Embedded and validated control algorithms for the spacecraft rendezvous

Paulo Ricardo Arantes Gilz

THÈSE

En vue de l'obtention du

DOCTORAT DE L'UNIVERSITÉ FÉDÉRALE TOULOUSE MIDI-PYRÉNÉES

Délivré par :
l'Université Toulouse 3 Paul Sabatier (UT3 Paul Sabatier)

Présentée et soutenue le *17/10/2018* **par :**

PAULO RICARDO ARANTES GILZ

Embedded and validated control algorithms for the spacecraft rendezvous

---

JURY

M. LUCA ZACCARIAN	LAAS-CNRS	Président du Jury
M. ALEJANDRO HERNÁN GONZÁLEZ	INTEC / CONICET / UNL	Rapporteur
M. ERIC KERRIGAN	Imperial College London	Rapporteur
MME ESTELLE COURTIAL	Université d'Orléans / PRISME	Examinateur
MME GABRIELLA GAIAS	DLR München	Examinateur
M. MASSIMILIANO VASILE	University of Strathclyde	Examinateur
M. FRÉDÉRIC CAMPS	LAAS-CNRS	Membre invité

École doctorale et spécialité :
MITT : Domaine Mathématiques : Mathématiques appliquées
Unité de Recherche :
Laboratoire d'analyse et d'architecture des systèmes
Directeur(s) de Thèse :
Mme Mioara JOLDEŞ et *M. Christophe LOUEMBET*
Rapporteurs :
M. Alejandro HERNÁN GONZÁLEZ et *M. Eric KERRIGAN*

Acknowledgments

Infinitely many thanks to Mioara Joldeş and Christophe Louembet for enlightening my path during this three-years-long journey. I will never be able to show you how much I am thankful for your advices and collaboration.

A special thanks to Frédéric Camps, whose help and dedication were essential for the accomplishment and quality of the produced work.

Thanks to all the members of the Methods and Algorithms in Control group of the LASS-CNRS for hosting me and for supporting me in this project.

I would also like to thank my coworkers from INSA Toulouse, Saïd Zabi, Pierre Dauchez, Audine Subias and Vincent Mahout, for all the moments together and for everything they taught me.

Huge thanks to all the PhD students and interns that I had the pleasure to meet and with whom I could share my joys and sorrows all along these years. You did made it much more enjoyable this way. ¡Sofia, gracias por la ayuda, consejos y conversaciones, sis!

Obrigado a minha mãe, meu pai e minha irmã. Tirei de vocês a oportunidade de estarmos juntos durante três longos anos. Serei sempre grato pelo carinho e preocupação de vocês. Obrigado também a meus avós Zofia e Viktor. Só pude chegar onde cheguei pela ajuda de vocês, mas infelizmente vocês partiram antes de poder ver. Esta tese é para vocês.

Funding Sources

This PhD thesis was supported by the FastRelax (ANR-14-CE25-0018-01) project of the French National Agency for Research (ANR).

Abstract

Autonomy is one of the major concerns during the planning of a space mission, whether its objective is scientific (interplanetary exploration, observations, etc.) or commercial (service in orbit). For space rendezvous, this autonomy depends on the on-board capacity of controlling the relative movement between two spacecraft. In the context of satellite servicing (troubleshooting, propellant refueling, orbit correction, end-of-life deorbit, etc.), the feasibility of such missions is also strongly linked to the ability of the guidance and control algorithms to account for all operational constraints (for example, thruster saturation or restrictions on the relative positioning between the vehicles) while maximizing the life of the vehicle (minimizing propellant consumption). The literature shows that this problem has been intensively studied since the early 2000s. However, the proposed algorithms are not entirely satisfactory. Some approaches, for example, degrade the constraints in order to be able to base the control algorithm on an efficient optimization problem. Other methods accounting for the whole set of constraints of the problem are too cumbersome to be embedded on real computers existing in the spaceships.

The main object of this thesis is the development of new efficient and validated algorithms for the impulsive guidance and control of spacecraft in the context of the so-called "hovering" phases of the orbital rendezvous, i.e. the stages in which a secondary vessel must maintain its position within a bounded area of space relatively to another main vessel. The first contribution presented in this manuscript uses a new mathematical formulation of the space constraints for the relative motion between spacecraft for the design of control algorithms with more efficient computational processing compared to traditional approaches. The second and main contribution is a predictive control strategy that has been formally demonstrated to ensure the convergence of relative trajectories towards the "hovering" zone, even in the presence of disturbances or saturation of the actuators. Specific computational developments have demonstrated the embeddability of these control algorithms on a board containing a FPGA-synthesized LEON3 microprocessor certified for space flight, reproducing the performance of the devices usually used in flight. Finally, tools for rigorous approximation of functions were used to obtain validated solutions of the equations describing the linearized relative motion, allowing a simple certified propagation of the relative trajectories via polynomials and the verification of the respect of the constraints of the problem.

Résumé

L'autonomie est l'une des préoccupations majeures lors du développement de missions spatiales que l'objectif soit scientifique (exploration interplanétaire, observations, etc) ou commercial (service en orbite). Pour le rendez-vous spatial, cette autonomie dépend de la capacité embarquée de contrôle du mouvement relatif entre deux véhicules spatiaux. Dans le contexte du service aux satellites (dépannage, remplissage additionnel d'ergols, correction d'orbite, désorbitation en fin de vie, etc), la faisabilité de telles missions est aussi fortement liée à la capacité des algorithmes de guidage et contrôle à prendre en compte l'ensemble des contraintes opérationnelles (par exemple, saturation des propulseurs ou restrictions sur le positionnement relatif entre les véhicules) tout en maximisant la durée de vie du véhicule (minimisation de la consommation d'ergols). La littérature montre que ce problème a été étudié intensément depuis le début des années 2000. Les algorithmes proposés ne sont pas tout à fait satisfaisants. Quelques approches, par exemple, dégradent les contraintes afin de pouvoir fonder l'algorithme de contrôle sur un problème d'optimisation efficace. D'autres méthodes, si elles prennent en compte l'ensemble du problème, se montrent trop lourdes pour être embarquées sur de véritables calculateurs existants dans les vaisseaux spatiaux.

Le principal objectif de cette thèse est le développement de nouveaux algorithmes efficaces et validés pour le guidage et le contrôle impulsif des engins spatiaux dans le contexte des phases dites de *"hovering"* du rendez-vous orbital, i.e. les étapes dans lesquelles un vaisseau secondaire doit maintenir sa position à l'intérieur d'une zone délimitée de l'espace relativement à un autre vaisseau principal. La première contribution présentée dans ce manuscrit utilise une nouvelle formulation mathématique des contraintes d'espace pour le mouvement relatif entre vaisseaux spatiaux pour la conception d'algorithmes de contrôle ayant un traitement calculatoire plus efficace comparativement aux approches traditionnelles. La deuxième et principale contribution est une stratégie de contrôle prédictif qui assure la convergence des trajectoires relatives vers la zone de *"hovering"*, même en présence de perturbations ou de saturation des actionneurs. Un travail spécifique de développement informatique a pu démontrer l'embarquabilité de ces algorithmes de contrôle sur une carte contenant un microprocesseur LEON3 synthétisé sur FPGA certifié pour le vol spatial, reproduisant les performances des dispositifs habituellement utilisés en vol. Finalement, des outils d'approximation rigoureuse de fonctions ont été utilisés pour l'obtention des solutions validées des équations décrivant le mouvement relatif linéarisé, permettant ainsi une propagation certifiée simple des trajectoires relatives via des polynômes et la vérification du respect des contraintes du problème.

Contents

Nomenclature

ΔV	Vector of instantaneous velocity corrections
$\mathcal{J}$	Fuel consumption
μ	Earth's gravitational constant
ν	True anomaly
Ω	Longitude of the ascending node
ω	Argument of perigee
$\overline{\Delta V}$	Thrusters saturation threshold
$\Phi,\ \Phi_D$	State-transition matrices
a	Semi-major axis
D	Deaconu's vector of parameters
e	Eccentricity
G	Universal gravitational constant
i	Orbit inclination
I_n	Identity matrix of oder n
O_n	Null matrix of oder n
X	Relative state between spacecraft in the LVLH frame
LMI	Linear matrix inequality
LODE	Linear ordinary differential equations
LP	Linear program
LTV	Linear Time-Varying
LVLH	Local-vertical / local-horizontal
MPC	Model predictive control
RPA	Rigorous polynomial approximation
SDP	Semi-definite program
SIP	Semi-infinite program

Introduction

Spacecraft autonomy has become an important feature in the development of space missions, especially when ground operations are impracticable due to a large number of operations or an elevated communication time. Efficient embedded algorithms dedicated to autonomous decision making and maneuvering have already been used in many space projects, such as: the Soyuz and ATV automated docking systems [37, 46], which are highly sophisticated spacecraft capable of automatically docking to the International Space Station using their own propulsion and navigation systems; and the Japanese Hayabusa asteroid touchdown mission [61], during which a spacecraft was supposed to touch the surface of an asteroid with its sample capturing device and then take off again.

Indeed, mastering the implementation of efficient embedded algorithms and assessing their performance are crucial for the accomplishment of mission goals, but also open a venue of economical opportunities and allow the feasibility of future space applications. One example in the context of space exploration is the Mars Sample Return (MSR)[1] joint mission between ESA and NASA, during which a rover would be deployed on the surface of Mars to collect samples that later would be sent back to Earth using a third orbiter spacecraft (Fig. 1).

Figure 1 – Illustration of the Mars Sample Return mission (property of NASA/JPL).

A trending topic that is currently being studied by several space agencies and private companies is the on-orbit servicing. Two examples can be mentioned to illustrate this appli-

[1] https://www.jpl.nasa.gov/missions/mars-sample-return-msr/

cation: the DARPA's Robot Servicing of Geosynchronous Satellites (RSGS)[2] program that seeks to demonstrate the feasibility of a generic robotic servicing vehicle, with the ability of executing a variety of on-orbit missions in different scenarios; and the ESA's e.deorbit[3] active debris removal mission, in which a primary satellite would chase a secondary ESA-owned uncooperative satellite in low orbit, capture it (using a net and the other a robotic arm) and safely burn it up in a controlled atmospheric reentry (Fig. 2).

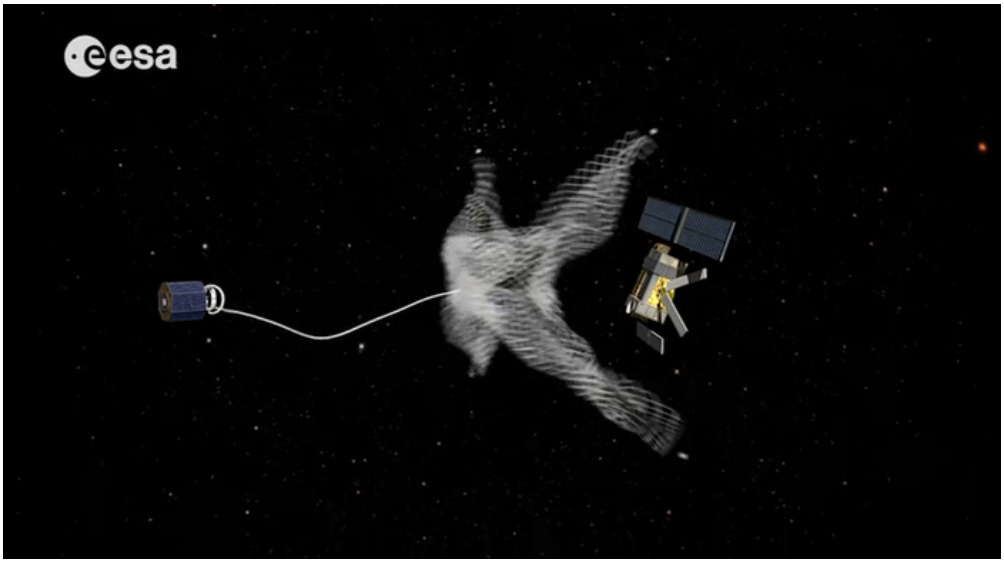


Figure 2 – Illustration of the e.deorbit mission (property of the European Space Agency).

The previously cited missions have at least one phase in which a spacecraft performs a rendezvousing operation. The spacecraft rendezvous consists in a sequence of maneuvers performed by an active follower satellite with the goal of getting closer or even docking to a leader satellite (see Fig. 3).

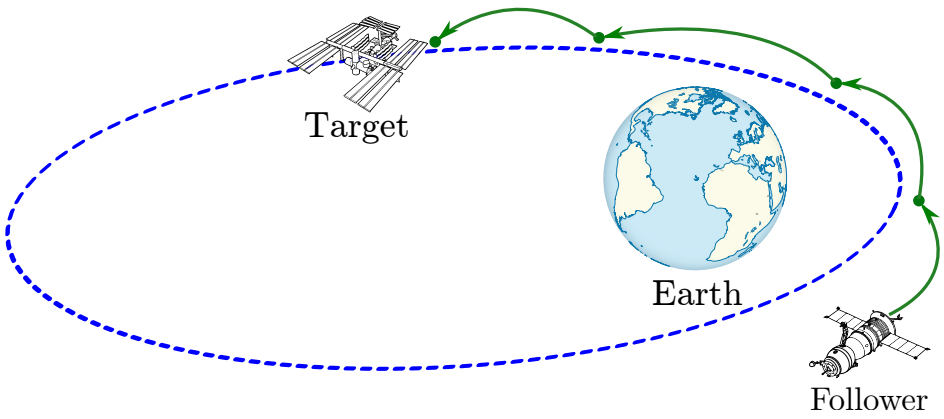


Figure 3 – Rendezvous scheme.

[2] https://www.darpa.mil/program/robotic-servicing-of-geosynchronous-satellites

[3] http://www.esa.int/Our_Activities/Space_Engineering_Technology/Clean_Space/e.Deorbit

These approaching maneuvers are divided into several phases which are defined according to the inter-satellite distance, communication, visibility and other constraints. One of these phases is the so-called *hovering* phase [54, 56–58, 71, 72, 74, 105] (see Fig. 4), in which the follower spacecraft is required to remain in a delimited region of the space relatively to the leader, while the mission control awaits for other events to be accomplished (measurements, synchronization, mission control decisions, visibility, etc).

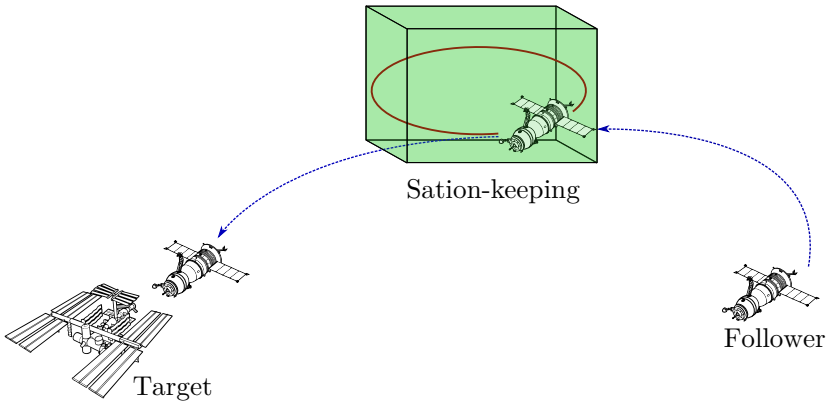


Figure 4 – Hovering phases scheme.

One of the main objectives of this dissertation is the conception of autonomous guidance and control algorithms capable of complying with the complex restrictions of the rendezvous hovering phases, such as the time-continuous space constraints describing the hovering zone and the limitations of the thrusters. Moreover, the computation of the control actions must account for the minimization of fuel consumption, reducing the necessary fuel payload, ensuring feasibility and increasing the lifetime of the missions.

Another major concern of this work is to demonstrate that the proposed algorithms can be efficiently executed on devices dedicated to space applications. For this purpose, the control algorithms presented herein have to comply with the performances of a board containing a FPGA-synthesized LEON3 microprocessor. In fact, although this board is certified for space flights, it lacks computational power when compared to generic commercially available devices. The compilation chains and libraries used in the embedding of these algorithms are also provided.

A final and essential feature investigated in this work is mission safety. In order to ensure the successful mission accomplishment, the numerical results obtained during the execution of the proposed guidance and control algorithms must be validated[4]. The final developments of this thesis focus on the application of validation techniques for obtaining validated bounds for the relative trajectories generated by the control actions computed by the proposed algorithms.

[4] i.e., the accuracy of the results must be verified by an estimation of the committed numerical error.

In the next section, the organization and main contributions of this work are presented.

Organization and contributions

In Chapter 1, the context and assumptions adopted for the hovering phases of the spacecraft orbital rendezvous missions are described and the mathematical formulation of the problem is established. The concepts of synthesis and simulation models are also introduced. The synthesis model is characterized by a structure which is adapted for the conception of control algorithms and model predictive control strategies thanks to the existence of a formal propagation of the relative trajectories. On the other hand, the simulation model provides a high fidelity description of the physical phenomena involved in the orbital motion and, consequently, a more realistic representation of the spacecraft trajectories, being used to simulate the relative motion under the action of guidance algorithms. A first contribution of this work is the implementation of this simulation model in C and on Matlab®/Simulink® software, which led to the release of the two following simulators (freely available on-line):

1. *A Matlab®/Simulink® non-linear simulator for orbital spacecraft rendezvous applications*[5] [4];

2. *A non-linear simulator written in C for orbital spacecraft rendezvous applications*[6] [5].

In the end of Chapter 1, the synthesis model is used in the formulation of the fixed-time impulsive optimal guidance problem for relative motion in the context of the rendezvous hovering zone phases.

In Chapter 2, the theoretical aspects of the resolution of the guidance problem formulated in Chapter 1 are discussed. With the goal of producing efficient optimization-based algorithms, three reformulations of the original problem are then proposed. The first reformulation is based on traditional discretization techniques, leading to a linear program (LP). The second one converts the original problem into a semi-definite program (SDP), using the relation between the cone of non-negative univariate polynomials and the cone of semi-definite positive matrices. The final proposition is an original contribution of this work, which consists in employing a geometrical approach based on the computation of the envelopes of the families of inequalities to describe the set of periodic constrained relative trajectories. This provides a reformulation of the original problem which relies on semi-algebraic functions, leading to a non-smooth optimization problem presented in the article:

[5] https://hal.archives-ouvertes.fr/hal-01413328

[6] https://hal.archives-ouvertes.fr/hal-01410075

3. *Model predictive control for rendezvous hovering phases based on a novel description of constrained trajectories* [7], joint work with M. Joldeş, C. Louembet and F. Camps (research engineer, LAAS-CNRS), published in Proceedings of the 20th IFAC World Congress (IFAC 2017).

Numerical methods for solving each of these optimization problems are given and their practical aspects are considered. These optimization methods are then coded in C and embedded on a board certified for space applications containing a FPGA-synthesized LEON3 microprocessor. The details about the employed libraries and compilation chains were published in the following article:

4. *Embedding an SDP-based control algorithm for the orbital rendezvous hovering phases* [24], joint work with F. Camps, M. Joldeş and C. Louembet, published in Proceedings of the 2018 25th Saint Petersburg International Conference on Integrated Navigation Systems (ICINS 2018).

In Chapter 3 a control strategy is proposed to steer the relative motion and keep it periodic and included in a given hovering zone. This strategy consists of a closed-loop model predictive control (MPC) algorithm, which is proven to make the relative movement converge to the hovering zone even when the presence of saturation constraints on controls may make the space window unreachable from the current state. The performance of this proposed control strategy is assessed via processor-in-the-loop simulations: the control computation is executed on a board containing an FPGA-synthesized LEON3 microprocessor and the propagation of the disturbed relative motion under uncertainties is performed on the previously mentioned simulators. These tests highlight the efficiency of the proposed control strategy in terms of control quality, numerical burden and rejection of disturbances. These original developments were presented in:

5. *Stable Model Predictive Strategy for Rendezvous Hovering Phases Allowing for Control Saturation* [8], joint work with M. Joldeş, C. Louembet and F. Camps, submitted to the 2018 AIAA Journal of Guidance and Control and Dynamics and currently in revision (JGCD 2018).

Chapter 4 is dedicated to the validation of numerical results obtained from the proposed on-board executed algorithms. In fact, during guidance and control procedures of orbiting spacecraft, the respect of positioning and space constraints is decisive for successful missions achievement. Since result accuracy is essential for these procedures, the prevention

and estimation of errors arising from approximations and numerical computations become critical. In this context, a symbolic-numerical method for validating the solutions generated by the guidance and control algorithms is proposed. This approach provides error-bounded polynomial for the solutions of the linear ordinary differential equations (LODE) describing the linearized spacecraft relative motion. These developments led to the article:

6. *Validated Semi-Analytical Transition Matrices for Linearized Relative Spacecraft Dynamics via Chebyshev Series Approximations* [6], joint work with F. Bréhard (PhD student, LAAS-CNRS) and C. Gazzino (post-doctoral researcher, Technion Israel Institute of Technology), published in Proceedings of the 28th Space Flight Mechanics Meeting of the AIAA SciTech Forum (AIAA 2018).

Acknowledgments

This thesis was conducted in the Methods and Algorithms for Control group of the Laboratory for Analysis and Architecture of Systems (LAAS-CNRS), under the supervision of Mioara Joldeş and Christophe Louembet. The developments related to the embedding of algorithms on space dedicated hardware were only possible with the help and technical competences of Frédéric Camps, research engineer at LAAS-CNRS.

The FastRelax (ANR-14-CE25-0018-01) project of the French National Agency for Research (ANR) was the funding source of this work.

Chapter 1

Modeling the problem

Contents

1.1 Introduction

In order to study the rendezvous mission context of a *leader* and a *follower* spacecraft orbiting a main body (the Earth, hereafter), mathematical models must be adopted to describe the relative motion, the actuators and the technological constraints for different purposes. Depending on the level of details and on the assumed hypothesis, several mathematical models can be employed in the description of these phenomena. First, the relative motion can be expressed by different state-space representations (e.g., Cartesian coordinates, orbital elements, equinoctial elements), using nonlinear or linearized dynamics. Then, since the object of study consists of mechanical systems, the significant forces acting over them must be inventoried. Concerning the gravitational forces, for example, the attraction between spacecraft themselves or between the spacecraft and other celestial bodies (Sun, Moon, Jupiter, etc.) can be taken into account or neglected. Other effects such as the Earth's oblateness and non-homogeneous mass distribution, solar pressure or atmospheric drag may be taken into account as intrinsic dynamics or exogenous disturbances to the system. While the follower is usually equipped

with thrusters, the leader spacecraft can be considered active or passive depending on the ability of controlling its inertial orbit. Moreover, the thrusters can be modeled in several different ways, depending on the type of employed propulsion engines and their geometrical disposition. Depending on the rendezvous phase, many type of constraints can be accounted for: hovering zone, visibility cone, safety, orientation, control action dates, thrusters saturation and dead-zone, etc. This chapter aims to set the choices and specify the framework of this thesis.

In the sequel, the context and assumptions adopted for the hovering phases of the spacecraft orbital rendezvous missions are described and the mathematical formulation of the problem is established. First, the concepts of simulation and synthesis models are introduced. The simulation model is more comprehensive with respect to the physical phenomena involved in the orbital motion and, consequently, provides a more realistic representation of the spacecraft trajectories, being used to simulate the relative motion under the action of the conceived guidance algorithms; on the other hand, the synthesis model is less exhaustive and complex and is characterized, in this work, by linearized differential equations, which provides both a formal propagation and a structure adapted to the conception of control algorithms and model predictive control strategies.

The simulation model is obtained by the study of the two-body problem under Keplerian assumptions, which can be described by the Gauss planetary equations [47]. The orbital disturbances, that are not accounted in these models, are introduced as exogenous disturbances and a simulator for the non-linear disturbed relative motion developed in C and on Matlab®/Simulink® is presented.

In order to obtain a synthesis model, the non-linear Tschauner-Hempel equations [100] are linearized [99, 109] and an analysis of the impact of the linearization hypothesis on the evolution of the relative trajectories is carried out. Once a linear true anomaly-varying state-space representation of the relative motion is obtained, a parametrization of the relative trajectories is introduced [29]. This parametrization allows a straightforward characterization of the periodicity property, which is a very desirable feature from the point of view of fuel saving [44]. After that, the hovering region is described as a polytopic rectangular cuboid. The nature and geometrical placement of the propellers are discussed and the equations providing a characterization of the control actions and the fuel consumption are also exhibited. To conclude, the fixed-time impulsive optimal guidance problem for relative motion in the context of the rendezvous hovering zone phases is formulated.

1.2 Relative motion

The objective of this section is to detail the mathematical models that will be further employed in the description of the relative motion between spacecraft throughout this dissertation. In the next developments, the approach synthesis-simulation models is adopted.

The simulation model is used to separately simulate more realistically the behavior of each spacecraft involved in the rendezvous mission via the Gauss planetary equations (12 degrees of freedom, 6 orbital elements for each spacecraft). This model accounts for the intrinsic non-linearities and disturbances of the orbital motion. The relative motions is obtained by performing the passage from orbital elements to inertial states, then computing the difference between the positions and velocities of the leader and the follower spacecraft.

The synthesis model is obtained by performing a parametrization of the state vector employed in the simplified linearized Tschauner-Hempel equations for the relative motion (6 degrees of freedom, 3 relative position and 3 relative velocity coordinates). This variable change allows the conversion of the current relative state into parameters that are directly related to the shape of the relative orbits. This model is applied in the formulation of the fixed-time impulsive optimal guidance problem for the rendezvous hovering phases.

This choice is motivated by the fact that each of the models has advantages that can be exploited separately. The synthesis model is linear and admits a closed-form state-transition matrix that describes the propagation of the relative trajectories departing from a given initial state. This feature is interesting for model predictive control purposes, since it allows the computation of the evolution of the relative trajectory after the application of a control action. Moreover, the fact that this parametrization is related to the shape of the relative orbits leads to a formal description of the periodic relative orbits included in the hovering zone. The simulation model, even though more complex (since it does not admit a closed-from state transition and requires numerical integration), describes with more verisimilitude the motion and significant disturbances acting over each of the spacecraft separately. It provides, in a certain sense, a way of testing the behavior and robustnesses of the control strategies that are conceived and developed with less comprehensive models.

1.2.1 Simulation model

In this subsection, the simulation model is presented. It is obtained by first deducing the equations describing the movement of a single spacecraft orbiting a massive body in a Keplerian framework. Then, the relevant orbital disturbances (the J_2 effect and the atmospheric drag) are included in these equations as exogenous accelerations. The obtained equations are

then employed in the simulation of the dynamics of two spacecraft orbiting the same central body and the relative motion is then obtained by performing the difference between their trajectories. Finally, the obtained model for simulating the relative motion is implemented on C and on Matlab®/Simulink®.

1.2.1.1 Keplerian hypothesis

Consider two bodies in space with homogeneous mass distribution (m_1, m_2) and suppose that the first body is much more massive than the second $m_1 \gg m_2$ (e.g., the Earth and the International Space Station in Fig. 1.1) and that the only forces present in this system are the gravitational attractions (Keplerian motion).

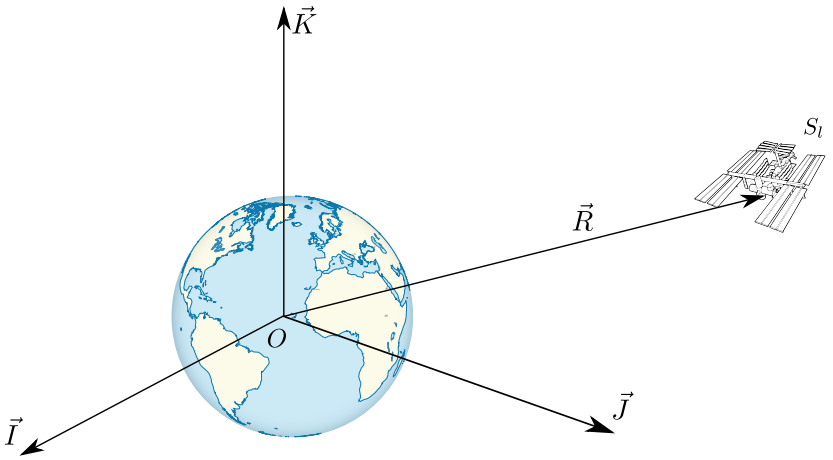


Figure 1.1 – Two bodies in space.

In this case, the influence of the second body on the dynamics of the first one is considered negligible. Take an arbitrary inertial frame fixed on the first body $\mathcal{F}_E = (O, \vec{I}, \vec{J}, \vec{K})$ and let $\vec{R}$ be the position of the second body in this frame. By applying Newton's second law of movement, we obtain:

$$m_2 \frac{d^2}{dt^2}\vec{R}(t) = -\frac{Gm_1m_2}{\|\vec{R}(t)\|^3}\vec{R}(t) \quad \Rightarrow \quad \frac{d^2}{dt^2}\vec{R}(t) = -\frac{\mu}{\|\vec{R}(t)\|^3}\vec{R}(t), \tag{1.1}$$

where G is the universal gravitational constant, $\mu = Gm_1$ is the first body's standard gravitational parameter (e.g., for the Earth, $\mu_\oplus = (398\,600.4405 \pm 0.001)$ km^3 s^{-2} [92]). The general solutions of 1.1 are trajectories that assume the form of conic sections: circles, ellipses, parabolas and hyperbolas (more details in Fig. 1.2 and [13, Chapters 3-4]).

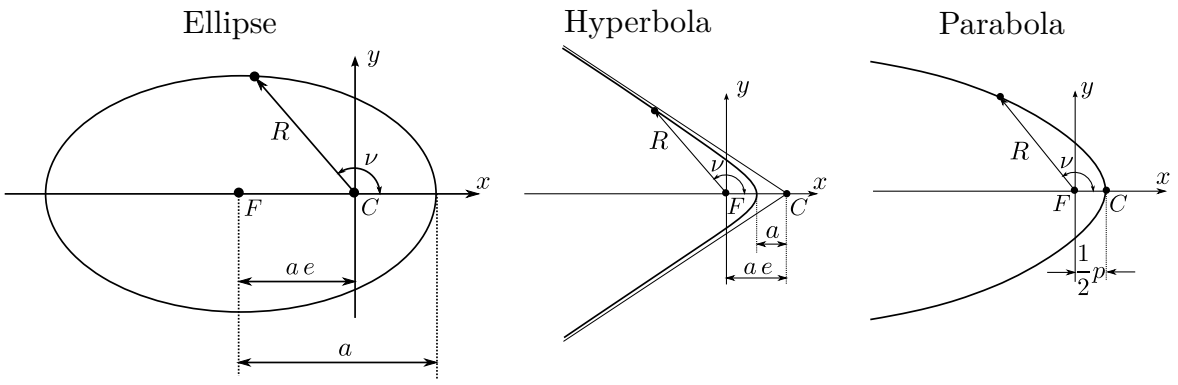


Figure 1.2 – General solutions of the two body problem.

The following equation describes these general orbits in polar coordinates $\nu(t)$ and $R(\nu(t))$:

$$R(t) = \|\vec{R}(t)\| = \frac{p}{1 + e\cos\nu(t)}, \qquad \begin{array}{ll} e = 0, & \text{circular orbit} \\ 0 < e < 1, & \text{elliptical orbit} \\ e = 1, & \text{parabolic orbit} \\ e > 1, & \text{hyperbolic orbit} \end{array} \tag{1.2}$$

where e is the eccentricity, ν is the true anomaly (position of the spacecraft on its orbit) and $p = a(1 - e^2)$ the is the so-called *semilatus rectum*. For the elliptical and circular cases, a corresponds to the semi-major axis of the orbit.

In the sequel, we focus exclusively on the bounded periodic solutions: elliptical (and circular) orbits. In Fig. 1.3 we depict a spacecraft moving in a generic elliptical orbit around Earth.

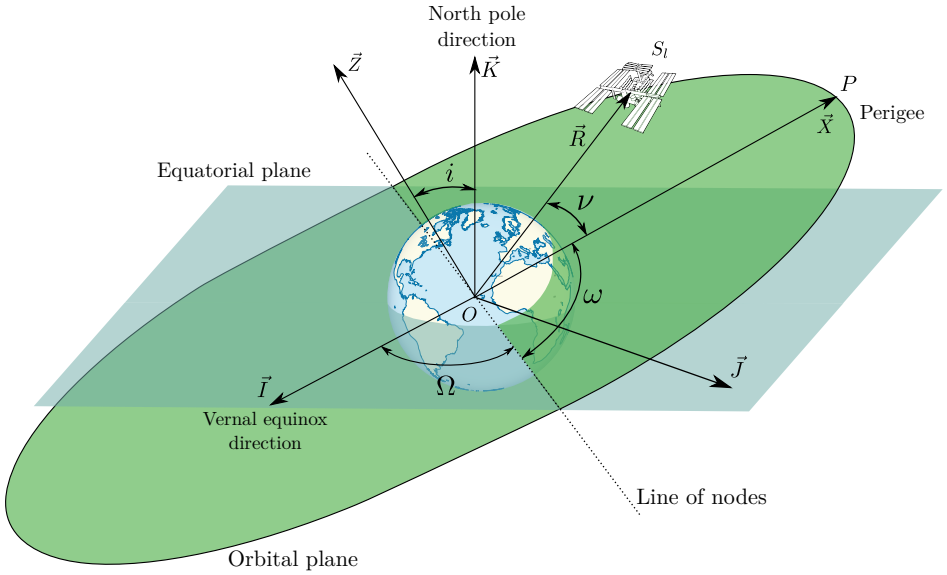


Figure 1.3 – Orbital parameters.

The Earth centered equatorial inertial frame $\mathcal{F}_E = (O, \vec{I}, \vec{J}, \vec{K})$ is such that $\vec{I}$ is the vector

that lies on the equatorial plane and has the direction of the vernal equinox (can be seen as the direction of the vector that goes from the center of the Earth to the center of the Sun when both are located on the equatorial plane), $\vec{K}$ is a vector perpendicular to the equatorial plane, pointing towards the north pole and $\vec{J}$ is a vector that lies on the equatorial plane that completes the orthogonal basis. The Earth centered orbital plane inertial frame $\mathcal{F}_O = (O, \vec{X}, \vec{Y}, \vec{Z})$ is such that $\vec{X}$ is equivalent to $\overrightarrow{OP}$, pointing towards the perigee of the trajectory, $\vec{Z}$ is perpendicular to the orbital plane and $\vec{Y}$ completes the orthogonal basis (omitted in Fig. 1.3).

The rotations allowing the passage from the frame $\mathcal{F}_E$ to $\mathcal{F}_O$ are depicted in Fig. 1.3 and given by:

$$\begin{bmatrix} \vec{X} \\ \vec{Y} \\ \vec{Z} \end{bmatrix} = \underbrace{\begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\omega & -\sin\omega \\ 0 & \sin\omega & \cos\omega \end{bmatrix}}_{\text{rotation of } \omega \text{ around } \vec{K}} \underbrace{\begin{bmatrix} \cos i & -\sin i & 0 \\ \sin i & \cos i & 0 \\ 0 & 0 & 1 \end{bmatrix}}_{\text{rotation of } i \text{ around } \vec{J}} \underbrace{\begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\Omega & -\sin\Omega \\ 0 & \sin\Omega & \cos\Omega \end{bmatrix}}_{\text{rotation of } \Omega \text{ around } \vec{K}} \begin{bmatrix} \vec{I} \\ \vec{J} \\ \vec{K} \end{bmatrix} \tag{1.3}$$

The parameters describing the configuration of the elliptical orbit with respect to the $\mathcal{F}_E$ equatorial inertial frame and the position of the spacecraft on its orbit are the so-called *classical orbital elements* [77]:

$$OE_c = \begin{bmatrix} a, & e, & i, & \Omega, & \omega, & \nu(t) \end{bmatrix}^T.$$

The size and shape of the orbit are given by the *semi-major axis* a and the *eccentricity* e. The *line of nodes* is given by the intersection between the orbital plane and the equatorial plane. The *ascending node* is the orbital position that lies on the line of nodes when the satellite enters the north half-space defined by the equatorial plane separation of the space. The orientation of the orbit with respect to the inertial frame is represented by the *longitude of the ascending node* Ω (the angle between $\vec{I}$ and the ascending node), the *argument of perigee* ω (the angle between the ascending node. and the perigee direction) and the *inclination* i (the angle between the orbital plane and the equatorial plane). Finally, the position of the satellite on its orbit is given by the *true anomaly* ν.

In absence of exogenous forces or disturbances, the free evolution of the position of a spacecraft on its elliptical orbit is expressed via the orbital elements by (1.2):

$$R(t) = \|\vec{R}(t)\| = \frac{a(1-e^2)}{1 + e\cos\nu(t)},$$

where the evolution of the true anomaly is given by the expression of its rate of change (see [37, Chapter 3.2] for further details):

$$\frac{d\nu}{dt} = \sqrt{\frac{\mu}{a^3(1-e^2)^3}}(1+e\cos\nu)^2. \tag{1.4}$$

Evidently, this orbit can also be represented in the $\mathcal{F}_O$ and $\mathcal{F}_E$ frames by simply projecting the vector $\vec{R}(t)$ on the vectors composing their respective bases.

1.2.1.2 Non-Keplerian hypothesis

The previous obtained equations describe the shape of the spacecraft orbit as being a perfect ellipse (1.2). The speed at which a spacecraft travels on its orbit are given by the second and third Kepler's laws (1.4). However, these equations do not account for all the effects acting over a spacecraft in orbit. Among these effect are:

- the atmospheric drag, which consists in the deceleration of the motion in the sense of the along track velocity provoked by the interaction of the spacecraft external area with the particles present in the atmosphere;
- the Earth's gravitational disturbances provoked by its inhomogeneous mass distribution;
- the gravitational pull of the Sun, the Moon and other planets;
- solar radiation pressure, which is the pressure applied on spacecraft's surface provoked by the exchange of momentum between the object and the incoming radiation beam.

These disturbances are included in the spacecraft orbital dynamics as accelerations provoked by exogenous disturbing forces. The evolution of the spacecraft's orbital elements under these accelerations is given by the Gauss planetary equations [47]:

$$\begin{aligned}
\frac{da}{dt} &= \frac{2}{n\sqrt{1-e^2}}\left[-\gamma_z\, e\sin\nu + (1+e\cos\nu)\gamma_x\right] \\
\frac{de}{dt} &= \frac{\sqrt{1-e^2}}{n\,a}\left[-\gamma_z\sin\nu + \left(\cos\nu + \frac{e+\cos\nu}{1+e\,\cos\nu}\right)\gamma_x\right] \\
\frac{di}{dt} &= -\frac{R\cos\theta}{na^2\sqrt{1-e^2}}\,\gamma_y \\
\frac{d\Omega}{dt} &= -\frac{R\sin\theta}{na^2\sqrt{1-e^2}\sin i}\,\gamma_y \\
\frac{d\omega}{dt} &= \frac{\sqrt{1-e^2}}{n\,a\,e}\left[\gamma_z\cos\nu + \left(1+\frac{1}{1+e\cos\nu}\sin\nu\right)\gamma_x\right] + \frac{R\sin\theta\cos i}{n\,a^2\sqrt{1-e^2}\sin i}\gamma_y \\
\frac{d\nu}{dt} &= n\frac{(1+e\cos\nu)^2}{(1-e^2)^{3/2}} + \frac{\sqrt{1-e^2}}{n\,a\,e}\left[-\cos\nu\,\gamma_z - \frac{2+e\cos\nu}{1+e\cos\nu}\,\gamma_x\right],
\end{aligned} \tag{1.5}$$

where $n = \sqrt{\frac{\mu}{a^3}}$, $\theta = \nu + \omega$ and $\vec{\gamma} = [\gamma_x, \gamma_y, \gamma_z]^T$ represents the sum of the accelerations provoked by disturbances in the Local-Vertical/Local-Horizontal frame $LVLH = (S_l, \vec{x}, \vec{y}, \vec{z})$, depicted in Fig. 1.4. Evidently, if $\vec{\gamma} = \vec{0}$, the same motion described by (1.2) and (1.4) is obtained:

$$\frac{da}{dt} = \frac{de}{dt} = \frac{di}{dt} = \frac{d\Omega}{dt} = \frac{d\omega}{dt} = 0, \qquad \frac{d\nu}{dt} = \sqrt{\frac{\mu}{a^3(1-e^2)^3}}(1 + e\cos\nu)^2.$$

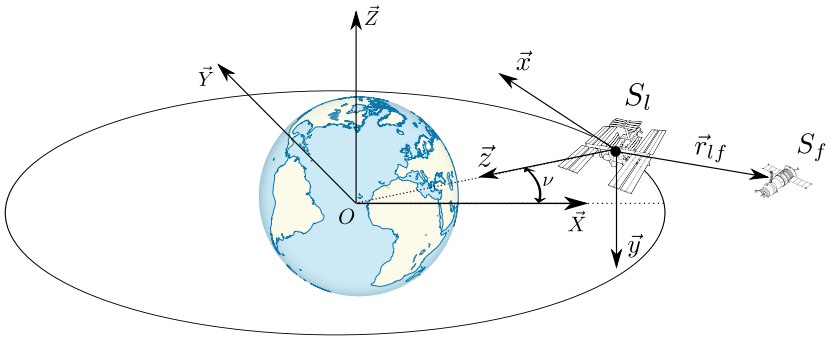


Figure 1.4 – LVLH frame: the z-axis points from the spacecraft to the center of the Earth; the y-axis is normal to the orbital plane, negative in the direction of the angular momentum; the x-axis is mutually perpendicular to the y and z-axes.

The classical orbital elements employed in (1.5) may produce singularities ($e = 0$ or $i = 0$). In order to avoid it, a variable change is performed and the classical orbital elements are converted into the following modified equinoctial orbital elements:

$$OE_{eq} = \Big[p, \quad f, \quad g, \quad h, \quad k, \quad L\Big]^T, \tag{1.6}$$

which can be obtained from the classical orbital elements via:

$$\begin{aligned} p &= a(1-e^2) & h &= \tan(i/2)\cos\Omega, \\ f &= e\cos(\Omega+\omega), & k &= \tan(i/2)\sin\Omega, \\ g &= e\sin(\Omega+\omega), & L &= \Omega+\omega+\nu. \end{aligned} \tag{1.7}$$

The Gauss planetary equations (1.5) can then be rewritten using the equinoctial orbital elements, which produces the following equations [104]:

$$\frac{dOE_{eq}}{dt} = A_{eq}\vec{\gamma} + B_{eq} \tag{1.8}$$

where:

$$A_{eq} = \sqrt{\frac{p}{\mu}} \begin{bmatrix} \frac{2p}{w} & 0 & 0 \\ \frac{(w+1)\cos L + f}{w} & \frac{g(h\sin L - k\cos L)}{w} & -\sin L \\ \frac{(w+1)\sin L + g}{w} & -\frac{f(h\sin L - k\cos L)}{w} & \cos L \\ 0 & -\frac{s^2\cos L}{2w} & 0 \\ 0 & -\frac{s^2\sin L}{2w} & 0 \\ 0 & -\frac{h\sin L - k\cos L}{w} & 0, \end{bmatrix} \tag{1.9}$$

$$B_{eq} = \begin{bmatrix} 0, & 0, & 0, & 0, & 0, & \frac{w^2}{p^2}\sqrt{\mu p} \end{bmatrix}^T, \tag{1.10}$$

where $w = 1 + f\cos L + g\sin L$.

In [34, Section 16.4], the author demonstrates that for flights at height of a few thousands of kilometers and less above the surface of the Earth, the perturbations related to the Moon and Sun pull are insignificant compared to the gravity anomalies and the second zonal harmonic of the geopotential (the so-called J_2 effect). Also in [34, Section 17.3], it is shown that the solar radiation pressure is relevant only for small light-weight satellites orbiting at flight heights above 500 km. For all other satellites the perturbations produced by radiation pressure are small compared to the other disturbing effects. For the simulations performed throughout this dissertation, missions based on the International Space Station orbital parameters are employed [80] and, given that its flight height is below 500 km[1], only the J_2 effect and the atmospheric drag will be taken into account in the following developments:

- J_2 disturbance: the acceleration provoked by the Earth's flatness is given in the LVLH frame by [10, 60]:

$$\vec{\gamma}_{J_2} = -\frac{3\mu J_2 R_e^2}{2R^4} \begin{bmatrix} \frac{8(h\sin L - k\cos L)(h\cos L + k\sin L)}{(1+h^2+k^2)^2} \\ -\frac{4(h\sin L - k\cos L)(1-h^2-k^2)}{(1+h^2+k^2)^2} \\ \frac{12(h\sin L - k\cos L)^2}{(1+h^2+k^2)^2} - 1 \end{bmatrix} \tag{1.11}$$

[1] https://www.heavens-above.com/IssHeight.aspx

where J_2 is the second degree term in Earth's gravity potential and R_e is the Earth's radius.

- Atmospheric drag: the disturbing acceleration provoked by the atmospheric drag is given in the LVLH frame by [10, 60]:

$$\vec{\gamma}_d = -\frac{\rho(R)SC_d}{2m}\frac{\mu}{p}\sqrt{1+2(g\sin L + f\cos L)+f^2+g^2}\begin{bmatrix}1+f\cos L+g\sin L\\0\\-f\sin L+g\cos L\end{bmatrix}, \quad (1.12)$$

where $\rho(R)$ is the atmospheric density and m, S and C_d are respectively the mass, the cross sectional area and the drag coefficient of the spacecraft. The atmospheric density is given in function of the distance between the satellite and the center of the Earth by the following equation:

$$\rho(R) = \bar{\rho}\,\exp\left(\frac{R_e - R + 400000}{46830}\right), \quad (1.13)$$

where $\bar{\rho}$ is a constant that depends of the solar activity (2.2644×10^{-12} for low and 3.5475×10^{-11} for high solar activity).

1.2.1.3 Relative motion

In order to obtain the relative motion, the differential equations (1.8) are integrated for both leader and follower spacecraft independently, leading to a 12 degree of freedom model. At each integration step, the equinoctial orbital elements can be converted into the Cartesian inertial position and velocity via the following transformation:

$$\begin{aligned}
x_{\mathcal{F}_E} &= \frac{r}{s^2}(\cos L + \alpha^2\cos L + 2hk\sin L)\\
y_{\mathcal{F}_E} &= \frac{r}{s^2}(\sin L - \alpha^2\sin L + 2hk\cos L)\\
z_{\mathcal{F}_E} &= \frac{2r}{s^2}(h\sin L - k\cos L)\\
\dot{x}_{\mathcal{F}_E} &= -\frac{1}{s^2}\sqrt{\frac{\mu}{p}}(\sin L + \alpha^2\sin L - 2hk\cos L + g - 2fhk + \alpha^2 g)\\
\dot{y}_{\mathcal{F}_E} &= -\frac{1}{s^2}\sqrt{\frac{\mu}{p}}(-\cos L + \alpha^2\cos L - 2hk\sin L - f + 2ghk + \alpha^2 f)\\
\dot{z}_{\mathcal{F}_E} &= \frac{2}{s^2}\sqrt{\frac{\mu}{p}}(h\cos L + k\sin L + fh + gk),
\end{aligned} \quad (1.14)$$

where $r = p/w$, $s^2 = 1 + h^2 + k^2$, and $\alpha^2 = h^2 - k^2$.

Then, the vector $X = [x, y, z, \dot{x}, \dot{y}, \dot{z}]^T$ representing the difference between the spacecraft's positions and velocities projected on the leader's LVLH frame is given by the following relations:

$$\begin{bmatrix} x \\ y \\ z \\ \dot{x} \\ \dot{y} \\ \dot{z} \end{bmatrix} = \begin{bmatrix} \langle \vec{x}^l, \vec{p}_{\mathcal{F}_E} \rangle \\ -\langle \vec{y}^l, \vec{p}_{\mathcal{F}_E} \rangle \\ \langle \vec{z}^l, \vec{p}_{\mathcal{F}_E} \rangle \\ \langle \vec{x}^l, \vec{v}_{\mathcal{F}_E} \rangle + \dot{\nu}^l \langle \vec{z}^l, \vec{p}_{\mathcal{F}_E} \rangle \\ -\langle \vec{y}^l, \vec{v}_{\mathcal{F}_E} \rangle \\ \langle \vec{z}^l, \vec{v}_{\mathcal{F}_E} \rangle - \dot{\nu}^l \langle \vec{x}^l, \vec{p}_{\mathcal{F}_E} \rangle \end{bmatrix} \tag{1.15}$$

where $\langle \cdot, \cdot \rangle$ is the dot product in $\mathbb{R}^3$,

$$\vec{p}_{\mathcal{F}_E} = (x^f_{\mathcal{F}_E} - x^l_{\mathcal{F}_E}, y^f_{\mathcal{F}_E} - y^l_{\mathcal{F}_E}, z^f_{\mathcal{F}_E} - z^l_{\mathcal{F}_E})$$

is the difference between the positions of the follower and leader spacecraft in the inertial frame,

$$\vec{v}_{\mathcal{F}_E} = (\dot{x}^f_{\mathcal{F}_E} - \dot{x}^l_{\mathcal{F}_E}, \dot{y}^f_{\mathcal{F}_E} - \dot{y}^l_{\mathcal{F}_E}, \dot{z}^f_{\mathcal{F}_E} - \dot{z}^l_{\mathcal{F}_E})$$

is the difference between the velocities of the follower and leader spacecraft in the inertial frame, $\dot{\nu}^l$ is the derivative with respect to time of the leader's true anomaly and the unitary vectors $\vec{x}^l$, $\vec{y}^l$, $\vec{z}^l$ compose the leader's LVLH frame orthonormal basis and are given by the following expressions:

$$\begin{aligned} \vec{z}^l &= -\frac{(x^l_{\mathcal{F}_E}, y^l_{\mathcal{F}_E}, z^l_{\mathcal{F}_E})}{\sqrt{(x^l_{\mathcal{F}_E})^2 + (y^l_{\mathcal{F}_E})^2 + (z^l_{\mathcal{F}_E})^2}}, \\ \vec{x}^l &= \frac{\vec{V}}{\|\vec{V}\|}, \\ \vec{y}^l &= -\frac{\vec{U}}{\|\vec{U}\|}, \end{aligned} \qquad \text{where} \quad \begin{aligned} \vec{T} &= \frac{(\dot{x}^l_{\mathcal{F}_E}, \dot{y}^l_{\mathrm{I}}, \dot{z}^l_{\mathcal{F}_E})}{\sqrt{(\dot{x}^l_{\mathcal{F}_E})^2 + (\dot{y}^l_{\mathcal{F}_E})^2 + (\dot{z}^l_{\mathcal{F}_E})^2}} \\ \vec{U} &= \vec{T} \times \vec{z}^l \\ \vec{V} &= -\vec{U} \times \vec{z}^l \end{aligned} \tag{1.16}$$

A simulator for the nonlinear disturbed movement based on the previously presented equations (1.8), (1.11), (1.14) and (1.15) was developed on Matlab®/Simulink® and in C. The Matlab®/Simulink® simulator is a modified version of the one proposed by Mounir

Kara-Zaitri in his PhD thesis [60, Chapter 4]. Some adjustments were performed in order to obtain a dedicated tool for simulating and developing control algorithms for the orbital spacecraft rendezvous in the case where the leader spacecraft is passive and the control applied on follower spacecraft is originally computed on the leader LVLH frame. On the other hand, the C version of the simulator is an original contribution of this work. For a given orbital rendezvous scenario, the output of both simulators is the evolution of the relative position and velocity between the two spacecraft, obtained by the integration of the Gauss equations for the orbital motion under the disturbances provoked by the Earth's flatness (the J_2 effect) and the atmospheric drag. These simulators are available at:

- C version: https://hal.archives-ouvertes.fr/hal-01410075 [5].
- Matlab®/Simulink®: https://hal.archives-ouvertes.fr/hal-01413328 [4];

This simulator will be employed in the remainder of this dissertation for assessing the behavior of the proposed control algorithms in a non-linear and disturbed context.

1.2.2 Synthesis model

In this section, the model representing the relative dynamics between spacecraft that will be used throughout this dissertation for the development of control algorithms is presented. Several representations can be used for this purpose. For example, in the literature the differential orbital elements or modified versions of them are employed in formation flight applications [2, 19]. Hereafter, the Cartesian local relative coordinates are chosen (6 degrees of freedom, 3 position and 3 velocity coordinates) for the initial study of the relative motion instead of the differential orbital elements. This choice is preferred for modeling problems in which space restriction are present [31, 50, 52], just as the rendezvous hovering phases problem. Nevertheless, later in this section, a parametrization of these Cartesian coordinates is introduced exhibiting an intrinsic relation between the obtained parameters and the shapes and boundedness of the relative orbits.

In the sequel, the equations describing the evolution of the relative motion between two spacecraft on elliptical orbits are presented (the reader may consult Appendix A for more details about the deduction of these equations). Let be $X(t) = [x,\ y,\ z,\ \dot{x},\ \dot{y},\ \dot{z}]^T$ the state vector containing the relative positions and velocities in the leader's Local-Vertical/Local-Horizontal frame $LVLH = (S_l, \vec{x}, \vec{y}, \vec{z})$ (see Fig. 1.5).

By applying Newton's Second Law of motion to both spacecraft and subtracting the dynamics of the leader from the follower spacecraft, the following system of nonlinear differential

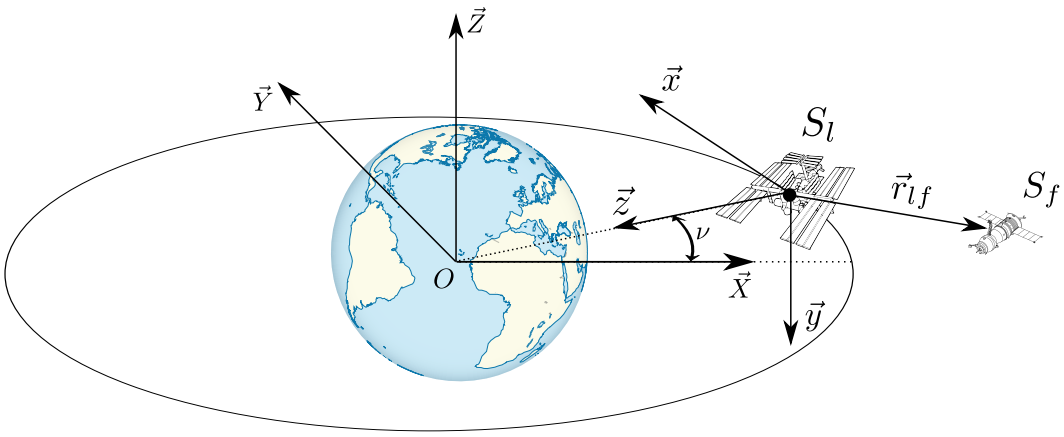


Figure 1.5 – LVLH frame and rendezvous scheme.

equations is obtained [100]:

$$\begin{aligned}
\ddot{x} &= 2\dot{\nu}\dot{z} + \ddot{\nu}z + \dot{\nu}^2 x - \frac{\mu x}{\sqrt{(x^2+y^2+(R-z)^2)^3}} \\
\ddot{y} &= -\frac{\mu y}{\sqrt{(x^2+y^2+(R-z)^2)^3}} \\
\ddot{z} &= -2\dot{\nu}\dot{x} - \ddot{\nu}x + \dot{\nu}^2 z - \frac{\mu(z-R)}{\sqrt{(x^2+y^2+(R-z)^2)^3}} - \frac{\mu}{R^2},
\end{aligned} \tag{NLTH}$$

These are the so-called nonlinear Tschauner-Hempel equations. Assuming as linearization hypothesis that the distance between spacecraft is much smaller that the distance from the leader spacecraft to the center of the Earth:

$$\sqrt{x^2+y^2+z^2} \ll R, \tag{1.17}$$

the linearized Tschauner-Hempel equations are obtained:

$$\begin{aligned}
\ddot{x} &= 2\dot{\nu}\dot{z} + \ddot{\nu}z + \dot{\nu}^2 x - \frac{\mu}{R^3}x \\
\ddot{y} &= -\frac{\mu}{R^3}y \\
\ddot{z} &= -2\dot{\nu}\dot{x} - \ddot{\nu}x + \dot{\nu}^2 z + \frac{2\mu}{R^3}z.
\end{aligned} \tag{LTH}$$

Let be $\tilde{X}(\nu) = [\tilde{x},\ \tilde{y},\ \tilde{z},\ \tilde{x}',\ \tilde{y}',\ \tilde{z}']^T$ a new state vector such that:

$$\tilde{X}(\nu) = T(\nu)X(t), \qquad X(t) = T^{-1}(\nu)\tilde{X}(\nu), \tag{1.18}$$

where the similarity transformation $T(\nu)$ is given by:

$$T(\nu) = \begin{bmatrix} \rho_\nu I_3 & 0_3 \\ -es_\nu I_3 & (k^2\rho_\nu)^{-1} I_3 \end{bmatrix}.$$

with $k^2 = \sqrt{\frac{\mu}{a^3(1-e^2)^3}}$, $s_\nu = \sin(\nu)$ and $\rho_\nu = (1 + e\,\cos\nu)$.

By introducing these new variables in (LTH), the so-called simplified linearized Tschauner-Hempel equations [99, 109] are obtained (for details, see Appendices A.2 and A.3):

$$\begin{aligned} \tilde{x}'' &= 2\tilde{z}', \\ \tilde{y}'' &= -y, \\ \tilde{z}'' &= \frac{3}{\rho_\nu}\tilde{z} - 2\tilde{x}', \end{aligned} \tag{SLTH}$$

where $(\cdot)' = \frac{d(\cdot)}{d\nu}$, $(\cdot)'' = \frac{d^2(\cdot)}{d\nu^2}$, $\frac{d(\cdot)}{dt} = (\cdot)'\dot{\nu}$, $\frac{d^2(\cdot)}{dt^2} = (\cdot)''\dot{\nu}^2 + (\cdot)'\ddot{\nu}$.

1.2.2.1 Evaluating the linearization hypothesis

In this section, the validity of the linearization hypothesis (1.17) is assessed. For that, let there be an initial true anomaly $\nu_0 = 0^\circ$ and an initial relative state $X_0(n) = [10^n,\ 10^n,\ 10^n,\ 0,\ 0,\ 0]$. For $a = \{7 \times 10^6,\ 8 \times 10^6\}$ meters, $e = \{0.04,\ 0.1\}$ and $n = \{1,\ 2,\ 3\}$, the evolution of the relative trajectory is propagated during $T_{\text{sim}} = \{200,\ 1500\}$ seconds, considering the nonlinear (NLTH) and the linearized (LTH) relative dynamics[2]. The absolute difference between the final states is presented in Table 1.1 .

From Table 1.1 it is possible to remark that the absolute difference between the nonlinear and the linearized dynamics increases as the simulation time, the initial relative distance and the eccentricity increase; on the other hand, higher values of the semi-major axis result in higher distances between the leader spacecraft and the Earth, which makes the linearization hypothesis stronger and results in lower mismatches. For short simulation times (200 s), the committed errors are of the order of the centimeters for all configurations of initial states and eccentricities; however, for longer simulations (1500 s), the error can escalate to values equivalent to the Earth radius (see Fig. 1.6).

As a consequence of this, for certain scenarios, the relative trajectories obtained via linear propagation may present immense discrepancies with respect to those obtained via nonlinear propagation. In order to avoid these inconsistencies, control laws and theoretical studies synthesized using models based on the linearized Tschauner-Hempel equations should also be

[2]The MATLAB® function `ode45` is employed, with options `RelTol` $= 10^{-6}$ and `AbsTol` $= 10^{-6}$

Table 1.1 – Evaluating the linearization hypothesis: difference between nonlinear and linearized Tschauner-Hempel equations.

T_{sim}	a	e	n	$\lvert\Delta x_f\rvert$	$\lvert\Delta y_f\rvert$	$\lvert\Delta z_f\rvert$	$\lvert\Delta \dot{x}_f\rvert$	$\lvert\Delta \dot{y}_f\rvert$	$\lvert\Delta \dot{z}_f\rvert$
200	7e+6	0.04	1	1.17e−6	1.19e−6	2.22e−7	1.17e−8	1.22e−8	3.59e−9
-	-	-	2	1.17e−4	1.19e−4	2.22e−5	1.17e−6	1.22e−6	3.59e−7
-	-	-	3	1.17e−2	1.19e−2	2.22e−3	1.17e−4	1.22e−4	3.59e−5
-	-	0.1	1	1.52e−6	1.59e−6	3.44e−7	1.53e−8	1.69e−8	5.82e−9
-	-	-	2	1.52e−4	1.59e−4	3.44e−5	1.53e−6	1.69e−6	5.82e−7
-	-	-	3	1.52e−2	1.59e−2	3.43e−3	1.53e−4	1.69e−4	5.81e−5
-	8e+6	0.04	1	6.87e−7	6.96e−7	1.03e−7	6.87e−9	7.08e−9	1.63e−9
-	-	-	2	6.87e−5	6.96e−5	1.03e−5	6.87e−7	7.08e−7	1.63e−7
-	-	-	3	6.87e−3	6.96e−3	1.03e−3	6.87e−5	7.08e−5	1.63e−5
-	-	0.1	1	8.91e−7	9.18e−7	1.56e−7	8.96e−9	9.59e−9	2.56e−9
-	-	-	2	8.91e−5	9.18e−5	1.56e−5	8.96e−7	9.59e−7	2.56e−7
-	-	-	3	8.92e−3	9.18e−3	1.56e−3	8.96e−5	9.60e−5	2.55e−5
1500	7e+6	0.04	1	6.77e−2	1.71e−4	1.31e−1	2.51e−3	7.65e−7	1.46e−3
-	-	-	2	6.67e+0	1.71e−2	1.14e+1	2.51e−1	7.65e−5	1.51e−1
-	-	-	3	5.43e+2	1.65e+0	1.29e+3	2.48e+1	7.56e−3	2.07e+1
-	-	0.1	1	1.45e+2	9.09e−3	2.90e+3	1.16e+1	1.19e−4	4.66e+1
-	-	-	2	3.72e+4	4.79e−1	1.23e+5	2.89e+2	3.83e−3	1.90e+3
-	-	-	3	1.14e+6	5.01e+1	9.45e+5	5.19e+3	1.50e−1	1.25e+4
-	8e+6	0.04	1	6.48e−3	7.18e−5	1.55e−2	7.96e−5	4.75e−7	1.76e−5
-	-	-	2	6.46e−1	7.16e−3	1.55e+0	7.98e−3	4.75e−5	1.69e−3
-	-	-	3	6.29e+1	6.96e−1	1.56e+2	8.18e−1	4.67e−3	9.78e−2
-	-	0.1	1	2.57e+1	1.28e−3	9.44e+0	7.70e−1	1.78e−6	2.77e−1
-	-	-	2	2.30e+3	1.34e−1	5.85e+2	5.77e+1	2.69e−4	2.64e+1
-	-	-	3	1.53e+5	2.25e+1	5.11e+3	2.20e+3	7.34e−2	5.66e+2

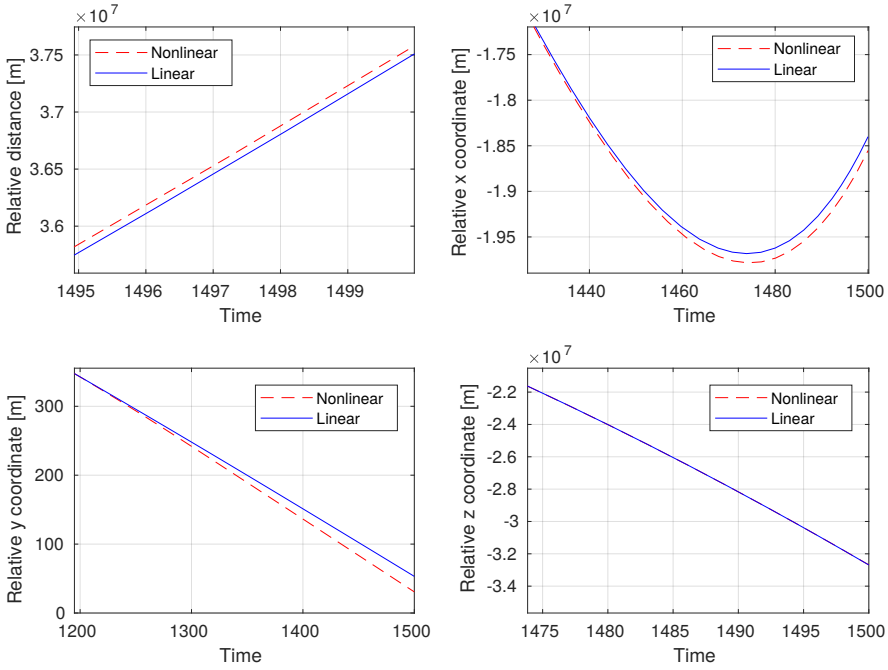


Figure 1.6 – Linear and nonlinear simulations ($T_{sim} = 1500$, $a = 8e{+}6$, $e = 0.1$, $n = 3$).

tested on models based on the nonlinear dynamics. This conclusion corroborates the adopted strategy of using a synthesis model for conception of control algorithms and a simulation model for their "validation".

1.2.2.2 State-transition matrix

The linear true anomaly-varying system of equations (SLTH) can be expressed under the following state-space representation:

$$\tilde{X}'(\nu) = \underbrace{\begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 2 \\ 0 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & \dfrac{3}{\rho_\nu} & -2 & 0 & 0 \end{bmatrix}}_{\tilde{A}(\nu)} \tilde{X}(\nu) \tag{1.19}$$

Yamanaka and Ankersen propose in [109] a fundamental solution matrix ($\varphi(\nu) \in \mathbb{R}^{6\times 6}$ non-singular, such that $\varphi(\nu)' = \tilde{A}(\nu)\varphi(\nu)$) for this system:

$$\varphi(\nu) = \begin{bmatrix} 1 & 0 & -c_\nu(1+\rho_\nu) & s_\nu(1+\rho_\nu) & 0 & 3\rho_\nu^2 J_{\nu_0}(\nu) \\ 0 & c_\nu & 0 & 0 & s_\nu & 0 \\ 0 & 0 & s_\nu\rho_\nu & c_\nu\rho_\nu & 0 & 2 - 3es_\nu\rho_\nu J_{\nu_0}(\nu) \\ 0 & 0 & 2s_\nu\rho_\nu & 2c_\nu\rho_\nu - e & 0 & 3 - 6es_\nu\rho_\nu J_{\nu_0}(\nu) \\ 0 & -s_\nu & 0 & 0 & c_\nu & 0 \\ 0 & 0 & c_\nu + ec_{2\nu} & -s_\nu - es_{2\nu} & 0 & -3e\left((c_\nu + ec_{2\nu})J_{\nu_0}(\nu) + \dfrac{s_\nu}{\rho_\nu}\right) \end{bmatrix}, \tag{1.20}$$

where $s_\nu = \sin\nu$ and $c_\nu = \cos\nu$,

$$J_{\nu_0}(\nu) := \int_{\nu_0}^{\nu} \frac{d\tau}{\rho(\tau)^2} = \sqrt{\frac{\mu}{a^3}} \frac{t - t_0}{(1-e^2)^{3/2}}, \tag{1.21}$$

and ν_0 is an arbitrary initial true anomaly of reference.

The propagation of an initial trajectory $\tilde{X}(\nu_0)$ can be performed via the state-transition matrix $\Phi(\nu, \nu_0) = \varphi(\nu)\varphi^{-1}(\nu_0)$:

$$\tilde{X}(\nu) = \Phi(\nu, \nu_0)\tilde{X}(\nu_0), \tag{1.22}$$

where $\varphi^{-1}(\nu_0)$ is given by:

$$\varphi^{-1}(\nu_0) = \begin{bmatrix} 1 & 0 & -\frac{3es_{\nu_0}(1+\rho_{\nu_0})}{\rho_{\nu_0}(e^2-1)} & \frac{es_{\nu_0}(1+\rho_{\nu_0})}{e^2-1} & 0 & \frac{ec_{\nu_0}\rho_{\nu_0}-2}{e^2-1} \\ 0 & c_{\nu_0} & 0 & 0 & -s_{\nu_0} & 0 \\ 0 & 0 & \frac{3s_{\nu_0}(\rho_{\nu_0}+e^2)}{\rho_{\nu_0}(e^2-1)} & -\frac{s_{\nu_0}(1+\rho_{\nu_0})}{e^2-1} & 0 & \frac{2e-c_{\nu_0}\rho_{\nu_0}}{e^2-1} \\ 0 & 0 & \frac{3(e+c_{\nu_0})}{e^2-1} & -\frac{2c_{\nu_0}+ec_{\nu_0}^2+e}{e^2-1} & 0 & \frac{s_{\nu_0}\rho_{\nu_0}}{e^2-1} \\ 0 & s_{\nu_0} & 0 & 0 & c_{\nu_0} & 0 \\ 0 & 0 & -\frac{3ec_{\nu_0}+e^2+2}{e^2-1} & \frac{\rho_{\nu_0}^2}{e^2-1} & 0 & -\frac{es_{\nu_0}\rho_{\nu_0}}{e^2-1} \end{bmatrix}. \tag{1.23}$$

1.2.3 Deaconu's parametrization and periodic relative trajectories

In this section, a transformation allowing the description of the relative state by a vector of parameters is introduced. This parametrization is demonstrated to be in a half-way between the representation of relative trajectories via Cartesian coordinates and via orbital elements, since it provides both a representation of the relative position and velocity between spacecraft and an interpretation of the shape and boundedness of these relative orbits.

In [28, Chapter 2], Deaconu remarked that the term $\varphi^{-1}(\nu_0)\tilde{X}(\nu_0)$ in (1.22) is a constant that only depends on the evaluation of φ and $\tilde{X}$ at ν_0. Inspired by this observation, the author proposed the following variable change:

$$D(\nu) = \underbrace{\begin{bmatrix} 0 & 0 & -\frac{3\,ec_\nu+e^2+2}{e^2-1} & \frac{\rho_\nu^2}{e^2-1} & 0 & -\frac{es_\nu\rho_\nu}{e^2-1} \\ 0 & 0 & \frac{3(e+c_\nu)}{e^2-1} & -\frac{2c_\nu+ec_\nu^2+e}{e^2-1} & 0 & \frac{s_\nu\rho_\nu}{e^2-1} \\ 0 & 0 & \frac{3s_\nu(\rho_\nu+e^2)}{\rho_\nu(e^2-1)} & -\frac{s_\nu(1+\rho_\nu)}{e^2-1} & 0 & \frac{2e-c_\nu\rho_\nu}{e^2-1} \\ 1 & 0 & -\frac{3\,es_\nu(1+\rho_\nu)}{\rho_\nu(e^2-1)} & \frac{es_\nu(1+\rho_\nu)}{e^2-1} & 0 & \frac{ec_\nu\rho_\nu-2}{e^2-1} \\ 0 & c_\nu & 0 & 0 & -s_\nu & 0 \\ 0 & s_\nu & 0 & 0 & c_\nu & 0 \end{bmatrix}}_{C(\nu)} \tilde{X}(\nu), \tag{1.24}$$

where $D(\nu_0) = [d_0(\nu_0),\ d_1(\nu_0),\ d_2(\nu_0),\ d_3(\nu_0),\ d_4(\nu_0),\ d_5(\nu_0)]^T$ and $C(\nu)$ is equivalent to $\varphi^{-1}(\nu)$, but with some lines permuted: $1 \to 4$, $2 \to 5$, $4 \to 2$, $5 \to 6$ and $6 \to 1$. Although this variable change may resemble like a mere replacement of the constant term $\varphi^{-1}(\nu_0)\tilde{X}(\nu_0)$, the parameterization of the trajectories $\tilde{X}(\nu)$ by the vector $D(\nu)$ brings out many simplifications and advantages on the modeling of the problem:

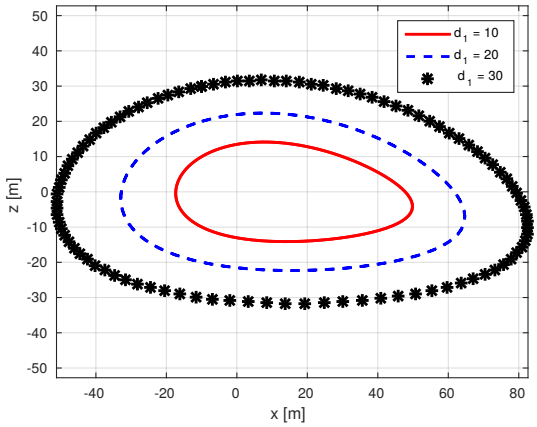
1. **The entries of the vector $D(\nu)$ describe the shape and size of the trajectory:**

by replacing $\varphi^{-1}(\nu_0)\tilde{X}(\nu_0)$ by $D(\nu_0)$ in (1.22), the following equations are obtained:

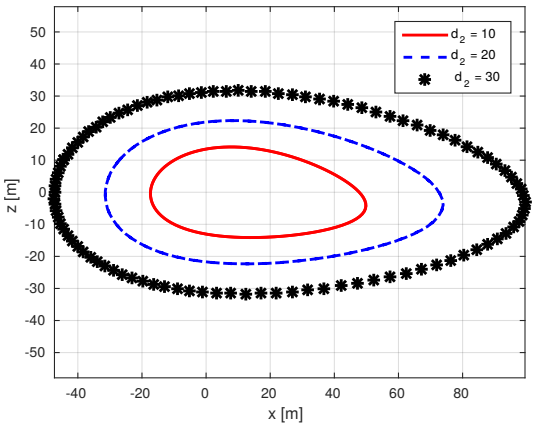
$$\begin{aligned}
\tilde{x}(\nu) &= (2 + e\,c_\nu)(d_1(\nu_0)\,s_\nu - d_2(\nu_0)\,c_\nu) + d_3(\nu_0) + 3\,(1 + e\,c_\nu)^2\,d_0(\nu_0)\,J_{\nu_0}(\nu), \\
\tilde{y}(\nu) &= d_4(\nu_0)\,c_\nu + d_5(\nu_0)\,s_\nu, \\
\tilde{z}(\nu) &= (1 + e\,c_\nu)(d_2(\nu_0)\,s_\nu + d_1(\nu_0)\,c_\nu - 3\,e\,s_\nu\,d_0(\nu_0)\,J_{\nu_0}(\nu)) + 2\,d_0(\nu_0).
\end{aligned} \tag{1.25}$$

and, as one can remark, the transition of the states $\tilde{x}(\nu)$, $\tilde{y}(\nu)$ and $\tilde{z}(\nu)$ depends linearly on $D(\nu_0)$.

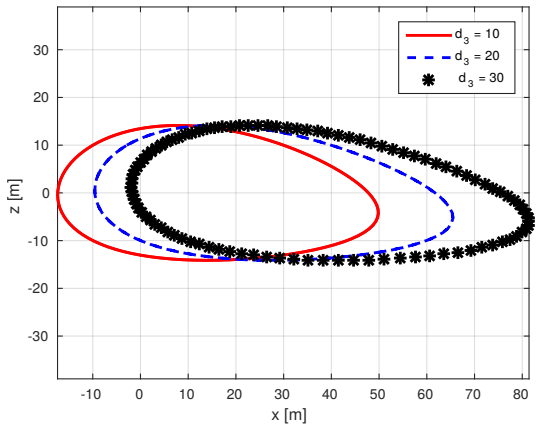
In Fig. 1.7 and Fig. 1.8, the shape of the relative trajectory associated to vector $D(\nu) = [0,\ 10,\ 10,\ 10,\ 10,\ 10]^T$ for an eccentricity $e = 0.4$ is illustrated by the continuous red line. In each figure, the other two relative trajectories in blue dashed line and black stars are obtained by changing one parameter of D at time.



(a) Impact of d_1 on the shape of the trajectory.



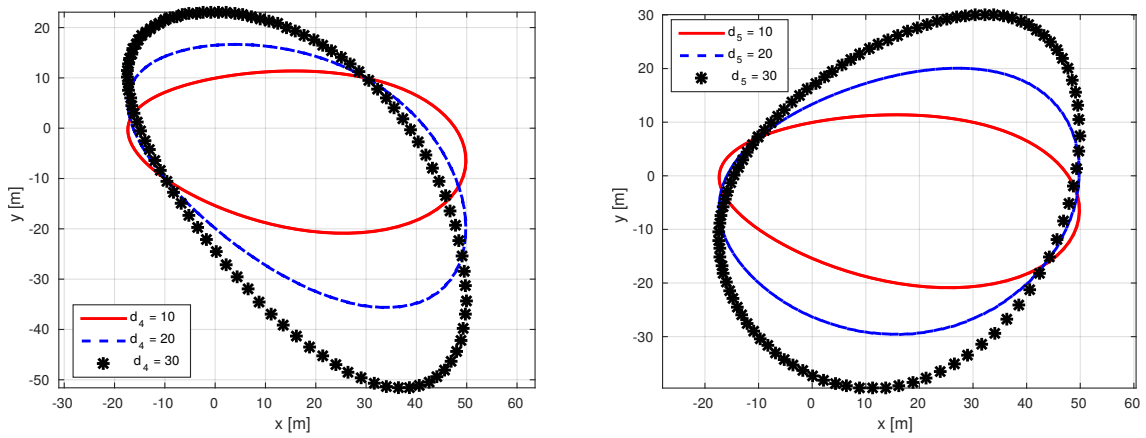
(b) Impact of d_2 on the shape of the trajectory.



(c) Impact of d_3 on the shape of the trajectory.

Figure 1.7 – Impact of d_1, d_2 and d_3 on the shape of the trajectory, $d_0 = 0$ and $e = 0.4$.

From (1.24) and from Fig. 1.7 and Fig. 1.8, one can observe that the first four entries of D (d_0, d_1, d_2 and d_3) characterize the shape of the trajectory in the XZ-plane and, the las two entries (d_4 and d_5), the shape of the Y-axis motion.



(a) Impact of d_4 on the shape of the trajectory.

(b) Impact of d_5 on the shape of the trajectory.

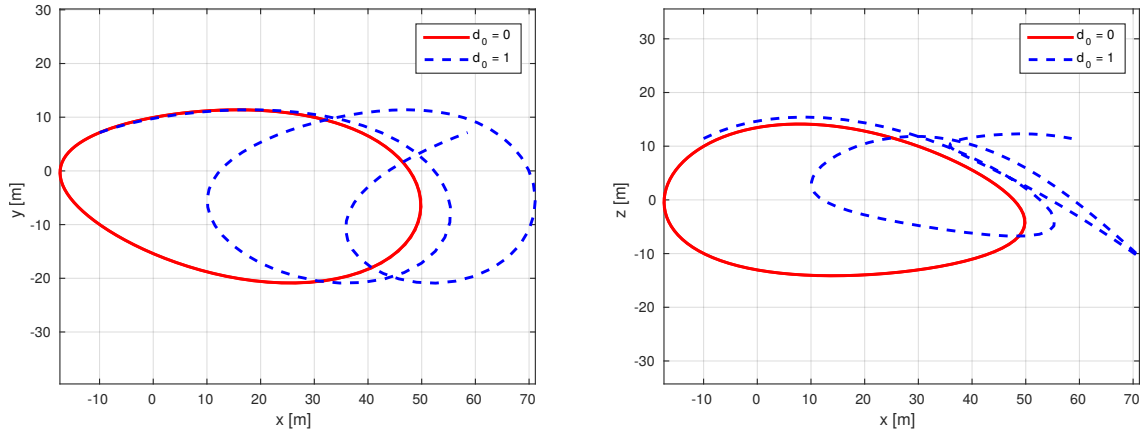
Figure 1.8 – Impact of d_4 and d_5 on the shape of the trajectory, $e = 0.4$.

2. **A simple way to characterize the periodicity property:** although the relative motion between spacecraft is not generally periodic, the periodicity property is interesting from the point of view of fuel consumption minimization [44]. This is mainly because in the absence of exogenous disturbances, once both spacecraft start to describe a periodic relative motion that respects the mission constraints, no further corrective control actions are required.

   Hence, several control algorithms which minimize the fuel consumption require the generated relative trajectories to be periodic [7, 9, 22, 44]. In order to develop control algorithms that minimize the fuel consumption, the generated relative trajectories are required to be periodic. However, in order to integrate this constraint in the formulation of the control algorithms, a mathematical description is needed.

   One can observe that in (1.25), the only non-periodic divergent term in these equations is $J_{\nu_0}(\nu)$, which always appears multiplied by the parameter $d_0(\nu)$. Therefore, it is evident that a sufficient condition to obtain a periodic relative trajectory is to have $d_0(\nu) = 0$, for all ν (see Fig. 1.9). However, from (1.26) we observe that, if for some ν, $d_0(\nu) = 0$, then $d_0(\nu) = 0$, for all ν. We conclude then that a relative trajectory is periodic if and only if for some ν the computation of $D(\nu) = C(\nu)\tilde{X}(\nu)$ produces a parameter $d_0 = 0$.

3. **The state propagation of the vector $D(\nu)$ is simpler than the dynamics of $\tilde{X}(\nu)$:** since for all ν, $\det(C(\nu)) \neq 0$, for a given ν, any vector $\tilde{X}(\nu)$ has a single correspondent $D(\nu)$ and vice-versa. This means that in order to study the evolution of the state vector $\tilde{X}(\nu)$, it suffices to analyze the behavior of $D(\nu)$.



(a) XY-plane view (b) XZ-plane view

Figure 1.9 – Link between d_0 and the periodicity property.

By manipulating (1.19) and (1.24) (see [28, Chapter 2] for details), we obtain the following dynamical system and state propagation representing the evolution of the vector of parameters:

$$D'(\nu) = \underbrace{\begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ -3e/\rho_\nu^2 & 0 & 0 & 0 & 0 & 0 \\ 3/\rho_\nu^2 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}}_{A_D(\nu)} D(\nu), \quad \text{or} \quad D(\nu) = \underbrace{\begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ -3eJ_{\nu_0}(\nu) & 0 & 1 & 0 & 0 & 0 \\ 3J_{\nu_0}(\nu) & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}}_{\Phi_D(\nu,\nu_0)} D(\nu_0), \tag{1.26}$$

and one can notice that the state propagation of $D(\nu)$ expressed in (1.26) is straightforward compared to the one corresponding to the vector $\tilde{X}(\nu)$, given by $\Phi(\nu, \nu_0) = \varphi(\nu)\varphi^{-1}(\nu_0)$.

1.3 Guidance of the relative motion

The guidance problem for the rendezvous missions consists in computing the control actions and the generated relative trajectories that satisfy a set of constraints over the actuators and the trajectories, that are modeled by some system of nonlinear controlled dynamical equations:

$$\frac{d}{dt}X(t) = f(X(t), t) + B(t)u(t), \tag{1.27}$$

or by some Linear Time-Varying (LTV) system of equations (as in (1.22), for example):

$$\frac{d}{dt}X(t) = A(t)X(t) + B(t)u(t), \tag{1.28}$$

for which a linear state-transition is available for the modeling of the dynamics:

$$X(t) = \Phi(t, t_0)X(t_0) + \int_{t_0}^{t} \Phi(t, s)B(s)u(s)ds \tag{1.29}$$

where $X(t) \in \mathbb{R}^6$ is the relative state and $u(t) \in \mathbb{R}^3$ is the vector that represents the control actions (the free variable can be chosen as t or ν, since they are in a one-to-one correspondence). As discussed previously, the dynamics presented in (1.26) are used in the sequel to model the relative dynamics in the phase of conception of control algorithms and the disturbed nonlinear Gauss equations in equinoctial orbital elements (1.8), to simulate and validate the execution of the computed control actions.

So far, the nature of the terms $B(t)u(t)$ and $\int_{t_0}^{t} \Phi(t, s)B(s)u(s)ds$ have not yet been discussed. In this section, the physical model adopted to represent the spacecraft's propellers, the effect of the application of the control actions on the relative dynamics and the metrics used to measure the fuel consumption is presented. Another subject treated in this section is the set of constraints that must be respected by the control actions and relative trajectories. The actuator constraints, as well as the space constraints of the rendezvous hovering phases, are also introduced. By the end of this section, all the necessary elements for the formulation of the guidance optimal problem for the rendezvous hovering phases will have been presented: a model for the propagation of the relative controlled dynamics, the fuel consumption that must be minimized, the space constraints describing the hovering zone and the restrictions on the control actions.

1.3.1 Space constraints: describing the hovering region

Several different types of space constraints must be satisfied by the relative motion between spacecraft during the rendezvous missions. Depending on the type of sensors used for the estimation of the relative distance and velocity of the spacecraft, a field of view is imposed to ensure the required conditions for the measurements (see Fig. 1.10). For close-range and proximity operations, a safety radius distance is imposed in order to avoid collisions (see Fig. 1.11). During the transition between checkpoints of the mission, the follower spacecraft must keep station in a delimited zone of the space relative to the leader spacecraft, the so-called hovering zone (see Fig. 1.12, more details in [37]). Hereafter, since the focus of

the developments presented in this dissertation are the station-keeping capabilities during the rendezvous hovering phases, only the restrictions constraining the relative motion to be included in a delimited zone of the space relative to the leader spacecraft are considered. This delimited zone, the hovering region, is assumed to be a rectangular cuboid.

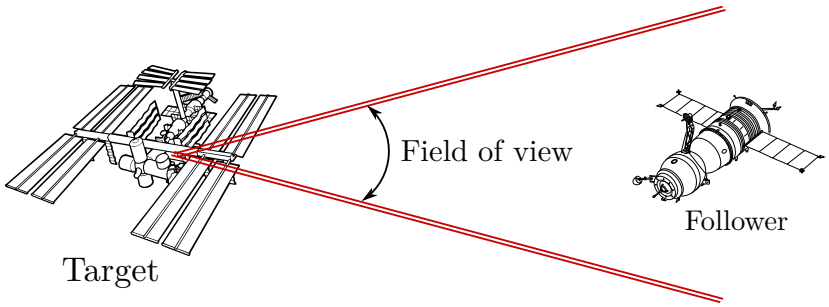


Figure 1.10 – Field of view.

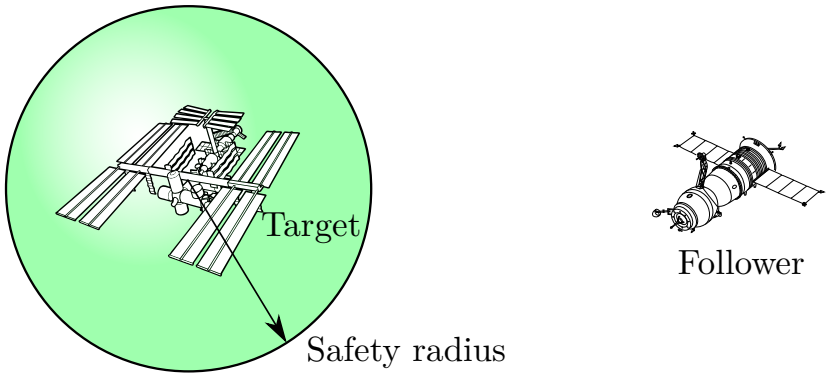


Figure 1.11 – Safety radius.

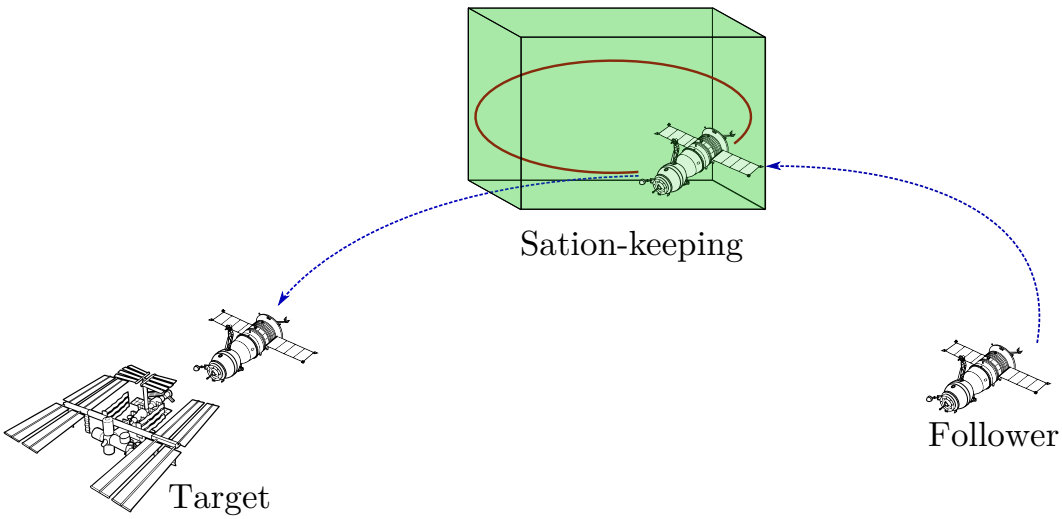


Figure 1.12 – Station-keeping.

These space constraints are generally modeled as:

$$\begin{aligned} &g_1(x(\nu), y(\nu), z(\nu), \nu) \leqslant 0, \\ &g_2(x(\nu), y(\nu), z(\nu), \nu) \leqslant 0, \\ &\quad\vdots \\ &g_m(x(\nu), y(\nu), z(\nu), \nu) \leqslant 0 \end{aligned} \qquad \forall \nu \geqslant \nu_0, \tag{1.30}$$

where the functions g_i are parametrized by the true anomaly ν and represent the geometrical restrictions that must be satisfied. For instance, the relative trajectories included in a rectangular cuboid hovering zone (see Fig. 1.13) are defined by the following inequalities:

$$\underline{x} \leqslant x(\nu) \leqslant \overline{x}, \quad \underline{y} \leqslant y(\nu) \leqslant \overline{y}, \quad \underline{z} \leqslant z(\nu) \leqslant \overline{z}, \quad \forall \nu \geqslant \nu_0. \tag{1.31}$$

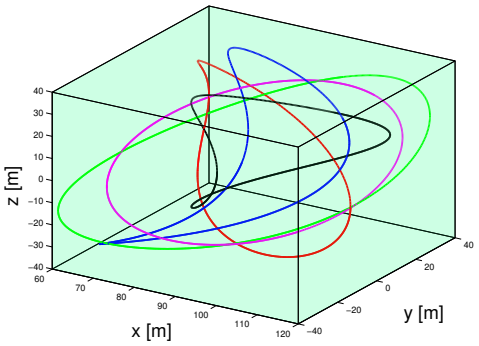


Figure 1.13 – Trajectories included in the hovering region.

Remark 1.3.1. *Hereafter d_0 is assumed to be null, since only periodic relative trajectories are considered.*

These trajectories can be also defined as a function of the vector of parameters. Using the relations (1.18) and (1.25) to replace $x(\nu)$, $y(\nu)$ and $z(\nu)$ in (1.31), the following inequalities are obtained:

$$\underline{x} \leqslant M_x(\nu)D(\nu_0) \leqslant \overline{x}, \quad \underline{y} \leqslant M_y(\nu)D(\nu_0) \leqslant \overline{y}, \quad \underline{z} \leqslant M_z(\nu)D(\nu_0) \leqslant \overline{z}, \quad \forall \nu, \tag{1.32}$$

where:

$$\begin{aligned} M_x(\nu) &= \left[0, \quad \frac{(2+e\,c_\nu)s_\nu}{1+e\,c_\nu}, \quad -\frac{(2+e\,c_\nu)c_\nu}{1+e\,c_\nu}, \quad \frac{1}{1+e\,c_\nu}, \quad 0, \quad 0\right] \\ M_y(\nu) &= \left[0, \quad 0, \quad 0, \quad 0, \quad \frac{c_\nu}{1+e\,c_\nu}, \quad \frac{s_\nu}{1+e\,c_\nu}\right] \\ M_z(\nu) &= \left[0, \quad c_\nu, \quad s_\nu, \quad 0, \quad 0, \quad 0\right] \end{aligned} \tag{1.33}$$

The constraints presented in (1.32) can be seen as an infinite number of affine inequalities on the entries of the vector $D(\nu_0)$ parametrized by ν. This class of constraints is hard to be treated in practice [68]. In the next chapters, alternative finite formulations for these constraints are presented.

1.3.2 Actuators and fuel consumption

The goal of this section is to provide the mathematical models and assumptions adopted for the actuators embedded on the spacecraft involved in the rendezvous missions. These actuators are mainly characterized by the nature of their propulsion method and by their geometrical configuration. Several propulsion methods can be adopted for spacecraft control, amongst them: mono-propellant rockets, electrostatic ion thrusters, Hall-effect thrusters, field-emission electric propulsion, etc. (see [23, 27, 75, 102] for more details). Concerning the geometrical configuration of the actuators, the spacecraft may have propellers mounted on each of its axes or be controlled by gimbaling and, for each case, an accurate criterion to measure the fuel consumption must be adopted (see [94] and Fig. 1.14 for details).

1.3.2.1 Defining the fuel consumption cost

In equations (1.27) and (1.29), the control action was presented as $u(t)$. In fact, this variable can be seen as the quotient between the thrust provoking an instantaneous force $F(t)$ and the vehicle mass $m(t)$:

$$u(t) = \frac{F(t)}{m(t)}. \tag{1.34}$$

The usual criterion adopted to measure the fuel consumption [94, Sec. II-III] is the following one:

$$\mathcal{J}(u(t)) := \left(\int_{t_0}^{t_f} (|u_x(t)| + |u_y(t)| + |u_z(t)|)^p \, dt \right)^{\frac{1}{p}}, \tag{1.35}$$

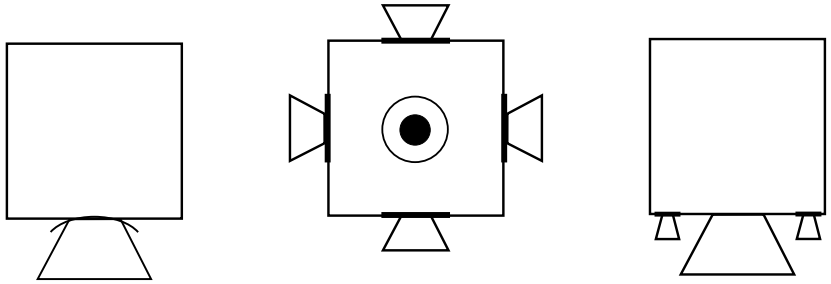
where t_0 and t_f define the time interval over which the fuel consumption is computed and p is a design parameter associated to the geometrical configuration of the propellers:

(a) if the vehicle is guided by one single gimbaled thruster, p is set to 2;

(b) if multiple ungimbaled thrusters are present, p is set to 1;

(c) if vehicle is guided by one main thruster accompanied by vernier engines, p is set to ∞.

These configurations are illustrated in Fig. 1.14.



(a) (b) (c)

Figure 1.14 – Thruster configurations: (a) $p = 2$, (b) $p = 1$ and (c) $p = \infty$.

Throughout this dissertation, the following assumptions are adopted:

1. the follower spacecraft is equipped with two identical thrusters symmetrically disposed on each axis, which is equivalent to the configuration (b) presented in Fig. 1.14;

2. the spacecraft uses chemical propulsion, which is characterized by a high magnitude of thrust and small changes in mass and small burn times. Given that, the characteristic time of the dynamics of the propellers can be considered much smaller than the time constant of the periodic relative motion (the interval of propulsion is of the order of the second, while the orbital period is of the order of the hour) and the controlled relative motion can be modeled as an impulsive system [11, 12, 38, 45];

3. a finite number N of control actions are applied at *a priori* fixed dates.

The first assumption indicates that the $p = 1$ must be used in the definition of the fuel consumption criterion (see Fig. 1.14):

$$\mathcal{J}(u(t)) := \int_{t_0}^{t_f} \left(|u_x(t)| + |u_y(t)| + |u_z(t)|\right) dt, \tag{1.36}$$

The second and third hypothesis provide a first motivation to model the control actions as instantaneous velocity changes applied at specific *a priori* known dates. In fact, the results presented by Neustadt in [82] corroborates this idea. The author demonstrates that the optimal controls obtained for the minimum-fuel Keplerian linearized elliptic rendezvous problem are purely impulsive and that the number of impulses is upper-limited by the dimension of the fixed final conditions of the optimal guidance problem. However, the algorithms proposed by Neustadt for generic functions representing the control actions $u(t)$ are not capable of accounting for constraints on the relative dynamics or the saturation of the propellers. Based on these results and, with the goal of obtaining a simple optimization problem formulation capable of accounting for the problem constraints, the control actions

are modeled as follows:

$$u(t) = \sum_{i=1}^{N} \Delta V(t_i)\, \delta(t - t_i), \tag{1.37}$$

where $\Delta V(t) = [\Delta V_x(t),\ \Delta V_y(t),\ \Delta V_z(t)]^T \in \mathbb{R}^3$ represents the velocity change of a spacecraft and $\delta(t - t_i)$ is the Dirac delta function representing an instantaneous pulse of magnitude 1 at t_i. This choice provides two important assets: the optimization is no longer performed over a space of functions $u(t)$, but over the space of finite-dimension vectors

$$\Delta V := [\Delta V(t_1)^T,\ \Delta V(t_2)^T, \ldots,\ \Delta V(t_{N-1})^T,\ \Delta V(t_N)^T]^T,$$

which is more efficiently tractable from a numerical point of view; moreover, this choice allows a straightforward imposition and tractability of constraints on the relative movement and on the control actions (as will be presented in Chapters 2 and 3).

Remark 1.3.2. *The obtained control actions will be validated in Chapter 4 by a technique based on simple computations (evaluation of polynomials and matrix multiplication using interval arithmetics). This technique will also provide an* a posteriori *validation for the continuous propagation of the relative trajectories.*

Under the previous assumptions, the fuel consumption criterion is expressed as a function of the vector ΔV:

$$\mathcal{J}(\Delta V) := \sum_{i=1}^{N} |\Delta V_x(t_i)| + |\Delta V_y(t_i)| + |\Delta V_z(t_i)|. \tag{1.38}$$

1.3.2.2 Effect of the control actions on the relative dynamics

The evolution of the state $\tilde{X}(\nu)$ can be formally described by the following impulsive system:

$$\begin{array}{rcll} \tilde{X}'(\nu) & = & \tilde{A}'(\nu)\tilde{X}(\nu), & \nu \neq \nu_i \\ \tilde{X}^+(\nu_i) & = & \tilde{X}(\nu_i) + T(\nu_i)B\Delta V(\nu_i), & i \in \mathbb{N} \end{array} \tag{1.39}$$

where $B = [0_3\ \mathbb{I}_3]^T$, $(\nu_i)_{i\in\mathbb{N}}$ are the true anomaly instants at which control actions are applied and $\tilde{X}^+(\nu_i) = \lim_{t\to\nu_i^+} \tilde{X}(\nu_i)$ represents the state obtained right after the application of an impulse $\Delta V(\nu_i)$.

Performing the variable change $D(\nu) = C(\nu)\tilde{X}(\nu)$ given in (1.24), the parameter vector

obtained after the application of an impulse is given by:

$$D^+(\nu_i) = D(\nu_i) + B_D(\nu_i)\Delta V(\nu_i), \tag{1.40}$$

where $B_D(\nu_i) = C(\nu_i)T(\nu_i)B$. Using the representation given in (1.26), the parameter vector right after a sequence of impulses applied at $\nu_1 \ldots \nu_{N-1}, \nu_N$ can be expressed as follows:

$$D^+(\nu_N) = \Phi_D(\nu_N, \nu_1)D(\nu_1) + \sum_{i=1}^{N} \Phi_D(\nu_N, \nu_i)B_D(\nu_i)\Delta V(\nu_i). \tag{1.41}$$

1.3.2.3 Saturation of thrusters

Considering that the saturation limit for each propeller is $\overline{\Delta V} > 0$, the saturation constraint is formulated as:

$$|\Delta V_{i,x}| \leqslant \overline{\Delta V}, \quad |\Delta V_{i,x}| \leqslant \overline{\Delta V}, \quad |\Delta V_{i,x}| \leqslant \overline{\Delta V}, \quad i \in \{1, \ldots, N\}, \tag{1.42}$$

where $\Delta V_{i,j} = \Delta V_j(t_i)$.

1.3.3 Optimal guidance problem formulation

At this point, all the relative dynamics, the behavior of the actuators and the saturations and space constraints of the problem have been presented. The fixed-time impulsive optimal guidance problem for the rendezvous hovering phases is then formulated as:

Problem 1.3.1 (Guidance problem)**.** *For a given scenario characterized by eccentricity e, semi-major axis a and initial state $D(\nu_1) = C(\nu_1)T(\nu_1)X(\nu_1) \in \mathbb{R}^6$, given $N \in \mathbb{N}$, find N impulsive controls (represented by $\Delta V \in \mathbb{R}^{3N}$) applied at given fixed instants $\nu_1, \ldots, \nu_N$, such*

that:

$$\min_{\Delta V} \ \mathcal{J}(\Delta V)$$
$$s.t. \begin{cases} D^+(\nu_N) = \Phi_D(\nu_N, \nu_1) D(\nu_1) + \sum_{i=1}^{N} \Phi_D(\nu_N, \nu_i) B_D(\nu_i) \Delta V_i, & \textit{state propagation} \\ d_0^+(\nu_N) = 0 & \textit{periodicity} \\ |\Delta V_{i,j}| \leqslant \overline{\Delta V}, \quad \forall i \in \{1, \dots, N\}, \ \forall j \in \{x, y, z\} & \textit{thruster saturation} \\ \underline{x} \leqslant M_x(\nu) D^+(\nu_N) \leqslant \overline{x} \\ \underline{y} \leqslant M_y(\nu) D^+(\nu_N) \leqslant \overline{y}, \quad \forall \nu \geqslant \nu_N & \textit{space constraints} \\ \underline{z} \leqslant M_z(\nu) D^+(\nu_N) \leqslant \overline{z} \end{cases}$$

(GP)

1.4 Conclusion

This chapter presents the context and hypotheses adopted to address the rendezvous hovering phases and the respective mathematical models for the relative motion, actuators and space constraints. The study of the two-body problem under Keplerian assumptions leads to the nonlinear Tschauner-Hempel equations. These resulting equations are then linearized and an analysis of the impact of the linearization hypothesis on the evolution of the relative trajectories is carried out. Simulations should account for the nonlinear dynamics in order to avoid discrepancies between the linear predictions and the nonlinear behavior. The Deaconu's parametrization is introduced, simplifying the propagation of the relative trajectory dynamics and bringing out the fact that the first parameter d_0 is a straightforward way to assess the periodicity feature for an arbitrary trajectory. The nature and geometrical placement of the propellers are discussed, as well as the space constraints of the problem. The equations providing a characterization of the control actions and the fuel consumption are exhibited. Finally, the guidance problem for the relative motion is formulated.

These previously presented developments are the basis for the next chapters: with the goal of obtaining problems that are more straightforward to be solved from a numerical point of view, reformulations of the guidance problem are proposed in Chapter 2. Algorithms for the resolution of these reformulated problems are also proposed therein, and their numerical efficiency is evaluated on a board certified for spatial applications. In Chapter 3, a model predictive control strategy is proposed to allow for the convergence of the relative motion towards the hovering region even under saturation of the propellers and under the presence of disturbances that are not taken into account by the synthesis model. In Chapter 4 a rigorous

function approximation technique is employed to produce validated solutions of the simplified linearized Tschauner-Hempel equations, providing certified bounds for the propagated relative trajectory that can be used to assess the violation of the imposed space constraints.

Chapter 2

Solving the guidance optimization problem

Contents

2.1 Introduction

In the previous chapter, the mathematical models for the constrained spacecraft relative motion were introduced. These models were employed in the formulation of the fixed-time impulsive optimal guidance problem for the rendezvous hovering phases. In this formulation, the relative trajectory solutions of this problem must satisfy infinitely many affine inequality constraints representing the hovering zone restriction. This makes this problem cumbersome to be solved from a numerical point of view (this type of optimization problems is known as a semi-infinite program (SIP) [14, 68]). Hence, this formulation is not directly suitable for

efficient autonomous control algorithms. For this reason, this chapter is dedicated to providing alternative formulations of the guidance problem GP, allowing for the use of optimization techniques that are efficiently tractable by computing devices usually employed in space applications.

Firstly, the theoretical aspects of the resolution of GP are discussed. We present a brief literature survey of techniques for guiding and maintaining the relative trajectories inside the hovering zone. Then, we discuss the conservativeness of these approaches compared to the ones employing generic periodic constrained relative trajectories, such as in the formulation of GP. Then, problem GP is reformulated in three different ways, which provide a finite description (i.e. description via a finite number of constraints) of the relative constrained orbits. The first reformulation is obtained by a discretization of the infinitely many constraints of the SIP, leading to a linear program (LP). This is a discretized version of the original problem, while the other two proposed reformulations are completely equivalent to the original one. The second one converts the GP problem into a semi-definite program (SDP), using the relation between the cone of non-negative univariate polynomials and the cone of semi-definite positive matrices. Finally, our new geometrical approach is based on the computation of the envelopes of the families of inequalities describing the set of periodic constrained relative trajectories. This provides a reformulation which relies on semi-algebraic functions, leading to a non-smooth optimization problem.

Secondly, numerical methods for solving each of these optimization problems are given and their practical aspects are considered. These optimization methods are then coded in C and embedded on a board certified for space applications containing a FPGA-synthesized LEON3 microprocessor. To conclude, the performances of the proposed approaches are assessed and compared for four different rendezvous scenarios.

2.2 Hovering zone guidance methods: bibliographic review

Let us briefly discuss the existing approaches for guiding and maintaining the relative trajectories between two spacecraft orbiting a central body inside a hovering zone.

A first series of works is based on the computation of a sequence of impulsive control actions generating segments of trajectories whose initial and final positions are located on the border of the hovering region [54, 56–58, 71, 72, 74, 105]. These control actions must minimize a criterion defined by the ratio between the fuel consumption and the sum of the free-flight durations inside the hovering zone:

$$J(\Delta V) = \left(\sum_{i=1}^{N} \|\Delta V\|^2 \right) \Big/ \left(\sum_{i=1}^{N} T_i \right),$$

where T_i represents the amount of time the trajectory remains inside the hovering region before reaching one of its bounds again for the i-th segment of trajectory. One example of this kind of approach is the "teardrop" strategy, which consists in computing an impulsive thrust such that the initial and final relative positions on the bounds of the hovering zone are equal for a given period of flight [57, 58] (see Fig. 2.1).

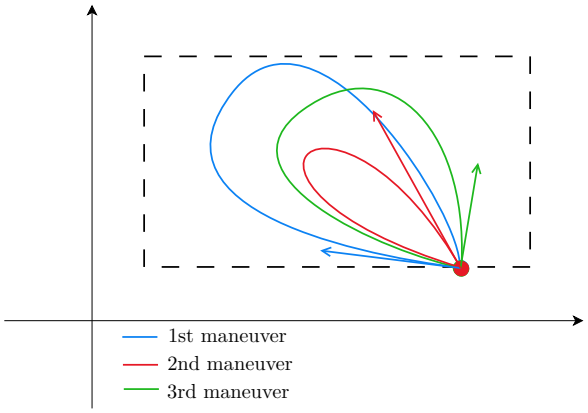


Figure 2.1 – Illustration of the teardrop strategy.

Another technique, that can be seen as more general version of the "teardrop" one, is the "pogo" strategy. It consists in computing an impulsive control each time the vehicle hits the bounds of the lobe to maintain it inside the hovering zone as long as possible while minimizing the fuel consumption [57, 74, 105] (see Fig. 2.2).

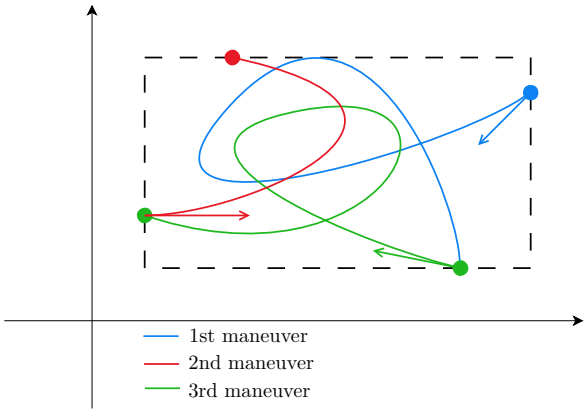


Figure 2.2 – Illustration of the pogo strategy.

For spacecraft sharing the same circular orbit, another possible strategy is the one based on "football" relative orbits [25, 106]. This technique is explained in [106, Section 2.2.2] by Woffinden: both vehicles are supposed to be in the same circular orbit with the chaser downrange; an impulsive velocity correction applied in radial direction makes the chaser

increase in altitude (this maneuver only changes the direction of the chaser's velocity vector and, consequently, the chaser maintains the same semi-major axis and orbital period. The shape of the orbit, however, is modified - the eccentricity of the chaser's orbit increases, causing apogee to increase and perigee to decrease as shown in Fig 2.3.

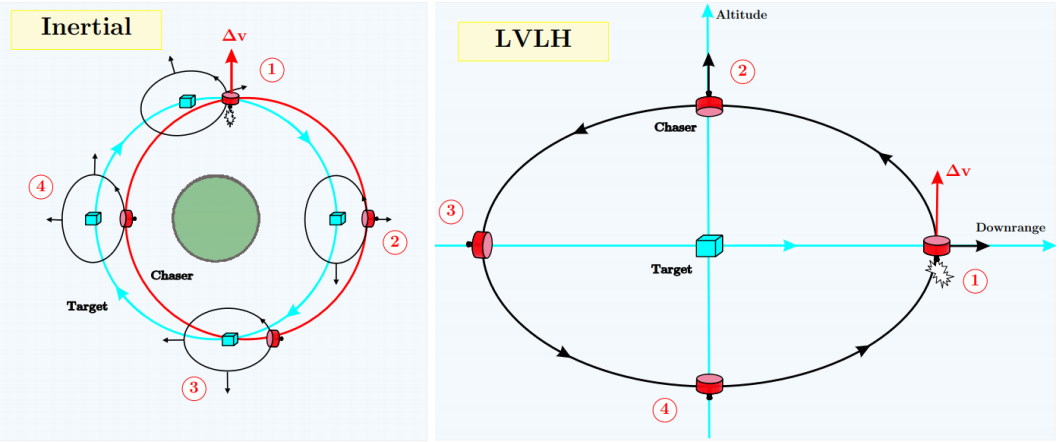


Figure 2.3 – Inertial and LVLH views of a "football" orbit (source [106], Fig. 2-6).

This change in apogee and perigee causes the football shaped relative trajectory, which are in fact ellipses with 2:1 ratio of the semi-major axis to the semi-minor axis. This particular repeating relative motion can then be employed to keep the chaser spacecraft in a holding pattern downrange form the target for station-keeping, as shown in Fig. 2.4.

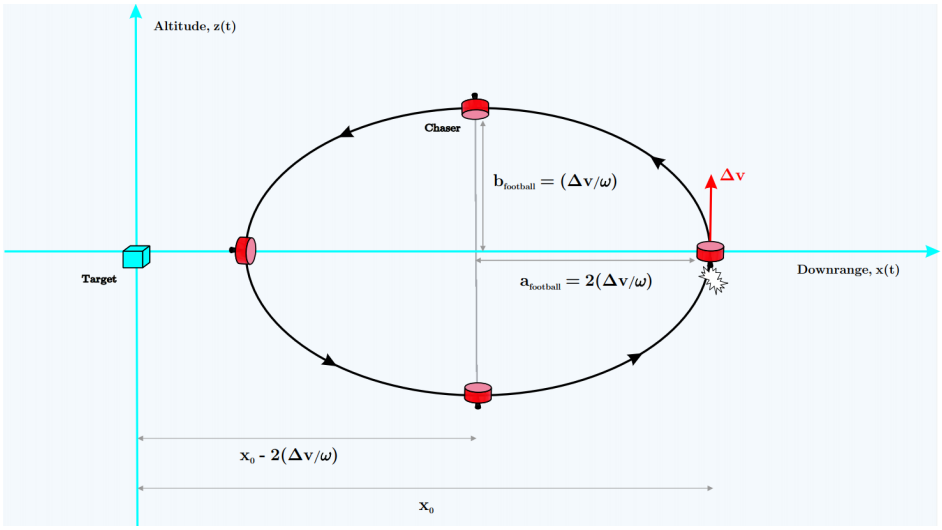


Figure 2.4 – Example of "football" orbit not centered at the target spacecraft (source [106], Fig. 2-7).

Techniques making use of relative orbital parameters, such as those presented by D'Amico and Gaias in [33, 40], are also employed to keep the relative trajectories in restricted zones of the space (see Fig. 2.5). Although these methods provide a straightforward characterization of relative trajectories satisfying the visibility and safety constraints, they are not as adapted

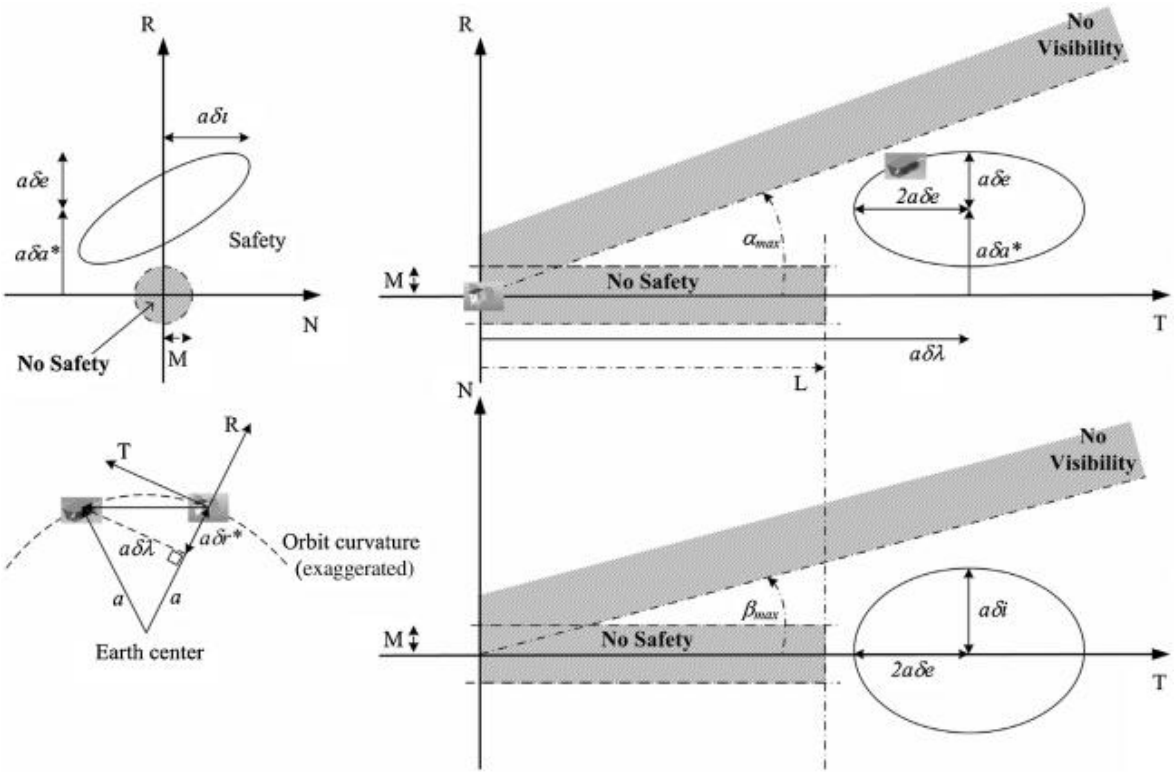


Figure 2.5 – Relative trajectories satisfying safety and visibility constraints (source [33], Fig. 3).

as the local relative Cartesian coordinates for the formulation of space constraints such as those describing a generic hovering region, requiring further variable changes.

Compared to the approach considering periodic constrained relative trajectories described in Chapter 1, these previously presented techniques are either harder to be numerically solved on low-performance devices dedicated to space applications or less efficient from the perspective of fuel saving. The "teardrop" and "pogo" strategies are numerically cumbersome, since they are based on nonlinear and non-convex optimization programs. Moreover, they also require the systematic application of impulses to keep the relative trajectory in the hovering zone. The "football" approach requires the vehicles to share the same circular orbit and does not account for all existent sizes and shapes of periodic relative orbits. The methods using relative orbital parameters cannot be directly employed in the formulation of the constraints describing the hovering region.

2.3 Proposed methods to solve the guidance problem

The developments presented in [7, 9, 22, 29, 30] have employed the naturally periodic relative orbits presented in Sec. 1.2.3 to design hovering control laws, showing the advantage of this approach with respect to the previously mentioned strategies. These periodic relative

trajectories were employed in Section 1.3.3 for the formulation of the fixed-time impulsive optimal guidance problem for the rendezvous hovering phases:

Problem 2.3.1 (Guidance problem)**.** *For a given scenario characterized by eccentricity e, semi-major axis a and initial state $D(\nu_1) = C(\nu_1)T(\nu_1)X(\nu_1) \in \mathbb{R}^6$, given $N \in \mathbb{N}$, find N impulsive controls (represented by $\Delta V \in \mathbb{R}^{3N}$) applied at given fixed instants $\nu_1, \dots, \nu_N$, such that:*

$$\min_{\Delta V} \ \mathcal{J}(\Delta V)$$

$$s.t. \begin{cases} D^+(\nu_N) = \Phi_D(\nu_N, \nu_1)D(\nu_1) + \sum_{i=1}^{N} \Phi_D(\nu_N, \nu_i)B_D(\nu_i)\Delta V_i, & \textit{state propagation} \\ d_0^+(\nu_N) = 0 & \textit{periodicity} \\ |\Delta V_{i,j}| \leqslant \overline{\Delta V}, \quad \begin{matrix} \forall i \in \{1, \dots, N\} \\ \forall j \in \{x, y, z\} \end{matrix} & \textit{thruster saturation} \\ \underline{x} \leqslant M_x(\nu)D^+(\nu_N) \leqslant \overline{x} & \\ \underline{y} \leqslant M_y(\nu)D^+(\nu_N) \leqslant \overline{y}, \quad \forall \nu & \textit{space constraints} \\ \underline{z} \leqslant M_z(\nu)D^+(\nu_N) \leqslant \overline{z} & \end{cases} \tag{GP}$$

In the following, this optimization problem GP is studied in detail. Several counterparts are proposed. The first one is based on constraint discretization and provides a suboptimal solution of the original problem, while the two others are strictly equivalent reformulations.

2.3.1 Discretization approach

The admissible set, i.e. the set of vectors D corresponding to periodic relative trajectories included in the hovering zone, is described both by the equation $d_0 = 0$ and by the inequalities in (1.32):

$$\underline{x} \leqslant M_x(\nu)D(\nu_0) \leqslant \overline{x}, \quad \underline{y} \leqslant M_y(\nu)D(\nu_0) \leqslant \overline{y}, \quad \underline{z} \leqslant M_z(\nu)D(\nu_0) \leqslant \overline{z}, \quad \forall \nu.$$

Since these inequalities must be satisfied for infinitely many values of ν, they are difficult to be addressed in practice. The traditional approach to take these space restrictions into account is to take finitely many values of ν the interval and evaluate the inequalities only on

these values [50, 90, 91]. In this case, the inequalities become:

$$\begin{aligned} &\underline{x} \leqslant M_x(\nu_i)D(\nu_0) \leqslant \overline{x}, \\ &\underline{y} \leqslant M_y(\nu_i)D(\nu_0) \leqslant \overline{y}, \\ &\underline{z} \leqslant M_z(\nu_i)D(\nu_0) \leqslant \overline{z}, \end{aligned} \quad , \quad \begin{aligned} &\nu_k \in [0, 2\pi], \\ &\forall k \in \{1, \dots, N_{disc}\} \end{aligned} \tag{2.1}$$

Using the previously provided discretized description of the admissible set, the (degraded) fixed-time impulsive optimal guidance problem for the rendezvous hovering phases can then be formulated as follows:

Problem 2.3.2 (LP)**.** *For a given scenario characterized by eccentricity e, semi-major axis a and initial state* $D(\nu_1) = C(\nu_1)T(\nu_1)X(\nu_1) \in \mathbb{R}^6$*, given* $N \in \mathbb{N}$*, find* N *impulsive controls (represented by* $\Delta V \in \mathbb{R}^{3N}$*) applied at given fixed instants* $\nu_1, \dots, \nu_N$*, such that:*

$$\min_{\Delta V} \mathcal{J}(\Delta V)$$

$$s.t. \begin{cases} D(\nu_1) = C(\nu_1)\tilde{X}(\nu_1), \\ D^+(\nu_N) = \Phi_D(\nu_N, \nu_1)D(\nu_1) + \sum_{i=1}^{N} \Phi_D(\nu_N, \nu_i)B_D(\nu_i)\Delta V_i, \\ d_0^+(\nu_N) = 0 \\ |\Delta V_{i,j}| \leqslant \overline{\Delta V}, \quad \begin{matrix} \forall i \in \{1, \dots, N\} \\ \forall j \in \{x, y, z\} \end{matrix} \\ \underline{x} \leqslant M_x(\nu_k)D^+(\nu_N) \leqslant \overline{x} \\ \underline{y} \leqslant M_y(\nu_k)D^+(\nu_N) \leqslant \overline{y} \quad , \quad \nu_k \in [0, 2\pi], \quad \forall k \in \{1, \dots, N_{disc}\} \\ \underline{z} \leqslant M_z(\nu_k)D^+(\nu_N) \leqslant \overline{z} \end{cases} \tag{LP}$$

The advantage of this approach is that only a finite number of linear inequalities must be satisfied by the vector D, which can be easily treated in practice by the state-of-the-art LP solvers. However, there is no *a priori* guarantee that the relative trajectory satisfies the original inequalities constraints (1.32) for the values of ν that do not belong to $\{\nu_i\}_{i \in I}$. Since this approach produces an outer-approximation of the set of relative trajectories that are enclosed in the hovering region, it leads to systematical violations of the original space constraints. In fact, the solution of a minimization problem containing a convex criterion function over polytopes described by finitely many affine inequalities generally lies on one of its vertices, which does not belong to the original admissible set (see Fig. 2.6).

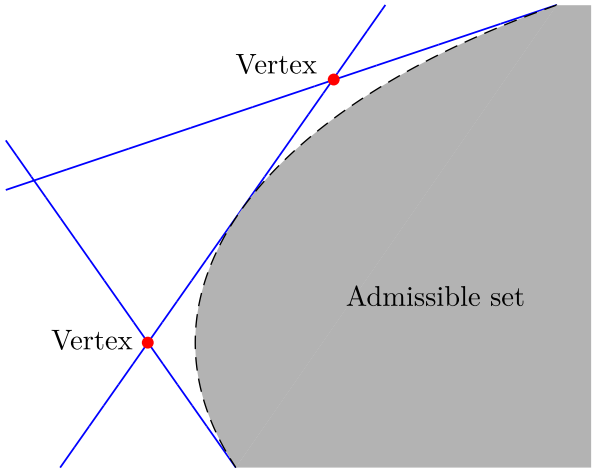


Figure 2.6 – Original admissible set (dashed line) and admissible set obtained after discretization (continuous lines).

2.3.2 Polynomial non-negativity approach

Another approach to address the infinitely many constraints in (1.32) is to perform a variable change in order to obtain polynomial non-negativity constraints and convert them into linear matrix inequalities (LMIs) using the result presented by Nesterov in [81]. This procedure produces an exact and finite description of the periodic relative trajectories included in the polytopic hovering region.

First, the tangent of the half-angle (Weierstrass) substitution is performed:

$$\tan(\nu/2) = w, \quad \cos(\nu) = \frac{1-w^2}{1+w^2}, \quad \sin(\nu) = \frac{2w}{1+w^2}. \tag{2.2}$$

The inequalities in (1.31) are then reformulated as univariate polynomial non-negativity constraints:

$$\begin{aligned} &\Gamma_{\underline{x}}(D(\nu_0), w) \geqslant 0, \quad &&\Gamma_{\overline{x}}(D(\nu_0), w) \geqslant 0, \\ &\Gamma_{\underline{y}}(D(\nu_0), w) \geqslant 0, \quad &&\Gamma_{\overline{y}}(D(\nu_0), w) \geqslant 0, \quad \forall w \in \mathbb{R}, \\ &\Gamma_{\underline{z}}(D(\nu_0), w) \geqslant 0, \quad &&\Gamma_{\overline{z}}(D(\nu_0), w) \geqslant 0, \end{aligned} \tag{2.3}$$

where $\Gamma_{(\cdot)}(D(\nu_0), w)$ are univariate polynomials in w whose coefficients depend linearly on the vector $D(\nu_0)$ (more details in [29] and [28, Chapter 3 and Appendix B]).

Then, using the relation between the cone of nonnegative univariate polynomials and the cone of semi-definite positive matrices presented by Nesterov in [81, Theorems 9 and 10],

these constraints can be converted into LMIs:

$$\begin{aligned}
&\exists\ Y_{\underline{y}}, Y_{\overline{y}} \geq 0 \in \mathbb{R}^{2\times 2},\ Y_{\underline{x}}, Y_{\overline{x}}, Y_{\underline{z}}, Y_{\overline{z}} \geq 0 \in \mathbb{R}^{3\times 3} \text{ s.t. :} \\
&\quad \gamma_{\underline{x}}(D(\nu_0)) = \Lambda^*(Y_{\underline{x}}), \quad \gamma_{\overline{x}}(D(\nu_0)) = \Lambda^*(Y_{\overline{x}}), \\
&\quad \gamma_{\underline{y}}(D(\nu_0)) = \Lambda^*(Y_{\underline{y}}), \quad \gamma_{\overline{y}}(D(\nu_0)) = \Lambda^*(Y_{\overline{y}}), \\
&\quad \gamma_{\underline{z}}(D(\nu_0)) = \Lambda^*(Y_{\underline{z}}), \quad \gamma_{\overline{z}}(D(\nu_0)) = \Lambda^*(Y_{\overline{z}}),
\end{aligned} \tag{2.4}$$

where $\gamma_{(\cdot)}(D(\nu_0))$ is the vector of coefficients of the respective $\Gamma_{(\cdot)}(D(\nu_0), w)$ polynomial and the linear operator $\Lambda^*(\cdot)$ is given in Appendix B. The admissible set, here denoted as S_D^p, can then be defined as:

$$S_D = \left\{ D \in \mathbb{R}^6 \text{ s.t } d_0 = 0 \ \middle|\ \exists Y_l \succeq 0 \text{ s.t } \gamma_l(D) = \Lambda^*(Y_l),\ \forall l \in \left\{\underline{x}, \overline{x}, \underline{y}, \overline{y}, \underline{z}, \underline{z}\right\} \right\} \tag{2.5}$$

and the fixed-time impulsive optimal guidance problem for the rendezvous hovering phases can then be formulated as follows:

Problem 2.3.3 (SDP)**.** *For a given scenario characterized by eccentricity e, semi-major axis a and initial state $D(\nu_1) = C(\nu_1)T(\nu_1)X(\nu_1) \in \mathbb{R}^6$, given $N \in \mathbb{N}$, find N impulsive controls (represented by $\Delta V \in \mathbb{R}^{3N}$) applied at given fixed instants $\nu_1, \ldots, \nu_N$, such that:*

$$\min_{\Delta V} \mathcal{J}(\Delta V)$$
$$s.t. \begin{cases}
D(\nu_1) = C(\nu_1)\tilde{X}(\nu_1), \\
D^+(\nu_N) = \Phi_D(\nu_N, \nu_1)D(\nu_1) + \sum_{i=1}^{N} \Phi_D(\nu_N, \nu_i)B_D(\nu_i)\Delta V_i, \\
d_0^+(\nu_N) = 0 \\
|\Delta V_{i,j}| \leqslant \overline{\Delta V}, \qquad \forall i \in \{1, \ldots, N\} \\
\qquad\qquad\qquad\quad \forall j \in \{x, y, z\} \\
\exists Y_l \succeq 0 \ s.t.\ \gamma_l(D^+(\nu_N)) = \Lambda^*(Y_l),\ \forall l \in \left\{\underline{x}, \overline{x}, \underline{y}, \overline{y}, \underline{z}, \underline{z}\right\}
\end{cases} \tag{SDP}$$

This new approach replaces the verification of infinitely many inequalities depending on ν by six semi-definite positive matrices of size at most 3 that must satisfy linear equalities constraints involving the coefficients of the polynomials presented above. As well as the discretization approach, this technique produces a finite description of the hovering region, with the advantage of generating a reformulated admissible set which is equivalent to the original admissible set.

2.3.3 Envelopes

An original contribution provided in this thesis is a new finite description of the admissible set different described in [30] that only depends on the evaluation of closed-form expressions on the entries of the vector D and on the space constraints. The objective of this new description is to provide an exact formulation that guarantees the non-violation of the constraints and enables an efficient computation on spacecraft compatible devices.

This is a joint work with M. Joldeş, C. Louembet and F. Camps (research engineer, LAAS-CNRS), presented in the 20th IFAC World Congress (IFAC 2017) under the title *Model predictive control for rendezvous hovering phases based on a novel description of constrained trajectories* [7].

Let us first denote by $\gamma_w(\cdot) \leqslant 0$ each inequality of (1.32) where the index $w \in \{\underline{x}, \overline{x}, \underline{y}, \overline{y}, \underline{z}, \overline{z}\}$. One can remark that these inequalities describe a family of surfaces (or lines) parametrized by ν. Moreover, the boundary of the set of points satisfying each of these inequalities is included in the envelope associated to these family of surfaces.

Definition 2.3.1 (Envelopes)**.** *Let $\gamma(\alpha_1, \ldots, \alpha_L, \nu) = 0$ be a family of one-parameter surfaces, with $L \in \mathbb{N}, L \leqslant 3$ depending on the parameter ν. Its envelope is the subset of points $(\alpha_1, \ldots, \alpha_L) \subseteq \mathbb{R}^L$ for which the following system of equations is satisfied:*

$$\gamma(\alpha_1, \ldots, \alpha_L, \nu) = \frac{\partial \gamma}{\partial \nu}(\alpha_1, \ldots, \alpha_L, \nu) = 0. \tag{2.6}$$

For each constraint $\gamma_w(\cdot)$, $w = \{\underline{x}, \overline{x}, \underline{y}, \overline{y}, \underline{z}, \overline{z}\}$, the resolution of (2.6) provides an implicit equation $g_w(D) = 0$ for which the set of solutions contains a surface in space that separates the points that satisfy the constraint inequality from those that do not.

Thus, a verification method is obtained for checking that the vector D belongs to S_D:

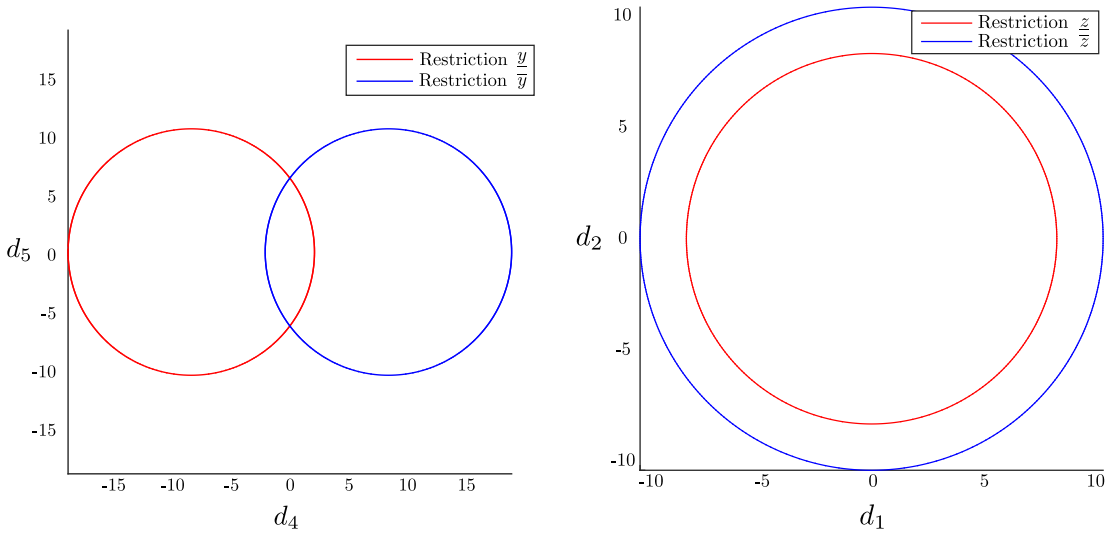
$$D \in S_D \Leftrightarrow (d_0 = 0 \text{ and } \forall w,\ g_w(D) \leqslant 0).$$

Solving (2.6) for indices $w \in \{\underline{y}, \overline{y}, \underline{z}, \overline{z}\}$ and using Sylvester's matrix implicitization method (see [53]), the following expressions are obtained for the envelopes $g_w(\cdot)$ on y and z axis:

$$\begin{aligned} g_{\underline{y}}(d_4, d_5) &= (d_4 - e\underline{y})^2 + d_5^2 - \underline{y}^2, \\ g_{\overline{y}}(d_4, d_5) &= (d_4 - e\overline{y})^2 + d_5^2 - \overline{y}^2, \end{aligned} \tag{2.7}$$

$$\begin{aligned} g_{\underline{z}}(d_1, d_2) &= d_1^2 + d_2^2 - \underline{z}^2, \\ g_{\overline{z}}(d_1, d_2) &= d_1^2 + d_2^2 - \overline{z}^2. \end{aligned} \tag{2.8}$$

Note that (2.7) and (2.8) describe circles in the (d_4, d_5) plane and (d_1, d_2) plane respectively (see Fig. 2.7(a) and 2.7(b)). For instance the boundary of the admissible set for the constraint $\gamma_{\overline{y}}$ is a circle of center $(e.\overline{y}, 0)$ and radius $\overline{y}$.



(a) Contour lines $g_{\overline{y}} = g_{\underline{y}} = 0$ for $e = 0.8$, $\underline{y} = -10$, $\overline{y} = 10$.

(b) Contour lines $g_{\overline{z}} = g_{\underline{z}} = 0$ for $e = 0.8$, $\underline{z} = -8$, $\overline{z} = 10$.

Figure 2.7 – Contour lines for y and z coordinates.

Finding an implicit function for the x constraints requires further work. After solving the system of equations from (2.6), a parametric description is obtained in function of ν, d_3 and $\underline{x}$ or $\overline{x}$:

$$\begin{aligned} d_1 &= \frac{((2x_m - 2d_3)(1 + ec_\nu) + e^2 x_m c_\nu^2)s_\nu}{(2 + ec_\nu)^2}, \\ d_2 &= -\frac{((2x_m - 2d_3)(1 + ec_\nu) + e^2 x_m c_\nu^2)c_\nu + e(d_3 + x_m)}{(2 + ec_\nu)^2}, \end{aligned} \tag{2.9}$$

where x_m stands for $\underline{x}$ or $\overline{x}$.

Using Sylvester's matrix implicitization method, (2.9) produce a multivariate polynomial in d_1, d_2, d_3:

$$\hat{g}_{x_m}(d_1, d_2, d_3) = \textstyle\sum_{\xi \in \mathbb{N}^3} \theta_\xi d_1^{\xi_1} d_2^{\xi_2} d_3^{\xi_3}. \tag{2.10}$$

As shown in Fig. 2.8, the envelope $g_{\overline{x}} = 0$ contains the boundary of the inner-convex set of points satisfying the inequality $\overline{x}$ for all ν.

This set of solutions can be described by first remarking that the multivariate polynomial

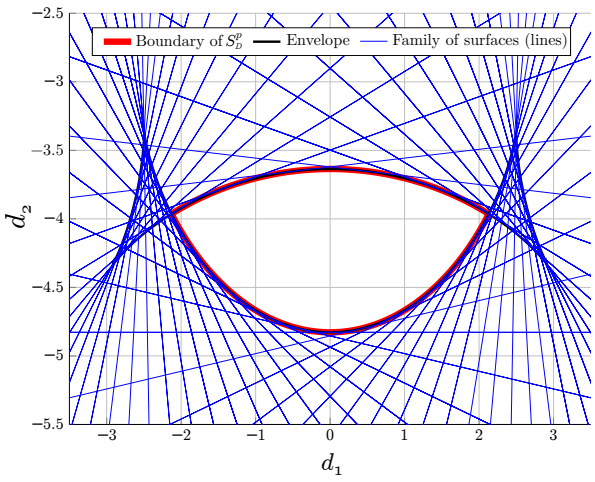


Figure 2.8 – Family of surfaces, envelope and boundary of admissible set for $\hat{g}_{\overline{x}} = 0$, with $\overline{x} = 10$, $e = 0.9$, $d_3 = 5$.

(2.10) can be seen as a fourth degree polynomial in d_3:

$$\hat{g}_{x_m}(d_1, d_2, d_3) = \sum_{0 \leqslant i \leqslant 4} \bar{\theta}_i(d_1, d_2)\, d_3^i \ . \tag{2.11}$$

Then, it suffices to choose the root of (2.11) that describes the inner convex set presented in Fig. 2.8. Considering $\hat{g}_{\overline{x}}(\cdot)$, the smallest real root is the one to be selected. Conversely, the largest one is chosen when considering the envelope $\hat{g}_{\underline{x}}(\cdot)$. This is done by choosing the "right" roots of the fourth degree polynomial in d_3 [1], producing the desired g_{x_m} functions that describe the set of admissible points with respect to $\gamma_{x_m} \leqslant 0$:

$$\begin{aligned} g_{\underline{x}}(d_1, d_2, d_3) &= r_{\underline{x}}(d_1, d_2, e) - d_3, \\ g_{\overline{x}}(d_1, d_2, d_3) &= d_3 - r_{\overline{x}}(d_1, d_2, e), \end{aligned} \tag{2.12}$$

where $r_{\underline{x}}(d_2, d_3, e)$ and $r_{\overline{x}}(d_2, d_3, e)$ are the functions that return respectively the greatest real root of $\hat{g}_{\underline{x}}$ and the lowest real root $\hat{g}_{\overline{x}}$ (2.11). In Fig. 2.9 an illustration of the geometrical volumes containing the vectors D that respect the $\underline{x}$ and $\overline{x}$ restrictions is provided.

Using the functions $g_{\underline{x}}$, $g_{\overline{x}}$, $g_{\underline{y}}$, $g_{\overline{y}}$, $g_{\underline{z}}$ and $g_{\overline{z}}$ obtained previously, the admissible set can be redefined as:

$$S_D = \left\{ D \in \mathbb{R}^6 \text{ s.t } d_0 = 0 \;\middle|\; g_w(D) \leqslant 0, \forall w \in \left\{\underline{x}, \overline{x}, \underline{y}, \overline{y}, \underline{z}, \overline{z}\right\} \right\}, \tag{2.13}$$

and the fixed-time impulsive optimal guidance problem for the rendezvous hovering phases can then be formulated as follows:

Problem 2.3.4 (ENV)**.** *For a given scenario characterized by eccentricity e, semi-major axis*

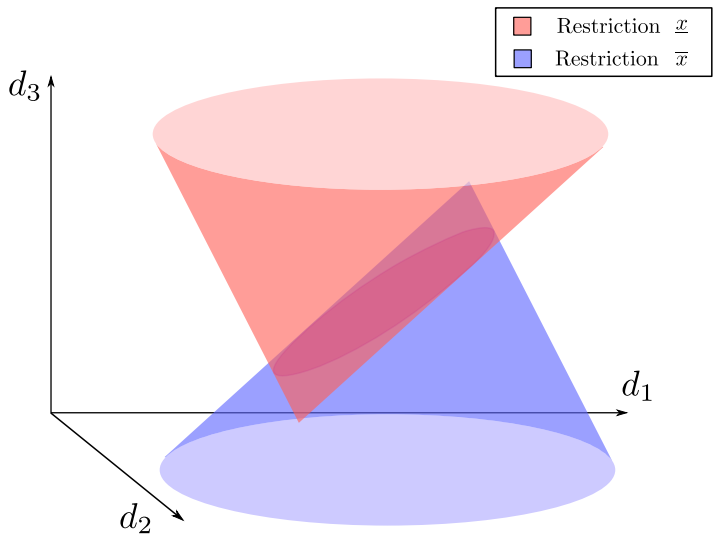


Figure 2.9 – Isosurfaces $g_{\underline{x}} = g_x max = 0$ for $e = 0.8$, $\underline{x} = -10$, $\overline{x} = 10$.

a and initial state $D(\nu_1) = C(\nu_1)T(\nu_1)X(\nu_1) \in \mathbb{R}^6$*, given* $N \in \mathbb{N}$*, find* N *impulsive controls (represented by* $\Delta V \in \mathbb{R}^{3N}$*) applied at given fixed instants* $\nu_1, \dots, \nu_N$*, such that:*

$$\min_{\Delta V} \mathcal{J}(\Delta V)$$
$$s.t. \begin{cases} D(\nu_1) = C(\nu_1)\tilde{X}(\nu_1), \\ D^+(\nu_N) = \Phi_{\nu_1}^{\nu_N} D(\nu_1) + \sum_{i=1}^{N} \Phi_{\nu_i}^{\nu_N} B_D(\nu_i)\Delta V_i, \\ d_0^+(\nu_N) = 0 \\ |\Delta V_{i,j}| \leqslant \overline{\Delta V}, \quad \forall i \in \{1, \dots, N\} \\ \qquad\qquad\qquad\quad \forall j \in \{x, y, z\} \\ g_w(D^+(\nu_N)) \leqslant 0, \quad w \in \{\underline{x}, \overline{x}, \underline{y}, \overline{y}, \underline{z}, \overline{z}\} \end{cases} \tag{ENV}$$

2.3.4 Conclusions

Three distinct reformulations of the problem GP were proposed.

The first one is based on a discretization of the infinitely many constraints describing the hovering zone, consecutively consisting in a degraded version of the original problem. Although this process leads to a LP problem, which can be efficiently solved by numerical methods, the loss of information provoked by the discretization cannot ensure the continuous satisfaction of the space constraints.

Contrarily to the LP approach, the two other proposed counterparts of GP provide perfectly equivalent reformulations. The SDP version is obtained by converting the infinitely many inequalities present in GP into finitely many LMIs, using the results presented by Nesterov in [81]. The numerical resolution of this class of problems requires complex iterative algorithms that have not been extensively tested on space dedicated devices, which motivates

the proposition of the last reformulation of problem GP, the ENV approach.

This last approach is an original contribution of this thesis, and relies on a geometrical analysis of the set of periodic relative trajectories included in the hovering region (the admissible set) to provide closed-form "check" functions that indicate the violation or satisfaction of the constraints. One of the advantages of the ENV approach is that the numerical algorithms that can be employed in the resolution of the associated optimization problem are lees cumbersome than those used for the SDP approach, as will be discussed in the sequel.

2.4 Embedded algorithms

Hereafter the approach employed to address the optimization problems previously formulated is outlined. The software and hardware environment on which their performances are assessed is also presented.

This is a joint work with F. Camps (research engineer, LAAS-CNRS), M. Joldeş and C. Louembet, under the title of *Embedding a SDP-based control algorithm for the orbital rendezvous hovering phases* [24], presented in the 2018 25th Saint Petersburg International Conference on Integrated Navigation Systems (ICINS 2018).

2.4.1 Test environment

The tests were performed on an AEROFLEX GAISLER GR-XC6S board (Fig. 2.10) that contains a synthesized LEON3 microprocessor, which has a SPARC V8 architecture (see [84] for further specifications), and supports a IEEE-754 compliant floating-point unit with single and double precision (32- and 64-bit floats), running a Linux 2.6 environment that simulates the performance of devices usually employed in space applications (see [35]). The compilation chain and the used libraries are detailed hereafter.

2.4.2 Solving the LP problems

The LP problems are solved using the GLPK library, a linear programming kit intended for solving large-scale linear programming (LP), mixed integer programming (MIP), and other related problems [41]. The technique used for solving the problems is the primal simplex method, with default GLPK parameters and stop criteria.

The GLPK library solves linear programs of the form:

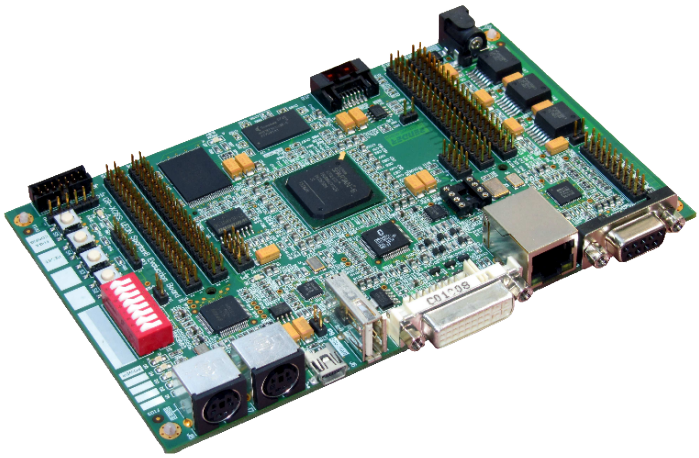


Figure 2.10 – AEROFLEX GAISLER GR-XC6S board.

$$\begin{aligned} \max_{x} \quad & c^T x \\ & A_{eq} x = b_{eq} \\ & A_{in} x \leqslant b_{in} \end{aligned} \tag{GLPK}$$

where $A_{eq}x = b_{eq}$ represents a system of equations and $A_{in}x \leqslant b_{in}$ represents a system of inequalities that must be satisfied by the decision variables x.

In order to account for the absolute values appearing in the criterion $\mathcal{J}(\Delta V) = \sum_{i=1}^{N} |\Delta V_{i,x}| + |\Delta V_{i,y}| + |\Delta V_{i,z}|$ and in the saturation constraint $|\Delta V_{i,j}| \leqslant \overline{\Delta V}$, the saturation constraints are split into two inequalities:

$$|\Delta V_{i,j}| \leqslant \overline{\Delta V} \Leftrightarrow \begin{array}{r} -\Delta V_{i,j} + \overline{\Delta V} \geqslant 0 \\ \Delta V_{i,j} + \overline{\Delta V} \geqslant 0 \end{array} \tag{2.14}$$

and the slack variables $Z_{i,j} \geqslant 0$ are introduced, producing:

$$\begin{array}{c} J(\Delta V) = \sum_{i=1}^{N} |\Delta V_{i,x}| + |\Delta V_{i,y}| + |\Delta V_{i,z}| \\ |\Delta V_{i,j}| \leqslant \overline{\Delta V} \\ \Updownarrow \\ J(Z) = \sum_{i=1}^{N} Z_{i,x} + Z_{i,y} + Z_{i,z} \\ Z_{i,j} - \Delta V_{i,j} \geqslant 0, \quad Z_{i,j} + \Delta V_{i,j} \geqslant 0, \quad Z_{i,j} \leqslant \overline{\Delta V} \end{array} \tag{2.15}$$

which results in $9N$ affine inequalities to be satisfied.

Each of the inequalities describing the discretized admissible set produce N_{disc} affine

inequalities to be satisfied, which results in a total of $6N_{disc}$ affine inequalities:

$$\begin{aligned} &\underline{x} \leqslant M_x(\nu_i)D(\nu_0) \leqslant \overline{x}, \\ &\underline{y} \leqslant M_y(\nu_i)D(\nu_0) \leqslant \overline{y}, \\ &\underline{z} \leqslant M_z(\nu_i)D(\nu_0) \leqslant \overline{z}, \end{aligned} \quad , \quad \begin{aligned} &\nu_k \in [0, 2\pi], \\ &\forall k \in \{1, \ldots, N_{disc}\} \end{aligned}$$

One single equation must be satisfied, the one related to the periodicity constraint:

$$d_0^+(\nu_N) = 0.$$

In summary, the number of decision variables is $6N$, the number of equality constraints is equal to 1 and the number of inequality constraints is equal to $9N + 6N_{disc}$.

2.4.3 Solving the SDP problems

Even though several libraries and solvers can be employed in the resolution of the SDP problems (SeDuMi [97], SDPT3 [98], SDPA [39], MOSEK [3]), the CSDP [16, 17] library is used throughout this work. This choice is motivated by the fact that the CSDP library is open-source, available in C and can be cross-compiled for the SPARC V8 architecture of the test environment previously presented in Sec. 2.4.1. This library executes the routines that implement a predictor corrector variant of the interior-point algorithm of Helmberg, Rendl, Vanderbei, and Wolkowicz for semidefinite programming [51].

The CSDP library solves semidefinite programs of the form:

$$\begin{aligned} \max_{\mathcal{X}} \quad & \operatorname{tr}(\mathcal{C}\mathcal{X}) \\ & \operatorname{tr}(\mathcal{A}_i\mathcal{X}) = \alpha_i, \quad \forall i \in \{1, \ldots, m\} \\ & \mathcal{X} \succeq 0 \end{aligned} \tag{CSDP}$$

where $\alpha_n \in \mathbb{R}$ and all the matrices $\mathcal{C}$, $\mathcal{A}_n$, and $\mathcal{X}$ are real and symmetric matrices (see [17] for details). To write SDP in the form of CSDP, the decision variables must be identified:

$$Y_x^l,\ Y_x^u,\ Y_z^l,\ Y_z^u \in \mathbb{R}^{3\times 3},\quad Y_y^l,\ Y_y^u \in \mathbb{R}^{2\times 2},\quad \Delta V \in \mathbb{R}^{3N} \tag{2.16}$$

Since only equalities of the type $\operatorname{tr}(\mathcal{A}_n\mathcal{X}) = \alpha_n$ and semi-definiteness constraints of the type $\mathcal{X} \succeq 0$ can be used in CSDP, the variables of SDP must be modified in order to obtain only symmetric semidefinite positive matrices (or positive scalars). The problem is then

reformulated as follows: first the saturation constraints are split into two inequalities:

$$|\Delta V_{i,j}| \leqslant \overline{\Delta V} \Leftrightarrow \begin{array}{c} -\Delta V_{i,j} + \overline{\Delta V} \geqslant 0 \\ \Delta V_{i,j} + \overline{\Delta V} \geqslant 0 \end{array}$$

Then the variables $W^-_{i,j} = \max\{0, -\Delta V_{i,j}\}$, $W^+_{i,j} = \max\{0, \Delta V_{i,j}\}$ (which will be used instead of $\Delta V_{i,j}$, that will be lately reconstructed via the relation $\Delta V_{i,j} = -W^-_{i,j} + W^+_{i,j}$) and the auxiliary variables $Z_{i,j}$ and $\bar{Z}_{i,j}$ are introduced, producing:

$$\begin{array}{c} J(\Delta V) = \sum_{i=1}^{N} |\Delta V_{i,x}| + |\Delta V_{i,y}| + |\Delta V_{i,z}| \\ -\Delta V_{i,j} + \overline{\Delta V} \geqslant 0, \quad \Delta V_{i,j} + \overline{\Delta V} \geqslant 0 \\ \Updownarrow \\ J(Z) = \sum_{i=1}^{N} Z_{i,x} + Z_{i,y} + Z_{i,z} \\ W^-_{i,j} + W^+_{i,j} = Z_{i,j}, \quad Z_{i,j} + \bar{Z}_{i,j} = \overline{\Delta V} \end{array} \tag{2.17}$$

with $W^-_{i,j}, W^+_{i,j}, Z_{i,j}, \bar{Z}_{i,j} \geqslant 0$. The variables $W^-_{i,j}$ and $W^+_{i,j}$ play the role of the norm of the negative and the positive part of $\Delta V_{i,j}$, respectively; $Z_{i,j}$ are slack variables that assume the value of $|\Delta V_{i,j}|$; $\bar{Z}_{i,j}$ are the complementary of $Z_{i,j}$ with respect to $\overline{\Delta V}$ ($Z_{i,j} + \bar{Z}_{i,j} = \overline{\Delta V}$).

Now that all the necessary variables have been introduced, $\mathcal{X}$ can be structured as a symmetric semidefinite positive block-diagonal matrix:

$$\begin{array}{c} \mathcal{X} = \operatorname{diag}(Y^l_x,\ Y^u_x,\ Y^l_z,\ Y^u_z,\ Y^l_y,\ Y^u_y,\ W^-,\ W^+, \\ Z,\ \bar{Z}) \in \mathbb{R}^{(16+12N)\times(16+12N)}, \end{array}$$

where $W^- = \operatorname{diag}(W^-_{1,x}, \ldots, W^-_{N,z})$ and W^+, Z, $\bar{Z}$ are defined analogously. The fact that this matrix is semidefinite positive accounts for the inequality constraints previously presented. With these presented modifications, the equality constraints can then be written in the form $\operatorname{tr}(\mathcal{A}_n \mathcal{X}) = \alpha_n$.

In summary, the total number of these constraints adds up to $27 + 6N$: for each coefficient of the polynomials Γ^k_j presented in SDP there is one equality constraint $\gamma^k_j = \Lambda^*(Y^k_j)$ involving a semidefinite positive matrix. The polynomials related to the x and z constraints are of degree 4, with 5 coefficients; the polynomials related to y are of degree 2, with 3 coefficients; each axis has 2 inequality constraints, which results in a subtotal of $2(5+5+3) = 26$ equality constraints; the periodicity constraint is expressed via one single equality constraint $d_0 = 0$, which results in a subtotal of $26+1 = 27$ equality constraints; there are $3N$ equality constraints $W^+_{i,j} + W^-_{i,j} = Z_{i,j}$, which results in a subtotal of $27 + 3N$ constraints; finally, there are $3N$

equality constraints $Z_{i,j} + \bar{Z}_{i,j} = \overline{\Delta V}$, which results in a total of $27 + 3N + 3N = 27 + 6N$ constraints.

Given that the criterion in CSDP is maximized and that $J(Z) = \sum_{i=1}^{N} Z_{i,x} + Z_{i,y} + Z_{i,z}$ has to be minimized, the matrix $\mathcal{C}$ will be filled with -1 in the entries that occupy the same positions as the variables $Z_{i,j}$ in $\mathcal{X}$ and with zeros elsewhere.

2.4.4 Embedding the libraries for the LP and SDP methods

The detailed command descriptions for the installation of cross-compilers and libraries used in the resolution of LP and SDP problems are available in Appendix D. Hereafter a brief summary of this procedure is presented:

- *Cross-compiler*: in order to perform the tests in an environment as close as possible to those employed in space systems, an embedded synthesized LEON3 microprocessor is employed. The architecture of this microprocessor requires a particular X86/SPARC cross-compiler. The first step to use this environment is to install the cross-compilers proposed by AEROFLEX GAISLER[1], which are then used for the cross-compilation of computation libraries such as LAPACK.

- *LAPACK*[2] *and GLPK*: LAPACK is a widely used Linear Algebra PACKage. Since LAPACK is written in Fortran, it requires a Fortran cross compiler.

- *CSDP*[3]: a C Library for Semidefinite Programming, it also uses the LAPACK library.

- *Embedding the libraries*: The computation libraries must be installed on the board (in the case of a non-static compilation) in the standard directory of the libraries.

2.4.5 Solving the ENV problems

The ENV problems are solved using a combination of a penalty technique and two non-smooth optimization algorithms. The penalty technique consists in adding functions representing the violation of the constraints to the minimization criterion, multiplying them by a big penalty coefficient. The idea is to penalize any intermediary solution that does not satisfy

[1] http://gaisler.com/anonftp/linux/linux-2.6/toolchains/sparc-linux-4.4.2/sparc-linux-ct-multilib-0.0.7.tar.bz2

[2] LAPACK official site: http://www.netlib.org/lapack/, Archives: http://www.netlib.org/lapack/lapack-3.7.0.tgz

[3] CSDP official site : https://projects.coin-or.org/Csdp/

the constraints of the problem. This leads to an unconstrained optimization problem (see Appendix E.1 for further details):

$$\min_{\Delta V} J(\Delta V) + \eta \left(\max\left\{|d_0^+(\nu_N)|, 0\right\} + \sum_{i,j} \max\left\{|\Delta V_{i,j}| - \overline{\Delta V}, 0\right\} + \sum_{w} \left\{g_w(D^+(\nu_N)), 0\right\} \right)$$

where $D^+(\nu_N) = \Phi_{\nu_1}^{\nu_N} C(\nu_1)\tilde{X}(\nu_1) + \sum_{i=1}^{N} \Phi_{\nu_i}^{\nu_N} B_D(\nu_i)\Delta V_i$ and $\eta \gg 1$ (hereafter η is set to 10^8).

As one can remark, the unconstrained problem contains a non-smooth objective function (max, absolute value and the envelope functions $g_w(\cdot)$). For the resolution of this type of problems, two iterative algorithms with different properties are chosen to be employed: the sub-gradient method proposed by Shor in [95], a first-order approach that resembles the steepest descent optimization algorithm, but uses sub-gradients instead of gradients (see Appendix E.2); and the BFGS quasi-Newton method proposed by Lewis and Overton in [64], which consists in building local quadratic approximations of the penalized objective function and computing descent steps using the sub-gradients and the inverse Hessian (see Appendix E.3).

A hybrid method combining both algorithms (see Appendix E.4 for details) is employed in order to take advantage of their distinct strengths:

The quasi-Newton algorithm benefits from a faster decrease of the penalized objective function along the iterations, while only the sub-gradient method has guaranteed convergence, being used to refine the approximated solution obtained in the previous step.

Since this approach does not require particular optimization solvers/libraries to be implemented, it is completely coded using standard C functions (see Algorithm 5 in Appendix E.4). This gives total control of the user interactivity, general behavior and termination conditions of the algorithm (for example, a maximal computing time for solving the optimization problem can be set as termination condition, with the option of allowing further iterations - which is not the usual behavior for generic solvers).

2.5 Simulations and results

Hereafter the practical performances of the LP, SDP and ENV approaches are assessed for the resolution of the rendezvous hovering phases problem by carrying out simulations on the space dedicated test environment described in Sec. 2.4.1. These simulations account for four distinct ISS rendezvous missions in which the follower spacecraft is maneuvered from four different initial states $X_{01} - X_{04}$ to a periodic orbit enclosed by the hovering zone described by

Algorithm 1: Solving non-smooth convex optimization problems

Input : ξ_1 - initial point
$\Upsilon(\cdot)$ - non-smooth convex function to minimize
$g(\cdot)$ - oracle that returns one of the sub-gradients of $\Upsilon(\cdot)$ at a given point
I_{qn}, I_{sg} - number of iterations
σ - coefficient for the sub-gradient steps (set to 1 if not explicitly specified)
Output : ξ - solution

$\xi_{best} \leftarrow \xi_1$
`// Quasi-Newton method`
$H_1 = \mathbb{I}_n$;
for $k = 1$ **to** I_{qn} **do**
  $\xi_{k+1} \leftarrow \xi_k - \lambda_k H_k g(\xi_k)$, where $\lambda_k > 0$ computed by inexact line search;
  if $\Upsilon(\xi_{k+1}) < \Upsilon(\xi_{best})$ **then**
    $\xi_{best} \leftarrow \xi_{k+1}$
  end
  Update H_{k+1} as positive definite matrix satisfying secant condition
  $H_{k+1}(g(\xi_{k+1}) - g(\xi_k)) = -\lambda_k H_k g(\xi_{k+1})$;
end
$\xi_1 \leftarrow \xi_{best}$;
`// Subgradient method`
for $k = 1$ **to** I_{sg} **do**
  $\xi_{k+1} \leftarrow \xi_k - \sigma \frac{1}{k} \frac{g(\xi_k)}{\|g(\xi_k)\|_2}$;
  if $\Upsilon(\xi_{k+1}) < \Upsilon(\xi_{best})$ **then**
    $\xi_{best} \leftarrow \xi_{k+1}$
  end
end
$\xi \leftarrow \xi_{best}$;

the bounds given in the "Space constraints" section of Table 2.1. The actual orbital elements obtained from [80] for the vector time GMT 2018/064/12:00:00.000 (section "Parameters" of Table 2.1) are employed.

Table 2.1 – Scenarios

Parameters					
a [m]	e	ν_1 [rad]	$\Delta\nu$ [rad]	N	$\overline{\Delta V}$ [m/s]
6777280	0.00039	π	$\pi/2$	5	2

Initial states [m,m/s]							
$X_{01}(\nu_1)$	=	[400,	300,	−40,	0,	0,	0]^T
$X_{02}(\nu_1)$	=	[−800,	600,	200,	0,	0,	0]^T
$X_{03}(\nu_1)$	=	[−1500,	1300,	150,	0,	0,	0]^T
$X_{04}(\nu_1)$	=	[5000,	1300,	500,	0,	0,	0]^T

Space constraints [m]
$\underline{x} = 50,\ \overline{x} = 150,\ \underline{y} = -25,\ \overline{y} = 25,\ \underline{z} = -25,\ \overline{z} = 25$

$\Delta\nu$ represents the true anomaly interval between two impulsive velocity corrections.

The C codes and the data files for the resolution of these scenarios are available at http://homepages.laas.fr/fcamps/CSDP/test_files.zip.

In Table 2.2 and 2.3, the total computation time to solve the optimization problems (Time, in seconds), the fuel consumption (Cons., in meters per second) and the maximal violation of the space constraints (Viol., in meters) for a single run of each simulated case are presented (particularly for ENV, the violation of the constraints is evaluated by computing $|d_0|$ and the maximal value obtained for the envelope functions, presented in Table 2.2). Nine different configurations based on the number of iterations of the BFGS (I_{qn}) and sub-gradient (I_{sg}) methods are adopted for the ENV method: $I_{qn} = \{25, 50, 100\}$ and $I_{sg} = \{250, 500, 1000\}$.

Figure 2.11 depicts the obtained periodic relative trajectory after the application of the 5 impulsive velocity corrections computed by the SDP algorithm departing from the initial state X_{01}.

In Table 2.2, an analysis of the performance of the ENV method, with different choices for I_{qn} and I_{sg} (number of iterations for the BFGS and sub-gradient methods respectively) are presented. One can remark that, as the number of iterations is increased, the violation of the constraints is reduced. For instance, for $I_{qn} \geqslant 100$ and $I_{sg} \geqslant 250$, the hybrid algorithm produced $|d_0| < 10^{-3}$ and no violation of the space constraints for all four proposed scenarios.

Table 2.3 shows that both LP and SDP methods result in a equivalent fuel consumption up to the third decimal case. The ENV method produces higher fuel consumptions for the three first scenarios. However, for the fourth scenario, a small violation of the periodicity constraint

Table 2.2 – Results for the ENV method

X_{01}	ENV								
I_{qn}	25	25	25	50	50	50	100	100	100
I_{sg}	250	500	1000	250	500	1000	250	500	1000
Time	2.064	2.900	4.669	2.181	2.562	3.344	2.164	2.578	3.353
Cons.	0.461	0.461	0.461	0.461	0.461	0.461	0.461	0.461	0.461
$\lvert d_0^+(\nu_N)\rvert$	0.049	0.049	0.049	0.000	0.000	0.000	0.000	0.000	0.000
$\max_w \{g_w(D^+(\nu_N))\}$	$\leqslant 0$	$\leqslant 0$	$\leqslant 0$	$\leqslant 0$	$\leqslant 0$	$\leqslant 0$	$\leqslant 0$	$\leqslant 0$	$\leqslant 0$

X_{02}	ENV								
I_{qn}	25	25	25	50	50	50	100	100	100
I_{sg}	250	500	1000	250	500	1000	250	500	1000
Time	2.048	2.466	3.325	2.328	2.865	3.776	2.334	2.862	3.763
Cons.	1.312	1.314	1.314	1.299	1.299	1.299	1.299	1.299	1.299
$\lvert d_0^+(\nu_N)\rvert$	1.512	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
$\max_w \{g_w(D^+(\nu_N))\}$	$\leqslant 0$	$\leqslant 0$	$\leqslant 0$	$\leqslant 0$	$\leqslant 0$	$\leqslant 0$	$\leqslant 0$	$\leqslant 0$	$\leqslant 0$

X_{03}	ENV								
I_{qn}	25	25	25	50	50	50	100	100	100
I_{sg}	250	500	1000	250	500	1000	250	500	1000
Time	2.064	2.957	4.702	2.196	2.695	3.477	2.186	2.696	3.484
Cons.	1.920	1.920	1.921	1.920	1.920	1.920	1.920	1.920	1.920
$\lvert d_0^+(\nu_N)\rvert$	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
$\max_w \{g_w(D^+(\nu_N))\}$	$\leqslant 0$	$\leqslant 0$	$\leqslant 0$	$\leqslant 0$	$\leqslant 0$	$\leqslant 0$	$\leqslant 0$	$\leqslant 0$	$\leqslant 0$

X_{04}	ENV								
I_{qn}	25	25	25	50	50	50	100	100	100
I_{sg}	250	500	1000	250	500	1000	250	500	1000
Time	2.107	2.987	4.877	2.487	3.188	3.997	2.796	3.455	4.345
Cons.	3.798	3.927	4.110	4.042	4.139	4.139	4.145	4.145	4.145
$\lvert d_0^+(\nu_N)\rvert$	554.910	300.883	42.760	127.105	0.002	0.000	0.000	0.000	0.000
$\max_w \{g_w(D^+(\nu_N))\}$	0.367	$\leqslant 0$	$\leqslant 0$	0.516	$\leqslant 0$	$\leqslant 0$	$\leqslant 0$	$\leqslant 0$	$\leqslant 0$

Table 2.3 – Results for the LP, SDP and ENV methods

X_{01}	LP					SDP	ENV
	N_{disc} =40	80	120	160	200	—	$I_{qn} = 100,\ I_{sg} = 250$
Time	1.156	3.714	7.309	12.228	17.834	5.043	2.164
Cons.	0.402	0.402	0.402	0.402	0.402	0.402	0.461
Viol.	0.152	0.037	0.016	0.009	0.006	0	see Table 2.2

X_{02}	LP					SDP	ENV
	N_{disc} =40	80	120	160	200	—	$I_{qn} = 100,\ I_{sg} = 250$
Time	1.303	4.013	8.423	13.460	20.133	4.661	2.334
Cons.	1.103	1.103	1.103	1.103	1.103	1.103	1.299
Viol.	0.152	0.037	0.016	0.009	0.006	0	see Table 2.2

X_{03}	LP					SDP	ENV
	N_{disc} =40	80	120	160	200	—	$I_{qn} = 100,\ I_{sg} = 250$
Time	0.892	2.815	5.895	10.910	16.353	4.684	2.186
Cons.	1.781	1.781	1.781	1.781	1.781	1.781	1.920
Viol.	0.015	0.015	0.015	0.009	0.006	0	see Table 2.2

X_{04}	LP					SDP	ENV
	N_{disc} =40	80	120	160	200	—	$I_{qn} = 100,\ I_{sg} = 250$
Time	0.941	2.915	5.413	8.703	12.746	3.391	2.796
Cons.	4.204	4.204	4.204	4.204	4.204	4.204	4.145
Viol.	0.154	0.039	0.017	0.010	0.006	0	see Table 2.2

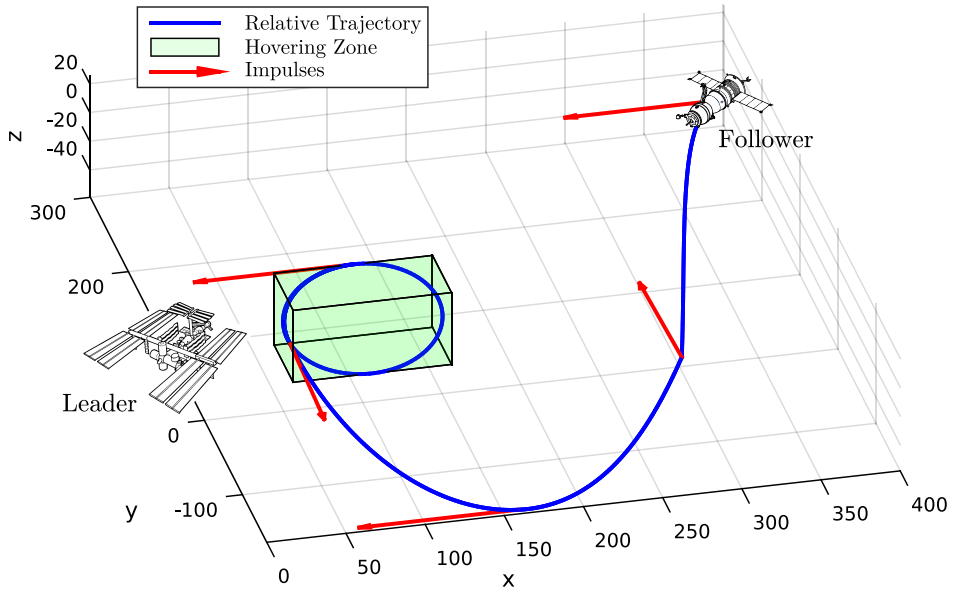


Figure 2.11 – Obtained relative trajectory for the initial state X_{01}, SDP method.

$(0 < |d_0| < 10^{-4})$ produces a lower fuel consumption for the ENV method. This is due to the fact that while attempting to solve the penalized problem, the hybrid optimization algorithm has to account for both the minimization of the fuel consumption and the satisfaction of the constraints (by the minimization of the penalty functions) simultaneously. For the particular case of the resolution of the fourth scenario, the iterations terminated during a phase in which the fuel consumption was prioritized over the periodicity constraint.

Concerning the computation timings, the time spent on the computation of the solution of the SDP problem is of the same order of magnitude of those of the resolution of the LP problem with 80∼120 discrete values of ν. Although the ENV method generally leads to slightly higher fuel consumptions, its timings (for $I_{qn} = 100$, $I_{sg} = 250$) are always lower than those obtained for the SDP approach. This indicates that the SDP approach should be employed in missions in which fuel saving has a high priority status, while the ENV approach should be employed whenever the control of the computing time is a more important aspect.

Another important aspect is the fact that while the approaches SDP and ENV with $I_{qn} \geqslant 100$ and $I_{sg} \geqslant 250$ produce quasi-exact solutions (very small violation of the constraints), the LP approach systematically produces residual violations of the space constraints, even when a high number of discrete values of ν are employed (this is explained by the phenomenon illustrated in Fig. 2.6). Since these systematical violations are not known *a priori* and the timings of both SDP and ENV methods are lower those obtained for LP with $N_{disc} \geqslant 120$, one can conclude that SDP and ENV are more adapted for the resolution of GP.

2.6 Conclusions

In this chapter, possible finite reformulations of the space constraints of the fixed-time impulsive optimal guidance problem for the rendezvous hovering phases are presented. These reformulations allow the numerical resolution of the problem via well-known state-of-the-art optimization libraries and algorithms. The procedure to embed these libraries and algorithms a board dedicated to space applications is also exhibited. The performance of the proposed approaches are compared for four different scenarios. During numerical tests the SDP and the ENV approaches showed a computation time of the order of magnitude of those obtained for the LP method, with the advantage of not systematically violating the space constraints.

In the next chapter, a model predictive control strategy based on the optimization techniques previously discussed in this chapter will be proposed, allowing the convergence of the relative trajectories towards the admissible set of periodic constrained trajectories, even under saturation of the propellers.

Chapter 3

Model predictive control strategy

Contents

3.1 Introduction

In the previous chapters, the guidance problem for the rendezvous hovering phases has been introduced, reformulated and treated from a numerical point of view. However, this guidance problem employs the simplified synthesis model for the relative motion, which does not account for many of the disturbances and uncertainties occurring in the relative dynamics between spacecraft. In other words, the *open-loop* use of the solution obtained by the resolution of the guidance problem is not enough to ensure that the produced relative trajectory is periodic and included in the hovering region.

The goal of this chapter is to propose an original *feedback* control strategy to control the relative motion and keep it periodic and included in a given hovering zone. This adopted control strategy consists in a closed-loop model predictive control (MPC) algorithm [70, 79], which is proven to make the relative movement converge to the desired target region even when

the presence of saturation constraints on controls may make the space window unreachable from the current state. The performance of this proposed control strategy is assessed via simulations that follow the FPGA framework of the tests carried out in [48, 49], where a processor-in-the-loop is implemented: the control computation is executed on a synthesized FPGA LEON3 board certified for spacecraft usage and the propagation of the disturbed relative motion under uncertainties is performed on an external simulator. These tests highlight the efficiency of the proposed control strategy in terms of control quality, numerical burden and rejection of disturbances.

This is a joint work with M. Joldeş, C. Louembet and F. Camps (research engineer, LAAS-CNRS), presented in the paper *Stable Model Predictive Strategy for Rendezvous Hovering Phases Allowing for Control Saturation*, which was submitted to the 2018 AIAA Journal of Guidance Control and Dynamics and is currently in revision (JGCD 2018)

3.2 Impulsive model predictive control

The MPC strategy employs a certain knowledge about the behavior of a system (for example, a state-space model or a state-transition matrix) to predict its evolution and steer it to a certain desired configuration (see Fig. 3.1).

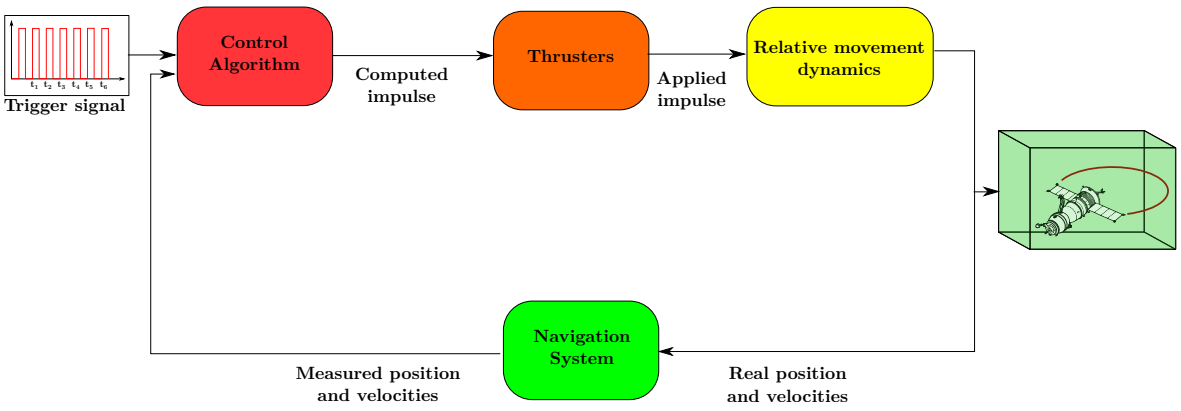


Figure 3.1 – Model predictive control feedback scheme.

The idea is to iteratively solve an optimization problem over a fixed number of time instants (the so-called time horizon). The decision variables are control actions and the criterion to be minimized accounts for the difference between the current state and a reference state. At each iteration, control actions are applied, the current state is updated and an instance of the optimization problem is solved, bringing the system closer to the desired reference configuration. This strategy is illustrated in Fig. 3.2.

The first works about this control strategy date back to the 1960s. It was widely used

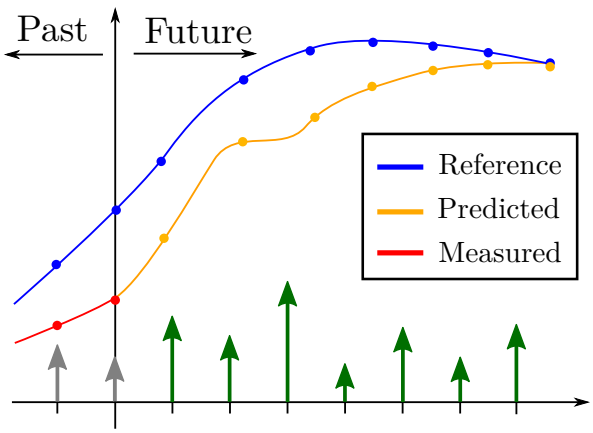


Figure 3.2 – Trajectory evolution and control actions.

for the control of chemical processes during the 1980s and 1990s and, since then, became very popular in many industrial fields and academic research (an overview of the origins and history of MPC is given in [78]).

For the particular case of impulsive systems, different thematics have been using MPC: drug/anesthesia administration [42, 93], networked multi-agent systems [111], spacecraft rendezvous [7, 28] and others. For this class of systems, a theoretical study of properties such as invariance and stability related to MPC strategies are present in the literature [85–87, 96].

In the next section, an MPC strategy is proposed for the rendezvous hovering phases. We prove that the closed-loop system obtained by plugging this control strategy into the system describing the evolution of the relative spacecraft trajectories produces a sequence of relative trajectories that converges to the admissible set. Moreover, it is also proved that for this closed-loop system the hovering region is an invariance set. The proofs are based on the particular structure of the problem (the periodicity and the decoupling of the in and out-of-plane motions) and geometrical arguments.

3.3 The proposed model predictive control strategy

In this section, the model predictive control strategy is presented for problem GP, which is recalled for completeness:

Problem 3.3.1 (Guidance problem)**.** *For a given scenario characterized by eccentricity e, semi-major axis a and initial state $D(\nu_1) = C(\nu_1)T(\nu_1)X(\nu_1) \in \mathbb{R}^6$, given $N \in \mathbb{N}$, find N impulsive controls (represented by $\Delta V \in \mathbb{R}^{3N}$) applied at given fixed instants $\nu_1, \ldots, \nu_N$, such*

that:

$$\min_{\Delta V} \ \mathcal{J}(\Delta V)$$
$$s.t. \begin{cases} D^+(\nu_N) = \Phi_D(\nu_N,\nu_1)D(\nu_1) + \sum_{i=1}^{N} \Phi_D(\nu_N,\nu_i)B_D(\nu_i)\Delta V_i, & \textit{state propagation} \\ |\Delta V_{i,j}| \leqslant \overline{\Delta V}, \quad \forall i \in \{1,\dots,N\} \\ \qquad\qquad\quad \forall j \in \{x,y,z\} & \textit{thruster saturation} \\ D^+(\nu_N) \in S_D & \textit{periodicity + space constraints} \end{cases} \tag{GP}$$

where $\mathcal{J}(\Delta V)$ *is the fuel consumption given in* (1.38)*,* $\Phi_D(\nu_N,\nu_1)$ *is the state-transition matrix given in* (1.41) *and*

$$S_D = \left\{ D \in \mathbb{R}^6 \ s.t. \ d_0 = 0 \ \middle| \ \underline{x} \leqslant M_x(\nu)D \leqslant \overline{x}, \ \underline{y} \leqslant M_y(\nu)D \leqslant \overline{y}, \ \underline{z} \leqslant M_z(\nu)D \leqslant \overline{z}, \ \forall \nu \right\}$$

is the set of admissible trajectories satisfying the periodicity and space constraints.

This controller computes the sequences of impulses generating a sequence of states $(D_k)_{k\in\mathbb{N}}$ that, even under saturation of the actuators, iteratively converges to a state belonging to a given non-empty admissible set S_D. Recall that the admissible set also admits the equivalent reformulations given in (2.5) and (2.13):

$$S_D = \left\{ \ D \in \mathbb{R}^6 \text{ s.t } d_0 = 0 \ \middle| \ \exists Y_l \succeq 0 \text{ s.t } \gamma_l(D) = \Lambda^*(Y_l), \ l \in \left\{\underline{x},\overline{x},\underline{y},\overline{y},\underline{z},\underline{z}\right\} \ \right\},$$
$$S_D = \left\{ \ D \in \mathbb{R}^6 \text{ s.t } d_0 = 0 \ \middle| \ g_w(D) \leqslant 0, \forall w \in \left\{\underline{x},\overline{x},\underline{y},\overline{y},\underline{z},\overline{z}\right\} \right\}.$$

First, the formal version of the algorithm is presented and commented. In a second moment, the main ideas used for its conception are discussed in more details.

3.3.1 Control algorithm

The proposed control algorithm is presented in Alg. 2. This algorithm takes as input a relative state in the LVLH framework $X(\nu_1)$, a saturation threshold $\overline{\Delta V}$, a non-empty admissible set S_D described by $\underline{x},\overline{x},\underline{y},\overline{y},\underline{z},\overline{z}$, a sequence of N true anomaly instants $\nu_1,\dots,\nu_N$ equally spaced by a true anomaly interval τ_I, the true anomaly interval between impulses to generate a periodic trajectory τ_P and the true anomaly interval between sequences of impulses τ_S (these true anomaly intervals are illustrated in Fig. 3.4).

For a given vector of parameters $D(\nu_1)$ and a sequence of impulse dates $\nu_1,\dots,\nu_N$, the functions γ_{xz} and γ_y provide respectively the in-plane and out-of-plane fuel-optimal sequences of impulses generating admissible periodic relative trajectories. The function $\gamma_{\overline{\Delta V}}$ scales down

Algorithm 2: Model predictive control strategy

Require : $X(\nu_1)$, $\overline{\Delta V}$, $S_D \neq \varnothing$, $\tau_S, \tau_P, \tau_I \in \mathbb{R}_{>0}$ s.t. $\forall k \in \mathbb{Z}_{>0}, \tau_I \neq k\pi$, $N \geqslant 3$, $\nu_1, \ldots, \nu_N$ s.t. $\nu_{k+1} = \nu_k + \tau_I$

$D(\nu_1) \leftarrow C(\nu_1)T(\nu_1)X(\nu_1)$;
$\Delta V_{xz}^* \leftarrow \gamma_{xz}(D_{xz}(\nu_1), \nu_1, \ldots, \nu_N)$;
$\Delta V_y^* \leftarrow \gamma_y(D_y(\nu_1), \nu_1, \ldots, \nu_N)$;
// If the saturation is violated by the in-plane impulses
if $\|\Delta V_{xz}^*\|_\infty > \overline{\Delta V}$ **then**
  // If the trajectory is periodic
  if $d_{0_{xz}}(\nu_1) == 0$ **then**
    $\Delta V_{xz}^* \leftarrow \gamma_{\overline{\Delta V}}(\Delta V_{xz}^*)$
  // If the trajectory is not periodic
  else
    $\Delta V_y^* \leftarrow 0$;
    $\Delta V_{xz}^* \leftarrow \gamma_p(d_{0_{xz}}(\nu_1), \nu_1)$;
    if $\|\Delta V_{xz}^*\|_\infty > \overline{\Delta V}$ **then**
      $\Delta V_{xz}^* \leftarrow \gamma_{\overline{\Delta V}}(\Delta V_{xz}^*)$;
    apply impulse ΔV_{xz}^* and ΔV_y^* at ν_1
    $\nu_1 \leftarrow \nu_1 + \tau_P$; // wait τ_P before call algorithm again
    call Algorithm 2 with updated inputs ; // recursive call of the algorithm
// If the saturation is violated by the out-of-plane impulses
if $\|\Delta V_y^*\|_\infty > \overline{\Delta V}$ **then**
  $\Delta V_y^* \leftarrow \gamma_{\overline{\Delta V}}(\Delta V_y^*)$;
apply impulses ΔV_{xz}^* and ΔV_y^* at $\nu_1 \ldots \nu_N$
$\nu_1 \leftarrow \nu_N + \tau_S$; // wait τ_S before call algorithm again
call Algorithm 2 with updated inputs; // recursive call of the algorithm

the vectors of impulses that do not comply with the saturation constraint, keeping their original direction, but changing their magnitude so that their ℓ_∞ norm is equivalent to $\overline{\Delta V}$. These γ-functions will be thoroughly defined and proved consistent in following sections.

The main ideas used in the conception of this control strategy are listed hereafter:

- Systematic computation of impulses that produce periodic orbits;
- Exploitation of the stability of the periodic orbits to stop the evolution and consequent drift of the state;
- Once periodic orbits are obtained, the process of convergence to the admissible set is initiated;
- Only feasible optimization problems are solved in the definition of the functions that return the computed control actions γ_p, γ_{xz} and γ_y;
- The *a posteriori* saturation of the computed control actions does not impede the convergence.

3.3.1.1 Algorithm behavior

The behavior of Algorithm 2 is represented in Figure 3.3 and 3.4. One can observe that two phases appear while converging to the admissible set. First, the state D is steered to the plane $d_0 = 0$. In fact, recalling the transition matrix for the in-plane motion:

$$D_{xz}(\nu) = \underbrace{\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ -3eJ_{\nu_0}(\nu) & 0 & 1 & 0 \\ 3J_{\nu_0}(\nu) & 0 & 0 & 1 \end{bmatrix}}_{\Phi_{D_{xz}}(\nu,\nu_0)} D_{xz}(\nu_0). \tag{3.1}$$

one can remark that if $d_0(\nu_0) = 0$, the state vector D only evolves if $d_0 \neq 0$. This evolution is illustrated in Fig. 3.3 by the trajectories beginning at a black star and finishing at a green triangle in the admissible set. In order to eliminate this "drift" effect and keep the distance between the vector of parameters and the admissible set constant, Algorithm 2 first focuses on bringing d_0 to 0, which is illustrated in Fig. 3.3 by the vertical lines linking red circles and red triangles. The reaching of a periodic trajectory is illustrated by the green triangle on the lower left side of the figure (this state does not freely evolve within time, only under control actions). Once periodicity is obtained, the state is steered to the admissible, which is represented in Fig. 3.3 by the sequence of green triangles.

Moreover, the MPC algorithm produces the pattern of impulsive velocity corrections presented in Fig. 3.4, where the true anomaly intervals τ_P, τ_I and τ_S are shown.

Remark 3.3.1. *The method requires the iterative application of a sequence of at least* 3 *impulsive velocity corrections separated by a true anomaly interval that is not a multiple of* π*. Although for general MPC strategies one single control action is applied by iteration, this approach is adopted because, once periodicity is achieved, at least* 3 *impulsive velocity corrections separated by a true anomaly interval* $\tau_I \neq k\pi$, $k \in \mathbb{N}$ *are sufficient conditions for reaching any state* $D \in \mathbb{R}^6$*, which can be seen as sufficient controllability conditions of the system over the space of periodic orbits [82].*

3.3.1.2 Achieving periodicity using the γ_p function

For the development of the control strategy, a non-saturated version of the guidance problem GP is used. The idea is to obtain an optimization problem which is always feasible, as long as the admissible set of periodic constrained trajectories is not empty. The *ad hoc* function $\gamma_{\overline{\Delta V}}$

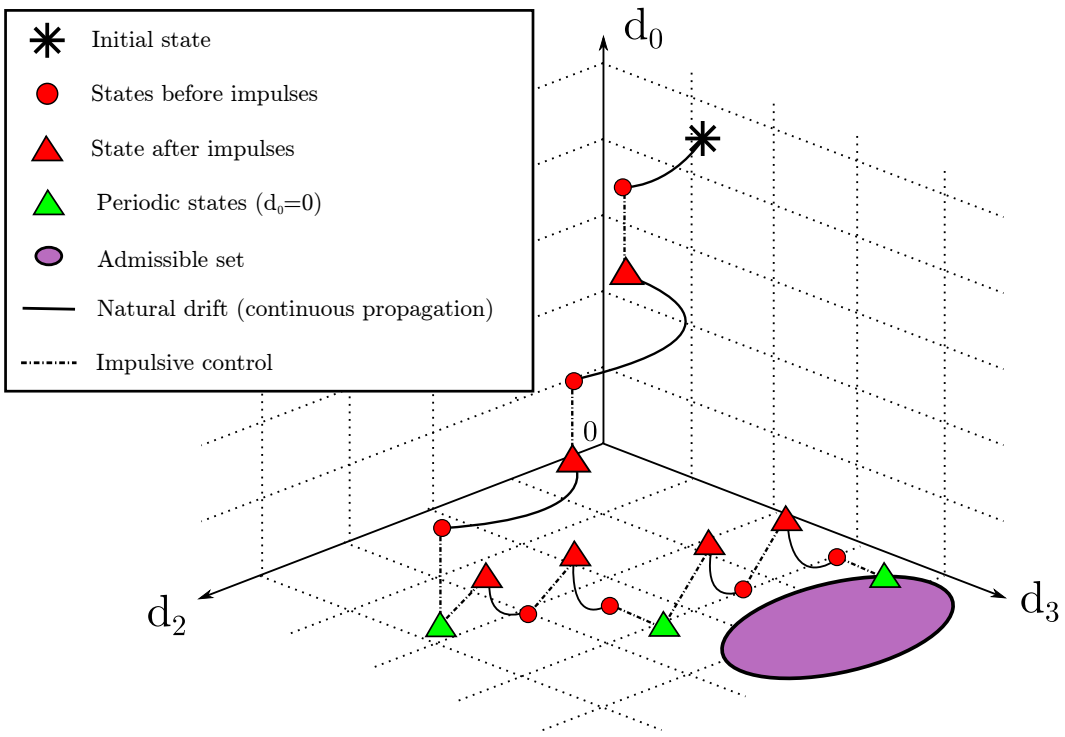


Figure 3.3 – Generation of an admissible periodic trajectory.

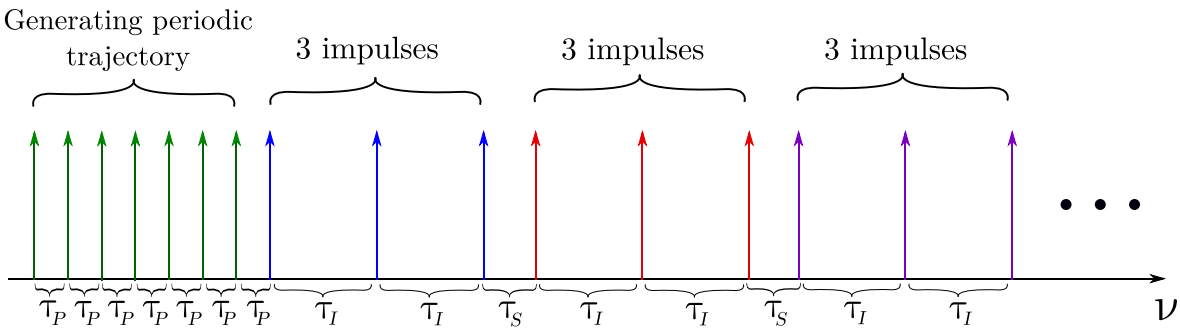


Figure 3.4 – Pattern of impulsive velocity corrections along the true anomaly for a number of impulses $N = 3$.

will account for these saturation constraints in the further developments. The non-saturated problem is formulated as follows:

Problem 3.3.2 (Non-saturated guidance problem)**.** *For a given scenario characterized by eccentricity e, semi-major axis a and initial state $D(\nu_1) = C(\nu_1)T(\nu_1)X(\nu_1) \in \mathbb{R}^6$, given $N \in N$, find N impulsive controls (represented by $\Delta V \in \mathbb{R}^{3N}$) applied at given fixed instants $\nu_1, \ldots, \nu_N$, such that:*

$$\min_{\Delta V} \ \mathcal{J}(\Delta V)$$
$$s.t. \begin{cases} D^+(\nu_N) = \Phi_D(\nu_N, \nu_1)D(\nu_1) + M(\nu_1, \ldots, \nu_N)\Delta V, & \textit{state propagation} \\ d_0^+(\nu_N) = 0 & \textit{periodicity} \\ \underline{x} \leqslant M_x(\nu)D^+(\nu_N) \leqslant \overline{x} & \\ \underline{y} \leqslant M_y(\nu)D^+(\nu_N) \leqslant \overline{y}, \quad \forall \nu & \textit{space constraints} \\ \underline{z} \leqslant M_z(\nu)D^+(\nu_N) \leqslant \overline{z} & \end{cases} \tag{NSGP}$$

where $M(\nu_1, \ldots, \nu_N)\Delta V = \sum_{i=1}^{N} \Phi_D(\nu_N, \nu_i)B_D(\nu_i)\Delta V_i$ represents the effect of the control actions on the propagation of the trajectory.

Remark 3.3.2. *Problem NSGP is always feasible as long as the admissible set $S_D \neq \varnothing$.*

Remark 3.3.3. *Problem NSGP accounts for both the in-plane and out-of-plane dynamics, even though they can be solved separately (these dynamics are split and represented by the functions γ_{xz} and γ_y respectively, defined in the sequel).*

In the sequel, it is demonstrated that the periodicity equality constraint $d_0^+(\nu_N) = 0$ can be systematically satisfied by considering only the impulsive velocity corrections that produce periodic relative trajectories. This idea is a generalization of the developments presented in [22] and consists in constructing a basis for the affine subspace to which the vectors of impulsive velocity corrections such that $d_0^+(\nu_N) = 0$ belong.

In order to do so, the first line of the equation $D^+(\nu_N) = \Phi_D(\nu_N, \nu_1)D(\nu_1) + M(\nu_1, \ldots, \nu_N)\Delta V$ is extracted:

$$d_0^+(\nu_N) = d_0(\nu_1) + \sum_{i=1}^{N} \frac{1}{k^2(e^2-1)} \begin{bmatrix} \rho_\nu & -es_\nu \end{bmatrix} \Delta V_{xz}(\nu_i). \tag{3.2}$$

By defining the row vector

$$M_0(\nu_1, \ \ldots, \ \nu_N) := \frac{1}{k^2(e^2-1)} \begin{bmatrix} \rho_{\nu_1}, & -es_{\nu_1}, & \ldots & \rho_{\nu_N}, & -es_{\nu_N} \end{bmatrix},$$

the expression given in (3.2) can be rewritten as:

$$d_0^+(\nu_N) = d_0(\nu_1) + M_0(\nu_1,\ \ldots,\ \nu_N)\Delta V_{xz}. \tag{3.3}$$

Then ΔV_{xz} is expressed as:

$$\Delta V_{xz}(\lambda) = \underbrace{\begin{bmatrix} | & & | \\ v_1 & \ldots & v_{2N-1} \\ | & & | \end{bmatrix}}_{M_0^{\perp}(\nu_1,\ \ldots,\ \nu_N)} \lambda + \Delta V_0, \tag{3.4}$$

where $\Delta V_0 \in \mathbb{R}^{2N}$ is an arbitrary sequence of in-plane impulses producing a periodic relative trajectory (in other words, a particular solution of (3.3) with $d_0^+(\nu_N) = 0$) and $M_0^{\perp}(\nu_1,\ \ldots,\ \nu_N)\lambda$ represents a linear combination of the v_i vectors belonging to the kernel of the row vector $M_0(\nu_1,\ \ldots,\ \nu_N)$, given by:

$$v_i := [\underbrace{0,\ \ldots\ 0}_{(i-1)\text{ zeros}},\ a_{i+1},\ \ -a_i,\ \ \underbrace{0,\ \ldots\ 0}_{(2N-1-i)\text{ zeros}}\ ]^T, \tag{3.5}$$

where a_i is the i-th entry of $M_0(\nu_1,\ \ldots,\ \nu_N)$.

Now that the periodicity constraint is systematically satisfied, it is demonstrated that there always exist a single optimal impulse generating a periodic relative orbit. This is done by introducing the function γ_p and by demonstrating that it is well-defined.

Proposition 3.3.1 (Optimal periodic impulse)**.** *Consider the function*

$$\begin{aligned} \gamma_p :\quad \mathbb{R}\times\mathbb{R} &\to \mathbb{R}^2 \\ d_{0_{xz}}(\nu_1), \nu_1 &\mapsto \underset{\Delta V_0}{\operatorname{argmin}} \ \|\Delta V_0\|_1 \\ &\qquad s.t. \quad d_{0_{xz}}^+(\nu_1) = d_{0_{xz}}(\nu_1) + M_0(\nu_1)\Delta V_0(\nu_1) = 0 \end{aligned} \tag{3.6}$$

For any set of inputs, the function γ_p is well-defined in the sense that the feasible set of the minimization problem is not empty.

Proof. The line vector $M_0(\nu_1)$ has the following expression:

$$M_0(\nu_1) = (k^2(e^2-1))^{-1}\begin{bmatrix} 1 + e\cos(\nu_1) & -e\sin(\nu_1) \end{bmatrix}$$

and since the term $1 + e\cos(\nu_1) \neq 0,\ \forall \nu_1$ (because $0 < e < 1$), it is always possible to set:

$$\Delta V_0 = k^2(e^2-1)\begin{bmatrix} -\frac{d_{0_{xz}}(\nu_1)}{1+e\cos(\nu_1)} & 0 \end{bmatrix}^T,$$

satisfying the equation $0 = d_{0_{xz}}(\nu_1) + M_0(\nu_1)\Delta V_0$. ■

Remark 3.3.4. *Since the minimization problem in (3.6) contains ℓ_1-norm criteria (which are not strictly convex), infinitely many solutions may exist. In order to enforce uniquenesses, the solution with minimal ℓ_2-norm (which is strictly convex) could have been chosen. However, for the sake of brevity, only one minimum is considered to be arbitrarily chosen in these special cases.*

3.3.1.3 Convergence to the admissible set under the control actions produced by γ_{xz} and γ_y

The functions γ_{xz} and γ_y are responsible for generating optimal non-saturated in-plane and out-of-plane sequences of impulses that generate periodic relative trajectories. They are defined hereafter:

Proposition 3.3.2 (Optimal out-of-plane impulses)**.** *Given $N \geqslant 3$, $\tau_I \in \mathbb{R}_{>0}$ s.t. $\forall k \in \mathbb{Z}_{>0}, \tau_I \neq k\pi$, and $\nu_1, \ldots, \nu_N$ s.t. $\nu_{k+1} = \nu_k + \tau_I$, consider the function:*

$$\begin{array}{rccll} \gamma_y: & \mathbb{R}^2 \times \mathbb{R} \times \ldots \times \mathbb{R} & \to & \mathbb{R}^N & \\ & D_y(\nu_N), \nu_1, \ldots, \nu_N & \mapsto & \operatorname{argmin}_{\Delta V_y} & \|\Delta V_y\|_1 \\ & & & s.t. & D_y^+(\nu_N) = D_y(\nu_1) + M_y(\nu_1, \ldots, \nu_N)\Delta V_y \in S_{D_y} \end{array} \tag{3.7}$$

The function γ_y is well-defined if the feasible set S_{D_y} is not empty. Equivalently, $\forall D_y \in \mathbb{R}^2, D_y^* \in S_{D_y} \neq \varnothing,\ \exists \Delta V_y \in \mathbb{R}^N$ s.t. $D_y^* = D_y + M_y(\nu_1, \ldots, \nu_N)\Delta V_y$.

Proof. The matrix $M_y(\nu_1, \ldots, \nu_N)$ has the following expression:

$$M_y(\nu_1, \ldots, \nu_N) = k^{-2}\begin{bmatrix} -\frac{\sin(\nu_1)}{\rho(\nu_1)} & \cdots & -\frac{\sin(\nu_N)}{\rho(\nu_N)} \\ \frac{\cos(\nu_1)}{\rho(\nu_1)} & \cdots & \frac{\cos(\nu_N)}{\rho(\nu_N)} \end{bmatrix}.$$

This matrix has rank 2, since $\det(M_y(\nu_1, \nu_2)) = \frac{\sin(\nu_2-\nu_1)}{\rho(\nu_1)\rho(\nu_2)} \neq 0$ because of the hypothesis on the impulse dates $\tau_I \neq k\pi$. Then, $\forall D_y, D_y^* \in \mathbb{R}^2$ the vector $\Delta V_y = (M_y^T M_y)^{-1} M_y^T (D_y^* - D_y)$ is well-defined and satisfies $D_y^* = D_y + M_y(\nu_1, \ldots, \nu_N)\Delta V_y$. This is also particularly true if $D_y^* \in S_{D_y} \neq \varnothing$. ■

Proposition 3.3.3 (Optimal in-plane impulses)**.** *Given $N \geqslant 3$, $\tau_I \in \mathbb{R}_{>0}$ s.t. $\forall k \in \mathbb{Z}_{>0}, \tau_I \neq k\pi$ and $\nu_1, \dots, \nu_N$ s.t. $\nu_{k+1} = \nu_k + \tau_I$, consider the function:*

$$\begin{array}{rccll} \gamma_{xz}: & \mathbb{R}^4 \times \mathbb{R} \times \ldots \times \mathbb{R} & \to & \mathbb{R}^{2N} & \\ & D_{xz}(\nu_1), \nu_1, \dots, \nu_N & \mapsto & \operatorname{argmin}_{\Delta V_{xz}} & \|\Delta V_{xz}\|_1 \\ & & & s.t. & D_{xz}^+(\nu_N) = D_{xz}(\nu_1) + M_{xz}(\nu_1, \dots, \nu_N)\Delta V_{xz} \in S_{D_{xz}} \end{array} \tag{3.8}$$

Then the function γ_{xz} is well-defined if the feasible set $S_{D_{xz}}$ is not empty. Equivalently, $\forall D_{xz} \in \mathbb{R}^4,\ D_{xz}^* \in S_{D_{xz}} \neq \varnothing,\ \exists \Delta V_{xz} \in \mathbb{R}^{2N}$ s.t. $D_{xz}^* = D_{xz} + M_{xz}(\nu_1, \dots, \nu_N)\Delta V_{xz}$.

Proof. Consider $\Delta V_{xz} = M_0^{\perp}(\nu_1, \dots, \nu_N)\lambda + \Delta V_0$ as in (3.4). As demonstrated in Proposition 3.3.1 it is always possible to set $d_{0_{xz}}^+$ to 0 with a single impulse. Since $M_0^{\perp}(\nu_1, \dots, \nu_N)\lambda$ has no influence on the first entry of $D_{xz}^* - D_{xz}$, one can conclude that ΔV_0 can be chosen to set the first entry of $D_{xz}^* - D_{xz}$ to any arbitrary value.

Now, it is demonstrated that it is always possible to choose λ that allow the other three entries of $D_{xz}^* - D_{xz}$ to be set to any arbitrary values. Computing $M_{xz}M_0^{\perp}(\nu_1, \dots, \nu_N)$, the following expression is obtained:

$$M_{xz}M_0^{\perp}(\nu_1, \dots, \nu_N) = \frac{1}{k^6(e^2-1)^2} \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & \dots & 0 & 0 \\ \frac{\sin(\nu_1)}{\rho(\nu_1)} & \star & \frac{\sin(\nu_2)}{\rho(\nu_2)} & \star & \frac{\sin(\nu_3)}{\rho(\nu_3)} & \dots & \star & \frac{\sin(\nu_N)}{\rho(\nu_N)} \\ -\frac{\cos(\nu_1)}{\rho(\nu_1)} & \star & -\frac{\cos(\nu_2)}{\rho(\nu_2)} & \star & -\frac{\cos(\nu_3)}{\rho(\nu_3)} & \dots & \star & -\frac{\cos(\nu_N)}{\rho(\nu_N)} \\ \frac{1+\rho(\nu_1)}{\rho(\nu_1)} & \star & \frac{1+\rho(\nu_2)}{\rho(\nu_2)} & \star & \frac{1+\rho(\nu_3)}{\rho(\nu_3)} & \dots & \star & \frac{1+\rho(\nu_N)}{\rho(\nu_N)} \end{bmatrix}.$$

and this matrix has rank 3, since:

$$\begin{aligned} \det\left(\begin{bmatrix} \frac{\sin(\nu_1)}{\rho(\nu_1)} & \frac{\sin(\nu_2)}{\rho(\nu_2)} & \frac{\sin(\nu_3)}{\rho(\nu_3)} \\ -\frac{\cos(\nu_1)}{\rho(\nu_1)} & -\frac{\cos(\nu_2)}{\rho(\nu_2)} & -\frac{\cos(\nu_3)}{\rho(\nu_3)} \\ \frac{1+\rho(\nu_1)}{\rho(\nu_1)} & \frac{1+\rho(\nu_2)}{\rho(\nu_2)} & \frac{1+\rho(\nu_3)}{\rho(\nu_3)} \end{bmatrix}\right) &= -2\frac{\sin(\nu_2-\nu_1)+\sin(\nu_3-\nu_2)-\sin(\nu_3-\nu_1)}{k^{18}(e^2-1)^6\rho(\nu_1)\rho(\nu_2)\rho(\nu_3)} \\ &= -8\frac{\sin\frac{\nu_2-\nu_1}{2}\sin\frac{\nu_3-\nu_2}{2}\sin\frac{\nu_3-\nu_1}{2}}{k^{18}(e^2-1)^6\rho(\nu_1)\rho(\nu_2)\rho(\nu_3)} \neq 0, \end{aligned} \tag{3.9}$$

because of the hypothesis on $\nu_1, \dots, \nu_N$.

This implies that $\forall D_{xz}, D_{xz}^* \in \mathbb{R}^4$ it is possible to chose λ and ΔV_0 in such a manner that $\Delta V_{xz}(\lambda) = M_0^{\perp}(\nu_1, \dots, \nu_N)\lambda + \Delta V_0$ satisfies $D_{xz}^* = D_{xz} + M_{xz}(\nu_1, \dots, \nu_N)\Delta V_{xz}$. This is also particularly true if $D_{xz}^* \in S_{D_{xz}} \neq \varnothing$. ■

Remark 3.3.5. *Although only 2 impulses are needed to generate an out-of-plane admissible*

trajectory, N is set to a number greater or equal to 3 just like for the in-plane case for sake of simplicity.

Remark 3.3.6. *Similarly, in the minimization problems (3.7) and (3.8), one arbitrary minimum is chosen when the problem is not strictly convex.*

Combined with the idea to first reach the periodic orbit set, the convergence of Algorithm 2 is based on the following idea that accounts for eventual saturation that is performed by function $\gamma_{\overline{\Delta V}}$:

Definition 3.3.1 (Rescaling function)**.** *Consider the function*

$$\begin{array}{rccc} \gamma_{\overline{\Delta V}}: & \mathbb{R}^n \backslash \{\vec{0}\} & \to & \mathbb{R}^n \\ & v & \mapsto & \dfrac{\overline{\Delta V}}{\|v\|_\infty}.v, \end{array} \tag{3.10}$$

This function is responsible for scaling down the vector of impulses that do not respect the saturation constraint, keeping its direction, but reducing its magnitude so that its norm is equivalent to $\overline{\Delta V}$.

Now, consider an arbitrary periodic relative trajectory parametrized by $D(\nu_1)$ and a sequence of impulses ΔV^* provided by the functions γ_{xz} and γ_y producing a trajectory $D^*(\nu_N)$ belonging to the admissible set S_D:

$$D^*(\nu_N) = \Phi_D(\nu_N, \nu_1)D(\nu_1) + M(\nu_1, \ldots, \nu_N)\Delta V^* \in S_D.$$

As previously discussed, since $D(\nu_1)$ is a periodic trajectory, $\Phi_D(\nu_N, \nu_1)D_1(\nu_1) = D_1(\nu_1)$, the previous equation can be rewritten as:

$$D^*(\nu_N) = D(\nu_1) + M(\nu_1, \ldots, \nu_N)\Delta V^*.$$

If this sequence of impulses is scaled by a real number η between 0 and 1 (representing the saturation), it generates a vector of parameters $D^\circ(\nu_N)$, given by:

$$D^\circ(\nu_N) = D(\nu_1) + \eta M(\nu_1, \ldots, \nu_N)\Delta V^*.$$

This new vector of parameters $D^\circ(\nu_N)$ is closer to the admissible trajectory $D^*(\nu_N)$ than the original $D(\nu_1)$:

$$\|D^*(\nu_N) - D^\circ(\nu_N)\|_2 \leqslant \|D^*(\nu_N) - D(\nu_1)\|_2,$$

since

$$D^*(\nu_N) - D(\nu_1) = M(\nu_1, \ldots, \nu_N)\Delta V^*$$

and

$$D^*(\nu_N) - D^\circ(\nu_N) = (1-\eta)M(\nu_1, \ldots, \nu_N)\Delta V^*.$$

This idea basically expresses the fact that iterative application of Algorithm 2 produces a sequence of states D that are closer and closer to the admissible set. In the next section, convergence is proved in details.

3.3.2 Proof of convergence and invariance

In this section, the convergence of the previously described control strategy is proven by demonstrating that the iterative application of the command actions computed in Algorithm 2 produces a sequence of states $(D_k)_{k\in\mathbb{N}}$ that converges to an element of S_D. Moreover, the admissible set is proved to be invariant under the action of the proposed controller, which guarantees that the state remains in the admissible set once the convergence is completed.

In Algorithm 2, if saturation constraints are complied, the convergence of the state D to the admissible set S_D is trivial (by definition, one single call of γ_{xz} and γ_y is necessary to produce an admissible state D). However, for the cases in which the magnitude of the computed impulses goes beyond the saturation threshold, the convergence proof is based on a geometrical property of convex sets: the idea is to prove that for a given convex $K \subset \mathbb{R}^n$ and any three elements $A, A', C \in \mathbb{R}^n$ such that $A \in \mathbb{R}^n \backslash K$, $C \in K$ and $A' = C + \lambda(A - C)$, $0 < \lambda < 1$, the distance from A' to K is less than the distance from A to K (in the sense of the ℓ_2 norm):

Proposition 3.3.4. *Let K be a convex set in $\mathbb{R}^n$, $A \in \mathbb{R}^n$ not in K, C a point in K, B the projection of A onto K, $0 < \lambda < 1$, $A' = C + \lambda(A - C)$ and B' the projection of A' onto K. Then, $\underset{K}{\text{dist}}(A') = \|B' - A'\|_2 < \|B - A\|_2 = \underset{K}{\text{dist}}(A)$.*

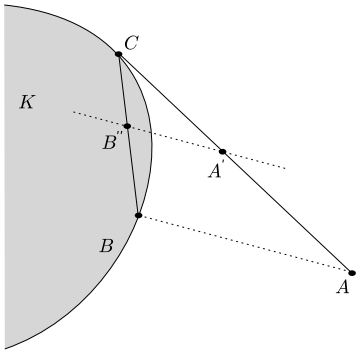


Figure 3.5 – Illustration of points A, A', B, B'', C and convex K

Proof. Consider the point $B^{''} = C + \lambda(B - C)$ (see Figure 3.5). Since K is convex and $B^{''}$ belongs to the segment $\overline{BC}$, $B^{''}$ is a point of K. By developing the difference $B^{''} - A^{'}$:

$$B^{''} - A^{'} = \lambda(B - A) \Rightarrow \|B^{''} - A^{'}\|_2 = \lambda\|B - A\|_2.$$

But since $B^{'}$ is the projection of $A^{'}$ onto K:

$$\|B^{'} - A^{'}\|_2 \leqslant \|B^{''} - A^{'}\|_2 = \lambda\|B - A\|_2 \Rightarrow \|B^{'} - A^{'}\|_2 < \|B - A\|_2.$$

■

The proof of convergence is split in two parts: first Algorithm 2 is demonstrated to bring the out-of-plane related entries D_y to an element of S_{D_y}; then, Algorithm 2 is demonstrated to bring the in-plane related entries D_{xz} to an element of $S_{D_{xz}}$:

3.3.2.1 Convergence of the out-of-plane motion:

In the following proposition, the sequence $(\phi_k)_{k\in\mathbb{N}}$ represents the iterative application of the control Algorithm 2 on the state D_y related to the out-of-plane motion:

Proposition 3.3.5 (Convergence of the out-of-plane trajectory)**.** *Let be $D_y \in \mathbb{R}^2$, $\nu, \tau_I, \tau_S \in \mathbb{R}_{>0}$ s.t. $\forall k \in \mathbb{Z}_{>0}$, $\tau_I \neq k\pi$, $N \geqslant 3$ and $S_{D_y} \neq \varnothing$. Then, the following sequence:*

$$(\phi_k)_{k\in\mathbb{N}} := \begin{cases} \phi_0 = D_y, & \\ \phi_k = \phi_{k-1} + M_y(\nu_k^{(1)}, \ldots, \nu_k^{(N)})\Delta V_{y_k}, & \text{if } \|\Delta V_{y_k}\|_\infty \leqslant \overline{\Delta V} \\ \phi_k = \phi_{k-1} + M_y(\nu_k^{(1)}, \ldots, \nu_k^{(N)})\gamma_{\overline{\Delta V}}(\Delta V_{y_k}), & \text{if } \|\Delta V_{y_k}\|_\infty > \overline{\Delta V} \end{cases},$$

where $\nu_k^{(i)} = \nu + (i-1)\tau_I + (k-1)\tau_S$ and $\Delta V_{y_k} = \gamma_y(\phi_{k-1}, \nu_k^{(1)}, \ldots, \nu_k^{(N)})$, converges to an element of S_{D_y}.

Proof. From Proposition 3.3.2, the function γ_y returns a sequence of impulses that generates an admissible trajectory, i.e. $\phi_k = \phi_{k-1} + M_y(\nu_k^{(1)}, \ldots, \nu_k^{(N)})\Delta V_{y_k} \in S_{D_y}$. Given that, if for some k^* the impulse $\Delta V_{y_{k^*}}$ respects the saturation constraint, then $\forall k \geqslant k^*$, $\phi_k \in S_D$.

However, suppose that the saturation is always violated for any $k \in \mathbb{N}$ (worst case scenario). By writing the expressions of ϕ_k considering the non-scaled and the scaled sequence of impulses:

$$\phi_k^* = \phi_{k-1} + M_y(\nu_k^{(1)}, \ldots, \nu_k^{(N)})\Delta V_{y_k}$$
$$\phi_k = \phi_{k-1} + M_y(\nu_k^{(1)}, \ldots, \nu_k^{(N)})\Delta V_{y_k}\overline{\Delta V}/\|\Delta V_{y_k}\|_\infty$$

By manipulating the previous equations, the following expression are obtained:

$$\phi_k^* - \phi_k \quad = (1 - \overline{\Delta V}/\|\Delta V_{y_k}\|_\infty)(\phi_k^* - \phi_{k-1})$$

From the saturation hypothesis $\overline{\Delta V} < \|\Delta V_{y_k}\|_\infty$ and consequently:

$$\|\phi_k^* - \phi_k\|_2 \quad = (1 - \overline{\Delta V}/\|\Delta V_{y_k}\|_\infty)\|\phi_k^* - \phi_{k-1}\|_2 < \|\phi_k^* - \phi_{k-1}\|_2$$

Since ϕ_k^* belongs to S_{D_y}, which is a convex set, from Proposition 3.3.4, it is possible to conclude that:

$$\underset{S_{D_y}}{\text{dist}}(\phi_k) \leqslant (1 - \overline{\Delta V}/\|\Delta V_{y_k}\|_\infty) \underset{S_{D_y}}{\text{dist}}(\phi_{k-1}) < \underset{S_{D_y}}{\text{dist}}(\phi_{k-1})$$

The following sets are then defined:

$$P_k := \left\{ D \in \mathbb{R}^2 \;\middle|\; \underset{S_{D_y}}{\text{dist}}(D) \leqslant \underset{S_{D_y}}{\text{dist}}(\phi_{k-1}) \right\},$$

$$Q_k := \left\{ \Delta V_y \in \mathbb{R}^2 \;\middle|\; \exists\, \nu \in \mathbb{R},\ \exists D \in P_k \text{ s.t. } \Delta V_y = \gamma_y(D, \nu, \ldots, \nu + (N-1)\tau_I) \right\},$$

and $\Delta V_{y_k}^\star := \max_{\Delta V_y \in Q_k} \|\Delta V_y\|_\infty$. One can remark that since $\underset{S_{D_y}}{\text{dist}}(\phi_k) < \underset{S_{D_y}}{\text{dist}}(\phi_{k-1})$, the sets Q_k form a sequence of inclusions $Q_{k+1} \subseteq Q_k$ and, consequently, $\Delta V_{y_{k+1}}^\star \leqslant \Delta V_{y_k}^\star$. Consider the following two sequences:

$$(a_k)_{k\in\mathbb{N}} := \begin{cases} a_0 = \underset{S_{D_y}}{\text{dist}}(\phi_0), \\ a_k = \alpha\, a_{k-1} \end{cases} \quad \text{and} \quad (b_k)_{k\in\mathbb{N}} := \begin{cases} b_0 = \underset{S_{D_y}}{\text{dist}}(\phi_0) \\ b_k = \underset{S_{D_y}}{\text{dist}}(\phi_k) \end{cases}$$

where $\alpha = (1 - \overline{\Delta V}/\Delta V_{y_1}^\star)$. The sequence $(a_k)_{k\in\mathbb{N}}$ has a general term of the form $a_k = \alpha^k\, a_0$ and converges to zero when k tends to infinite: $\alpha < 1 \Rightarrow a_k \underset{k\to\infty}{\to} 0$. The second sequence represents the distance of the terms of the sequence ϕ_k to the admissible set S_{D_y}. Since the saturation is supposed to be violated, the following inequalities hold:

$$\overline{\Delta V} < \|\Delta V_{y_k}\|_\infty < \Delta V_{y_k}^\star \leqslant \Delta V_{y_1}^\star, \quad \forall k \in \mathbb{N}$$

Then, since $\underset{S_{D_y}}{\text{dist}}(\phi_k) \leqslant (1 - \overline{\Delta V}/\|\Delta V_{y_k}\|_\infty) \underset{S_{D_y}}{\text{dist}}(\phi_{k-1})$ and $\forall k \in \mathbb{N},\ (1 - \overline{\Delta V}/\|\Delta V_{y_k}\|_\infty) < \alpha$, by comparing the sequences $(a_k)_{k\in\mathbb{N}}$ and $(b_k)_{k\in\mathbb{N}}$, it is possible to prove that $b_k \underset{k\to\infty}{\to} 0$, which is equivalent to $\underset{S_{D_y}}{\text{dist}}(\phi_k) \underset{k\to\infty}{\to} 0$.

■

By demonstrating that the sequence $(\phi_k)_{k\in\mathbb{N}}$ converges to a point in the admissible set the MPC strategy is proven to be convergent for the out-of-plane motion.

3.3.2.2 Convergence of the in-plane motion:

If the initial state D_{xz} is not periodic and Algorithm 2 does not produce a sequence of impulses that respects the saturation threshold, the algorithm is executed until the periodic trajectory is generated. In the following proposition, the sequence $(\theta_k)_{k\in\mathbb{N}}$ represents the behavior of the first entry of the state vector D_{xz} during this process:

Proposition 3.3.6 (Convergence to a periodic trajectory)**.** *Let be* $d_{0_{xz}} \in \mathbb{R},\ \nu \in \mathbb{R},\ \tau_P \in \mathbb{R}_{>0}$. *Then, the sequence* $(\theta_k)_{k\in\mathbb{N}}$ *defined by:*

$$(\theta_k)_{k\in\mathbb{N}} := \begin{cases} \theta_0 = d_{0_{xz}}, \\ \theta_k = \theta_{k-1} + M_0(\nu_k)\Delta V_{xz_k}, & \text{if } \|\Delta V_{xz_k}\|_\infty \leqslant \overline{\Delta V} \\ \theta_k = \theta_{k-1} + M_0(\nu_k)\gamma_{\overline{\Delta V}}(\Delta V_{xz_k}), & \text{if } \|\Delta V_{xz_k}\|_\infty > \overline{\Delta V} \end{cases},$$

where $\Delta V_{xz_k} = \gamma_p(\theta_{k-1}, \nu_k)$ *and* $\nu_k = \nu + (k-1)\tau_P$, *converges to* 0.

Proof. The proof is *mutatis mutandis* similar to that presented in Proposition 3.3.5. ■

By demonstrating that the sequence $(\theta_k)_{k\in\mathbb{N}}$ converges to zero, the iterative calls of the MPC strategy is proven to produce a periodic trajectory. Once the trajectory becomes periodic, the behavior of the state D_{xz} under iterative calls of Algorithm 2 can be represented by the sequence $(\varphi_k)_{k\in\mathbb{N}}$ introduced in the following proposition:

Proposition 3.3.7 (Convergence of the in-plane periodic trajectory to $S_{D_{xz}}$)**.** *Let be* $D_{xz} \in \mathbb{R}^4$ *s.t.* $d_{0_{xz}} = 0,\ \nu, \tau_I, \tau_S \in \mathbb{R}_{>0}$ *s.t.* $\forall k \in \mathbb{Z}_{>0},\ \tau_I \neq k\pi,\ N \geqslant 3$ *and* $S_{D_{xz}} \neq \varnothing$. *Then, the following sequence:*

$$(\varphi_k)_{k\in\mathbb{N}} := \begin{cases} \varphi_0 = D_{xz}, \\ \varphi_k = \varphi_{k-1} + M_{xz}(\nu_k^{(1)}, \ldots, \nu_k^{(N)})\Delta V_{xz_k}, & \text{if } \|\Delta V_{y_k}\|_\infty \leqslant \overline{\Delta V} \\ \varphi_k = \varphi_{k-1} + M_{xz}(\nu_k^{(1)}, \ldots, \nu_k^{(N)})\gamma_{\overline{\Delta V}}(\Delta V_{xz_k}), & \text{if } \|\Delta V_{xz_k}\|_\infty > \overline{\Delta V} \end{cases},$$

where $\nu_k^{(i)} = \nu + (i-1)\tau_I + (k-1)\tau_S$ *and* $\Delta V_{xz_k} = \gamma_{xz}(\varphi_{k-1}, \nu_k^{(1)}, \ldots, \nu_k^{(N)})$, *converges to an element of* $S_{D_{xz}}$.

Proof. The proof is *mutatis mutandis* similar to that presented in Proposition 3.3.5. ■

By demonstrating that the sequence $(\varphi_k)_{k\in\mathbb{N}}$ converges to a point in the admissible set, the MPC strategy is proven to be convergent for the in-plane motion.

3.3.2.3 Invariance

So far, only the convergence of the vector of parameters to an element of the admissible set has been established. In the following it is demonstrated that once an admissible trajectory is obtained, it is preserved by the proposed model predictive control algorithm. But first, let notice that the S_D is naturally invariant as a subset of the invariant set of periodic orbits.

Proposition 3.3.8 (Invariance)**.** *The set S_D is invariant under the action of the instructions defined in Algorithm 2.*

Proof. This is evident, since $D \in S_D \Rightarrow d_0 = 0$ (periodicity), the function γ_p is never called; the functions γ_y and γ_{xz} compute the fuel-optimal sequence of impulses that generate a trajectory respecting the out-of-plane and the in-plane space constraints respectively. But since

$$D \in S_D \Rightarrow \underline{x} \leqslant x(\nu) \leqslant \overline{x},\ \underline{y} \leqslant y(\nu) \leqslant \overline{y},\ \underline{z} \leqslant z(\nu) \leqslant \overline{z},\ \forall \nu,$$

the functions will return a sequence of null impulses. ■

3.4 Simulations and results

Hereafter the results obtained by employing the proposed MPC algorithm to control the relative motion between spacecraft during the rendezvous hovering phases are presented.

3.4.1 Processor-in-the-loop

The tests are performed in a processor-in-the-loop environment (see Fig. 3.6): each call of the MPC algorithm is executed on the board dedicated to space applications presented in Section 2.4.1; the computed control actions are sent via a network connection to a computer running the simulation model that integrates the relative motion between spacecraft described in Section 1.2.1.

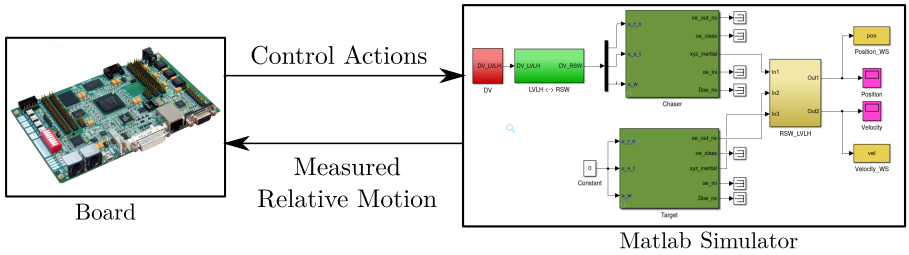


Figure 3.6 – Processor-in-the-loop environment: network connection between board and simulator via user datagram protocol.

3.4.1.1 Software

Both the SDP and the ENV approaches are adopted to model the fuel-optimal control problem. The SDP problems are solved via the CSDP library [16], using the standard options and parameters. The envelope problems are solved by a combination of penalty method with iterative optimization algorithms based on sub-gradients: the constraints are weighted by a coefficient equivalent to 10^8 and added to the objective function and the resulting unconstrained problem is solved by performing $I_{qn} = 50$ iterations of the BFGS method, followed by $I_{qn} = 500$ iterations of the sub-gradient method (see Section 2.4.5 and Appendix E for details). All embedded programs are coded in C.

3.4.1.2 Relative motion

Two types of simulator are used: the linear simulations are performed in order to verify the theoretical results about the convergence provided the proposed algorithm. The non-linear simulations assess the robustness of the proposed algorithm under disturbances and nonlinearities that are not taken into account by the linear model.

1. The linear model representing the relative motion is given by equation (1.41), using the vector of parameters or, equivalently, by equation (SLTH) in Cartesian coordinates;

2. The nonlinear simulator based on the Gauss planetary equations for the relative motion that takes into account the effects of disturbances, such as the atmospheric drag, Earth's oblateness presented in Sec. 1.2.1.

In addition, navigation system uncertainties and errors on the application of control actions are simulated along with the nonlinear simulator. A white noise on position and velocity is added to the integrated relative state (standard deviations: $d_p = 10^{-2}$ m, $d_v = 10^{-5}$ m/s); control mis-execution consist of errors are considered for the orientation and magnitude of the applied impulsive velocity corrections (a mismatch of $\pm 1^\circ$ in orientation and $\pm 1\%$ in magnitude is considered).

3.4.2 Scenarios

In order to compare the obtained results to those presented in [22], the same scenarios (based on the PRISMA mission [15]) are studied: Earth's gravitational constant: $\mu = 3.986004418 \times 10^{14}$; leader's orbital parameters: $e = 0.004$, $a = 7011$ km, $i = 98^\circ$, $\Omega = 0^\circ$, $\omega = 0^\circ$; leader's initial true anomaly $\nu_0 = 0^\circ$; number of impulses adopted is $N = 3$;

true anomaly interval between impulses $\tau_I = 120°$; true anomaly interval between sequence of impulses $\tau_S = 120°$; true anomaly interval to achieve periodicity $\tau_P = 3,6°$; space constraints: $\underline{x} = 50$, $\overline{x} = 150$, $\underline{y} = -25$, $\overline{y} = 25$, $\underline{z} = -25$, $\overline{z} = 25$; propellers saturation threshold: 0.5 m/s; duration of simulation: 10 orbital periods; initial relative state:

$$\begin{array}{rcl} X_{01} & = & [\ \ 400,\ \ 300,\ -40,\ 0,\ 0,\ 0]^T \\ X_{02} & = & [\ -800,\ \ 600,\ \ 200,\ 0,\ 0,\ 0]^T \\ X_{03} & = & [\ -1500,\ 1300,\ \ 150,\ 0,\ 0,\ 0]^T \\ X_{04} & = & [\ \ 5000,\ 1300,\ \ 500,\ 0,\ 0,\ 0]^T \end{array},$$

where the first three components of each vector represent the relative LVLH positions (in meters) and the last three, the relative LVLH velocities (in meters per second).

3.4.3 Convergence definition

To evaluate the convergence, the mismatch ratio η is employed:

$$\eta(\nu) := \frac{\underset{S_D}{\text{dist}}\, D(\nu)}{\underset{S_D}{\text{dist}}\, D(\nu_0)} = \frac{\underset{S_D}{\text{dist}}\, C(\nu)T(\nu)X(\nu)}{\underset{S_D}{\text{dist}}\, C(\nu_0)T(\nu_0)X(\nu_0)}, \tag{3.11}$$

This is the ratio between the distance to the admissible set of the current and the initial vector of parameters. For a given $\delta \in [0,1]$ the convergence time T_c is defined as:

$$T_c(\delta) \in \mathbb{R}_{>0} \text{ s.t. } \forall \nu \geqslant T_c,\ \eta(\nu) \leqslant \delta. \tag{3.12}$$

In the sequel, the convergence time is set to the time T_c at which $\delta = 0.05$.

3.4.4 Consumption, convergence time and running time

Convergence and hovering are obtained for all performed simulations. Table 3.1, presents the obtained fuel consumption $J(\Delta V)$. From this point of view, the SDP-based controller is the best performing with respect to the ENV-based controller. This is due to the fact that the limited number of iterations of the BFGS and sub-gradient algorithms generate suboptimal solutions of ENV, while the SDP approach always returns the optimal solution of SDP (recalling the numbers of iterations: $I_{qn} = 50$ iterations of the BFGS method, followed by $I_{qn} = 500$ iterations of the sub-gradient method). Nevertheless, both approaches engender fuel consumptions that are approximatively half of the lower values produced by any of the three control laws proposed in [22].

Convergence times T_c for each simulations are reported in Table 3.2. The non linear simulation environment has little impact on convergence performances except for the initial condition X_{04}. Comparing with the hybrid controller developed in [22], the proposed approaches are not generally the best. For instance, for X_{01} the control law B in [22] generates a convergence time equal to 0.34 orbits, while the proposed SDP and ENV approaches take twice as much time to converge. This indicates that the proposed strategy gives more emphasis to reducing the consumption than producing short convergence times. Besides, when the initial condition recedes from the hovering zone, the MPC controller abilities to account for input constraints permit to ensure the convergence and limit the convergence time. On the other hand, the behavior of hybrid controllers is degraded in terms of convergence and consumption due to the presence of the saturation (one of the examples of application of the hybrid controller even diverges).

Table 3.1 – Consumption J (m/s)

Initial condition	SDP		ENV	
	LIN	NLIN	LIN	NLIN
X_{01}	0.45	0.47	0.53	0.58
X_{02}	1.20	1.25	1.31	1.32
X_{03}	2.21	2.26	2.31	2.38
X_{04}	4.75	4.71	5.41	6.69

Table 3.2 – Convergence time T_c (number of orbits)

Initial condition	SDP		ENV	
	LIN	NLIN	LIN	NLIN
X_{01}	0.66	0.66	0.64	0.69
X_{02}	0.66	0.66	0.64	0.69
X_{03}	1.65	1.65	1.65	1.65
X_{04}	2.69	1.70	1.67	2.66

Table 3.3 and 3.4 permit one to compare the numerical performance both SDP and ENV based controllers. For both approaches, the average time to compute a sequence of $N = 3$ impulses is lower than 3.0 seconds and the maximal running time is never longer than 4.0 seconds (this time is negligible when compared to the orbital period $T = 2\pi\sqrt{a^3/\mu} \approx 5842$ seconds). Moreover, the amount of memory allocated by the execution of the binaries are 5056 Kbyte for the SDP approach and 5584 Kbyte for the ENV approach - these are reasonable values compared to the available memory of approximatively 90 Mbyte.

Table 3.3 – Average running time (s)

Initial condition	SDP		ENV	
	LIN	NLIN	LIN	NLIN
X_{01}	2.82	2.94	0.30	2.71
X_{02}	2.82	2.86	0.31	2.68
X_{03}	2.83	2.87	0.41	2.52
X_{04}	2.70	2.89	1.54	2.81

Table 3.4 – Maximal running time (s)

Initial condition	SDP		ENV	
	LIN	NLIN	LIN	NLIN
X_{01}	2.93	3.31	3.05	2.91
X_{02}	2.96	3.23	3.07	3.09
X_{03}	2.93	3.05	3.23	3.12
X_{04}	2.79	3.54	3.29	3.37

3.4.5 Relative trajectories, impulses and distance to the admissible set

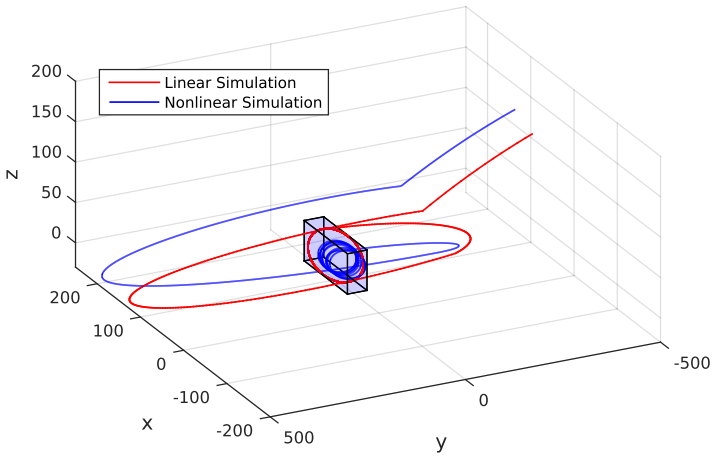
In Figures 3.7 - 3.10, the resulting 3D relative trajectories are exhibited (zoom into the hovering region), the computed and applied impulses for the nonlinear simulations and the evolution of the mismatching ratio $\eta(\nu)$ for the initial conditions X_{03} and X_{04}. By observing the relative trajectories obtained for the linear simulations, one can notice that the relative movement converges to a periodic trajectory included in the hovering zone and, once this trajectory is reached, it remains unchanged - this fact illustrates the convergence and invariance results demonstrated in Prop. 3.3.5 - 3.3.8. The same behavior, however, is not observed for nonlinear simulations: due to the presence of disturbances and uncertainties, the control actions are not able to produce perfect periodic orbits. This is also observed in Fig. 3.11(a) where for the nonlinear simulation, the mismatch ratio oscillates close to zero, but never reaches it. Moreover, although some impulses are saturated (Fig. 3.7(b), 3.8(b), 3.9(b) and 3.10(b)), the convergence is achieved for both linear and nonlinear simulations.

In Fig. 3.11(b) the four initial impulses applied in order to reduce the absolute value of d_0 are shown in details (these impulses are computed via γ_p and are separated by true anomaly intervals of τ_P, indicated in the figure; for nonlinear simulations, due to the disturbances, the condition $d_0 = 0$ is never reached, being therefore replaced by another condition $|d_0| <$ *threshold*). In Fig. 3.12 it is demonstrated that after each sequence of $N = 3$ impulses, the distance to the admissible set decreases (indicated by the dotted lines). Furthermore, during the interval between sequences of impulses (indicated by τ_S and the shaded zones), the mismatch ratio remains constant.

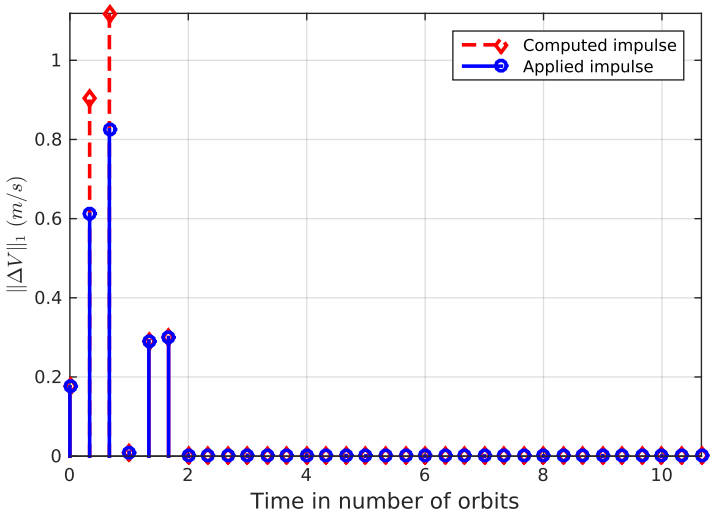
3.4.6 Impact of parameters on fuel consumption

Hereafter, the effect of some parameters (eccentricity, number of impulses, initial true anomaly and the three true anomaly intervals $\tau_P,\ \tau_I,\ \tau_S$) on the total fuel consumption are evaluated. Linear simulations using the SDP approach for the four initial states $X_{01} - X_{04}$ are performed; one single parameter varies at time and the others are kept at the same values employed in the previous simulations. The obtained results are presented in Fig. 3.14 - 3.18.

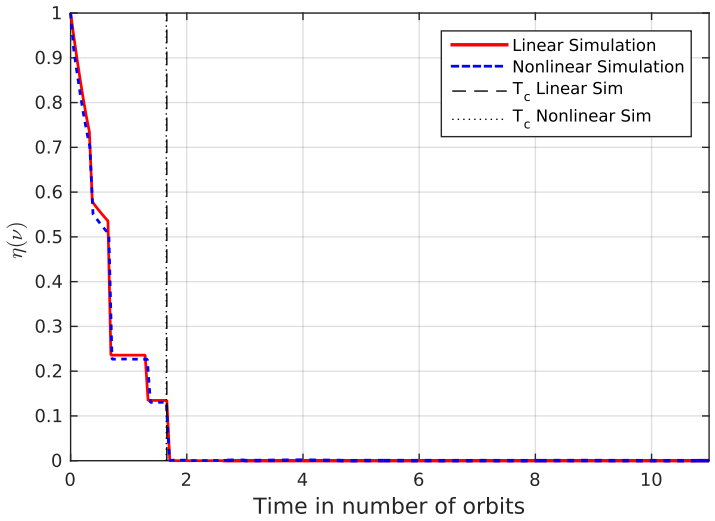
Fig. 3.13 indicates that a small number of impulses should be chosen, since the fuel consumption increases with the growth of this parameter. The increase of the fuel consumption with the eccentricity (Fig. 3.14) or with the reduction of the interval between impulses (Fig. 3.15) are consistent with results previously presented in the literature (similar behavior was observed in [28, Section 6.4]). Different choices of initial true anomaly produce a sinusoidal profile for the fuel consumption, which implies the existence of a fuel-optimal choice for



(a) 3D Relative Trajectory.

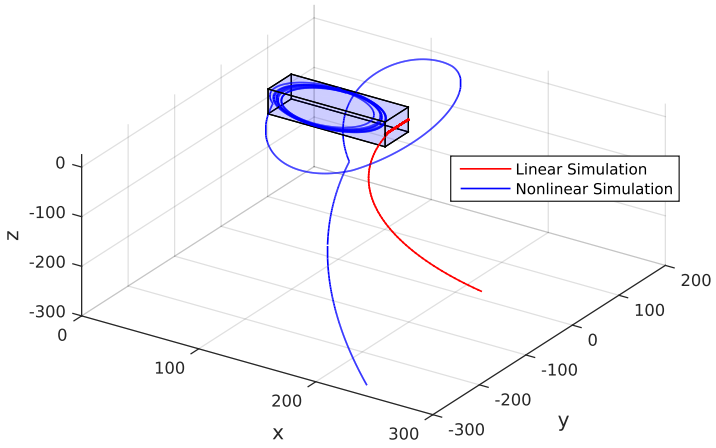


(b) Computed and applied impulses during nonlinear simulation.

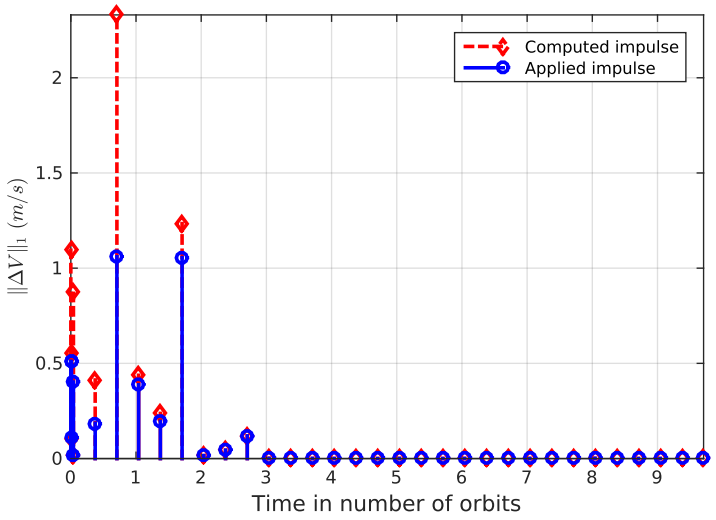


(c) Convergence to admissible set.

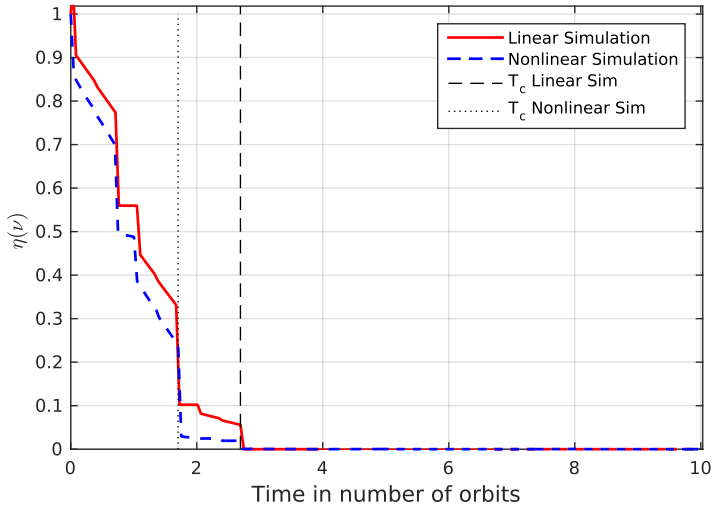
Figure 3.7 – Results for trajectory X_{03} (SDP approach).



(a) 3D Relative Trajectory.

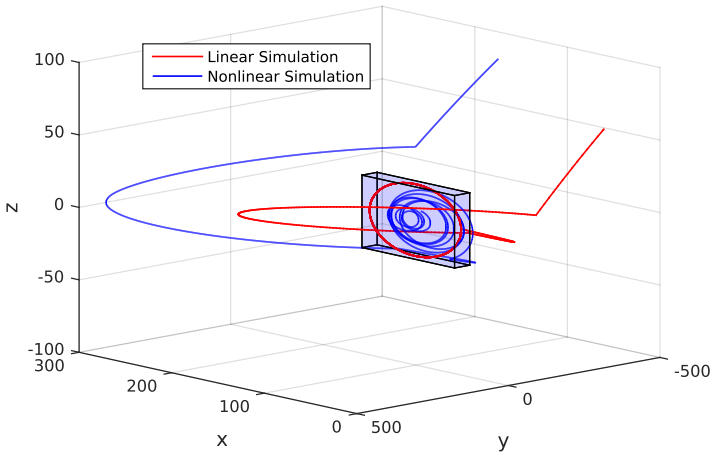


(b) Computed and applied impulses during nonlinear simulation.

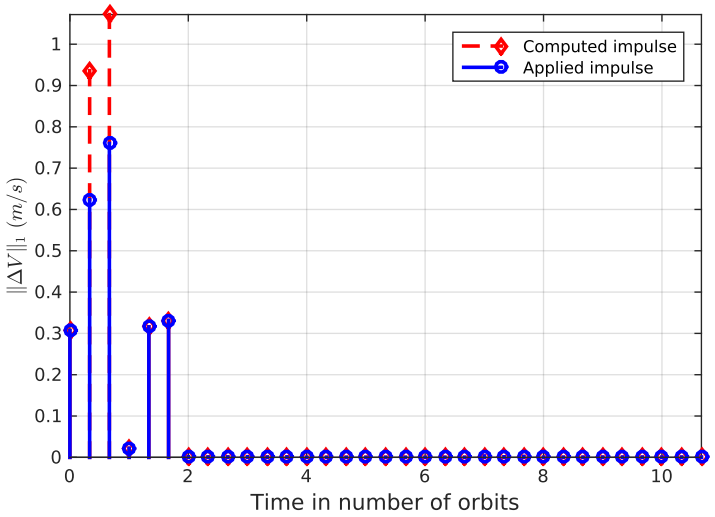


(c) Convergence to admissible set.

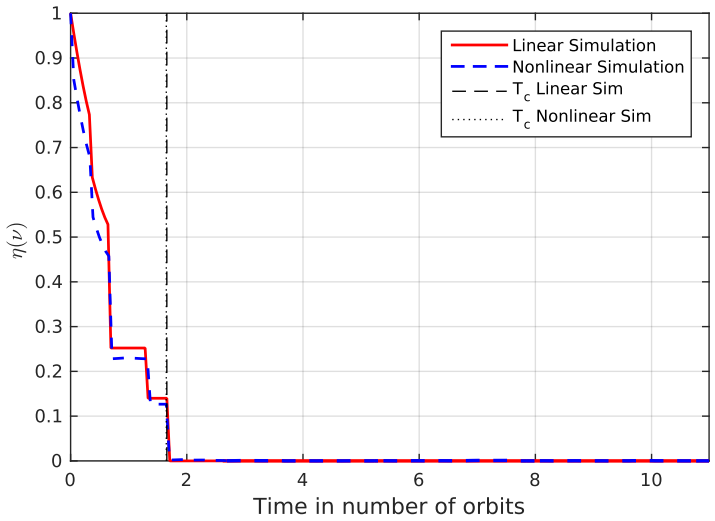
Figure 3.8 – Results for trajectory X_{04} (SDP approach).



(a) 3D Relative Trajectory.

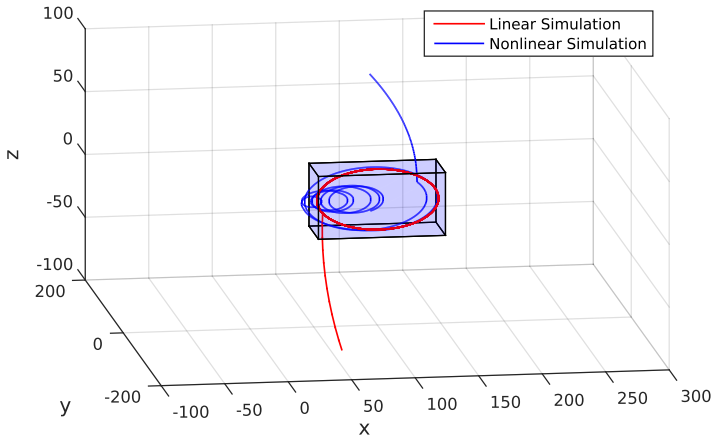


(b) Computed and applied impulses during nonlinear simulation.

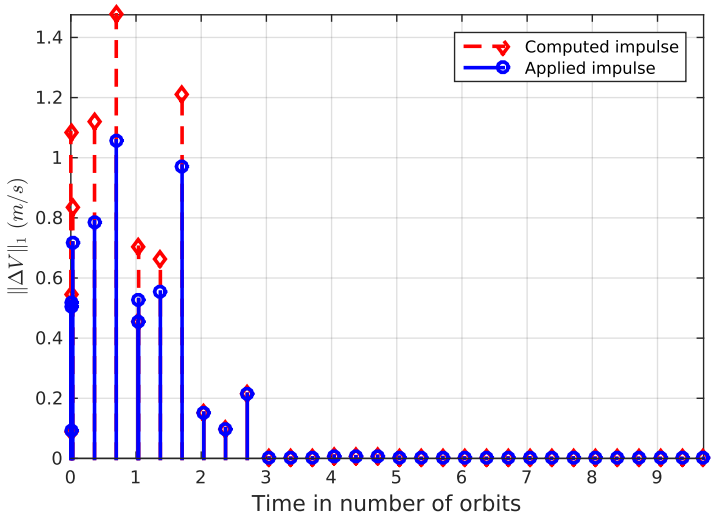


(c) Convergence to admissible set.

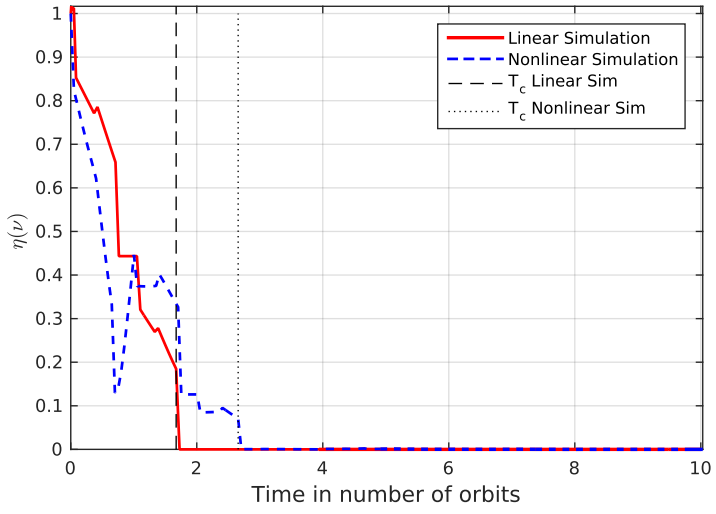
Figure 3.9 – Results for trajectory X_{03} (ENV approach).



(a) 3D Relative Trajectory.

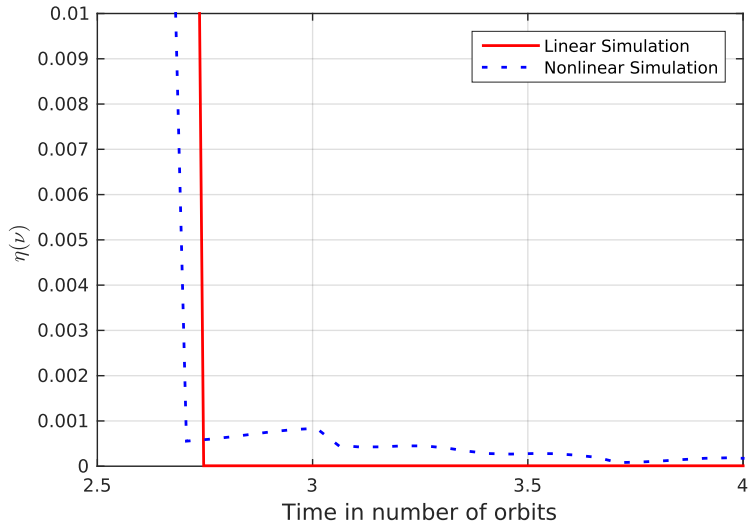


(b) Computed and applied impulses during nonlinear simulation.

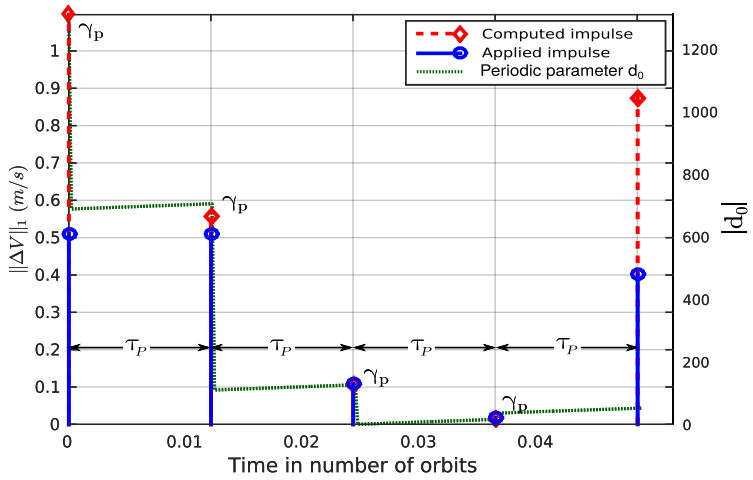


(c) Convergence to admissible set.

Figure 3.10 – Results for trajectory X_{04} (ENV approach).



(a) Effect of disturbances in nonlinear simulations.



(b) Impulses producing a periodic trajectory.

Figure 3.11 – Details of results obtained for trajectory X_{04} (SDP approach).

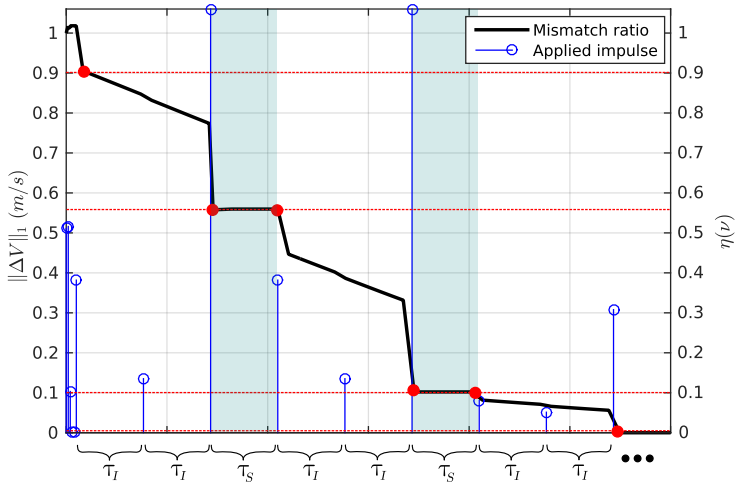


Figure 3.12 – Trajectory X_{04}, linear simulation. Decrease of the mismatch ratio after each sequence of 3 impulses (SDP approach).

the initial firing instant ν_0 (Fig. 3.16). In Fig. 3.17, the consumption increases until it reaches its maximum around $\tau_p = 2.4^\circ$ or 0.042 rad, then starts to decrease; in order to minimize consumption and convergence time, this parameter should be set to the smallest value possible, which is defined by the physical limitations of the spacecraft propellers. The profile of consumption obtained by varying the interval between sequences of impulses does not present a particular shape or behavior and therefore no general conclusion can be obtained from it (Fig. 3.18).

Remark 3.4.1. *In Fig. 3.16 - 3.18, for each initial state, the fuel consumptions are normalized between 0 and 1.*

3.5 Conclusions

In this chapter, an original model predictive control strategy is proposed for the fixed-time impulsive control of the spacecraft rendezvous hovering phases. A theoretical convergence proof is provided, demonstrating that, even when the saturation of the propellers is taken into account, the proposed strategy produces a sequence of control actions generating a periodic relative trajectory included in the hovering region.

Processor-in-the-loop simulations using a LEON3 synthesized microprocessor reveal that although the proposed approach may produce greater convergence times, it is more efficient with respect to fuel consumption than other methods proposed in the literature. Moreover, the timings obtained during these tests bring out the fact that this approach can be efficiently embedded in space dedicated devices. An analysis of the impact of the parameters rendezvous scenarios on the fuel consumption is also presented.

Although robustness is not demonstrated in this dissertation, the results show that, in practice, the proposed control method is capable of providing convergence even under the presence of disturbances. A theoretical evaluation of this robustness property should be carried in future works.

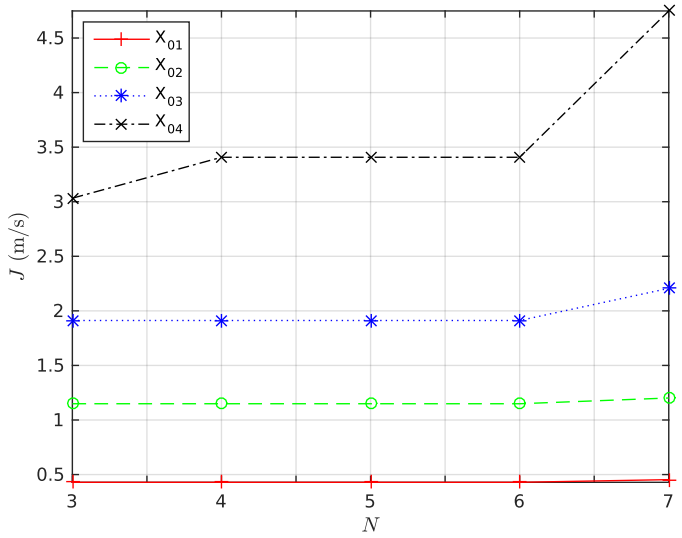


Figure 3.13 – Impact of number of impulses on fuel consumption.

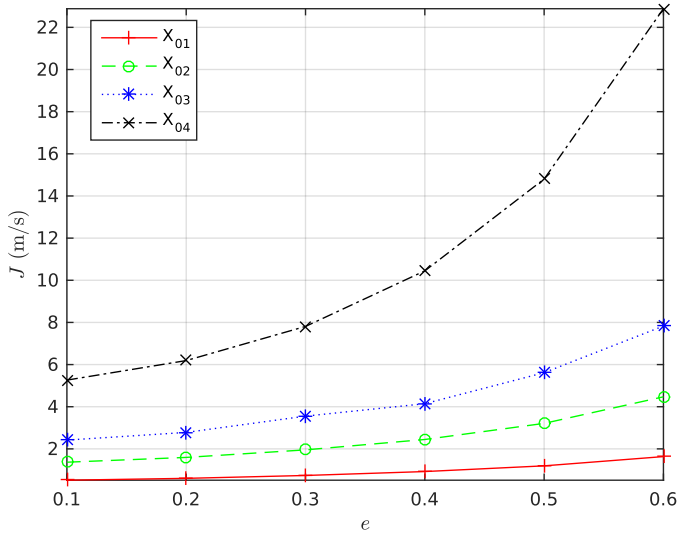


Figure 3.14 – Impact of eccentricity on fuel consumption.

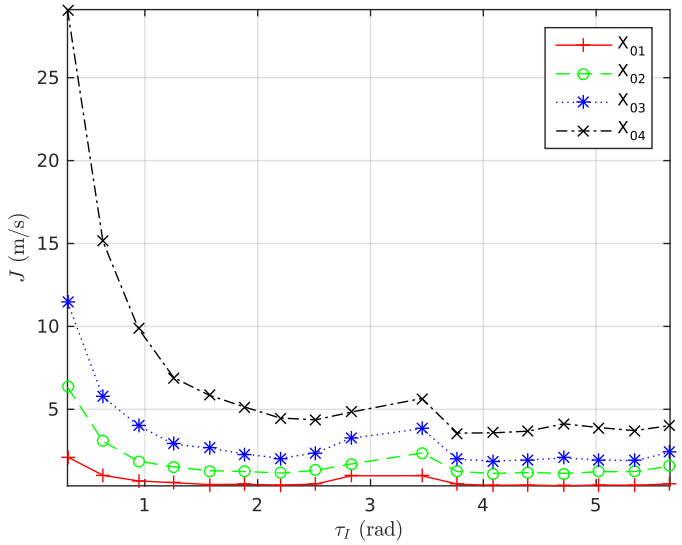


Figure 3.15 – Impact of true anomaly interval between impulses on fuel consumption.

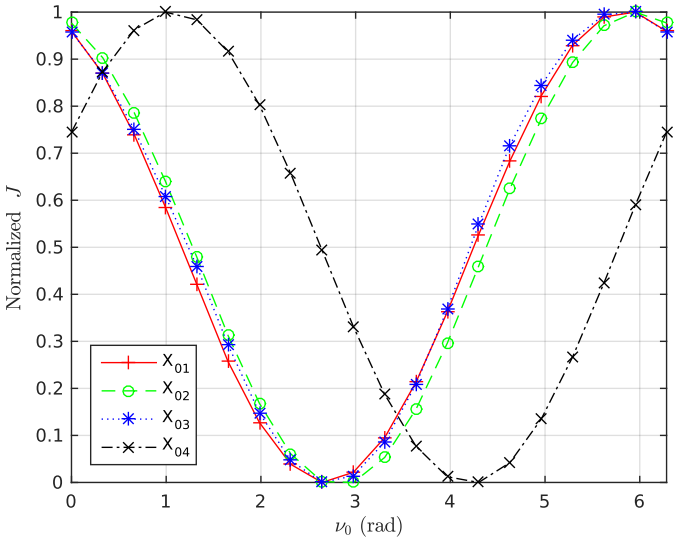


Figure 3.16 – Impact of initial true anomaly on fuel consumption.

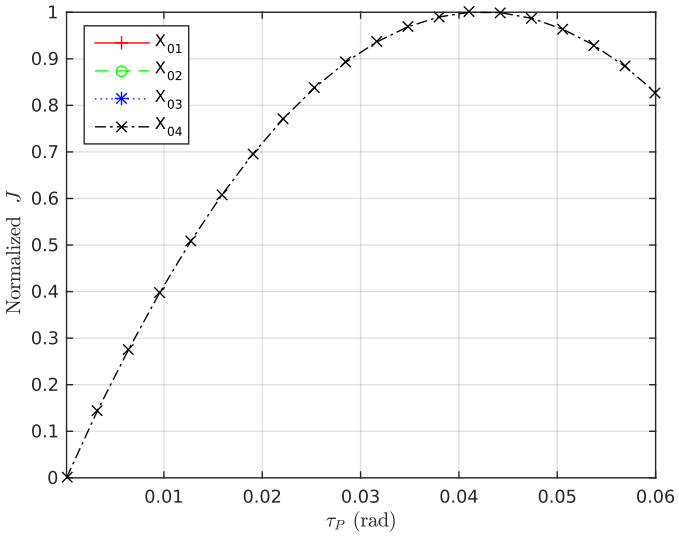


Figure 3.17 – Impact of true anomaly interval between impulses to generate peridic trajectories on fuel consumption

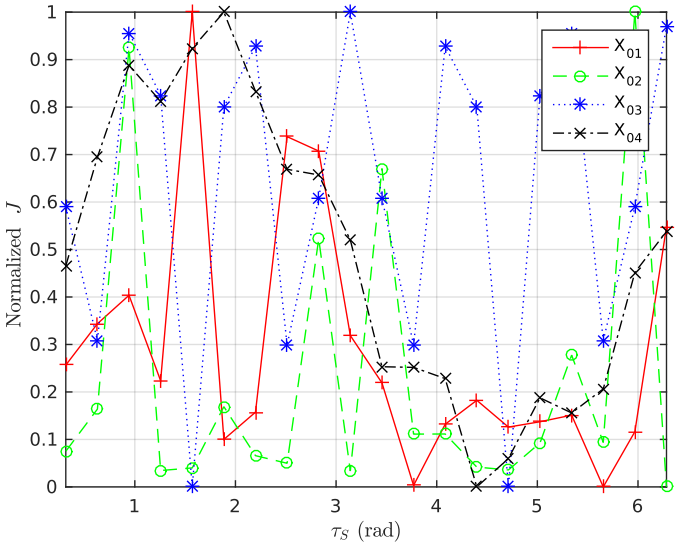


Figure 3.18 – Impact of true anomaly interval between sequences of impulses on fuel consumption

Chapter 4

Validated optimal guidance problem

Contents

4.1 Introduction

The previous chapters treated the problematic of modeling, simulating and conceiving guidance and control algorithms for the relative motion between spacecraft during the rendezvous hovering phases. Another important aspect is the validation of numerical results obtained from such on-board executed algorithms. In fact, during guidance and control procedures of orbiting spacecraft, the respect of positioning and space constraints is decisive for successful mission achievement. Since result accuracy is essential for these procedures, the prevention and estimation of errors arising from approximations and numerical computations become critical.

In this context, this chapter aims to provide a symbolic-numerical method for validating the solutions generated by the previously presented guidance and control algorithms.

The idea is to use the technique presented in [20, 21] to provide rigorous polynomial approximations (RPA) for the solutions of the linear ordinary differential equations (LODE) describing the linearized spacecraft relative motion. Specifically, RPA are polynomials together with an error bound accounting for both approximation and rounding errors and can be obtained using Chebyshev series.

This approach is chosen instead of conceiving validated numerical iterative schemes for the integration of the LODE (like Euler or Runge-Kutta), because the number of needed evaluation points used in these schemes can be prohibitive and the discretization error is difficult to estimate precisely. Moreover, for guidance and control laws design purposes, analytical solutions are preferable, since various constraints (such as saturation, restricted space regions, etc.) can be satisfied on continuous time domains and not only on discretization grids.

The choice for Chebyshev expansions instead of any other basis is motivated by recent works that highlight the advantage of using Chebyshev series expansions in orbital mechanics [89] and by the fact that they started to successfully replace the classical Taylor series expansions based algebra for intrusive approaches, which has already many applications to astrodynamics and optimal control for proximity operations [32, 65, 66], proving to be very efficient and accurate.

This is a joint work with F. Bréhard (PhD student, LAAS-CNRS) and C. Gazzino (post-doctoral researcher, Technion, Israel Institute of Technology), under the title of *Validated Semi-Analytical Transition Matrices for Linearized Relative Spacecraft Dynamics via Chebyshev Series Approximations* [6], presented at the 28th Space Flight Mechanics Meeting of the AIAA SciTech Forum (AIAA 2018).

This chapter is organized as follows: first, a brief introduction on RPA and Chebyshev truncated series is presented; then, the main ideas of the technique for obtaining RPAs of the solutions of LODEs are exhibited; this technique is then used on the linearized Tschauner-Hempel equations, producing a rigorous approximated state-transition matrix with polynomial entries for the propagation of the spacecraft relative motion; this state-transition matrix is then used in the conception of a guidance algorithm for the hovering phases of the rendezvous; finally, a scenario is simulated to illustrate the application of the guidance algorithm and validation technique.

4.2 Rigorous approximations and Chebyshev polynomials

Hereafter some definitions and concepts are exhibited in order to provide the reader with some mathematical background for the developments that will be presented in the sequel.

The aim is to provide a validation (an error-bounded approximation) of the solutions of a system of linear ordinary differential equations representing the relative motion between spacecraft. In a more generic framework, the following system can be considered:

$$X^{(r)}(t) + A_{r-1}(t)X^{(r-1)}(t) + \cdots + A_1(t)X'(t) + A_0(t)X(t) = G(t), \qquad \text{(d-LODE)}$$

where the time variable belongs to the compact interval $[t_0, t_f]$, of unknown $X : [t_0, t_f] \to \mathbb{R}^d$, $A_i : [t_0, t_f] \to \mathbb{R}^{d\times d}$ and $G : [t_0, t_f] \to \mathbb{R}^d$ are at least Lipschitz-continuous.

Hereafter, a *spectral method* is adopted for validating the solutions of d-LODE. This class of methods provide error-bounded approximations for the solutions of LODE over the global time interval (in this case $[t_0, t_f]$), given as the linear combination of well chosen basis functions, whose coefficients have to be computed [18, 43]. This choice is motivated by the fact that, in situations where functions are smooth enough, spectral methods have the advantage over iterative ones of providing a smooth approximation of the solution over the continuous time range, which can be easily derived or integrated, for example.

For efficiency reasons among others, one often chooses families of polynomials for the basis functions, since addition and multiplication composing them are the basic operations implemented in floating-point units of processors. Besides that, the recent advances in polynomial based optimization methods allowed for very efficient solutions in optimal control problems (see [26, 36, 62] and references therein). This polynomial framework is adopted hereafter.

Since a spectral method providing a representation of the approximated solution on a polynomial basis is employed, the term *rigorous polynomial approximation* (adapted from [59]) is employed in the sequel:

Definition 4.2.1 (Rigorous polynomial approximation - RPA)**.** *Let f be a function belonging to some specified function class Ω over a given interval $[a, b]$ and let P be a specified family of polynomials with coefficients (in a given basis) exactly representable in some specified format F. If $p \in P$ together with a "good" bound B are such that $||f - p|| \leqslant B$, where $||\cdot||$ is an appropriate norm*[1]*, then the couple (p, B) is a* rigorous polynomial approximation *(RPA) of the function f.*

[1] the sup norm $\|\cdot\|_\infty$ or the $\mathcal{H}^1$-norm $\|\cdot\|_{\mathcal{H}^1}$ defined later in this section in (4.1).

The approximation problem that motivates this chapter is then formally formulated as follows:

Problem 4.2.1 (RPAs of d-LODE solutions)**.** *Consider the system d-LODE. Let* $\Phi(t,t_0)$ *be its exact state-transition matrix. Then, provide an approximate transition matrix* $\widetilde{\Phi}(t,t_0)\in\mathbb{R}^{rd\times rd}$ *whose entries are polynomials and rigorous error bounds* $\epsilon_{ij}\geqslant 0$ *(*$1\leqslant i,j\leqslant rd$*) satisfying:*

$$|\widetilde{\Phi}_{ij}(t,t_0)-\Phi_{ij}(t,t_0)|\leqslant\varepsilon_{ij},\qquad 1\leqslant i,j\leqslant d,\quad t\in[t_0,t_f].$$

Each pair $(\widetilde{\Phi}_{ij},\epsilon_{ij})$ *is denoted a Rigorous Polynomial Approximation (RPA) of* Φ_{ij}*.*

In this work, the Chebyshev polynomials are chosen. This choice is motivated by the fact that this family of orthogonal polynomials has very convenient algebraic and approximation properties. At first glance, working with polynomials in the standard monomial basis and approximating functions with their Taylor development seems to be a convenient choice. In practice, this method goes along with some shortcomings (approximation of non-smooth functions, limited convergence radius due to complex singularities, numerical instability, etc.). For these reasons, Chebyshev polynomials are preferable in the general case [18].

Hereafter, some definitions and properties related to Chebyshev polynomials are given:

Definition 4.2.2 (Chebyshev polynomials)**.** *The polynomials defined by the three-term recurrence*

$$T_{n+2}=2XT_{n+1}-T_n,$$

with initial terms $T_0=1$ *and* $T_1=X$ *are the so-called Chebyshev polynomials.*

Definition 4.2.3 (Chebyshev coefficients)**.** *Let* f *be a continuous function over the interval* $[-1,1]$*. Then, its Chebyshev coefficients are given by:*

$$[f]_0=\frac{1}{\pi}\int_0^\pi f(\cos\vartheta)d\vartheta,\qquad [f]_n=\frac{2}{\pi}\int_0^\pi f(\cos\vartheta)\cos(n\vartheta)d\vartheta\quad(n\geqslant 1).$$

Definition 4.2.4 (Truncated Chebyshev series)**.** *Let* f *be a continuous function over the interval* $[-1,1]$*. Then, a truncated Chebyshev series of* f *is given by:*

$$f^{[n]}=\sum_{i=0}^{n}[f]_iT_i,$$

which is simply the orthogonal projection of f *onto the finite-dimensional subspace spanned by* $T_0,\ldots,T_n$*.*

Analogously to Fourier series, Chebyshev series have excellent approximation properties [18]. For example, if f is of class $\mathcal{C}^r$ over $[-1,1]$ with $r \geqslant 1$, then $f^{[n]}$ uniformly converges to f in $O(n^{-r})$. Moreover, at fixed degree n, the n-th truncated Chebyshev series $f^{[n]}$ is a near-best approximation of f among degree n polynomials, with a factor growing relatively slowly, in $O(\log(n))$ [76].

Using these convergence results, one can easily identify a sufficiently smooth function space with the space of corresponding Chebyshev coefficients. Let Ч^1 be the Banach space of continuous functions with absolutely summable Chebyshev series, and define the associated norm $\|f\|_{\text{Ч}^1} = \sum_{i\geqslant 0} |[f]_i|$. Then, a Banach algebra structure is obtained, since $\|fg\|_{\text{Ч}^1} \leqslant \|f\|_{\text{Ч}^1}\|g\|_{\text{Ч}^1}$. Moreover, this norm is a safe overestimation of the supremum norm $\|\cdot\|_\infty$ over $[-1,1]$:

$$\|f\|_{\text{Ч}^1} = \sum_{i\geqslant 0} |[f]_i| \geqslant \sup_{-1\leqslant t\leqslant 1} \sum_{i\geqslant 0} |[f]_i T_i(t)| \geqslant \sup_{-1\leqslant t\leqslant 1} |f(t)| = \|f\|_\infty. \tag{4.1}$$

4.3 The approximation method

Now that a background on RPA and Chebyshev polynomials has been presented, the approximation method employed in this chapter is briefly explained. It is described in detail in [20, 21] and its experimental C source code is available at `http://perso.ens-lyon.fr/florent.brehard`.

For the sake of clarity, the one-dimensional case for Problem 4.2.1 is considered, that is $d = 1$:

$$x^{(r)}(t) + a_{r-1}(t)x^{(r-1)}(t) + \cdots + a_1(t)x'(t) + a_0(t)x(t) = g(t). \tag{4.2}$$

over $[t_0, t_f] = [-1, 1]$ (up to a rescaling of the independent variable, if necessary), together with prescribed initial values at -1:

$$x(-1) = v_0, \quad x'(-1) = v_1, \quad \ldots, \quad x^{(r-1)}(-1) = v_{r-1}, \tag{4.3}$$

where $a_i, g : [-1,-1] \to \mathbb{R}$ are functions in Ч^1 approximated by a truncated Chebyshev series, and $x : [-1,1] \to \mathbb{R}$ is the unknown function which will be approximated with a truncated Chebyshev series. For this, a common solution is to rephrase the differential equation (4.2) into an equivalent integral equation. For instance, as detailed in [20, 21], one considers $\varphi = x^{(r)}$ as the unknown function and expresses lower-order derivatives of x as integrals of φ.

This gives:

$$\varphi + \boldsymbol{K} \cdot \varphi = \psi, \qquad \text{where} \quad \boldsymbol{K} \cdot \varphi(t) = \int_{-1}^{t} k(t,s)\varphi(s)ds, \tag{4.4}$$

where $k(t,s)$ is a bivariate polynomial computed from the polynomials $a_i(t)$ and $\psi(t)$ is a univariate polynomial computed from the second member $g(t)$, the polynomials $a_i(t)$ and the initial values v_i (details in [21]). A symbolic computation shows that for $i \in \mathbb{N}$, $\boldsymbol{K} \cdot T_i$ is a polynomial with non-zero Chebyshev coefficients between indices 0 and h (initial coefficients) and between $i-d$ and $i+d$ (diagonal coefficients), where the bandwidths h and d directly depend on the maximum degree of the a_i. Hence, the operator $\boldsymbol{K} : \mathrm{Ч}^1 \to \mathrm{Ч}^1$ has a so-called almost-banded structure [20, 21, 83] in the Chebyshev basis, which is depicted on Figure 4.1.

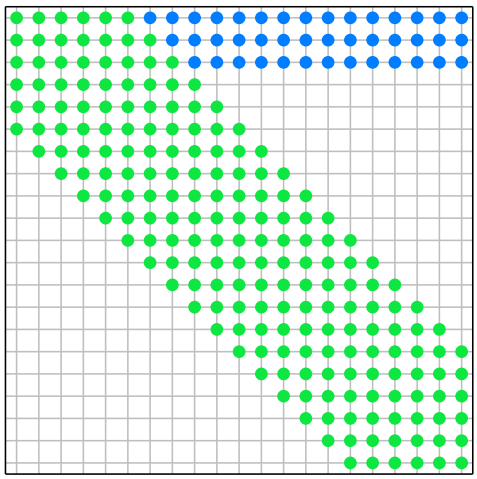


Figure 4.1 – Matrix representation of $\boldsymbol{K}$ in Chebyshev basis, truncated at order 20. Almost-banded structure given by initial coefficients (blue) and diagonal ones (green)

Following the general scheme of spectral Galerkin methods [18], this problem is projected onto a finite dimensional space by taking the truncated operator $\boldsymbol{K}^{[n]} = \Pi_n \cdot \boldsymbol{K} \cdot \Pi_n$ where Π_n is the orthogonal projection from $\mathrm{Ч}^1$ to the finite-dimensional space spanned by $T_0, \ldots, T_n$. Now, it remains to determine the $n+1$ first (approximated) Chebyshev coefficients of φ by solving the following finite-dimensional problem:

$$\varphi + \boldsymbol{K}^{[n]} \cdot \varphi = \psi.$$

Such an almost-banded system is efficiently solved using the algorithm presented in [83]. The mathematical statements and proofs establishing the uniqueness of the solution and the exponential convergence of the numerical truncated solutions to the exact one are to be found in [20, 21].

4.3.1 Principles of the validation method

The goal of validating techniques is to provide effective and rigorous error bounds for given approximations. For the proposed method, an *a posteriori validation method* is employed. This validation works in two steps: first, the user provides an approximation of the solution, obtained with the procedure of his choice; then, the validation method computes a rigorous error bound without knowing how this approximation was built.

The paradigm of *a posteriori* validation methods is particularly well suited for spectral methods. Since it was explained in the previous section how to compute approximate solutions for LODE (4.2), it can now be considered that some approximating truncated Chebyshev series is given and focus on the validation method itself. A wide majority of *a posteriori* validation methods are based on the Banach fixed-point theorem and variations around it. This theorem can be stated in this simplified version:

Theorem 4.3.1. *Let $(E, \|\cdot\|_E)$ be a Banach space and $\boldsymbol{T} : E \to E$ an affine operator whose linear part $\mathcal{D}\boldsymbol{T}$ is a bounded linear endomorphism. If $\boldsymbol{T}$ is contracting, that is, if $\|\mathcal{D}\boldsymbol{T}\|_E = \mu < 1$, then it admits a (necessarily unique) fixed point $x^* \in E$, which is a solution of the equation:*

$$\boldsymbol{T} \cdot x = x, \tag{4.5}$$

and for a given $x \in E$, the following enclosure holds for its distance to x^*:

$$\frac{\|x - \boldsymbol{T} \cdot x\|_E}{1 + \mu} \leqslant \|x - x^*\|_E \leqslant \frac{\|x - \boldsymbol{T} \cdot x\|_E}{1 - \mu}. \tag{4.6}$$

Hence, designing a fixed-point based validation method for a linear problem of the form $\boldsymbol{F} \cdot x = y$ essentially boils down to rephrasing it as a fixed-point equation $\boldsymbol{T} \cdot x = x$ for some contracting affine operator $\boldsymbol{T}$, which has to be explicitly computable (in order to bound $\|x - \boldsymbol{T} \cdot x\|_E$) and whose operator norm can be effectively upper-bounded so to obtain a rigorous $\mu < 1$.

A rather generic way to design such a contracting method is to use an adaptation of Newton's method to find zeros of maps [108], which can be used even for non-linear problems. Here the idea in the linear case is sketched. Consider the equation $\boldsymbol{F} \cdot x = y$, where $\boldsymbol{F}$ is a linear automorphism. Let $\boldsymbol{A}$ be an approximation of its inverse $\boldsymbol{F}^{-1}$. Then the unique

solution is also the unique fixed-point of $\boldsymbol{T}$ defined by:

$$\boldsymbol{T} \cdot x = x - \boldsymbol{A} \cdot (\boldsymbol{F} \cdot x - y), \tag{4.7}$$

as soon as $\boldsymbol{A}$ is injective. The underlying idea is that if $\boldsymbol{A}$ is sufficiently close to the inverse of $\boldsymbol{F}$, then $\boldsymbol{T}$ will be contracting.

The remaining work then consists in finding an appropriate $\boldsymbol{A}$ and bounding the linear part of the resulting $\boldsymbol{T}$. Such techniques are widely advocated in, for example [108], but quite often, technical tools to design such a Newton operator are treated by hand for precise examples. On the contrary, the proposed method is fully algorithmic over the general case of LODEs and implemented into a C library.

Consider again the integral reformulation (4.4) of the problem and choose a truncation index n. The idea is to seek for an approximate inverse $\boldsymbol{A}$ of $\boldsymbol{I} + \boldsymbol{K}$ as an approximation of the finite-dimensional truncated operator $\boldsymbol{I} + \boldsymbol{K}^{[n]}$. In [20, 21] the possibility to approximate this inverse with an almost-banded matrix itself is discussed.

This $\boldsymbol{A}$ fully determines the Newton-like affine operator $\boldsymbol{T} : \varphi \mapsto \varphi - \boldsymbol{A} \cdot (\varphi + \boldsymbol{K} \cdot \varphi - \psi)$. Its linear part $\boldsymbol{I} - \boldsymbol{A} \cdot (\boldsymbol{I} + \boldsymbol{K})$ may be bounded using the following decomposition of its operator norm:

$$\|\boldsymbol{I} - \boldsymbol{A} \cdot (\boldsymbol{I} + \boldsymbol{K})\|_{\mathrm{Q}^1} \leqslant \|\boldsymbol{I} - \boldsymbol{A} \cdot (\boldsymbol{I} + \boldsymbol{K}^{[n]})\|_{\mathrm{Q}^1} + \|\boldsymbol{A} \cdot (\boldsymbol{K} - \boldsymbol{K}^{[n]})\|_{\mathrm{Q}^1}. \tag{4.8}$$

- The first term is the approximation error, since $\boldsymbol{A}$ is only an approximation of $(\boldsymbol{I} + \boldsymbol{K}^{[n]})^{-1}$. It boils down to the computation of an $n + 1$ order square matrix using multiplications and additions, which is carried out using interval arithmetics to avoid rounding errors.

- The second part is the truncation error, due to the fact that $\boldsymbol{K}^{[n]}$ is only a finite-dimensional approximation of $\boldsymbol{K}$.

The difficulty lies in this second error term, obtained by uniformly bounding $\|\boldsymbol{A} \cdot (\boldsymbol{K} - \boldsymbol{K}^{[n]}) \cdot T_i\|_{\mathrm{Q}^1}$ with respect to i. A method for choosing a sufficiently small value of n such that the Newton-like operator $\boldsymbol{T}$ be contracting, as well as its overall complexity of is given in [20, 21]. While the worst-case bound of n is exponential with respect to the magnitude of the Chebyshev coefficients of a_i, in practice and for a wide range of examples, this method is quite efficient and fully automated.

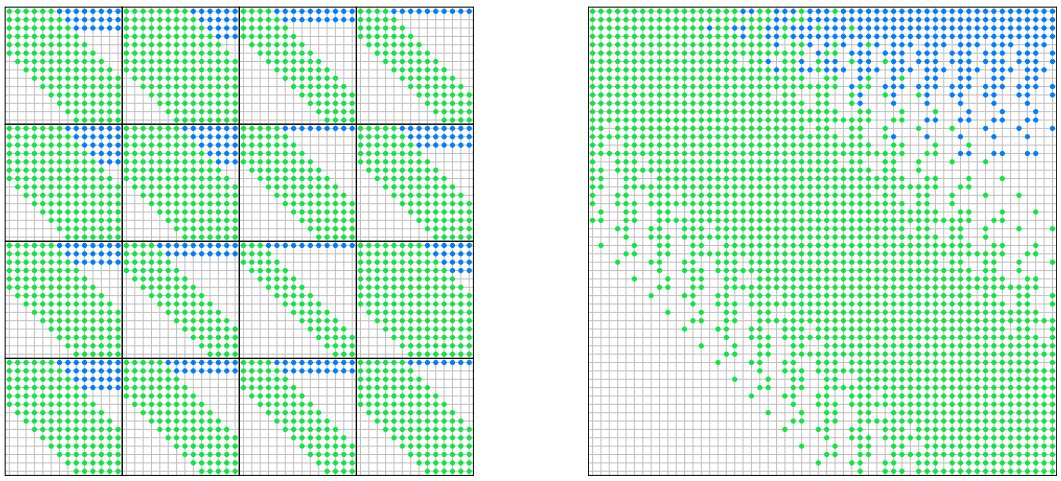
The sum of these two error terms provides the required Lipschitz constant μ. Next, take

φ an approximate truncated Chebyshev series for the exact solution φ^*. Obtaining $\boldsymbol{T} \cdot \varphi$ and then $\|\varphi - \boldsymbol{T} \cdot \varphi\|_{\mathrm{Y}^1}$ only requires arithmetic operations for polynomials in Chebyshev basis, plus a multiplication of a vector of coefficients by the matrix $\boldsymbol{A}$. The computations are rigorously performed using interval arithmetic. Applying Theorem 4.3.1, we obtain an upper bound for the approximation error $\varphi - \varphi^*$, for the Y^1-norm, and hence for the uniform norm. Integrating r times φ provides approximations of $x^{*(j)}$ $(0 \leqslant j \leqslant r)$, where x^* is the exact solution of (4.2) with initial conditions (4.3), together with rigorous error bounds with respect to the uniform norm.

Finally, to solve Problem 4.2.1 in the one-dimensional case, the above approximation and validation method is applied r times, for the canonical set of initial conditions. Since the initial conditions appear in the integral equation (4.4) only in the right-hand side ψ, operators $\boldsymbol{K}$ and hence $\boldsymbol{T}$ need to be computed and bounded only once for the r validation processes.

4.3.2 Extensions of the method

This method can be extended to the vectorial case, where $X, G : [-1, 1] \to \mathbb{R}^d$ and $A_i : [-1, 1] \to \mathbb{R}^{d \times d}$. To see this, first notice that the integral transform can be applied as in the scalar case described above. The resulting operator $\boldsymbol{K} : (\mathrm{Y}^1)^d \to (\mathrm{Y}^1)^d$ is made of $d \times d$ (scalar) integral operators $\boldsymbol{K}_{ij} : \mathrm{Y}^1 \to \mathrm{Y}^1$, which are similar to the one presented in Fig. 4.1. By rearranging the basis of $(\mathrm{Y}^1)^d$ from $T_0e_1,\ T_1e_1,\ T_2e_1,\ \ldots,\ T_0e_d,\ T_1e_d,\ T_2e_d,\ \ldots$ to $T_0e_1,\ \ldots,\ T_0e_d,\ T_1e_1,\ \ldots,\ T_1e_d,\ \ldots$, where $(e_1, \ldots, e_d)$ designates the canonical basis of $\mathbb{R}^d$, we end up again with an almost-banded structure depicted in Figure 4.2. Hence, the numerical solving essentially works as in the scalar case.



(a) Block matrix representation of vectorial $\boldsymbol{K}$

(b) Almost-banded structure of vectorial $\boldsymbol{K}$ in the rearranged basis

Figure 4.2 – Two representations of vectorial integral operator $\boldsymbol{K}$

In many problems, including the spacecraft dynamics studied in this dissertation, the coefficients and right hand side in LODE (4.2) are not polynomials but rational functions, special functions, etc. If they belong to $\mathrm{Ч}^1$ and are given through truncated Chebyshev series with a certified error bound with respect to the $\mathrm{Ч}^1$-norm, then the exact integral operator $\boldsymbol{K}$ is non-polynomial but well approximated by the polynomial integral operator $\boldsymbol{K}_P$ obtained by replacing the exact coefficients by their polynomial approximations. An additional term $\|\boldsymbol{A}\cdot(\boldsymbol{K}-\boldsymbol{K}_P)\|_{\mathrm{Ч}^1}$ appends to the two others in (4.8), but the essential ideas of the method remain unchanged.

4.4 Example: the simplified linearized Tschauner-Hempel equations

Hereafter the usage of the approximation method is illustrated by employing it on the in-plane motion described in the simplified linearized Tschauner-Hempel equations:

$$\begin{aligned} \tilde{x}'' &= 2\tilde{z}', \\ \tilde{z}'' &= \frac{3}{1+e\,\cos\nu}\tilde{z} - 2\tilde{x}' \end{aligned} \tag{SLTH}$$

First of all, one can remark that the in-plane motion can be reformulated as follows:

$$\begin{aligned} &\tilde{x}'' = 2\tilde{z}', \\ &\tilde{z}'' = \frac{3}{1+e\,\cos(\nu)}\tilde{z} - 2\tilde{x}' \end{aligned} \quad\Rightarrow\quad \begin{aligned} &\tilde{x}' = 2\tilde{z} + \frac{c}{2}, \\ &\tilde{z}'' + \left(4 - \frac{3}{1+e\,\cos\nu}\right)\tilde{z} = c, \end{aligned} \tag{4.9}$$

where c is an arbitrary integration constant. In the sequence, only the validation of a solution for the $\tilde{z}$ coordinate is studied. Before running the method, the interval $[\nu_0, \nu_f]$ to $[-1,1]$ is rescaled by introducing the independent variable $\tau \in [-1,1]$ and letting $\nu(\tau) = \nu_0(1-\tau)/2 + \nu_f(1+\tau)/2 = \omega\tau + \theta$ with $\omega = (\nu_f - \nu_0)/2$, $\theta = (\nu_0 + \nu_f)/2$, and $Z(\tau) = \tilde{z}(\nu(\tau))$. The following equation is then obtained for the $\tilde{z}$ coordinate:

$$Z''(\tau) + \omega^2\left(4 - \frac{3}{1+e\cos\nu(\tau)}\right)Z(\tau) = \omega^2 c, \tag{4.10}$$

together with rescaled initial conditions:

$$Z(-1) = z(\nu_0), \qquad Z'(-1) = \omega z'(\nu_0).$$

In particular, one can observe that the magnitude of the coefficients in (4.10) grows quadrati-

cally with the length of the interval $[\nu_0, \nu_f]$.

4.4.1 Rigorous approximation of the non-polynomial coefficient

Since the coefficient of (4.10) is not polynomial, the first step is to provide a rigorous polynomial approximation for it. The cosine function $\tau \mapsto \cos\nu(\tau)$ is approximated by applying the validation method to the harmonic oscillator differential equation:

$$\xi''(\tau) + \omega^2\xi(\tau) = 0, \qquad \xi(-1) = \cos\nu_0, \quad \xi'(-1) = -\omega\sin\nu_0. \tag{4.11}$$

From this, a rigorous approximation of $\tau \mapsto 1 + e\cos\nu(\tau)$ is deduced.

Finally, it must be composed with the reciprocal function. Numerical approximations can be obtained using interpolation at Chebyshev nodes, which is a very standard and rather efficient method [18]. Validation is performed through another Newton-like fixed-point method as follows. Let f be a function in $\mathrm{Ч}^1$, non-zero over $[-1, 1]$, to be inverted, and $g = 1/f$ the solution function. The functional equation $fg - 1 = 0$ of unknown $g \in \mathrm{Ч}^1$ must be solved (the fact that $g = 1/f$ belongs to $\mathrm{Ч}^1$ comes from Wiener's Tauberian theorem). If g_0 is a polynomial approximation of g satisfying $\|1 - g_0 f\|_{\mathrm{Ч}^1} = \mu < 1$, then g is the unique fixed-point of the affine operator $\boldsymbol{T}$ defined by $\boldsymbol{T}\cdot g = g - g_0(fg - 1)$ and of Lipschitz constant $\mu < 1$. Hence, Theorem 4.3.1 applies and provides an error enclosure for any candidate approximation $\tilde{g}$ of g. Figure 4.3 shows the evolution of the minimal degree p needed to approximate the coefficient $\tau \mapsto \omega^2(4 - 3/(1 + e\cos\nu(\tau)))$ within a $\mathrm{Ч}^1$-error less than 1, in function of the eccentricity e and the total time interval $[\nu_0, \nu_f]$.

4.4.2 Integral transform and numerical solution

Following the integral transform technique described above, consider $\varphi(\tau) = Z''(\tau)$, so that $Z(\tau)$ now becomes:

$$Z(\tau) = Z(-1) + (\tau+1)Z'(-1) + \int_{-1}^{\tau}\int_{-1}^{s}\varphi(u)du = Z(-1) + (\tau+1)Z'(-1) + \int_{-1}^{\tau}(\tau - s)\varphi(s)ds.$$

Thus the following integral equation is obtained:

$$\varphi(\tau) + \int_{-1}^{\tau}\alpha(\tau)(\tau - s)\varphi(s)ds = \omega^2 c - \alpha(\tau)(Z(-1) + (\tau+1)Z'(-1)),$$

where $\alpha(\tau) = 4 - 3/(1 + e\cos\nu(\tau))$. For the numerical solving, replace $\alpha(\tau)$ by a polynomial approximation $a(\tau)$ and proceed as in Section 4.3: truncate the resulting infinite-dimensional

equations at a chosen index n and solve the resulting almost-banded system using the algorithm presented in [83] to obtain a degree n polynomial approximation $\tilde{\varphi}$ of the solution.

4.4.3 Validation

The validation method presented in Section 4.3.1 is fully automated. Hence, in this practical example, it suffices to provide to the implemented procedure the differential equation (4.10) where $\alpha(\tau)$ is given as a polynomial approximation $a(\tau)$ together with the error bound ε, and the candidate polynomial approximate solution $\tilde{\varphi}$ obtained above. The procedure will return a rigorous upper bound of the approximation error, with respect to the Ч^1-norm.

An important remark is that the timings strongly depend on the minimal value for the truncation index that the method finds and which ensures that the obtained operator is contracting. Figure 4.3 gives these values in function of the time interval $\nu_f - \nu_0$ and the eccentricity e of the target reference orbit. We remark that these values only depend on the equation (that is, ν_0, ν_f and e) and not on the degree of the candidate approximate solution $\tilde{\varphi}$, since the contracting operator $\boldsymbol{T}$ is completely independent of this approximation. In Figure 4.3 the evolution of the approximation degree p for the coefficient $\omega^2(4 - 3/(1 + e\cos\nu(\tau)))$ and the truncation order n is a function of the eccentricity e and the total time.

4.5 Validated guidance algorithm

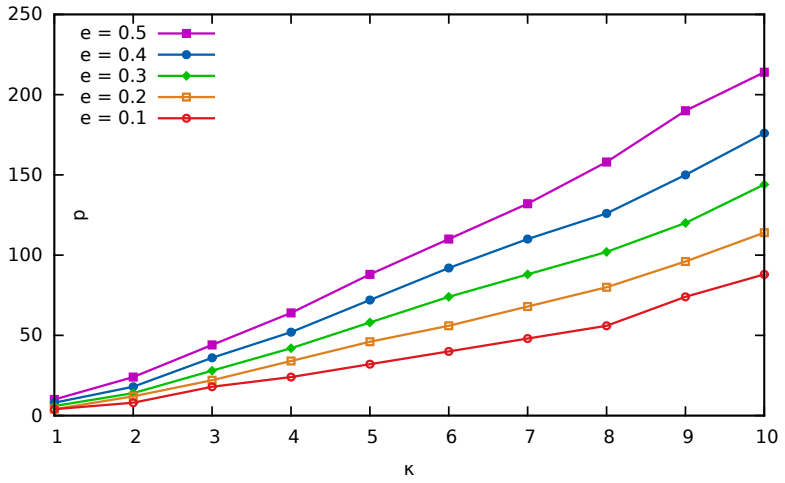
Hereafter, the validated guidance problem for the hovering phases of the orbital spacecraft rendezvous missions is studied. The idea is to provide a validation of the trajectories generated by the optimization algorithm that steers the follower satellite in a fuel-optimal way to the hovering region.

4.5.1 Validated constrained relative dynamics

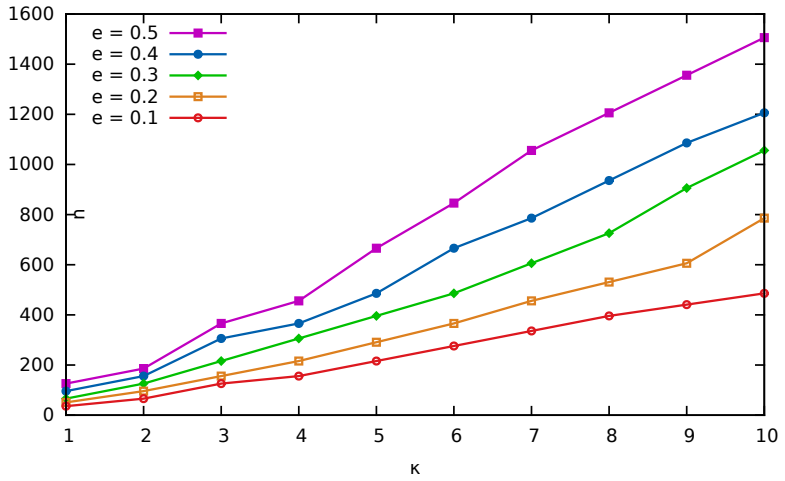
Let $\bar{\Phi}(\nu_f, \nu_0)$ be the real transition matrix of the simplified linearized Tschauner-Hempel equations (SLTH) from an initial ν_0 to a final ν_f. By considering $N \in \mathbb{N}$ impulsive velocities corrections applied at given fixed $\nu_1, \ldots, \nu_N$, the propagation of the state for $\nu \geqslant \nu_N$ is formulated as:

$$\bar{X}^+(\nu_N) = \bar{\Phi}(\nu_N, \nu_1)\bar{X}(\nu_1) + \textstyle\sum_{k=1}^{N} \bar{\Phi}(\nu_N, \nu_k)\bar{B}(\nu_k)\Delta V_k, \quad \forall \nu \geqslant \nu_N. \tag{4.12}$$

By applying the Chebyshev series approximation method previously presented, one can obtain rigorous polynomial approximations $\widetilde{\Phi}(\nu, \nu_0)$ over an arbitrary interval $[\nu_0, \nu_f]$ satisfy-



(a) Approximation degree p needed to approximate coefficient $\tau \mapsto \omega^2(4 - 3/(1 + e\cos\nu(\tau)))$ with a $\mathcal{H}^1$-error less than 1



(b) Truncation order n needed to obtain a contracting Newton-like operator for LODE (4.10)

Figure 4.3 – Parameters evolution during validation of LODE (4.10) in function of eccentricity e and total time $[\nu_0, \nu_f] = [0, 2\kappa\pi]$

ing:

$$|\widetilde{\Phi}_{ij}(\nu_0,\nu) - \bar{\Phi}_{ij}(\nu_0,\nu)| \leqslant \varepsilon_{ij}(\nu_0,\nu_f), \quad \forall \nu \in [\nu_0,\nu_f]. \tag{4.13}$$

Then, the approximated state propagation is given by:

$$\widetilde{X}(\nu) = \widetilde{\Phi}(\nu,\nu_1)\bar{X}(\nu_1) + \sum_{k=1}^{N} \widetilde{\Phi}(\nu_N,\nu_k)\bar{B}(\nu_k)\Delta V_k, \quad \nu \in [\nu_N, \nu_N + \Delta\nu], \tag{4.14}$$

where $\Delta\nu$ depends on the interval over which the approximation $\widetilde{\Phi}(\nu,\nu_1)$ is valid.

During the rendezvous hovering phases, the follower spacecraft is required to steer and remain in the interior of a certain limited region of the space. The idea is to compute a sequence of N velocity corrections generating a relative trajectory that remains inside the hovering region during the interval $[t_N,\ t_{N+1}]$.

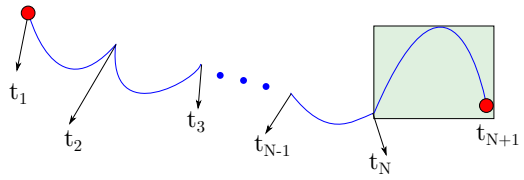


Figure 4.4 – Steering into the hovering region within N velocity corrections

Assuming, as in previous chapters, that this hovering range is a rectangular cuboid:

$$\underline{x} \leqslant x(t) \leqslant \overline{x} \quad \underline{y} \leqslant y(t) \leqslant \overline{y} \quad \underline{z} \leqslant z(t) \leqslant \overline{z}, \quad \forall t \in [t_N, t_{N+1}]. \tag{4.15}$$

Changing the simplified coordinates back to the Cartesian coordinates (1.18):

$$\begin{bmatrix} x(t), & y(t), & z(t) \end{bmatrix} = \underbrace{\frac{1}{1 + e\cos\nu}}_{P(\nu)} \begin{bmatrix} \bar{x}(\nu), & \bar{y}(\nu), & \bar{z}(\nu) \end{bmatrix}. \tag{4.16}$$

Let $\widetilde{P}(\nu) > 0$ be a positive RPA of $P(\nu)$ on the interval $[\nu_N, \nu_{N+1}]$ such that $|\widetilde{P}(\nu) - P(\nu)| < \varepsilon_P,\ \forall \nu \in [\nu_N, \nu_{N+1}]$ and the polynomials $\widetilde{P}_x(\nu)$, $\widetilde{P}_y(\nu)$, $\widetilde{P}_z(\nu)$ defined as:

$$\begin{bmatrix} \widetilde{P}_x(\nu) & \widetilde{P}_y(\nu) & \widetilde{P}_z(\nu) \end{bmatrix}^T = \begin{bmatrix} \mathbb{I}_3 & \mathbb{O}_3 \end{bmatrix} \widetilde{X}(\nu). \tag{4.17}$$

Then, a certified rational polynomial approximation for the LVLH relative positions is given by:

$$\widetilde{x}(t) = \widetilde{P}_x(\nu)/\widetilde{P}(\nu), \quad \widetilde{y}(t) = \widetilde{P}_y(\nu)/\widetilde{P}(\nu), \quad \widetilde{z}(t) = \widetilde{P}_x(\nu)/\widetilde{P}(\nu)$$

and the inequalities in (4.15) can be approximated by:

$$\begin{aligned}
&\widetilde{P}_x(\nu) - \widetilde{P}(\nu)\underline{x} \geqslant 0, \qquad \widetilde{P}(\nu)\overline{x} - \widetilde{P}_x(\nu) \geqslant 0,\\
&\widetilde{P}_y(\nu) - \widetilde{P}(\nu)\underline{y} \geqslant 0, \qquad \widetilde{P}(\nu)\overline{y} - \widetilde{P}_y(\nu) \geqslant 0, \quad \forall \nu \in [\nu_N, \nu_{N+1}].\\
&\widetilde{P}_z(\nu) - \widetilde{P}(\nu)\underline{z} \geqslant 0, \qquad \widetilde{P}(\nu)\overline{z} - \widetilde{P}_z(\nu) \geqslant 0,
\end{aligned} \tag{4.18}$$

4.5.2 Fuel-optimal impulsive validated guidance problem

Using the developments presented in the previous section for obtaining a validated description of the constrained relative trajectories, the fuel-optimal impulsive validated guidance problem for the rendezvous hovering phases is formulated as follows:

Problem 4.5.1. *(Validated guidance problem) Consider the scenario: eccentricity e, semi-major axis a, initial state $X(\nu_1) \in \mathbb{R}^6$, number of control actions $N \in \mathbb{N}$ and final true anomaly ν_{N+1}. Find N impulses $\Delta V \in \mathbb{R}^{3N}$ applied at given fixed instants $\nu_1, \ldots, \nu_N$, such that:*

$$\begin{aligned}
&\underset{\Delta V}{\operatorname{argmin}} \quad \mathcal{J}(\Delta V)\\
&s.t. \quad \begin{cases}
\widetilde{P}_x(\nu) - \widetilde{P}(\nu)\underline{x} \geqslant 0, \qquad \widetilde{P}(\nu)\overline{x} - \widetilde{P}_x(\nu) \geqslant 0,\\
\widetilde{P}_y(\nu) - \widetilde{P}(\nu)\underline{y} \geqslant 0, \qquad \widetilde{P}(\nu)\overline{y} - \widetilde{P}_y(\nu) \geqslant 0, \quad \forall \nu \in [\nu_N, \nu_{N+1}].\\
\widetilde{P}_z(\nu) - \widetilde{P}(\nu)\underline{z} \geqslant 0, \qquad \widetilde{P}(\nu)\overline{z} - \widetilde{P}_z(\nu) \geqslant 0,
\end{cases}
\end{aligned} \tag{$\mathcal{P}.\mathcal{SIP}$}$$

This problem is a semi-infinite program (SIP), since the space constraints must be satisfied for infinitely many values of ν. However, the polynomial inequalities in $\mathcal{P}.\mathcal{SIP}$ can be converted into so-called linear matrix inequalities (LMIs) using the results on the parametrization of non-negative polynomials on the cone of semi-definite positive matrices presented by [81, Theorems 9 and 10], resulting in the semi-definite program (SDP) described in $\mathcal{P}.\mathcal{SDP}$ (see Appendix B).

Problem 4.5.2. *Consider the scenario: eccentricity e, semi-major axis a, initial state $X(\nu_1) \in \mathbb{R}^6$, number of control actions $N \in \mathbb{N}$ and final true anomaly ν_{N+1}. Find N impulses $\Delta V \in \mathbb{R}^{3N}$ applied at given fixed instants $\nu_1, \ldots, \nu_N$, such that:*

$$\begin{aligned}
&\underset{\Delta V}{\operatorname{argmin}} \quad \mathcal{J}(\Delta V)\\
&s.t. \quad \begin{cases}
\widetilde{p}_{\underline{x}} = \Lambda^*(Y_{1\underline{x}}, Y_{2\underline{x}}), \qquad \widetilde{p}_{\overline{x}} = \Lambda^*(Y_{1\overline{x}}, Y_{2\overline{x}}),\\
\widetilde{p}_{\underline{y}} = \Lambda^*(Y_{1\underline{y}}, Y_{2\underline{y}}), \qquad \widetilde{p}_{\overline{y}} = \Lambda^*(Y_{1\overline{y}}, Y_{2\overline{y}}),\\
\widetilde{p}_{\underline{z}} = \Lambda^*(Y_{1\underline{z}}, Y_{2\underline{z}}), \qquad \widetilde{p}_{\overline{z}} = \Lambda^*(Y_{1\overline{z}}, Y_{2\overline{z}}),
\end{cases}
\end{aligned} \tag{$\mathcal{P}.\mathcal{SDP}$}$$

where $\widetilde{p}_w$ is the vector containing the coefficients of the respective non-negative polynomials

in the SIP formulation and Λ^* is a bilinear operator (more details in [28, 29, 81]).

The advantages of reformulating $\mathcal{P}.\mathcal{SIP}$ are twofold:

- In the SIP formulation, the space constraints are described by infinitely many constraints on the true anomaly, requiring discretization techniques to efficiently compute a "solution". This solution, however, will systematically violate the constraints of the original problem. On the other hand, the SDP formulation provides a finite and exact description of the constraints;
- In previous works [24, 29], SDP solvers were employed in the conception of control strategies for the spacecraft rendezvous problems, showing good performances even in environments with limited computational resources, such as devices dedicated to space applications.

4.5.3 Results

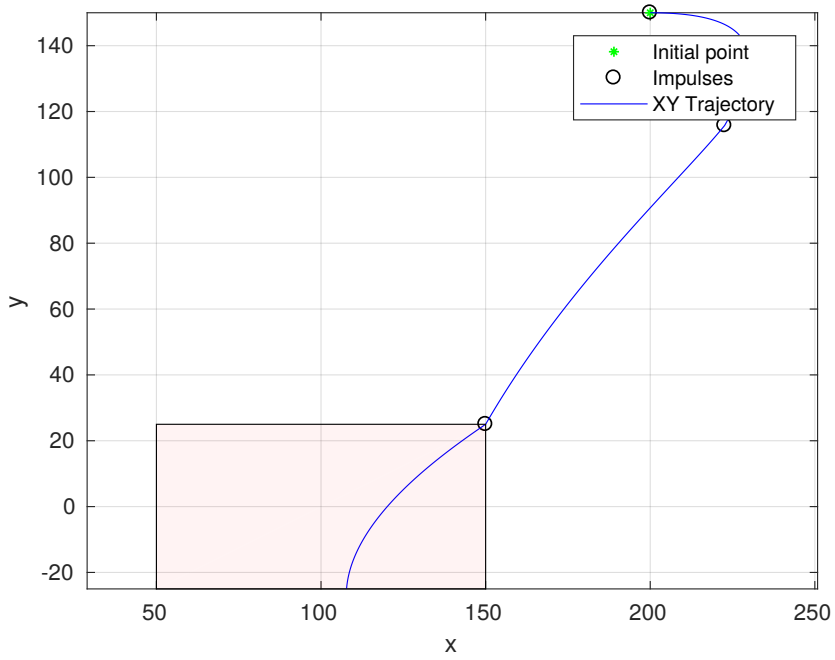
Hereafter the results obtained by solving the validated guidance algorithm for the scenario described by the parameters given in Table 4.1 are presented.

Table 4.1 – Scenario parameters

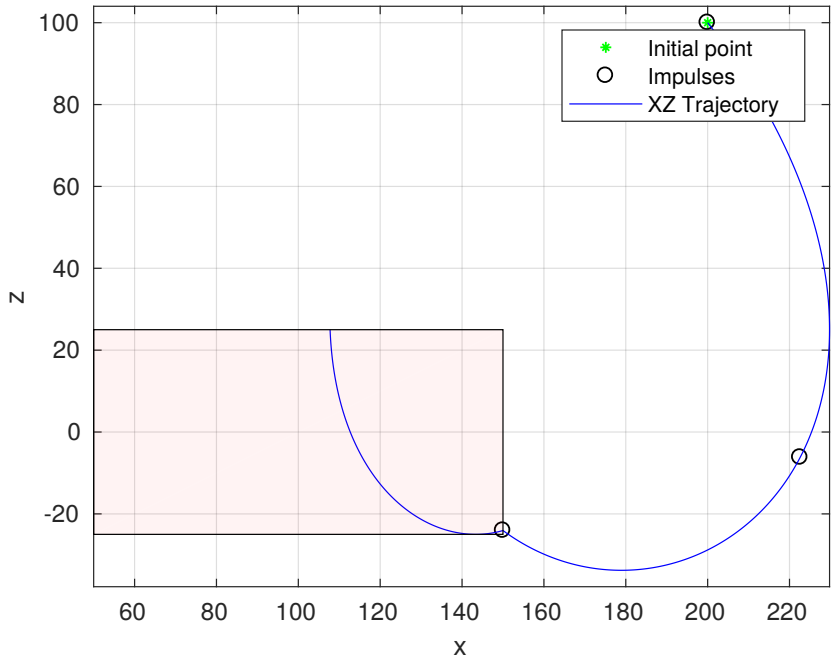
Semi-major axis:	$a = 7011$ km
Eccentricity:	$e = 0.4$
Initial true anomaly:	$\nu_1 = 0$ rad
Interval between impulses:	$\Delta\nu = \pi/4$ rad
Number of impulsive velocity corrections:	$N = 3$
Saturation:	$\overline{\Delta V} = 1$ m/s
Initial relative state [m, m/s]:	$X(\nu_1) = [200, 150, 100, 0, 0, 0]$
Hovering zone $[\underline{x}, \overline{x}, \underline{y}, \overline{y}, \underline{z}, \overline{z}]$ [m]:	$[50, 150, -25, 25, -25, 25]$
Degree of RPAs:	5, 7

The RPAs were computed by a C implementation of the Chebyshev approximation method previously presented. The respective SDP problem was formulated in Matlab® via Yalmip [67] (`https://yalmip.github.io/`) and solved using the SDPT3 solver [103] (`http://www.math.nus.edu.sg/~mattohkc/sdpt3.html`). The certificate enclosures are evaluated by treating the bounds of the RPAs via arithmetic interval [101] with the help of the b4m interval arithmetic toolbox library [110] (`http://www.ti3.tu-harburg.de/zemke/b4m/`).

In Fig. 4.5 the nominal relative trajectory obtained by simply propagating the initial state under the effect of the control actions is presented. Fig. 4.6, 4.7 and 4.8 show the evolution of the x, z and y coordinates of the relative trajectory and their respective certificates.

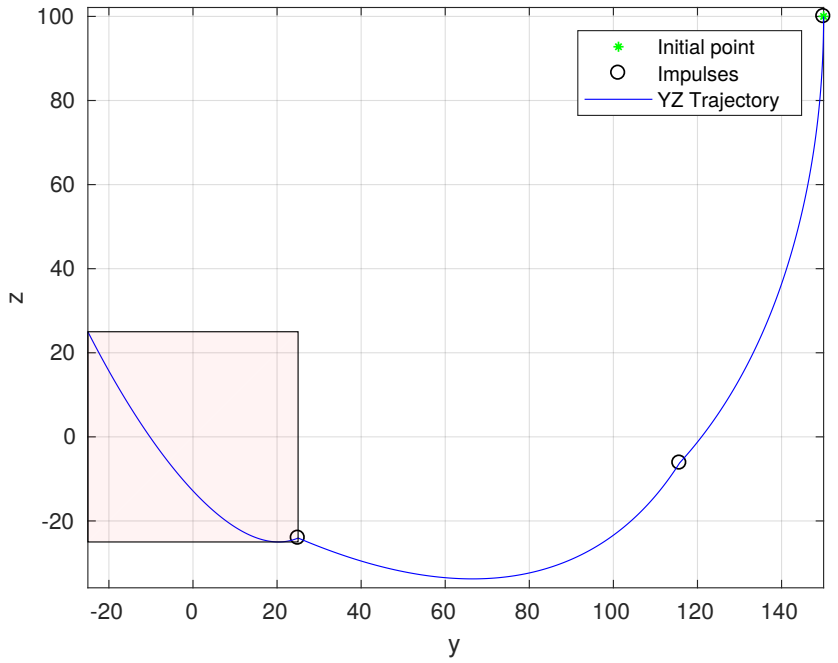


(a) XY projection of obtained trajectory.

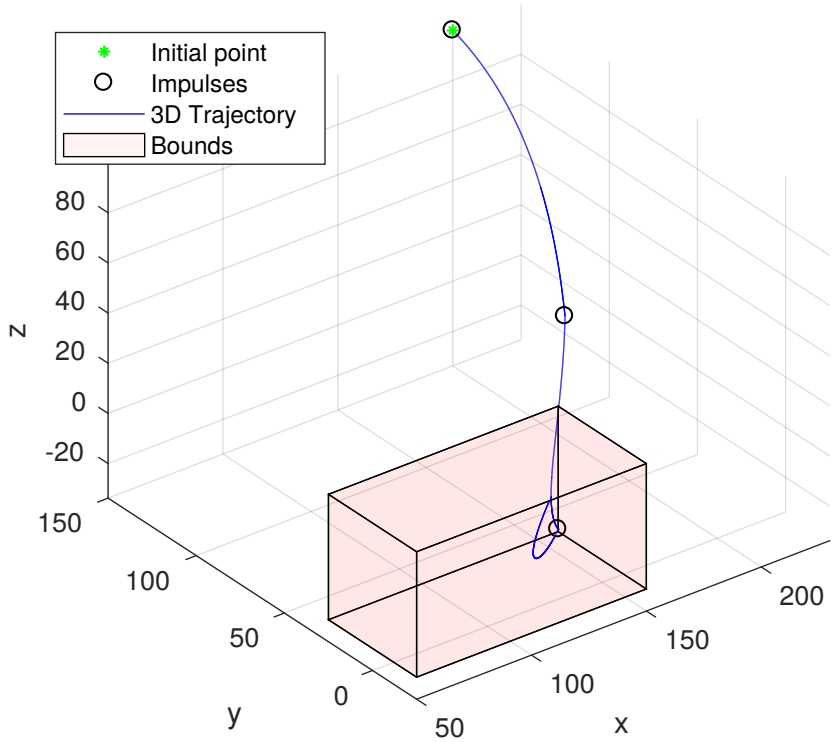


(b) XZ projection of obtained trajectory.

Figure 4.5 – Obtained relative trajectory without certification.



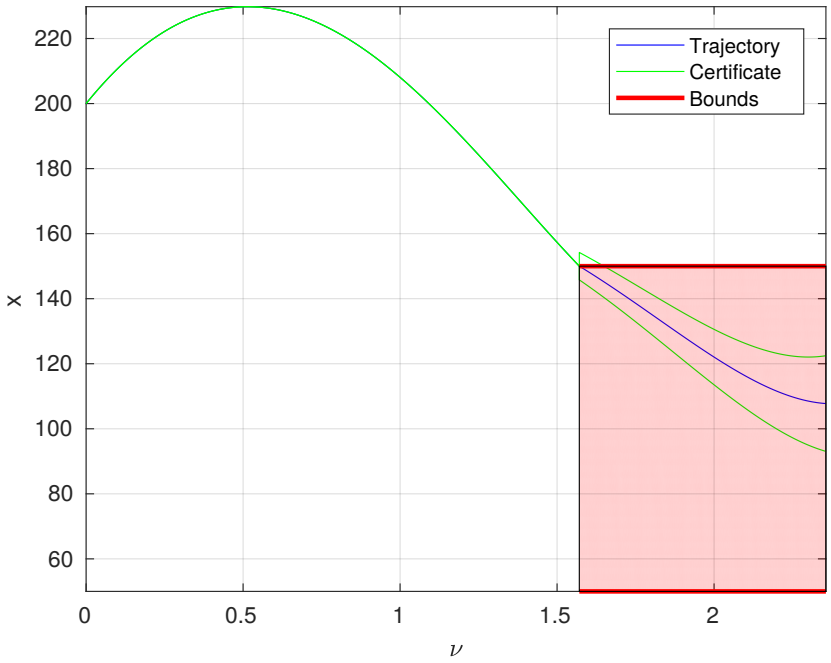
(c) YZ projection of obtained trajectory.



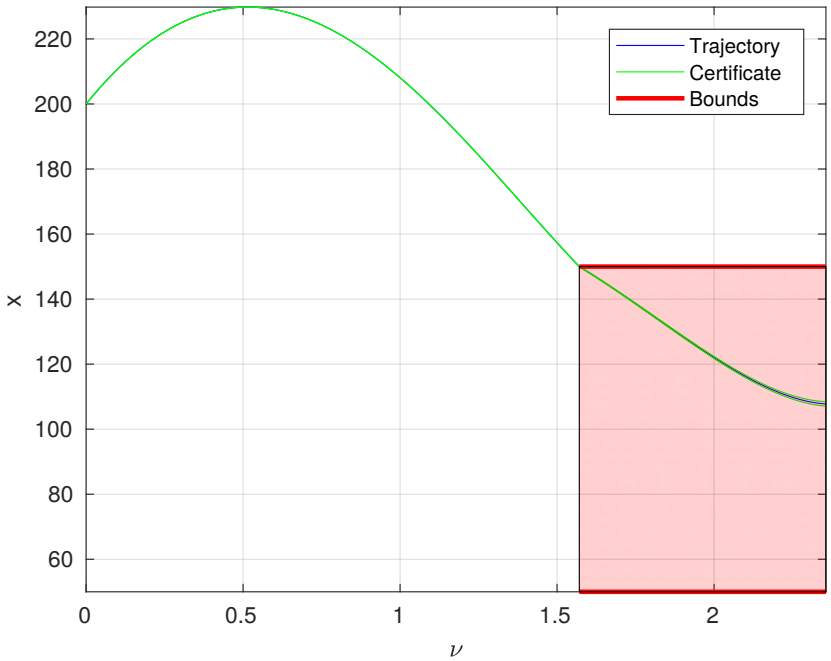
(d) 3D view of obtained trajectory.

Figure 4.5 – Obtained relative trajectory without certification.

From Fig. 4.5, one can remark that the computed impulses produce a relative trajectory that enters the hovering region and remains therein for the imposed true anomaly interval.



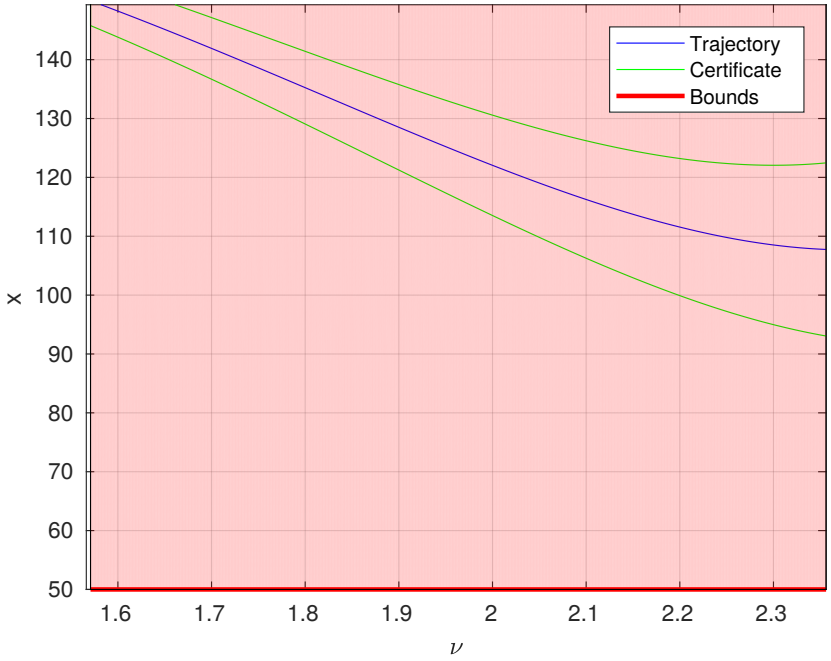
(a) X-trajectory, RPAs of degree 5.



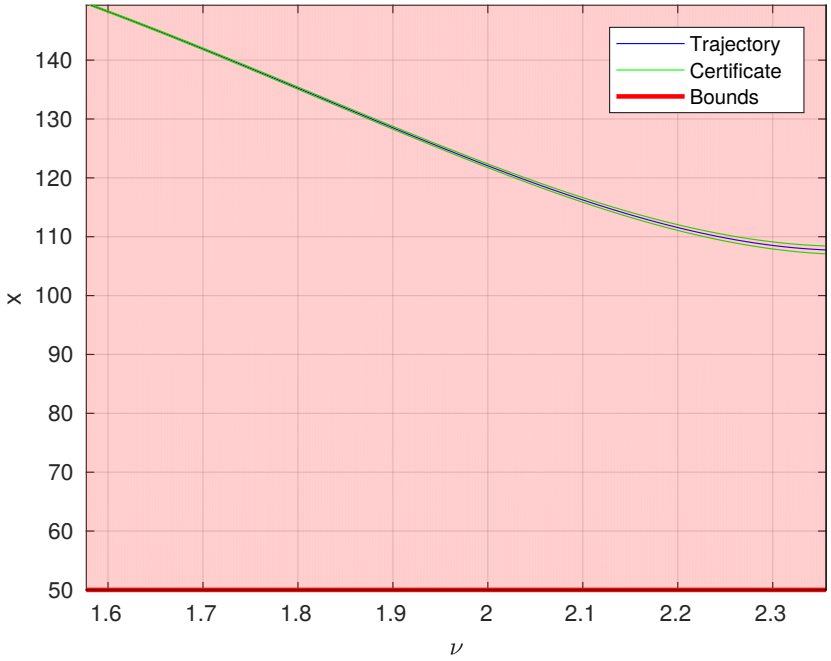
(b) X-trajectory, RPAs of degree 7.

Figure 4.6 – X-coordinate in function of true anomaly for RPAs of degrees 5 and 7.

From the figures depicting the evolution of the in-plane x and z coordinates trajectories, one can remark that the increase of the degree of the RPAs produce tighter certificate envelopes (in green) for the errors. This is well illustrated by comparing the detailed views presented in Fig. 4.7: the RPAs of degree 5 produce a certificate of approximatively 5 m for the excursion

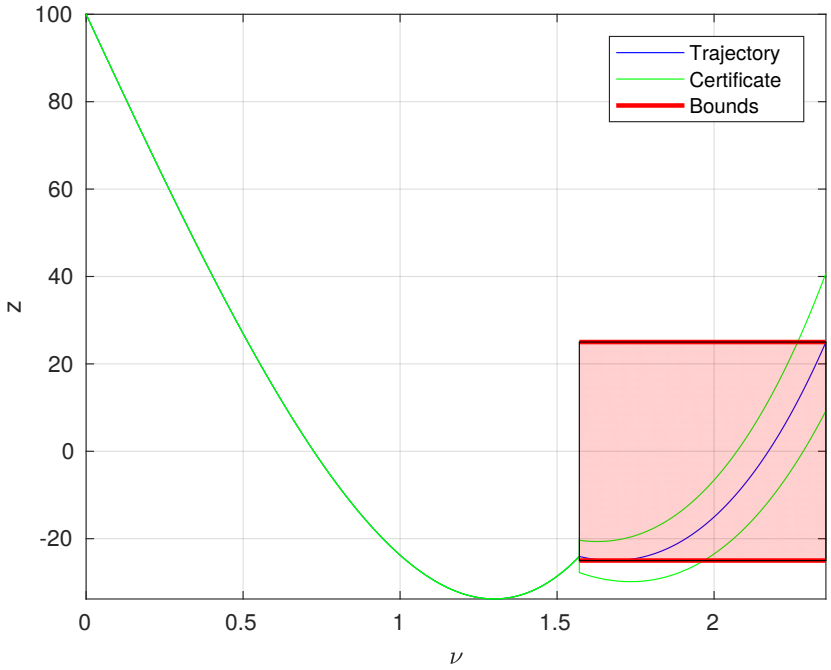


(c) Final uncertainty for x-trajectory, RPAs of degree 5.

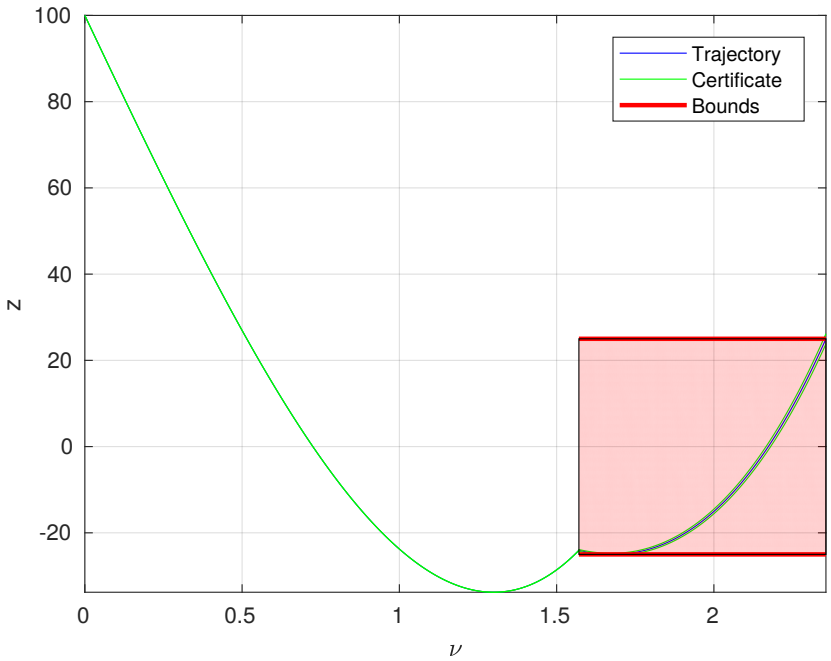


(d) Final uncertainty for x-trajectory, RPAs of degree 7.

Figure 4.6 – X-coordinate in function of true anomaly for RPAs of degrees 5 and 7.

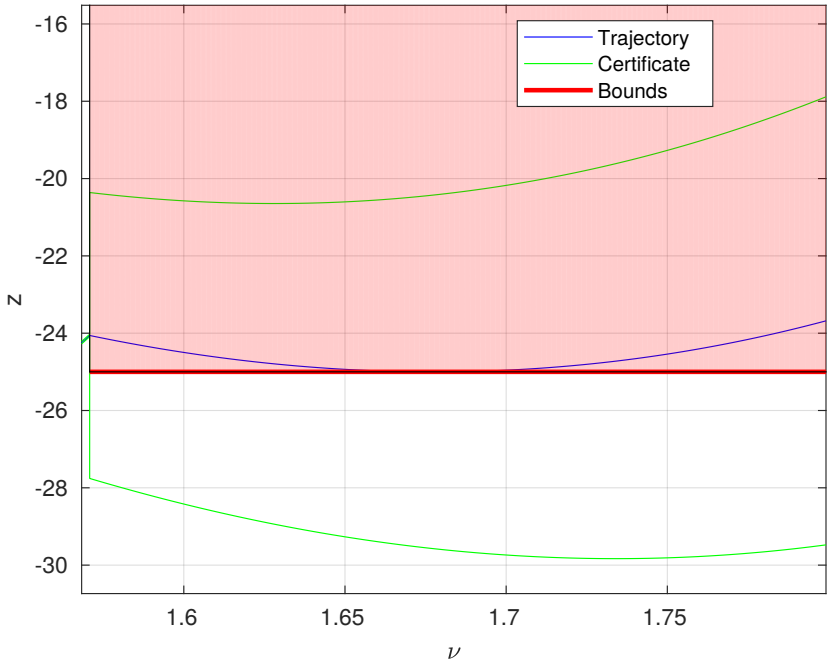


(a) Z-trajectory, RPAs of degree 5.

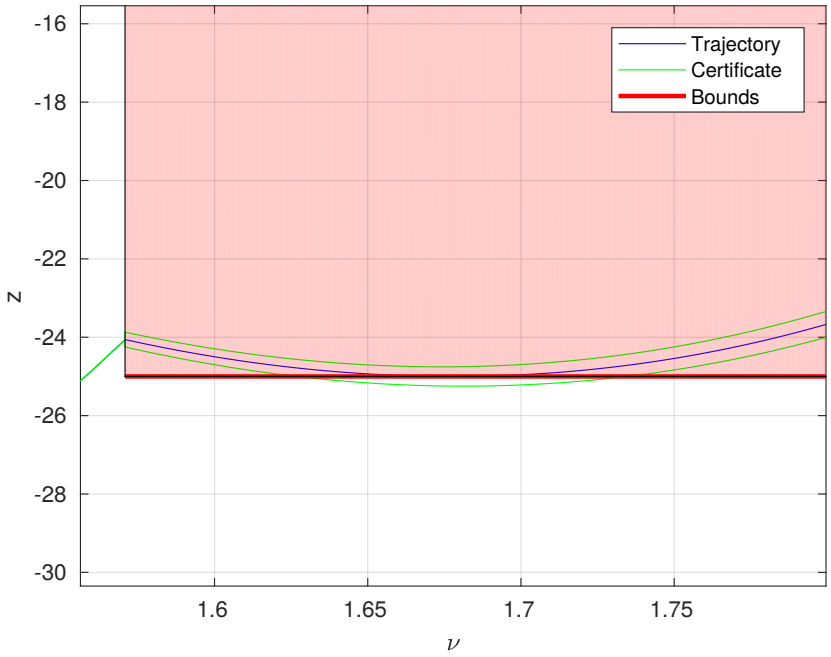


(b) Z-trajectory, RPAs of degree 7.

Figure 4.7 – Z-coordinate in function of true anomaly for RPAs of degrees 5 and 7.



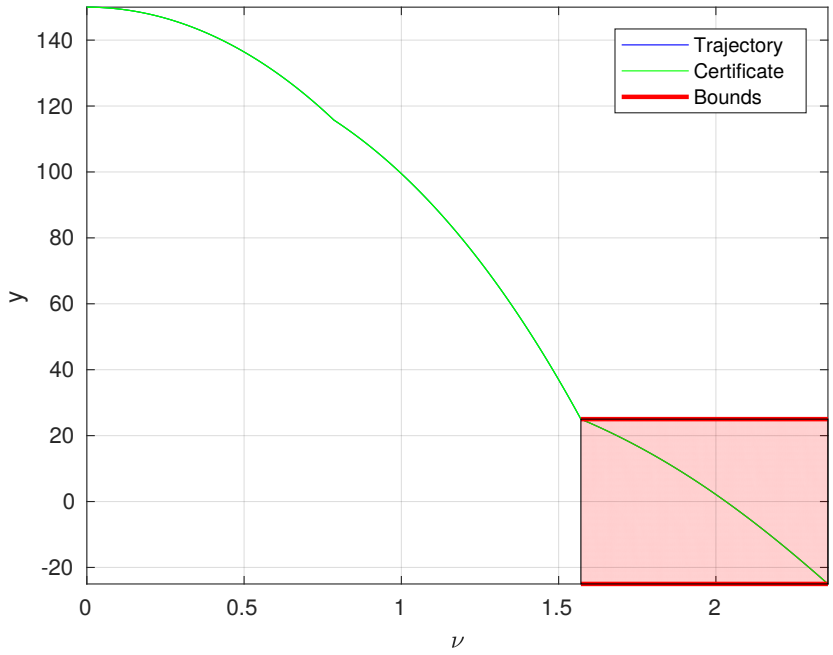
(c) Excursion of z-trajectory, RPAs of degree 5.



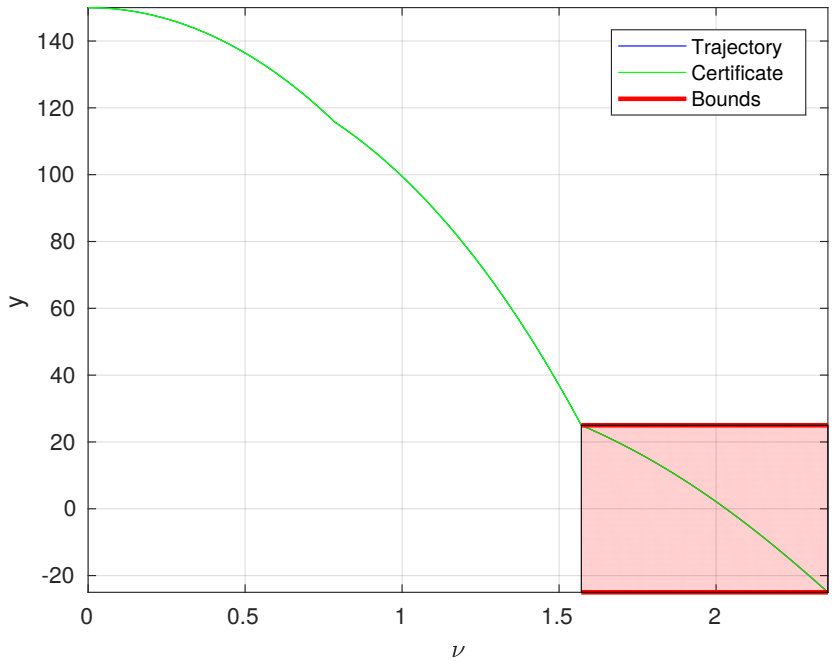
(d) Excursion of z-trajectory, RPAs of degree 7.

Figure 4.7 – Z-coordinate in function of true anomaly for RPAs of degrees 5 and 7.

of the trajectory, while the RPAs of degree 7 produce a much lower excursion certificate of approximatively 25 cm.



(a) Y-trajectory, RPAs of degree 5.



(b) Y-trajectory, RPAs of degree 7.

Figure 4.8 – Y-coordinate in function of true anomaly for RPAs of degrees 5 and 7.

From Fig. 4.8, it is possible to observe that the certificates for y are tighter than those obtained for the in-plane coordinates x and y (the upper and lower certificates cannot even be distinguished without zooming). This is explained by the fact that the out-of-plane dynamics is described by a simpler harmonic oscillator as previously presented in (4.11), which can be

efficiently approximated by low degree polynomials. For instance, both RPAs produce final certificate enclosure of less than 2 cm for the relative trajectory.

4.6 Conclusions

An approximation method for obtaining rigorous polynomial approximations of the solutions of LODE via Chebyshev series has been described in this chapter. An extension of this technique for the case of multidimensional LODEs with non-polynomial coefficients was employed to obtain a validated guidance algorithm based on SDP for the rendezvous hovering phases. The use of the obtained RPAs and interval arithmetic allows the validation of the propagation of the relative trajectories, providing an automated *a posteriori* certification method to certify the computed control actions. Future experiments would assess the tractability of problem $\mathcal{P}.\mathcal{SDP}$ on devices dedicated to space applications, focusing on the analysis of the relation between the computational burden and the precision of the polynomial approximations.

Conclusion and future works

The main goal of this work is the development of new validated optimization-based guidance and control algorithms for the spacecraft rendezvous hovering phases. These algorithms are both sufficiently comprehensive to account for all the complex constraints of the problem, and efficient to be numerically solved in reasonable time on devices dedicated to space applications (LEON3 microprocessors).

A model predictive control strategy is proposed, ensuring the convergence and station-keeping of the relative motion inside the hovering region, even when the saturation of the thrusters is taken into account. This control strategy is based on the iterative resolution of an optimal fixed-time guidance problem that employs a simplified linearized synthesis model (described in Chap. 1) to perform the propagation of the spacecraft relative trajectories. In order to obtain efficient algorithms, we focused on several reformulations of this optimal guidance problem. One of them (see Chap. 2) is an original contribution of this work, and consists in a new exact and finite description of the set of admissible periodic relative trajectories that only relies on closed-form expressions depending on the space constraints and on the entries of the state vector. During numerical tests, this new finite approach showed computation times of the order of magnitude of those obtained for traditional discretization methods, with the advantage of not systematically violating the space constraints.

In future works, a further generalization of this optimal guidance problem could consider that the control application dates are not *a priori* fixed and periodical. In this sense, we proposed a very recent and preliminary result in [69]. Ideally, in order to reduce even more the conservativeness, the number of control actions and their application dates could be set as decision variables of the problem. This could, however, generate problems that are too cumbersome to be solved in low-performance devices, and a trade-off remains to be found.

To assess the performance of our algorithms, a particular effort was devoted to the construction of a processor-in-the-loop simulator, described in Chap. 3, which replicates the behavior of control algorithms embedded on a real spacecraft. Its operation consists in a loop-communication between the nonlinear simulator that computes the evolution of the spacecraft relative trajectories and the device that employs the LEON3 microprocessor to compute the control actions. Simulations performed in this processor-in-the-loop framework reveal that although the proposed model predictive approach may produce greater convergence times, it is more efficient with respect to fuel consumption than other methods proposed in the literature. Moreover, this approach is numerically tractable in reasonable time when embedded in space

dedicated devices. The proven convergence (see Chap. 3) of the proposed model predictive control strategy intuitively suggests a certain level of robustness, which is observed in practice during the simulations. An extension of our work will consist in a deeper theoretical study of the properties of the closed-loop system, which should provide an idea of how robust the system is to uncertainties.

Another important aspect treated in this thesis is mission safety. For this purpose, two (ground support) validation strategies for our algorithms are presented: first, a "classical" simulator (described in Chap. 1) relying on the numerical integration of the Gauss planetary equations under disturbances provides a high fidelity reproduction of the (nonlinear, perturbed) relative motion. This is employed to validate via numerical simulations the relative trajectories of the linearized synthesis model.

However, since these simulations contain numerical errors, we also focused on a validation technique which takes both approximation and rounding errors into account. We debuted our study in Chap. 4, with the validation of the relative trajectories generated by the control actions computed at each call of the optimal guidance problem, and provided an automated numerical certification method to inspect the satisfaction of the constraints of the problem. This is based on rigorous polynomial approximations (error-bounded polynomials that approximate functions on a given interval) of the linearized synthesis model and interval arithmetic. In future works, we intend to develop validation techniques for the solutions of more general differential equations, such as the disturbed nonlinear Tschauner-Hempel equations, which would provide also a certification tool for the general simulation model.

Finally, synchronous programming languages would help to certify the proposed controller codes from the industrial point of view. Then, these controller codes could be embedded on small or end-of-life satellites for the accomplishment of short missions, which would allow the evaluation of the behavior and robustness of the proposed techniques in a real space environment.

Appendix A

Tschauner-Hempel equations

A.1 Nonlinear Tschauner-Hempel equations

Hereafter we deduce the equations describing the relative dynamics between two spacecraft orbiting the Earth in elliptical orbits (see Fig. A.1).

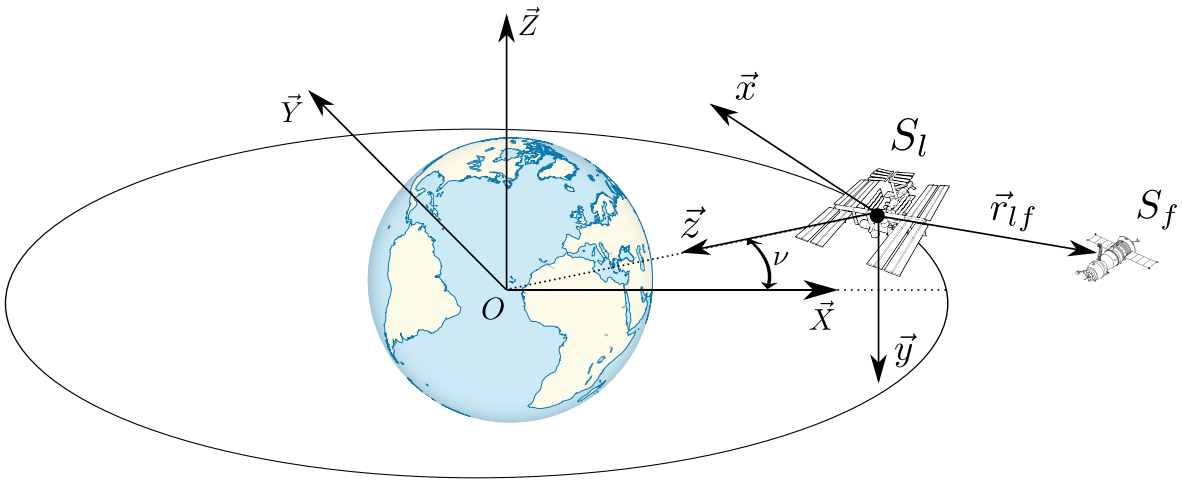


Figure A.1 – Rendezvous illustration.

Let be the vectors $\vec{r}_{lf} = \overrightarrow{S_l S_f}$, $\vec{r}_{el} = \overrightarrow{OS_l}$ and $\vec{r}_{ef} = \overrightarrow{OS_f}$. Supposing that the gravitational attraction between the spacecraft is negligible and applying Newton's second law of motion to both of them, we obtain:

$$\left.\frac{d^2\vec{r}_{el}}{dt^2}\right|_{\mathcal{F}_O} = -\frac{\mu}{\|\vec{r}_{el}\|^3}\vec{r}_{el}, \qquad \left.\frac{d^2\vec{r}_{ef}}{dt^2}\right|_{\mathcal{F}_O} = -\frac{\mu}{\|\vec{r}_{ef}\|^3}\vec{r}_{ef}.$$

By subtracting the second equation from the first one, for the left-hand side we have:

$$\left.\frac{d^2\vec{r}_{ef}}{dt^2}\right|_{\mathcal{F}_O} - \left.\frac{d^2\vec{r}_{el}}{dt^2}\right|_{\mathcal{F}_O} = \left.\frac{d^2\left(\vec{r}_{ef} - \vec{r}_{el}\right)}{dt^2}\right|_{\mathcal{F}_O} = \left.\frac{d^2\vec{r}_{lf}}{dt^2}\right|_{\mathcal{F}_O}.$$

Combining with the right-hand side, we obtain an expression for the dynamics of the

relative motion in the ECI frame:

$$\left.\frac{d^2\vec{r}_{lf}}{dt^2}\right|_{\mathcal{F}_O} = -\frac{\mu}{\|\vec{r}_{ef}\|^3}\vec{r}_{ef} + \frac{\mu}{\|\vec{r}_{el}\|^3}\vec{r}_{el}. \tag{A.1.1}$$

The relation between accelerations in the frames $\mathcal{F}_O$ and $\mathcal{F}_L$ is given by:

$$\left.\frac{d^2\vec{r}_{lf}}{dt^2}\right|_{\mathcal{F}_O} = \underbrace{\left.\frac{d^2\vec{r}_{lf}}{dt^2}\right|_{\mathcal{F}_L}}_{\text{Apparent acceleration}} + \underbrace{2\vec{\Omega}_{\mathcal{F}_L/\mathcal{F}_O} \times \left.\frac{d\vec{r}_{lf}}{dt}\right|_{\mathcal{F}_L}}_{\text{Coriolis acceleration}} + \underbrace{\left.\frac{d\vec{\Omega}_{\mathcal{F}_L/\mathcal{F}_O}}{dt}\right|_{\mathcal{F}_L} \times \vec{r}_{lf}}_{\text{Euler acceleration}} + \underbrace{\vec{\Omega}_{\mathcal{F}_L/\mathcal{F}_O} \times \left(\vec{\Omega}_{\mathcal{F}_L/\mathcal{F}_O} \times \vec{r}_{lf}\right)}_{\text{centrifugal acceleration}}.$$

Using the expressions of $\vec{r}_{lf}$ and $\vec{\Omega}_{\mathcal{F}_L/\mathcal{F}_O}$ in the LVLH frame:

$$\vec{r}_{lf}|_{\mathcal{F}_L} = \begin{bmatrix} x, & y, & z \end{bmatrix}^T, \tag{A.1.2}$$

$$\left.\vec{\Omega}_{\mathcal{F}_L/\mathcal{F}_O}\right|_{\mathcal{F}_L} = \begin{bmatrix} 0, & 0, & -\dot{\nu} \end{bmatrix}^T,$$

each component of the acceleration can be computed as follows:

- The apparent acceleration:

$$\left.\frac{d^2\vec{r}_{lf}}{dt^2}\right|_{\mathcal{F}_L} = \begin{bmatrix} \ddot{x} \\ \ddot{y} \\ \ddot{z} \end{bmatrix}.$$

- The Coriolis acceleration:

$$2\vec{\Omega}_{\mathcal{F}_L/\mathcal{F}_O} \times \left.\frac{d\vec{r}_{lf}}{dt}\right|_{\mathcal{F}_L} = 2\begin{bmatrix} 0 \\ 0 \\ -\dot{\nu} \end{bmatrix} \times \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \end{bmatrix} = \begin{bmatrix} -2\dot{\nu}\dot{z} \\ 0 \\ 2\dot{\nu}\dot{x} \end{bmatrix}.$$

- The Euler acceleration:

$$\left.\frac{d\vec{\Omega}}{dt}\right|_{\mathcal{F}_L} \times \vec{r}_{lf} = \begin{bmatrix} 0 \\ 0 \\ -\ddot{\nu} \end{bmatrix} \times \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} -\ddot{\nu}z \\ 0 \\ \ddot{\nu}x \end{bmatrix}.$$

- The centrifugal acceleration:

$$\vec{\Omega}_{\mathcal{F}_L/\mathcal{F}_O} \times \left(\vec{\Omega}_{\mathcal{F}_L/\mathcal{F}_O} \times \vec{r}_{lf}\right) = \begin{bmatrix} 0 \\ 0 \\ -\nu \end{bmatrix} \times \left(\begin{bmatrix} 0 \\ 0 \\ -\nu \end{bmatrix} \times \begin{bmatrix} x \\ y \\ z \end{bmatrix}\right) = \begin{bmatrix} 0 \\ 0 \\ -\nu \end{bmatrix} \times \begin{bmatrix} -\nu z \\ 0 \\ \nu x \end{bmatrix} = \begin{bmatrix} -\nu^2 x \\ 0 \\ -\nu^2 z \end{bmatrix}.$$

These accelerations add up to:

$$\left.\frac{d^2\vec{r}_{lf}}{dt^2}\right|_{\mathcal{F}_O} = \begin{bmatrix} \ddot{x} - 2\dot{\nu}\dot{z} - \ddot{\nu}z - \nu^2 x \\ \ddot{y} \\ \ddot{z} + 2\dot{\nu}\dot{z} + \ddot{\nu}x - \nu^2 z \end{bmatrix}_{\mathcal{F}_L}. \tag{A.1.3}$$

The expression of $\vec{r}_{el}$ in the LVLH frame is:

$$\vec{r}_{el}|_{\mathcal{F}_L} = \begin{bmatrix} 0, & 0, & -R \end{bmatrix}^T, \tag{A.1.4}$$

where $R(t) = \|\vec{R}(t)\| = \dfrac{a(1-e^2)}{1 + e\cos\nu(t)}$.

By replacing (A.1.2), (A.1.3) and (A.1.4) in (A.1.1), we obtain:

$$\begin{aligned} \ddot{x} &= 2\dot{\nu}\dot{z} + \ddot{\nu}z + \dot{\nu}^2 x - \frac{\mu x}{\sqrt{(x^2+y^2+(R-z)^2)^3}} \\ \ddot{y} &= -\frac{\mu y}{\sqrt{(x^2+y^2+(R-z)^2)^3}} \\ \ddot{z} &= -2\dot{\nu}\dot{x} - \ddot{\nu}x + \dot{\nu}^2 z - \frac{\mu(z-R)}{\sqrt{(x^2+y^2+(R-z)^2)^3}} - \frac{\mu}{R^2}, \end{aligned} \tag{NLTH}$$

the so-called nonlinear Tschauner-Hempel equations.

A.2 Linearized Tschauner-Hempel equations

Assuming that the distance between spacecraft is much smaller that the distance from the leader spacecraft to the center of the Earth:

$$\sqrt{x^2+y^2+z^2} \ll R, \tag{A.2.1}$$

we perform first order approximations of the nonlinear terms $f_x(x,y,z) = -\frac{\mu x}{\sqrt{(x^2+y^2+(R-z)^2)^3}}$, $f_y(x,y,z) = -\frac{\mu y}{\sqrt{(x^2+y^2+(R-z)^2)^3}}$ and $f_z(x,y,z) = -\frac{\mu(z-R)}{\sqrt{(x^2+y^2+(R-z)^2)^3}}$ around the point $(x,y,z) = (0,0,0)$ via truncated Taylor series expansions:

$$f(x,y,z) \approx f(0,0,0) + x\left.\frac{\partial f}{\partial x}\right|_{(0,0,0)} + y\left.\frac{\partial f}{\partial y}\right|_{(0,0,0)} + z\left.\frac{\partial f}{\partial z}\right|_{(0,0,0)}$$

For $f_x(x,y,z)$ we have:

$$f_x(0,0,0) = 0,$$

$$\left.\frac{\partial f_x}{\partial x}\right|_{(0,0,0)} = \left.-\frac{\mu(R^2-2Rz-2x^2+y^2+z^2)}{\sqrt{(x^2+y^2+(R-z)^2)^5}}\right|_{(0,0,0)} = -\frac{\mu}{R^3},$$

$$\left.\frac{\partial f_x}{\partial y}\right|_{(0,0,0)} = \left.\frac{3\mu xy}{\sqrt{(x^2+y^2+(R-z)^2)^5}}\right|_{(0,0,0)} = 0,$$

$$\left.\frac{\partial f_x}{\partial z}\right|_{(0,0,0)} = \left.-\frac{3\mu x(R-z)}{\sqrt{(x^2+y^2+(R-z)^2)^5}}\right|_{(0,0,0)} = 0,$$

which adds up to:

$$f_x(x,y,z) \approx f_x(0,0,0) + x\left.\frac{\partial f_x}{\partial x}\right|_{(0,0,0)} + y\left.\frac{\partial f_x}{\partial y}\right|_{(0,0,0)} + z\left.\frac{\partial f_x}{\partial z}\right|_{(0,0,0)} = -\frac{\mu}{R^3}x. \tag{A.2.2}$$

For $f_y(x,y,z)$ we have:

$$f_y(0,0,0) = 0,$$

$$\left.\frac{\partial f_y}{\partial x}\right|_{(0,0,0)} = \left.\frac{3\mu xy}{\sqrt{(x^2+y^2+(R-z)^2)^5}}\right|_{(0,0,0)} = 0,$$

$$\left.\frac{\partial f_y}{\partial y}\right|_{(0,0,0)} = \left.-\frac{\mu(R^2-2Rz+x^2-2y^2+z^2)}{\sqrt{(x^2+y^2+(R-z)^2)^5}}\right|_{(0,0,0)} = -\frac{\mu}{R^3},$$

$$\left.\frac{\partial f_y}{\partial z}\right|_{(0,0,0)} = \left.-\frac{3\mu y(R-z)}{\sqrt{(x^2+y^2+(R-z)^2)^5}}\right|_{(0,0,0)} = 0,$$

which adds up to:

$$f_y(x,y,z) \approx f_y(0,0,0) + x\left.\frac{\partial f_y}{\partial x}\right|_{(0,0,0)} + y\left.\frac{\partial f_y}{\partial y}\right|_{(0,0,0)} + z\left.\frac{\partial f_y}{\partial z}\right|_{(0,0,0)} = -\frac{\mu}{R^3}y \tag{A.2.3}$$

For $f_z(x,y,z)$ we have:

$$f_z(0,0,0) = \frac{\mu}{R^2},$$

$$\left.\frac{\partial f_z}{\partial x}\right|_{(0,0,0)} = \left.\frac{3\mu x(z-R)}{\sqrt{(x^2+y^2+(R-z)^2)^5}}\right|_{(0,0,0)} = 0,$$

$$\left.\frac{\partial f_z}{\partial y}\right|_{(0,0,0)} = \left.\frac{3\mu y(z-R)}{\sqrt{(x^2+y^2+(R-z)^2)^5}}\right|_{(0,0,0)} = 0,$$

$$\left.\frac{\partial f_z}{\partial z}\right|_{(0,0,0)} = -\left.\frac{\mu(-2R^2+4Rz+x^2+y^2-2z^2)}{\sqrt{(x^2+y^2+(R-z)^2)^5}}\right|_{(0,0,0)} = \frac{2\mu}{R^3}z,$$

which adds up to:

$$f_z(x,y,z) \approx f_z(0,0,0) + x\left.\frac{\partial f_z}{\partial x}\right|_{(0,0,0)} + y\left.\frac{\partial f_z}{\partial y}\right|_{(0,0,0)} + z\left.\frac{\partial f_z}{\partial z}\right|_{(0,0,0)} = \frac{\mu(R+2z)}{R^3} \qquad \text{(A.2.4)}$$

Replacing (A.2.2), (A.2.3) and (A.2.4) in (NLTH), the linearized Tschauner-Hempel equations are obtained:

$$\begin{aligned}
\ddot{x} &= 2\dot{\nu}\dot{z} + \ddot{\nu}z + \dot{\nu}^2 x - \frac{\mu}{R^3}x \\
\ddot{y} &= -\frac{\mu}{R^3}y \\
\ddot{z} &= -2\dot{\nu}\dot{x} - \ddot{\nu}x + \dot{\nu}^2 z + \frac{2\mu}{R^3}z.
\end{aligned} \qquad \text{(LTH)}$$

A.3 Simplified linearized Tschauner-Hempel equations

In [55], Humi performs a sequence of manipulations in order to produce a simplified version of the linearized Tschauner-Hempel equations, in which the true anomaly appears as independent variable. These manipulations are presented step-by-step hereafter:

Step 1: Replace $\dot{\nu} = k^2\rho_\nu^2$, where $k^2 = \sqrt{\frac{\mu}{a^3(1-e^2)^3}}$, and $\rho_\nu = (1 + e\,\cos\nu)$ in (LTH):

$$\begin{aligned}
\ddot{x} &= 2\,\dot{\nu}\,\dot{z} + \ddot{\nu}\,z + (\dot{\nu}^2 - k\dot{\nu}^{\frac{3}{2}})x \\
\ddot{y} &= -k\dot{\nu}^{\frac{3}{2}}y \\
\ddot{z} &= -2\,\dot{\nu}\,\dot{x} - \ddot{\nu}\,x + (\dot{\nu}^2 + 2k\dot{\nu}^{\frac{3}{2}})z,
\end{aligned}$$

Step 2: Replace time by true anomaly as independent variable using the following relations:

$$(\cdot)' = \frac{d(\cdot)}{d\nu}, \qquad (\cdot)'' = \frac{d^2(\cdot)}{d\nu^2}, \qquad \frac{d(\cdot)}{dt} = (\cdot)'\dot{\nu}, \qquad \frac{d^2(\cdot)}{dt^2} = (\cdot)''\dot{\nu}^2 + (\cdot)'\ddot{\nu},$$

obtaining the following expressions:

$$\begin{aligned}
\dot{\nu}^2 x'' + \ddot{\nu}x' &= 2\dot{\nu}^2 z' + \ddot{\nu}z + (\dot{\nu}^2 - k\dot{\nu}^{\frac{3}{2}})x \\
\dot{\nu}^2 y'' + \ddot{\nu}y' &= -k\dot{\nu}^{\frac{3}{2}}y \\
\dot{\nu}^2 z'' + \ddot{\nu}z' &= -2\dot{\nu}^2 x' - \ddot{\nu}x + (\dot{\nu}^2 + 2k\dot{\nu}^{\frac{3}{2}})z,
\end{aligned}$$

where $s_\nu = \sin\nu$ and $c_\nu = \cos\nu$.

Step 3: Substitute $\dot{\nu}$ by $k^2\rho_\nu^2$:

$$\begin{aligned} k^4\rho_\nu^4 x'' - 2k^4\rho_\nu^3 e s_\nu x' &= 2k^4\rho_\nu^4 z' - 2k^4\rho_\nu^3 e s_\nu z + k^4\rho_\nu^3(\rho_\nu - 1)x \\ k^4\rho_\nu^4 y'' - 2k^4\rho_\nu^3 e s_\nu y' &= -k^4\rho_\nu^3 y \\ k^4\rho_\nu^4 z'' - 2k^4\rho_\nu^3 e s_\nu z' &= -2k^4\rho_\nu^4 x' + 2k^4\rho_\nu^3 e s_\nu x + k^4\rho_\nu^3(\rho_\nu + 2)z. \end{aligned}$$

Step 4: Divide both sides by $k^3\rho_\nu^3$:

$$\begin{aligned} \rho_\nu x'' - 2e s_\nu x' &= 2\rho_\nu z' - 2e s_\nu z + e c_\nu x \\ \rho_\nu y'' - 2e s_\nu y' &= -y \\ \rho_\nu z'' - 2e s_\nu z' &= -2\rho_\nu x' + 2e s_\nu x + (3 + e c_\nu)z. \end{aligned}$$

Step 5: Subtract ec_ν multiplied by the respective coordinate from each equation:

$$\begin{aligned} \rho_\nu x'' - 2e s_\nu x' - e c_\nu x &= 2(\rho_\nu z' - e s_\nu z) \\ \rho_\nu y'' - 2e s_\nu y' - e c_\nu y &= -\rho_\nu y \\ \rho_\nu z'' - 2e s_\nu z' - e c_\nu z &= -2(\rho_\nu x' - e s_\nu x) + 3z. \end{aligned} \tag{A.3.1}$$

Step 6: Let be $X(t) = [x,\ y,\ z,\ \dot{x},\ \dot{y},\ \dot{z}]^T$ and $\tilde{X}(\nu) = [\tilde{x},\ \tilde{y},\ \tilde{z},\ \tilde{x}',\ \tilde{y}',\ \tilde{z}']^T$. such that:

$$\tilde{X}(\nu) = T(\nu)X(t), \qquad X(t) = T^{-1}(\nu)\tilde{X}(\nu), \tag{A.3.2}$$

where $T(\nu)$ is given by:

$$T(\nu) = \begin{bmatrix} \rho_\nu I_3 & 0_3 \\ -e s_\nu I_3 & (k^2\rho_\nu)^{-1} I_3 \end{bmatrix}.$$

This variable change induces the following relations:

$$\begin{aligned} \tilde{x} &= \rho_\nu x & \tilde{x}' &= \rho_\nu x' - e s_\nu\, x & \tilde{x}'' &= \rho_\nu x'' - 2e s_\nu\, x' - e c_\nu\, x \\ \tilde{y} &= \rho_\nu y\ , & \tilde{y}' &= \rho_\nu y' - e s_\nu\, y\ , & \tilde{y}'' &= \rho_\nu y'' - 2e s_\nu\, y' - e c_\nu\, y \\ \tilde{z} &= \rho_\nu z & \tilde{z}' &= \rho_\nu z' - e s_\nu\, z & \tilde{z}'' &= \rho_\nu z'' - 2e s_\nu\, z' - e c_\nu\, z, \end{aligned}$$

By introducing these new tilde-variables in (A.3.1), we finally obtain the so-called simplified linearized Tschauner Hempel equations::

$$\begin{aligned} \tilde{x}'' &= 2\tilde{z}' \\ \tilde{y}'' &= -y \\ \tilde{z}'' &= \frac{3}{\rho_\nu}\tilde{z} - 2\tilde{x}'. \end{aligned} \tag{SLTH}$$

Appendix B

Univariate non-negative polynomials

In [81, Theorems 9 and 10], Nesterov presents the conditions that an univariate polynomial must satisfy in order to be non-negativity on finite, semi-infinite or infinite intervals of the reals. These conditions are obtained by exploring the existent relations between the cone of of non-negative polynomials and the cone of positive semi-definite matrices.

In the sequel, we present the linear matrix inequalities that the coefficients of a non-negative univariate polynomial over the reals must satisfy. The reader is invited to consult [81, Sections 3.1-3.3] and [28, Appendix B] for the other cases and further details.

Proposition B.1 (Non-negative univariate polynomials over the reals)**.**
Let be an even $n \in \mathbb{N}$*, the vector* $p = \begin{bmatrix} p_0 & \dots & p_n \end{bmatrix}^T \in \mathbb{R}^{n+1}$ *and the polynomial* $P(w) = \sum_{i=0}^{n} p_i w^i,\ \forall w \in \mathbb{R}$*. Then, the condition for the non-negativity of* $P(w)$ *is:*

$$P(w) \geqslant 0,\ \forall w \in \mathbb{R} \ \Leftrightarrow\ \exists\, Y \succeq 0 \ s.t.\ p = \Lambda^*(Y),$$

where $Y \in \mathbb{R}^{(n/2+1)\times(n/2+1)}$ *is a symmetric matrix and the operator* $\Lambda^*(\cdot)$ *is given by* $\Lambda^*(Y) = \begin{bmatrix} \mathrm{tr}(Y\, H_{n/2+1,1}) & \dots & \mathrm{tr}(Y\, H_{n/2+1,n+1}) \end{bmatrix}^T$*, with*

$$\mathbb{R}^{(m+1)\times(m+1)} \ni H_{m,k} = \begin{cases} H_{m,k}(i,j) = 1, & \text{if } \ i+j = k+1 \\ H_{m,k}(i,j) = 0, & \text{otherwise.} \end{cases}$$

Example B.1:

Consider the polynomial

$$Q(w) = 232 + 196w + 77w^2 + 14w^3 + w^4$$

and the vector of its coefficients $q = \begin{bmatrix} 232 & 196 & 77 & 14 & 1 \end{bmatrix}^T$. Let us verify if there exists $\mathbb{R}^{3\times 3} \ni Y \succeq 0$ that satisfies $q = \Lambda^*(Y)$:

$$
\begin{aligned}
232 &= \operatorname{tr}\left(\begin{bmatrix} Y_{11} & Y_{12} & Y_{13} \\ Y_{12} & Y_{22} & Y_{23} \\ Y_{13} & Y_{23} & Y_{33} \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}\right) = Y_{11} && \Rightarrow Y_{11} = 232 \\
196 &= \operatorname{tr}\left(\begin{bmatrix} Y_{11} & Y_{12} & Y_{13} \\ Y_{12} & Y_{22} & Y_{23} \\ Y_{13} & Y_{23} & Y_{33} \end{bmatrix} \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}\right) = 2Y_{12} && \Rightarrow Y_{12} = 98 \\
77 &= \operatorname{tr}\left(\begin{bmatrix} Y_{11} & Y_{12} & Y_{13} \\ Y_{12} & Y_{22} & Y_{23} \\ Y_{13} & Y_{23} & Y_{33} \end{bmatrix} \begin{bmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \end{bmatrix}\right) = 2Y_{13} + Y_{22} && \Rightarrow 2Y_{13} + Y_{22} = 77 \\
14 &= \operatorname{tr}\left(\begin{bmatrix} Y_{11} & Y_{12} & Y_{13} \\ Y_{12} & Y_{22} & Y_{23} \\ Y_{13} & Y_{23} & Y_{33} \end{bmatrix} \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix}\right) = 2Y_{23} && \Rightarrow Y_{23} = 7 \\
1 &= \operatorname{tr}\left(\begin{bmatrix} Y_{11} & Y_{12} & Y_{13} \\ Y_{12} & Y_{22} & Y_{23} \\ Y_{13} & Y_{23} & Y_{33} \end{bmatrix} \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}\right) = Y_{33} && \Rightarrow Y_{33} = 1
\end{aligned}
$$

Let us choose $Y_{13} = 10$ and $Y_{22} = 57$ and test the positivity of the leading principal minors of Y:

$$
\begin{aligned}
D_1(Y) &= 232 > 0, \\
D_2(Y) &= \begin{vmatrix} 232 & 98 \\ 98 & 57 \end{vmatrix} = 3620 > 0, \\
D_3(Y) &= \begin{vmatrix} 232 & 98 & 10 \\ 98 & 57 & 7 \\ 10 & 7 & 1 \end{vmatrix} = 272 > 0.
\end{aligned}
$$

Since all the leading principal minors of Y are positive, we conclude that Y is positive definite and, consequently, $Q(w) \geqslant 0, \forall w \in \mathbb{R}$.

Appendix C

Envelopes and implicitization

C.1 Envelope definition

Let be $X \in \mathbb{R}^n$, $p \in \mathbb{R}$ and $F(X,p)$ an one-parameter function, differentiable with respect to p. Let $F(X,p) = 0$ be the equation that represents a family of hypersurfaces parametrized by p in the n-dimensional Euclidean space.

Definition C.1.1 (Envelope of a family of hypersurfaces)**.** *The hypersurface tangent to each of the hypersurfaces of the p-parametrized family $F(X,p) = 0$ is called the* envelope *of F and is given by the set of points $\mathcal{E}$ defined by:*

$$\mathcal{E} := \left\{ X \in \mathbb{R}^n \;\middle|\; F(X,p) = 0,\ \frac{\partial F}{\partial p}(X,p) = 0 \right\}. \tag{C.1.1}$$

Example C.1: Envelope

Consider the function

$$F(x,y,p) = \sin(p)x + \cos(p)y - 1$$

and the associated family of lines given by $F(x,y,p) = 0$ depicted in Fig. C.1.

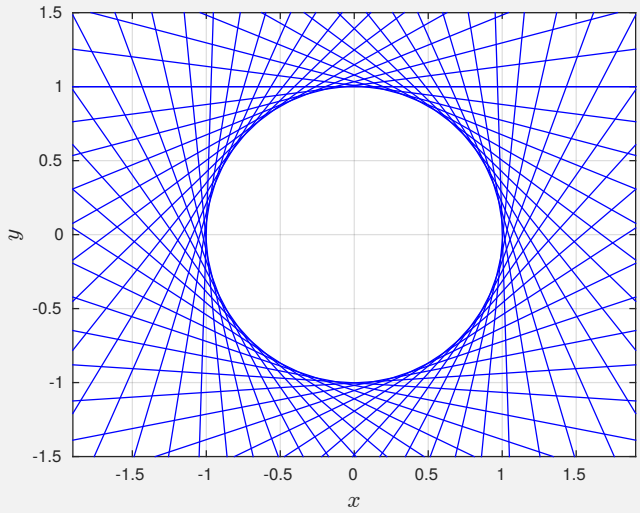


Figure C.1 – Family of lines $\sin(p)x + \cos(p)y = 1$.

We remark in Fig. C.1 that the set of points at which two neighbor lines intersect are exactly the circle of radius 1 centered at $(0, 0)$. In fact, we can prove that this circle corresponds to the envelope of $F(x, y, p)$ using Def. C.1.1. In order to do so, it suffices to compute the pairs (x, y) for which $F(x, y, p) = \frac{\partial F}{\partial p}(x, y, p) = 0$:

$$\begin{cases} \sin(p)x + \cos(p)y = 1 \\ \cos(p)x - \sin(p)y = 0 \end{cases} \overset{\text{squaring}}{\Rightarrow} \begin{cases} \sin^2(p)x^2 + 2\sin(p)\cos(p) + \cos^2(p)y^2 = 1 \\ \cos^2(p)x^2 - 2\sin(p)\cos(p) + \sin^2(p)y^2 = 0 \end{cases} \overset{\text{adding}}{\Rightarrow} x^2 + y^2 = 1 \ .$$

Although Ex. C.1 illustrates with simplicity how one can obtain the expression of the envelope of a family of curves, for more general cases these expression are not always straightforward to be computed. In the sequel we present an implicitization method that can be employed for this purpose.

C.2 Implicitization method

After computing $F(X, p) = 0$ and $\frac{\partial F}{\partial p}(X, p) = 0$, we obtain two equations parametrized by p. In some cases, it is not evident how to proceed in order to obtain an implicit expression for the envelope from these equations. However, if both equalities can be reformulated as polynomial equations with complex coefficients, we can employ Sylvester's Matrix implicitization method in order to do so.

We use the example given in [107, Section 2] to explain how this implicitization method can be employed:

Example C.2: Sylvester's matrix implicitization method

Consider the parametric expressions:

$$\begin{cases} x = 0.4 + 0.5\cos(p) - 2\sin(2p) \\ y = 0.6 + 0.2\sin(p) + 0.7\sin(2p), \end{cases}$$

producing the polar plot presented in Fig. C.2.

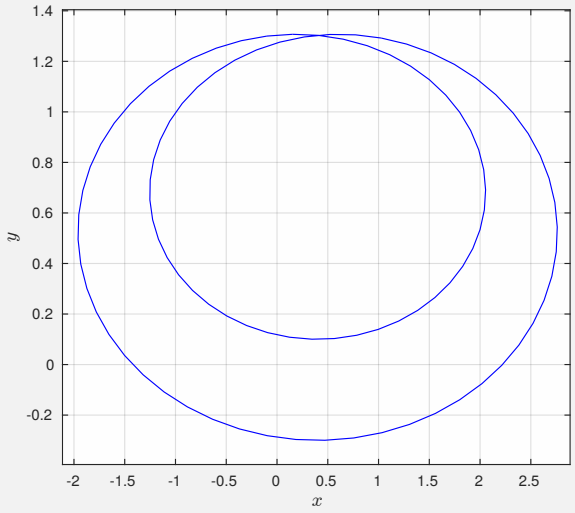


Figure C.2 – Polar plot for $x = 0.4 + 0.5\cos(p) - 2\sin(2p)$ and $y = 0.6 + 0.2\cos(p) + 0.7\cos(2p)$.

In order to obtain polynomial equations with complex coefficients, we perform the following change of variables:

$$\cos(p) = \frac{\exp(it) + \exp(-it)}{2} = \frac{z + 1/z}{2} \quad \text{and} \quad \sin(p) = \frac{\exp(it) - \exp(-it)}{2i} = \frac{z - 1/z}{2i}.$$

Then, multiplying the obtained equalities by z^2, we obtain the following polynomial equations:

$$\begin{cases} p_x(z) = iz^4 + 0.25z^3 + (0.4 - x)z^2 + 0.25z - i = 0 \\ p_y(z) = 0.35z^4 - 0.1z^3 + (0.6 - y)z^2 + 0.1z + 0.35 = 0. \end{cases}$$

The implict expression producing the plot presented in Fig. C.2 can then be obtained by evaluating the resultant of the polynomials $p_x(z)$ and $p_y(z)$, i.e., by computing the

determinant of the following Sylvester's Matrix:

$$S = \begin{bmatrix} -i & 0.25 & 0.4-x & 0.25 & i & 0 & 0 & 0 \\ 0 & -i & 0.25 & 0.4-x & 0.25 & i & 0 & 0 \\ 0 & 0 & -i & 0.25 & 0.4-x & 0.25 & i & 0 \\ 0 & 0 & 0 & -i & 0.25 & 0.4-x & 0.25 & i \\ 0.35 & 0.1i & 0.6-y & -0.1i & 0.35 & 0 & 0 & 0 \\ 0 & 0.35 & 0.1i & 0.6-y & -0.1i & 0.35 & 0 & 0 \\ 0 & 0 & 0.35 & 0.1i & 0.6-y & -0.1i & 0.35 & 0 \\ 0 & 0 & 0 & 0.35 & 0.1i & 0.6-y & -0.1i & 0.35 \end{bmatrix},$$

which produces the bivariate polynomial:

$$f(x,y) = \frac{2401}{160000}x^4 + \frac{49}{200}x^2y^2 + y^4 - \frac{2401}{100000}x^3 - \frac{18879}{64000}x^2y - \frac{49}{250}xy^2 - \frac{77}{32}y^3 - \frac{159927}{8000000}x^2 + \frac{18879}{80000}xy + \frac{3036377}{2560000}y^2 + \frac{236759}{10000000}x + \frac{4743897}{12800000}y - \frac{358805651}{6400000000}.$$

The pairs $(x,y) \in \mathbb{R}^2$ such that $f(x,y) = 0$ are exactly those generating the plot presented in Fig. C.2.

Appendix D

Embedding the libraries for the LP and SDP methods

---

1. *Cross-compiler*: the architecture of the synthesized LEON3 microprocessor requires a particular X86/SPARC cross-compiler. This cross-compiler[1] is installed on a Linux machine in the `/opt` directory using the following commands:

```
$ cd /opt
$ tar xvjf sparc−linux−ct−multilib−0.0.7.tar.bz2
$ export PATH=/opt/sparc−linux−4.4.2−toolchains/multilib/bin:$PATH
```

2. *LAPACK*[2] *and GLPK*: the procedure of cross-compilation of LAPACK for SPARC/LEON3 is the following one:

   2.1) Shell variables for LAPACK cross compilation:

```
$ export CPPFLAGS="-I/opt/sparc-linux-4.4.2-toolchains/multilib/sparc-leon-
    ↪ linux-gnu/sys-root/usr/include"
$ export CFLAGS="-g -O2"
$ export LD="/opt/sparc-linux-4.4.2-toolchains/multilib/sparc-leon-linux-gnu/
    ↪ bin/ld"
$ export LDFLAGS="-L/opt/sparc-linux-4.4.2-toolchains/multilib/lib/gcc/sparc-
    ↪ leon-linux-gnu/4.4.2"
$ export CC=sparc−linux−gcc
```

   2.2) CFLAG integrates the“-g" option for debugging reasons, then use only the “-O2" option. In the `glpk-4.60` directory is the `configure` file that generates the `makefile`:

```
$ ./configure −−host = sparc−sun−sunos4.1.1
$ make
```

[1] http://gaisler.com/anonftp/linux/linux-2.6/toolchains/sparc-linux-4.4.2/sparc-linux-ct-multilib-0.0.7.tar.bz2

[2] LAPACK official site: http://www.netlib.org/lapack/, Archives: http://www.netlib.org/lapack/lapack-3.7.0.tgz

2.3) Once the previous steps are executed, it is possible to compile a test program with the LAPACK library:

```
$ sparc-linux-g++ -Wall test.c -o test -I. -L./.libs -lglpk
```

2.4) At this point the `libglpk.a` library is created in the directory `glpk-4.60/src/.libs`.

3. *CSDP*[3]: the usual CSDP compilation chain must be substantially modified for cross-compilation. Each `Makefile` file must be modified to integrate the LAPACK library and the paths to the compilers.

   The Linux diff command can be used to highlight new instructions in makefiles. A complete CSDP archive with all the changes for a cross-compilation is available at `http://homepages.laas.fr/fcamps/CSDP/CSDP.tar`. Each `Make.diff` file makes it possible to apply a patch to the Makefile of the original archive. CSDP source code is available with the Linux command:

   ```
   svn co https://projects.coin-or.org/svn/Csdp/trunk
   ```

   Once the makefiles have been modified, a simple run of the `Make` command is needed to generate the CSDP library.

4. *Embedding the libraries*: The computation libraries must be installed on the board (in the case of a non-static compilation) in the standard directory of the libraries.

   The variable `LD_LIBRARY_PATH` must be modified, for example:

   ```
   $ export LD_LIBRARY_PATH=$LD_LIBRARY_PATH:/usr/lib
   ```

[3]CSDP official site : `https://projects.coin-or.org/Csdp/`

Appendix E

Non-smooth optimization methods

Hereafter we present methods to address the following problem:

Problem E.0.1 (Non-smooth optimization problem)**.** Find $\xi^* \in \mathbb{R}^n$ such that

$$\xi^* = \begin{cases} \text{argmin} & \phi(\xi) \\ s.t. & \xi \in C \end{cases} \tag{$\mathcal{P}_0$}$$

where $\phi : \mathbb{R}^n \mapsto \mathbb{R}$ *is a convex function and* C *is a compact set in* $\mathbb{R}^n$ *defined by equalities and/or inequalities:*

$$C = \{\xi \in \mathbb{R}^n \mid \psi_i(\xi) \leqq 0, \quad i = 1, 2, ..., m\} \tag{E.0.1}$$

where each function $\psi_i : \mathbb{R}^n \mapsto \mathbb{R}$ *is convex.*

The proposed problem may have more than one solution: in the case where the constraint set is feasible, there exists one global minimum for the problem (guaranteed by the extreme value theorem, compactness of C and convexity of the objective function) that can possibly be attained at more than one point.

E.1 Penalty method

Luenberger and Ye propose in [73] a penalty method to approximate this constrained problem by an unconstrained problem. This method consists on removing the constraints and adding a function multiplied by a constant to the objective function to represent the violation of the constraints, the so-called "penalty function". One example of penalty function proposed in [73] is:

$$\Psi(\xi) = \frac{1}{2} \sum_{i=1}^{m} (\max\{0,\ \psi_i(\xi)\})^2 \tag{E.1.1}$$

and the penalized objective function is defined by:

$$\Upsilon(\xi) = \{\phi(\xi) + \eta\Psi(\xi)\}, \tag{E.1.2}$$

where η is a positive multiplicative penalizing constant. The penalized problem is then formulated as follows:

Problem E.1.1 (Penalized problem)**.** Find $\xi_\eta^* \in \mathbb{R}^n$ such that

$$\xi_\eta^* = \operatorname{argmin} \ \Upsilon(\xi) \tag{$\mathcal{P}_k$}$$

The idea is that by making the multiplicative constant η tend to infinity, the solution of the penalized problem converges to a solution for the original problem. [73, Chapter 13.1] present the following theorem:

Theorem E.1.1 (Penalty method convergence)**.** *Let C be a feasible constraint set in ($\mathcal{P}_0$). Let $(\eta_i)_{i\in\mathbb{N}}$ be a sequence tending infinity such that for each i, $\eta_i \geqslant 0$, $\eta_{i+1} > \eta_i$. Then, any limit point of $\left\{\xi_{\eta_i}^*\right\}$ is a solution to the original problem ($\mathcal{P}_0$).*

Notice that so far we did not present a method to solve ($\mathcal{P}_k$). Although several methods are available to address convex smooth optimization problems, the optimization problems we are interested in are characterized by non-differentiable objective functions and constraints. In the sequel, two methods that can be applied to this class of problems are presented: the quasi-Newton method proposed by Lewis and Overton in [64], which consists in building local quadratic approximations of the penalized function (E.1.2) and computing descent steps using the sub-gradients and the inverse Hessian; and the sub-gradient method proposed by Shor in [95], a first-order approach that resembles the steepest decent optimization algorithm, but uses sub-gradients instead of gradients.

E.2 Sub-gradient method

Before presenting the method, let us first give the formal definition of sub-gradients:

Definition E.2.1 (Sub-gradient)**.** *Let θ be a convex function with domain $I \subset \mathbb{R}^n$ and let ξ_0 be an interior point of I. A sub-gradient or a generalized gradient of θ at ξ_0 is any vector $g(\xi_0)$ satisfying*

$$\theta(\xi) - \theta(\xi_0) \geqslant \langle g(\xi_0), \xi - \xi_0 \rangle, \ \forall \xi \in I$$

where $\langle \cdot, \cdot \rangle$ is the scalar product in $\mathbb{R}^n$.

Techniques to compute a sub-gradient for different types of non-smooth functions are presented in the first chapter of [95]. The sub-gradient method consists in an algorithm that generates a sequence $\{\xi_k\}_{k=0}^{\infty}$ according to the formula

$$\xi_{k+1} = \xi_k - h_{k+1} g_\theta(\xi_k) \tag{E.2.1}$$

for a given convex function θ defined on E_n, the euclidean n-space, and an initial point ξ_0. Shor presents in [95, Theorem 2.2] the following theorem providing a method to solve the intermediate problems ($\mathcal{P}_k$):

Theorem E.2.1. *Let θ be a convex function defined on E_n which has a bounded set of minimum points I^* and let a sequence of positive numbers $\{h_k\}$, $k = 1, 2, \ldots$, satisfy the conditions:*

$$\lim_{k\to\infty} h_k = 0, \quad \sum_{k=1}^{\infty} h_k = +\infty$$

Then for any $\xi_0 \in E_n$ the sequence $\{\xi_k\}$, $k = 1, 2, \ldots$, generated according to the formula

$$\xi_{k+1} = \xi_k - h_{k+1} \frac{g_\theta(\xi_k)}{\|g_\theta(\xi_k)\|_2} \tag{E.2.2}$$

has the following property: either an index $\overline{k}$ exists such that $\xi_{\overline{k}} \in I^*$, or

$$\lim_{k\to\infty} \min_{\xi' \in I^*} \|\xi_k - \xi'\|_2 = 0 \quad \textit{and} \quad \lim_{k\to\infty} \theta(\xi_k) = \min_{\xi\in E_n} \theta(\xi) = \theta^*$$

E.3 BFGS method applied to non-smooth functions

The BFGS (Broyden-Fletcher-Goldfarb-Shanno) method is a quasi-Newton optimization method that, for a given continuous and differentiable function f, iteratively employs the known information about its gradients ∇f to build an estimate of its inverse Hessian $\nabla^2 f^{-1}$, performing local quadratic approximations of the original function.

Under certain assumptions [88], this method produces a sequence of iterations that converges to the minimal value of f. Nevertheless, Lewis and Overton have assessed the behavior of this method when applied to non-smooth functions, showing that convergence rates better than those obtained by the sub-gradient method are generally observed [63, 64]. The main shortcoming of this method is that it may *break down*, stagnating at a point at which the function is not differentiable.

In [64, Algorithm 2.1], Lewis and Overton define a quasi-Newton method as follows: *"let x_k denote the current point at iteration $k = 0, 1, \ldots$ The gradient of f at x_k is denoted $\nabla f(x_k)$*

and abbreviated to ∇f_k. We use H_k to denote a positive definite matrix which is an estimate of the inverse Hessian $\nabla^2 f(x_k)^{-1}$:"

Algorithm 3: Quasi-Newton algorithm

Require : x_0 with f differentiable at x_0, set H_0 to a positive definite matrix and $k \leftarrow 0$; tolerance ϵ

repeat
  set $p_k \leftarrow -H_k \nabla f_k$
  set $x_{k+1} \leftarrow x_k + t_k p_k$, where $t_k > 0$ is chosen by a line search
  if f is not differentiable at x_{k+1}, or $\nabla f_{k+1} = 0$, stop
  set $y_k \leftarrow \nabla f_{k+1} - \nabla f_k$ choose H_{k+1} to be a positive definite matrix satisfying the secant condition $H_{k+1} y_k = t_k p_k$
  $k \leftarrow k + 1$
until $\|\nabla f_{k+1}\|_2 < \epsilon$;

Output : x_{k+1}

For the BFGS method for non-smooth functions with inexact line search in particular, the update of H is made by:

$$H_{k+1} = V_k H_k V_k^T + t_k (p_k^T y_k)^{-1} p_k p_k^T, \quad \text{where} \quad V_k = I - (p_k^T y_k)^{-1} p_k y_k^T \tag{E.3.1}$$

and the line search algorithm is given by Algorithm 4 [64, Algorithm 4.6]. The Algorithms 3 and 4 are inspired on the implementation of the non-smooth BFGS method with inexact line search in a MATLAB®package called HANSO (Hybrid Algorithm for Non-Smooth Optimization[1], see [63]).

E.4 BFGS + sub-gradient hybrid method

Hereafter we present an algorithm based on the two optimization methods previously presented that can used to address the problem ($\mathcal{P}_k$). We opt to employ both the algorithms in order to take advantage of their distinct strengths: the quasi-Newton algorithm benefits of a faster decrease of the penalized objective function along the iterations, while only the sub-gradient method has guaranteed convergence, being used to refine the approximated solution obtained in the previous step. A pseudo-code representing the described strategy is given in Algorithm 5.

[1] http://www.cs.nyu.edu/overton/software/hanso/

Algorithm 4: Inexact line search algorithm

Require : Function f and its gradient ∇f, point x, direction p, $0 < c_1 < c_2 < 1$
$\alpha \leftarrow 0$
$\beta \leftarrow +\infty$
$t \leftarrow 1$
repeat
  if $f(x + t\,p) - f(x) \geqslant c_1\; t\,\nabla f(x)^T\,p$ **then**
    $\beta \leftarrow t$
  else if $\lim_{t\to 0} \frac{f(x+t\,p)-f(x)}{t}$ *does not exist* **then**
    stop, *break down*
  else if $\nabla f(x + t\,p)^T\,p \leqslant c_2\,\nabla f(x)^T\,p$ **then**
    $\alpha \leftarrow t$
  else
    stop, *admissible step found*
  if $\beta < +\infty$ **then**
    $t \leftarrow (\alpha + \beta)/2$
  else
    $t \leftarrow 2\alpha$
  end
until *Maximum number of iterations reached*;
Output : t

E.5 Testing the algorithms

Hereafter, we assess the optimization methods previously presented by analyzing their performances when applied to the following non-smooth optimization problem:

$$[x^*,\ y^*] = \left\{ \begin{array}{ll} \text{argmin} & |x| + |y| \\ \text{s.t.} & (x-10)^2 - y \leqslant 0 \end{array} \right. \tag{Ex}$$

which has $[x^*,\ y^*] = [9.50,\ 0.25]$ as solution.

We define the penalized problems as:

$$[x^*,\ y^*]_\eta = \text{argmin}\left\{|x| + |y| + \eta\,.\,\max\left\{0, (x-10)^2 - y\right\}\right\} \tag{Ex'}$$

The following expression provides one sub-gradient of the penalized objective function at any arbitrary point $(x,\ y)$:

$$g(x,y) = \begin{bmatrix} \text{sgn}(x) \\ \text{sgn}(y) \end{bmatrix} + \eta\,.\,\mathbf{1}_{\{(x-10)^2 - y \geqslant 0\}}(x,y) \begin{bmatrix} 2(x-10) \\ -1 \end{bmatrix} \tag{E.5.1}$$

Algorithm 5: Algorithm to solve non-smooth convex optimization problems

Input : ξ_1 - initial point
$\Upsilon(\cdot)$ - non-smooth convex function to minimize
$g(\cdot)$ - oracle that returns one of the sub-gradients of $\Upsilon(\cdot)$ at a given point
I_{qn}, I_{sg} - number of iterations
σ - coefficient for the sub-gradient steps (set to 1 if not explicitly specified)
Output : ξ - solution

$\xi_{best} \leftarrow \xi_1$
// Quasi-Newton method
// [64, Algorithm 2.1]:
$H_1 = \mathbb{I}_n$;
for $k = 1$ **to** I_{qn} **do**
  $\xi_{k+1} \leftarrow \xi_k - \lambda_k H_k g(\xi_k)$, where $\lambda_k > 0$ computed by inexact line search;
  if $\Upsilon(\xi_{k+1}) < \Upsilon(\xi_{best})$ **then**
    $\xi_{best} \leftarrow \xi_{k+1}$
  end
  Update H_{k+1} as positive definite matrix satisfying secant condition
  $H_{k+1}(g(\xi_{k+1}) - g(\xi_k)) = -\lambda_k H_k g(\xi_{k+1})$;
end
$\xi_1 \leftarrow \xi_{best}$;
// Subgradient method
// [95, Theorem 2.2]:
for $k = 1$ **to** I_{sg} **do**
  $\xi_{k+1} \leftarrow \xi_k - \sigma \frac{1}{k} \frac{g(\xi_k)}{\|g(\xi_k)\|_2}$;
  if $\Upsilon(\xi_{k+1}) < \Upsilon(\xi_{best})$ **then**
    $\xi_{best} \leftarrow \xi_{k+1}$
  end
end
$\xi \leftarrow \xi_{best}$;

where sgn(·) is the signum function and $\mathbf{1}_A(\cdot)$ is the indicator function defined as

$$\mathbf{1}_A(x) := \begin{cases} 1, & \text{if } x \in A \\ 0, & \text{if } x \notin A \end{cases}$$

E.5.1 Sub-gradient method

We take $[x, y]_0 = [0, 0]$ as initial point, $\eta = 10^8$ as penalty parameter and the sequence $\{h_k\} = \sigma/k$, and we compute the iterations given in (E.2.2). The Fig. E.1 represents the evolution of the distance between the points obtained at each iteration and the point $[9.50, 0.25]$ (at which the optimum is attained) for four distinct values of $\sigma = 1,\ 2,\ 3,\ 10$.

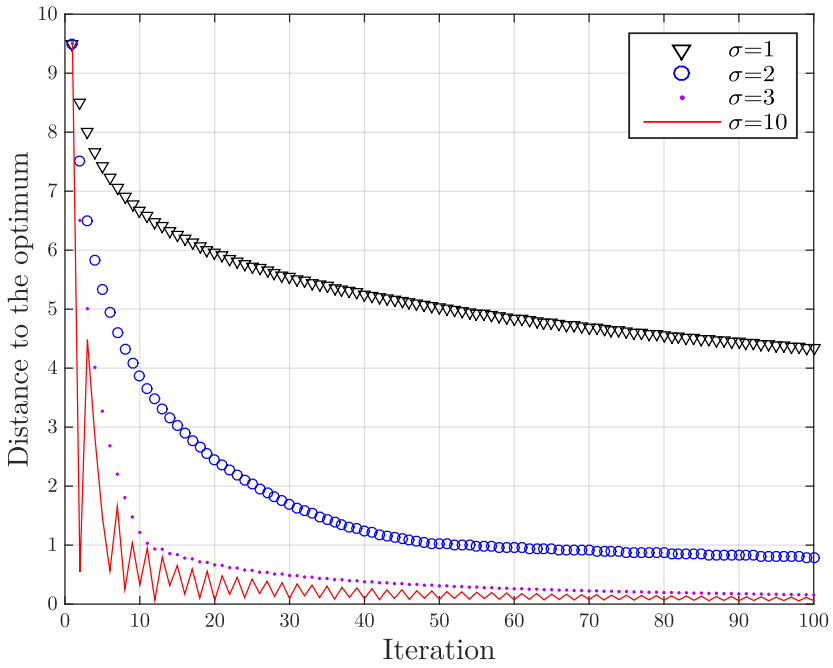


Figure E.1 – Performance of the sub-gradient method

We remark that the choice of σ can speed up or retard the convergence and, for the case $\sigma = 10$, an oscillatory behavior is observed. However, for all the cases, the distance between the iterations presents a tendency to converge to zero.

E.5.2 BFGS method

Using the BFGS method given by Algorithms 3 and 4, we obtain the evolution presented in Fig. E.2.

We observe that the decrease of the distance to the optimum is much steeper than that presented by the curves in Fig. E.1, which confirms the fact that the quasi-Newton method presents better convergence rates that the sub-gradient method. Nevertheless, even before

the 20 first iterations, the decrease of the distance to the optimum stagnates and no further improvements are possible (this is due to the fact that the iterations converge to a point where the function is not differentiable, see Fig. E.3).

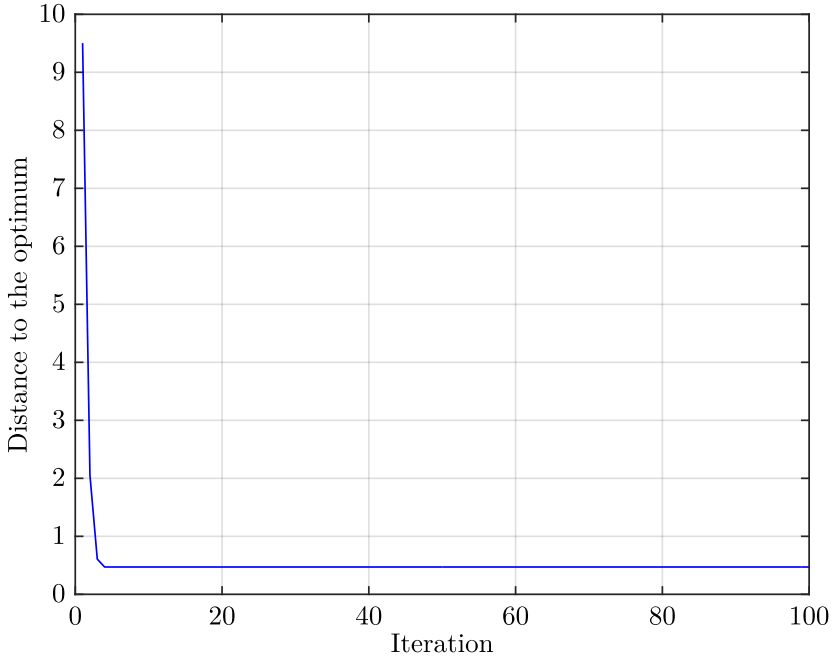


Figure E.2 – Performance of the BFGS method

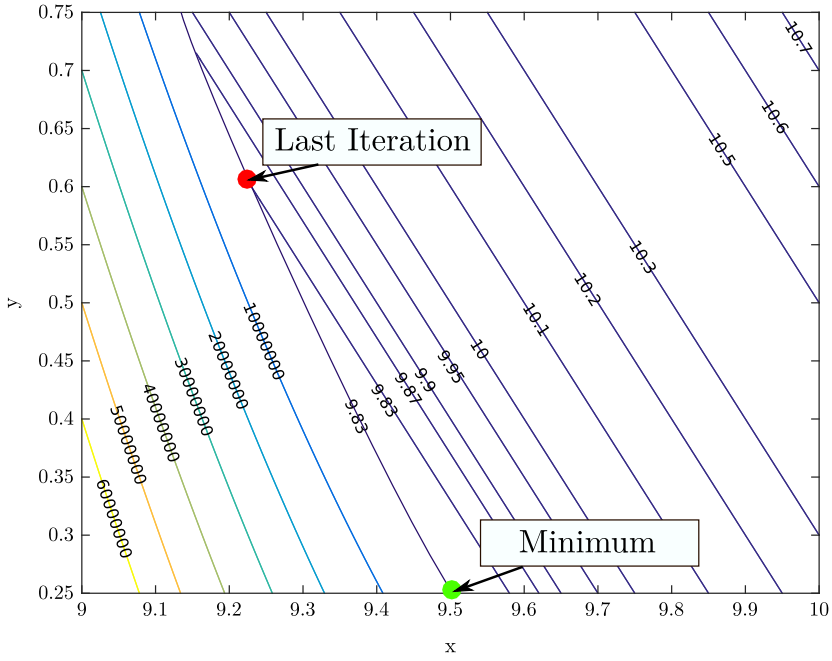


Figure E.3 – Last iteration before stagnation of BFGS method

E.5.3 BFGS + sub-gradient hybrid method

Using the hybrid method described by the Algorithm 5 with 5 iterations of the BFGS method and 20 iterations of sub-gradient method with $\sigma = 1$, we obtain the evolution presented in Fig. E.4.

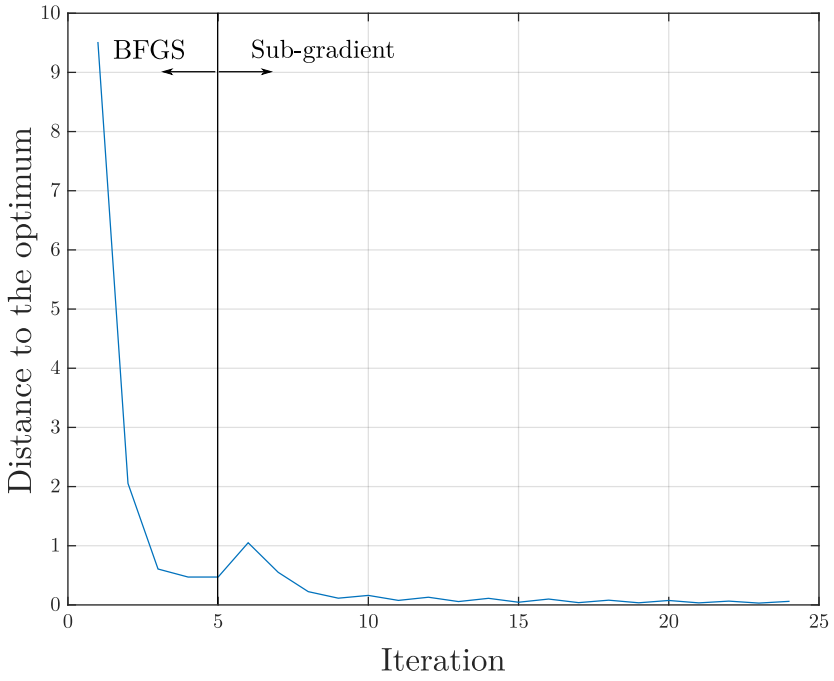


Figure E.4 – Performance of the hybrid algorithm (BFGS + sub-gradient)

After employing this performing a total number of 25 iterations of this algorithm, the distance to the optimum is already smaller that those obtained after 100 iterations in Fig. E.1 and Fig. E.2. We also observe that the distinct strengths of both methods are present: a very steep decrease during the BFGS method and convergence to zero during the sub-gradient steps.

Bibliography

[1] Abramowitz, M., and Stegun, I. A. *Handbook of mathematical functions: with formulas, graphs, and mathematical tables*. No. 55. Courier Corporation, 1964. 48

[2] Alfriend, K. T., and Schaub, H. Dynamic and control of spacecraft formations: challenges and some solutions. *Journal of Astronautical Sciences 48*, 2 (2000), 249–267. 18

[3] Andersen, E., Roos, C., and Terlaky, T. On implementing a primal-dual interior-point method for conic quadratic optimization. *Mathematical Programming 95*, 2 (Feb 2003), 249–277. 52

[4] Arantes Gilz, P. R. A Matlab®/Simulink® non-linear simulator for orbital spacecraft rendezvous applications., Dec. 2016. 4, 18

[5] Arantes Gilz, P. R. A non-linear simulator written in C for orbital spacecraft rendezvous applications., Dec. 2016. 4, 18

[6] Arantes Gilz, P. R., Bréhard, F., and Gazzino, C. Validated Semi-Analytical Transition Matrices for Linearized Relative Spacecraft Dynamics via Chebyshev Series Approximations. In *SciTech 2018 - AIAA Science and Technology Forum and Exposition, 28th Space Flight Mechanics Meeting* (Kissimmee, United States, Jan. 2018), AIAA, pp. 1–23. 6, 92

[7] Arantes Gilz, P. R., Joldeş, M., Louembet, C., and Camps, F. Model predictive control for rendezvous hovering phases based on a novel description of constrained trajectories. In *IFAC World Congress* (Toulouse, France, July 2017), pp. pp. 7490–7495. 5, 25, 41, 46, 63

[8] Arantes Gilz, P. R., Joldes, M., Louembet, C., and Camps, F. Stable Model Predictive Strategy for Rendezvous Hovering Phases Allowing for Control Saturation. working paper, Jan. 2018. 5

[9] Arantes Gilz, P. R., and Louembet, C. Predictive control algorithm for spacecraft rendezvous hovering phases. In *Control Conference (ECC), 2015 European* (2015), IEEE, pp. 2085–2090. 25, 41

[10] Arzelier, D., Théron, A., and Kara-Zaitri, M. Étude bibliographique sur la modélisation du mouvement relatif pour le problème du rendezvous, convention cnes no. 71372/00, eads-astrium. Tech. rep., LAAS-CNRS/CNES, 2008. 15, 16

[11] Aubin, J.-P., Lygeros, J., Quincampoix, M., Sastry, S., and Seube, N. Impulse differential inclusions: A viability approach to hybrid systems. *IEEE Transactions on Automatic Control 47*, 1 (2002), 2–20. 31

[12] Bainov, D., and Simeonov, P. *Impulsive Differential Equations: Periodic Solutions and Applications*. Monographs and Surveys in Pure and Applied Mathematics. Taylor & Francis, 1993. 31

[13] Battin, R. H. *An introduction to the mathematics and methods of astrodynamics*. AIAA, 1999. 10

[14] Betts, J. T. Survey of numerical methods for trajectory optimization. *Journal of Guidance, Control, and Dynamics 21*, 2 (1998), 193–207. 37

[15] Bodin, P., Larsson, R., Nilsson, F., Chasset, C., Noteborn, R., and Nylund, M. Prisma: an in-orbit test bed for guidance, navigation, and control experiments. *Journal of Spacecraft and Rockets 46*, 3 (2009), 615. 78

[16] Borchers, B. CSDP, A C library for semidefinite programming. *Optimization Methods and Software 11*, 1-4 (1999), 613–623. 52, 78

[17] Borchers, B. CSDP User's Guide, June 2006. Accessed Mar. 7, 2018. 52

[18] Boyd, J. P. *Chebyshev and Fourier spectral methods*. Dover Publications, 2001. 93, 94, 95, 96, 101

[19] Breger, L., and How, J. P. Gauss's variational equation-based dynamics and control for formation flying spacecraft. *Journal of Guidance, Control, and Dynamics 30*, 2 (2007), 437–448. 18

[20] Bréhard, F. A Newton-like Validation Method for Chebyshev Approximate Solutions of Linear Ordinary Differential Systems. In *ISSAC 2018* (New York, United States, July 2018). 92, 95, 96, 98

[21] Bréhard, F., Brisebarre, N., and Joldes, M. Validated and numerically efficient Chebyshev spectral methods for linear ordinary differential equations. *ACM Transactions on Mathematical Software* (2018), 1–50. 92, 95, 96, 98

[22] Brentari, M., Arzelier, D., Louembet, C., Urbina, S., and Zaccarian, L. A hybrid control framework for impulsive control of satellite rendezvous. In *2016 American Control Conference (ACC)* (July 2016), pp. 7414–7419. 25, 41, 68, 78, 79, 80

[23] Brown, C. D. *Spacecraft propulsion*. Aiaa, 1996. 30

[24] Camps, F., Gilz, P. R. A., Joldes, M., and Louembet, C. Embedding a sdp-based control algorithm for the orbital rendezvous hovering phases. In *2018 25th Saint Petersburg International Conference on Integrated Navigation Systems (ICINS)* (May 2018), pp. 1–7. 5, 50, 106

[25] Chari, R. J. Autonomous orbital rendezvous using angles-only navigation. Tech. rep., Massachusetts Institute of Technology, 2001. 39

[26] Claeys, M., Arzelier, D., Henrion, D., and Lasserre, J.-B. Moment LMI approach to LTV impulsive control. In *Decision and Control (CDC), 2013 IEEE 52nd Annual Conference on* (2013), IEEE, pp. 5810–5815. 93

[27] Czysz, P. A., and Bruno, C. *Future spacecraft propulsion systems: enabling technologies for space exploration*. Springer Science & Business Media, 2006. 30

[28] Deaconu, G. *On the trajectory design, guidance and control for spacecraft rendezvous and proximity operations*. PhD thesis, Univ. Toulouse 3 - Paul Sabatier, Toulouse, France, Oct. 2013. 23, 26, 44, 63, 81, 106, 123

[29] Deaconu, G., Louembet, C., and Theron, A. Constrained periodic spacecraft relative motion using non-negative polynomials. In *American Control Conference (ACC), 2012* (2012), IEEE, pp. 6715–6720. 8, 41, 44, 106

[30] Deaconu, G., Louembet, C., and Théron, A. Designing continuously constrained spacecraft relative trajectories for proximity operations. *Journal of Guidance, Control, and Dynamics 38*, 7 (2015), 1208–1217. 41, 46

[31] Di Cairano, S., Park, H., and Kolmanovsky, I. Model predictive control approach for guidance of spacecraft rendezvous and proximity maneuvering. *International Journal of Robust and Nonlinear Control 22*, 12 (2012), 1398–1427. 18

[32] Di Mauro, G., Schlotterer, M., Theil, S., and Lavagna, M. Nonlinear control for proximity operations based on differential algebra. *Journal of Guidance, Control, and Dynamics 38*, 11 (2015), 2173–2187. 92

[33] D'Amico, S., Ardaens, J.-S., Gaias, G., Benninghoff, H., Schlepp, B., and Jørgensen, J. Noncooperative rendezvous using angles-only optical navigation: system design and flight results. *Journal of Guidance, Control, and Dynamics 36*, 6 (2013), 1576–1595. 40, 41

[34] El'iasberg, P. E. *Introduction to the Theory of Flight of Artificial Earth Satellites* (Введение в теорию полета искусственных спутников Земли). Israel Program for Scientific Translations, 1967. 15

[35] ESA. Onboard computer and data handling, May 2014. Accessed Mar. 7, 2018. 50

[36] Fahroo, F., and Ross, I. M. Direct trajectory optimization by a chebyshev pseudospectral method. *Journal of Guidance, Control, and Dynamics 25*, 1 (2002), 160–166. 93

[37] Fehse, W. *Automated rendezvous and docking of spacecraft*, vol. 16. Cambridge university press, 2003. 1, 13, 27

[38] Fraga, S. L., and Pereira, F. L. Hamilton-jacobi-bellman equation and feedback synthesis for impulsive control. *IEEE Transactions on Automatic Control 57*, 1 (2012), 244–249. 31

[39] Fujisawa, K., Kojima, M., Nakata, K., and Yamashita, M. Sdpa (semidefinite programming algorithm) user's manual—version 6.2. 0. *Department of Mathematical and Com-puting Sciences, Tokyo Institute of Technology. Research Reports on Mathematical and Computing Sciences Series B: Operations Research* (2002). 52

[40] Gaias, G., D'Amico, S., and Ardaens, J.-S. Angles-only navigation to a noncooperative satellite using relative orbital elements. *Journal of Guidance, Control, and Dynamics 37*, 2 (2014), 439–451. 40

[41] GLPK. GLPK (GNU linear programming kit v4.6), June 2012. Accessed Mar. 7, 2018. 50

[42] González, A. H., Rivadeneira, P. S., Ferramosca, A., Magdelaine, N., and Moog, C. H. Impulsive zone mpc for type i diabetic patients based on a long-term model. *IFAC-PapersOnLine 50*, 1 (2017), 14729 – 14734. 20th IFAC World Congress. 63

[43] Gottlieb, D., and Orszag, S. A. *Numerical Analysis of Spectral Methods: Theory and Applications*, vol. 26. Siam, 1977. 93

[44] Gurfil, P. Relative motion between elliptic orbits: generalized boundedness conditions and optimal formationkeeping. *Journal of Guidance, Control, and Dynamics 28*, 4 (2005), 761–767. 8, 25

[45] Haddad, W. M., Chellaboina, V., and Nersesov, S. G. Impulsive and hybrid dynamical systems. *Princeton Series in Applied Mathematics* (2006). 31

[46] Hall, R., and Shayler, D. *Soyuz: a universal spacecraft*. Springer Science & Business Media, 2003. 1

[47] Haranas, I. I., and Harney, M. Gauss planetary equations in a non-singular gravitational potential. 8, 13

[48] Hartley, E. N., and Maciejowski, J. M. Predictive control for spacecraft rendezvous in an elliptical orbit using an FPGA. In *Control Conference (ECC), 2013 European* (2013), IEEE, pp. 1359–1364. 62

[49] Hartley, E. N., and Maciejowski, J. M. Field programmable gate array based predictive control system for spacecraft rendezvous in elliptical orbits. *Optimal Control Applications and Methods 36*, 5 (2015), 585–607. 62

[50] Hartley, E. N., Trodden, P. A., Richards, A. G., and Maciejowski, J. M. Model predictive control system design and implementation for spacecraft rendezvous. *Control Engineering Practice 20*, 7 (2012), 695 – 713. 18, 43

[51] Helmberg, C., Rendl, F., Vanderbei, R. J., and Wolkowicz, H. An interior-point method for semidefinite programming. *SIAM Journal on Optimization 6*, 2 (1996), 342–361. 52

[52] Holzinger, M., DiMatteo, J., Schwartz, J., and Milam, M. Passively safe receding horizon control for satellite proximity operations. In *Decision and Control, 2008. CDC 2008. 47th IEEE Conference on* (2008), IEEE, pp. 3433–3440. 18

[53] Hong, H. Implicitization of curves parameterized by generalized trigonometric polynomials. In *Applied Algebra, Algebraic Algorithms and Error-Correcting Codes*, vol. 948. Springer Berlin Heidelberg, 1995, pp. 285–296. 46

[54] Hope, A., and Trask, A. Pulsed thrust method for hover formation flying. 3, 38

[55] Humi, M. Fuel-optimal rendezvous in a general central force field. *Journal of Guidance, Control, and Dynamics 16*, 1 (1993), 215–217. 121

[56] Irvin, D. J., Cobb, R. G., and Lovell, T. A. A general methodology for minimum-fuel hovering satellite formation. *Advances in the Astronautical Sciences 129* (2007), 329–349. 3, 38

[57] Irvin, D. J., Cobb, R. G., and Lovell, T. A. Fuel-optimal maneuvers for constrained relative satellite orbits. *Journal of guidance, control, and dynamics 32*, 3 (2009), 960–973. 39

[58] Irvin, D. J., Cobb, R. G., and Lovell, T. A. An investigation of teardrop relative orbits for circular and elliptical chief satellites. *Advances in the Astronautical Sciences 134* (2009), 121–140. 3, 38, 39

[59] Joldeş, M. *Rigorous Polynomial Approximations and Applications*. PhD thesis, École normale supérieure de Lyon – Université de Lyon, Lyon, France, 2011. 93

[60] Kara-Zaitri, M. *Modélisation et guidage robuste et autonome pour le problème du rendez-vous orbital*. PhD thesis, Univ. Toulouse 3 - Paul Sabatier, Toulouse, France, oct 2010. (In French). 15, 16, 18

[61] Kubota, T., Hashimoto, T., Kawaguchi, J., Uo, M., and Shirakawa, K. Guidance and navigation of hayabusa spacecraft for asteroid exploration and sample return mission. In *SICE-ICASE, 2006. International Joint Conference* (2006), IEEE, pp. 2793–2796. 1

[62] Lasserre, J.-B. *Moments, positive polynomials and their applications*, vol. 1. World Scientific, 2009. 93

[63] Lewis, A. S., and Overton, M. L. Nonsmooth optimization via BFGS. 133, 134

[64] Lewis, A. S., and Overton, M. L. Nonsmooth optimization via quasi-newton methods. *Mathematical Programming 141*, 1-2 (2013), 135–163. 55, 132, 133, 134, 136

[65] Lizia, P. D., Armellin, R., and Lavagna, M. Application of high order expansions of two-point boundary value problems to astrodynamics. *Celestial Mechanics and Dynamical Astronomy 102*, 4 (2008), 355–375. 92

[66] Lizia, P. D., Armellin, R., Morselli, A., and Bernelli-Zazzera, F. High order optimal feedback control of space trajectories with bounded control. *Acta Astronautica 94*, 1 (2014), 383 – 394. 92

[67] Löfberg, J. YALMIP: A toolbox for modeling and optimization in MATLAB. In *Computer Aided Control Systems Design, 2004 IEEE International Symposium on* (2004), IEEE, pp. 284–289. 106

[68] López, M., and Still, G. Semi-infinite programming. *European Journal of Operational Research 180*, 2 (2007), 491–518. 30, 37

[69] Louembet, C., and Arantes Gilz, P. R. Event-triggered Model predictive control for spacecraft rendezvous. In *6th IFAC Conference on Nonlinear Model Predictive Control (NMPC 2018)* (Madison, United States, Aug. 2018), p. 2p. 115

[70] Louembet, C., Arzelier, D., and Deaconu, G. Robust rendezvous planning under maneuver execution errors. *Journal of Guidance, Control, and Dynamics 38*, 1 (Oct. 2015), 76–93. 61

[71] Lovell, T. A., and Brown, D. Impulsive-hover satellite trajectory design for rendezvous and proximity operation missions. *AAS Rocky Mountain Guidance, Navigation and Control Conference* (2007). 3, 38

[72] Lovell, T. A., and Tollefson, M. V. Calculation of impulsive hovering trajectories via relative orbit elements. *Advances in the Astronautical Sciences 123* (2006), 2533–2548. 3, 38

[73] Luenberger, D. G., and Ye, Y. *Linear and nonlinear programming*, vol. 2. Springer, 1984. 131, 132

[74] Luo, Y.-Z., Tang, G.-J., and Lei, Y.-J. Optimal multi-objective linearized impulsive rendezvous. *Journal of guidance, control, and dynamics 30*, 2 (2007), 383–389. 3, 38, 39

[75] Marcuccio, S., Genovese, A., and Andrenucci, M. Experimental performance of field emission microthrusters. *Journal of Propulsion and Power 14*, 5 (1998), 774–781. 30

[76] Mason, J. C., and Handscomb, D. C. *Chebyshev polynomials*. CRC Press, 2002. 95

[77] Montenbruck, O., and Gill, E. *Satellite orbits: models, methods and applications*. Springer Science & Business Media, 2012. 12

[78] Morari, M., and Lee, J. H. Model predictive control: past, present and future. *Computers and Chemical Engineering 23*, 4 (1999), 667 – 682. 63

[79] Mueller, J. B., and Larsson, R. Collision avoidance maneuver planning with robust optimization. In *International ESA Conference on Guidance, Navigation and Control Systems, Tralee, County Kerry, Ireland* (2008). 61

[80] NASA. ISS Trajectory Data, August 2009. Accessed Mar. 7, 2018. 15, 57

[81] Nesterov, Y. Squared functional systems and optimization problems. In *High performance optimization*. Springer, 2000, pp. 405–440. 44, 49, 105, 106, 123

[82] Neustadt, L. Optimization, a moment problem, and nonlinear programming. *Journal of the Society for Industrial and Applied Mathematics Series A Control 2*, 1 (1964), 33–53. 31, 66

[83] Olver, S., and Townsend, A. A fast and well-conditioned spectral method. *SIAM Review 55*, 3 (2013), 462–489. 96, 102

[84] PENDER. GR-XC6S Development Board - User Manual, July 2011. Accessed Mar. 7, 2018. 50

[85] Pereira, F. L., Fontes, F. A. C. C., Pedro Aguiar, A., and Borges de Sousa, J. *An Optimization-Based Framework for Impulsive Control Systems*. Springer International Publishing, Cham, 2015, pp. 277–300. 63

[86] Pereira, F. L., and Silva, G. N. Stability for impulsive control systems. *Dynamical Systems 17*, 4 (2002), 421–434.

[87] Pereira, F. L., Silva, G. N., and Oliveira, V. Invariance for impulsive control systems. *Automation and Remote Control 69*, 5 (2008), 788–800. 63

[88] Powell, M. J. D. Some global convergence properties of a variable metric algorithm for minimization without exact line searches. *Nonlinear programming 9*, 1 (1976), 53–72. 133

[89] Riccardi, A., Tardioli, C., and Vasile, M. *An intrusive approach to uncertainty propagation in orbital mechanics based on Tchebycheff polynomial algebra*. Advances in Astrnautical Sciences, AAS/AIAA Astrodynamics Specialist Conference, August 9-13, 2015, Vail, Colorado, U.S.A. American Astronautical Society, 8 2015, pp. 707–722. 92

[90] Richards, A., How, J., Schouwenaars, T., and Feron, E. Plume avoidance maneuver planning using mixed integer linear programming. In *Proceedings of the AIAA Guidance, Navigation, and Control Conference* (2001), pp. 6–9. 43

[91] Richards, A., Schouwenaars, T., How, J. P., and Feron, E. Spacecraft trajectory planning with avoidance constraints using mixed-integer linear programming. *Journal of Guidance, Control, and Dynamics 25*, 4 (2002), 755–764. 43

[92] Ries, J., Eanes, R., Huang, C., Schutz, B., Shum, C., Tapley, B., Watkins, M., and Yuan, D. Determination of the gravitational coefficient of the earth from near-earth satellites. *Geophysical research letters 16*, 4 (1989), 271–274. 10

[93] Rivadeneira, P. S., Ferramosca, A., and González, A. H. Mpc with state window target control in linear impulsive systems. *IFAC-PapersOnLine 48*, 23 (2015), 507 – 512. 5th IFAC Conference on Nonlinear Model Predictive Control NMPC 2015. 63

[94] Ross, I. M. Space trajectory optimization and l1-optimal control problems. *Modern astrodynamics 1* (2006), 155. 30

[95] Shor, N. Z. *Minimization methods for non-differentiable functions*, vol. 3. Springer-Verlag, 1985. 55, 132, 133, 136

[96] Sopasakis, P., Patrinos, P., Sarimveis, H., and Bemporad, A. Model predictive control for linear impulsive systems. In *Decision and Control (CDC), 2012 IEEE 51st Annual Conference on* (2012), IEEE, pp. 5164–5169. 63

[97] Sturm, J. F. Using sedumi 1.02, a matlab toolbox for optimization over symmetric cones. *Optimization methods and software 11*, 1-4 (1999), 625–653. 52

[98] Toh, K.-C., Todd, M. J., and Tütüncü, R. H. Sdpt3—a matlab software package for semidefinite programming, version 1.3. *Optimization methods and software 11*, 1-4 (1999), 545–581. 52

[99] Tschauner, J. Elliptic orbit rendezvous. *AIAA Journal 5*, 6 (1967), 1110–1113. 8, 20

[100] Tschauner, J., and Hempel, P. Rendezvous zu einem in elliptischer bahn umlaufenden ziel. *Astronautica Acta 11*, 2 (1965). 8, 19

[101] Tucker, W. *Validated numerics: a short introduction to rigorous computations*. Princeton University Press, 2011. 106

[102] Turner, M. J. *Rocket and spacecraft propulsion: principles, practice and new developments*. Springer Science & Business Media, 2008. 30

[103] Tütüncü, R. H., Toh, K. C., and Todd, M. J. Solving semidefinite-quadratic-linear programs using sdpt3. *Mathematical programming 95*, 2 (2003), 189–217. 106

[104] Walker, M., Ireland, B., and Owens, J. A set modified equinoctial orbit elements. *Celestial mechanics 36*, 4 (1985), 409–419. 14

[105] Williams, J., and Lightsey, E. Optimal impulsive maneuvering within a confined hover region. In *AIAA Guidance, Navigation and Control Conference and Exhibit*, p. 6835. 3, 38, 39

[106] Woffinden, D. C. *On-orbit satellite inspection: navigation and [Delta] v analysis*. PhD thesis, Massachusetts Institute of Technology, 2004. 39, 40

[107] Yalcin, H., Unel, M., and Wolovich, W. Implicitization of parametric curves by matrix annihilation. *IEEE International Conference on Image Processing (ICIP '02)* (September 2002), 22–25. 126

[108] Yamamoto, N. A numerical verification method for solutions of boundary value problems with local uniqueness by Banach's fixed-point theorem. *SIAM Journal on Numerical Analysis 35*, 5 (1998), 2004–2013. 97, 98

[109] Yamanaka, K., and Ankersen, F. New state transition matrix for relative motion on an arbitrary elliptical orbit. *Journal of guidance, control, and dynamics 25*, 1 (2002), 60–66. 8, 20, 22

[110] Zemke, J. b4m: A free interval arithmetic toolbox for matlab, 1999. 106

[111] Zhan, J., and Li, X. Flocking of multi-agent systems via model predictive control based on position-only measurements. *IEEE Transactions on Industrial Informatics 9*, 1 (Feb 2013), 377–385. 63