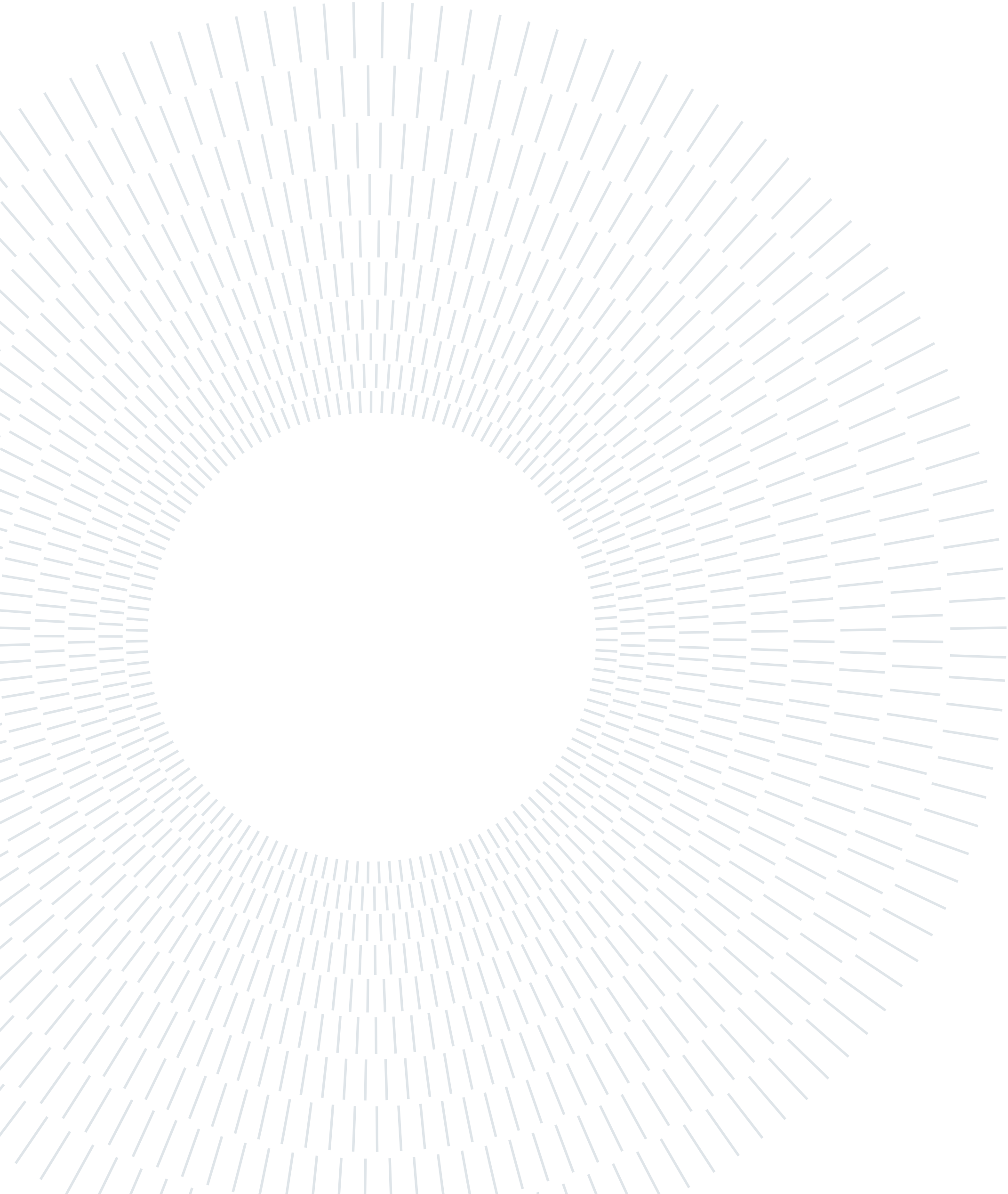
Abstract

In this document, a novel Model Predictive Control (MPC) technique for multi-satellite formation flying geometry acquisition and maintenance in low-orbit high-drag environment is presented. The proposed controller optimizes fuel efficiency and tracking accuracy by exploiting the knowledge of the formation dynamics. In particular, the MPC relies on a linearized and convexified quasi-nonsingular Relative Orbital Elements (ROE) model based on state transition matrices propagation, allowing to include the effect of perturbations in the prediction. The formation is controlled with respect to a non-decaying orbiting point to perform absolute and relative station keeping simultaneously. For this purpose, a dedicated plant matrix to include drag effects in the propagation is derived and validated with respect to numerical results. The linearization and the convexification of the model allow the use of fast and efficient solvers, viable for autonomous control and on-board implementation.

In all simulations, the satellites are assumed to be equipped with a single low-thrust propulsion unit, therefore, specific constraints are included in the controller to obtain a feasible solution in a real operational scenario. Moreover, a collision avoidance constraint is added in case of close proximity operations to avoid collisions, exploiting a linear mapping between the set of ROE and cartesian coordinates expressed in the Local-Vertical-Local-Horizontal (LVLH) reference frame.

The controller response is simulated in several realistic mission contexts with a high-fidelity orbital propagator and the results are validated both for fuel efficiency and collision avoidance effectiveness. To understand if the provided solutions are sub-optimal in terms of fuel consumption, the results are compared to similar approaches available in literature and to optimal solutions obtained respectively with a direct single shooting algorithm and with a closed-form impulsive formulation.

Keywords: autonomous guidance and control; formation flying; low earth orbit; model predictive control; relative orbital dynamics; relative orbital elements



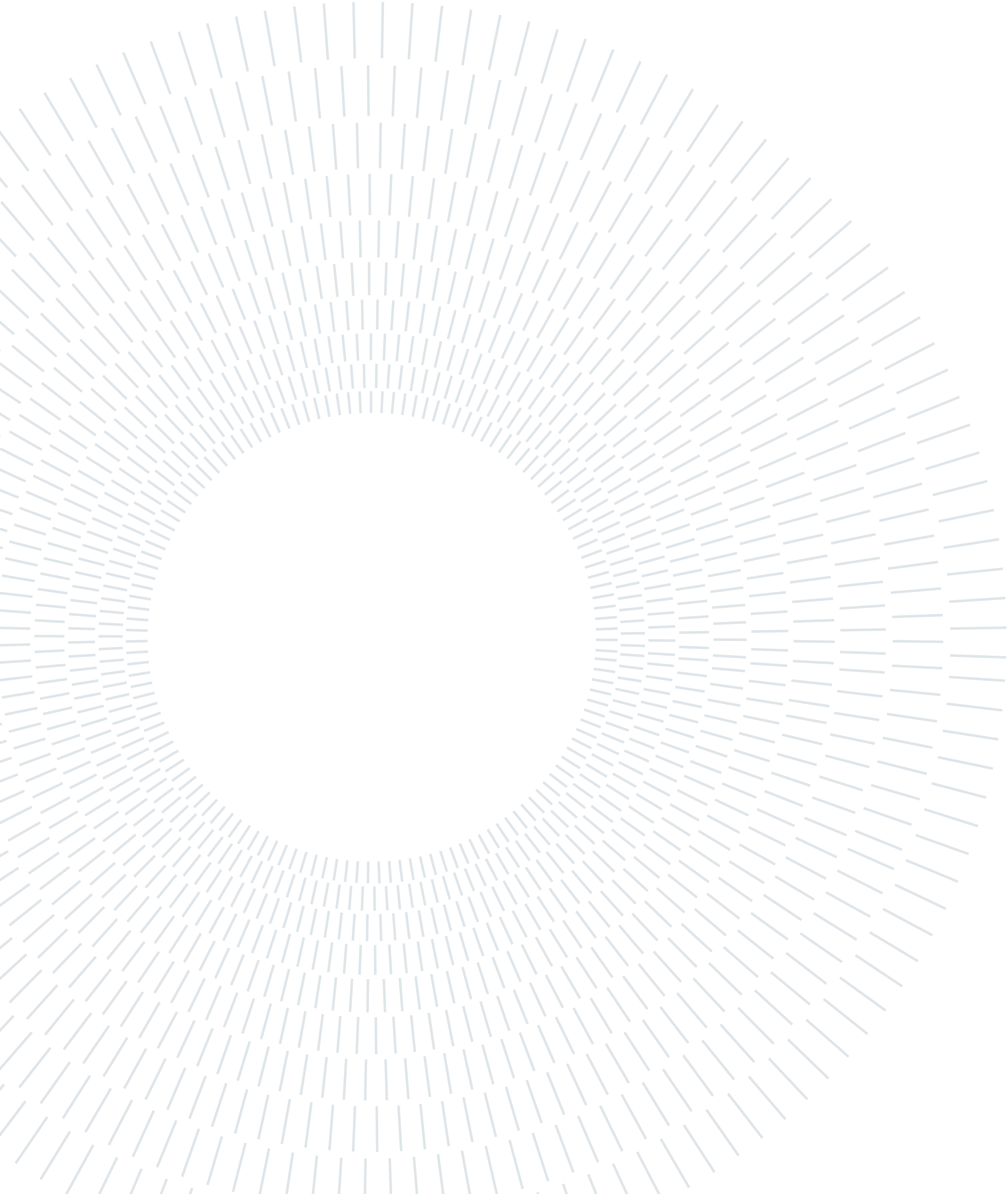
Abstract in lingua italiana

In questo documento è proposta una nuova tecnica di Model Predictive Control (MPC) per acquisizione e mantenimento di geometria relativa di satelliti in missioni di formation flying (FF) in bassa orbita terrestre. Il sistema di controllo ottimizza il consumo di propellente e la precisione di tracciamento sfruttando la conoscenza della dinamica del sistema. In particolare, il controllo predittivo si basa su un modello linearizzato e convessificato della dinamica espressa in Elementi Orbitali Relativi (ROE) propagata con matrici di transizione di stato, permettendo una facile inclusione degli effetti dovuti a perturbazioni orbitali nella propagazione. La formazione è controllata rispetto a un punto orbitante ad altitudine costante, in modo da mantenere lo stato assoluto e relativo allo stesso tempo. A tale scopo è formulata una matrice dinamica che permette di includere gli effetti della resistenza aerodinamica nella propagazione lineare, i cui risultati sono validati rispetto ad una propagazione numerica. La linearizzazione e la convessificazione del modello permettono l'utilizzo di algoritmi di ottimizzazione convessa rapidi ed efficienti, adatti a controllo autonomo e implementazione a bordo.

In tutte le simulazioni si considera che i satelliti siano dotati di un unico propulsore a bassa spinta, pertanto, vincoli specifici sono inclusi nel modello in modo da ottenere una soluzione che possa essere attuata in uno scenario di missione realistico. Inoltre, un vincolo di prevenzione di collisioni è aggiunto in caso di operazioni di prossimità, utilizzando una mappatura lineare tra elementi orbitali relativi e coordinate cartesiane espresse nel sistema di riferimento locale LVLH.

La risposta del controllore è simulata in molteplici contesti realistici con un propagatore orbitale ad alta precisione e i risultati sono validati in merito alla efficienza nel consumo di propellente e alla efficacia dell'algoritmo di prevenzione di collisioni. Per verificare che il controllo sia subottimale in termini di consumo di propellente, questo è comparato ad approcci simili reperibili in letteratura e a soluzioni ottime ottenute rispettivamente con un algoritmo direct single shooting e con una formulazione in forma chiusa per controllo impulsivo.

Parole chiave: guida e controllo autonomi; volo in formazione; bassa orbita terrestre; controllo predittivo; dinamica orbitale relativa; elementi orbitali relativi



POLITECNICO
MILANO 1863

SCUOLA DI INGEGNERIA INDUSTRIALE
E DELL'INFORMAZIONE

Executive Summary of the Thesis

Adaptive Guidance and Control for Formation Flying in High-Drag Low-Orbit Environment

Laurea Magistrale in Space Engineering - Ingegneria Spaziale

Author: Enrico Belloni

Advisor: Prof. Michèle Lavagna

Co-advisors: Stefano Silvestrini, Jacopo Prinetto

Academic year: 2021-2022

---

1. Introduction

Over the last few decades, the space sector and its applications enormously grew, as depicted by the constantly increasing number of satellites that are put into orbit every year. The traditional approach to space has always been to build a big monolithic satellite that would deal with all the tasks. However, the development of new technologies allows to split the duties of a single spacecraft between multiple smaller units flying in formation, with several advantages in terms of reliability and scientific return. In the thesis, an optimal autonomous guidance and control strategy for formation acquisition and maintenance in high-drag environment is proposed, inspired by some of the upcoming trends in space technology. In particular, a novel Model Predictive Control (MPC) acting on a single low-thrust propulsion unit is introduced. The MPC exploits a linear propagation of the dynamics in Relative Orbital Elements, allowing the inclusion of J2 effect and drag in the optimisation problem while keeping the computational effort reduced. The formation is controlled with respect to a non-decaying orbiting point in order to perform absolute and relative station keeping simultaneously. For this purpose, a dedicated plant matrix to include drag effects in the propagation is derived and validated with respect to numerical results. The problem is also convexified to allow the use of fast optimisation tools viable for autonomous control and on-board implementation. Constraints are introduced to simulate the presence of a single engine on board by providing limits of thrust module, thrusting angles and slew rates, in order to obtain feasible control profiles. In addition, a collision avoidance constraint is included in case of proximity operations.

2. Dynamics model

To describe the relative motion between the spacecrafts, the state is expressed in terms of quasi-nonsingular Relative Orbital Elements (ROE). These are nonlinear combinations of mean orbital elements (MOE), shown in Equation (1), which allow to easily introduce the effect of perturbations by linearly propagating the system with plant matrices. This set of Relative Orbital Elements is also valid in noncircular orbits and for large spacecrafts separations, showing several advantages with respect to the commonly used Hills-Clohessy-Wiltshire (HCW) equations. Moreover, their slowly varying nature is beneficial to computational efficiency. The state is augmented by adding the ballistic coefficient difference ΔB, in order to introduce the dependence of the other parameters on this term in presence of differential drag.

$$\begin{pmatrix} \delta a \\ \delta \lambda \\ \delta e_x \\ \delta e_y \\ \delta i_x \\ \delta i_y \\ \Delta B \end{pmatrix} = \begin{pmatrix} (a - a_c)/a_c \\ u - u_c + (\Omega - \Omega_c) \cdot \cos(i_c) \\ e \cdot \cos(\omega) - e_c \cdot \cos(\omega_c) \\ e \cdot \sin(\omega) - e_c \cdot \sin(\omega_c) \\ i - i_c \\ (\Omega - \Omega_c) \cdot \sin(i_c) \\ (B_d - B_c)/B_d \end{pmatrix} \tag{1}$$

The natural dynamics of the system in Relative Orbital Elements is linearly propagated by defining a plant matrix $\mathbf{A}$, obtained as the sum of the plant matrices relative to keplerian motion and all considered perturbations. Since the following study will be focused on formations flying in Low Earth Orbit (LEO), only J2 and atmospheric drag will be included:

$$\mathbf{A} = \mathbf{A_{kep}} + \mathbf{A_{J2}} + \mathbf{A_{drag}} \tag{2}$$

Adding the control term, the linearized dynamics for the j-th satellite of the formation can be expressed in the form:

$$\dot{\underline{\mathbf{x}}}_{\mathbf{j}}(\mathbf{t}) = \mathbf{A}(t)\underline{\mathbf{x}}_{\mathbf{j}}(t) + \mathbf{B}(t)\underline{\mathbf{u}}_{\mathbf{j}}(t) \tag{3}$$

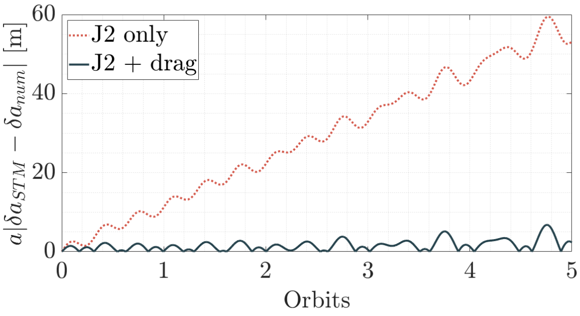
Where $\underline{\mathbf{x}} = (\delta a, \delta\lambda, \delta e_x, \delta e_y, \delta i_x, \delta i_y, \Delta B)^T$ is the augmented vector of quasi-nonsingular ROE, $\mathbf{A}$ is the plant matrix describing the natural evolution of the system including orbital perturbations, $\mathbf{B}$ is the control matrix and $\underline{\mathbf{u}}$ is the control input in RTN coordinates. The plant matrices for keplerian motion and J2 effects are taken from state-of-the art formulations available in literature, as well as the control matrix $\mathbf{B}$ [1]. On the other hand, a drag plant matrix is derived under the assumption that the control is performed with respect to a non-decaying orbiting point. In this case the relative change in semi-major axis and eccentricity vector components reduce to the absolute variations of these quantities for the deputy spacecrafts, which can be retrieved from Gauss Variational Equations:

$$\begin{cases} \dot{a} = \frac{2a^2 v}{\mu} \cdot u_{drag} \\ \dot{e}_x = \frac{2(e+\cos(\theta))}{v} \cdot \cos(\omega) \cdot u_{drag} \\ \dot{e}_y = \frac{2(e+\cos(\theta))}{v} \cdot \sin(\omega) \cdot u_{drag} \end{cases} \tag{4}$$

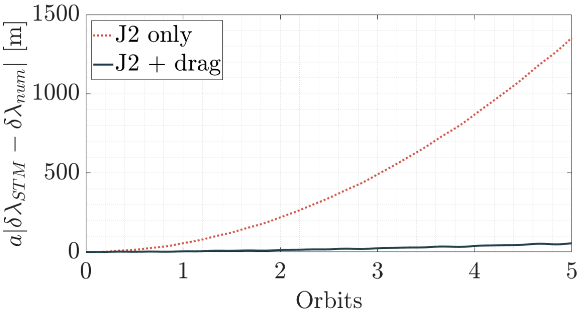
Where the drag acceleration u_{drag} is retrieved from the expression:

$$u_{drag} = \frac{1}{2} B_d \rho v^2 \tag{5}$$

The computed expression for the derivatives are placed in the last column of the plant matrix in order to multiply the differential ballistic coefficient term. In presence of a non-decaying chief it yields that $\Delta B = 1$. The retrieved plant matrix accuracy is validated with a numerical propagator by comparing its relative error with the one resulting from a plant matrix which does not include the effect of differential drag. The results for relative semi-major axis and relative mean longitude for a circular LEO at 400 km altitude are reported in Figure 1.



(a) δa error with respect to numerical propagation



(b) $\delta\lambda$ error with respect to numerical propagation

Figure 1: Drag plant matrix validation results for a circular LEO at 400 km altitude.

3. Model Predictive Control

Model Predictive Control (MPC) is a modern control technique merging the advantages of optimal and feedback control. First, the controller exploits the knowledge of the system dynamics to solve an Optimal Control Problem (OCP) over a specified period of time defined as prediction horizon, discretized according to a selected sampling step. Once the optimization is completed, the resulting control is applied for a number of time steps defined by the so-called control horizon, then the optimisation starts again taking as initial condition the new observed state of the system. MPC was chosen as the guidance and control technique for the easy implementation of constraints and easy handling of multi-input multi-output (MIMO) systems. Its basic implementation is shown in Figure 2.

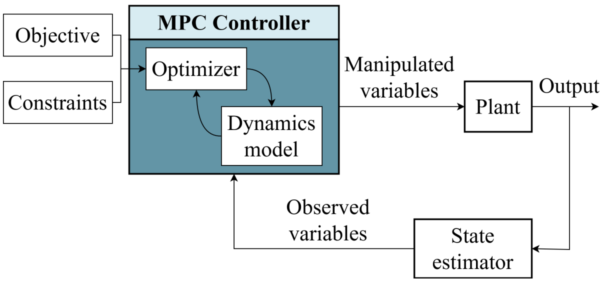


Figure 2: Model Predictive Control loop.

The dynamic nature of Model Predictive Control makes the algorithm effective in compensating any kind of disturbance, even the ones which are not modelled in the dynamics propagation. However, a good model improves both accuracy and fuel efficiency of the resulting trajectory.

4. OCP formulation

The Optimal Control Problem is convexified in order to take advantage of convex optimisation solvers, obtaining a much faster and efficient solution viable for autonomous control. To write the optimal control problem in convex form, the first step is to discretize it. Following the steps of Reference [2], time is divided into finite steps defined by the Model Predictive Controller sampling time, representing the sample interval for the state $\underline{\mathbf{x}}$ and the update interval for the control term $\underline{\mathbf{u}}$, which is considered piecewise constant for each time step.

Linear dynamics: The linear dynamics previously derived in Equation (3) is discretized according to the Laplace transformation of the state equations, leading to the finite differences expression:

$$\underline{\mathbf{x}}_{\mathbf{j}}[k+1] = (\mathbf{A}[k]\Delta t + \mathbf{I})\ \underline{\mathbf{x}}_{\mathbf{j}}[k] + \mathbf{B}[k]\Delta t\ \underline{\mathbf{u}}_{\mathbf{j}}[k] \tag{6}$$

The dynamics can now be expressed as a linear equality constraint by defining a proper optimization vector containing not only the control input, but also the state vector of relative orbital elements at each time step. In particular, for each satellite it is defined a vector $\underline{\hat{\mathbf{x}}}_{\mathbf{j}}$ which contains the ROE state and the control input at each time instant k. $\underline{\hat{\mathbf{x}}}_{\mathbf{j}}$ has size $M = 7K + 3(K-1)$, where K is the total number of time steps. Subsequently, the entire decisional vector of the optimisation problem can be defined as:

$$\underline{\hat{\mathbf{X}}} = (\underline{\hat{\mathbf{x}}}_{\mathbf{1}},\ \cdots,\underline{\hat{\mathbf{x}}}_{\mathbf{j}},\ \cdots,\underline{\hat{\mathbf{x}}}_{\mathbf{N}})^T \tag{7}$$

with size $N \cdot M$, where N is the number of chaser satellites in the formation. Following the work in Reference [3], this formulation allows to rewrite the dynamics of each spacecraft in Equation (6) in matrix form by defining a proper matrix $\mathbf{A_{sd}}$ so that for each satellite it yields:

$$\mathbf{A_{sd}} \cdot \underline{\hat{\mathbf{x}}}_{\mathbf{j}} = 0 \tag{8}$$

with $k = 1, ..., K$. Accordingly, the dynamics of the entire formation expressed in convex form as a linear equality constraint can be written as:

$$\begin{bmatrix} \cdots & \cdots & \cdots \\ \mathbf{0}_{7(K+1)\times M(j-1)} & \mathbf{A_{sd}} & \mathbf{0}_{7(K+1)\times M(N-j)} \\ \cdots & \cdots & \cdots \end{bmatrix} \cdot \underline{\hat{\mathbf{X}}} = 0 \tag{9}$$

for $j = 1, ..., N$.

Initial and final conditions: Similarly, the initial and final state can be extracted from the decisional vector by defining proper matrices, which can be arranged to isolate initial and final conditions from the optimisation vector $\underline{\hat{\mathbf{X}}}$.

Cost function: The cost functional to be minimized in the OCP must be expressed in terms of the discretized decisional vector $\underline{\hat{\mathbf{X}}}$ as well. Hence, also the control term must be isolated. In particular, a matrix $\hat{\mathbf{H}}$ can be defined so that:

$$\underline{\hat{\mathbf{U}}} = \hat{\mathbf{H}} \cdot \underline{\hat{\mathbf{X}}} \tag{10}$$

Where the $\underline{\hat{\mathbf{U}}}$ vector contains the control accelerations in their respective position and is equal to zero in its other components. By defining the weighting matrices $\mathbf{P}$ and $\mathbf{Q}$, the cost function is then expressed as:

$$\begin{aligned} J =&||\hat{\mathbf{H}} \cdot \underline{\hat{\mathbf{X}}}||_1 + ||\mathbf{P}(\mathbf{A_{FC}} \cdot \underline{\hat{\mathbf{X}}} - \underline{\hat{\mathbf{X}}}_{\mathbf{T}})||_1 + ... \\ &||\mathbf{Q}(\underline{\hat{\mathbf{X}}}_{\mathbf{ROE}} - \underline{\hat{\mathbf{X}}}_{\mathbf{ROE_T}})||_1 \end{aligned} \tag{11}$$

Namely:

- $||\hat{\mathbf{H}} \cdot \underline{\hat{\mathbf{X}}}||_1$ is the control effort term, added to minimize the propellant consumption finding a sub-optimal fuel efficient solution.
- $||\mathbf{P}(\mathbf{A_{FC}} \cdot \underline{\hat{\mathbf{X}}} - \underline{\hat{\mathbf{X}}}_{\mathbf{T}})||_1$ is the tracking error of the final state with respect to the target ROE vector. Its minimization leads to the convergence of the state to the desired one.
- $||\mathbf{Q}(\underline{\hat{\mathbf{X}}}_{\mathbf{ROE}} - \underline{\hat{\mathbf{X}}}_{\mathbf{ROE_T}})||_1$ identifies the difference between the target ROE vector and the ROE state of the satellites at each time step. The introduction of this expression avoids a stall of the MPC which may happen if control actions would be prescribed only after the control horizon.

Since the cost function is a sum of 1-norms, it is convex and also linear. Moreover, the introduction of tracking terms in the objective, rather than as a hard constraint on the final target, ensures the feasibility and also the convergence of the problem, if these terms are weighted enough. The choice of proper weighting matrices allows to prioritize or even isolate the tracking error relative to some Relative Orbital Elements with respect to others, depending on the scenario. The chosen values strongly depend on mission requirements, thus, their selection differs from case to case. However, two general considerations that must be taken into account are that the tracking errors should be weighted enough to make the spacecrafts converge to the desired state, as anticipated previously, and that the higher the weights on tracking error terms, the quicker the transfer will be. This last characteristic can be exploited in particular operational situations where timeliness is crucial.

Thrust module constraint: The constraints added in the control problem to obtain a feasible control profile are strongly linked to the available propulsion solution, a single low-thrust engine in this case. In current research on relative dynamics, guidance and control, it is often assumed that the spacecraft is capable of thrusting in any direction at any time. This is however rarely true, in particular when employing electric engines, unless the satellite is equipped with attitude thrusters that can be also used for orbit control. Dealing with a single engine, the constraint on the maximum acceleration is given on the module rather than for each component. This is achieved isolating and reshaping the control at each time step k for each satellite j from the decisional vector $\underline{\hat{\mathbf{X}}}$. Then, it is enough to impose that the norm of each control accelerations vector is lower than the maximum acceleration that the engine is able to provide to the spacecraft:

$$\begin{gathered} ||\underline{\mathbf{u}}_j^k||_2 \leq \frac{T_{max}}{m_{s/c}} \\ k = 1, ..., K, \quad j = 1, ..., N \end{gathered} \tag{12}$$

Most of low-thrust propulsion units also have a limit in the lowest thrust they are able to provide. This constraint is impossible to express in convex form, as it would represent a "hole" in the control three-dimensional space. Therefore this constraint is imposed a posteriori on the minimization results, just by ignoring control actions which have a norm lower than the minimum acceleration of the engine. The algorithm will adapt just by providing a higher control action at a later time, converging anyway to the target.

Constraints on thrust angles and slew rates: During nominal operations of a spacecraft, additional constraints may be imposed on the thrusting angles and on the maximum allowed slew rate to point the engine in the desired direction. The inclusion of a constraint on the maximum slewing rate is not natural in convex form, however, a procedure is found to at least deny the presence of immediate 180° slew around each axis. Adopting a similar strategy, additional constraints can be added to cover slews between different axes.

Collision avoidance: Collision avoidance must be imposed as a constraint on the relative position of the satellites in the RTN frame. Since the state vector is expressed in ROE form, a linear transformation is needed to retrieve the RTN coordinates before imposing collision avoidance. The selected linear mapping, represented by the matrix **T**, is derived following the procedure reported by Silvestrini et al. in [4], The transformation matrix is derived by using an intermediate change of coordinates exploiting the classical orbital elements difference $\Delta OE = [\Delta a, \Delta M, \Delta\omega, \Delta e, \Delta i, \Delta\Omega]$ as follows:

$$\mathbf{T} = \frac{\partial \underline{\mathbf{x}}_{\mathbf{RTN}}}{\partial \Delta OE} \cdot \frac{\partial \Delta OE}{\partial \boldsymbol{\delta}\underline{\boldsymbol{\alpha}}} \tag{13}$$

Once the x, y and z components of the decisional vector of each satellite are isolated, it is necessary to impose the minimum distance constraint in a convex formulation. The selected methodology is the one formulated by Morgan et al. in Reference [2]. Such strategy consists in generating separating planes among the satellites, transforming the circular prohibited zone into a suitable convex formulation.

5. Validation

Fuel efficiency: The reliability and optimality of the designed control solution are verified by comparing it with other options available in literature. A close match to the proposed strategy is found in Reference [5], in which Catanoso et al. develop a Lyapunov-based Model Predictive Controller acting on Relative Orbital Elements. With respect to this reference, fuel efficiency is assessed performing a comparison between the two MPC solutions of the same out-of-plane transfer and with the closed-form fuel-optimal impulsive solution developed by Chernick et al. in Reference [1]. Moreover, a solution obtained with a direct single shooting (DSS) method is added to the comparison to have an optimal low-thrust benchmark.
The obtained trajectory in the RTN frame is shown in Figure 3. The resulting ΔV needed to complete the transfer with the desired accuracy is compared to the other results in Table 1. The proposed solution demonstrates to be close to optimal, performing better than the nonconvex counterpart and coming close to the single shooting and impulsive optimal solutions, while being viable for on-board autonomous low-thrust control.

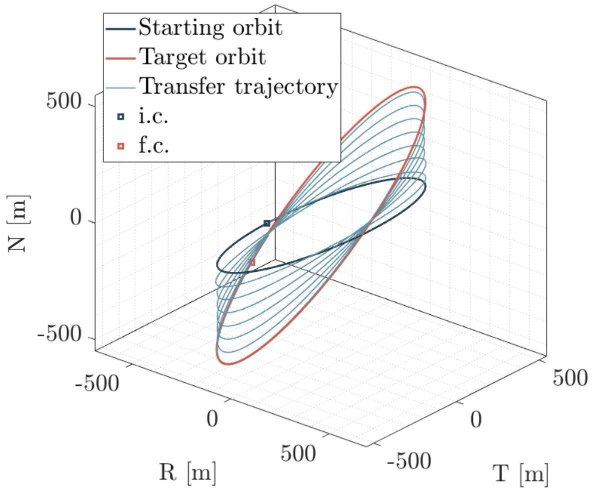


Figure 3: Transfer trajectory in the RTN reference frame for the out-of-plane transfer used for fuel efficiency validation.

Table 1: ΔV comparison between the proposed MPC, the two considered literature solutions, and the DSS algorithm result.

Method	ΔV [m/s]
Reference MPC [5]	0.5554
Proposed MPC	0.4915
Direct single shooting	0.4680
Closed-form impulsive [1]	0.4373

Drag compensation: Due to the feedback nature of Model Predictive Control, it can be argued that the inclusion of a drag plant matrix is not mandatory, as the relative drift would be corrected anyway by the MPC prescribing a tangential manoeuvre when an undesired along-track separation is created. To prove the effectiveness of the proposed drag model, the tracking of an holding point placed in the chief spacecraft's position is simulated. This equals to controlling the deputy spacecraft in its absolute dynamics to compensate the effects of drag and to maintain its along-track position. By simulating with and without including the drag plant matrix in the dynamics propagation, the results can be compared and the effectiveness of the drag model can be assessed. Variations in time of mean semi-major axis and $\delta\lambda$ are represented in Figures 4 and 5. The semi-major axis is averaged over one orbital period to remove the periodic oscillations due to J2 effect.

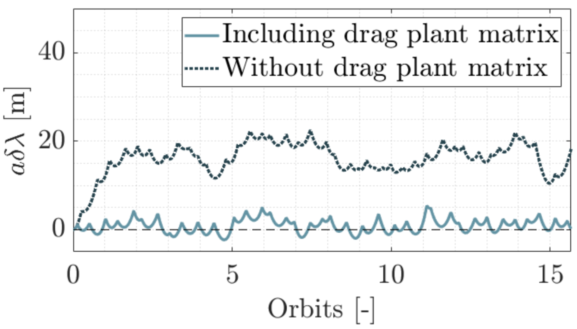


Figure 4: $\delta\lambda$ evolution in time for one day of station keeping.

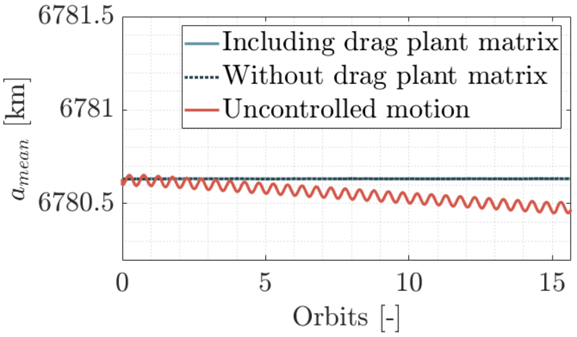


Figure 5: Mean semi-major axis evolution in time for one day of station keeping.

From the figures it is evident that the inclusion of the plant matrix improves the accuracy of the solution in terms of tracking of the desired along-track separation. Indeed in both cases a constant mean altitude is maintained, but, when drag is not considered in the propagation, a $\delta\lambda$ error of about 20 m is accumulated throughout the day. This error is never corrected even when the simulation is left running for several more orbit revolutions.

6. Conclusions

In the thesis, a novel on-board Model Predictive Controller for optimal formation acquisition and maintenance in high-drag environments is proposed. The main contribution to the state of the art is the use of a novel convexified linear dynamics expressed in Relative Orbital Elements in a Model Predictive Controller, instead of the classical cartesian representation in the Local-Vertical-Local-Horizontal reference frame. The formulation is augmented to control the spacecrafts both relatively and absolutely in a high-drag environment with respect to a non-decaying orbiting point. For this scope, the derivation of a dedicated plant matrix is proposed and validated. Furthermore, constraints are introduced in the convexified formulation to resemble the limitations of mounting a single low-thrust engine on board, in order to obtain a feasible control profile for a micro-satellite.
The designed controller demonstrated to be able to perform formation acquisition, reconfiguration and maintenance in a timely and efficient manner in a wide variety of realistic operational scenarios at very low altitude, while satisfying constraints on allowed thrusting cones and slewing limitations. Fuel efficiency was assessed by a comparison with a nonconvex nonlinear MPC, a closed-form fuel-optimal impulsive solution and an optimal trajectory obtained with a direct single shooting algorithm, showing that the designed guidance and control strategy is able to provide a close to optimal result even in presence of multiple constraints.

References

[1] M. Chernick and S. D'Amico. New Closed-Form Solutions for Optimal Impulsive Control of Spacecraft Relative Motion. *Journal of Guidance, Control, and Dynamics*, November 2017.

[2] D. Morgan, S. Chung, and F. Y. Hadaegh. Model Predictive Control of Swarms of Spacecraft Using Sequential Convex Programming. *Journal of Guidance, Control, and Dynamics*, November 2014.

[3] S. Sarno, J. Guo, M. D'Errico, and E. Gill. A guidance approach to satellite formation reconfiguration based on convex optimization and genetic algorithms. *Advances in Space Research*, January 2020.

[4] S. Silvestrini and M. Lavagna. Neural-aided GNC reconfiguration algorithm for distributed space system: development and PIL test. *Advances in Space Research*, March 2021.

[5] D. Catanoso, F. Kempf, K. Schilling, and S. D'Amico. Networked Model Predictive Control for Satellite Formation Flying. *10th International Workshop of Satellites Constellations and Formation Flying*, July 2019.

Acknowledgements

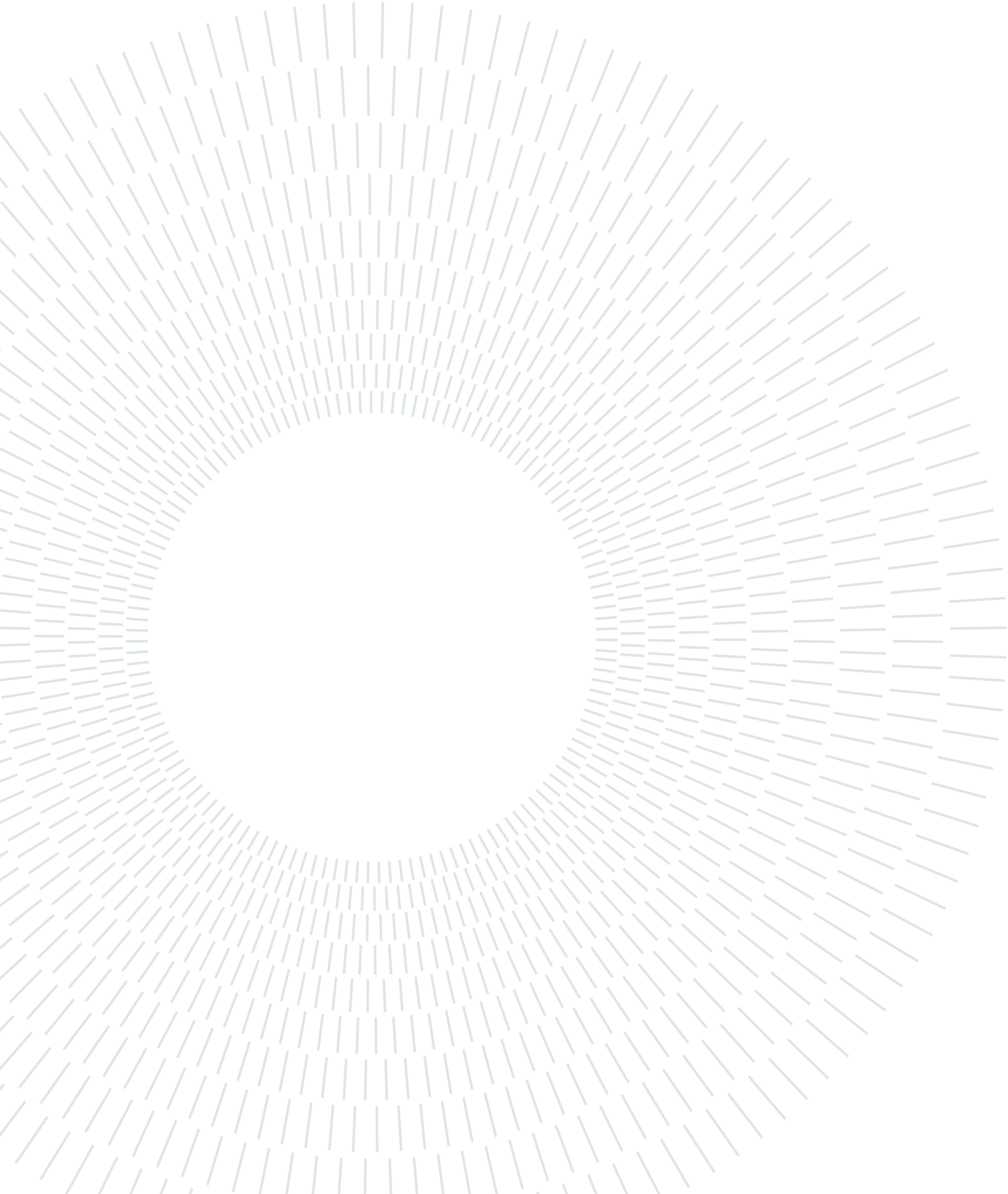
I would like to thank my supervisor Prof. Lavagna for giving me the opportunity to work on such an interesting topic, allowing me to discover and learn new aspects of a field that I would like to study more into deep in the future. Special thanks to my co-supervisors Stefano and Jacopo for the precious help and support that was given to me throughout my research, always with a positive attitude.

Thanks to all the friends that I met during these years in Pisa and Milan for the many moments of fun during the long days at university and not only. I am sure we are going to stay in touch even at the end of our academic journey.

Thanks to my friends from Livorno, my city, you are a constant source of joy and inspiration.

Thanks to my girlfriend Sara for the never ending help and understanding on which I can always rely on, even from a distance. Thanks for always supporting me in my choices and for cheering me up and bearing me in times of pressure.

Thanks to my family. To Mom and Dad in particular for always encouraging me and allowing me to pursue my dreams.



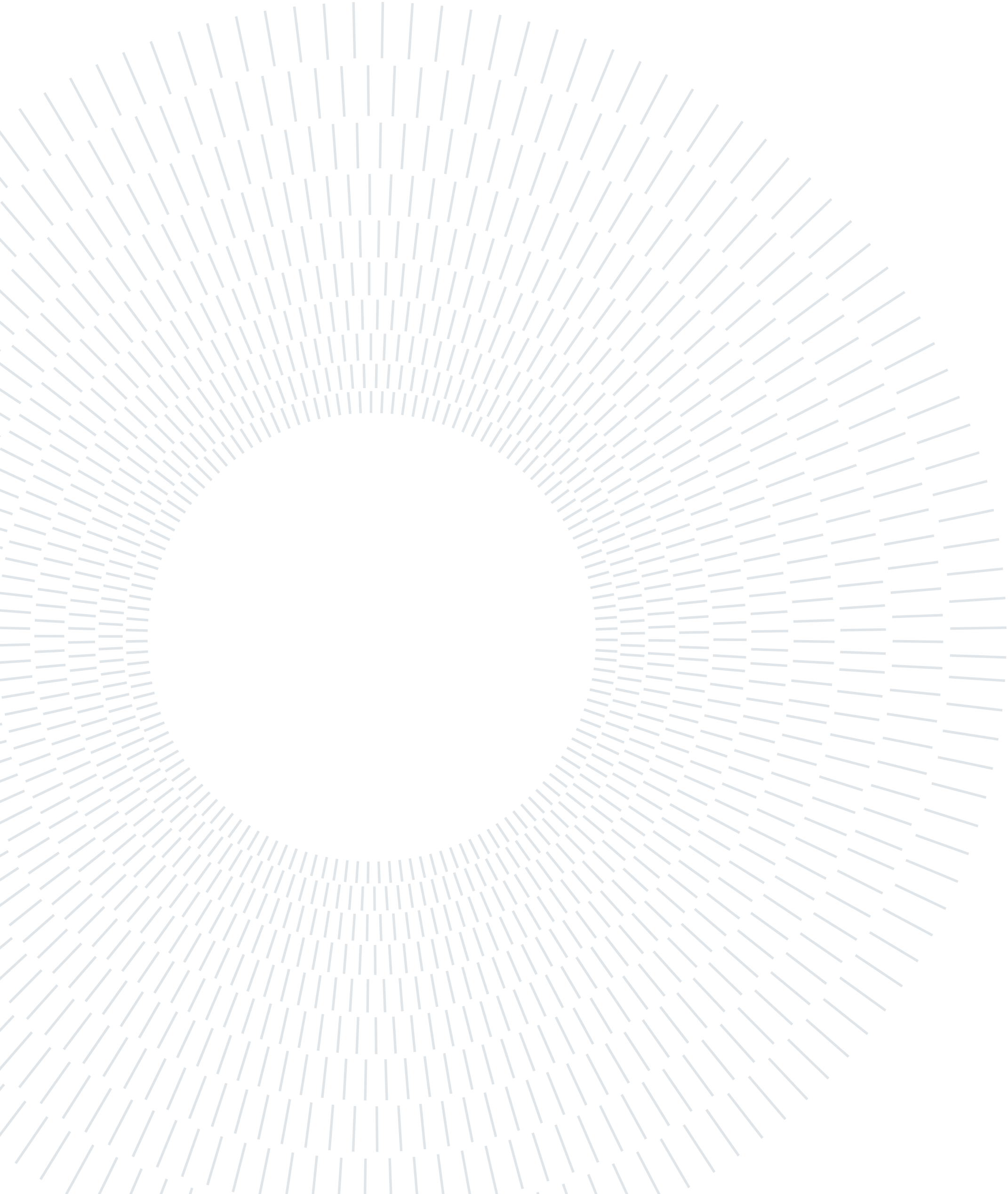
Contents

List of Figures

List of Tables

Physical constants

Symbol	**Description**	**Value in SI units**
μ	Earth gravitational parameter	398600 $[\mathrm{km}^3/\mathrm{s}^2]$
$J2$	Second zonal harmonic coefficient	1.08 $\cdot 10^{-3}$ [-]
$J3$	Third zonal harmonic coefficient	-2.51 $\cdot 10^{-6}$ [-]
R_E	Earth radius at equator	6378 [km]



Nomenclature

Variable	Description	SI Unit
$\underline{\boldsymbol{\alpha}}$	Relative Orbital Elements state	-
$\Delta(\cdot)$	$(\cdot)$ difference	
$\delta(\cdot)$	relative $(\cdot)$	
θ	true anomaly	° or rad
θ	off-plane angle	° or rad
λ	mean longitude	
ρ	atmospheric density	$\mathrm{kg/m^3}$
ϕ	in-plane angle	° or rad
Ω	right-ascension of the ascending node	° or rad
ω	argument of perigee	° or rad
$\mathbf{A}$	plant matrix	1/s
A	area	$\mathrm{m^2}$
a	orbit semi-major axis	km
a	acceleration	$\mathrm{m/s^2}$
$\mathbf{B}$	control matrix	s/m
B	ballistic coefficient	$\mathrm{m^2/kg}$
C_D	drag coefficient	-
C_R	reflectance coefficient	-
e	orbit eccentricity	-
i	orbit inclination	° or rad
M	mean anomaly	° or rad
n	mean motion	1/s
p	semilatus rectum	km

Variable	**Description**	**SI Unit**
T	orbit period	s
T	thrust	N
T, t	time	s
$\underline{\mathbf{u}}$	control input	m/s^2
u	argument of latitude	° or rad
V, v	velocity	m/s
$\underline{\mathbf{X}}, \underline{\mathbf{x}}$	state	

Acronyms

Acronym	Description
APF	Artificial Potential Fields
CA	Collision Avoidance
CH	Control Horizon
DCP	Disciplined Convex Programming
DEM	Digital Elevation Models
DLR	Deutsches Zentrum für Luft- und Raumfahrt
DoF	Degrees of Freedom
DSS	Direct Single Shooting
ECI	Earth-Centered Inertial
ESA	European Space Agency
FC	Final Conditions
FF	Formation Flying
GNC	Guidance Navigation and Control
GVE	Gauss Variational Equations
HCW	Hills-Clohessy-Wiltshire
HEO	Highly Elliptical Orbit
IC	Initial Conditions
LEO	Low Earth Orbit
LQR	Linear Quadratic Regulator
LVLH	Local-Vertical-Local-Horizontal
MIMO	Multi-Input Multi-Output
MPC	Model Predictive Control
NASA	National Aeronautics and Space Administration
OCP	Optimal Control Problem

Acronym	Description
OE	Orbital Elements
ODE	Ordinary Differential Equation
PH	Prediction Horizon
QV	Quadratic Volterra
ROE	Relative Orbital Elements
RTN	Radial-Tangential-Normal
SAR	Sinthetic Aperture Radar
SK	Station Keeping
SRP	Solar Radiation Pressure

1 | Introduction

1.1. Introduction to multi-satellite missions

Over the last few decades, the space sector and its applications enormously grew, as depicted by the constantly increasing number of satellites that are put into orbit every year. The traditional approach to space has always been to build a big monolithic satellite that would deal with all the tasks. However, the development of new technologies allows to split the duties of a single spacecraft between multiple smaller units, with several advantages:

1. The intrinsic redundancy of a multi-satellite formation increases robustness, whereas for a single satellite, a component failure may jeopardize the entire mission.
2. Spacecrafts forming a formation are usually less complex than large monolithic units, as tasks and components can be distributed, simplifying and speeding up the production phase.
3. New technology is enabled by flying in proximity. One of the most relevant examples at the moment is Synthetic Aperture Radar (SAR) interferometry, achieving resolutions that could never be achieved otherwise.

Multi-satellite missions are traditionally categorized into two types:

- **Constellations**: formed by spacecrafts independently controlled with respect to a reference orbiting slot.
- **Formation Flying (FF)**: when the satellites are controlled relatively to one another, flying usually in close proximity.

The following work will be focused on the latter, proposing an autonomous control strategy for formation acquisition and maintenance in high-drag environment using a low-thrust propulsion unit.

1.2. Formation flying state of the art

To date, several are the FF missions that have successfully flown, in a wide range of applications, sizes, and orbit shapes. In 2002, GRACE mission by DLR and NASA flew two spacecrafts with a 220 km along-track separation on a circular polar orbit to derive global high-resolution models of the mean and the time-variable components of earth's gravity field. Both GRACE spacecrafts were ground controlled to keep the separation in the desired range, while the formation was left decaying under the effect of drag [1]. In 2010, the DLR mission TanDEM-X joined TerraSAR-X on its Sun-synchronous frozen orbit. The objective of the mission is to form a close formation in order to build Digital Elevation Models (DEM) of the earth's surface using SAR interferometry. Both spacecrafts must perform the same orbital maneuvers to counteract lunisolar perturbations and to compensate atmospheric drag, but formation acquisition and maintenance maneuvers are exclusively performed by TanDEM-X [2]. Also the Swedish Prototype Research Instruments and Space Mission technology Advancement (PRISMA) mission was launched in the same year as a formation flying technology demonstrator. The formation consists of a main satellite (Mango, 150 kg) and a target satellite (Tango, 50 kg) released on a dawn-to-dusk Sun-synchronous orbit to test and validate guidance, navigation, and control (GNC) hardware, software, and algorithms for autonomous formation flying, homing and rendezvous, proximity operations, and final approach and recede operations [3]. Another technology demonstrator, ESA's PROBA-3, is going to be launched in 2023 on a High Elliptical Orbit (HEO), to demonstrate technologies and techniques for highly precise satellite formation flying. Moreover, the mission will host an external solar coronograph instrument to discover and analyze terrestrial extrasolar planets, providing also a valuable scientific return [4].

The first tool which is needed to design a formation flying mission is a mathematical formulation for the relative dynamics between the spacecrafts. A literature survey highlights two main categories:

1. **Solutions based on the translational state**:
    - *Hill-Clohessy-Wiltshire (HCW) equations*: HCW equations provide a linearized model for the evolution of the relative dynamics in the Local-Vertical-Local-Horizontal (LVLH) reference frame, obtained with the simplifying assumptions of circular reference orbit of the target spacecraft and small separations [5].
    - *Quadratic-Volterra models*: Second-order solutions to the circular orbit problem were independently derived shortly after CW by London and Sasaki [6, 7]

    and later by Stringer and Newman [8]. These models are frequently referred to as the Quadratic-Volterra (QV) solutions, due to the polynomial approximation technique they are based upon.

    - *Yamanaka-Ankersen model*: in 2002 Yamanaka and Ankersen derived the state transition matrix for the evolution of relative dynamics in arbitrarily eccentric orbits, but keeping the hypothesis of small separations [9].

    - *Second-order analytical solution for arbitrarily eccentric orbits*: in 2019 Willis et al. derived a new second-order solution for the relative position and velocity of two spacecrafts on keplerian orbits of arbitrary eccentricity [10].

   The main advantage of using the relative translational state is that it is closely tied to the state of the satellites as observed by navigation instruments.

2. **Solutions based on Relative Orbital Elements (ROE)**:

    - *Linear propagation with state transition matrices*: state transition matrices for linear propagation of Relative Orbital Elements have been extensively studied and used in real applications. Matrices for almost all perturbations are provided in literature [11–14], and can be used to obtain a quick and precise propagation.

    - *Nonlinear propagation based on Gauss Variational Equations (GVE)*: nonlinear models are more precise with respect to linear ones as they include coupling effects which become evident after several orbits or in particularly challenging environments. However, these models are inherently more computationally intensive. Moreover, if the model is not used for long term propagation but for times similar to one orbital period (as in the application proposed in this thesis), nonlinear coupling effects can be considered negligible [11].

   Relative Orbital Elements representations are better connected to the physics and relative motion geometry, providing a better platform to introduce the effect of perturbations. Moreover, they are slowly varying in time, offering a higher computational efficiency.

Once the description of the dynamics is selected, a proper control strategy can be derived, in order to acquire or track the desired states. Also in this case several options can be found in literature, each proving its advantages and disadvantages. A first solution is impulsive control, for which even closed form solutions exist [15], but these cannot be applied for low-thrust applications. A suitable option for continuous low-thrust control is to use a Linear Quadratic Regulator (LQR) built on the linearized dynamics. However, even if this is an optimal technique, it is hard to include constraints on the solution, which therefore may result to be unfeasible when real operational constraints are present. Similar limitations exist for nonlinear control with Lyapunov functions. A novel technique is the use of Artificial Potential Fields (APF) as a guidance layer to generate the trajectory to follow by chaser spacecrafts [16, 17]. This method allows an easy implementation of collision avoidance constraints, but may easily lead to instabilities and lacks a measure of optimality. A technique that ensures the sub-optimality of the solution while allowing easy implementation of constraints is Model Predictive Control (MPC). The main drawback of MPC is the computational burden imposed on the hardware, which may be excessive for on-board implementation. However, a convexification of the problem allows for a quick and efficient solution using convex optimization solvers, which may unlock on-board implementation possibilities.

The formulation of a convexified guidance and control problem for formation reconfiguration has been studied in literature in several instances [18–21]. In 2006, Acikmese et al. proposed a fuel-optimal convex guidance algorithm for formation reconfiguration including collision avoidance suitable for on-board implementation [18]. In 2014, Morgan et al. proposed both an optimal guidance solution and a Model Predictive Control implementation for reconfiguration of swarms of spacecrafts between J2 invariant orbits, again including collision avoidance [19]. In 2020, Sarno et al. proposed a similar method for autonomous formation reconfiguration of distributed systems, adding a task-assignment strategy optimising via a genetic algorithm [20]. Finally, in 2021 Scala et al. proposed a design strategy of optimal low-thrust manoeuvres for remote sensing multi-satellite formation flying in LEO. In the discussed solution an interface between ROE and cartesian LVLH state is provided [21]. In all the listed references the states of the satellites are expressed in cartesian coordinates in the local LVLH reference frame.

Starting from previous work, this thesis proposes a guidance and control strategy for formation flying missions which is inspired by some of the upcoming trends in space technology:

- The proposed control strategy is **autonomous**. Autonomy is almost mandatory for close proximity operations, to be able to react immediately in a quickly evolving environment. Moreover, it is an increasing trend to try to lower the number of operations to be carried on ground, in order to reduce time consumption and costs needed to monitor a constantly increasing population of satellites.

- The control is actuated by a **low-thrust** engine. Low-thrust propulsion units, in particular electric and electromagnetic ones, are becoming the upcoming standard for space propulsion due to their considerably lower propellant consumption with respect to traditional thermochemical thrusters. Only a single propulsion unit is assumed to be available on board. This is true in the vast majority of micro-satellite missions, in which usually only an electric engine for orbit control is mounted due to issues related to size, weight, and power generation.

- The formation is assumed to perform scientific activities in **low-orbit** high-drag environment, particularly suited for earth observation missions when mounting instruments whose resolution benefits from flying at low altitudes.
- The approach is **scalable** and **flexible**, being able to deal with reconfiguration and maintenance of large formations of satellites.

In particular, a novel Model Predictive Control (MPC) acting on a single low-thrust propulsion unit for formation acquisition and maintenance in high-drag environment is presented. The MPC exploits the linear propagation of the dynamics expressed in Relative Orbital Elements state, allowing the inclusion of J2 effect and drag in the optimization problem while keeping the computational effort reduced. The problem is also convexified to allow the use of fast optimization tools to further reduce the computing time. Some constraints are introduced to simulate the presence of a single engine on board, providing limits of thrust module, thrusting angles, and maximum slew rates according to the operational scenario. In addition, a collision avoidance constraint is included for close proximity operations. Formation control is performed with respect to a non-decaying orbiting point, in order to perform relative and absolute orbit control simultaneously. This point may represent a chief spacecraft that is controlled only absolutely to counteract drag-induced orbit decay, or a virtually propagated orbiting point placed in the centroid of the formation. With respect to open-loop optimal guidance algorithms, the feedback nature of Model Predictive Control is robust to changes in external conditions, like a sudden increase in atmospheric density due to a rise in solar activity.

1.3. Thesis outline

This document is organized into seven chapters, including the introduction:

- In Chapter 2, a description of the linear dynamics formulation in quasi-nonsingular Relative Orbital Elements state is provided, to introduce the assumptions and considerations which drove the derivation of the model. Particular attention is focused on the derivation of the drag plant matrix, for which validation of the results with respect to a numerical propagator is also provided.
- In Chapter 3, Model Predictive Control technique is introduced and its tuning pa-

rameters are discussed. In addition, a scheme of the implemented simulation loop is provided together with its algorithmic implementation.

- In Chapter 4, the mathematical formulation of the Optimal Control Problem, the core of the MPC, is thoroughly examined. In this section of the document particular efforts are devoted to finding a convex formulation to allow the use of convex optimization solvers, following the common approach present in literature. Moreover, the introduction of constraints specific to the selected actuation technology is discussed. Finally, the implementation of collision avoidance via a linear mapping between ROE and the local cartesian state is described.
- In Chapter 5, a validation of the fuel efficiency of the proposed solution is carried out, comparing its formation reconfiguration response to a similar option available in literature [22], to a closed-form impulsive solution [15], and to an offline optimizer acting on the full nonlinear dynamics adopting a single shooting algorithm. Moreover, the effectiveness of the inclusion of the derived plant matrix in the dynamics propagation and its impact on the control are analyzed.
- In Chapter 6, the results of all performed simulations for different realistic operational scenarios in high-drag environment are reported, together with a discussion on the performance provided by the MPC. Different aspects of the algorithm are tested, including the collision avoidance constraint.
- Finally, in Chapter 7 the outcomes of the work are summarized with some thoughts on the obtained results. Moreover, some ideas for possible future developments are proposed.

2 | Dynamics model

2.1. Relative Orbital Elements formulation

To describe the relative motion between the spacecrafts, the state is expressed in terms of quasi-nonsingular Relative Orbital Elements (ROE). These are nonlinear combinations of mean orbital elements (MOE), shown in Equation (2.1) [23], which allow to easily introduce the effect of perturbations by linearly propagating the system with proper plant matrices. Moreover, their slowly varying nature is beneficial to computational efficiency.

$$\boldsymbol{\delta}\underline{\boldsymbol{\alpha}} = \begin{pmatrix} \delta a \\ \delta\lambda \\ \delta e_x \\ \delta e_y \\ \delta i_x \\ \delta i_y \end{pmatrix} = \begin{pmatrix} (a - a_c)/a_c \\ u - u_c + (\Omega - \Omega_c) \cdot \cos(i_c) \\ e \cdot \cos(\omega) - e_c \cdot \cos(\omega_c) \\ e \cdot \sin(\omega) - e_c \cdot \sin(\omega_c) \\ i - i_c \\ (\Omega - \Omega_c) \cdot \sin(i_c) \end{pmatrix} \tag{2.1}$$

In the previous set of equalities, $u = \omega + M$ is the mean argument of latitude and the subscript c refers to the chief satellite with respect to which the relative dynamics of each deputy spacecraft is defined. The semi-major axis difference δa is normalized to be a dimensionless quantity, $\delta\lambda$ is defined as relative mean longitude and defines the along-track separation of the spacecrafts, and the remaining four terms form what are called relative eccentricity and relative inclination vectors, which can also be expressed in polar coordinates as:

$$\boldsymbol{\delta}\underline{\mathbf{e}} = \begin{pmatrix} \delta e_x \\ \delta e_y \end{pmatrix} = \delta e \cdot \begin{pmatrix} \cos(\phi) \\ \sin(\phi) \end{pmatrix} \qquad \boldsymbol{\delta}\underline{\mathbf{i}} = \begin{pmatrix} \delta i_x \\ \delta i_y \end{pmatrix} = \delta i \cdot \begin{pmatrix} \cos(\theta) \\ \sin(\theta) \end{pmatrix} \tag{2.2}$$

This set of Relative Orbital Elements is also valid in noncircular orbits and for large spacecrafts separations, showing further advantages with respect to HCW equations. Only a singularity is present in case of orbits at zero inclination.

2.1.1. State augmentation

In general, perturbations causing a variation of the orbital elements are of two types, conservative and non-conservative. In the second instance, as in presence of differential drag, the perturbing effect depends not only on the differences in the orbit geometry, but also on the difference between satellites features, represented by a ballistic coefficient difference ΔB in this case. This parameter is therefore included in an augmented state representation as the last term.

$$\boldsymbol{\delta}\underline{\boldsymbol{\alpha}}^{aug} = \begin{pmatrix} \delta a \\ \delta\lambda \\ \delta e_x \\ \delta e_y \\ \delta i_x \\ \delta i_y \\ \Delta B \end{pmatrix} = \begin{pmatrix} (a - a_c)/a_c \\ u - u_c + (\Omega - \Omega_c) \cdot \cos(i_c) \\ e \cdot \cos(\omega) - e_c \cdot \cos(\omega_c) \\ e \cdot \sin(\omega) - e_c \cdot \sin(\omega_c) \\ i - i_c \\ (\Omega - \Omega_c) \cdot \sin(i_c) \\ (B_d - B_c)/B_d \end{pmatrix} \tag{2.3}$$

From now on the augmented state $\boldsymbol{\delta}\underline{\boldsymbol{\alpha}}^{aug}$ will be referred to as $\boldsymbol{\delta}\underline{\boldsymbol{\alpha}}$.

2.2. Linear dynamics propagation

The natural dynamics of the system in Relative Orbital Elements is propagated by defining a plant matrix, $\mathbf{A}$, which includes the effects of keplerian motion and all the considered perturbations. This is obtained as the sum of the plant matrices relative to each of these contributions. Since the following study will be focused on formations flying in Low Earth Orbit (LEO), only J2 and atmospheric drag disturbances will be included, thus, the natural dynamics plant matrix is defined as:

$$\mathbf{A} = \mathbf{A_{kep}} + \mathbf{A_{J2}} + \mathbf{A_{drag}} \tag{2.4}$$

Adding the control term, the linearized dynamics for the j-th satellite of the formation can be expressed in the form:

$$\dot{\underline{\mathbf{x}}}_{\mathbf{j}}(t) = \mathbf{A}(t)\underline{\mathbf{x}}_{\mathbf{j}}(t) + \mathbf{B}(t)\underline{\mathbf{u}}_{\mathbf{j}}(t) \tag{2.5}$$

Where $\underline{\mathbf{x}} = \boldsymbol{\delta}\underline{\boldsymbol{\alpha}} = (\delta a, \delta\lambda, \delta e_x, \delta e_y, \delta i_x, \delta i_y, \Delta B)^T$ is the augmented vector of quasi-nonsingular Relative Orbital Elements, $\mathbf{A}$ is the plant matrix describing the natural evolution of the system including orbital perturbations, **B** is the control matrix, and $\underline{\mathbf{u}}$ is the control input in RTN coordinates.

2.2.1. Keplerian motion

The general linearized relative motion of a deputy satellite relative to the chief for arbitrary eccentricities in a keplerian two-body problem is provided in terms of ROE as [12]:

$$\delta\lambda(t) = -\frac{3}{2}n_c(t - t_0)\delta a_0 + \delta\lambda_0 \tag{2.6}$$

Where n_c is the chief mean angular motion, and the "0" subscript indicates the quantities at initial time. From the previous expression, a simple plant matrix $\mathbf{A_{kep}}$ can be retrieved as:

$$\mathbf{A_{kep}} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ -\frac{3}{2}n_c & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \tag{2.7}$$

It is evident that keplerian relative motion depends only on the relative semi-major axis difference. Accordingly, the only nonzero higher-order terms will be proportional to powers of δa. Therefore, this plant matrix is valid for unperturbed orbits with small δa and arbitrary separation in all other state components [11].

2.2.2. J2 effect

The J2 plant matrix can be retrieved from the differential effect of the earth oblateness acting on the chief and on the deputy. In particular, first-order secular effects of the second-order zonal geopotential harmonic J2 are included in the propagation. The formulation that will be used is found in Reference [12] and reported in the following:

$$\mathbf{A_{J2}} = \kappa_{J2} \cdot \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ -\frac{7}{2}EP & 0 & e_x GFP & e_y GFP & -FS & 0 & 0 \\ \frac{7}{2}e_y Q & 0 & -4e_x e_y GQ & -(1+4Ge_y^2)Q & 5e_y S & 0 & 0 \\ -\frac{7}{2}e_y Q & 0 & (1+4Ge_x^2)Q & 4e_x e_y GQ & -5e_y S & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \frac{7}{2}S & 0 & -4e_x GS & -4e_y GS & 2T & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \tag{2.8}$$

Where the terms in the matrix are defined as:

$$\eta = \sqrt{1-e_c^2}, \qquad \kappa_{J2} = \frac{3}{4}\frac{J_2 R_E^2 \sqrt{\mu}}{a^{3.5}\eta^4},$$

$$e_x = e_c \cdot \cos(\omega_c), \qquad e_y = e_c \cdot \sin(\omega_c),$$

$$E = 1+\eta, \qquad F = 4+3\eta, \qquad G = \frac{1}{\eta^2}, \qquad P = 3\,\cos^2(i_c) - 1,$$

$$Q = 5\,\cos^2(i_c) - 1, \qquad S = \sin(2i_c), \qquad T = \sin^2(i_c)$$

The combined effects of keplerian motion and J2 perturbation on Relative Orbital Elements can be visualized as drifts or rotations of relative mean longitude, and relative eccentricity and inclination vectors [11]. These effects are represented in Figure 2.1 and can be summarized as:

- a constant drift of $\delta\lambda$ due to both keplerian motion and J2 effect,
- a rotation of the relative eccentricity vector $\boldsymbol{\delta}\underline{\mathbf{e}}$ due to J2,
- a secular drift of the relative eccentricity vector proportional to the chief eccentricity and orthogonal to the phase angle of the chief argument of perigee due to J2,
- a constant drift of δi_y due to J2.

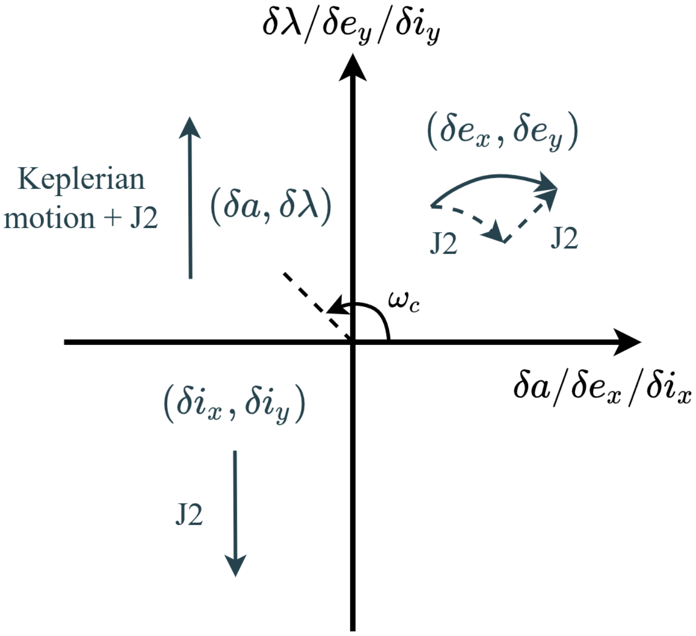


Figure 2.1: Combined effects of keplerian motion and J2 on ROE [11].

Analyzing the two defined plant matrices it can be derived that, for $\delta a = 0$, the energy of the two orbits is matching and no along-track drift is present. In this case the relative orbit is closed and is defined as bounded. Relative bounded orbits, when close to circular, are fully described by the relative eccentricity and inclination vectors and have an intuitive graphical interpretation in the Radial-Tangential-Normal (RTN) reference frame centered in the chief spacecraft. This interpretation is shown in Figure 2.2, whereas a representation of the RTN frame with respect the the Earth-Centered Inertial (ECI) frame is shown in Figure 2.3.

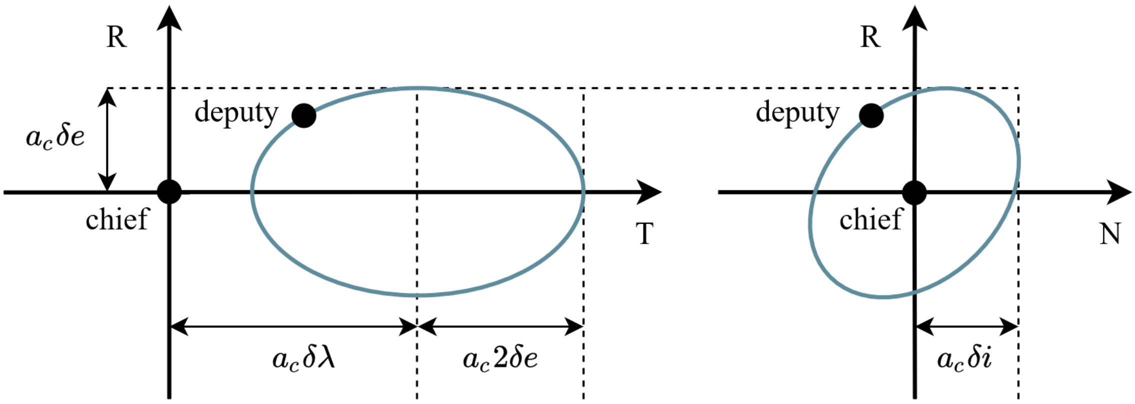


Figure 2.2: Graphical representation of Relative Orbital Elements in the Radial-Tangential-Normal reference frame centered in the chief spacecraft [24].

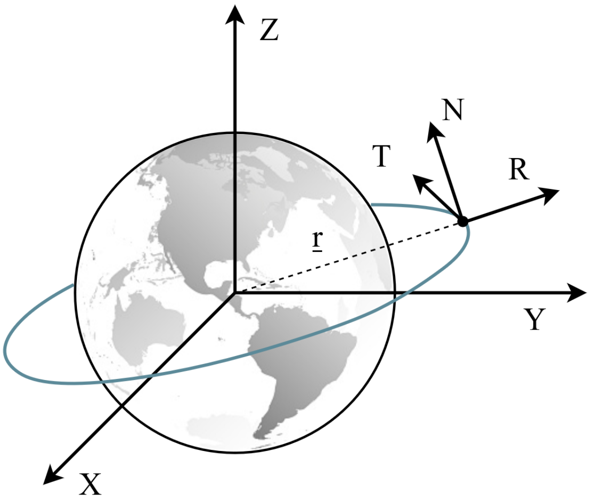


Figure 2.3: Radial-Tangential-Normal reference frame representation with respect to the Earth-Centered Inertial (ECI) reference frame. X axis coincides with Vernal equinox.

In a drag-free environment, a smart design of J2-invariant orbits allows to maintain the formation geometry with collision-free motion for hundreds of orbits with no additional station keeping [25]. That is why the majority of current research is focused on formation reconfiguration between invariant orbits, rather than formation maintenance for extended periods of time.

2.2.3. Atmospheric drag

One of the scopes of this thesis is to propose a guidance and control strategy for formation control in an environment in which drag effects are dominant and must be considered in the dynamics. In presence of drag, relative and absolute formation maintenance becomes mandatory, as also invariant relative bounded orbits will start drifting over time if uncontrolled. Different techniques have been proposed in recent research to include the effect of drag on Relative Orbital Elements by defining a proper state transition matrix. The main two options are the ones provided in References [11] and [13], namely:

- Density-free models which describe the effect of the presence of differential drag on the relative orbit geometry. In this case the state needs to be augmented with the time derivatives of the semi-major axis difference and of the two components of the relative eccentricity vector.
- A fully analytical density-specific model based on simplifying assumptions on the Harris-Priester atmosphere. The presented STM describes the variations of Relative

Orbital Elements depending on the orbit geometry and on the ballistic coefficient difference.

The accuracy of the first model only depends on the choice of the initial conditions for the augmented state variables $\delta\dot{a}$, $\delta\dot{e}_x$ and $\delta\dot{e}_y$. These can be found from previous integration of the nonlinear dynamics, finding the values which minimize the error. Even though the accuracy that can be reached is really good, this model totally lacks information on ballistic coefficient and density and, as a result, on a possible variation in time of these two quantities. It is true that atmospheric density and its evolution are unpredictable, but sudden variations would also affect a density-free model in which the initial conditions are evaluated on previous flight data. Moreover, a change in ballistic coefficient in time, e.g. for observations at different locations every orbit, would be impossible to be considered in the propagation.

On the other hand, the second model does introduce a dependence on density and on the ballistic coefficient, but is based on empirical correlations deriving from the assumed atmospheric model. Furthermore, it is reported that the model provides a reasonable approximation of drag dynamics for orbits of eccentricity between 0.1 and 0.9, leaving circular orbits outside the validity range.

In this document a third approach to derive a plant matrix is proposed, under the following assumptions:

- Control is assumed to be performed with respect to a non-decaying orbiting point, in order to perform relative transfers while maintaining the desired orbit altitude. In this case, the ΔB term in the ROE vector will be equal to 1.
- The perturbing acceleration due to the presence of drag is purely tangential.
- The time variations of the argument of perigee and true anomaly due to drag are negligible.
- The difference between absolute velocity in the ECI frame and velocity relative to the atmosphere is negligible.

Under these simplifications, the time variations of relative semi-major axis and components of the relative eccentricity vector due to the presence of drag reduce to:

$$\begin{cases} \delta\dot{a} = \frac{\dot{a}}{a_c} \\ \delta\dot{e}_x = \dot{e} \cdot \cos(\omega) \\ \delta\dot{e}_y = \dot{e} \cdot \sin(\omega) \end{cases} \tag{2.9}$$

The effect of the presence of atmospheric drag on semi-major axis and eccentricity can be retrieved from Gauss Variational Equations (GVE) [26] as:

$$\begin{cases} \dot{a} = \frac{2a^2 v}{\mu} \cdot u_{drag} \\ \dot{e} = \frac{2(e+\cos(\theta))}{v} \cdot u_{drag} \end{cases} \tag{2.10}$$

In which the drag disturbance acceleration u_{drag} is retrieved from the expression:

$$u_{drag} = \frac{1}{2} B_d \rho v^2 \tag{2.11}$$

Where $B_d = \frac{C_D A}{m}$ is the deputy spacecraft ballistic coefficient and ρ is the atmospheric density computed with any model of choice. The resulting drag plant matrix $\mathbf{A_{drag}}$ can then be calculated as:

$$\mathbf{A_{drag}} = \frac{1}{2} B_d \rho v^2 \cdot \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & \frac{2a^2 v}{\mu} \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & \frac{2(e+\cos(\theta))}{v} \cdot \cos(\omega) \\ 0 & 0 & 0 & 0 & 0 & 0 & \frac{2(e+\cos(\theta))}{v} \cdot \sin(\omega) \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \tag{2.12}$$

In order to gain a better understanding of the physical effects of differential drag on the formation dynamics, these can be added to the ones of keplerian motion and J2 and be visualized as drifts or rotations in the Relative Orbital Elements space. The combined effects are shown in Figure 2.4. In particular, differential drag introduces:

- a linear drift of δa.
- a quadratic motion of $\delta\lambda$ due to the coupling between differential drag and keplerian relative motion.
- a linear drift of the relative eccentricity vector parallel to the phase angle of the chief argument of perigee.

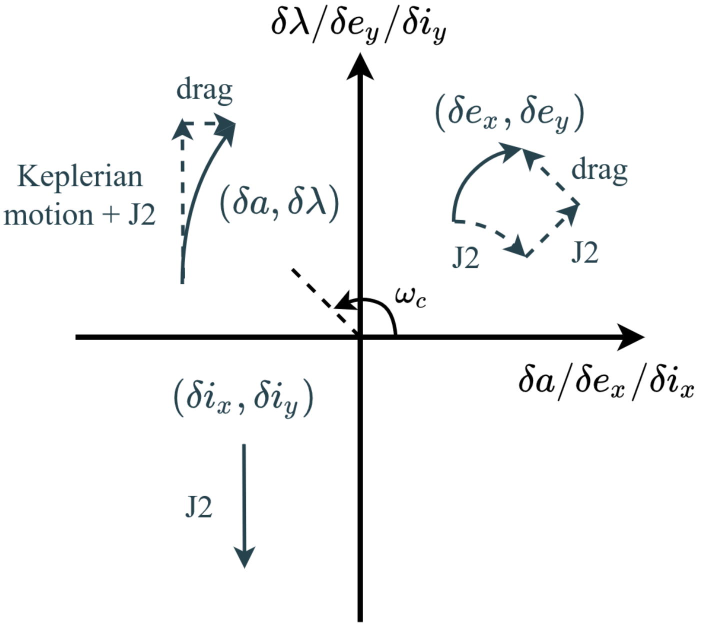


Figure 2.4: Combined effects of keplerian motion, J2 and atmospheric drag on ROE [11].

2.2.4. Drag plant matrix validation

The previously derived plant matrix is validated against the numerical results obtained with an orbital propagator considering only the effects of earth geopotential and drag, in order to isolate their influence. In particular, the error in the first four components of the Relative Orbital Elements state are retrieved and compared to the ones obtained with a plant matrix only considering the effects of J2 and keplerian motion. The validation is performed for an integration time of five orbital periods both for a circular and for a highly eccentric orbit with low altitudes of perigee, to check for the generality of the model. The starting osculating orbital elements of the selected orbits are reported in Table 2.1. The properties of the propagated satellites are listed in Table 2.2. Simulations start on 21st March 2021 at midnight.

Table 2.1: Starting orbital elements of the numerically propagated orbits used in the drag plant matrix validation.

Orbit	a [km]	e [-]	i [°]	Ω [°]	ω [°]	M_0 [°]
Case 1, LEO	6771	0.001	97.05	30	90	45
Case 2, HEO	16928	0.6	63.4	120	270	45

Table 2.2: Properties of the propagated satellites to validate the drag plant matrix. The chief satellite is propagated with $B = 0$ in order to be a non-decaying point.

	Chief	Deputy
Mass [kg]	20	20
B [m^2/kg]	0	0.0105
C_R [-]	1	1

The results of the simulations are reported in Figures 2.5 and 2.6. In both cases it is possible to observe that, using the plant matrix including drag effect, a considerably higher accuracy is obtained for all the analyzed Relative Orbital Elements.

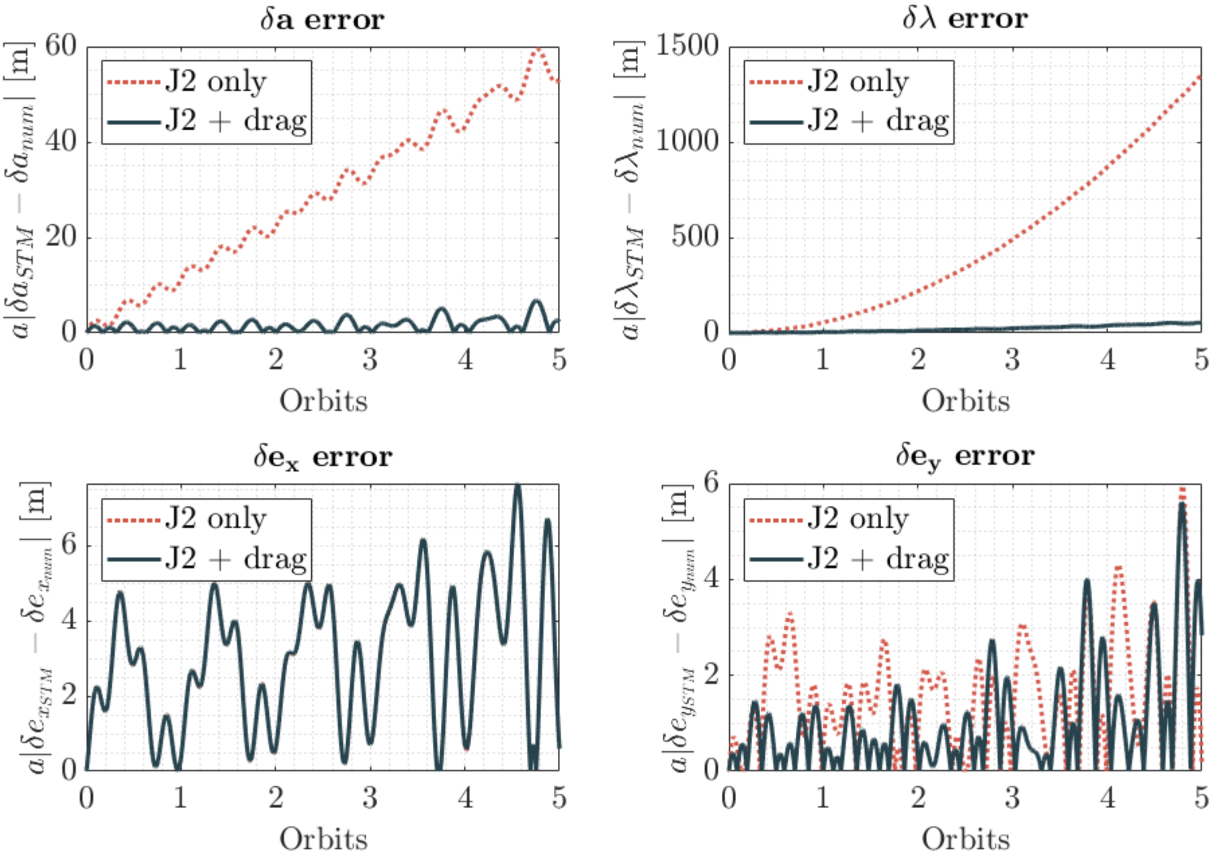


Figure 2.5: Results of the drag matrix validation for the circular LEO of Case 1. The error is shown as the absolute value of the difference between the ROE propagated with STMs and the ones that result from the numerical propagation.

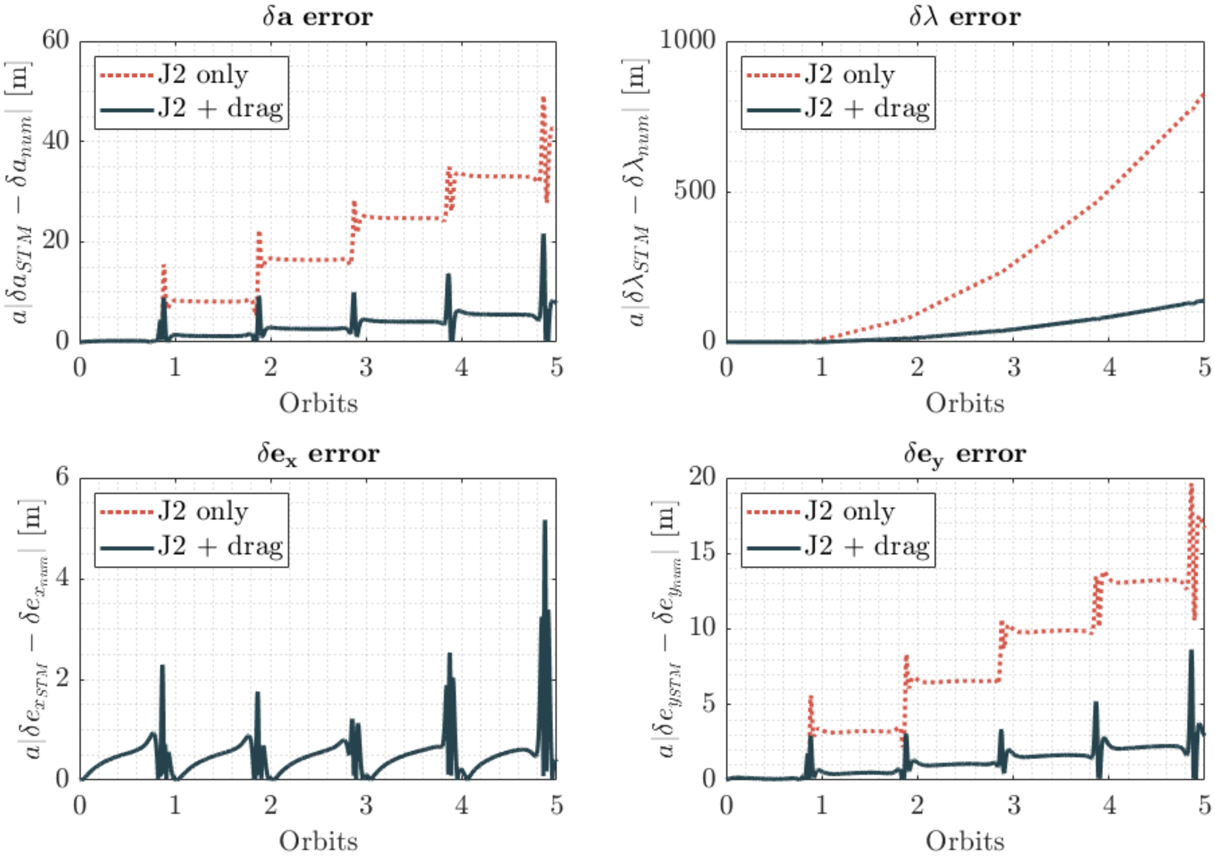


Figure 2.6: Results of the drag matrix validation for the elliptical orbit of Case 2. The error is shown as the absolute value of the difference between the ROE propagated with STMs and the ones that result from the numerical propagation.

As predictable, the most noticeable improvements are obtained for relative semi-major axis and for relative mean longitude, as these are the two ROEs which are most affected by the presence of a differential drag. An increase in accuracy can be noticed also for the relative eccentricity vector elements. This is more evident for the highly eccentric orbit and for the e_y component, since the argument of perigee is equal to 90°.

2.2.5. Control input

In order to map the control accelerations in the Radial-Tangential-Normal (RTN) reference frame into their effect onto the variation of Relative Orbital Elements, the so-called control matrix $\mathbf{B}$ must be defined.

For quasi-nonsingular Relative Orbital Elements the expression is taken from [24] and is reported in the following:

$$\mathbf{B} = \frac{1}{a_c n_c}\begin{bmatrix} \frac{2}{\eta}e_c\sin(\theta_c) & \frac{2}{\eta}(1+e_c\cos(\theta_c)) & 0 \\ -\frac{2\eta^2}{1+e_c\cos(\theta_c)} & 0 & 0 \\ \eta\cdot\sin(u_c) & \eta\frac{(2+e_c\cos(\theta_c))\cos(u_c)+e_x}{1+e_c\cos(\theta_c)} & \frac{\eta e_y}{\tan(i_c)}\frac{\sin(u_c)}{1+e_c\cos(\theta_c)} \\ -\eta\cdot\cos(u_c) & \eta\frac{(2+e_c\cos(\theta_c))\sin(u_c)+e_y}{1+e_c\cos(\theta_c)} & -\frac{\eta e_x}{\tan(i_c)}\frac{\sin(u_c)}{1+e_c\cos(\theta_c)} \\ 0 & 0 & \eta\frac{\cos(u_c)}{1+e_c\cos(\theta_c)} \\ 0 & 0 & \eta\frac{\sin(u_c)}{1+e_c\cos(\theta_c)} \\ 0 & 0 & 0 \end{bmatrix} \tag{2.13}$$

2.2.6. ΔV lower bound for impulsive control

For impulsive control, a closed-form solution for arbitrary changes in Relative Orbital Elements is available in literature under the assumption of purely keplerian unperturbed relative motion [27]. This solution derives from the study of the components of the control matrix **B** and establishes the absolute minimum required ΔV for any given reconfiguration, together with the optimal locations in terms of argument of latitude for the prescribed manoeuvres. This information can be used as a useful benchmark to test the performances of the Model Predictive Controller, evaluating its fuel efficiency. A summary of these closed-form solutions is provided in Table 2.3, the reported ΔV lower bound and optimal locations are derived under the assumption that only tangential maneuvers are needed to reconfigure the in-plane ROEs. This is true even if, as appearing in the **B** matrix, relative mean longitude can be directly controlled only with radial accelerations, since it is possible to control it by acting on the relative drift induced by a semi-major axis difference.

Table 2.3: Summary of closed-form impulsive solutions for formation reconfigurations. Only tangential manoeuvres are considered for changes of in-plane Relative Orbital Elements.

ROE shift	Thrust direction	Optimal location	ΔV lower bound [m/s]
$\Delta\delta a$	tangential	$u = 0$	$\frac{\eta_c}{2(1+e_c)}n_c a_c\lvert\Delta\delta a\rvert$
$\Delta\delta\lambda$	tangential	$u = 0$	$\frac{\eta_c}{3(1+e_c)\Delta M}n_c a_c\lvert\Delta\delta\lambda\rvert$
$\lvert\lvert\Delta\delta\underline{\mathbf{e}}\rvert\rvert$	tangential	$u = \pi \pm \arccos(e)$	$\frac{\eta_c}{\sqrt{3e_c^4-7e_c^2+4}}n_c a_c\lvert\lvert\Delta\delta\underline{\mathbf{e}}\rvert\rvert$
$\lvert\lvert\Delta\delta\underline{\mathbf{i}}\rvert\rvert$	normal	$u = \arctan\left(\Delta\delta i_y/\Delta\delta i_x\right)$	$\frac{1-e_c}{\eta_c}n_c a_c\lvert\lvert\Delta\delta\underline{\mathbf{i}}\rvert\rvert$

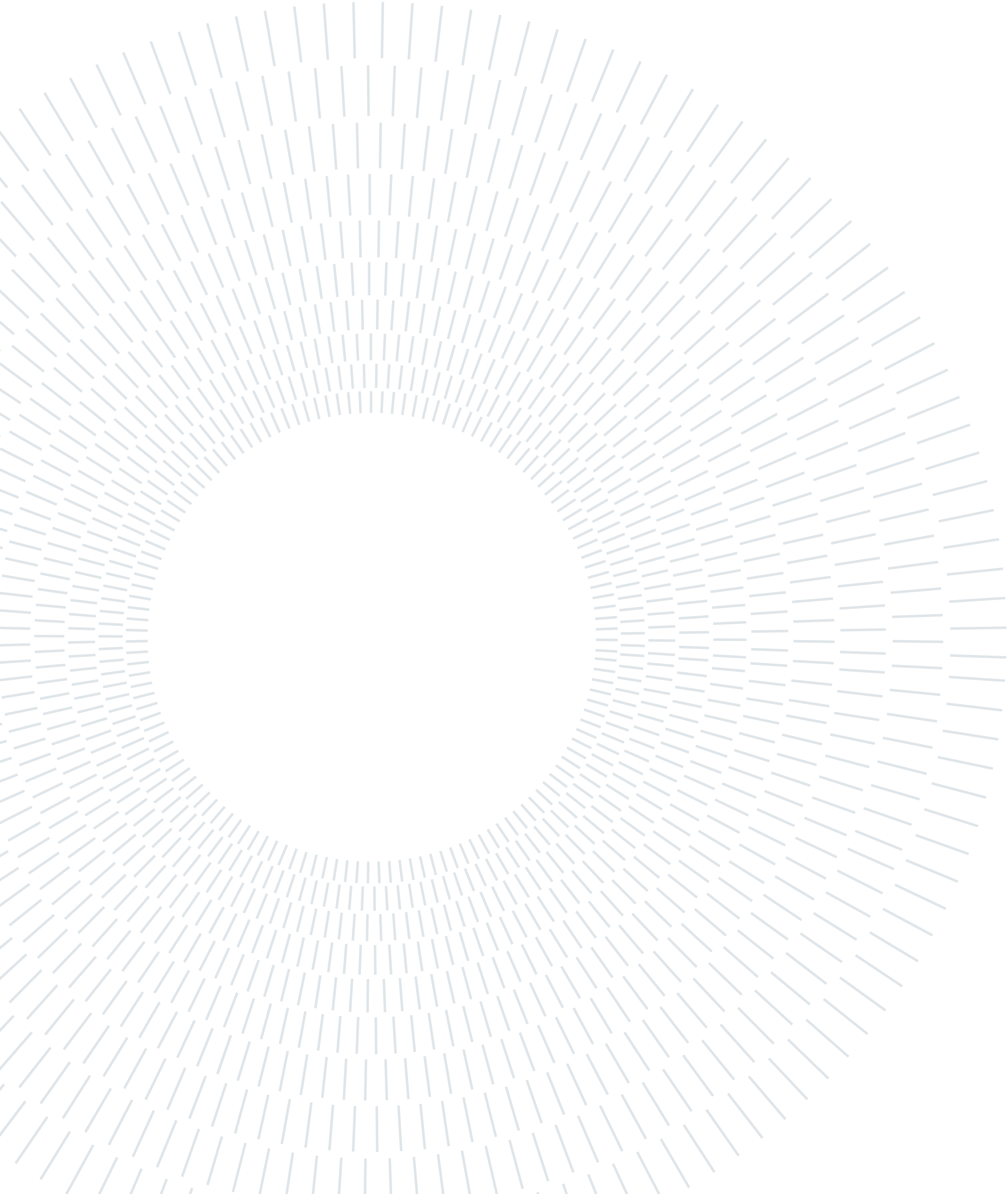
It can be noticed that the lower bound for a reconfiguration of relative mean longitude $\delta\lambda$ depends on the mean anomaly shift $\Delta M = M_f - M_0$ during the control interval. Thus, as the transfer time increases, the same mean longitude separation can be obtained with a lower ΔV.

2.2.7. Mean Orbital Elements propagation

Along with the propagation of ROE, a linear propagation of Mean Orbital Elements is necessary to compute the updated plant and control matrices at each time step, according to the orbit shape. In this case, the analytic propagation in performed by considering keplerian motion and J2 secular effects on Ω, ω and M:

$$\begin{cases} \dot{a} = 0 \\ \dot{e} = 0 \\ \dot{\Omega} = -\frac{3}{2} n J_2 \left(\frac{R_E}{p}\right)^2 \cos(i) \\ \dot{i} = 0 \\ \dot{\omega} = \frac{3}{4} n J_2 \left(\frac{R_E}{p}\right)^2 (4 - 5\,\sin^2(i)) \\ \dot{M} = n + \frac{3}{2} n J_2 \left(\frac{R_E}{p}\right)^2 (1 - \frac{3}{2}\,\sin^2(i))\sqrt{1-e^2} \end{cases} \tag{2.14}$$

The initial MOE to propagate can be retrieved with an osculating-to-mean transformation of the observed satellite state at the beginning of the propagation.



3 | Model Predictive Control

3.1. Control logic

Model Predictive Control (MPC) is a modern control technique merging the advantages of optimal and feedback control. First, the controller exploits the knowledge of the system dynamics to solve an Optimal Control Problem (OCP) over a specified period of time defined as prediction horizon, discretized according to a selected sampling step. The OCP allows for easy implementation of constraints to be satisfied by the controller and easy handling of multi-input multi-output (MIMO) systems. Once the optimization is completed, the resulting control is applied for a number of time steps defined by the so-called control horizon, then the optimization starts again taking as initial condition the new observed state of the system, converging to the target and compensating the inaccuracies of the dynamics model implemented in the OCP. A simple flow diagram representing the Model Predictive Control loop is reported in Figure 3.1.

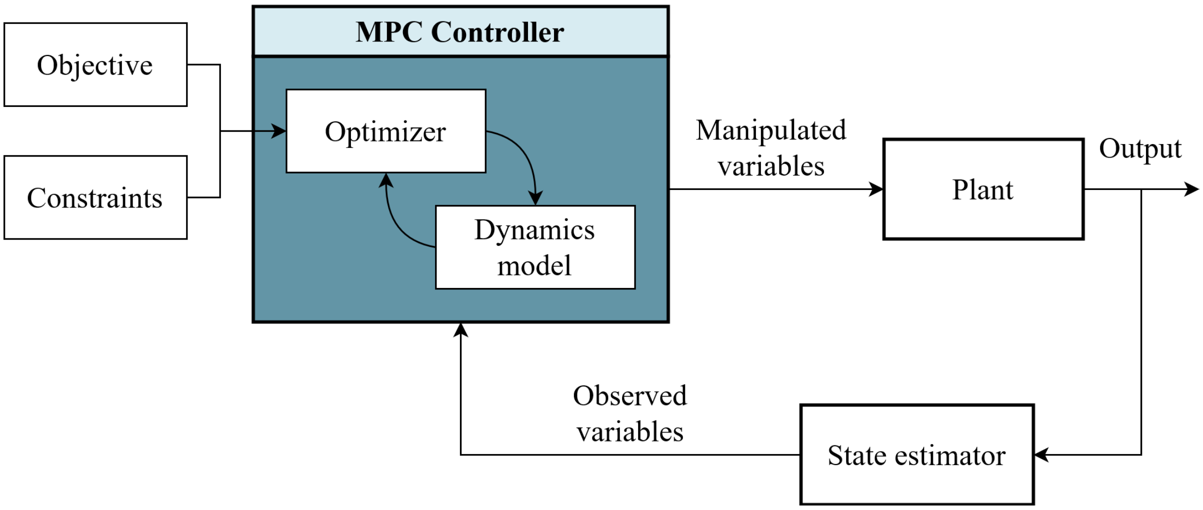


Figure 3.1: Model Predictive Control simplified scheme.

The dynamic nature of Model Predictive Control makes the algorithm effective in compensating any kind of disturbance, even the ones which are not modeled in the dynamics propagation. However, a good modeling of the dynamics improves both accuracy and fuel efficiency of the resulting trajectory.

3.1.1. Choice of MPC parameters

The overall performance of a Model Predictive Controller is strongly linked with the selection of its three fundamental parameters. The specific values of these numbers depend from one application to another, but some general rules can be identified to help in the MPC tuning process.

- **Prediction horizon**: the prediction horizon (PH) defines how far ahead in the future the controller is able to foresee the evolution of the dynamics. A value which is too low should be avoided, as the controller would not be able to take advantage of the long-term dynamics evolution to find the optimal control. On the contrary, a value which is too large leads to an excessive computational time which may even be unnecessary, compromising on-board implementation.
- **Sampling time**: the sampling time T_s defines the time span every which the parameters in the optimization vector are updated. A smaller value is always desirable in terms of accuracy, as a faster update is able to better represent the dynamics evolution and to modulate the control action more precisely if needed. However, keeping constant the prediction horizon, a smaller sampling time implies a larger dimension of the optimization vector. Since the optimization computational time depends on the problem size, a sampling time which is too small may prohibit on-board implementation.
- **Control horizon**: the control horizon (CH) indicates for how many sampling steps the control resulting from the solved optimal problem is applied, before restarting the optimization to update the control profile. The most robust option is to recompute the control every sampling time, in order to deal with unpredicted changes in the external conditions as soon as possible. However, longer control horizons relax the frequency at which the on-board computer needs to recompute the control profile, relieving computational effort that could be allocated to other tasks.

3.2. MPC simulation scheme

The scheme of the full simulation loop implemented in MATLAB to simulate the MPC response is shown in the flow diagram in Figure 3.2. The part of software that should be implemented on-board is highlighted in the figure in the light blue area.

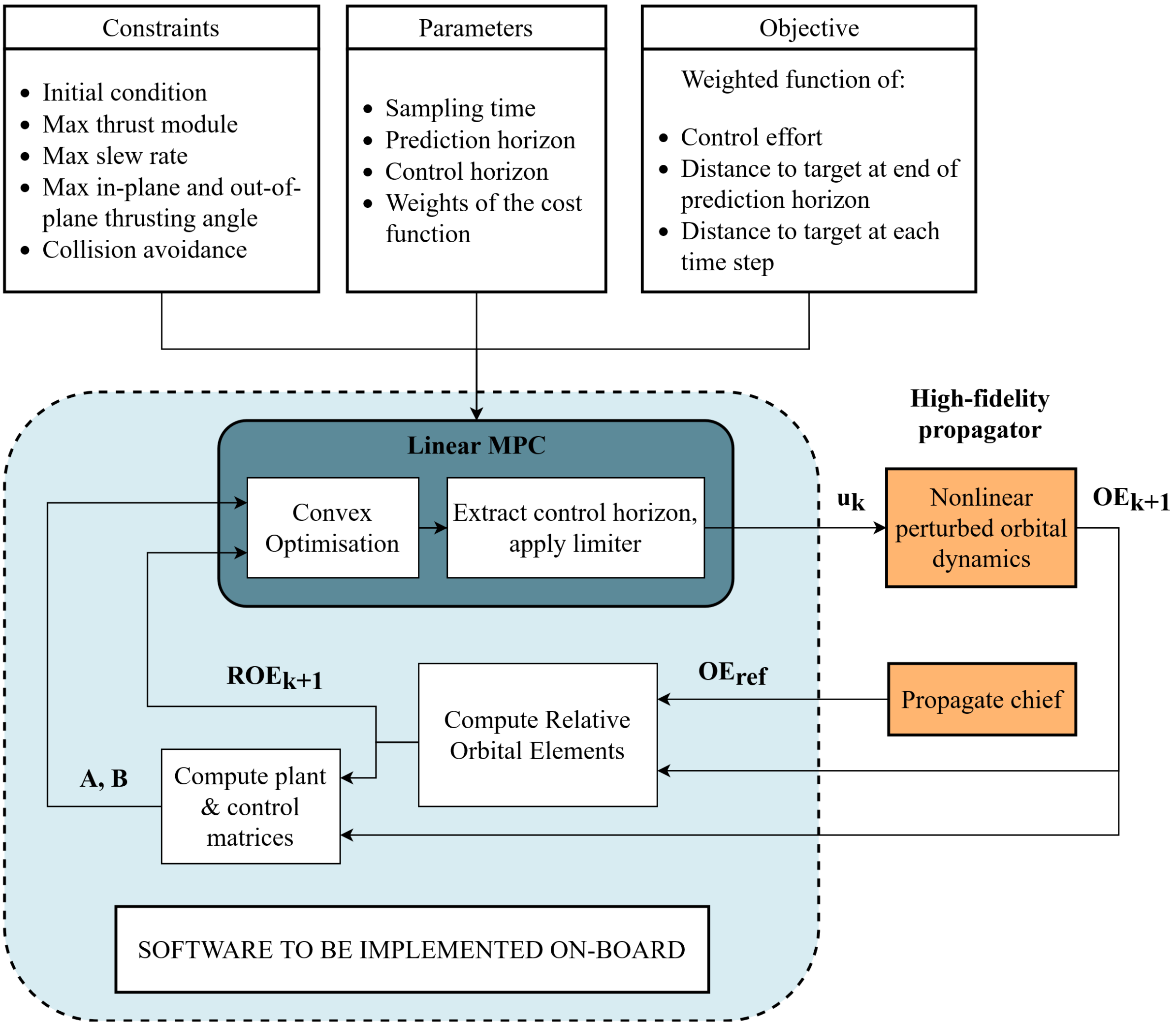


Figure 3.2: Simulation flow diagram. In the light blue box is reported the part of software that should be implemented on-board for a real application. In orange is shown the propagation part of the loop.

The convex optimisation block is the core of the Model Predictive Controller and its implementation will be discussed in detail in the following chapter. Another fundamental task of the on-board algorithm is to compute after each control horizon the new initial conditions in Relative Orbital Elements from the estimated states of the satellites in orbit and to retrieve the updated plant and control matrices, necessary to represent the dynamics with the desired accuracy. The actual dynamics of the spacecrafts in orbit is reproduced with a high-fidelity orbital propagator by integrating the MPC optimal control profile.

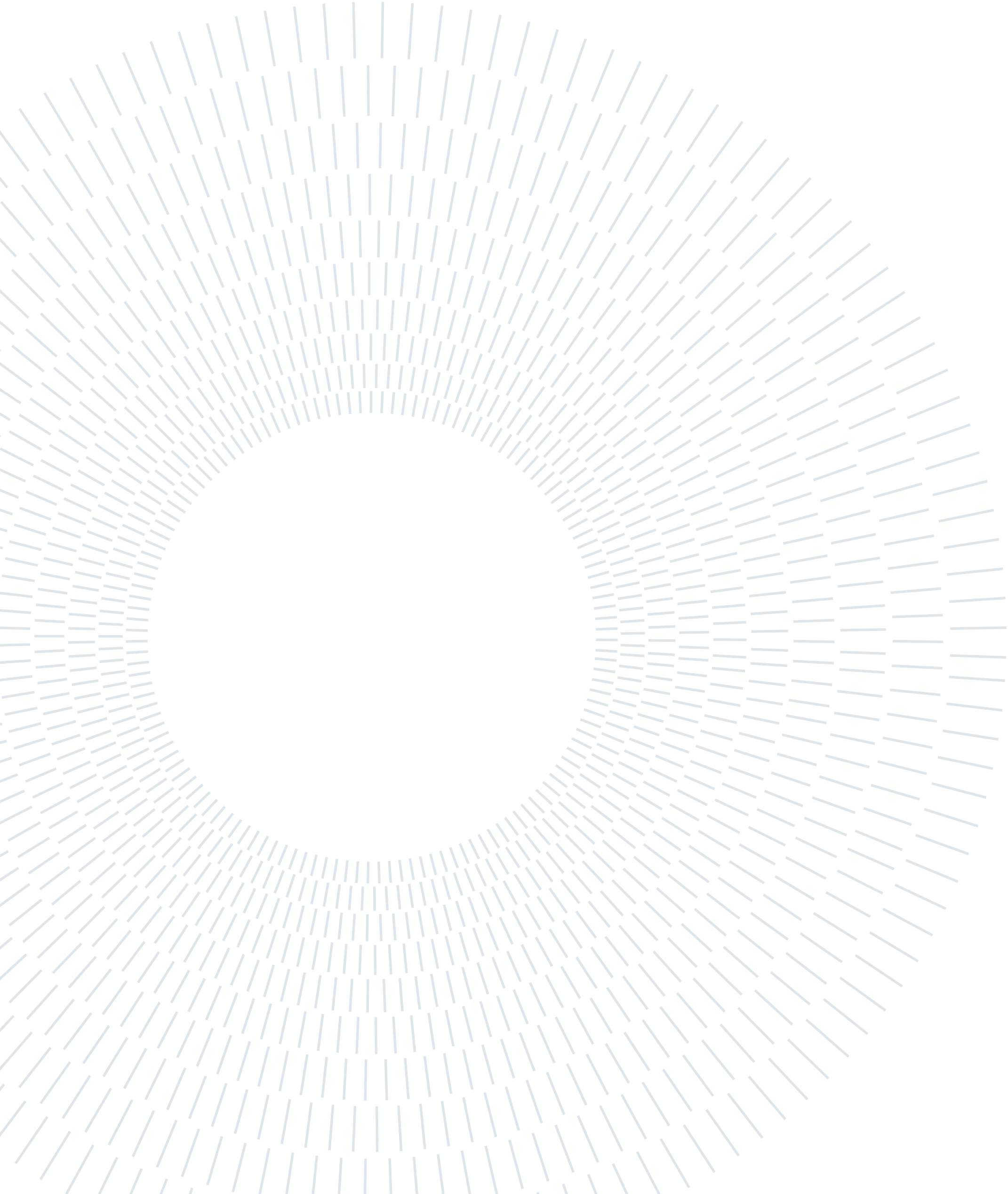
The simulation loop implementation is also reported in pseudo-code form in Algorithm 3.1. The number of propagated control horizons is defined as N_{steps}. For relative orbit transfers the simulation is stopped when the controlled satellites reach the desired relative state with a maximum allowed tolerance which is chosen depending on the application.

Algorithm 3.1 **Simulation scheme for MPC performance evaluation**

```
***Inputs:*** initial conditions $(\underline{\mathbf{OE}}_{\mathbf{ref}}(0), \underline{\mathbf{OE}}(0), \boldsymbol{\delta}\underline{\boldsymbol{\alpha}}(0), \mathbf{A}(0), \mathbf{B}(0))$
              target $(\boldsymbol{\delta}\underline{\boldsymbol{\alpha}_{\mathbf{T}}})$

for $i = 1 : N_{steps}$ do
    $\underline{\hat{\mathbf{X}}} \leftarrow$ [Solve convex Optimal Control Problem]
    $\underline{\mathbf{u}} \leftarrow$ [Extract control over CH]
    $\mathbf{OE}_{\mathbf{ref}}, \mathbf{OE} \leftarrow$ [Propagate reference and controlled spacecrafts over CH]
    $\boldsymbol{\delta}\underline{\boldsymbol{\alpha}}(t_{CH}) \leftarrow$ [Compute Relative Orbital Elements at end of CH]
    [Save trajectory, control and ROE history]
    if $|\boldsymbol{\delta}\underline{\boldsymbol{\alpha}}(t_{CH}) - \boldsymbol{\delta}\underline{\boldsymbol{\alpha}_{\mathbf{T}}}| \leq$ tol then
        [Break for loop]
    else
        [Update plant and control matrices]
        [Update initial conditions]
    end if
end for

***Outputs:*** trajectory, control profile, $\Delta V$
```

3.2.1. Chief and chasers propagation

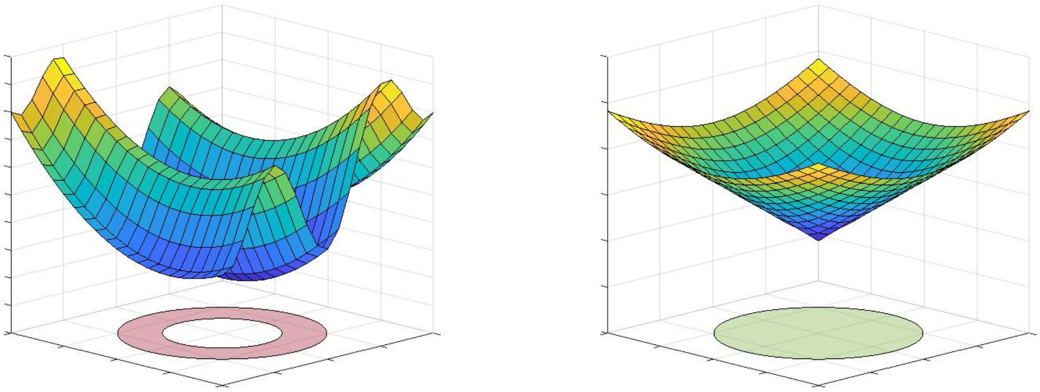
Chief and chaser spacecrafts are propagated independently in the algorithm. The reference virtual orbiting point is propagated taking into account only the desired perturbations effects, i.e. secular J2 effects for a sun-synchronous orbit. As anticipated in previous chapters, this non-decaying point may represent a chief satellite which is controlled absolutely to track a desired slot or a virtual point placed in the centroid of the formation. In doing so, the formation is controlled relatively while performing absolute station keeping at the same time. Another advantage of the virtual center strategy is that fuel balancing naturally results from the symmetry of the formulation. On the other hand, chaser spacecrafts are propagated applying the MPC control profiles with an orbital propagator taking into account all relevant perturbations, namely, earth geopotential up to the third zonal harmonic, atmospheric drag, solar radiation pressure (SRP), and third-body perturbations induced by the moon and the sun.



4 | Optimal Control Problem Formulation

4.1. Problem convexification

The main drawback of Model Predictive Control is the computational time needed to solve the optimal control problem at each time step. This is the reason why a linear relative orbital elements dynamics is introduced in the first place, along with the other previously described advantages. In addition, the optimization cost function and constraints can be convexified in order to take advantage of convex optimization algorithms, obtaining a much faster and more efficient solution viable for autonomous control. In fact, these solvers take advantage of the property that a point that is a local optimum is also a global optimum, thus, it is sufficient to find a local optimum to solve the problem [28]. Three-dimensional visual representations of convex and nonconvex functions and constraints are provided in Figure 4.1.



(a) Nonconvex objective function and constraints (b) Convex objective function and constraints

Figure 4.1: Examples of a convex and a nonconvex problem in three-dimensional space.

The objective of the OCP is to find the control input which minimizes a cost functional expressed as a weighted function of the tracking error and fuel consumption. In order to ensure feasibility of the optimal control problem, and therefore the convergence of the MPC, the final condition cannot be imposed as a hard constraint on the final state, but shall be included in the cost function.

To write the optimal control problem in convex form, the first step is to discretize it using a zero-order-hold approach. Time is divided into finite steps defined by the Model Predictive Controller sampling time, representing the sample interval for the state $\underline{\mathbf{x}}$ and the update interval for the control term $\underline{\mathbf{u}}$, which is considered piecewise constant for each time step. In the following, the complete convexification procedure for the problem dynamics and all included constraints is shown, following the approach that is commonly found in literature [18–21], modified where necessary to deal with the dynamics conveyed in quasi-nonsingular Relative Orbital Elements.

System dynamics: The linear dynamics previously derived in Equation (2.5) is discretized according to the Laplace transformation of the state equations. The procedure, described in Reference [18] discretizes the dynamics as:

$$\underline{\mathbf{x}}_{\mathbf{j}}[k+1] = e^{\mathbf{A}[k]\Delta t}\ \underline{\mathbf{x}}_{\mathbf{j}}[k] + \int_0^{\Delta t} e^{\mathbf{A}[k]\tau} d\tau\ \mathbf{B}[k]\ \underline{\mathbf{u}}_{\mathbf{j}}[k] \tag{4.1}$$

By recalling that [21]:

$$\int_0^{\Delta t} e^{\mathbf{A}[k]\Delta t} d\tau = \mathbf{A}^{-1}[k](e^{\mathbf{A}[k]\Delta t} - \mathbf{I}) \tag{4.2}$$

$$e^{\mathbf{A}[k]\Delta t} = \mathbf{I} + \mathbf{A}[k]\Delta t \tag{4.3}$$

this approach leads to the finite differences expression:

$$\underline{\mathbf{x}}_{\mathbf{j}}[k+1] = (\mathbf{A}[k]\Delta t + \mathbf{I})\ \underline{\mathbf{x}}_{\mathbf{j}}[k] + \mathbf{B}[k]\Delta t\ \underline{\mathbf{u}}_{\mathbf{j}}[k] \tag{4.4}$$

Now the problem can be expressed in convex formulation, with only convex equality or inequality constraints. To impose the dynamics as a linear equality constraint, the decisional optimisation vector will not contain only the control input, but also the state vector of relative orbital elements at each time step. In particular, for each satellite it is defined a column vector $\underline{\hat{\mathbf{x}}}_{\mathbf{j}}$ which contains the ROE state and the control input at each time instant k arranged as follows:

$$\underline{\hat{\mathbf{x}}_{\mathbf{j}}} = (\underline{\mathbf{x}_{\mathbf{j}}}^{1},\ \dots,\underline{\mathbf{x}_{\mathbf{j}}}^{k},\ \dots,\underline{\mathbf{x}_{\mathbf{j}}}^{K},\underline{\mathbf{u}_{\mathbf{j}}}^{1},\ \dots,\underline{\mathbf{u}_{\mathbf{j}}}^{k},\ \dots,\underline{\mathbf{u}_{\mathbf{j}}}^{K})^{T} \tag{4.5}$$

The vector $\underline{\hat{\mathbf{x}}_{\mathbf{j}}}$ has size $M = 7K + 3(K-1)$, where K is the number of time steps. Subsequently, the decisional vector of the optimisation problem is defined as:

$$\underline{\hat{\mathbf{X}}} = (\underline{\hat{\mathbf{x}}_{\mathbf{1}}},\ \dots,\underline{\hat{\mathbf{x}}_{\mathbf{j}}},\ \dots,\underline{\hat{\mathbf{x}}_{\mathbf{N}}})^{T} \tag{4.6}$$

with size $N \cdot M$, where N is the number of chaser satellites in the formation. This formulation allows to rewrite the dynamics of each spacecraft in Equation (4.4) in matrix form, following the work of Sarno et al. in [20]:

$$\mathbf{A_{sd}} \cdot \underline{\hat{\mathbf{x}}_{\mathbf{j}}} = 0 \tag{4.7}$$

Where $\underline{\mathbf{A}}_{\mathbf{sd}}$ is defined as:

$$\mathbf{A_{sd}} = \begin{bmatrix} \dots & \dots & \dots & \dots & \dots & \dots \\ \mathbf{0}_{7\times 7(k-1)} & -(\mathbf{I}_7 + \mathbf{A}[k]\Delta t) & \mathbf{I}_7 & \mathbf{0}_{7\times 7K-4k-10} & -\mathbf{B}[k]\Delta t & \mathbf{0}_{7\times 3(K-k-1)} \\ \dots & \dots & \dots & \dots & \dots & \dots \end{bmatrix} \tag{4.8}$$

with $k = 1, ..., K$. Accordingly, the dynamics of the entire formation expressed in convex form as a linear equality constraint can be written as:

$$\hat{\mathbf{A}}_{\mathbf{sd}} \cdot \underline{\hat{\mathbf{X}}} = 0 \tag{4.9}$$

Where:

$$\hat{\mathbf{A}}_{\mathbf{sd}} = \begin{bmatrix} \dots & \dots & \dots \\ \mathbf{0}_{7(K+1)\times M(j-1)} & \mathbf{A_{sd}} & \mathbf{0}_{7(K+1)\times M(N-j)} \\ \dots & \dots & \dots \end{bmatrix} \qquad j = 1, ..., N \tag{4.10}$$

Initial and final conditions: similarly, the initial and final ROE state can be isolated from the decisional vector by defining some proper matrices as shown in the following relations:

$$\begin{cases} \delta\underline{\boldsymbol{\alpha}}_{\mathbf{j}}^{\mathbf{0}} = \begin{bmatrix} \mathbf{I}_7 & \dots \\ \dots & \mathbf{0}_{M-7} \end{bmatrix} \cdot \hat{\underline{\mathbf{x}}}_{\mathbf{j}} = \mathbf{A}_{\mathbf{IC}_\mathbf{j}} \cdot \hat{\underline{\mathbf{x}}}_{\mathbf{j}} \\ \\ \delta\underline{\boldsymbol{\alpha}}_{\mathbf{j}}^{\mathbf{F}} = \begin{bmatrix} \mathbf{0}_{7K-7} & \dots & \dots \\ \dots & \mathbf{I}_7 & \dots \\ \dots & \dots & \mathbf{0}_{3(K-1)} \end{bmatrix} \cdot \hat{\underline{\mathbf{x}}}_{\mathbf{j}} = \mathbf{A}_{\mathbf{FC}_\mathbf{j}} \cdot \hat{\underline{\mathbf{x}}}_{\mathbf{j}} \end{cases} \tag{4.11}$$

Tins formulation can again be extended to all satellites in the formation:

$$\begin{cases} \delta\underline{\boldsymbol{\alpha}}^{\mathbf{0}} = \begin{bmatrix} \dots & \dots & \dots \\ \mathbf{0}_{M\times M(j-1)} & \mathbf{A}_{\mathbf{IC}_\mathbf{j}} & \mathbf{0}_{M\times M(N-j)} \\ \dots & \dots & \dots \end{bmatrix} \cdot \hat{\underline{\mathbf{X}}} = \mathbf{A}_{\mathbf{IC}} \cdot \hat{\underline{\mathbf{X}}} \\ \\ \delta\underline{\boldsymbol{\alpha}}^{\mathbf{F}} = \begin{bmatrix} \dots & \dots & \dots \\ \mathbf{0}_{M\times M(j-1)} & \mathbf{A}_{\mathbf{FC}_\mathbf{j}} & \mathbf{0}_{M\times M(N-j)} \\ \dots & \dots & \dots \end{bmatrix} \cdot \hat{\underline{\mathbf{X}}} = \mathbf{A}_{\mathbf{FC}} \cdot \hat{\underline{\mathbf{X}}} \end{cases} \tag{4.12}$$

Cost function: the cost functional to be minimized in the OCP must be expressed in terms of the discretized decisional vector $\hat{\underline{\mathbf{X}}}$. Tracking error is found by isolating the final condition with the matrix $\mathbf{A}_{\mathbf{FC}}$ defined in Equation (4.12). Likewise, also the control term must be isolated from $\hat{\underline{\mathbf{X}}}$ by using proper matrices. For each satellite it yields:

$$\hat{\underline{\mathbf{u}}}_{\mathbf{j}} = \hat{\mathbf{H}}_{\mathbf{j}} \cdot \hat{\underline{\mathbf{x}}}_{\mathbf{j}}, \qquad \hat{\mathbf{H}}_{\mathbf{j}} = \begin{bmatrix} \mathbf{0}_{7K} & \mathbf{0}_{7K\times 3(K-1)} \\ \mathbf{0}_{3(K-1)\times 7K} & \mathbf{I}_{3(K-1)} \end{bmatrix} \tag{4.13}$$

Accordingly, for the entire formation:

$$\hat{\underline{\mathbf{U}}} = \hat{\mathbf{H}} \cdot \hat{\underline{\mathbf{X}}}, \qquad \hat{\mathbf{H}} = \begin{bmatrix} \dots & \dots & \dots \\ \mathbf{0}_{M\times M(j-1)} & \hat{\mathbf{H}}_{\mathbf{j}} & \mathbf{0}_{M\times M(N-j)} \\ \dots & \dots & \dots \end{bmatrix} \tag{4.14}$$

Where the $\underline{\hat{\mathbf{U}}}$ vector contains the control accelerations in their respective position and is equal to zero in its other components. By defining the weighting matrices $\mathbf{P}$ and $\mathbf{Q}$, the cost function is then expressed as:

$$J = ||\hat{\mathbf{H}} \cdot \underline{\hat{\mathbf{X}}}||_1 + ||\mathbf{P}(\mathbf{A_{FC}} \cdot \underline{\hat{\mathbf{X}}} - \underline{\hat{\mathbf{X}}}_\mathbf{T})||_1 + ||\mathbf{Q}(\underline{\hat{\mathbf{X}}}_\mathbf{ROE} - \underline{\hat{\mathbf{X}}}_{\mathbf{ROE_T}})||_1 \tag{4.15}$$

Namely:

- $||\hat{\mathbf{H}} \cdot \underline{\hat{\mathbf{X}}}||_1$ is the control effort term, added to minimize the propellant consumption finding a sub-optimal fuel efficient solution.
- $||\mathbf{P}(\mathbf{A_{FC}} \cdot \underline{\hat{\mathbf{X}}} - \underline{\hat{\mathbf{X}}}_\mathbf{T})||_1$ is the tracking error of the final state with respect to the desired ROE vector. Its minimization leads to the convergence of the state to target.
- $||\mathbf{Q}(\underline{\hat{\mathbf{X}}}_\mathbf{ROE} - \underline{\hat{\mathbf{X}}}_{\mathbf{ROE_T}})||_1$ identifies the difference between the target state and the state of the satellites at each time step. The introduction of this expression avoids a stall of the MPC, which may happen if control actions would be prescribed only after the control horizon. Moreover, a higher weight on this term makes the spacecrafts converge faster to the desired state, adding a control on the transfer time, although indirect.

Since the cost function is a sum of 1-norms, it is convex and also linear. Moreover, the introduction of tracking terms in the objective, rather than as a hard constraint on the final target (as in [19]), ensures the feasibility and also the convergence of the problem, if these terms are weighted enough. The choice of proper weighting matrices allows to prioritize or even isolate the tracking error relative to some Relative Orbital Elements with respect to others, depending on the scenario. The chosen values strongly depend on mission requirements, thus, their selection differs from case to case. However, two general considerations that must be taken into account are that the tracking errors should be weighted enough to make the spacecrafts converge to the desired state, as anticipated previously, and that the higher the weights on tracking error terms, the quicker the transfer will be. This last characteristic can be exploited in particular operational situations where timeliness is crucial.

To better understand this link, a simple relative inclination change is simulated, in which the target spacecraft needs to change its $a\delta i_y$ component from 100 m to 150 m. The weighting matrices $\mathbf{P}$ and $\mathbf{Q}$ are expressed in the form $k \cdot \mathbf{I_{dim}}$, where k is a variable weight in the range $[1.5, 3]$ and $\mathbf{I_{dim}}$ is the identity matrix of the correspondent dimension. All simulations are stopped whenever the target $a\delta i_y$ is reached with an accuracy of three meters, the results are reported in Figures 4.2 and 4.3.

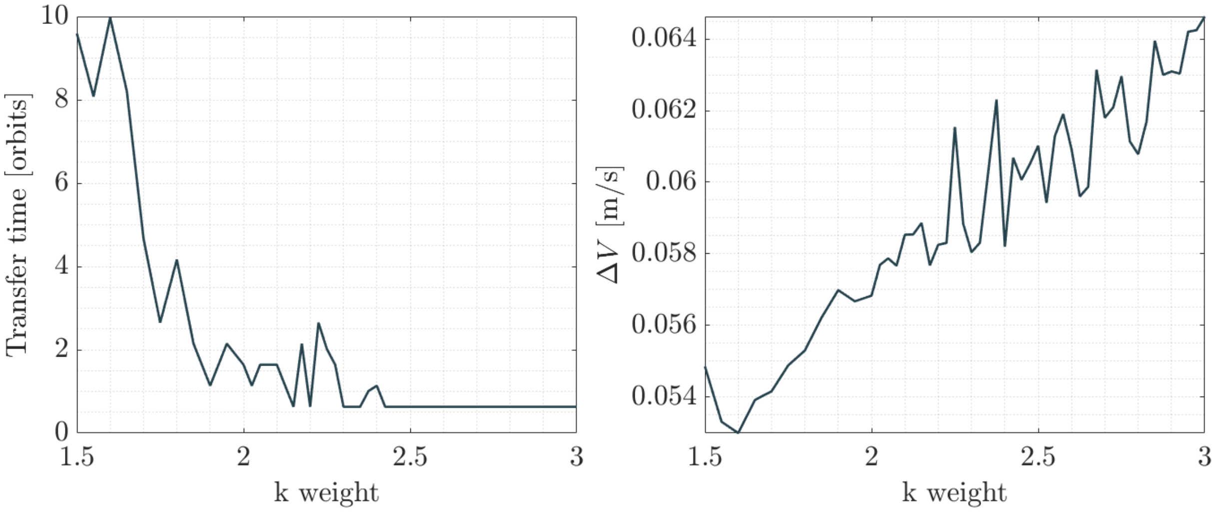


Figure 4.2: Transfer time and ΔV dependence on tracking error weights for a generic relative inclination change.

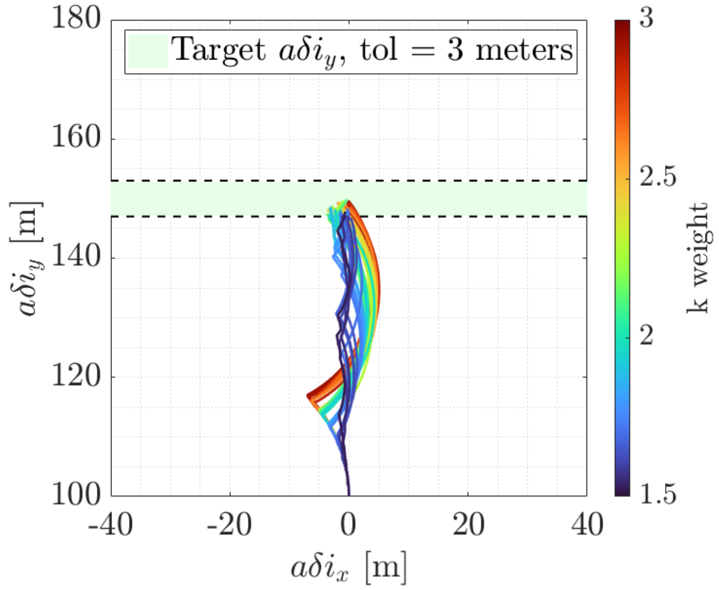


Figure 4.3: Impact of the tracking error weight on the evolution of relative inclination vector components.

In Figure 4.2 are represented the effects that an increase of the weight k has on transfer time and ΔV respectively. As anticipated, transfer time decreases by giving a higher weight on the tracking error terms in the cost function, reaching a saturation after a certain value, equal to about 2.5 in this specific case. Naturally, the effect extends to fuel consumption, indeed, even if the optimal impulsive solution in [15] does not expect a lower ΔV for longer transfer times in out-of-plane manoeuvres, an increase of the revolutions needed to complete the transfer allows the MPC to compute narrower thrusting arcs, more centered around fuel-optimal locations. This behaviour can also be visualized in the ROE space by plotting the evolution of the relative inclination vector components, shown in Figure 4.3. When a lower weight is given to tracking error terms, the trajectory in the ROE space gets closer to a straight line going from the initial condition to the target. Straight lines represent in the ROE space an optimal impulsive manoeuvre [15], therefore, a deviation from them displays a loss of optimality.

4.1.1. Actuation constraints

The constraints which should be added to the control problem to obtain a feasible control profile are strongly linked to the available propulsion solution, a single low-thrust engine in this case. In current research on relative dynamics, guidance and control it is often assumed that the spacecraft is capable of thrusting in any direction at any time. This is however rarely true, in particular for micro-satellite employing electric engines, unless the satellite is equipped with attitude thrusters that can be also used for orbit control. In the majority of cases, instead, the spacecraft mounts a single electric engine which can be manoeuvred acting on a gymbal or on the attitude of the spacecraft, with relevant limitations. This is why some constraints are included to take these effects into account.

Thrust module constraint: First of all, dealing with a single engine it is more convenient to pass from cartesian coordinates in the RTN frame to a module plus in-plane and off-plane angles representation. Accordingly, the constraint on the maximum achievable acceleration that the engine is capable of providing is given for the module rather than for each component, not to exceed its maximum capability. This is achieved isolating and reshaping the control at each time step k and for each satellite j from the decisional vector $\underline{\hat{\mathbf{X}}}$. Then, it is enough to impose that the norm of each control vector is lower than the maximum acceleration that the engine is able to provide to the spacecraft:

$$||\underline{\mathbf{u}}_j^k||_2 \leq \frac{T_{max}}{m_{s/c}} \qquad k = 1, ..., K, \quad j = 1, ..., N \tag{4.16}$$

Most of low-thrust propulsion units also have a limit in the lowest thrust they are able to provide. This constraint is impossible to express in convex form, as it would represent a "hole" in the control three-dimensional space. Therefore this constraint is imposed a posteriori on the minimization results, just by ignoring the rows of the output control matrix which have a norm lower than the minimum acceleration which can be provided by the engine. The algorithm will adapt just by providing a higher control action at a later time, converging anyway to the target.

Radial thrust: As can be noticed from the closed-form impulsive solutions reported in Chapter 2, all in-plane Relative Orbital Elements can be controlled effectively by only applying a tangential thrust. Therefore, radial thrust could be set to zero a priori in order to obtain a simpler solution. This constraint can be included in the algorithm just by isolating the radial control components of the decisional vector and imposing the equality to zero. Furthermore, setting the radial thrust to null, greatly simplifies the inclusion of constraints on thrusting angles in convex form. For the previous reasons, this constraint will be included in the algorithm for all the simulated scenarios.

Constraints on thrust angles and slew rates: During nominal operations of the spacecraft, having only one available propulsive unit, additional constraints should be imposed on the thrusting angle and on the maximum allowed slew rate to point the engine in the desired direction. For example, a thrust cone constraint may be included for orbit maintenance, trying to maintain the attitude as close as possible to earth-pointing in order to be ready for observation. On the other hand, a slew rate limit may be imposed for formation reconfiguration maneuvers, which may ask for a too quick and unfeasible redirectioning of the thrust vector.

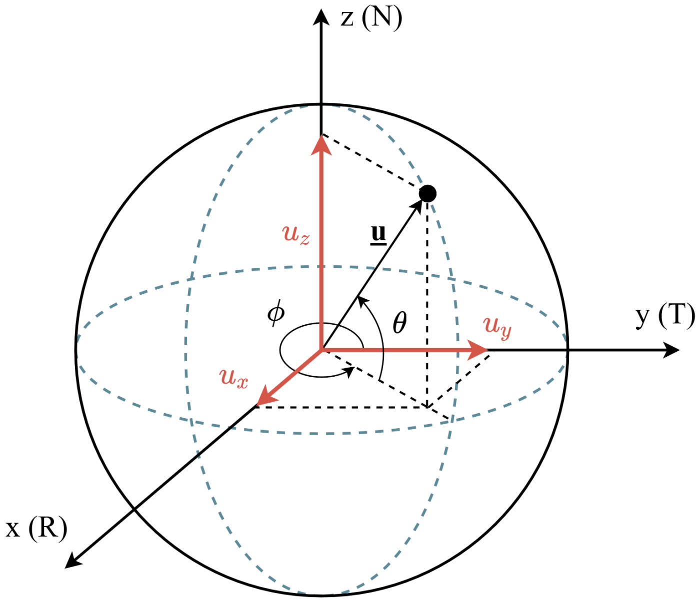


Figure 4.4: Spherical coordinates reference frame for the control acceleration vector. In red are represented the components of the control vector in the RTN reference frame.

The in-plane and off-plane thrusting angles are defined according to the spherical coordinates representation shown in Figure 4.4. In particular, the in-plane angle ϕ is defined starting from the T axis, which is coincident with the along-track direction for circular orbits, and taken positive in anticlockwise direction, on the other hand, the off-plane angle θ is the angle between the $\underline{\mathbf{u}}$ and the RT plane. These are defined in degrees as:

$$\begin{cases} \phi = \arctan\left(\frac{u_y}{u_x}\right) - 90° \\ \theta = \arcsin\left(\frac{u_z}{u}\right) \end{cases} \tag{4.17}$$

ϕ is defined from 0° to 360°, whereas θ is defined from -90° to 90°. Including the radial thrust constraint, the angle ϕ can only assume precisely a value of 0° or 180°. Therefore, a constraint on the in-plane angle can just be included on the sign of the u_y component. On the other hand, the constraint on the maximum off-plane angle θ is imposed in the form:

$$|\underline{\mathbf{u}}_{\mathbf{z}}| \leq |\underline{\mathbf{u}}_{\mathbf{y}}| \cdot \tan(\theta_{max}) \tag{4.18}$$

The inclusion of a constraint on the maximum slewing rate is not natural in convex form, due to the tight restrictions imposed by convex programming. The most basic idea consists in imposing a maximum difference between the components of the control vector at a given time step k and the same component at time $k+1$. The first obvious drawback of this strategy is that the same difference corresponds to different angles depending on the control vector module, until the introduction of the constraint becomes irrelevant for very small controlling accelerations. However, this effect is mitigated by the inclusion of the minimum thrust threshold explained before, which, if high enough, may be exploited for the inclusion of the slew constraint. Another undesired effect is that this type of check interprets the engine switch-off as a slew, due to the sudden variation of control components.

Despite all these stringent limitations, a procedure is found to at least deny the presence of immediate sign changes for each axis. This consists in imposing:

$$\begin{cases} |u_x^{k+1} - u_x^k| \leq a_{max} \\ |u_y^{k+1} - u_y^k| \leq a_{max} \\ |u_z^{k+1} - u_z^k| \leq a_{max} \end{cases} \tag{4.19}$$

Adding this constraint, if a minimum thrust limit higher than half of the maximum value is added, engine switch-offs are allowed, but at the same time sudden 180° slews are impossible. Indeed, at least a time-step with no control is mandatory before switching the

control sign of each component. This control-free step can be used to slew in the desired direction with enough time to perform a feasible manoeuvre, as the MPC sampling time is chosen in the order of the hundreds of seconds. If one time step is not enough to safely perform the slew in time, the previous formulation just needs to be extended also to the $k+2$ step and so on, according to the mission specifications and requirements.
Additional constraints may be added to cover slews between different axes if necessary, An example of a constraint for normal and tangential manoeuvre splitting can be imposed as:

$$\begin{cases} |u_y^{k+1}| + |u_z^k| \leq a_{max} \\ |u_z^{k+1}| + |u_y^k| \leq a_{max} \end{cases} \tag{4.20}$$

Naturally, this constraint does not deny impossible slews in all situations. However, in a real application, if the spacecraft is not able to perform a manoeuvre in time due to the saturation of attitude control actuators, the feedback behaviour of the MPC will compensate in future re-iterations, taking more time but converging to the target.

4.1.2. Collision avoidance

Collision avoidance must be imposed as an inequality constraint on the relative position of the satellites in the formation in the RTN frame. Since the state vector is expressed in ROE form, a linear transformation is needed to retrieve the RTN coordinates before imposing collision avoidance. The selected linear mapping, represented by the matrix **T**, is derived following the procedure reported by Silvestrini in [17], but neglecting the velocity components, which are not necessary for the collision avoidance algorithm. Also the seventh element of the state, the differential ballistic coefficient, is not needed and therefore is discarded.

$$\underline{\mathbf{x}}_{\mathbf{RTN}} = \begin{pmatrix} x \\ y \\ z \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix} = \mathbf{T} \cdot \boldsymbol{\delta}\underline{\boldsymbol{\alpha}} \tag{4.21}$$

The transformation matrix is derived by using an intermediate change of coordinates exploiting the classical orbital elements difference $\Delta OE = [\Delta a, \Delta M, \Delta\omega, \Delta e, \Delta i, \Delta\Omega]$ as follows:

$$\mathbf{T} = \frac{\partial \underline{\mathbf{x}}_{\mathbf{RTN}}}{\partial \Delta OE} \cdot \frac{\partial \Delta OE}{\partial \boldsymbol{\delta}\underline{\boldsymbol{\alpha}}} \tag{4.22}$$

The first-order approximation of the mapping between the RTN state and classical osculating orbital elements difference is taken from equations found in [23] and their derivatives:

$$\begin{cases} x = \frac{r}{a}\Delta a - a\cdot\cos(\theta)\Delta e + \frac{a\cdot e\cdot\sin(\theta)}{\sqrt{1-e^2}}\Delta M \\ y = \left(a + \frac{r}{1-e^2}\right)\sin(\theta)\Delta e + \frac{a^2}{r}\eta\Delta M + r\Delta\omega + r\cdot\cos(i)\Delta\Omega \\ z = r\cdot\sin(u)\Delta i - r\cdot\sin(i)\cos(u)\Delta\Omega \\ \dot{x} = -\frac{n\cdot e\cdot\sin(\theta)}{2\sqrt{1-e^2}}\Delta a + n\cdot\sin(\theta)\sqrt{1-e^2}\left(\frac{a^3}{r^2}\right)\Delta e + e\cdot n\cdot\cos(\theta)\left(\frac{a^3}{r^2}\right)\Delta M \\ \dot{y} = \left[n\sqrt{1-e^2}\left(1+\frac{r}{a(1-e^2)}\right)\left(\frac{a^3}{r^2}\right)\cos(\theta) + \frac{a\cdot e\cdot n\cdot\sin^2(\theta)}{(1-e^2)^{3/2}}\right]\Delta e + \\ \qquad -e\cdot n\cdot\sin(\theta)\left(\frac{a^3}{r^2}\right)\Delta M + \frac{a\cdot e\cdot n\cdot\sin(\theta)}{\sqrt{1-e^2}}\Delta\omega \\ \dot{z} = \frac{a\cdot n}{\sqrt{1-e^2}}[\sin(u) + e\cdot\sin(\theta)]\sin(i)\Delta\Omega + \frac{a\cdot n}{\sqrt{1-e^2}}[\cos(u) + e\cdot\cos(\theta)]\Delta i \end{cases} \tag{4.23}$$

From the previous equations, the first transformation matrix can be retrieved. Since only the relative positions are of interest, the last four rows can be set equal to zero to avoid useless computations. The same yields for the last column, as no dependence on the differential ballistic coefficient is present.

$$\frac{\partial \underline{\mathbf{x}}_{\mathbf{RTN}}}{\partial \Delta OE} = \begin{bmatrix} r/a & \frac{a\cdot e\cdot\sin(\theta)}{\sqrt{1-e^2}} & 0 & -a\cdot\cos(\theta) & 0 & 0 & 0 \\ 0 & \frac{a^2}{r}\eta & r & \left(a+\frac{r}{1-e^2}\right)\sin(\theta) & 0 & r\cdot\cos(i) & 0 \\ 0 & 0 & 0 & 0 & r\cdot\sin(u) & -r\cdot\sin(i)\cos(u) & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \tag{4.24}$$

In order to obtain the full mapping, the Jacobian of the transformation from classical orbital elements to quasi-nonsingular ROEs is necessary. This is obtained from the definition of $\boldsymbol{\delta}\underline{\boldsymbol{\alpha}}$ for $\delta OE \to 0$ as:

$$\frac{\partial \Delta OE}{\partial \boldsymbol{\delta}\underline{\boldsymbol{\alpha}}} = \begin{bmatrix} a & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & \frac{\sin(\omega)}{e} & -\frac{\cos(\omega)}{e} & 0 & \frac{\cos(i)}{\sin(i)} & 0 \\ 0 & 0 & -\frac{\sin(\omega)}{e} & \frac{\cos(\omega)}{e} & 0 & 0 & 0 \\ 0 & 0 & \cos(\omega) & \sin(\omega) & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & \sin(i) & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \tag{4.25}$$

Multiplying the two matrices, the complete transformation matrix $\mathbf{T}$ isolating relative positions in the RTN frame is retrieved.

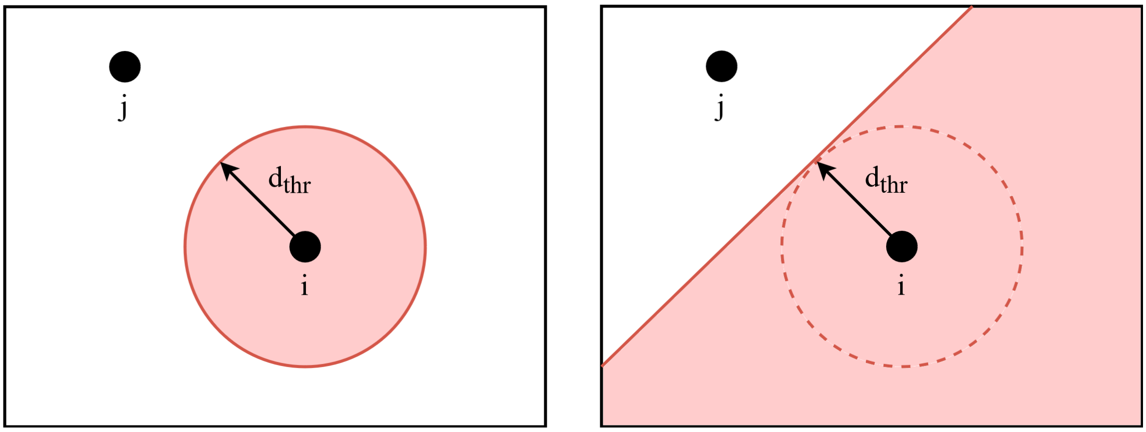
To impose collision avoidance on the full optimization decisional vector $\underline{\hat{\mathbf{X}}}$, the matrix $\mathbf{T}[k]$ needs to be evaluated at each time step to form the matrix $\hat{\mathbf{T}}_{\mathbf{j}}$, defined for the j-th satellite as follows:

$$\hat{\mathbf{T}}_{\mathbf{j}} = \begin{bmatrix} \begin{bmatrix} \cdots & \cdots & \cdots \\ \mathbf{0}_{7\times 7(k-1)} & \mathbf{T}[k] & \mathbf{0}_{7\times 7(K-k)} \\ \cdots & \cdots & \cdots \end{bmatrix} & \mathbf{0}_{7K\times 3(K-1)} \\ \mathbf{0}_{3(K-1)\times 7K} & \mathbf{0}_{3(K-1)} \end{bmatrix} \tag{4.26}$$

As for other previously derived matrices, the matrix for the entire formation can be assembled as [17]:

$$\hat{\mathbf{T}} = \begin{bmatrix} \cdots & \cdots & \cdots \\ \mathbf{0}_{M\times M(j-1)} & \hat{\mathbf{T}}_{\mathbf{j}} & \mathbf{0}_{M\times M(N-j)} \\ \cdots & \cdots & \cdots \end{bmatrix} \tag{4.27}$$

Once the x, y and z components of the decisional vector of each satellite are isolated, it is necessary to impose the minimum distance constraint in a formulation suitable for convex programming. The selected methodology is the one introduced by Morgan et al. in Reference [19].



(a) Nonconvex collision avoidance constraint. (b) Convex collision avoidance constraint.

Figure 4.5: 2D representation of the typical and convexified collision avoidance constraint [19]. In red it is represented the prohibited zone for the j-th satellite in the formation in the two cases.

Such strategy consists in generating separating planes among the satellites, transforming the circular prohibited zone into a suitable convex formulation. A graphical representation of the convexified constraint is reported in Figure 4.5. It can be observed that the convexified prohibited zone contains entirely the nonconvex one, this means that collision avoidance is guaranteed. Morgan's formulation states that, at each time instant k, the following inequality must be satisfied between each i/j couple of spacecrafts in the formation:

$$(\underline{\bar{\boldsymbol{\chi}}}_j[k]-\underline{\bar{\boldsymbol{\chi}}}_i[k])^T\mathbf{C}^T\mathbf{C}(\underline{\boldsymbol{\chi}}_j[k]-\underline{\boldsymbol{\chi}}_i[k]) \geq d_{thr}||\mathbf{C}(\underline{\bar{\boldsymbol{\chi}}}_j[k]-\underline{\bar{\boldsymbol{\chi}}}_i[k])||_2 \tag{4.28}$$

Where the solution vectors $\underline{\boldsymbol{\chi}}$ are expressed in cartesian coordinates in the RTN frame, $\underline{\bar{\boldsymbol{\chi}}}$ represents an initial guess of the optimal trajectory followed by the spacecraft and $\mathbf{C}$ is a matrix built to isolate only the position components of the solution. This formulation must be reformulated to deal with quasi-nonsingular ROEs exploiting the previously introduced linear mapping and with the entire decisional vector $\underline{\hat{\mathbf{X}}}$.

As explained previously, the linear mapping matrix $\hat{\mathbf{T}}$ is built to already isolate the positional components of the RTN frame. Therefore, the rigth term of Equation (4.28) can be rewritten for each time step k as:

$$d_{thr}||\mathbf{C}(\underline{\bar{\boldsymbol{\chi}}}_j[k]-\underline{\bar{\boldsymbol{\chi}}}_i[k])||_2 = d_{thr}||\mathbf{T}[k]\cdot(\underline{\bar{\mathbf{x}}}_j[k]-\underline{\bar{\mathbf{x}}}_i[k])||_2 \tag{4.29}$$

In order to deal with all time steps at once, i.e. to deal with the entire decisional vector, this expression must be generalized by defining a proper vector $\underline{\mathbf{d}}_{CA}$, built by taking the norm of difference of the positional components in the RTN frame of the initial guess vector at each time step. For the proposed algorithm, the initial guess is represented by a solution of the same problem obtained without imposing collision avoidance.

$$\underline{\mathbf{d}}_{CA} = d_{thr} \cdot \begin{bmatrix} \cdots \\ ||\mathbf{T}[k] \cdot (\underline{\bar{\mathbf{x}}}_j[k] - \underline{\bar{\mathbf{x}}}_i[k])||_2 \\ \cdots \end{bmatrix} \tag{4.30}$$

For what concerns the left term of Equation (4.28), the left half concerning the guess vector can be rewritten in matrix form by correctly rearranging the differences in the positional components of the initial guess vector transformed in RTN form, building a matrix $\bar{\mathbf{D}}$ as follows:

$$\bar{\mathbf{D}} = \begin{bmatrix} \cdots & \cdots & \cdots \\ \mathbf{0}_{1\times 3(k-1)} & [x_j^k - x_i^k,\ y_j^k - y_i^k,\ z_j^k - z_i^k] & \mathbf{0}_{1\times 3(K-k)} \\ \cdots & \cdots & \cdots \end{bmatrix} \tag{4.31}$$

On the other hand, the RTN positional differences between two satellites of the formation can be isolated from the decisional vector $\underline{\hat{\mathbf{X}}}$ defining a matrix $\hat{\mathbf{A}}_{CA}$ as:

$$\hat{\mathbf{A}}_{CA} = \begin{bmatrix} \cdots & \cdots & \cdots & \cdots & \cdots \\ \mathbf{0}_{3\times 7(k-1)} & \mathbf{I}_3 & \mathbf{0}_{3\times 2(5K-3)} & -\mathbf{I}_3 & \mathbf{0}_{3\times 10K-7k+1} \\ \cdots & \cdots & \cdots & \cdots & \cdots \end{bmatrix} \tag{4.32}$$

Finally, the reformulated expression for the convexified collision avoidance constraint to be implemented in the algorithm can be written for each satellite couple i/j as:

$$\bar{\mathbf{D}} \cdot (\hat{\mathbf{A}}_{CA} \hat{\mathbf{T}} \underline{\hat{\mathbf{X}}}_{\mathbf{i/j}}) \geq \underline{\mathbf{d}}_{CA} \tag{4.33}$$

Where the vector $\underline{\hat{\mathbf{X}}}_{\mathbf{i/j}}$ only contains the state and the control of the i-th and j-th satellites of the formation. The constraint ensures that the threshold distance is respected in the sample points of the optimization vector, but not between one sample and the following one. An analysis on the maximum crossing of the separating plane due to natural dynamics between two sampling steps could give a useful result to set a conservative threshold which takes this aspect into account.

Furthermore, the addition of the collision avoidance constraint may introduce a source of infeasibility in the problem, which may occur if the collision avoidance inequality clashes with the hard constraint on the initial condition. However, it is enough to impose a starting geometry in which all inter-satellite distances are larger than the CA threshold, then the MPC will guide the spacecrafts to the target respecting the safety margin.

4.2. OCP implementation in the convex solver

The problem is then implemented in a convex optimization solver as shown in Algorithm 4.1. The specific notation depends on the solver and on the language it operates on. For this study, `cvx 1.22` [29, 30] is used for simulations in MATLAB and the `CVXPY 1.1` library [31, 32] is used for simulations performed using Python. In Algorithm 4.1 the generic MATLAB implementation in cvx form is shown with the two selected parameters of solver and precision. `cvx 1.22` currently supports two different solvers: SeDuMi [33] and SDPT3 [34], with the first being the default and usually recommended option. The precision string argument allows to select the accuracy of the solution from a set of predefined precision models. For the proposed implementation the `best` precision model is selected, setting the so-called solver target to zero. This means that the solver continues as long as it can make progress, producing more accurate solutions. Precision can be lowered if faster solutions are desired.

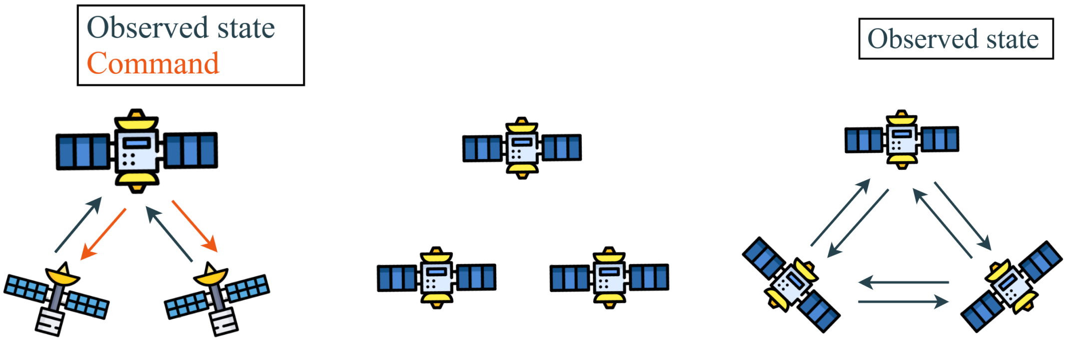
Algorithm 4.1 **Convex Optimal Control Problem formulation in cvx**

```
cvx_begin
cvx_solver: SeDuMi (or sdpt3)
cvx_precision: best

variable: $\underline{\hat{\mathbf{X}}}(N \cdot M)$

minimize: $J = ||\hat{\mathbf{H}} \cdot \underline{\hat{\mathbf{X}}}||_1 + ||\mathbf{P}(\mathbf{A_{FC}} \cdot \underline{\hat{\mathbf{X}}} - \underline{\hat{\mathbf{X}}}_\mathbf{T})||_1 + ||\mathbf{Q}(\underline{\hat{\mathbf{X}}}_\mathbf{ROE} - \underline{\hat{\mathbf{X}}}_{\mathbf{ROE_T}})||_1$

subject to:
    $\hat{\mathbf{A}}_\mathbf{sd} \cdot \underline{\hat{\mathbf{X}}} = 0 \leftarrow$ [Dynamics, eq.(4.9)]
    $\mathbf{A_{FC}} \cdot \underline{\hat{\mathbf{X}}} = \underline{\hat{\mathbf{X}}}_\mathbf{0} \leftarrow$ [Initial conditions, eq.(4.12)]
    $||\underline{\mathbf{u}}_j^k||_2 \leq T_{max}/m_{s/c} \leftarrow$ [Thrust module constraint, eq.(4.16)]
    $\bar{\mathbf{D}} \cdot (\hat{\mathbf{A}}_{CA} \hat{\mathbf{T}} \underline{\hat{\mathbf{X}}}_\mathbf{i/j}) \geq \underline{\mathbf{d}}_{CA} \leftarrow$ [Collision avoidance, eq.(4.33)]
    [Thrust angles constraints, eq.(4.18)]
    [Slew rate constraints, eq.(4.19), eq.(4.20)]
```

4.3. Considerations on the GNC architecture

The proposed guidance algorithm can be implemented onboard the satellites in the formation choosing between three different architectures [35, 36]:

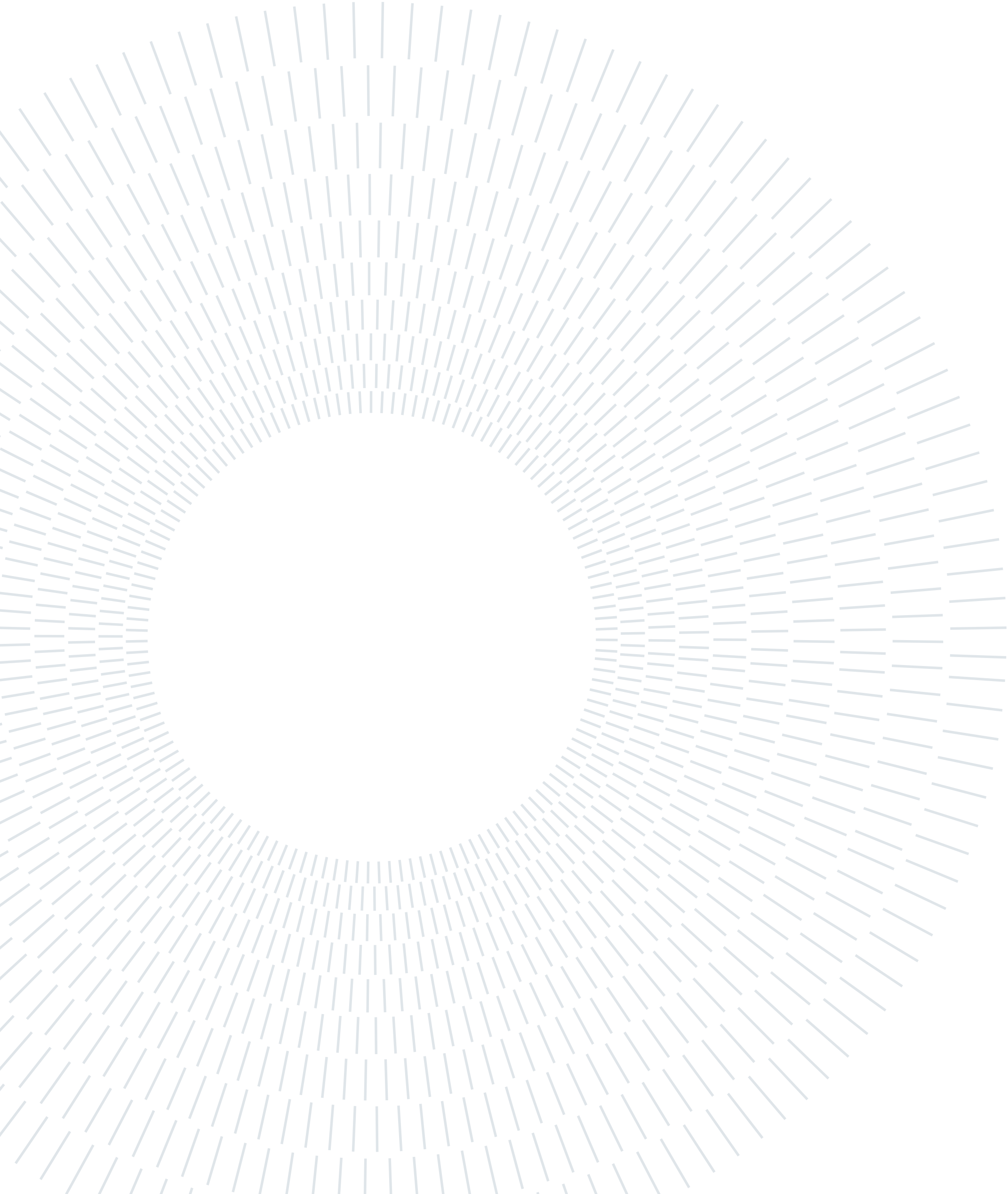
- **Centralized**: in a centralized architecture, a master spacecraft receives the observed state of all the other satellites in the formation, then it computes the guidance and control for all units and sends back commands to each spacecraft [37].
- **Decentralized**: in decentralized architectures, identical algorithms are implemented on each satellite, which then is capable of computing its own action based solely on on-board information, relying only on its observed state [38].
- **Distributed**: in distributed systems, each satellite computes its guidance based on its own information and at least one another observed state from a different agent of the formation, relying on the presence of inter-satellite links [17, 35].



(a) Centralized (b) Decentralized (c) Distributed

Figure 4.6: Schematic representation of centralized, decentralized, and distributed formation flying GNC architectures.

In presence of a collision avoidance constraint like the one introduced in the designed guidance strategy, a decentralized architecture must be discarded a priori, as the satellites are incapable of exchanging information between each other. In fact, the observed states of all satellites, or at least the state of the ones having the highest collision risk, is needed in the computation of the guidance. The remaining two alternatives, namely, centralized and distributed architectures, would be both available options to implement the proposed guidance and control algorithm, however, a distributed system is considered preferable due to its intrinsic higher reliability. Indeed, the presence of a mother satellite dealing with all the tasks introduces a single failure point in the system, as a malfunction would compromise the mission of all agents in the formation. A distributed system completely removes this problem, as a failure of a spacecraft would not directly affect other units. In addition, in case of a guidance software crash, the other satellites would be able to send the control command via the inter-satellite link, as each spacecraft computes the trajectory and control of the entire formation. In case of a missing MPC update, the previously computed solution can be followed in open-loop until a new command is available.



5 | Validation

5.1. Fuel efficiency validation

The optimality of the designed control solution must be verified by comparing it with other options available in literature. A close match to the proposed strategy is found in Reference [22], in which Catanoso et al. develop a Lyapunov-based Model Predictive Controller acting on Relative Orbital Elements. This controller, however, exploits a nonconvex nonlinear propagation of the dynamics based on Gauss Variational Equations (GVE), and no constraint regarding actuation or thrusting angles is present, except a maximum prescribed value for the control acceleration along each axis in the RTN frame. Furthermore, no collision avoidance constraint is imposed. With respect to this reference, fuel efficiency is assessed performing a comparison between the two MPC solutions of the same transfer and with the closed-form fuel-optimal impulsive solution developed by Chernick et al. in Reference [15]. Moreover, a solution obtained with a direct single shooting method is added to the comparison to have an optimal low-thrust benchmark.

The starting keplerian parameters of the orbit that was selected in the chosen reference are reported in Table 5.1, starting and target relative states are reported in Table 5.2. The simulated manoeuvre is an out-of plane transfer taking place in an inclined circular LEO.

Table 5.1: Orbital elements of the orbit used for fuel efficiency validation [22].

a [km]	e [-]	i [rad]	Ω [°]	ω [°]	M_0 [°]
6828	0.00001	1.361	0	0	0

Table 5.2: Starting and target relative states used for fuel efficiency validation [22].

	$a\delta a$ [m]	$a\delta\lambda$ [m]	$a\delta e_x$ [m]	$a\delta e_y$ [m]	$a\delta i_x$ [m]	$a\delta i_y$ [m]
$\mathbf{ROE_0}$	0	0	273	0	10	70
$\mathbf{ROE_T}$	0	0	273	0	400	120

No information about the simulated spacecrafts is given, except for the maximum acceleration that can be provided along each axis, which amounts to 3.2 $\cdot 10^{-5}$ m/s^2. In any case, in the provided solution no tangential thrust is present, therefore the validation will be focused on the out-of-plane control, with both satellites propagated including drag effects and left decaying together during the manoeuvre. The spacecrafts parameters used in the simulation are listed in Table 5.3.

Table 5.3: Properties of the propagated satellites to validate the MPC.

	Chief	Deputy
Mass [kg]	20	20
B [m^2/kg]	0.0105	0.0105
C_R [-]	1	1
Max thrust [mN]	-	0.64

The weights of the MPC cost function have been tuned in order to match the transfer time of the reference solution, corresponding to seven orbit periods. The obtained trajectory in the RTN reference frame is shown in Figure 5.1.

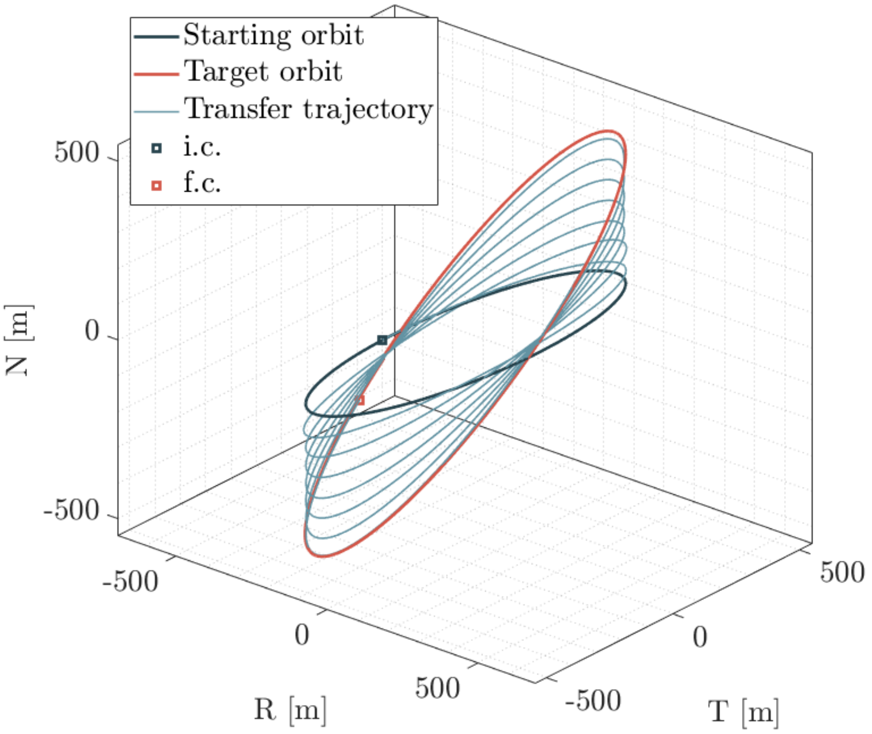


Figure 5.1: Transfer trajectory in the RTN reference frame for the out-of-plane transfer used for controller validation.

The resulting control profile in the normal direction is shown in Figure 5.2. In the same plot is reported the control profile for the same manoeuvre obtained by Catanoso et al. in Reference [22], extracted from the paper with a plot digitizer [39]. Due to the inaccuracies of plot digitization, this is used only to speculate on its general trend and qualitative behaviour. On the other hand, numerical data is taken directly from the document. Optimal out-of-plane manoeuvring locations according to the closed-form impulsive solution are also reported in the plot as dotted lines.

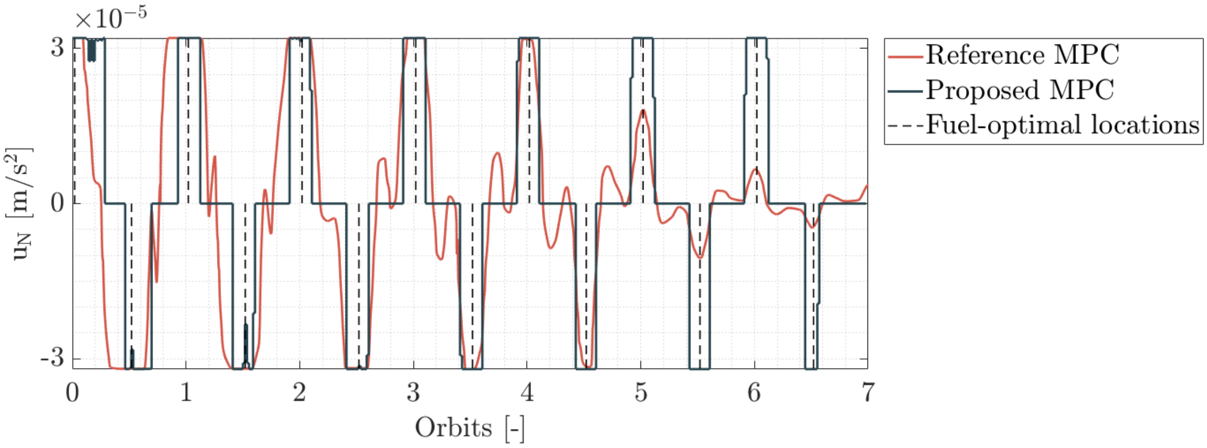


Figure 5.2: Control acceleration in normal direction for the out-of-plane transfer used for controller validation. In dark blue is reported the solution obtained with the convex MPC proposed in the document. In red is reported the profile shown in Reference [22].

From the figure it can be observed how the reference control profile puts more effort at the beginning , gradually reducing it approaching the target. On the other hand, the MPC proposed in this document distributes the thrusting arcs more evenly across the manoeuvring interval and is capable of centering them in the fuel-optimal locations. This is reflected in the ΔV required to complete the transfer, reported in Table 5.4 for the three comparison cases.

Table 5.4: ΔV comparison between the proposed MPC and the two considered literature solutions.

Method	ΔV [m/s]
Reference MPC [22]	0.5554
Proposed MPC	0.4915
Closed-form impulsive [15]	0.4373

The proposed Model Predictive Controller completes the transfer with a ΔV which is in line with the two other analyzed methods. In particular, it is able to get closer to the impulsive solution with respect to the nonlinear version for equal transfer times. The distance between the proposed controller and the impulsive solution can be decreased even further by decreasing the tracking error weights in the MPC cost function, allowing to perform the transfer in more time. In doing so, the MPC computes narrower thrusting arcs, more centered around optimal thrusting locations. An intrinsic limitation is the sampling time of the Model Predictive Controller, as this represents the minimum possible length of a thrusting arc.

5.1.1. Direct single shooting solution

To the previous comparison, a solution obtained with a direct single shooting (DSS) algorithm is provided as an optimal low-trust benchmark for the same transfer. By including the same low-thrust constraint, this solution is a better reference of what can be achieved with respect to the impulsive closed-form solution, which requires high-thrust extremely narrow impulses. Direct optimal control methods, single shooting in particular, do not provide the absolute optimal solution, but are able to provide a close-to-optimal result while allowing an easy introduction of constraints [40].

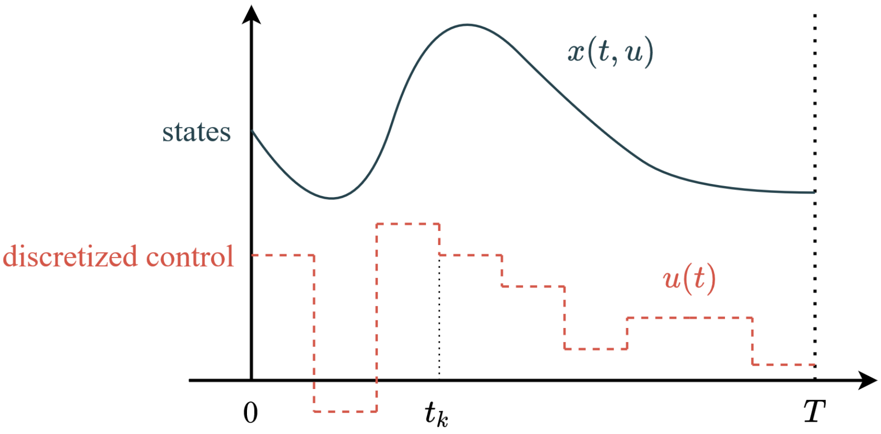


Figure 5.3: Domain discretization in direct single shooting trajectory optimization.

Direct single shooting optimization consists in discretizing control on a fixed grid treating the state as a dependent variable, as depicted in Figure 5.3. The dynamics is imposed as a nonlinear constraint by propagating with an ODE the initial conditions with the discretized control over the desired time interval and imposing a desired target state. The selected optimization solver will minimize a desired cost functional, the transfer ΔV in this case, while driving the nonlinear constraint to zero. For this optimization, the state is expressed in cartesian coordinates in the earth-centered inertial frame and the dynamics is propagated with an high fidelity propagator including all orbital disturbances. Final target conditions are imposed on the Relative Orbital Elements at the end of the propagation, calculated with the final states of the chief and deputy spacecrafts. The resulting control profile for the out-of-plane transfer of interest is represented in Figure 5.4, together with the one computed by the convex Model Predictive Controller.

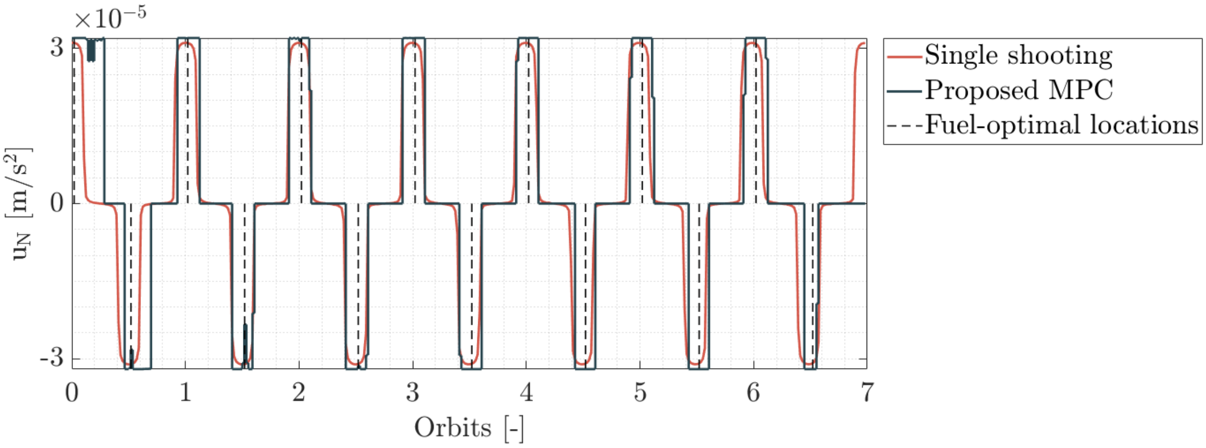


Figure 5.4: Control profile obtained with direct single shooting for the validation out-of-plane transfer compared to the results obtained with the proposed convex MPC.

The graph shows that the single shooting algorithm is capable of distributing the manoeuvres around fuel-optimal locations in an even cleaner way with respect to the proposed convex controller, requiring as expected a lower ΔV than the autonomous counterpart, as reported in Table 5.5. However, this optimization is far more computationally intensive than the convexified one used in the MPC, thus, it is not viable for on-board implementation. Moreover, the trajectory is integrated all at once, therefore, unlike the proposed autonomous solution, it is not robust to changes in external conditions as a sudden increase of atmospheric density due to a raise of solar activity.

Table 5.5: ΔV comparison between the proposed MPC, the two considered literature solutions, and the solution obtained with the direct single shooting algorithm.

Method	ΔV [m/s]
Reference MPC [22]	0.5554
Proposed MPC	0.4915
Direct single shooting	0.4680
Closed-form impulsive [15]	0.4373

5.1.2. Direct transcription

Also a direct transcription algorithm adopting an Hermite-Simpson collocation scheme was implemented to obtain a better solution still using a direct optimization technique. Indeed, direct transcription is in general capable of providing more optimal solutions with respect to shooting techniques. The working principle of this method consists in transcribing the dynamics of the system into a finite set of equality constraints. The solver will then minimize the cost functional while driving these constraints to zero, with an accuracy which depends on the adopted collocation scheme [41, 42].

Adopting an Hermite-Simpson numerical scheme, the algorithm converged to solutions which resulted to be incorrect when integrated with an orbital propagator. In other words, the accuracy on the absolute orbital dynamics transcription was not good enough to represent the dynamics at the formation scale, in the order of hundreds of meters. Two improvements to try to solve this problem could be to increase the number of discretization points along the orbit or to use a higher-order collocation scheme, e.g. Gauss-Lobatto.

5.2. Validation of drag model effectiveness

Due to the feedback nature of Model Predictive Control, it can be argued that the inclusion of a drag plant matrix is not mandatory, as the relative drift would be corrected anyway by the MPC prescribing a tangential manoeuvre when an undesired along-track separation is created. Moreover, as drag is a nonconservative force acting always in the same direction, there is no way of exploiting it to reduce transfers ΔV unless a positive shift in along-track direction is desired. Nevertheless, the inclusion of a drag plant matrix for an accurate propagation of the dynamics is reflected on the tracking precision of the desired spacecraft separation. To validate the effectiveness of the drag model proposed in Chapter 2 on the control accuracy, the tracking of a non-decaying holding point placed in the chief spacecraft is simulated. This equals to controlling the deputy spacecraft in its absolute dynamics to compensate the effects of drag and to maintain its along-track position. The geometry of the simulated orbit and the tracked Relative Orbital Elements are reported in Tables 5.6 and 5.7 respectively.

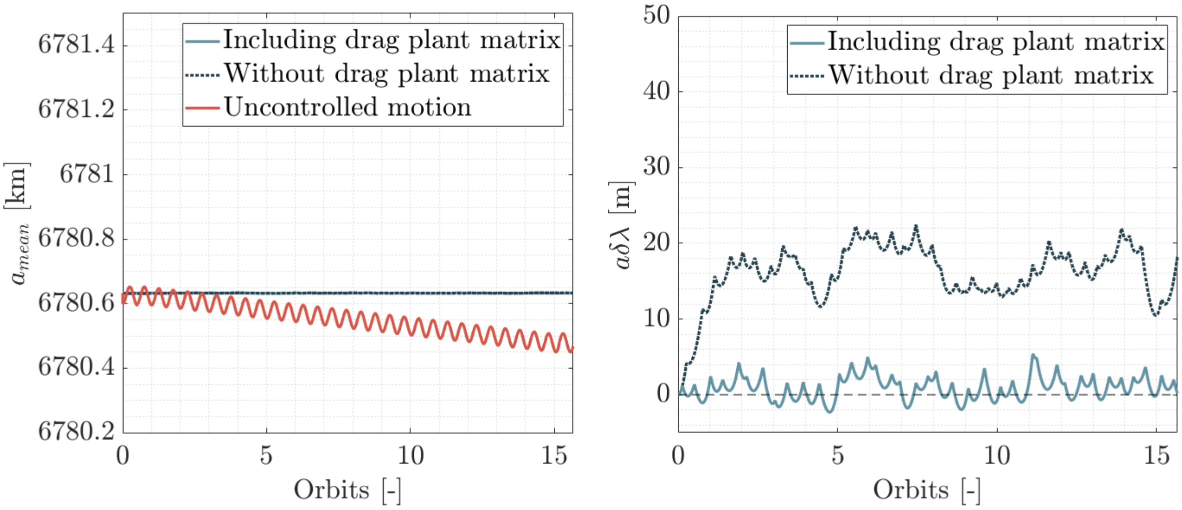
Table 5.6: Starting osculating orbital elements of the orbit used for the validation of drag plant matrix effectiveness.

a [km]	e [-]	i [°]	Ω [°]	ω [°]	M_0 [°]
6771	0.001	97.05	30	90	0

Table 5.7: Relative Orbital Elements to track in the drag plant matrix effectiveness validation.

$a\delta a$ [m]	$a\delta\lambda$ [m]	$a\delta e_x$ [m]	$a\delta e_y$ [m]	$a\delta i_x$ [m]	$a\delta i_y$ [m]
0	0	0	0	0	0

By simulating with and without including the drag plant matrix in the MPC dynamics propagation, the results can be compared and the effectiveness of the drag model can be assessed. In particular the variations in time of mean semi-major axis and along-track separation $\delta\lambda$ are represented in Figure 5.5. The semi-major axis is averaged over one orbital period to remove the periodic oscillations due to J2 effect.



(a) Mean semi-major axis in time

(b) $\delta\lambda$ evolution in time

Figure 5.5: Mean semi-major axis and relative mean longitude evolution in time for one day of station keeping. For the semi-major axis, the average is computed over one orbital period.

From the figure it is evident that the inclusion of the plant matrix improves the accuracy of the solution in terms of tracking of the desired along-track separation. Indeed in both cases a constant mean altitude is maintained, but, when drag is not considered in the propagation, an error of about 20 m in $\delta\lambda$ is accumulated throughout the day. This error is never corrected even when the simulation is left running for several more orbit revolutions. In addition, the computational cost added by the drag plant matrix inclusion is completely negligible.

The results of the two simulations are again compared in terms of accuracy and fuel optimality to the optimal solution provided a direct single shooting algorithm. The comparison between the three obtained results is reported in Table 5.8.

Table 5.8: Simulation results for drag plant matrix effectiveness validation.

	Without $\mathbf{A_{drag}}$	**Including $\mathbf{A_{drag}}$**	**DSS algorithm**
ΔV_T per year	$\sim$ 33 m/s	$\sim$ 33 m/s	$\sim$ 33 m/s
$a\delta\lambda_{mean}$	15.97 m	0.89 m	-0.02 m
$\text{STD}_{a\delta\lambda}$	3.36 m	3.02 m	0.013 m

By looking at the results it can be observed again that the inclusion of the plant matrix provides an increase of accuracy, but both solutions are fuel-optimal when compared to the direct optimization output. This means that both are capable of compensating the nonconservative effect of drag without providing any unrequired control actions. The direct single shooting method, in particular, provides a very low tracking error, directly compensating the effect of drag by supplying an equal acceleration in opposite direction. This is known as drag-free control and has been used in missions where the accuracy of drag compensation needed to be extremely high, as in the gravimetric study mission GOCE [43]. However, the implementation of this strategy must be justified by the need of extremely high precision in the acquired data, as it introduces multiple stringent requirements on the spacecraft design.

6 Simulation results

6.1. MPC settings

The parameters of the Model Predictive Controller have been chosen based on general considerations and simulation results. A prediction horizon of one orbit is selected due to the natural periodic nature of orbital evolution. This proved to be a good value to predict and exploit the natural evolution of the dynamics. The sampling time of the MPC is set as 100 s, a good compromise between MPC accuracy and computational efficiency. Finally, the control horizon is set as seven sampling steps. This means that the on-board computer shall solve the optimal control problem about eight equidistant times for a LEO orbit at 400 km altitude. A higher frequency of computation improves accuracy, but this improvement was found to be negligible for the simulated scenarios.

Table 6.1: Parameters of the Model Predictive Controller used in the simulations. These parameters will be used in all scenarios unless explicitly specified.

Parameter	Value
Prediction Horizon	5600 s, ∼1 LEO orbit
Sampling time, T_s	100 s
Control Horizon	$7 \cdot T_s$, ∼1/8 LEO orbit

6.2. Simulation scenarios

To check the robustness and flexibility of the proposed controller, several different operational scenarios are simulated for both formation reconfiguration and maintenance. In particular, the algorithm is tested in representative or challenging but realistic conditions. For all simulations the properties of the chaser spacecrafts are the ones reported in Table 6.2. According to the premises made in the introduction and along the document, the selected spacecrafts are micro-satellites actuated by a single low-thrust engine with very low authority and a minimum thrust limit.

Table 6.2: Properties of chaser satellites used in all simulations.

Parameter	Value
Mass	20 kg
Drag area	0.1 m^2
C_D	2.1
SRP area	0.1 m^2
C_R	1
Max thrust	0.65 mN
Min thrust	0.35 mN

Also the orbit geometry is shared by all simulations. The chosen reference orbit is a very low altitude Sun-synchronous orbit (SSO), in order to test the Model Predictive Controller performance in a high-drag environment which also represents a realistic choice for a scientific or earth observation mission in Low Earth Orbit. The selected orbital parameters are reported in Table 6.3. All simulations start on 21st March 2021 at midnight.

Table 6.3: Starting osculating orbital elements of the reference orbit used in the simulations.

a [km]	e [-]	i [°]	Ω [°]	ω [°]	M_0 [°]
6771	0.001	97.004	30	90	0

6.2.1. Out-of-plane transfer

The first simulated scenario is an out-of-plane relative orbit transfer, which corresponds to a change of relative inclination. The starting and target conditions for the Relative Orbital Elements are reported in Table 6.4, in both cases eccentricity/inclination vector separation is guaranteed. This concept, originally developed for GEO satellites, is used to impose a passive collision avoidance constraint with a parallel (or anti-parallel) alignment of the relative eccentricity and inclination vectors. This is equivalent to impose that, when the spacecraft crosses the target orbital plane, the radial distance is larger than $\min(a\delta e, a\delta i)$ even in the case of a vanishing along-track separation [44]. Constraints regarding thrusting angles are imposed for this simulation by enforcing a null radial acceleration and positive tangential control.

Table 6.4: Starting and target Relative Orbital Elements for the out-of-plane transfer.

	$a\delta a$ [m]	$a\delta\lambda$ [m]	$a\delta e_x$ [m]	$a\delta e_y$ [m]	$a\delta i_x$ [m]	$a\delta i_y$ [m]
$\mathbf{ROE_0}$	0	0	0	200	0	180
$\mathbf{ROE_T}$	0	0	0	200	0	420

The simulation is stopped when all Relative Orbital Elements reach the target with a tolerance of three meters on their value multiplied by the reference semi-major axis. The resulting trajectory in the RTN reference frame is shown in Figure 6.1, whereas relative inclination evolution in time and ROE space is shown in Figure 6.2. The spacecraft completes the transfer with the desired accuracy in about four and a half orbit periods.

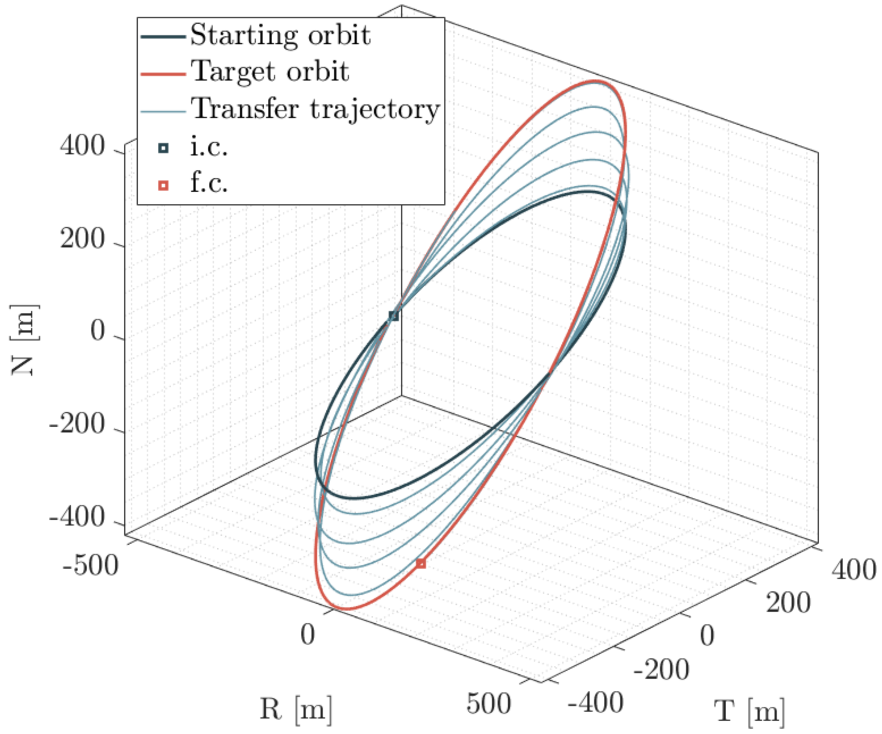


Figure 6.1: Out-of-plane transfer trajectory in the RTN reference frame.

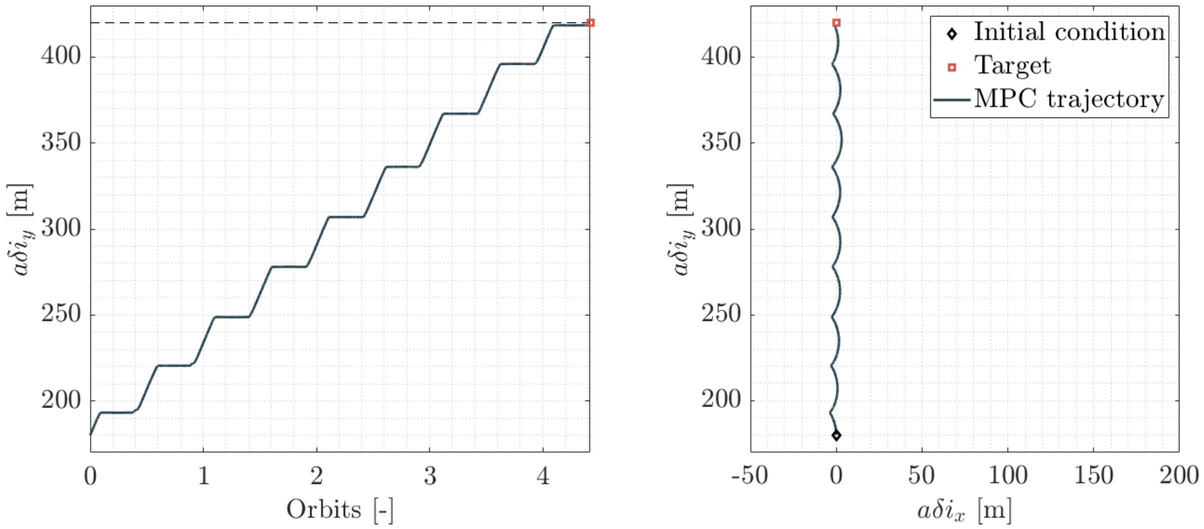


Figure 6.2: Relative inclination evolution in time and ROE space for the out-of-plane transfer.

In Figure 6.3 are shown the control actions along the three RTN directions. Naturally, u_z is dominant for an out-of-plane transfer, however, also tangential burns are provided sparsely in order to compensate the effect of differential drag, controlling $\delta\lambda$ and orbit semi-major axis throughout the transfer. The optimal thrusting locations for an out-of-plane transfer according to the closed-form impulsive solution in Reference [15], already reported in Chapter 2, are shown in the figure as dotted lines. It is possible to observe that the MPC is able to center the low-thrust manoeuvres around those locations, thrusting for more time due to the saturation of the available control effort.

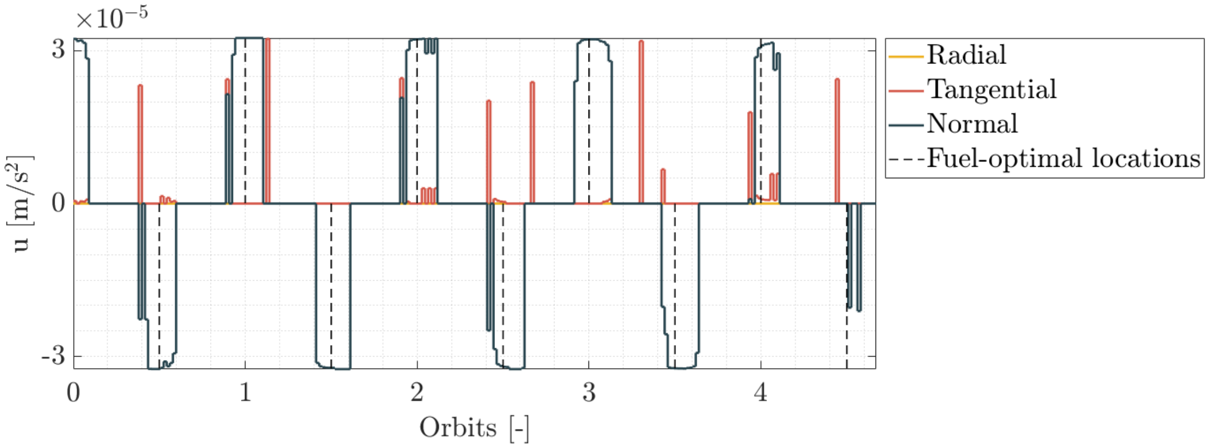


Figure 6.3: Out-of-plane transfer control accelerations in the RTN frame.

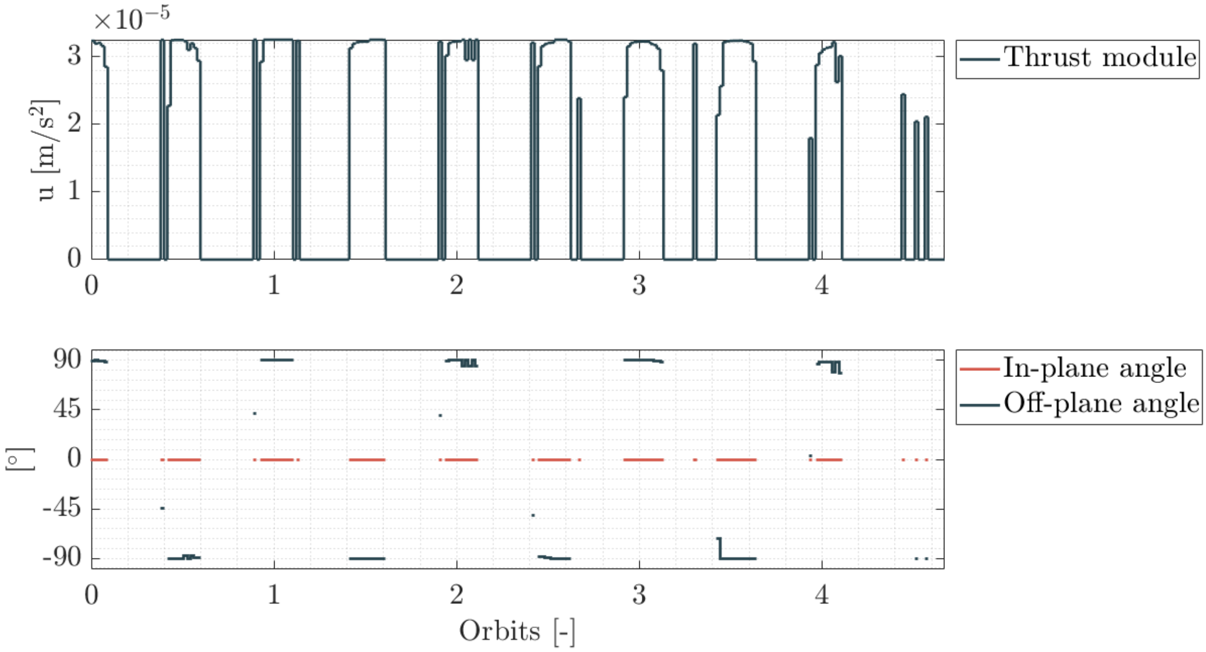


Figure 6.4: Out-of-plane transfer thrust profile and thrusting angles.

This saturation is clear also in Figure 6.4, in which the control profile is visualized in terms of module and angles. In the graphs is possible to observe that that the constraints on thrusting angles are satisfied and that the slew constraints separate the in-plane and off-plane correction manoeuvres, relieving the burden of attitude control. The total ΔV of the transfer amounts to 0.3052 m/s, most of which are supplied in normal direction to complete the reconfiguration, while the rest is given in positive along-track direction. Depending on the mission constraints, this number can be reduced by allowing the manoeuvre to be completed in more time. In this way, the MPC would be able to center and concentrate even more the thrusting arcs around optimal locations, getting closer and closer to the closed-form impulsive solution.

6.2.2. Relative eccentricity change

The second selected scenario is a change of relative eccentricity. The starting and target relative states for this reconfiguration, which may represent a close-up manoeuvre to the chief spacecraft, are reported in Table 6.5. Also in this case eccentricity/inclination vector separation is guaranteed and radial thrust is imposed to be null. On the other hand, tangential thrust is let free to assume both negative and positive values, to be able to perform the transfer in a quicker time.

Table 6.5: Starting and target Relative Orbital Elements for the in-plane transfer.

	$a\delta a$ [m]	$a\delta\lambda$ [m]	$a\delta e_x$ [m]	$a\delta e_y$ [m]	$a\delta i_x$ [m]	$a\delta i_y$ [m]
$\mathbf{ROE_0}$	0	0	0	400	0	100
$\mathbf{ROE_T}$	0	0	0	200	0	100

Also in this case the simulation is stopped when all Relative Orbital Elements reach the target with the same tolerance of three meters on their value multiplied by the reference semi-major axis. The resulting trajectory in the RTN reference frame is shown in Figure 6.5, whereas relative eccentricity evolution in time and ROE space is shown in Figure 6.6. The spacecraft completes the transfer with the desired accuracy in slightly less than three orbit periods.

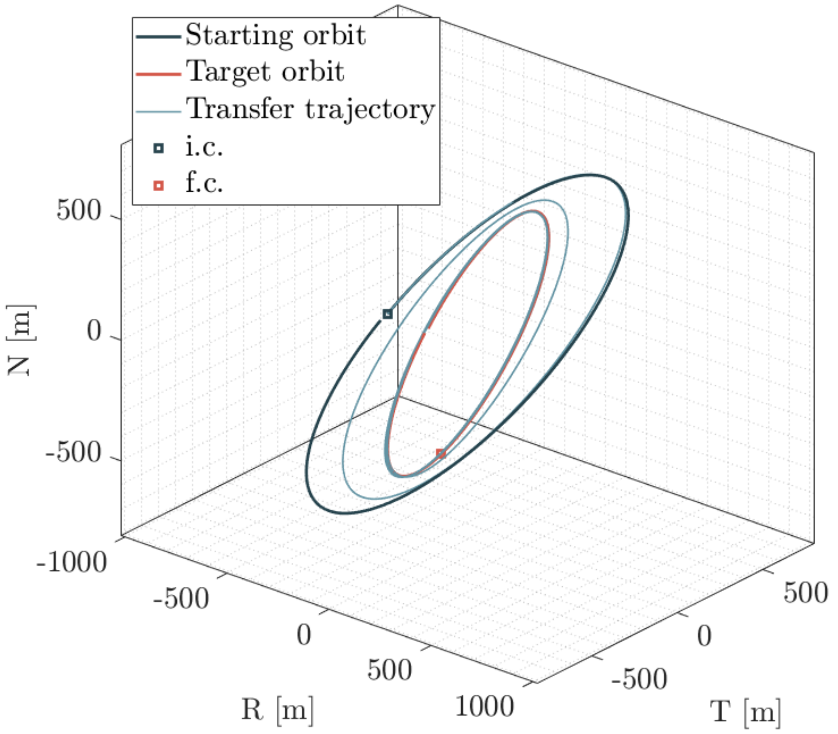


Figure 6.5: Transfer trajectory in the RTN reference frame for the relative eccentricity change.

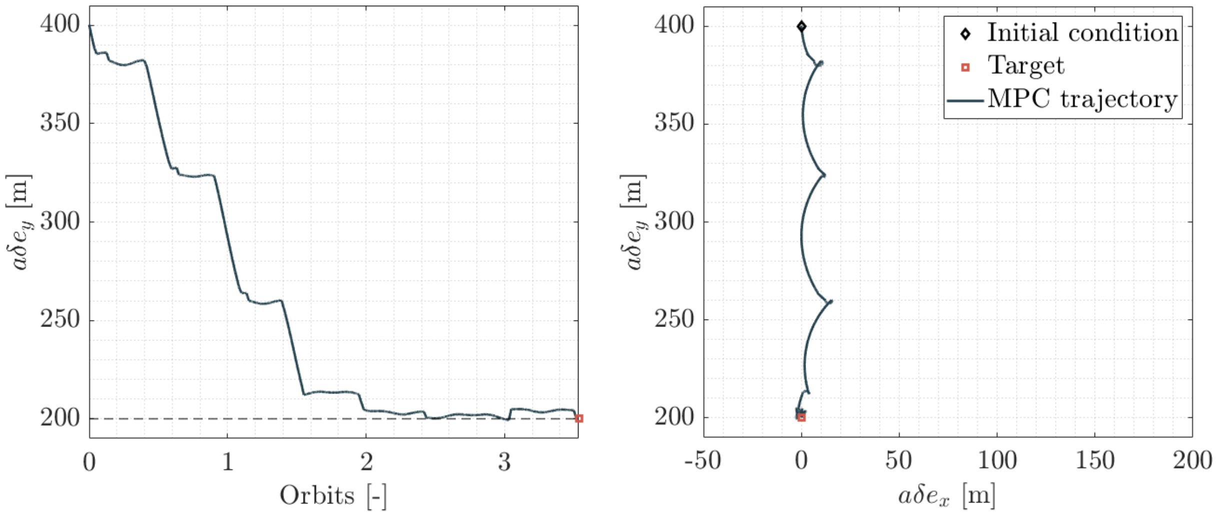


Figure 6.6: Relative eccentricity evolution in time and ROE space.

The control profile computed by the MPC throughout the manoeuvre is shown in Figure 6.7. Due to the absence of changes in out-of-plane ROEs, tangential thrust is sufficient to complete the transfer. The same control profile is represented in terms of thrust module and thrusting angles in Figure 6.8.

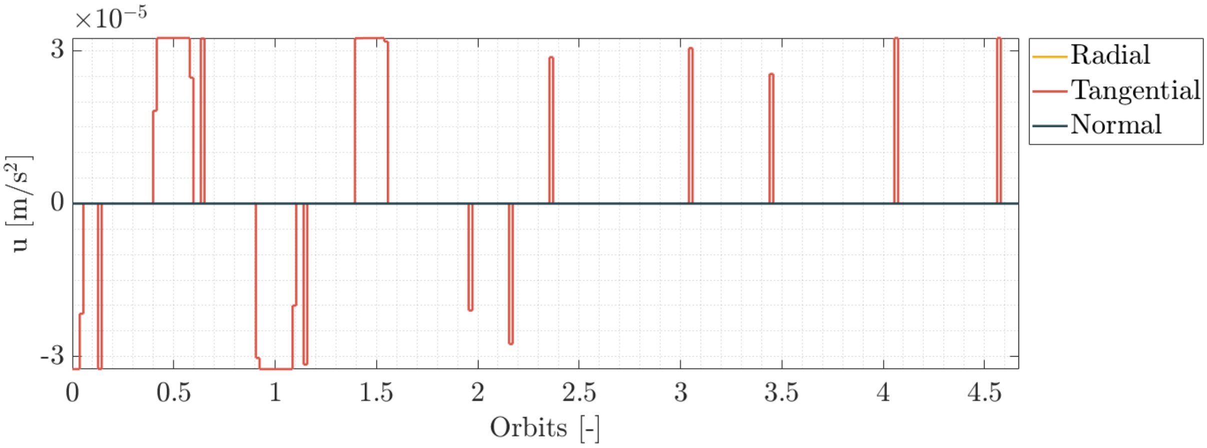


Figure 6.7: Control accelerations in the RTN frame for the relative eccentricity change.

From the two plots it is possible to notice the saturation of the control effort in the first phases of the transfer. As in the out-of-plane case, this saturation can be alleviated by reducing the weights of tracking error terms in the MPC cost function in Equation (4.15), allowing the transfer to be completed in more time. For what concerns attitude control, the spacecraft has enough time to perform all the slewing manoeuvres prescribed by the MPC.

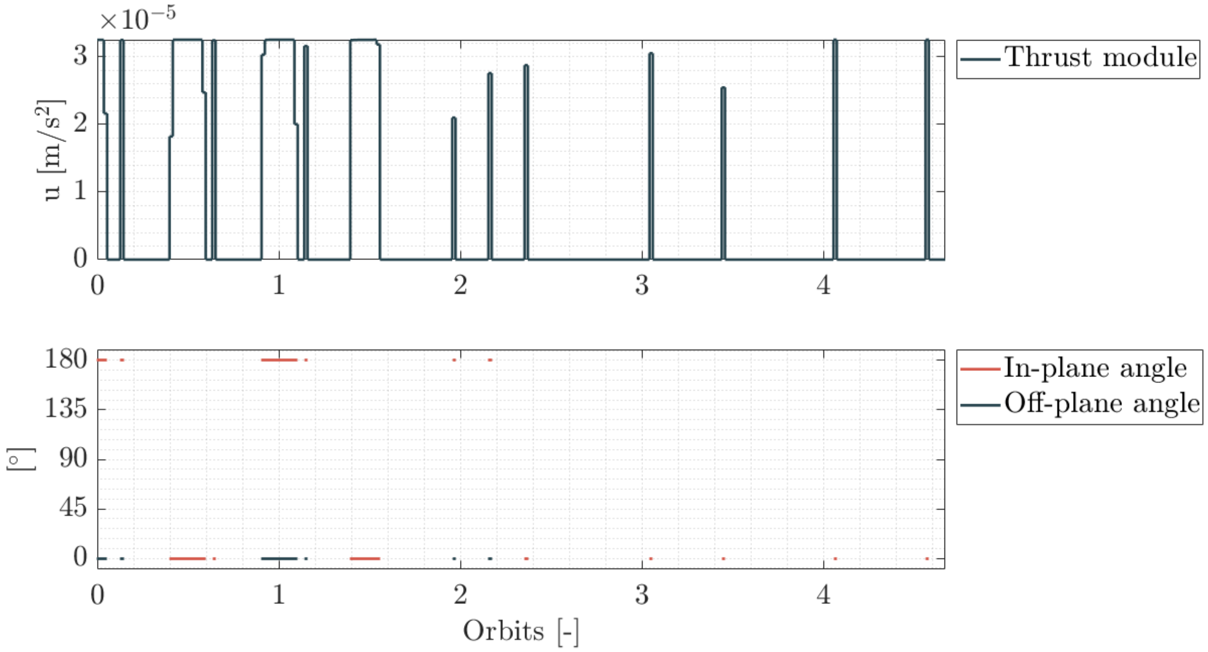


Figure 6.8: Thrust profile and thrusting angles for the relative eccentricity change.

For this transfer the required ΔV amounts to 0.1340 m/s, all to be provided in tangential direction.

6.2.3. Position swap

To test the effectiveness and efficiency of the collision avoidance algorithm, a setting is created in which two spacecrafts separated along-track must invert their position while maintaining a minimum inter-satellite distance of 300 m. The starting and target relative states of the two spacecrafts with respect to the virtual point placed in the centroid of the formation are listed in Table 6.6. Also in this case radial thrust is set to zero, but no further constraint on thrusting angles is added.

Table 6.6: Starting and target Relative Orbital Elements of the two spacecrafts for the position swap scenario.

Spacecraft A	$a\delta a$ [m]	$a\delta\lambda$ [m]	$a\delta e_x$ [m]	$a\delta e_y$ [m]	$a\delta i_x$ [m]	$a\delta i_y$ [m]
$\mathbf{ROE_0}$	0	-200	0	0	0	0
$\mathbf{ROE_T}$	0	200	0	0	0	0
Spacecraft B	$a\delta a$ [m]	$a\delta\lambda$ [m]	$a\delta e_x$ [m]	$a\delta e_y$ [m]	$a\delta i_x$ [m]	$a\delta i_y$ [m]
$\mathbf{ROE_0}$	0	200	0	0	0	0
$\mathbf{ROE_T}$	0	-200	0	0	0	0

The resulting trajectories in the RTN frame are shown in Figure 6.9, together with a solution of the same reconfiguration obtained without including the collision avoidance constraint. From the figure it can be seen how the spacecrafts take a longer path to reach the target in order to keep a minimum relative distance from one to another.

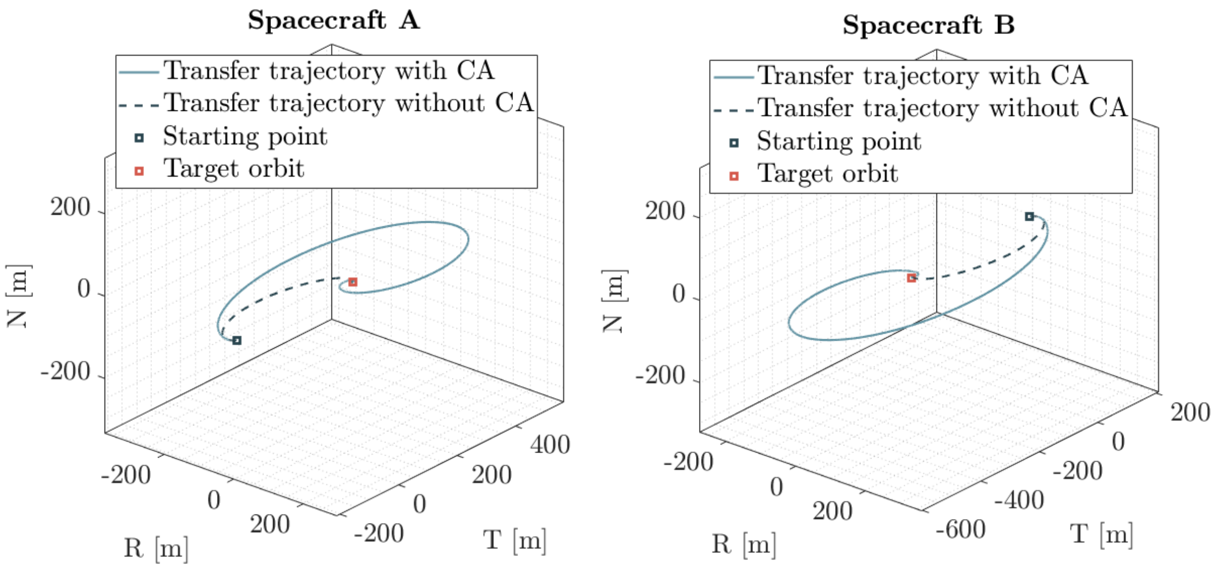


Figure 6.9: Trajectories in the RTN frame of the two spacecrafts during the position swap.

This is even more evident in Figure 6.10, in which the inter-satellite distance between the two spacecrafts is plotted in its time evolution. With respect to the simulation without collision avoidance, when the constraint is included the transfer takes a longer time to be concluded, but the safety threshold is always respected.

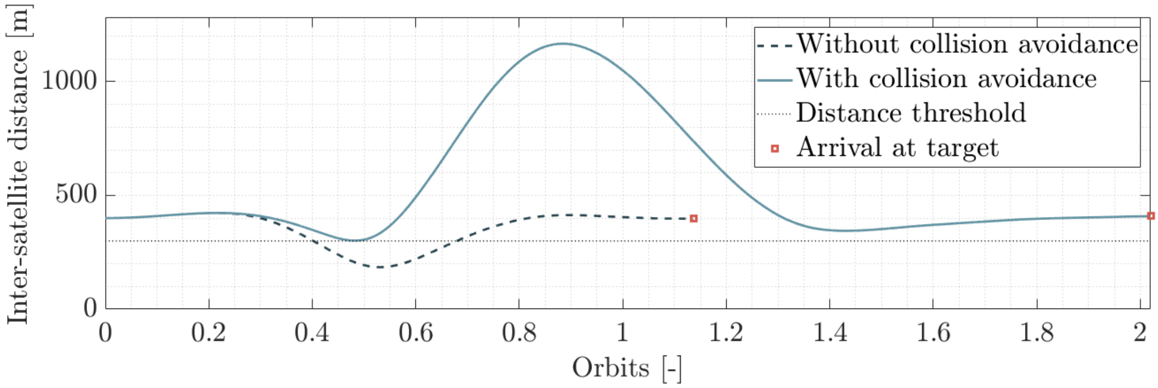


Figure 6.10: Inter-satellite distance evolution between the two spacecrafts during the position swap.

Naturally, the resulting transfer ΔV increases when the collision avoidance constraint is included. The values are reported in Table 6.7. A ΔV increase of about 0.15 m/s is found for both spacecrafts when the safety constraint is included.

Table 6.7: ΔV comparison for the position swap with and without the inclusion of the collision avoidance constraint.

	Without CA	With CA
Spacecraft A	0.0485 m/s	0.2257 m/s
Spacecraft B	0.0456 m/s	0.2071 m/s

It is interesting to notice the fuel balancing that naturally results by controlling the spacecrafts with respect to a point placed in the centroid of the formation.

6.2.4. Tetrahedron formation acquisition and maintenance

In previous sections, the Model Predictive Controller is tested in specific scenarios created to isolate some particular response. In the next simulation a more complex manoeuvre is simulated, in which a formation of four spacecrafts needs to acquire and maintain a tetrahedron formation geometry starting from an along-track in-line arrangement. Spacecrafts flying in tetrahedron formations are excellent instrument platforms for electromagnetic and plasma studies, as a minimum of four spacecrafts establishing a volume is required to study a planetary magnetic field [45]. To obtain the desired formation shape, the target relative states in Relative Orbital Elements have been identified and reported in Table 6.8, together with the ones of the starting in-line disposition. A null radial thrust is imposed and the previously introduced constraints on slew rates apply also in this case. In addition, a minimum inter-satellite distance of 50 m is enforced as the threshold for the collision avoidance algorithm.

Table 6.8: Starting and target Relative Orbital Elements of the four spacecrafts in the tetrahedron formation scenario.

	Initial ROE state [m]	Final ROE state [m]
Spacecraft A	$[0, 750, 0, 0, 0, 0]$	$[0, 400, 0, 0, 0, 0]$
Spacecraft B	$[0, 250, 0, 0, 0, 0]$	$[0, 100, 177, 177, 354, 354]$
Spacecraft C	$[0, -250, 0, 0, 0, 0]$	$[0, -100, -177, 177, -354, 354]$
Spacecraft D	$[0, -750, 0, 0, 0, 0]$	$[0, -400, 0, 0, 0, 0]$

By reaching and maintaining the desired geometry, the formation will keep a constant volume and will always assume the same orientation in the same specified points along the orbit. These are two common requirements for a tetrahedron formation taking scientific measurements. Spacecrafts A and D remain in-line but reduce their relative distance, whereas spacecrafts B and C transfer to two bounded orbits slightly separated along-track and with a shifted phase angle of 90°, needed to create a volume. The phase difference is created assigning an opposite value of relative eccentricity and inclination along x direction, but equal in module. This is done to split equally between the two spacecrafts the control needed to counteract the effect of J2 introduced in presence of a δi_x component, which can be seen in the J2 plant matrix in Equation (2.8).
The resulting trajectories of the satellites forming the formation are represented in the RN and TN planes in Figure 6.11, the desired relative positions of all spacecrafts are acquired with a ROE accuracy of three meters in slightly less than seven and a half orbital periods.

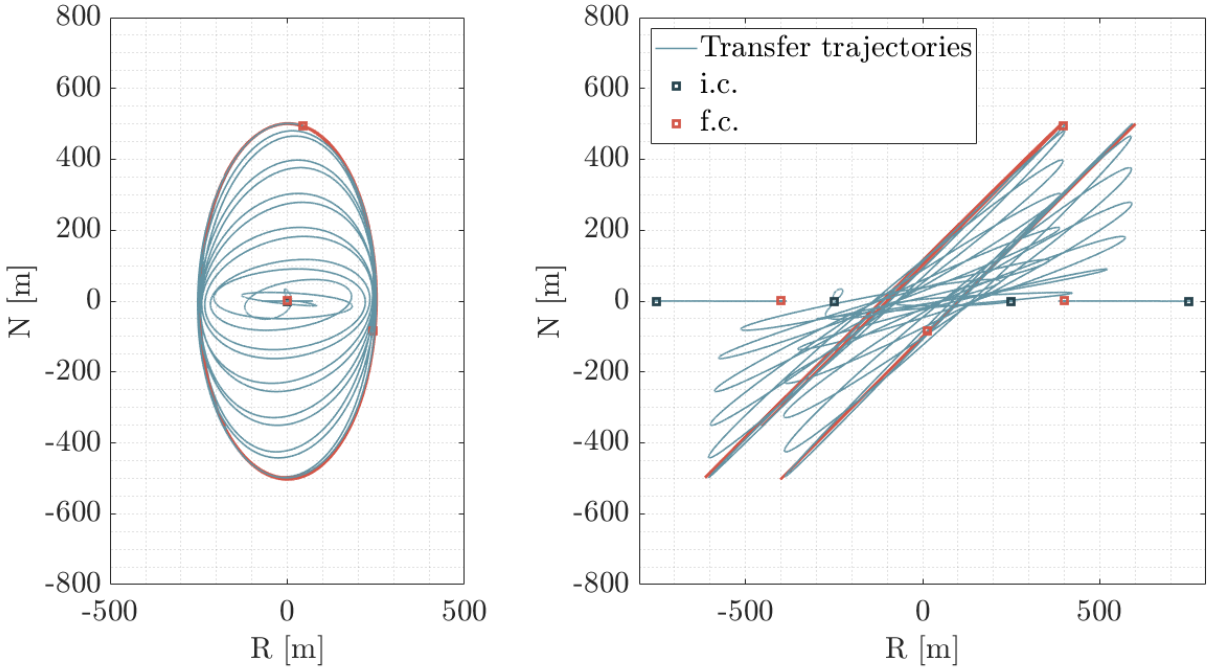


Figure 6.11: Trajectories in the RN and TN planes of the four spacecrafts in the tetrahedron acquisition manoeuvre. In red are represented the final conditions and the target holding orbits (or points) of the four spacecrafts.

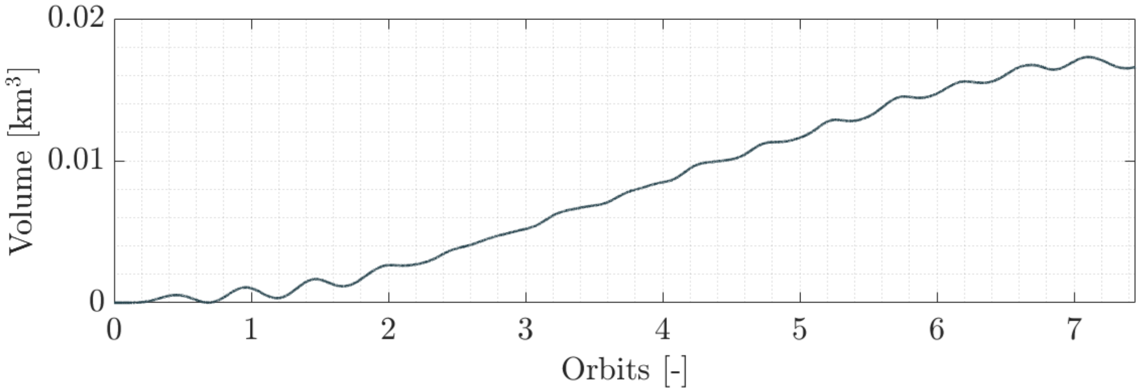


Figure 6.12: Evolution of tetrahedron volume during formation acquisition.

During the formation acquisition phase, the formed tetrahedral shape grows in time, from the zero volume of the in-line arrangement to the reference value, as depicted in Figure 6.12.

The prescribed control profiles of spacecrafts A and B are shown in Figure 6.13, the other two satellites show a very similar behaviour of their counterpart shifted along-track.

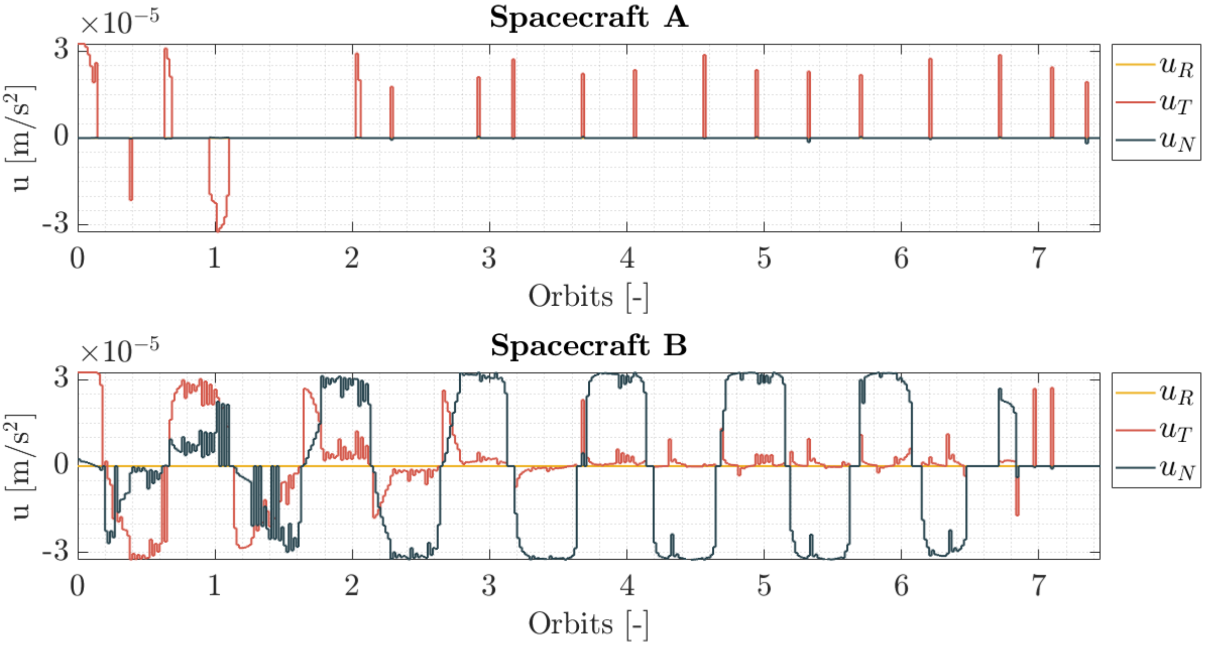


Figure 6.13: Controlling accelerations in the RTN frame of spacecrafts A and B in the tetrahedron acquisition manoeuvre.

Naturally, spacecrafts B and C require a higher control effort with respect to the other two formation components, as they need to acquire both relative eccentricity and relative inclination vectors. The resulting ΔV for the four satellites during the tetrahedron acquisition phase are reported in Table 6.9. The value for spacecrafts B and C can be decreased lowering the gains relative to target tracking, but increasing the transfer time.

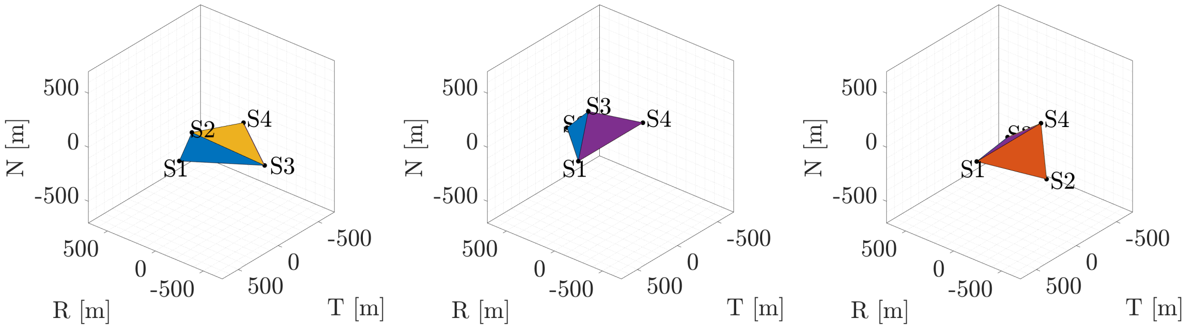
Table 6.9: Required ΔV of the four spacecrafts to acquire tetrahedron geometry.

	ΔV transfer [m/s]
Spacecraft A	0.0878
Spacecraft B	0.9659
Spacecraft C	0.9151
Spacecraft D	0.1187

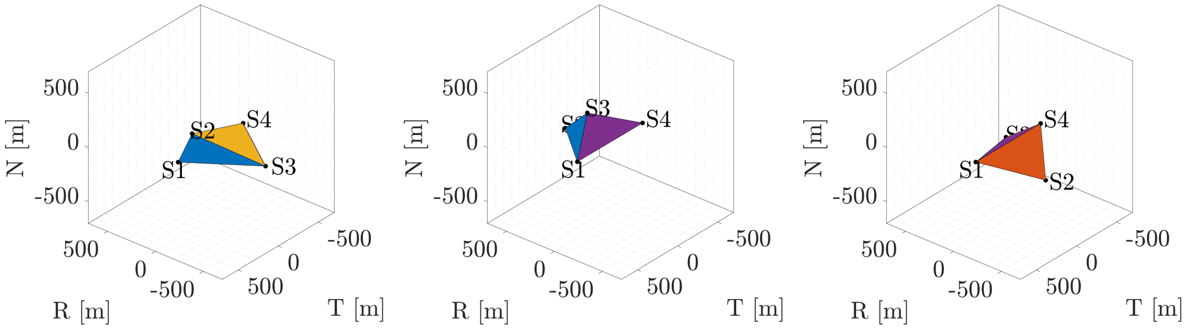
Once the satellites are in the desired position, it is necessary to maintain the relative states in order to meet the volume keeping and geometry repetition requirements. To verify the accuracy and efficiency of formation keeping, a simulation of one day is conducted with the spacecrafts maintaining not only their respective relative state, but also their absolute orbit shape, altitude in particular, being controlled with respect to a non-decaying orbiting slot. During station keeping, constraints on thrusting angles can be added to prescribe a maximum thrust cone, limiting slew manoeuvres during observation. In this case, the in-plane angle is fixed at a value of 0° and the off-plane angle has a maximum variation of 45° around zero. The resulting ΔV is reported in Table 6.10, together with the relative estimate over one year of operations. The values obtained for spacecrafts A and B are consistent with the ones found in the simulation used to validate the effectiveness of the drag plant matrix in Chapter 5. As expected, B and C spacecrafts need a higher control effort to counteract the effect of J2 induced by a δi_x component, but this is equally split among the two.

Table 6.10: Required ΔV to maintain the tetrahedron formation geometry and counteract orbit decay over one day of observations and related estimate over one year.

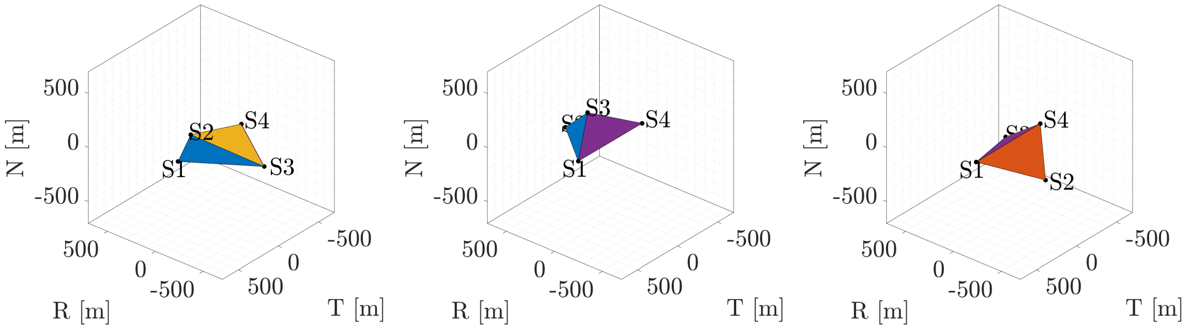
	ΔV one day of SK	ΔV one year of SK
Spacecraft A	0.090 m/s	$\sim$ 33 m/s
Spacecraft B	0.114 m/s	$\sim$ 41 m/s
Spacecraft C	0.112 m/s	$\sim$ 41 m/s
Spacecraft D	0.091 m/s	$\sim$ 33 m/s



(a) $N_{orbit} = 1$, $\theta = 0°$ (b) $N_{orbit} = 1$, $\theta = 120°$ (c) $N_{orbit} = 1$, $\theta = 240°$



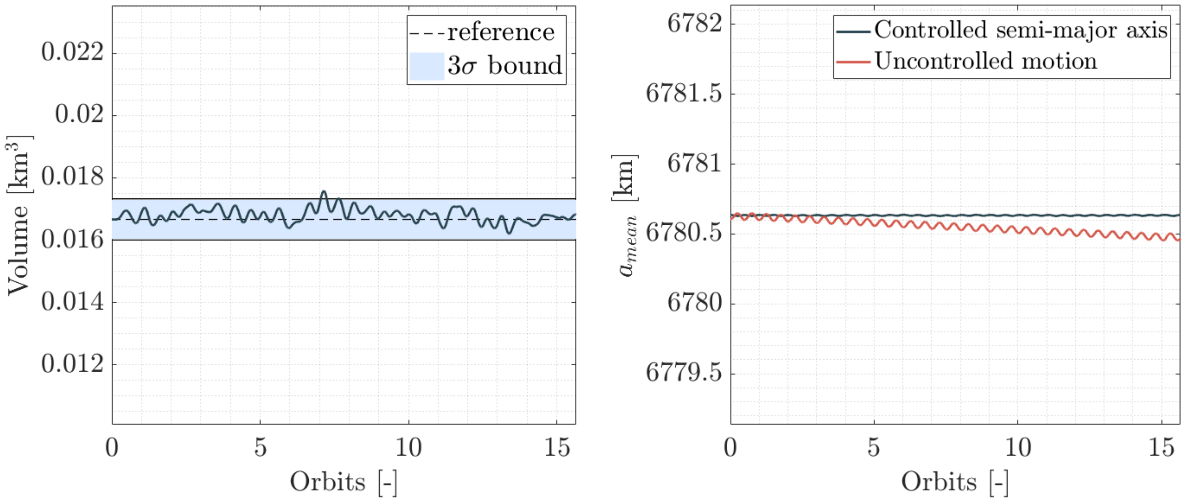
(d) $N_{orbit} = 8$, $\theta = 0°$ (e) $N_{orbit} = 8$, $\theta = 120°$ (f) $N_{orbit} = 8$, $\theta = 240°$



(g) $N_{orbit} = 15$, $\theta = 0°$ (h) $N_{orbit} = 15$, $\theta = 120°$ (i) $N_{orbit} = 15$, $\theta = 240°$

Figure 6.14: Evolution in time of the tetrahedron shape. The geometry repeats itself at regular points along the orbit.

In Figure 6.14 is represented the evolution in time of the tetrahedral shape of the formation. In particular, the geometry is shown in correspondence of particular values of true anomaly to show its periodic repetition. The other two requirements are to keep a constant tetrahedron volume and a constant orbit altitude throughout the observation. The evolution of these two quantities in time is shown in the graph in Figure 6.15.



(a) Tetrahedron volume in time

(b) Mean semi-major axis in time

Figure 6.15: Tetrahedron volume and mean semi-major axis evolution in time for one day of formation keeping. For the semi-major axis, the average is computed over one orbital period.

The volume is maintained around its nominal value, with a 3σ bound of $6.61 \cdot 10^{-4}$ km^3. This number, correspondent to about 4% of the reference, is the statistical value under which the variation from the nominal volume is kept for 99.73% of the time. For what concerns drag compensation, it can be observed that the mean semi-major axis is kept constant during the day of observation, whereas an uncontrolled satellite would have decayed by a small amount.

6.2.5. Station keeping at large distance

In the previous scenario, station keeping is performed at a close relative distance with respect to the reference non-decaying orbiting point. On the contrary, in the next simulation the MPC station keeping accuracy is tested when the non-decaying chief is instead placed at a large along-track distance of 50 km. The corresponding ROE state to be maintained is reported in Table 6.11. Constraints on thrusting angles are added also in this simulation, considering the same thrust cone imposed in the previous formation maintenance simulation, fixing the in-plane angle to zero and a allowing a maximum off-plane angle value of 45° in positive or negative direction.

Table 6.11: Relative Orbital Elements state to be maintained in the station keeping at far distance simulation.

$a\delta a$ [m]	$a\delta\lambda$ [m]	$a\delta e_x$ [m]	$a\delta e_y$ [m]	$a\delta i_x$ [m]	$a\delta i_y$ [m]
0	50000	0	0	0	0

Moreover, a larger sampling time of 200 s (1/28 of the prediction horizon) is selected in this specific case. The same control horizon of seven sampling steps corresponds then to about a quarter of an orbit, relaxing even more the computation frequency. The simulation is performed over a week, in order to check the long-term evolution of Relative Orbital Elements and to be able to assume that the resulting control will be repeated in time with sufficient similarity to retrieve a reliable yearly ΔV estimate.

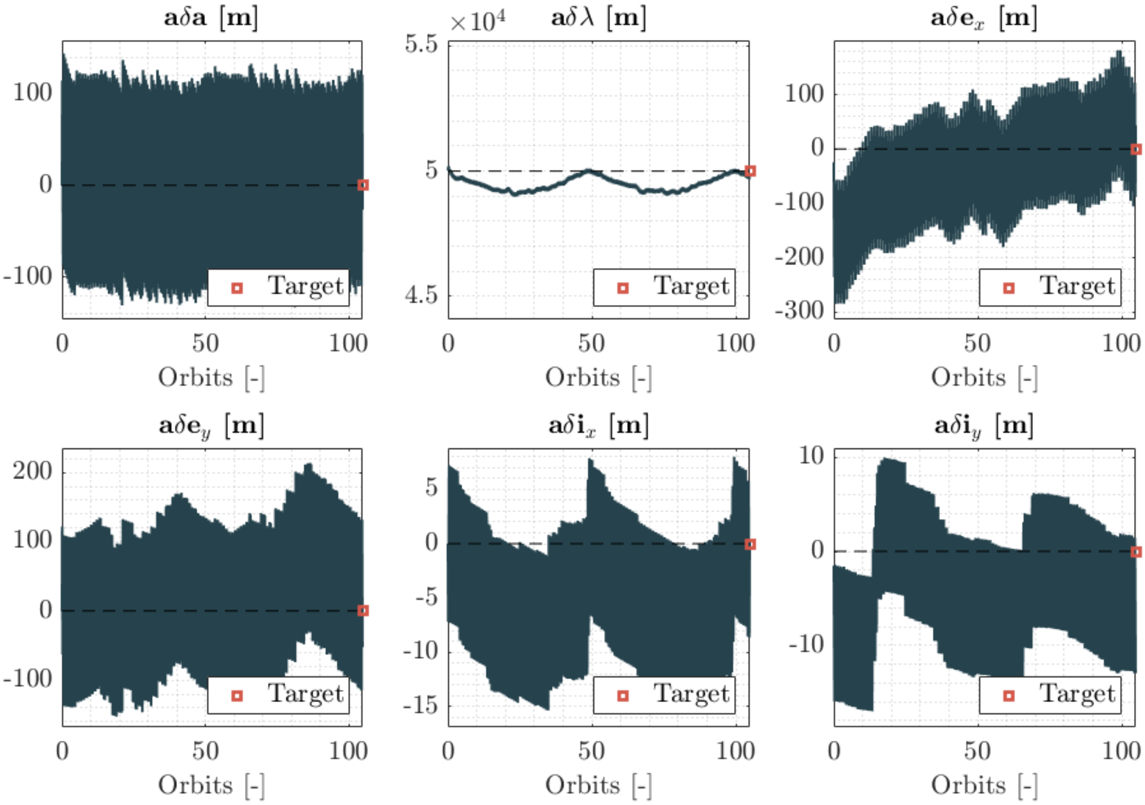


Figure 6.16: Evolution of Relative Orbital Elements in the station keeping at far distance simulation.

The evolution of ROEs in time is reported in Figure 6.16. From the figure it is possible to observe that all Relative Orbital Elements, both in-plane and off-plane, are controlled throughout the simulation time span. In this specific case, the along-track separation represented by the relative mean longitude is definitely the most relevant aspect. This is controlled with an accuracy of about 1 km, correspondent to 2% of the reference along-track distance. This error is far larger than the results obtained in previous simulations and also other Relative Orbital Elements oscillate much more around their nominal values. This inaccuracy is attributed to the large separation at which formation keeping is performed, bringing out the limits of the model linearization. Indeed, even though a ROE representation behaves better than HCW equations at far distances and in noncircular orbits, the simplifying assumptions made to derive the plant matrices show their limitations in more challenging situations. In any case the obtained accuracy of 1 km could be good enough in several applications, especially considering such a large along-track separation. Also in this case the spacecraft semi-major axis is controlled to counteract the effects of drag like in previous simulations.

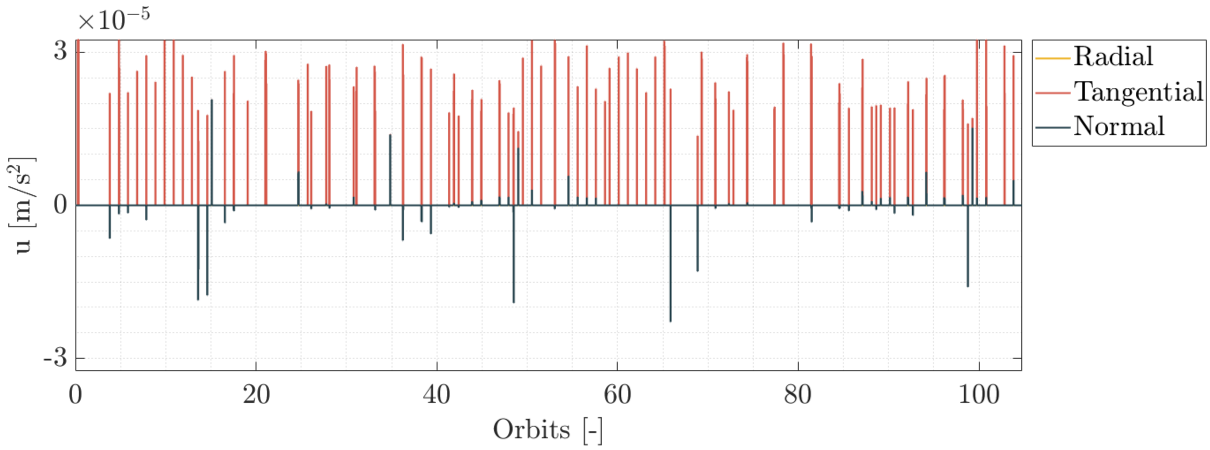


Figure 6.17: Control accelerations over a week in the RTN frame for the station keeping at far distance simulation.

In Figure 6.17 are reported the components of the controlling acceleration profile during the simulated week. Naturally, tangential acceleration is dominant, compensating the effect of differential drag and maintaining the orbit altitude. In addition, a normal component is provided more sparsely to control the relative inclination vector as well. The frequency of thrusting actions needed for station keeping is a relevant information for mission design. In this case the averaged frequency of control amounts to about 14 manoeuvres a day, slightly less than one per orbit. All these thrusting arcs are as long as the minimum impulse they can give, equal to the MPC's sampling time. The selected value of 200 s leaves more than enough time along the orbit to perform observation or scientific tasks and limits the burden on the electric power subsystem.

The station keeping ΔV necessary for the simulated window is of 0.6285 m/s, which corresponds to a yearly ΔV estimate of about 34 m/s. This value is coherent to the ones found in the previous formation keeping simulations and indicates that the large along-track distance has an impact on control accuracy but not on control effort. Several trade-offs between accuracy, yearly ΔV, manoeuvring frequency and length of thrusting arcs can be performed by acting on the MPC parameters, on its constraints and on the weights of the Optimal Control Problem cost function. This tuning is strongly requirements-specific, the results provided in this test aim at providing an example of what can be achieved taking all these aspects into consideration.

However, what is probably a better option for this application in which collision risk is negligible, is to control the two spacecrafts separately with respect to two virtual orbiting points separated along-track or even to choose ground-base control, in which satellite positional measurements are sent by each satellite of the formation to the ground control center, which will command the satellites by uploading the desired control action. Controlling from ground allows to use powerful trajectory optimization algorithms which cannot be implemented on board, but that are capable of providing the optimal solution in terms of ΔV when the frequency of manoeuvres is not so high and distances are large enough that requirements on quickness of response are not so strict.

6.2.6. Constant drift acquisition and maintenance

In previous simulations, only transfers between bounded orbits have been simulated. However, also unbounded drifting orbits can be useful in some mission applications, for example if used as inspection trajectories for noncooperative targets. Indeed, these orbits allow a close scanning of the target by moving in a helicoidal trajectory enclosing it. Given a drift of distance d_{drift} and duration T_{drift}, the drifting rate can be computed by inverting the keplerian motion equation, allowing to retrieve the desired $a\delta a$ as [46]:

$$a\delta a = \frac{2 \cdot d_{drift}}{3n \cdot T_{drift}} \tag{6.1}$$

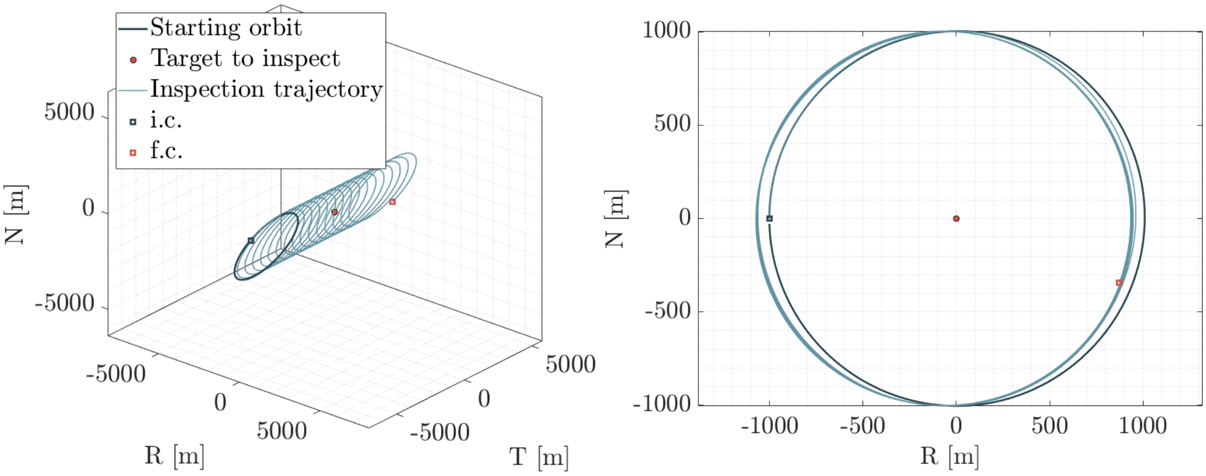
By imposing a drift in positive direction of 10 km to be performed over one day, the resulting value for the relative semi-major axis difference $a\delta a$ equals to about -70 m. Accordingly, the starting and target Relative Orbital Elements for this drift injection are shown in Table 6.12, assuming to transfer to the drifting trajectory from an holding orbit of same size. Also in this case eccentricity/inclination vector separation is used as a tool to design a passively safe trajectory.

Table 6.12: Starting and target Relative Orbital Elements for the drift injection. A symbolic value equal to the initial is given for $\delta\lambda$, which however is not weighted in the OCP cost function to be free to variate during the manoeuvre.

	$a\delta a$ [m]	$a\delta\lambda$ [m]	$a\delta e_x$ [m]	$a\delta e_y$ [m]	$a\delta i_x$ [m]	$a\delta i_y$ [m]
$\mathbf{ROE_0}$	0	-5000	0	1000	0	1000
$\mathbf{ROE_T}$	-70	-5000	0	1000	0	1000

Unlike previous simulations, in this case the chaser spacecraft cannot be assumed to be controlled with respect to a non-decaying point, as the target is possibly a noncooperative uncontrolled piece of debris. For this reason, in this specific scenario, the drag plant matrix is left out of the propagation and the correction of the undesired drift resulting from the presence of differential drag is left to the dynamic update of the Model Predictive Controller. For the target propagation it is assumed that its ballistic coefficient is three times larger than the one of the chaser spacecraft.

In Figure 6.18 is shown the inspection trajectory both in the RTN reference frame centered in the target and in the RN plane. It can be observed that the object of the inspection is kept at the center of the circular relative orbit throughout the entire scanning phase, which is performed at a constant drifting rate.



(a) Drifting orbit in the RTN reference frame.

(b) Drifting orbit in the RN plane.

Figure 6.18: Trajectory of the drifting orbit in RTN and RN frames. The target, represented by the red dot, is kept at the center of the helicoidal path throughout the whole drifting phase.

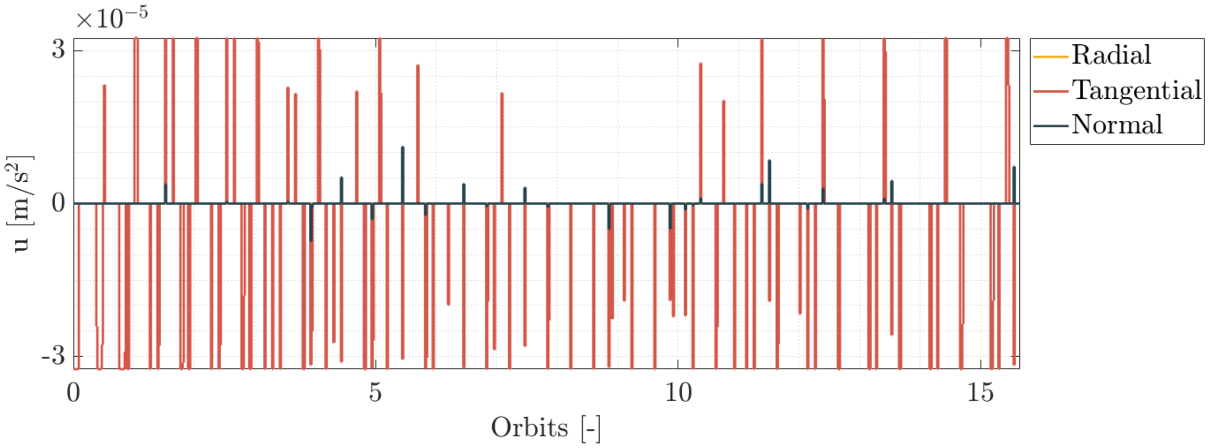


Figure 6.19: Control accelerations in the RTN frame throughout the inspection trajectory.

Without a modelling of differential drag, it is impossible for the MPC to predict and counteract its effects efficiently. However, its feedback behaviour allows to maintain a constant drifting rate by correcting the generating undesired drifts. This corrections, represented in the control accelerations plot in Figure 6.19, are excessively frequent, leading to a ΔV amounting to 0.4102 m/s over the day. Moreover, the frequently requested slew manoeuvres would be incompatible with the scanning operations, which would ask for an almost continuous pointing of the target. In this case, it would be better to exploit the drag-induced drift to achieve a control-free drifting trajectory, if mission requirements allow for a variable drift rate, or to retrieve an accurate model for the relative drift. This would be particularly hard to predict due to the uncertainties in modeling the aerodynamic properties of the uncooperative target.

6.2.7. Close-up manoeuvre in highly eccentric orbit

The choice of using Relative Orbital Elements instead of a representation based on the translational state allows to extend the validity of the model also to noncircular orbits. To check the accuracy of the algorithm in this environment, a close-up manoeuvre in High Elliptical Orbit (HEO) is simulated. For this particular case, a Molniya-inspired orbit is selected, whose initial keplerian parameters are reported in Table 6.13. The starting and target relative states are listed in Table 6.14.

Table 6.13: Starting orbital elements of the reference HEO orbit.

a [km]	e [-]	i [°]	Ω [°]	ω [°]	M_0 [°]
17445	0.6	63.4	120	270	0

Table 6.14: Starting and target Relative Orbital Elements for the transfer in HEO.

	$a\delta a$ [m]	$a\delta\lambda$ [m]	$a\delta e_x$ [m]	$a\delta e_y$ [m]	$a\delta i_x$ [m]	$a\delta i_y$ [m]
ROE$_0$	0	0	0	500	0	500
ROE$_T$	0	0	0	200	0	200

Due to the increase of orbital period with respect to the previous cases, the parameters of the Model Predictive Controller are changed accordingly. The new values are listed in Table 6.15.

Table 6.15: Parameters of the Model Predictive Controller used for the HEO simulation.

Parameter	Value
Prediction Horizon	23200 s, ∼1 orbit
Sampling time, T_s	200 s
Control Horizon	14 $\cdot T_s$, ∼1/8 orbit

Even in this application the MPC shows a good accuracy, indeed, the simulation is stopped when all Relative Orbital Elements reach the target with a tolerance of five meters on their value multiplied by the reference semi-major axis. The resulting transfer is completed successfully in about three orbits, the trajectory is shown in Figure 6.20.

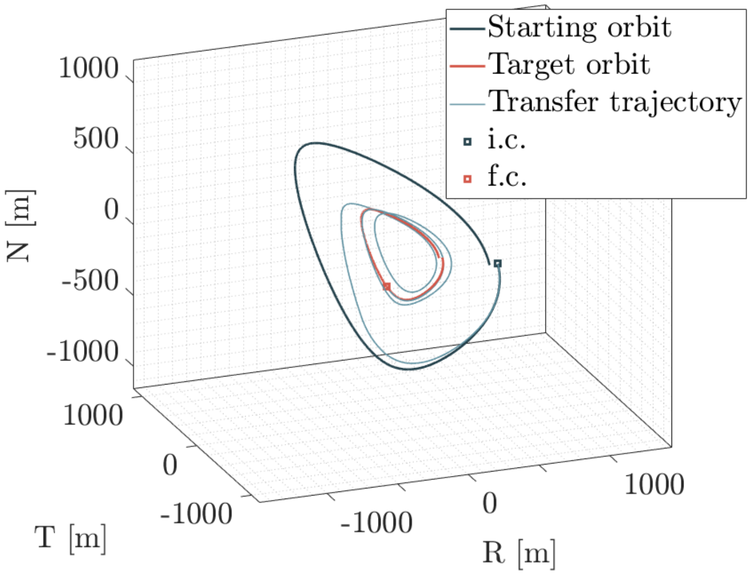


Figure 6.20: HEO close-up trajectory in the RTN reference frame.

The resulting control profile is shown in Figure 6.21. It can be observed that the prescribed thrusting arcs are quite sparse and very narrow. the correspondent ΔV needed to perform the transfer amounts to 0.1888 m/s.

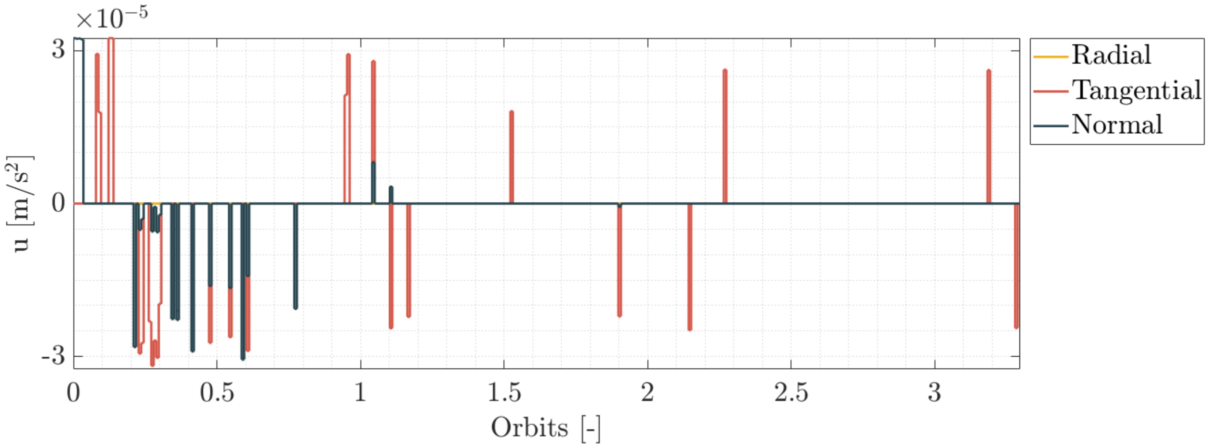


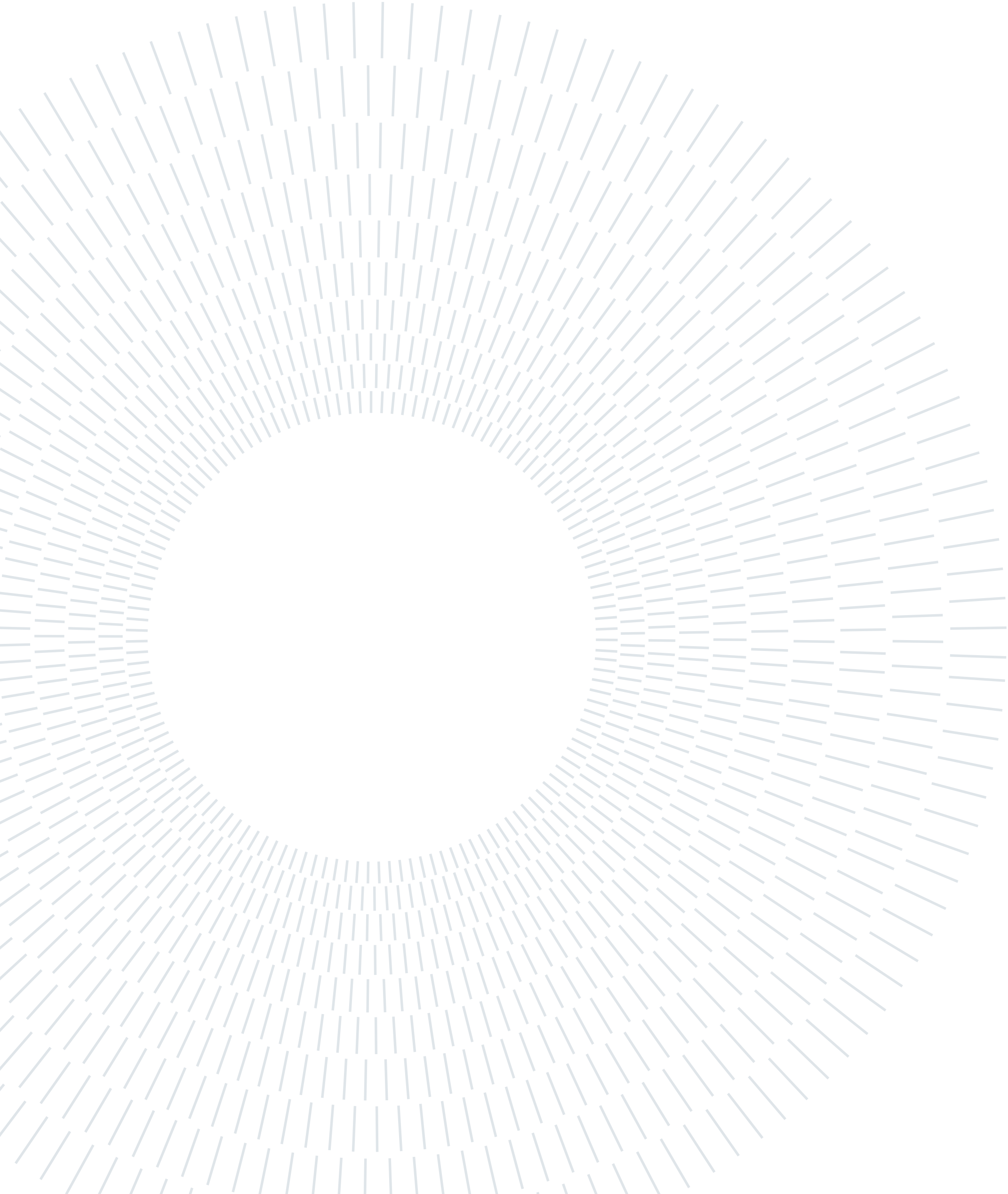
Figure 6.21: Control accelerations in the RTN reference frame for the close-up trajectory in Highly-Elliptical-Orbit.

6.3. Considerations on the simulations results

The developed convex Model Predictive Controller demonstrated good flexibility by performing well in various simulations of formation acquisition, reconfiguration, and maintenance for different relative and absolute orbit shapes.

For what concerns fuel efficiency, in the simpler manoeuvres isolating in-plane or out-of-plane control, the MPC was able to provide a close to optimal result, confirming the results obtained in the validation performed in Chapter 5. On the other hand, when these two components are mixed, or in particularly challenging scenarios, the optimality of the result could be improved. This is most evident for spacecrafts B and C in the tetrahedron acquisition scenario, in which they needed to acquire both relative eccentricity and relative inclination components and the trajectories of the four spacecrafts were optimized at the same time by the MPC. On the other hand, the ΔV that was provided for drag compensation and station keeping was consistent for all simulated scenarios with the results provided in the validation.

The accuracy that was provided by the MPC was good in all simulations, in which the spacecrafts were able to acquire the desired states and maintain them with an accuracy in the order of meters. The only exception was the station keeping at far distance simulation, for which, however, the relative error with respect to the along-track distance could still be considered satisfactory. It has to be considered, though, that this scenario and the drift acquisition one would probably benefit from a rethinking of the control strategy. To conclude, the collision avoidance constraint demonstrated to be effective in computing safe trajectories which respected the threshold distance in situations where its inclusion was necessary. Limits on thrust module and angles were always satisfied in all applications and the slew rate constraint proved to avoid immediate 180° rotations, providing some time to reorientate the engine.



7 | Conclusions and future developments

In this document a novel on-board Model Predictive Controller for optimal formation acquisition and maintenance in high-drag environment is proposed. The main advanced contributions to the state of the art are:

- The use of a convexified linear dynamics expressed in Relative Orbital Elements in a Model Predictive Controller, instead of the classical cartesian representation in the Local-Vertical-Local-Horizontal reference frame. The formulation is augmented to control the spacecrafts both relatively and absolutely in a high-drag environment with respect to a non-decaying orbiting point. For this scope, the derivation of a dedicated plant matrix is proposed.
- The introduction of constraints in the convexified formulation which resemble the limitations of mounting a single low-thrust engine on board, in order to obtain a feasible control profile for an average micro-satellite.

In Chapter 2, the dynamics linear model was introduced, along with the derivation of a specific plant matrix included to consider the effects of drag with respect to a non-decaying orbiting slot. The precision of the developed drag matrix was also validated using a numerical propagator to verify its accuracy. In Chapters 3 and 4, the Model Predictive Control implementation is described. Particular attention is focused on the convexification of the objective function and constraints of the MPC Optimal Control Problem. Constraints include limitations related to the spacecraft actuation and collision avoidance, for which a linear mapping between ROE and RTN states is provided. In Chapter 5, fuel efficiency was assessed via a comparison with a recent nonconvex nonlinear MPC solution and with a closed-form fuel-optimal impulsive solution available in literature, showing a close to optimal behaviour even imposing all the previously discussed constraints. An additional result obtained with a direct single shooting technique is added to the comparison to have an optimal low-thrust benchmark. In the same chapter, the effect of the inclusion of the drag plant matrix on the control accuracy is also assessed, showing an increment of tracking precision with no increase in computational effort. In Chapter 6, the controller performance was simulated in several mission scenarios, demonstrating to be able to perform formation acquisition, reconfiguration, and maintenance in a timely and efficient manner while keeping a constant semi-major axis at a very low altitude. Fuel optimality was confirmed in the simpler manoeuvres, but can be improved in more challenging situations where the effects of linearization come into play or when the guidance of plenty of satellites is computed at once. In addition, the collision avoidance algorithm was tested with good results, proving to maintain a minimum desired inter-satellite distance when introduced. The introduced approach is scalable to formations of arbitrary size, although it should be considered that computational time scales with the dimension of the optimization vector.

Despite the promising results, several future developments can be individuated to improve the controller performance and to test further its robustness, making it ready for on-board implementation:

- The model used for the linear propagation of the dynamics can be improved. The model can be extended to cover the effects of solar radiation pressure, third body perturbations and higher order geopotential effects. Plant matrices relative to these disturbances have already been proposed in literature [12, 14]. Moreover, a plant matrix for considering the effects of differential drag with respect to an uncooperative target can be added.

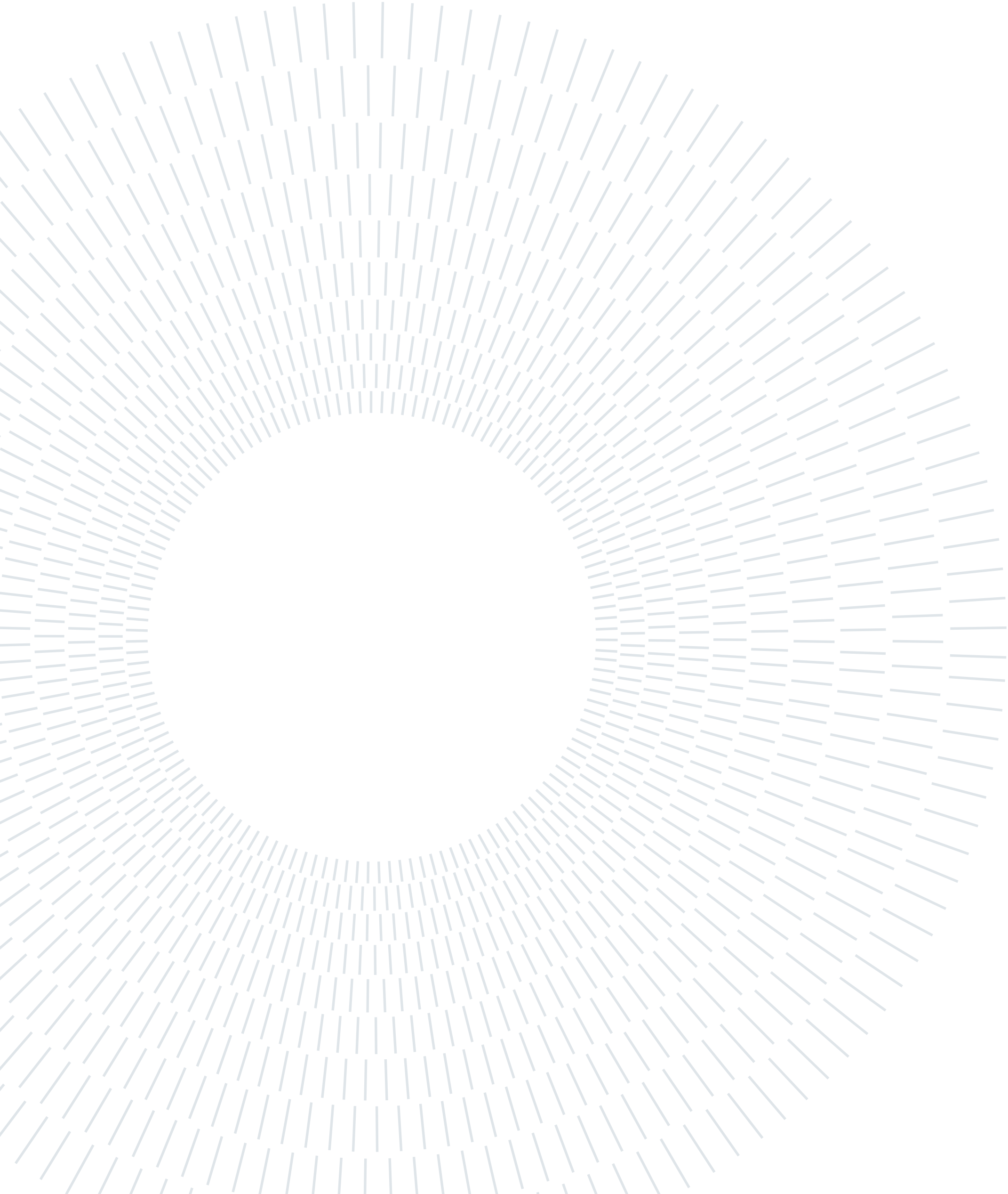
- A method to improve the optimality of the solution can be found. This would be especially beneficial in situations in which the algorithm is particularly stressed, as in environments where nonlinear effects are strong or when multiple spacecrafts are controlled together in the formation. The introduction of the so-called "d-term", developed by Catanoso in Reference [22], in the OCP cost function can be a first step in that direction.

- A cleaner formulation to add the constraints on thrusting angles or slewing rate can be found. In this respect, a full 6 DoF model can be developed including the spacecrafts attitude in the state vector.

- An objective strategy to find cost function weighting matrices can be derived to obtain optimal values according to the desired outcome. In this direction, a more direct link between cost function weights and transfer time can be found. A genetic algorithm could be a good tool to find these optimal gains.

- The simulation loop can be extended by adding a navigation filter. In this way, the algorithm robustness would be tested in presence of noisy data coming from simulated on-board sensors.

- The algorithm can be written in a programming language suitable for embedded software, mainly `C` or `C++`, allowing to perform tests on a board to really assess computational efficiency and on-board implementation viability.



Bibliography

[1] B. D. Tapley, S. Bettadpur, M. Watkins, and C. Reigber. The Gravity Recovery and Climate Experiment: Mission Overview and Early Results. *Geophysical Research Letters*, 2004.

[2] G. Krieger, A. Moreira, H. Fiedler, I. Hajnsek, M. Werner, M. Younis, and M. Zink. TanDEM-X: A Satellite Formation for High-Resolution SAR Interferometry. *IEEE Transactions on Geoscience and Remote Sensing*, November 2007.

[3] S. Persson, P. Bodin, E. Gill, and J. Jorgensen. PRISMA - An Autonomous Formation Flying Mission. *Proceedings of the ESA Small Satellite Systems and Services Symposium (4S)*, 2006.

[4] J. S. Llorente, A. Agenjo, C. Carrascosa, C. de Negueruela, A. Mestreau-Garreau, A. Cropp, and A. Santovincenzo. PROBA-3: Precise Formation Flying Demonstration Mission. *Acta Astronautica*, 2013.

[5] W. K. Clohessy and R. S. Wiltshire. Terminal Guidance System for Satellite Rendezvous. *Journal of Guidance, Control, and Dynamics*, 1960.

[6] H. S. London. Second Approximation to the Solution of the Rendezvous Equations. *AIAA Journal*, 1963.

[7] M. L. Anthony and F. T. Sasaki. Rendezvous Problem for Nearly Circular Orbits. *AIAA Journal*, 1965.

[8] M. T. Stringer, B. A. Newman, T. A. Lovell, and A. Omran. Analysis of a New Nonlinear Solution of Relative Orbital Motion. *23rd International Symposium on Space Flight Dynamics*, November 2012.

[9] K. Yamanaka and F. Ankersen. New State Transition Matrix for Relative Motion on an Arbitrary Elliptical Orbit. *Journal of Guidance, Control, and Dynamics*, January 2002.

[10] M. Willis, K. Alfriend, and S. D'Amico. Second-Order Analytic Solution for Relative

Motion on Arbitrarily Eccentric Orbits. *29th AAS/AIAA Space Flight Mechanics Meeting*, January 2019.

[11] A. W. Koenig, T. Guffanti, and S. D'Amico. New State Transition Matrices for Spacecraft Relative Motion in Perturbed Orbits. *Journal of Guidance, Control, and Dynamics*, August 2016.

[12] T. Guffanti, S. D'Amico, and M. Lavagna. Long-term Analytical Propagation of Satellite Relative Motion in Perturbed Orbits. *27th AAS/AIAA Space Flight Mechanics Meeting*, February 2017.

[13] G. Gaias, JS. Ardaens, and O. Montenbruck. Model of J2 perturbed satellite relative motion with time-varying differential drag. *Celestial Mechanics and Dynamical Astronomy*, August 2015.

[14] T. Guffanti and S. D'Amico. Linear Models for Spacecraft Relative Motion Perturbed by Solar Radiation Pressure. *Journal of Guidance, Control, and Dynamics*, September 2019.

[15] M. Chernick and S. D'Amico. New Closed-Form Solutions for Optimal Impulsive Control of Spacecraft Relative Motion. *Journal of Guidance, Control, and Dynamics*, November 2018.

[16] L. Steindorf, S. D'Amico, J. Scharnagl, F. Kempf, and K. Schilling. Constrained Low-Thrust Satellite Formation-Flying Using Relative Orbit Elements. *27th AAS/AIAA Space Flight Mechanics Meeting*, January 2017.

[17] S. Silvestrini and M. Lavagna. Neural-aided GNC reconfiguration algorithm for distributed space system: development and PIL test. *Advances in Space Research*, March 2021.

[18] B. Acikmese, D. Scharf, F. Hadaegh, and E. Murray. A Convex Guidance Algorithm for Formation Reconfiguration. *AIAA Guidance, Navigation, and Control Conference and Exhibit*, August 2006.

[19] D. Morgan, S. Chung, and F. Y. Hadaegh. Model Predictive Control of Swarms of Spacecraft Using Sequential Convex Programming. *Journal of Guidance, Control, and Dynamics*, November 2014.

[20] S. Sarno, J. Guo, M. D'Errico, and E. Gill. A guidance approach to satellite formation reconfiguration based on convex optimization and genetic algorithms. *Advances in Space Research*, January 2020.

[21] F. Scala, G. Gaias, C. Colombo, and M. Martin-Neira. Design of optimal low-thrust manoeuvres for remote sensing multi-satellite formation flying in low Earth orbit. *Advances in Space Research*, December 2021.

[22] D. Catanoso, F. Kempf, K. Schilling, and S. D'Amico. Networked Model Predictive Control for Satellite Formation Flying. *10th International Workshop of Satellites Constellations and Formation Flying*, July 2019.

[23] S. D'Amico. *Autonomous formation flying in low earth orbit.* PhD thesis, TU Delft, 2010.

[24] L. Steindorf, S. D'Amico, J. Scharnagl, F. Kempf, and K. Schilling. Constrained Low-Thrust Satellite Formation-Flying Using Relative Orbit Elements. *27th AAS/AIAA Space Flight Mechanics Meeting*, January 2017.

[25] D. Morgan, S. Chung, L. Blackmore, B. Acikmese, D. Bayard, and F. Y. Hadaegh. Swarm-Keeping Strategies for Spacecraft Under J2 and Atmospheric Drag Perturbations. *Journal of Guidance, Control, and Dynamics*, September 2012.

[26] R. Battin. *An Introduction to the Mathematics and Methods of Astrodynamics*. AIAA Education Series, 1999. Chapter 10.

[27] G. Gaias and S. D'Amico. Impulsive Maneuvers for Formation Reconfiguration Using Relative Orbital Elements. *Journal of Guidance, Control, and Dynamics*, June 2015.

[28] MathWorks. Convex optimization. Last visited on 2 April 2022.

[29] M. Grant and S. Boyd. CVX: Matlab software for disciplined convex programming, version 2.1. `http://cvxr.com/cvx`, March 2014.

[30] M. Grant and S. Boyd. Graph implementations for nonsmooth convex programs. In V. Blondel, S. Boyd, and H. Kimura, editors, *Recent Advances in Learning and Control*, Lecture Notes in Control and Information Sciences, pages 95–110. Springer-Verlag Limited, 2008. `http://stanford.edu/~boyd/graph_dcp.html`.

[31] S. Diamond and S. Boyd. CVXPY: A Python-embedded modeling language for convex optimization. *Journal of Machine Learning Research*, 17(83):1–5, 2016.

[32] A. Agrawal, R. Verschueren, S. Diamond, and S. Boyd. A rewriting system for convex optimization problems. *Journal of Control and Decision*, 5(1):42–60, 2018.

[33] J. F. Sturm. Using SeDuMi 1.02, A Matlab toolbox for optimization over symmetric cones. *Optimization Methods and Software*, 1999.

[34] K. C. Toh, M. J. Todd, and R. H. Tütüncü. Sdpt3 — a matlab software package for semidefinite programming, version 1.3. *Optimization Methods and Software*, 1999.

[35] J. Chu. *Dynamics, Distributed Control and Autonomous Cluster Operations of Fractionated Spacecraft.* PhD thesis, TU Delft, 2015.

[36] S. Silvestrini, V. Pesce, and M. Lavagna. Distributed Autonomous Guidance, Navigation and Control loop for Formation Flying Spacecraft Reconfiguration. *EUROGNC19 – 5th CEAS Conference on Guidance, Navigation & Control*, April 2019.

[37] F. Y. Hadaegh, A. R. Ghavimi, G. Singh, and M. Quadrelli. A Centralized Optimal Controller for Formation Flying Spacecraft. Technical report, Jet Propulsion Laboratory, California Institute of Technology, 2000.

[38] W. Ren and R. W. Beard. Decentralized Scheme for Spacecraft Formation Flying via the Virtual Structure Approach. *Journal of Guidance, Control, and Dynamics*, January 2004.

[39] A. Rohatgi. Webplotdigitizer: Version 4.5, 2021.

[40] M. P. Kelly. Transcription Methods for Trajectory Optimization. February 2015.

[41] A. V. Rao. A Survey of Numerical Methods for Optimal Control. *Advances in the Astronautical Sciences*, January 2010.

[42] F. Topputo and C. Zhang. Survey of Direct Transcription for Low-Thrust Space Trajectory Optimization with Applications. *Abstract and Applied Analysis*, June 2014.

[43] M. R. Drinkwater, R. Haagmans, D. Muzi, A. Popescu, R. Floberghagen, M. Kern, and M. Fehringer. The GOCE gravity mission: ESA's first core explorer. *Proceedings of the 3rd International GOCE User Workshop*, November 2006.

[44] S. D'Amico and O. Montenbruck. Proximity Operations of Formation-Flying Spacecraft Using an Eccentricity/Inclination Vector Separation. *Journal of Guidance, Control, and Dynamics*, May 2006.

[45] J. J. Guzman. Tetrahedron Formation Control. Technical report, a.i. solutions, Inc., 2003.

[46] S. Silvestrini, J. Prinetto, G. Zanotti, and M. Lavagna. Design of Robust Passively Safe Relative Trajectories for Uncooperative Debris Imaging in Preparation to Removal. *Journal of Guidance, Control, and Dynamics*, July 2019.