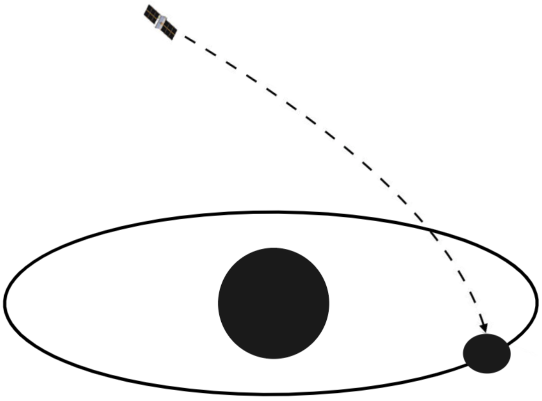
POLITECNICO
MILANO 1863

2021

LANDING A CUBESAT ON AN ASTEROID'S MOON

Guidance & Control design for asteroid landing using discrete firings of closed-loop sliding mode control



ÍCAR FONTCUBERTA

POLITECNICO DI MILANO
School of Industrial and
Information Engineering
MSc in Space Engineering
ID: 940343 // 10718929

Design, implementation and evaluaton of a Guidance & Control scheme for Milani CubeSat's landing on 65803 Didymos' moon.

Supervised by:
Dr. Fabio Ferrari
Prof. Dr. Francesco Topputo

Acknowledgments

This work, and maybe even these words, mark the end of my stay at Politecnico: two years of hard work and professional growth. I look back to September 2019 and I could not be happier about the decision I made to seize my time and truly focus on engineering. During these years, I gained the confidence in my skills that the job requires. Having the opportunity to think about how to land a spacecraft on an asteroid seems a dream, actually the result of constancy and perseverance.

I would like to thank Prof. Francesco Topputo for giving me the opportunity of developing this work even in the midst of the pandemic. Also, I would like to thank Dr. Fabio Ferrari for his trust in my work and his always precise and relieving feedback.

To all my friends, who have been here all along this journey: *gràcies*, *gracias*, *grazie*, *merci*, thank you. To learn Aerospace Engineering has been an amazing adventure, yet not always pleasurable. I would not have made it without you.

I would also like to thank my sister, for her advice in all that I lack. To my father, always interested and invested in my studies, who points the way. And to my mother, who will always be my stronger support.

Ícar

Abstract

Spacecraft bound to land on small bodies require reliable and robust Guidance & Control systems. This applies to Hera's Milani CubeSat, which will land on (65803) Didymos' moon, Dimorphos. In this work is selected an existing control algorithm to land Milani on Dimorphos and adapted to fire in $300s$ intervals. The landing performance is evaluated using Monte Carlo analyses. Here the Multiple Sliding Surface Guidance (MSSG) is selected. MSSG uses closed-loop sliding mode control to reject bounded perturbations and synchronize the spacecraft with the asteroid's body-attached reference frame. The MSSG is implemented using the Two Phased Descent (TPD) strategy, in which the thrust is first turned on and off in intervals of $300s$. Next, upon crossing a height of $15m$ above Dimorphos, the control is turned on for $300s$ more regardless of the control's past state. The TPD performance is mainly impacted by navigation position knowledge. With a position error of $\sigma_r \approx 1m$, collision velocities remain under $4.5cm/s$ with 98% of confidence. The incidence angles are of $\approx 20\,\mathrm{deg}$ and the landing ellipse is $\sigma_{r,land} \approx 1.1m$ wide. Fuel consumption is about $10g$ for I_{sp} between $40s$ and $80s$.

Sommario

Il Multiple Sliding Surface Guidance (MSSG) viene selezionato come algoritmo di controllo per l'atterraggio del CubeSat Milani su Dimorphos, la luna dell'asteroide 65803 Didymos. Nella selezione è stato ipotizzato che il suo sistema GNC possa eseguire una closed-loop navigation con continuous thrust. La scelta è guidata dalla robustezza dell'MSSG alle perturbazioni limitate, dalla sua semplicità e dal consumo di CPU relativamente basso. L'MSSG utilizza il sliding mode control per rifiutare le incertezze dinamiche limitate in un tempo finito, annulla le forze ambientali modellate e dopo genera un'accelerazione per impostare il s/c su una traiettoria idealmente lineare nel sistema di riferimento attaccato al corpo di Dimorphos. Le vibrazioni sono state mitigate utilizzando uno Schmitt Trigger e una boundary layer attorno al Sliding Manifold.

Viene studiato il comportamento del MSSG nell'ambiente dinamico di Didymos e mostra che i) tempi di atterraggio successivi al passaggio ravvicinato di una traiettoria di riferimento e ii) tempi di vuoli più lunghi che $300s$ (la rilevanza di questo limite è spiegata di seguito) sono vantaggiosi in termini di consumo di carburante (Δm) e velocità di collisione (V_c). Per valutare le prestazioni del MSSG, prima vengono selezionati i parametri di controllo (Λ e n) utilizzando una "traiettoria di atterraggio nominale", definita da un'epoca di inizio del controllo e un'epoca di atterraggio. Gli stati iniziali sono tratti da la citata traiettoria di riferimento. La "traiettoria di atterraggio nominale" dura $1h$ ed è lunga $\approx 0,5km$.

Dopo, viene progettata la strategia di accensione Two Phase Descent (TPD), in modo che il s/c esegua solo lanci di $300s$ durante $\approx 95\%$ della traiettoria. Nel TPD, la traiettoria di atterraggio è divisa in due fasi: in primo luogo, durante la Fase di Avvicinamento il controllo viene attivato e disattivato in sequenza. Lungo la citata "traiettoria nominale" questo fa sì che il $\approx 95\%$ della distanza da percorrere. Successivamente, a $15m$ sopra la superficie di Dimorphos, inizia la Fase di Discesa. In questo periodo, il controllo viene attivato per un massimo di $300s$ in attesa del touchdown, indipendentemente dallo stato del controllo all'inizio della fase. Per il TPD si presume che sia possibile una conoscenza di navigazione di $\sigma_r \approx 1m$. Un σ_r di questo ordine è richiesto per evitare di prendere di mira un punto più in basso o molto al di sopra della superficie di Dimorphos, che metterebbe in pericolo il successo dell'atterraggio. I parametri di controllo per le fasi Avvicinamento (1) e Discesa (2) sono diversi e utilizzando un processo di progettazione iterativo sono stati impostati su $\Lambda_1 = 2,5$, $n_1 = 0,2$, $\Lambda_2 = 3$ e $n_2 = 0,7$.

Infine, le prestazioni di atterraggio del TPD vengono valutate utilizzando un analisi Monte Carlo. Questi mostrano come le prestazioni di atterraggio siano maggiormente influenzate dalla conoscenza della navigazione. Con un errore di navigazione di $\sigma_r \approx 1m$, l'errore di dispersione degli atterraggi rispetto al punto di approdo utilizzando una perfetta conoscenza della navigazione è di $\sigma_{r,land} \approx 1.1m$. Le traiettorie tipicamente collidono con un angolo di $\approx 23\,\mathrm{deg}$ rispetto al Nadir locale. L'angolo di incidenza ha una dispersione di $\sigma_{angle} \approx 17\,\mathrm{deg}$. Il limite di $V_c < 4.5cm/s$ (velocità approssimata di scapo di Dimorphos) è rispettato con $\approx 98\%$ di confidenza. La componente locale-Nadir di V_c ha un valore tipico di $\approx 2cm/s$ con una dispersione di $\sigma \approx 1.3cm/s$. I componenti off-local-Nadir di V_c rimangono a valori inferiori (ordine di $0.5cm/s$) e seguono da vicino una distribuzione gaussiana. Il controllo è stato messo in discussione con errori di puntamento 1 deg e 5 deg, che hanno mostrato di avere un lieve impatto sulle prestazioni. Nel complesso, la forte autorità di controllo del MSSG è in grado di correggere rapidamente gli errori rispetto alla traiettoria lineare ideale. Può respingere le perturbazioni fino alla spinta massima del CubeSat, al momento della scrittura di $10mN$. Il consumo di carburante è strettamente correlato al valore di I_{sp}, ma sempre localizzato aprossimatamente a $\mathcal{O}(I_{sp}) = 10g$, ed è stato rilevato che l'aumento delle prestazioni è $-0.2g$ per $sec.$ di I_{sp}.

Contents

Nomenclature

Acronyms & Abbreviations

2BP	Two Body Problem
3BP	Three Body Problem
AOC	Attitude-Orbit Coupling
AU	Astronomical Unit
Bar	Barycenter
CTVG	Constrained Velocity Yerminal Guidance
D	Didymos system
D1	Didymos main
D2	Dimorphos
DART Group	Deep-Space Astrodynamics Research & Technologies Group
DART s/c	Double Asteroid Redirection Test spacecraft
DnL	Descend and Landing
DOF	Degrees of Freedom
DS	Deep space
EOM	Equations of motion
FTVG	Free Velocity Terminal Guidance
GNC	Guidance, Navigation and Control
GnC	Guidance and Control
HLR	High Level Requirements
LHS	Left Hand Side
MSSG	Multiple Sliding Guidance
N-O	Non-optimal
NEA	Near Earth Asteroids
NSG	Non-spherical gravity
O	Optimal
PNG	Proportional Navigation Guidance
RF	Reference Frame
RHS	Right Hand Side
s.t.	Such that
s/c	Spacecraft
SMC	Sliding Mode Control
SRP	Solar Radiation Pressure
std.	Standard
ToF	Time of flight
TPD	Two Phased Descent
wrt.	With respect to
YORP	Yarkovsky–O'Keefe–Radzievskii–Paddack

Physical Quantities

σ	Standard deviation
I	Identity Matrix
J	Inertia tensor
n	Time of flight until reaching time: $t_r = t_0 + n(t_f - t_0)$
t_f	Final time / landing time
t_r	Reaching time
t_{go}	Time-to-go $t_{go} = t_f - t$
d	Days
y	Years
ZEM	Zero-effort-miss
ZEV	Zero-effort-velocity

Chapter 1

Introduction

Motivation

The exploration of Solar System's bodies is an endeavour that has held on to the public's and scientific curiosity for decades. In particular, the study of asteroids has seen a reinvigorated scientific interest in the last years, stemming from the combination of three issues: First, because asteroids can remain relatively unaltered since their formation, they can become information vaults containing data regarding the creation of the Solar System [1]. It is on this topic where the sample return (SR) missions prime, which recover asteroid material to bring it back to Earth and allow the possibility to study it on laboratory. Examples of SR missions include NASA's Stardust, launched in 1999 and first to collect and return cosmic samples of Wild-2 comet [2]; JAXA's Hayabusa2 [3], that rendezvoused with asteroid Ryugu and returned samples on 2020; and NASA's OSIRIS-REx mission, that rendezvoused with asteroid Bennu on 2020 and is bound to return the regolith samples on 2023 [4]. Second, in recent years has awakened the regard of asteroids as sources of ore mining, which might offer economic returns in the long-term future or fuel-extraction possibilities for interplanetary missions [5]. Third, asteroids present a risk to human life in case of a collision with Earth [6]. This fact has stirred a political and engineering interest to investigate ways to assess the risk of asteroid impacts on Earth and to devise strategies to avoid them.

In this framework was created the Asteroid Impact & Deflection Assessment (AIDA) collaboration, a joint effort of NASA and ESA to demonstrate the effect of the kinetic impact on an asteroid trajectory. Such asteroid is the Didymos Binary system, a binary asteroid consisting of Didymos (main body) and Dimorphos (moon). Didymos is a $\approx 780m$ wide, approximately spherical body that has an orbital period of $2.11y$. Orbiting it is its moon Dimorphos, a small ellipsoid with a mean diameter of $\approx 160\ m$ and an orbiting period of $\approx 12h$ [7]. The system is eclipsing binary, and thus Dimorphos' orbit can be measured from optical Earth-based observations [8].

The AIDA collaboration encompasses a pair of missions (see Fig. 1.1). On one hand, NASA's Double Asteroid Redirection Test (DART) mission will send a spacecraft (homonymously referred as DART) on 2024 that will collide with Dimorphos on 2026. The orbit deflection will then be evaluated by ground-based observations [8]. On the other hand is Hera mission, with which ESA plans to send the Hera s/c to rendezvous with the Didymos Binary system on 2027 [9]. Hera's objective is to study the asteroids and DART's impact aftermath in order to boost the overall knowledge return of AIDA. To further enhance the scientific return, Hera spacecraft (s/c) will carry two 6U CubeSats, called Juventas and Milani. In particular, the Milani CubeSat (at the time of writing just finished Phase-B of development) will perform optical observations with ASPECT Hyperspectral camera [10] to characterize the asteroids and evaluate DART's impact. After its scientific goals have been achieved it is expected to attempt landing on Dimorphos.

Asteroid missions that are to touch-down on an asteroid's surface typically adopt either of two approaches: a) A touch-and-go approach, in which the s/c descends, touches the surface and immediately pushes back to orbit. This is the approach used by the asteroid SR missions up to date. b) A landing approach, in the s/c is bound to descend to the surface and remain thereafter. An example of this is the Rosetta s/c, that left its lander module Philae to ballistically descent upon Churyumov–Gerasimenko comet's surface [11].

The descent is one of the most critical phases of an asteroid mission, since a failure at this stage is likely to compromise the spacecraft's integrity. Having a suitable Guidance, Navigation and Control (GNC) system is key to properly control the s/c with the limited information on its states. The design of such system is challenging due to asteroids' particular characteristics [11]: The weak gravity field implies a dynamical environment ruled by relatively large perturbations (e.g. the Solar Radiation Pressure, and the Attitude-Orbit Coupling [12]), which usually rend the dynamical models less accurate, and degrade the performance of the GNC system with respect to the expected one. Also, usually the asteroid and dynamical environment is relatively poorly characterised, because data typically exists only from Earth-based observations. This entails the necessity to produce algorithms that can accommodate range-varying parameters. Hence, for asteroid landing missions is fundamental to equip the s/c with a dedicated GNC system.

The engineering job of designing a suitable GNC scheme to land on an asteroid meets the opportunity that the Milani CubeSat's mission offers. As mentioned, during its experimental phase Milani is bound to land on Dimorphos, yet how will it happen remains unanswered.

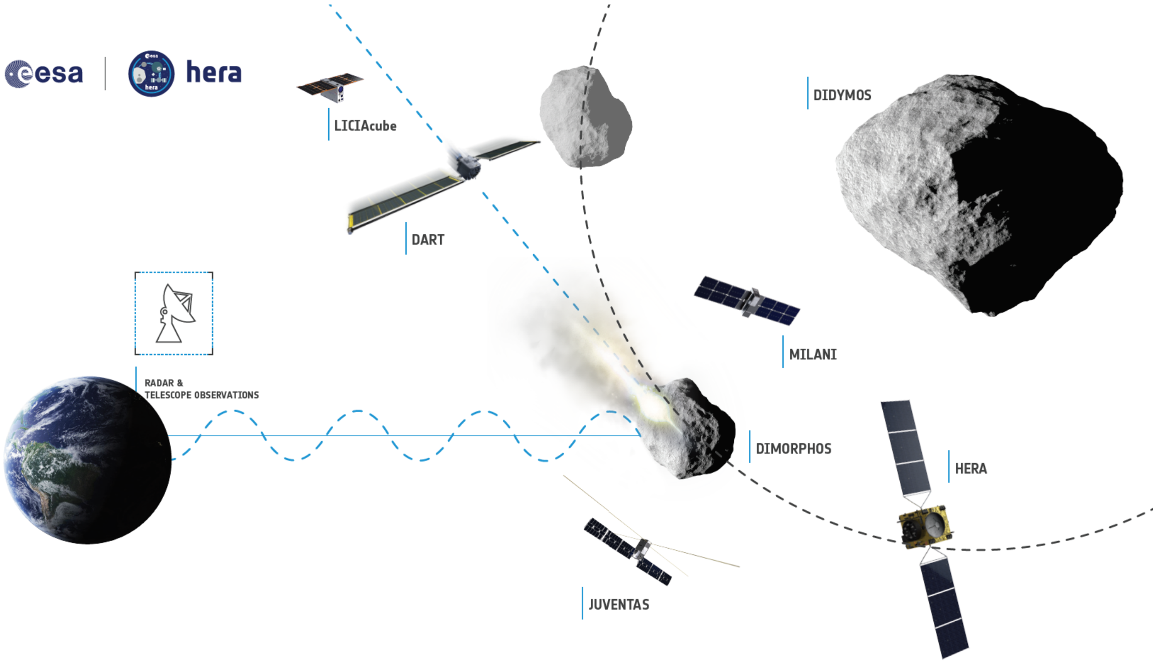


Figure 1.1: Asteroid Impact & Deflection Assessment (AIDA) collaboration. Credit: ESA Science Office, available at: `https://www.esa.int/ESA_Multimedia/Images/2019/01/Asteroid_Impact_Deflection_Assessment_AIDA_collaboration`.

Objectives

In this document I present a study aimed to answer the research question:

What is a good way to soft land the Milani CubeSat on Dimorphos?

To achieve this, the technical objective is to devise a control strategy that brings Milani onto Dimorphos surface with a velocity relative to the asteroid-attached reference frame that is less than $4.5cm/s$, which is a lower limit for the escape velocity from Dimorphos' surface [13].

In this work I develop a Guidance & Control scheme for Milani CubeSat's landing on Dimorphos, including the choice of the control algorithm and firing strategy. To this aim, I have reviewed the existing Guidance & Control algorithms for asteroid Landing & Descent, and performed a trade-off based on key requirements. It has been assumed that the GNC system is capable of closed-loop navigation. At the time of writing, the landing of Milani CubeSat is not critical to mission success and the GNC system has not been designed to optimize for descent and landing requirements. In particular, Milani is expected to perform only discrete ($300s$) firings. Nonetheless, the requirements of the landing phase (also known as "experimental phase") are still under consideration and might change. Hence it cannot be excluded that continuous thrust would be finally adopted. For this reason, in the choice of the algorithm, I first assume that continuous thrust is available. I propose to use the Multiple Sliding Surface Guidance (MSSG) algorithm for its robustness in front of dynamical uncertainty, simplicity, and relatively low impact on CPU resources.

Next, with the aim to minimize the impact of the proposed MSSG on the overall current GNC design, I have developed a firing strategy that uses $300s$ firings for $\approx 96\%$ of the approximation towards the asteroid. To ensure a successful and robust landing, during the last $15m$, the control has been allowed to remain open, if needed, for more than $300s$. I call such firing strategy the Two Phased Descent (TPD) strategy. The TPD requires a position knowledge with maximum error whose Gaussian std. deviation is of $\mathcal{O}(\sigma_r) = 1m$ to avoid target a point far below or far above Dimorphos' surface. The performance of the TPD has been evaluated with a Monte Carlo analysis, which show a very good performance (collision velocity is very likely to be below $4.5cm/s$) and that the landing ellipse is mainly affected by the navigation knowledge. The fuel consumption is about $10g$.

The document is organised as follows: Chapter 2 briefly introduces the reader to Didymos, its dynamical environment and the dynamical challenges that the GNC system must face. Chapter 3 consists of a review of existing GNC algorithms for Landing & Descent, as well as the choice of MSSG among them. Chapter 4 presents an in-depth study of the MSSG, which includes its derivation and a parametric analysis. In Chapter 5 I present the Two Phase Descent strategy and argue the choice of the control parameters. Finally, in Chapter 6 is the Monte Carlo analysis that evaluates of the Two Phase Descent strategy's performance.

Chapter 2

65803 Didymos

The aim of this chapter is to briefly introduce the 65803 Didymos binary system, its dynamical environment, and physical properties.

2.1 Asteroid Fundamentals

2.1.1 Asteroids across the Solar System

Asteroids are small, rocky bodies that orbit the Sun. In literature differ from comets in that asteroids do not have coma or outgassing, yet during recent years this boundary has become blurry and is under debate [1]. The dynamics of asteroids are mainly ruled by the Solar gravitational attraction, and it is a commonly good assumption that they follow a two-body problem with the Sun. The other forces to which they are subject are other planetary pulls (particularly Jupiter's) and forces derived from the Solar radiation. These stem into two main effects:

- The Yarkovsky effect: Solar radiation heats up one side of the asteroid, which is cooled down after the rotation of the asteroid puts that side facing deep space (DS). This temperature gradient along the surface yields a difference in the radiation that the asteroid emits for being at a given temperature, which then transmits a residual linear momentum along one direction. Even if the Yarkovsky force is small, it is the dominant perturbation on asteroids [1] and over the course of millions of years, the Yarkovsky effect can drive asteroids from the Main Belt towards the inner Solar System.

- The Yarkovsky–O'Keefe–Radzievskii–Paddack (YORP) effect: the rotation of an asteroid is changed due to the Solar Radiation Pressure (SRP) upon its asymmetrical surface. Changes in the rotation state can influence the orbital drift generated by the Yarkovsky effect.

Other effects of the solar radiation include:

- The drift due to impinging photons (not to be confused with YORP effect, which refers only to its effect on rotation).
- Recoil momentum of specularly reflected radiation.
- Recoil momentum of diffusely reflected radiation.
- Momentum of emitted radiation due to the asteroid's surface temperature (not to be confused with Yarkovsky's effect, that refers to the effect of having a temperature difference due to Sun heating the asteroid's surface).

Asteroids are present all throughout the Solar System, yet with varying distribution. One of the areas that show greatest concentration is the Main Asteroid Belt, a broad toroidal region of the Solar System located between the Earth and Mars. Other regions include Jupiter's L4 and L5 Lagrangian points, whose asteroids are classically named "Trojans". Fig. 2.1 shows the asteroid distribution across the Solar System. Near Earth Asteroids (NEAs) are instead those Asteroids that orbit the Sun closer to Earth. The Near Earth Object Program Office (NASA/JPL) classifies them as having a perihelion $q < 1.3AU$ [1].

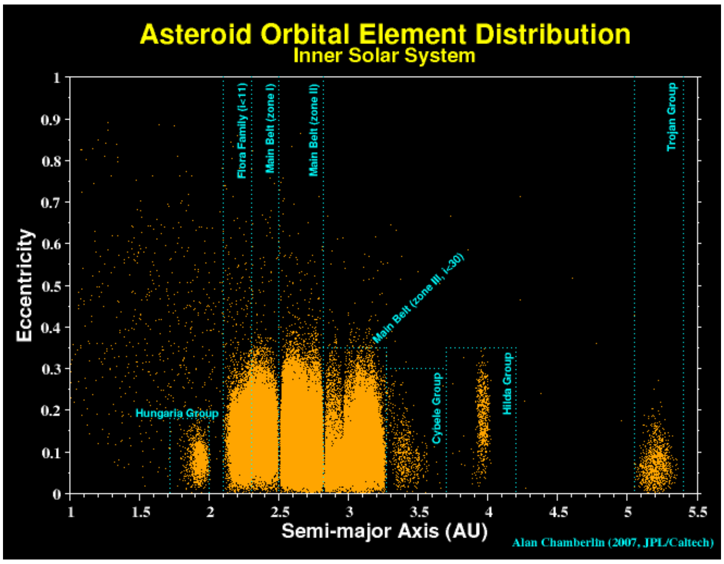


Figure 2.1: Asteroid Orbital Element Distribution. Credit: Alan Chamberlin (2007, JPL/Caltech), available at: `https://ssd.jpl.nasa.gov/?dist_ae_ast`.

Binary asteroids are asteroid systems composed by two bodies that orbit each other. Both bodies can be arranged in different ways and have different mass proportions. E.g. Castalia is a contact binary asteroid, where the two bodies touch and have similar size. Binary asteroids are very frequent around the Solar System [1]. Very frequent are those binary asteroids composed by a big, primary attractor, and a secondary (or moon) that orbits the primary. An example is asteroid 243 Ida with its moon Dactyl, that was photographed by Galileo on 1993 (see Fig. 2.2).

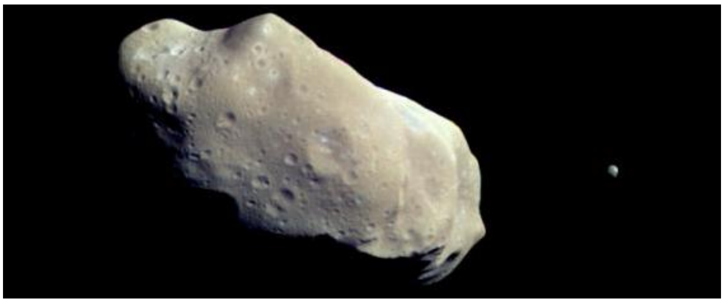


Figure 2.2: PIA00069: Ida and Dactyl in Enhanced Color. Credit: NASA/JPL, available at: `https://photojournal.jpl.nasa.gov/catalog/PIA00069`.

2.1.2 Small Bodies' Dynamical Environment

The dynamical environment around small bodies differs from that of major bodies in that the attractor's pull is not as prominent over other dynamical perturbations. Indeed, near a planet like Earth and far from the atmosphere, the motion of a spacecraft (s/c) is almost only influenced by Earth's gravity. Instead, for example, about small bodies the gravity pull can be of the same order as the SRP. This drives the necessity to include attitude information in the dynamical model. To do so has been used the cuboid model, in which the s/c is consists of a set of planar faces with some light emission coefficients. The model is based on that of Fig. 2.3.

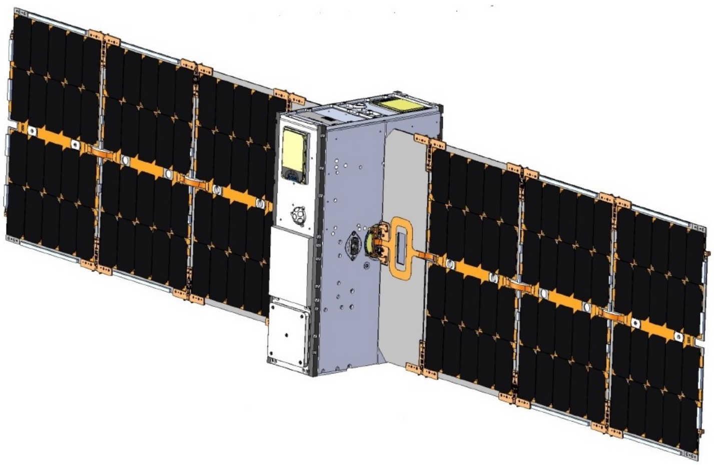


Figure 2.3: Model of the Milani CubeSat at the time of writing. Credit: Tyvak International, available at: `https://www.esa.int/ESA_Multimedia/Images/2021/06/Milani_CubeSat`.

A relevant phenomenon is what can be called Attitude-Orbit Coupling (AOC). It consists in the following: the gravity pull's magnitude on an asymmetric s/c depends on its attitude, since an appendix farther from the attractor will be attracted less and the inverse. Thus, the attitude influences the gravity vector, which then rules the translational motion. Hence, the translational motion (as well as the gravity pull on the s/c) induces changes in the attitude of the s/c, which in its turn affects the gravity pull's magnitude and so on. This effect depends on a small parameter

$$\epsilon = \frac{\rho}{r_0} \tag{2.1}$$

where ρ is a characteristic dimension of the s/c and r_0 the orbital radius [14]. Bigger s/c will have greater gradients of pull along their bodies. Smaller orbits will mean stronger local changes in gravity due to changes in position. About bigger planets (and hence larger orbits) the term ϵ is negligible, but not about small bodies, where ϵ can take values of the order of 0.01. Previous works show how the AOC can have a significant impact on the stability of self-stabilizing terminator orbits around asteroids and thus it is important to model this effect in order to evaluate the orbit's performance [12]. In this work it has been assumed that the attitude of Milani CubeSat is given because it is critical to the scientific return and optical navigation. Then, the translational motion is informed by the attitude. The full and restricted 6DOF model is derived in Appendix B.

2.2 Asteroid 65803 Didymos

The binary system 65803 Didymos (also "Didymos system") is a NEA that orbits the Sun in a region between the Earth and Mars (see Fig. 2.4). Didymos system's primary, referred as Didymos (also "D1"), is approximately a spherical body with diameter of $\approx 780m$ [7]. In this work D1 will be modelled as a sphere. The secondary (also "D2") is assumed to be a triaxial ellipsoid with radii $(\alpha, \beta, \gamma) = (104, 80, 66)\ m$. A representation of the Didymos system in MATLAB can be seen in Figs. 2.5 and 2.6. The

Table 2.1: Didymos system principal properties. Extracted from [7].

Property	Value
D1 Diameter	$0.780km \pm 10\%$
D2 Diameter	$0.163 \pm 0.018km$
System mass	$(5.278 \pm 0.04) \cdot 10^{11}kg$
Component bulk density	$2100kg\ m^{-3} \pm 30\%$
D1 spin period	$2.2600 \pm 0.0001h$
Component separation	$1.18 + 0.04/-0.02km$
D2 orbital period	$11.920 + 0.004/-0.006h$

data on asteroids' positions, trajectories and geometry has been retrieved from the up-to-date kernels of the Hera mission study[1].

Didymos system's basic properties are shown in Table 2.1. Importantly, D2 is assumed to be tidally locked with D1 [7], hence the biggest axis of the ellipsoid is pointing always along the vector D2-D1 [1]. Also, the spin of D1, D2 and the translation of D2 around D1 are opposite to the motion of D1 around the Sun. This is, the equatorial planes of D1 and D2 are nearly flipped upside down with respect to the ecliptic. Throughout this work will be used reference frames that are defined in terms of the asteroids' position, particularly the D2 Body-Attached Reference Frame (D2 B RF or D2 RF) shown in Fig. 2.6. Didymos is drawn in the N RF, which is aligned along Ecliptic J2000 RF and centered in Didymos' barycenter (see Table B.1 in Appendix B).

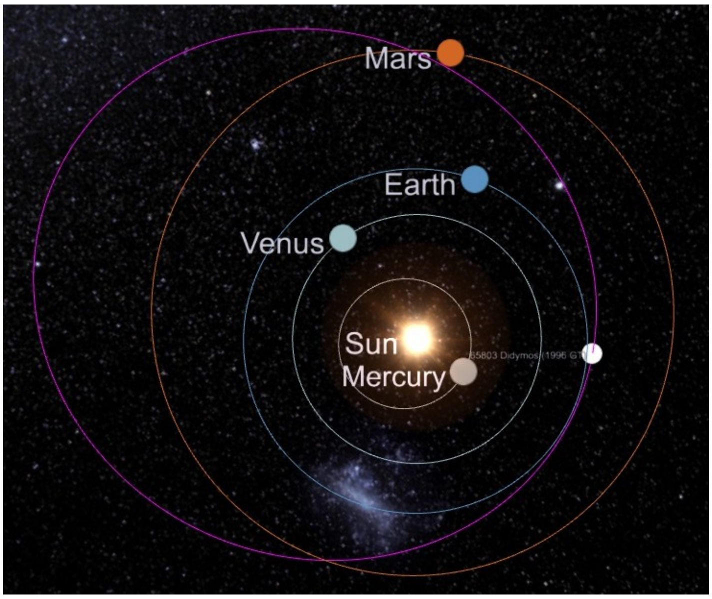


Figure 2.4: Inner Solar System at 25-Jan-2027 (UTC). Didymos system is marked in white and its orbit in violet. Credit: `TheSkyLive.com` 3d Solar System Viewer.

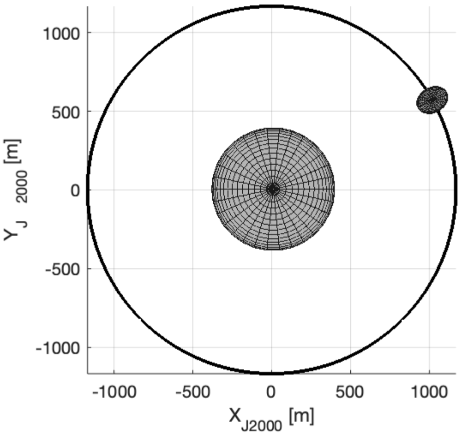


Figure 2.5: Top view of the Didymos system represented in MATLAB in the N RF.

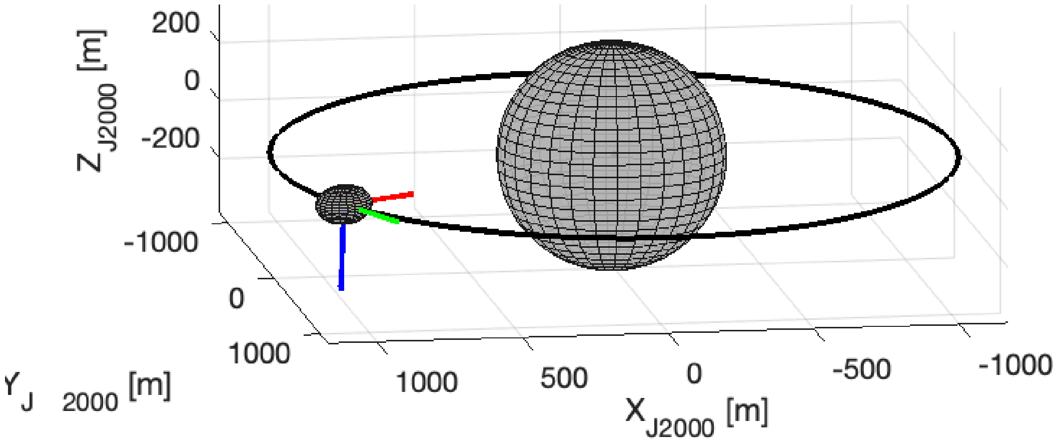


Figure 2.6: Side view of the Didymos system represented in MATLAB in the N RF. The red-green-blue vectors represent the D2 RF (x_{D2}, y_{D2}, z_{D2} respectively).

[1] ESA SPICE Service, Hera Working Groups, Instrument Teams. *Hera Spice Kernel Set.* 2020. Last Accessed: aug-2021. [online]. Available: `https://www.cosmos.esa.int/web/spice/spice-for-hera`

Chapter 3

Guidance & Control Algorithm Selection for Dimorphos Landing

The goal of this chapter is to choose and present a Guidance and Control (GnC) algorithm suitable for the soft landing of Milani CubeSat on Dimorphos. To this aim, first a comprehensive review of the state-of-the art GnC algorithms is performed; followed by an appropriate trade-off and selection of the algorithm. The navigation subsystem falls outside of the scope of this work, and it will be simplified to a stochastic knowledge of the states. For completeness sake, a review of current Navigation strategies can be found in Appendix A.

To design a GNC system for a small body descent is particularly challenging within GNC engineering due to three specific characteristics of the asteroid Descent & Landing (DnL) problem [11]. First, usually there is only limited prior information on important parameters that will be key on defining the s/c environment (e.g. asteroid geometry or its surface properties). This void of information imposes a big flexibility requirement on the GNC system, that needs to accommodate possibly widely ranging parameters. Secondly, the dynamical environment is particularly complex: an accurate gravity field is usually hard to model due to loosely known parameters of the asteroid, i.e. its rotational state or density distribution. This, alongside a strongly perturbed environment (e.g. the AOC, Solar Radiation or third-body perturbations), usually introduces big modelling errors, which the GNC must compensate or be very robust to. Third and finally, close-body operations entail a need of quickly reacting to information on the s/c state, which is made very difficult to implement by the ground segment due to the usual communication time delay. Such problem must be tackled by as much on-board autonomy as possible, both in Navigation and Guidance & Control.

The objective of a s/c Guidance and Control system during its descent upon an asteroid is to generate an acceleration command profile $\mathbf{a}(t)$ that generates a trajectory which satisfies a number of constraints. Such constraints can be:

- *Terminal constraints:* constraints on start/final states of the s/c trajectory or acceleration command. Examples are boundaries on the touch-down velocity or impact direction.
- *Path constraints:* constraints that are to be complied with during the trajectory itself. Examples are limitations on thrust or a given shape of the command function.

There are several schemes to generate an appropriate $\mathbf{a}(t)$, and a clear classification is of the uttermost importance to be able to choose among them. Different approaches to classify the current Guidance & Control (GnC) strategies are available. In Ref. [11] is proposed to depart from whether the GnC scheme requires information on the state error with respect to a reference trajectory (called *trajectory-tracking*) or not (called *trajectory-free*). On the other hand, the review by Simplício et al. [15] takes a more comprehensive approach. This section is mainly based on Simplício's et al. work and will follow their approach, to later merge it with contributions from Ref. [11] and other sources, cited accordingly. Finally, a diverse paradigm coming from other works is commented. The most fundamental classification between schemes is whether they are *closed-loop* or *open-loop*:

- *Open-loop*: a reference trajectory $\{\mathbf{r}(t)_{ref}, \mathbf{v}(t)_{ref}\}$ and its corresponding acceleration command profile $\mathbf{a}(t)_{ref}$ are computed before the operation. During the operation, the command profile is not re-computed by the Guidance CPU. Using measurements, it may be corrected by an appropriate compensator. These open-loop schemes have emanated from planetary landing trajectory design, and are based on imposing an acceleration function (e.g. $\mathbf{a}(t) = \mathbf{C}_0 + \mathbf{C}_1 t + ...\mathbf{C}_N t^N$) whose coefficients will implement both terminal and path constraints. The computation of these coefficients may require a relatively large CPU power which traditionally was only available on ground, and therefore constricted this approach to open-loop schemes. Recent advances in numerical performance have enabled the such computations to be performed on-board in a closed-loop fashion. This new paradigm is called "Computational GnC".
- *Closed-loop*: during the operation, measurements are used to compute a reference acceleration command $\mathbf{a}(t)_{ref}$, which in addition may be corrected by a compensator. They emanate from the Control Theory field of Optimal Feedback Control, a problem which is solved via the Pontryagin Maximum Principle or Calculus of Variations. These Traditional Closed-Loop schemes, take into account terminal constraints alone, and their operating principle is to construct an acceleration command from the difference between the desired states (e.g. $\mathbf{r}_f$ and $\mathbf{v}_f$) and the current ones. Recently, they have been augmented using Slide Mode Control (SMC) theory to provide robustness in presence of inaccurate measurements and unmodeled dynamics. Path constraints can be taken into account indirectly, as will be mentioned later.

Within both categories is available a palette of algorithms. Fig. 3.1 displays a classification map. Regarding the classification based on *trajectory-tracking* vs. *trajectory-free* proposed by [11], open-loop schemes (and therefore Computational GnC as well) would fall under the *trajectory-tracking* label, because they inevitably compute a full path which the control tracks. On the other hand, since closed-loop schemes work with the error between two states, the desired state may or may not be a reference trajectory (e.g. a landing target), and therefore they do not necessarily need a trajectory to track. Below are highlighted the characteristics of each algorithm of Fig. 3.1.

3.1 Mathematical Foundation

Before describing state-of-practice GnC algorithms, it is convenient to introduce mathematical nomenclature and physical concepts. The problem is traditionally cast as point-mass Full 3-Body-Problem, and the EOM are expressed in a Sun-Centered, Inertially-fixed RF (N RF). The equations of motion (EOM) of the Target (T) and s/c (S) with respect to the N RF (see Table B.1 in Appendix B) and the relative EOM are, respectively:

$$\begin{cases} \dot{\mathbf{r}}_T = \mathbf{v}_T \\ \dot{\mathbf{v}}_T = \mathbf{g}_T(\mathbf{r}_T) \end{cases} \tag{3.1}$$

$$\begin{cases} \dot{\mathbf{r}}_S = \mathbf{v}_S \\ \dot{\mathbf{v}}_S = \mathbf{g}_S(\mathbf{r}_T, \mathbf{r}_S) + \mathbf{a} + \mathbf{p} \end{cases} \tag{3.2}$$

$$\begin{cases} \mathbf{r} \triangleq \mathbf{r}_S - \mathbf{r}_T \\ \dot{\mathbf{r}} = \mathbf{v} \\ \dot{\mathbf{v}} = \mathbf{g}(\mathbf{r}_T, \mathbf{r}) + \mathbf{a} + \mathbf{p} \end{cases} \tag{3.3}$$

where $\mathbf{g}_T$ contains the gravity terms acting on the Target due to the sun and g_S contains the gravity terms acting on the s/c due to the Target and the Sun. $\mathbf{g} = \mathbf{g}_S - \mathbf{g}_T$. The term $\mathbf{p}$

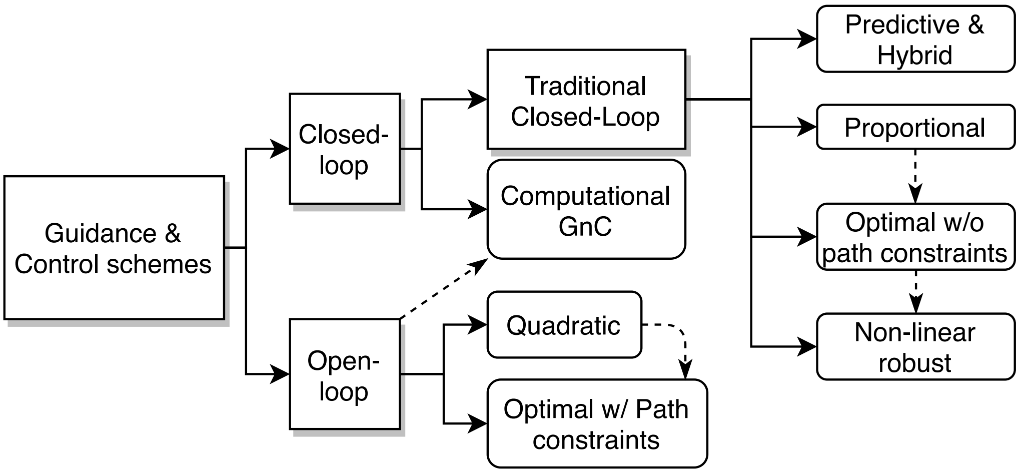


Figure 3.1: General classification of state-of-the art GnC schemes. Squared, shadowed boxes refer to categories; rounded boxes refer to algorithms. The abbreviations w/ and w/o refer to "with" and "without" respectively. Dashed arrows go from an algorithm towards its conceptual augmentation.

contains all the unmodeled dynamics and uncertainties and will hereafter be ignored. **a** is the acceleration command and is defined as

$$\mathbf{a}(t) = \frac{\mathbf{T}(t)}{m(t)} \tag{3.4}$$

For a given trajectory that goes from t_0 to t_f, we define:

$$\text{Time-to-Go:} \qquad t_{go}(t) \triangleq t_f - t \tag{3.5}$$

$$\text{Line-of-Sight:} \qquad \Lambda \triangleq \mathbf{r}(t)/|\mathbf{r}(t)| \equiv \hat{\mathbf{r}}(t) \tag{3.6}$$

$$\text{Closing velocity:} \qquad V_c(t) \triangleq -|\mathbf{v}(t)| \tag{3.7}$$

Another two important quantities, which are states that define the system, are:

- Zero-Effort-Miss: position error at t_f if no correction is applied after t (see Eq. (3.8)).
- Zero-Effort-Velocity: velocity error at t_f if no correction is applied after t. (see Eq. (3.9)).

$$\mathbf{ZEM}(t) \triangleq \mathbf{r}_f - \mathbf{r}(t_f)|_{\mathbf{a}(\tau)=0} = \mathbf{r}_f - \left[\mathbf{r}(t) + (t_f - t)\mathbf{v}(t) + \int_t^{t_f}\int_t^{\tau} \mathbf{g}(\tau')d\tau' d\tau\right] \tag{3.8}$$

$$\mathbf{ZEV}(t) \triangleq \mathbf{v}_f - \mathbf{v}(t_f)|_{\mathbf{a}(\tau)=0} = \mathbf{r}_f - \left[\mathbf{v}(t) + \int_t^{t_f} \mathbf{g}(\tau)d\tau\right] \tag{3.9}$$

Note that if the final states $\mathbf{r}_f$ and $\mathbf{v}_f$ are truly reached due to the effect of a command **a**, then

$$\mathbf{ZEM}(t) \equiv \int_t^{t_f}\int_t^{\tau} \mathbf{a}(\tau')d\tau' d\tau \Rightarrow \dot{\mathbf{ZEM}}(t) = \frac{d}{dt}\left(-\int_{t_f}^{t}(t-\tau)\mathbf{a}(\tau)d\tau\right) = -t_{go}\mathbf{a}(t) \tag{3.10}$$

$$\mathbf{ZEV}(t) \equiv \int_t^{t_f} \mathbf{a}(\tau)d\tau \Rightarrow \dot{\mathbf{ZEV}}(t) = \frac{d}{dt}\left(-\int_{t_f}^{t}\mathbf{a}(\tau)d\tau\right) = -\mathbf{a}(t) \tag{3.11}$$

3.2 Open-Loop schemes

Open-loop schemes (or alternatively trajectory-tracking) were originally developed for Planetary Landing, from which two scheme families have evolved, the Quadratic Scheme and its augmentation, which is Optimal with Path Constraints.

3.2.1 Quadratic

Used for the first time for Apollo's Moon landing, imposes the acceleration command function to have a linear and quadratic dependence on time, and lets the terminal conditions on $\mathbf{r}(t_f), (t_f), \dot{\mathbf{v}}(t_f)$ determine the coefficients of such function:

$$\mathbf{a}(t) = \mathbf{C}_0 + \mathbf{C}_1 t + \mathbf{C}_2 t^2 \tag{3.12}$$

The integration of the acceleration yields the velocity and position profiles, dependent on the coefficients. If the states are evaluated at t_f, a linear system that relates the coefficients with the terminal states is achieved (Eq. (3.13)), from which the terminal states are extracted.

$$\begin{cases} \mathbf{v}(t) = \mathbf{v}(t_0) + \int_{t_0}^{t} (\mathbf{a}(\tau, \mathbf{C}_i) + \mathbf{g}) d\tau \\ \mathbf{r}(t) = \mathbf{r}(t_0) + \int_{t_0}^{t} \mathbf{v}(\tau, \mathbf{C}_i) d\tau \end{cases} \qquad \begin{bmatrix} & & \\ & f(t_f) & \\ & & \end{bmatrix} \begin{Bmatrix} \mathbf{C}_0 + \mathbf{g} \\ \mathbf{C}_1 \\ \mathbf{C}_2 \end{Bmatrix} = \begin{bmatrix} \dot{\mathbf{v}}(t_f) \\ \mathbf{v}_f - \mathbf{v}_0 \\ \mathbf{r}_f - \mathbf{r}_0 - \mathbf{v}_0 t_f \end{bmatrix} \tag{3.13}$$

Note that the time-of-flight (or t_f if initial time is taken at 0) is a free variable and needs to be selected. This scheme is simplistic, doesn't enforce fuel-optimal constraints and imposes a certain polynomial shape on the acceleration profile, and thus the trajectory.

3.2.2 Optimal with Path Constraints

This scheme is an augmentation of the *Quadratic* scheme, based on the idea that selecting a function for $\mathbf{a}(t)$ with more coefficients will allow for enforcement of both terminal and path constraints; in particular fuel-optimality. The acceleration is therefore defined as:

$$\mathbf{a}(t) = \mathbf{C}_0 + \mathbf{C}_1 t + ...\mathbf{C}_N t^N \tag{3.14}$$

And the linear system of Eq. (3.13) alongside with other equations bounding the coefficients $\mathbf{C}_i$ determine the acceleration profile. There is a great number of solvers to solve such problem, which also allow to use a different profile than Eq. (3.14). In this scheme, the t_f is again a free parameter, but path constraints are enforced. In general, open-loop schemes are simple and do not require measurements, but manage poor accuracy and do not yield strategies that are suitable for complex gravitational environments [15].

3.3 Traditional Closed-Loop

Within Traditional Closed-Loop schemes are those that take navigation information to compute acceleration commands in real time. The flight heritage of them is by far the greatest among the schemes mentioned in this work.

3.3.1 Proportional

Proportional schemes have been typically used for the missile-interception problem, and the algorithm is labeled as Proportional Navigation Guidance (PNG). The operational principle is to apply acceleration commands perpendicular to the line-of-sight Λ. The performance improves deducting the gravity component perpendicular to Λ, which yields the Augmented Proportional Navigation Guidance:

$$PNG: \quad \mathbf{a}(t) \triangleq nV_c(t)\dot{\Lambda}(t); \quad n \in [3, 5] \tag{3.15}$$

$$APGN: \quad \mathbf{a}(t) \triangleq nV_c(t)\dot{\Lambda}(t) - \frac{n}{2}\mathbf{g}_{\perp}(t) \equiv \frac{n}{t_{go}^2(t)}\mathbf{ZEM}(t); \quad n \in [3, 5] \tag{3.16}$$

where a greater gain n yields more robustness, greater accelerations, but more fuel consumption. Note that PNG can be written as a proportional gain of *ZEM*. The time-of-flight is undetermined. The angle of the impact Λ with the surface can be bounded, a method for non-continuous thrust has been developed. The reader is referred to [15] and the references therein.

3.3.2 Predictive & Hybrid

The predictive algorithms function based on computing the difference between the predicted impact relative velocity direction and the current one. They cast the problem as Δv firings over a span Δt_p, and the acceleration command is defined as in Eq. (3.17), where T is the Set of times when the firing is allowed to occur. The predictive methods can be separated in Predictive-Impulsive (PI), and Kinematic-Impulsive (KI):

$$\mathbf{a}(t) = \begin{cases} \frac{\Delta \mathbf{v}(t)}{\Delta t_p}, & for \quad t \in T \\ \mathbf{0}, & for \quad t \notin T \end{cases} \tag{3.17}$$

- *Predictive-Impulsive*: The current states $\mathbf{r}(t)$ and $v(t)$ are known, and the line-of-sight at t_f ($r^p(t_f)$) is predicted using the state propagation matrix

  Finally, the $\mathbf{\Delta v}$ command is defined as:

$$\Delta\mathbf{v}(t) \triangleq V_c(t)\hat{\mathbf{r}}^p(t_f) - \mathbf{v}(t) \tag{3.18}$$

- *Kinetic-Impulsive*: If the estimates of the current states $r(t)$ and $v(t)$ are not known, they need to be estimated using optical measurements on $V_c(t)$ and $\Lambda(t)$, for then to initialize the process to obtain $\mathbf{a}(t)$ for the PI algorithm.

Therefore, KI is just a conceptual augmentation of PI, and in reality the one in practice, since the current states have to be measured somehow. Predictive algorithms are more fuel efficient than proportional ones, and earlier firings (i.e. T closer to t_0) reduce fuel consumption but degrade performance at impact. For this, *Hybrid algorithms* use PI/KI during first stages of flight and on mid-course they change to PNG to increase the final performance.

3.3.3 Optimal Without Path Constraints - Optimal Guidance Laws

The closed-loop algorithms that are fuel optimal and don't enforce path constrains are labelled "Optimal Guidance Laws". They emanate from the solution of the Feedback Control Problem, cast as the minimization of a the fuel cost function Eq. (3.19).

$$J(\mathbf{a}(t)) = \int_t^{t_f} L(\mathbf{x}(\tau), \mathbf{a}(\tau))d\tau = \int_t^{t_f} \frac{1}{2}\mathbf{a}(\tau)^T\mathbf{a}(\tau)d\tau \tag{3.19}$$

The solution to this problem can be approached by using a transformation of the equations of motion (EOM) using certain co-states, and can be shown that the acceleration profile that points at the extreme of Eq. (3.19) is of the shape:

$$\mathbf{a}(t) = \mathbf{p}_v(t_f) + \mathbf{p}_r(t_f)t_{go}(t) \tag{3.20}$$

where $p_v(t_f)$ and $p_r(t_f)$ are the aforementioned co-states, evaluated at t_f. Their value depends on the terminal constraints (e.g. $v(t_f) = v_f$) and on the time-of-flight through the value of t_f. Note that given a time-of-flight and the constraints, the acceleration profile for the trajectory will be linear with time. For a more detailed description the reader is referred to [15] and the references therein. Since in one hand the values of the terminal co-states will depend on the terminal conditions and the time-of-flight (or alternatively the time-to-go), but on the other so do the **ZEM/ZEV**, it can be shown how the acceleration laws can be written in terms of

ZEM/ZEV. Below are displayed those for the case of a constricted terminal velocity guidance (CTVG) and a free terminal velocity guidance (FTVG):

$$CTVG: \quad \mathbf{a}(t) = \frac{6}{t_{go}^2(t)}\mathbf{ZEM}(t) - \frac{2}{t_{go}(t)}\mathbf{ZEV}(t) \tag{3.21}$$

$$FTVG: \quad \mathbf{a}(t) = \frac{3}{t_{go}^2(t)}\mathbf{ZEM}(t) \tag{3.22}$$

The FTVG is also called ZEM/ZEV algorithm or ZEM/ZEV method. Note that:

- No path constraints have been imposed, in particular no control constraints such as the maximum magnitude of $\mathbf{a}(t)$ have been imposed. A list of ways to indirectly deal with such constraint can be found in [15].
- FTVG structure coincides with the algorithm used in PNG, meaning that with a proper selection of the n parameter the latter is fuel-optimal.
- The complexity of the gravity field modeling and integration $\int(\mathbf{g}(\mathbf{r}(\tau))d\tau$ is embedded in the computation of **ZEM** and **ZEV**. To perform accurately such integration is important in order to not degrade the performance of the control. If **ZEM/ZEV** is too heavy due to the time-dependent $\mathbf{g}$, an alternative is to compute segments of $\mathbf{a}(t)$ with the desired states being way points on a reference trajectory, while considering $\mathbf{g}$ constant within each segment.

Regarding the performance comparison between CTVG and FTVG, from the results a robustness-assessing simulation in [15] is shown how:

- CTVG spends more than FTVG due to the extra effort of controlling velocity magnitude and direction.
- Both CTVG and FTVG show a great dispersion facing uncertainties, with FTVG sensitivity being larger due to the usage of information on position error only.

3.3.4 Non-Linear Robust

Since the Optimal Guidance Laws strongly rely on the modeling of the problem's dynamics (which are introduced as the modeling of $\mathbf{g}$), the robustness of the algorithms on errors or uncertainties of such models is a figure of merit of the schemes. Recently, sliding mode control (SMC) theory has been used to generate algorithms that are robust to dynamical uncertainty. Two examples are the Optimal Sliding Guidance (OSG) algorithm, in which the CTVG (or ZEM/ZEV method) is augmented with a non-linear term; or the Multiple Sliding Surface (MSSG) algorithm, that uses two sliding surfaces to drive the s/c towards the desired states.

Optimal Sliding Guidance Algorithm

In the Optimal Sliding Guidance algorithm the ZEM/ZEV algorithm is augmented with a non-linear term that rends it robust to uncertainty [16]. The formulation stems from the definition of a sliding surface:

$$\mathbf{s}(t) = \mathbf{ZEM}(t) + \lambda(t)\mathbf{ZEV}(t) \Rightarrow \dot{\mathbf{s}}(t) = ... = -K(t)\mathbf{a}(t); \quad K(t) > 0 \tag{3.23}$$

The aim is to bring the system to $\mathbf{s}(t) = 0$, which is an equilibrium point since the **ZEM/ZEV** are zero and the acceleration command is zero. It can be shown how the usage of CTVG or FTVG results in $s \to 0$ when $t \to t_f$ [16], and the rate of convergence is $\propto e^{-\lambda(t)t}$ [15]. Then, to ensure the stability of the sliding surface dynamics, Lyapunov's Second Stability Theorem is used to search an $\mathbf{a}(t)$ that enforces:

$$V(\mathbf{s}) \triangleq \frac{1}{2}\mathbf{s}^T\mathbf{s} \qquad (3.24) \qquad\qquad \begin{cases} V(\mathbf{s}) > 0 & \forall \mathbf{s} > 0 \\ \dot{V}(\mathbf{s}) \leq 0 & \forall \mathbf{s} > 0 \end{cases} \qquad (3.25)$$

Complying acceleration commands can be as simple and robust as switching $\pm\mathbf{a}_{max}(sgn(\mathbf{s}))$ so that $\mathbf{s}(t)$ oscillates around 0. Eq. (3.25) is also satisfied with CTVG and FTVG augmented with a non-linear term, an example being:

$$\mathbf{a}(t) = \frac{6}{t_{go}^2(t)}\mathbf{ZEM}(t) - \frac{2}{t_{go}(t)}\mathbf{ZEV}(t) - \frac{\Phi}{t_{go}(t)} sign(\mathbf{s}(t)) \qquad (3.26)$$

Which will yield global stability if $\Phi \geq |\mathbf{p}_{max}| \quad \forall t$ [16], where $\mathbf{p}_{max}$ is the maximum perturbation or uncertainty. Eq. (3.26) is called the Optimal Sliding Guidance algorithm. Note:

- The non-linear term in Eq. (3.26) provides an additional contribution to the acceleration command, which in general will mean a higher control effort in exchange of increasing robustness.
- The increase in robustness also allows for a decrease on the number of tracked way points across a reference trajectory [15].
- The proposed non-linear term forces $s(t)$ to continuously chatter around $s(t) = 0$, which greatly degrades the performance with continuous firings around a nominal trajectory.
- In a numerical example in [15] is shown how the SMC augmentation reduces the sensitivity of the solution to uncertainties with respect to the optimal laws by a half.

In [15] is presented a parametrisation of the Traditional Closed-Loop, that share common structural properties. The guidance laws aforementioned can be generalised as:

$$\mathbf{a}(t) = [k_r \ k_v] \cdot \begin{bmatrix} \frac{\mathbf{ZEM}}{t_{go}^2} \\ \frac{\mathbf{ZEV}}{t_{go}} \end{bmatrix} - \phi \ \mathbf{h}(\mathbf{ZEM}, \mathbf{ZEV}, t_{go}) \qquad (3.27)$$

Where $\mathbf{h}$ is a non-linear function for the SMC augmentation and the parameters k_i weight the linear part of $\mathbf{a}(t)$. It has been shown above how fuel consumption optimization is brought by $k_r = 6$ and $k_v = -2$ but it can be shown how other configurations can yield more desirable results. In general appears a trade-off between the error in closing velocity and the spent Δv. References on how to systematically optimise such parameters can be found in [15]. Note that these laws become singular when $t \to t_f$, a situation that would yield bursts in acceleration commands. The singularity can be avoided by switching off the control system before impact.

Multiple Sliding Guidance Algorithm

The OSG method described in the section above strongly relies on the propagation of the uncontrolled dynamics in order to compute the ZEM/ZEV quantities. In a simple gravitational environment (e.g. $g(t) \approx ct.$), OSG may be effective, since an adequate change in the initial states that generate the propagated uncontrolled orbit may properly change the final states of the same. Nonetheless, in a highly changing gravity environment it can be advantageous to work with an algorithm that has more control authority and is still robust to uncertainty.

The Multiple Sliding Surface Guidance (MSSG) algorithm has been developed in the recent years [17] and spans from the ideas of SMC to achieve, just as with the OSG, a control that is robust to uncertainty. In the MSSG, the objective is to make the system reach a Sliding Phase in which the current difference between target and actual states will decrease in finite time. Then, the control is chosen such that the system converges to that Sliding Manifold in finite time and stays there in the face of perturbations. First, a sliding variable that contains the state errors is defined:

$$\mathbf{s}_1 \triangleq \mathbf{r}(t) - \mathbf{r}_d(t) \tag{3.28}$$

$$\partial(\mathbf{s}_1)/\partial t = \dot{\mathbf{s}}_1 = \nu(t) - \nu_d(t) \tag{3.29}$$

where ν is the velocity of the s/c with respect to an asteroid-attached RF and x_d is the desired state x. $\mathbf{r}$ is the position of the s/c with respect asteroid-attached RF. The goal is to put the system on a Sliding Manifold defined by the sliding variable $\mathbf{s}_2$:

$$\mathbf{s}_2 = \dot{\mathbf{s}}_1 + \frac{\Lambda}{t_{go}}\mathbf{s}_1, \qquad t_{go} = t_F - t \tag{3.30}$$

which, if $\mathbf{s}_2 \equiv 0$, will bring s_1 and $\dot{\mathbf{s}}_1$ to 0 at t_f. Using Lyapunov methods, it can be shown that a control that brings the system to $\mathbf{s}_2 = 0$ in finite time (FT) and makes the system stable around $\mathbf{s}_2 = 0$ is one such that:

$$\dot{\mathbf{s}}_2 = -\Phi sign(\mathbf{s}_2), \qquad \Phi > 0 \tag{3.31}$$

In such case, the system will reach $\mathbf{s}_2 = 0$ in t_r s.t. $\Phi = \frac{|\mathbf{s}_2(0)|}{\Delta t_r}$, and will yield global stability if $\Phi \geq |\mathbf{p}_{max}|$. Using the definition of $\mathbf{s}_2$, one can include the dynamics of the problem in Eq. (3.31), to obtain the expression for the control action $\mathbf{a}$:

$$\mathbf{a} = -\frac{\Lambda}{t_{go}}\dot{\mathbf{s}}_1 - \frac{\Lambda}{t_{go}^2}\mathbf{s}_1 - \Phi sign(\mathbf{s}_2) - \mathbf{g}_{NI}(t, \mathbf{r}) \tag{3.32}$$

where $\mathbf{g}_{NI}(\mathbf{r}, t) = \mathbf{g}(t, \mathbf{r}) - \mathbf{R}_a(t) - [2\omega \times \nu + \omega \times \omega \times \mathbf{r}]$; and $\mathbf{R}_a$ is the position of the asteroid-fixed RF center at every time. $\mathbf{g}(t, \mathbf{r})$ is the gravity force experienced by the s/c. The main difference wrt. the OSG algorithm is that the position and velocity errors are taken as the between of the *current* states and the target states. Also, the dynamics are cancelled at each instant. This method has been simulation-tested on a binary asteroid case [18], which is a good precedent of applicability for the current work. The fuel performance of the MSSG greatly depends on the choice of the algorithm parameters t_r and Λ, [17], [18] and developments have been made to tune the algorithm using Reinforcement Learning [19].

3.4 Computational Guidance & Control

Computational GnC is based on the idea to recurrently compute open-loop schemes during flight, each time with up-to-date navigation information. This allows to implement both path and terminal constraints. Such task is computationally heavy, but recent developments have provided a plethora of algorithms to efficiently tackle it are available. In [15] are mentioned two trends: Convex Optimisation and Pseudo Spectral Methods. In [11] are stated different examples in literature on how such problems have been achieved.

Regarding Convex Optimisation, both reviews highlight that one of the major problems is how to treat constraints that are not convex. In some cases, a process called *lossless convexification* (lc.) is performed, which transforms the constraints' variables so that the former become convex. If the constraints are hard to convexify, i.e. non-linearities in the constrain remain after the lc., an iterative process called *successive convexification* is performed, where the constraint is successively approximated with information from the previous iteration. This framework can handle hig-order gravitational harmonics and enables the extension of the 3DOF problem into a 6DOF by coupling the thrust with the attitude.

In Pseudospectral Methods, the infinite-dimensional problem of minimizing $J = \int \mathbf{a}^T \mathbf{a} d\tau$ is transformed in a discrete, finite-dimensional Non-Linear Programming problem, that can be solved using off-the-shelf solvers. Challenges in this approach are that the quality of the solution strongly depends on the Jacobian matrix of the model, and therefore on the ability to accurately model the dynamics. Another challenge is that the real-time implementation remains difficult due to the heavy computational cost.

Both in [11] and [15] can be found references that point towards works that explore different applications of the problem. In particular, [11] mentions how advances on Model Predictive Control theory promise to generate a framework of control methods that can account for weak gravity and strong perturbations, which are particular problems of the Asteroid's environment. To tackle the high uncertainty environment, [11] points towards the usage of *desensitized optimal control*, a framework to construct the reference trajectory where the sensitivity of the solution to the system's state uncertainty is penalised. Another approach to tackle the same problem is to search for, alongside the fuel-optimality, a solution that minimized the system's covariance.

3.5 Other Paradigms

There exist other algorithms (although in some form related to what has been shown) that have been used in past works. For example, in [20] is presented a general framework that includes trajectory propagation, optical-based navigation and an approach for maneuver design and targeting. Regarding the latter, the authors use what could be understood as a mix of the Proportional and Predictive algorithm to provide stochastic corrections among deterministic burns during an asteroid descent. The control algorithm used in [20] is Eq. (3.34). $\mathbf{r}_t$ is a target position state on the reference trajectory at the time of the next maneuver, $\mathbf{r}$ the current position and $\mathbf{K}$ a sensitivity matrix whose computation is done numerically.

$$K \triangleq \left[\frac{\delta \mathbf{r}_t}{\delta \Delta \mathbf{v}}\right] \tag{3.33}$$

$$\Delta \mathbf{v} \triangleq \mathbf{K}^{-1} \mathbf{r} \tag{3.34}$$

The reference trajectory around which these corrections are done uses 4 maneuvers to take the s/c from an asteroid-fixed hovering position towards a point in the surface. This hovering-to-surface guidance approach was also used in the Hayabusa2 mission [3] in an approach they call *Pin-point Touchdown*. In that mission, the descent phase was separated in a ground-aided navigation phase ("CGP-NAV", from 20km to 45m) and a final autonomous GNC landing ("6DOF Control", below 45m). During CGP-NAV, the guidance in the vertical direction was done first by a SMC law (aided by ground-segment) and upon smaller heights changed to a proportional-derivative law (completely autonomous). In the aforementioned cases the guidance is based on the availability of great control authority and both require of strict initial conditions to start the descent. In the experimental phase, Milani is likely to perform 5 control maneuvers, the last of which would put it in a interception ballistic trajectory from a height of 10km approx. Therefore, a GnC algorithm that operates within a framework of body interception is preferred, which in one hand will minimize the impact of the landing phase design on the rest of the mission; and in the other will be simpler and more reliable. Therefore, the GnC algorithm choice will be done among those summarized in section 3.6.

3.6 Summary of GnC Strategies

In the Table 3.1 below is available a summary of the aforementioned GnC strategies. In the conclusions of [15], the authors remark that Proportional and OGL are widespread algorithms for small body targets, while the use of Predictive & Hybrid algorithms should respond to an effort of reducing consumption at the expense of accuracy. They note that advancements in GnC-related solvers would eventually allow closed-loop on-board computation of open loop commands, but that the industrial trends are to either use solutions that combine open-loop trajectories corrected by compensators or use simple-closed loop laws (e.g. the OGL) aided by the usage of way points in a reference trajectory.

Table 3.1: Highlighted characteristics and relevant comments extracted from Simplício et al. review.

Name	Algorithm	Fuel-Optimal	Path Constraints	Particular comments
Quadratic	$\mathbf{a}(t) = \mathbf{C}_0 + \mathbf{C}_1 t + \mathbf{C}_2 t^2$	No	No	ToF can be selected with a minimum fuel-spending search
Optimal w/ Path Constraints	$\mathbf{a}(t) = \mathbf{C}_0 + \mathbf{C}_1 t + ...\mathbf{C}_N t^N$ (other shapes available)	Yes	Yes	Not suitable for complex and perturbed g
Proportional (APNG)	$\mathbf{a}(t) \triangleq nV_c(t)\dot{\Lambda}(t) - \frac{n}{2}\mathbf{g}_\perp(t) \equiv \frac{n}{t_{go}(t)}\mathbf{ZEM}(t)$ (gravity term optional)	No	No (Adaptation for non-continuous thrust can be found inside Simplício et al.)	Huge flight heritage, the complexity of gravity field integration is embedded on computation of ZEV
Predictive and Hybrid	$\Delta\mathbf{v}(t) \triangleq Vc(t)\hat{\mathbf{r}}^p(t_f) - \mathbf{v}(t)$	No (more fuel efficient than APNG)	No	There is a trade-off between fuel consumption (earlier firings) and impact performance degradation. To tackle this: Hybrid: use in early stages KI/PI and then switch to APGN. These laws are less accurate than Proportional and OGL because they rely on linearized propagation of states.
Optimal without Path Constraints	$CTVG: \quad \mathbf{a}(t) = \frac{6}{t_{go}^2(t)}\mathbf{ZEM}(t) - \frac{2}{t_{go}(t)}\mathbf{ZEV}(t)$ $FTVG: \quad \mathbf{a}(t) = \frac{3}{t_{go}^2(t)}\mathbf{ZEM}(t)$	Yes (up to choice)	Not explicitly, but in Simplício et al. there is a list of works to that have adapted the method to enforce particular path constraints.	Accurate integration of changing g is important. If it is hard to do, the integral can be computed piece-wise considering g constant. Each final states are way points in a reference trajectory. CTVG's fuel consumption is greater than FTVG's.
Optimal Sliding Guidance	$\mathbf{a}(t) = \frac{6}{t_{go}^2(t)}\mathbf{ZEM}(t) - \frac{2}{t_{go}(t)}\mathbf{ZEV}(t) - \frac{\phi}{t_{go}(t)}sgn(\mathbf{s}(t))$ $\mathbf{a}(t) = \frac{3}{t_{go}^2(t)}\mathbf{ZEM}(t) - \frac{\phi}{t_{go}(t)}sgn(\mathbf{s}(t))$	No (but close due to being an augmentation of optimal)	Same as above	SMC augmentation allows for a reduction in way points. Increased robustness degrades performance and increases roughness of acceleration commands. Globally stable if the upper bound of un-modeled dynamics is known. More simple commands that are SMC more robust exist.
Multiple Sliding Guidance	$\mathbf{a} = -\frac{\Lambda}{(t_f - t)}\dot{\mathbf{s}}_1 - \frac{\Lambda}{(t_f - t)^2}\mathbf{s}_1 - \Phi sign(\mathbf{s}_2) - \mathbf{g}_{NI}(t, \mathbf{r})$	No (tunable)	Same as above	Same SMC properties as above, but higher control authority. Has worse fuel performance than OSG.
Computational GnC	-	Yes	Yes	Main problem is to accurately and efficiently deal with either non-convex constraints or to solve a discrete, finite-dimensional NLP problem. Higher CPU consumption but specialized solvers are being developed.

3.7 Algorithm Selection for Milani's DnL

In this section is presented the rationale for the choice of one algorithm among those in Table 3.1. At the time of writing, the landing of Milani CubeSat is not critical to mission success and the GNC system has not been designed to optimize for descent and landing requirements. In particular, Milani is expected to perform only discrete ($300s$) firings. Nonetheless, the requirements of the landing phase (also known as "experimental phase") are still under consideration and might change. Hence it cannot be excluded that continuous thrust would be finally adopted. For this reason, in the choice of the MSSG, I first assume that continuous thrust and closed-loop control are available.

3.7.1 Selection Criteria

The High Level Requirements (HLR) with which the GnC algorithm shall comply are displayed in Table 3.2:

Table 3.2: High Level Requirements for the Descent & Landing guidance algorithm.

ID	Requirement statement
HLR-01	The algorithm shall operate in a binary asteroid dynamical environment.
HLR-02	The algorithm shall comply with the limited CubeSat's capabilities.
HLR-03	The GNC system shall operate autonomously

Each of the HLR of Table 3.2 leads to design drivers. These are:

1. The algorithm shall operate in a binary asteroid dynamical environment.
    1.1. The algorithm shall be robust in front of unmodeled dynamics.
    1.2. The algorithm shall provide an accuracy sufficient to perform a soft landing.
2. The algorithm shall comply with the limited CubeSat's capabilities.
    2.1. Lower CPU resources expense is preferable.
    2.2. Lower Fuel resources expense is preferable.
3. The GNC system shall operate autonomously.
    3.1. The algorithm shall be reliable.
        3.1.1. Simple algorithms are preferable.
        3.1.2. Algorithms with higher TRL are preferable.
    3.2. The possibility to enforce path constraints is preferable.

The algorithms' performance on compliance with these drivers has been evaluated and is visually shown in Table 3.3 below. From such evaluation it follows:

- Computational GnC poorly complies with 2.1 and 3.1
- Open-Loop schemes poorly comply with 1.1 and 1.2
- Proportional, Predictive & Hybrid poorly comply with 1.1, 1.2 and 3.2
- Optimal without Path Constraints algorithms poorly comply with 1.1

and hence they can be first ruled out. The ones remaining are the MSSG and the OSG algorithms. In the section below I discuss the choice between the two.

3.7.2 Choice between OSG and MSSG

The MSSG has been chosen over the OSG. The reasons can be synthesised in two points:

- A main feature of the OSG is that is near fuel-optimal. Nonetheless, it requires on-line propagation of the orbital dynamics which punishes its robustness. Since the s/c is at its mission end, the remaining fuel is fully available for the landing process, and thus the fuel consumption is secondary with respect to robustness. This means that the MSSG robustness primes over the OSG fuel optimization.
- It has been observed[1] that on the Didymos environment the MSSG algorithm has greater control authority than OSG at optimal gains. This results in the OSG having robustness weaknesses (detailed below) that the MSSG does not, and so the MSSG is chosen over the OSG.

The aforementioned weaknesses from which the OSG suffers in the Didymos dynamical environment are:

1. The fact that the landing point rotates wrt. inertial space with a relatively high velocity[2] imposes a problem that can be described from two points of view:
    - If the OSG command is computed on the N RF (see Table B.1), the landing point moves fast, rotating about an axis, and the algorithm needs to constantly update what would be a desired trajectory to impact the target with the relative desired states (namely zero position and velocity error). This makes the OSG fail to converge to a static landing trajectory.
    - If the OSG command is computed on the D2 RF (see Table B.1), the target states are constant, but the dynamics of the trajectory is ruled by the gravitational field and the quickly changing inertial forces. Then, the OSG computes at each time an acceleration command in the D2 RF that should move the uncontrolled trajectory endpoint towards a desired position, but this last step is greatly degraded by the strong non-inertial dynamics.
2. The second point of view above entails that the OSG is slower to converge to $s_1 = 0, \dot{s}_1 = 0$ than the MSSG (even the latter being at minimum gains and its slowest convergence rate). In the end, the OSG does reach $ZEM = 0$, $ZEV = 0$, but starts to strongly converge only when t_{go} gets closer to zero and the control gains grow. Also, since typically is needed to reduce the velocity to perform a soft-land interception, the OSG has the tendency to reach the landing target by doing a small loop (radius of $\mathcal{O}(1m)$) that passes below the surface of the asteroid. Path constraints to avoid this effect can be introduced by tuning the OSG gains previous or during flight [21], but such tuning would require dedicated studies and is left out of the scope of this work.
3. Another issue of the OSG method is related with the choice of the ToF and the ability of the algorithm to drive the endpoint of the virtual uncontrolled trajectory towards the desired landing point. If the ToF is selected such that the endpoint of the uncontrolled trajectory is further away than the asteroid, the OSG needs to bring the endpoint closer to the current position. A problem arises in the computation of these kinds of virtual uncontrolled trajectories. That computation can be done in two ways:
    - Do not inform the OSG method that there is an asteroid, and let the virtual uncontrolled trajectories pass through. It has been observed that this leads to the virtual

[1] Using the implementation proposed in Appendix B with unbounded thrust and unperturbed environment.

[2] Note that this issue is present because the target is Dimorphos. In past literature the Asteroid-attached RF was assumed to be rotating but at lower rates and also non-accelerated. See [15]–[17], [19].

trajectory pass near the asteroid's center and hence be impossibly accelerated by a fictitious gravity.

- Inform the OSG method that there is an asteroid, and stop the virtual trajectories if they collide with the surface. This has been observed to generate jumps in the position error, that greatly degrade convergence. Also, it has been noted that the virtual trajectories colliding with the asteroid will reach it with a relatively large relative velocity, which then the OSG needs to slow down and the problem mentioned in point 1 is encountered again

Finally, it could be argued that the gains of the OSG method could be tuned (probably by increasing them significantly) such that its control authority increases and the issues presented are nullified. In that case though, the fuel expense becomes larger than the optimal (which OSG would generate at optimal gains); and therefore it would become pointless to use an algorithm that is more complicated in terms of computation and CPU than the MSSG, and whose main advantage was to be derived from an Optimal Guidance Algorithm that minimized fuel usage. Another way to avoid the issues mentioned above would be to use the OSG method to track a reference trajectory. This has been done before [15] and the method is great at it, since it would target consecutive points on a dynamically compatible[3] reference trajectory. Nonetheless, to use a reference trajectory in Milani would require knowledge of Didymos' dynamical environment; which, even if available, would punish the robustness criterion.

On the contrary, as explained in sec. 3.3.4, the MSSG operates differently and does not suffer from the mentioned issues: the MSSG cancels the modelled perturbing dynamics at each instant, and if enforced in the asteroid-attached RF, synchronizes the s/c with the rotation of the asteroid. Then, in that frame, the MSSG exerts commands that monotonically decrease the distance between the s/c and the landing target, as well as the relative velocity.

From the above discussion, I have chosen the MSSG over the OSG method. Note that the elimination process is fundamentally ruled by robustness (which in small bodies' operations implies accuracy) and CPU, which are important drivers to adress the specific problem of operating a CubeSat in a distant small body environment.

[3] Meaning that the desired states (r, ν) would be relatively easy to achieve by deflecting the actual s/c trajectory to match a reference one.

Table 3.3: Trade-off aimed at the selection of a GnC algorithm for Milani's DnL. Greener colors represent a better performance and red represent a worse performance. White cells are those where a comparison in the given column does not apply. N-O: Non optimal. O: Optimal.

Parent Drivers	**CubeSat**		**Path Constraints**	**Asteroid problem particularities**			**Reliability**
Algorithms/Drivers	**CPU c.**	**Fuel c.**	**-**	**Robustness (sensitivity to uncer.)**	**Accuracy (dispersion at target)**	**Measurements (are needed?)**	**Flight Heritage**
Quadratic	Open-Loop	N-O (search ToF for fuel)	No	Low (errors in g prediction)	Low (no meas.)	No	Yes (Apollo, Curiosity)
Optimal w/ Path constraints	Open-Loop	O	Yes	Low (research to apply SMC has been done)	Low (no meas.)	No	Yes (Falcon)
Proportional	Low (g can be const. or not itegrated)	N-O	No	Mid (info on g optional, can not depend on model)	Mid (measurement on position)	Yes	Yes (Interceptors)
Predictive and Hybrid	Mid (linearized orbit perturbations propagation)	N-O (better than Propor.)	No	Low (dependence on propagation of dynamics)	Low (dependence on prediction)	Yes (KI)	Yes (Interceptors)
Optimal w/o Path Constraints	High (ZEV/ZEM integral. Use way points.)	O	No, but can be indirectly dealt with	Mid (usage of info on position and velocity.)	Mid (source of inaccuracies is modeling err.)	Yes	No, but industrial trend
Optimal Sliding Guidance	High (Same as above)	N-O (close if chattering-free augmentations)	No, but can be indirectly dealt with	High	High (improved due to high robustness)	Yes	No, but industrial trend
Multiple Surface Sliding G.	Low (no propagation)	N-O (tunable)	Same as above	High	High	Yes	No, but industrial trend. Applicable cases studied.
Computational GnC	Very High	Yes	O	High (methods to optimise robustness)	High (methods to optimise robustness)	Yes	No, in development

Chapter 4

The Multiple Sliding Surface Guidance Algorithm

In this chapter I study in detail the Multiple Sliding Surface Guidance Algorithm (MSSG); discuss how to implement it and how does it perform under Didymos' dynamical environment. The model and validation of the dynamical environment, as well as the summary of how the MSSG is implemented can be found in Appendix B. The chapter is organized as follows: Section 4.1 goes through an introduction to Sliding Mode Control theory; Section 4.2 presents the derivation of the MSSG; and Section 4.3 contains the study of the MSSG performance and a parametric analysis on Didymos' environment. In this last section are studied the sliding mode dynamics, the effect of the MSSG parameters, the effect of the trajectory's ToF, and the effect of random perturbations.

4.1 Introduction to Sliding Mode Control Theory

4.1.1 Basic Definitions

A system $\dot{x} = f(x,t)$ is said to be in Sliding Mode[1] if its states x evolve through a prescribed trajectory called Sliding Surface (if its defined by a linear combination of states) or a Sliding Manifold (if it is a non-linear combination of states). A Sliding Manifold is defined by the 0 contour of the Sliding Variable function, that we can call s:

$$s = s(x,t) \tag{4.1}$$

In other words, when $s = 0$ the system is said to be in Sliding Condition. Sliding Mode Control (SMC) is the control discipline whose objective is to rule a system by bringing it to a desired Sliding Manifold and make it remain thereafter, because in such case the dynamics of the system will be the ones defined by the Sliding Manifold. A particularly interesting aspect of SMC is its ability to guarantee that a system is brought to and maintained on a Sliding Condition regardless of the presence of bounded disturbances. The *order of the Sliding Mode* is determined by the time derivative of the output at which the control first appears explicitly:

Definition: Let $\dot{x} = f(x,t)$ be a smooth system with a smooth output $y(x,t)$. Provided that $y, \dot{y}, \ddot{y}..., y^{(r-1)}$ are continuous, and that $y, \dot{y}, \ddot{y}..., y^{(r-1)} = 0$, the motion on the set $\{y, \dot{y}, \ddot{y}..., y^{(r-1)}\} = \{0, 0, 0, ...0\}$ is said to exist in *r-th order sliding mode* (or r-sliding mode) [17]. Also, the output y is said to be of *relative degree r*.

This is, if the control first appears in the r-th time derivative of the output and not on the previous, but all of the derivatives are continuous, the system is in an r-th order sliding mode. E.g. a

[1] Also said to be on the Sliding Phase, or in the Sliding Condition

system excited by a control u will be on 2nd order sliding mode if $\ddot{y} = g(u)$ and $(y, \dot{y}) = 0 \neq h(u)$.

The phase during which the system is driven towards (but is not on) the Sliding Surface is called Sliding Phase. The desired dynamics of the Sliding Variable when outside the Sliding Surface is called Reachability Condition. Meeting it means that the system is driven towards the Sliding Surface. Typically the Reachability condition is designed using Lyapunov techniques, to ensure that the system is stable and is brought to zero ($s \to 0$) in finite-time (FT). Hence it is convenient to introduce the Second Lyapunov Stability Theorem.

4.1.2 Second Lyapunov Stability Theorem and Finite Time Stability

Theorem: Let $\dot{s} = f(s,t)$ be an autonomous non-linear dynamic system with an isolated equilibrium point at s^*. If in the neighborhood D of the point s^* exists a scalar function $V(s)$ s.t.:

$$V(s) > 0 \ \forall s \neq s^*, \qquad V(s^*) = 0 \tag{4.2}$$

$$d(V(s))/dt \leq 0 \ \forall s \neq s^*, \qquad d(V(s^*))/dt = 0 \tag{4.3}$$

then the system is said to be *asymptotically stable* (perturbations are driven to zero asymptotically). If D contains all possible states s then the system is *globally asymptotically stable.*

To make the system FT stable (perturbations about the equilibrium are driven to zero in FT), the Reachability Condition on Eq. (4.3) can be further restricted to:

$$d(V(s))/dt \leq -\eta|s| \ \forall \ s \neq s^*, \qquad d(V(s^*))/dt = 0, \qquad \eta > 0 \tag{4.4}$$

A typical candidate function satisfying Eq. (4.2) is $V(s) = \frac{1}{2}s^2$. Using such function, a system meeting the Reachability Condition of Eq. (4.4) will evolve as:

$$\dot{s} \leq -\eta|s| \to \dot{s} \leq -\eta|s|/s = -\eta sign(s) \to \forall s \geq 0 : \ \dot{s} \leq -\eta \to \int_{s_0}^{s(t)} ds \leq -\eta \int_{t0}^{tf} dt$$

$$\to s(t) \leq s_0 - \eta(t - t_0) \to s_0 = \eta(t_r - t_0) = \eta \Delta t_r$$

and therefore the system will reach $s(t_r) = 0$ if $\eta \geq \frac{s_0}{\Delta t_r}$. Importantly, using a control u that imposes $\dot{s} = -\Phi sign(s_2) \leq -\eta sign(s)$ is equivalent to meeting Eq. (4.4), and thus it yields a system that is finite time stable, with a reaching time[2] of $\Delta t_r = \frac{|s_0|}{\Phi}$.

Let s now be a sliding variable that we wish to bring to 0. If the derivative of the control already appears explicitly on the first derivative of s, i.e. the system of 1st relative order, using the process described above: $\dot{s} = -\Phi sign(s) = g^*(u) \to u = g(-\Phi sign(s))$, and thus u is very straight-forward to obtain. Then, u and can keep the system at exactly $s = 0$ by switching infinitely fast following the $sign(s)$ function about $s = 0$. If the system is of higher order this is not true and other techniques shall be used to design appropriate controllers, a discipline called Higher Order Mode Control (HOMC) [22].

4.1.3 Chattering for 1st order SMC

When the system reaches the sliding phase ($s(t) = 0 \ \forall t \ \geq t_r$), a control such as the proposed above (i.e. $u(t) \ s.t. \ \dot{s} = -\Phi sign(s)$) will keep $s = 0$, and thus $\dot{s} = 0$. This is achieved by an infinitely fast switching of the function $sign(s)$. In computer simulations or real systems,

[2] Reaching time: time at which the system reaches the Sliding Surface: t_r. In this work it is used synonymously to refer at the time it takes the system to reach the sliding surface: $\Delta t_r := t_r = \Delta t_r + t_0$. Also $\Delta t_r = n \Delta t_f = n \cdot ToF$.

this switching is not infinitely fast, and typically undesirable, since makes the control function $u(t)$ switch equally fast. To mitigate chattering in 1st relative order systems, there are different approaches [22], [23]:

- Boundary Layer (continuous control): in a boundary layer around $s = 0$, approximate the discontinuous $sign(s)$ function by a continuous one, such as the sigmoid function:

$$sigmoid(s) = \frac{s}{|s| + \epsilon} \quad \epsilon > 0 \tag{4.5}$$

  where ϵ is a small number. This approach degrades the performance and the system cannot maintain $s = 0$ exactly, although can be tuned to remain close. For this reason, this is called *quasi-sliding mode control.*

- Asymptotic Sliding Mode: a way to attenuate chattering is to use a virtual variable ν as control (the one ruled by $sign(s)$), and let the system input u be the integral of such control: $\dot{u} = \beta$. In this way, the physical input is

$$u(t) = \int_{t_0}^{t} \beta dt = \int_{t_0}^{t} \beta(sign(s))dt \tag{4.6}$$

  and the chattering is "hidden" in ν. The variable $u(t)$ still suffers from chattering, but much more reduced in magnitude. With this method a computer still needs to integrate a very fast switching function β.

4.2 MSSG Derivation

4.2.1 Dynamics Notation

To now tackle the Multiple Sliding Guidance method, first it's needed to introduce some notation describing the dynamics. Consider the RF described in Table 4.1. Let R be the inertial position of the s/c, and R_a the inertial position of the asteroid, then the relative position of the s/c with respect to the asteroid, in the D2 RF:

$$r = R - R_a \tag{4.7}$$

let now ν be the relative velocity of the s/c with respect to the D2 RF, expressed in the D2 RF:

$$\nu = \left.\frac{\partial r}{\partial t}\right|_{D2} = \dot{r} \tag{4.8}$$

Assume that D2 is in equilibrium within the full 2BP and hence its rotational state is constant [1]. Then, the dynamics of a constant-mass s/c relative to the asteroid are described by:

$$\begin{cases} \dot{r} = \nu \\ \dot{\nu} = -2\omega \times \nu - \omega \times \omega \times r - \ddot{R}_a + g(r,t) + u \end{cases} \tag{4.9}$$

where ω is the angular velocity of the D2 RF, $g(r, t)$ are the gravitational and perturbing forces on the s/c and u is the control acceleration. In Eq. (4.9) the D2 RF acceleration, $\ddot{R}_a = \left.\frac{d^2 R_a}{dt^2}\right|_N$, is extracted from the target asteroid's ephemeris.

Table 4.1: Reference frames being used for the derivation of the MSSG algorithm.

Abbreviation	Full name	Inertiallity	Comment
D2 RF	Dimorphos Body-attached RF	No	Centered in D2 CoM, fixed on the body. Defined by ESA's Kernels.

4.2.2 SMC Applied to Asteroid Landing

The derivation shown on this subsection is extracted from [17] unless otherwise stated. The objective of the MSSG is to control the relative position and velocity of the s/c (r, ν) such that they reach a desired value (r_d, ν_d) (landing states) at a time t_f (landing time). The control is sought to be robust to bounded perturbations. At the same time, it is advantageous to prevent chattering, and an option is to define a sliding surface that is only crossed when the problem ends (the s/c lands). To do so, a sliding surface s is defined as:

$$s_1(r, t) = r(t) - r_d(t) \tag{4.10}$$

where $r_d(t)$ is the desired landing position. All the vectors in Eq. (4.10) and all vectors mentioned hereafter are expressed in the D2 RF. Note that the problem driving and maintaining $s(r, t)$ on 0 is a 2nd order SMC, since the control first appears explicitly on the 2nd derivative of the surface s:

$$s_1 = r - r_d \tag{4.11}$$

$$\dot{s}_1 = \dot{r} - \dot{r}_d = \dot{r} \qquad \text{(the landing point is fixed on the D2 base)} \tag{4.12}$$

$$\ddot{s}_1 = \ddot{r} = -2\omega \times \nu - \omega \times \omega \times r - \ddot{R}_a + g(r, t) + u \tag{4.13}$$

to grant a sliding mode on the surface $s_1 = 0$, the Lyapunov FT stability conditions of Eqs. (4.2) and (4.4) have to be met. This is done by, instead of imposing a $\dot{s}_1 = g(u(t))$ (because the system has a 2 relative order), using the velocity $\dot{s}_1$ as the virtual controller, such that:

$$\dot{s}_1 = -\frac{\Lambda}{t_{go}} s_1, \quad t_{go} = t_f - t \tag{4.14}$$

where Λ is a diagonal matrix of positive gains. If such equality is fulfilled, it can be shown how:

$$s_1(t) = s_{1,0}(1 - t/t_f)^{\Lambda} \tag{4.15}$$

$$\dot{s}_1(t) = -\frac{\Lambda}{t_f} s_{1,0}(1 - t/t_f)^{\Lambda - I} \tag{4.16}$$

and therefore the surface will get to $s_1 = 0$ at t_f if $\Lambda > I$ and $\dot{s}_1 = 0$ at t_f if $\Lambda > 2I$. Then, the Lyapunov conditions are met and the s_1 will FT stable. Importantly, the minimum gain to achieve a landing with zero relative velocity is $\Lambda = 2I$. The second step is to seek a control such that Eq. (4.14) is met. This is done by defining another sliding surface:

$$s_2 = \dot{s}_1 + \frac{\Lambda}{t_{go}} s_1 \tag{4.17}$$

which is of 1 relative degree and therefore can be controlled by the standard SMC methods:

$$\dot{s}_2 = \ddot{s}_1 + \frac{\partial}{\partial t}\left(\frac{\Lambda}{t_{go}} s_1\right) = -2\omega \times \nu - \omega \times \omega \times r - \ddot{R}_a + g(r, t) + \frac{\Lambda}{t_{go}} \dot{s}_1 + \frac{\Lambda}{t_{go}^2} s_1 + u$$

note the control u appearing as the last term of the RHS of the equation above. Finally, the control u is set such that $\dot{s}_2 = -\Phi sign(s_2) \Rightarrow$ Eq. (4.3) is fulfilled and $s_2 \to 0$ in FT:

$$u = 2\omega \times \nu + \omega \times \omega \times r + \ddot{R}_a - g(r, t) - \Phi sign(s_2) - \frac{\Lambda}{t_{go}} \dot{s}_1 - \frac{\Lambda}{t_{go}^2} s_1 \tag{4.18}$$

where $g(r,t)$ stores all the modelled perturbations on the s/c (in this work, the gravitational pulls and the SRP). It can be easily shown how s_2 will remain stable ($sign(\dot{s}_2) = -sign(s_2)$) if $\Phi \geq |p_{max}|$, where p_{max} is the maximum unmodeled perturbation, and thus this algorithm is robust in front of bounded unmodeled dynamics [17]. The expression can be further compressed by storing all the non-inertial accelerations and perturbing accelerations in a term $g_{NI}(t,r)$:

$$u = -\Phi sign(s_2) - \frac{\Lambda}{t_{go}}\dot{s}_1 - \frac{\Lambda}{t_{go}^2}s_1 + g_{NI}(r,t) \tag{4.19}$$

where $g_{NI}(t,r) = 2\omega \times \nu + \omega \times \omega \times r + \ddot{R}_a - g(r,t)$.

4.3 MSSG Performance and Parametric Analysis

Eq. (4.19) is called the Multiple Sliding Guidance Algorithm. Note the following observations:

- The landing states, expressed in the D2 RF, are constant, and therefore the ToF is a free parameter: changing the ToF will change the landing time and therefore the inertial location of the landing target. Such change of position is "hidden" in the term $\ddot{R}_a$ of the dynamics' propagation in Eq. (4.9).
- Since the dynamics equations upon which the MSSG has been constructed describe the relative position and velocity of the s/c with respect to the D2 RF, the control effort will try to nullify ν and thus the s/c will synchronise itself with the rotation of D2, which is advantageous from the relative navigation perspective.
- The MSSG algorithm as presented does not have information about the presence of the asteroid's surface and the possibility of collision at a different point than r_d. Indeed, the trajectory generated will not collide with the asteroid surface in the case that the selected target point is on the surface cap visible to the s/c at the start of the control.
- Chattering will appear $s_2 = 0$, ideally only when $t > t_r$.
- Compared to the OSG (see section 3.3.4), the MSSG is a "brute force" approach, since it does not take into account that the dynamics would propagate on its own: instead, it cancels the dynamics and commands the s/c towards the landing target. By this, the MSSG sacrifices fuel performance but compensates it by having a greater control authority, since the trajectory on the D2 RF is no longer bounded by the propagation of gravitational dynamics.

The aim of the rest of Chapter 4 is to study the behaviour of the MSSG as derived using the implementation presented in Chapter B. Even if the dynamical environment is Didymos', the spirit of this study is to analyse the MSSG algorithm alone, and thus no thrust constraints derived from the Milani s/c design have been introduced. Specifically, in this section the thrust is left to be unbounded in magnitude and direction, and it is not quantized. Also, the states (r, ν) are assumed to be perfectly known in the closed-loop control. Nonetheless, since chattering of $sign(s_2)$ introduces simulation challenges, a Schmitt trigger and a *sigmoid()* function (see section 4.1) have been used in all simulations that appear hereafter. Both devices are tuned as explained in section 4.3.2.

4.3.1 Sliding Mode Dynamics

The standard order SMC action ($u(t)$ $s.t.$ $\dot{s}_2 = -\Phi sign(s_2)$) is responsible for

- Driving the surface s_2 to 0 in FT (see in section 4.1). The reaching time is: Δt_r : $\Phi = \frac{|s_2(0)|}{\Delta t_r}$; after which the s_1, $\dot{s}_1 \to 0$ in FT and s_1, $\dot{s}_1 = 0$ at t_f (landing time) due to the dynamics imposed by Eq. (4.14).

- Reject perturbations bounded by $|p| < \Phi$ and keep the system converging towards $s_2 = 0$ regardless of bounded unmodeled dynamics

Thus, the value of Φ can be determined from two requirements: i) the necessity to converge at a time t_r departing from an initial error $s_2(0)$ and ii) the necessity to to reject disturbances up to a given magnitude $\mathbf{p}_{max}$. Therefore, two Φ constants can be introduced: one that rules the reaching time and another that rules the stability on sliding mode:

$$\Phi_{conv} = \frac{|s_2(0)|}{\Delta t_r} \tag{4.20}$$

$$\Phi_{stab} > |p_{max}| \tag{4.21}$$

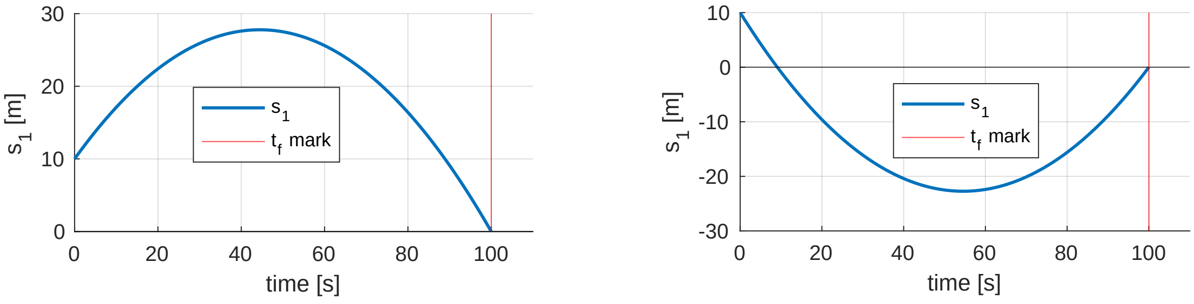
The value of Δt_r will usually be specified by the constant n: $\Delta t_r = n \cdot ToF$. Both Φ values can diverge significantly, as the $s_2(0)$ might be great when the control is started but the disturbances to reject can be relatively small. Thus, it could be advantageous to decouple the two problems and establish two different Φ. Importantly, note that if $\Phi_{stab} > \Phi_{conv}$, and $\Phi = \Phi_{conv}\ \forall\ t < t_r$, during the Reaching Phase the system would not be able to reject disturbances, and therefore it would not be guaranteed that $s \to 0$ in t_r. This could be tackled by setting Φ s.t.:

$$\Phi = \begin{cases} max\left(\frac{|s_2(0)|}{\Delta t_r}, |p_{max}|\right) & \forall\ t < t_r \\ |p_{max}| & \forall\ t \geq t_r \end{cases} \tag{4.22}$$

Another subject is the effect of the failure to impose s_2 during the flight. In practice it is not needed for the system to be at $s_2 = 0$ to drive the states $s_1 \to 0$, just with s_2 getting closer to 0 $s_1 \to 0$. E.g., if $s_2 = c \in \mathbb{R}\ \forall t$ and $\Lambda = 2$:

$$\dot{s}_1 = c - \frac{2}{t_{go}} s_1 \to s_1(t) = s_{1,0}\left(\frac{t_{go}}{t_f}\right)^2 + \frac{c}{t_f}\left[\frac{t_{go}}{t_f} - \left(\frac{t_{go}}{t_f}\right)^2\right] \tag{4.23}$$

an example of the evolution of $s_1(t)$ for the case where $c \gg 0$ is displayed in Fig. 4.1a below.



(a) $s_2 = ct. = c = 1m/s$ **(b)** $s_2 = ct. = c = -1m/s$

Figure 4.1: Evolution of $s_1(t)$ for fixed parameters: $t_f = 100$s, $s_1(0) = 10m$, and $\Lambda = 2$.

In Fig. 4.1a can be seen how if $s_2 = c \neq 0\ \forall t$, the control still brings s_1 to $s_1 = 0$ at t_f. This is important because i) it means we can expect to reach the landing point even if the reaching time is the landing time; and ii) even if the control u can't keep $s_2 = 0$ exactly (but $s_2 = ct. = c \approx 0$) the system will still reach the landing point. Nonetheless, note that a convex evolution is shown, which would mean that the s/c would start to drift away from the landing point before pointing back towards it. This can be avoided by ensuring that

$$\ddot{s}_1 \leq 0 \to 0 \geq \frac{d}{dt}\left[c - \frac{\Lambda}{t_{go}} s_1\right] \to c \leq (\Lambda - I)\frac{s_1}{t_{go}} \tag{4.24}$$

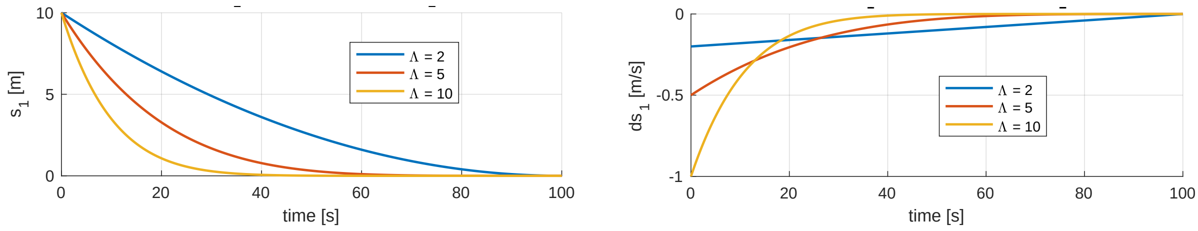
This is, to maintain the error of $s_2 \neq 0$ low enough. Note that if c is kept perfectly at 0 the condition is always met. In addition, there is a restriction on the fixed value of $s_2(0)$: the surface s_1 will cross 0 at a time determined by the solution of:

$$0 = s_{1,0}\left(\frac{t_{go}}{t_f}\right)^2 + \frac{c}{t_f}\left[\frac{t_{go}}{t_f} - \left(\frac{t_{go}}{t_f}\right)^2\right] \rightarrow t_{col} = f(c, s_1(0)) \tag{4.25}$$

or what is the same, at each t there is a limit value of c, the distance from $s_2 = 0$, beyond which s_1 will be crossing $s_1 = 0$:

$$c_{lim}(t) = \left(1 - \frac{t_f}{t}\right) s_1(0) \tag{4.26}$$

note that this is a non-issue if $t \rightarrow t_f$, because then $c_{lim} \rightarrow 0$ but at that moment the s/c is expected to be reaching the landing point $s_1 \rightarrow 0$ at a small relative velocity ($\dot{s}_1 \rightarrow 0$), and thus $s_2 \rightarrow 0$ naturally. Indeed, in Fig. 4.1b is displayed an example of such case. Luckily, in practical terms it is not true that $s_2 = ct. = c \ \forall t$ because the control $u \ s.t. \ \dot{s}_2 = -\Phi sign(s_2)$ will quickly drive s_2 to 0. Nonetheless, in reality unmodeled dynamics will make s_2 remain in a boundary layer around $s_2 = 0$, or at a value $s_2 = ct. = c \ll 1$, and thus the s/c will never i) first-time cross s_1 at t_f exactly and ii) with $\dot{s}_1 = 0$. Importantly, the velocity $\dot{s}_1$ will show the same behaviour, since in sliding mode $\dot{s_1}$ is constant-proportional to s_1 (see Eq. (4.9)).



(a) Evolution of $s_1(t)$

(b) Evolution of $\dot{s}_1(t)$

Figure 4.2: Dynamics of the s_1 surface in the ideal sliding mode, i.e. $s_2 = 0 \ \forall t > t_r$

The effect of changing the gain Λ can be understood as tuning how slow is the final approach to the landing point. Let the system be on the ideal sliding mode ($s_2 = 0 \ \forall t > t_r$). The evolution of s_1, $\dot{s}_1$ for different Λ is shown on Fig. 4.2. There, it can be seen how $\Lambda \uparrow$ makes the velocity $\dot{s}_1$ asymptotically approach the coordinates axis faster, meaning that the s/c will quickly approach the landing point to descend upon it progressively more slowly. Note that for $\Lambda = 2$, $\dot{s}_1(t)$ is a straight line: for smaller gains velocity would not converge.

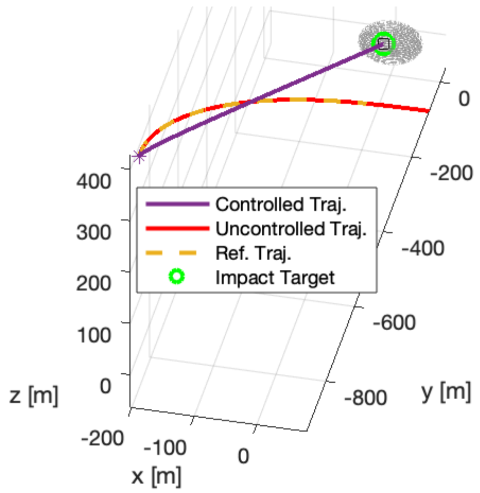
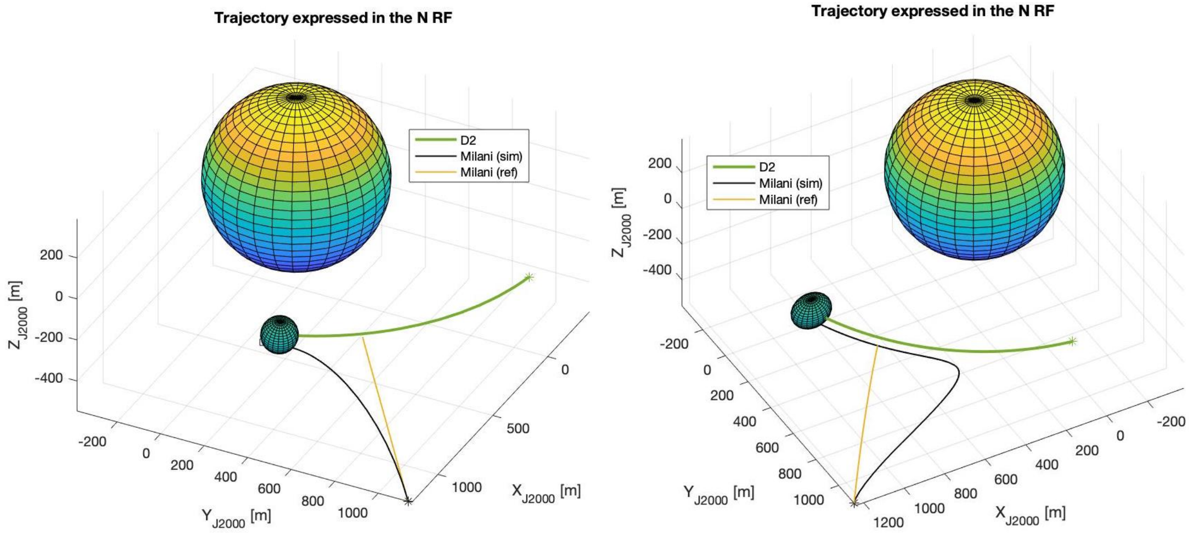


Figure 4.3: Example of a trajectory on the Sliding Surface $s_2 = 0$ expressed in the D2 RF.

Since the monotonic decrease of relative of position and velocity error is defined in the D2 RF, a decrease of t_r means that the s/c will synchronize itself with the asteroid motion quicker, "chasing" it so that the s/c remains in a relative linear descend towards the landing target. An example of that can be seen in Fig. 4.3. Also, the higher is Λ, the more time will be the s/c forced to slowly fall towards the target point. The combination of such effects mean that for a stronger control ($\Lambda \uparrow$ and $t_r \downarrow$) the s/c chases the target more intensely. An example of such phenomena can be seen in Fig. 4.4.



(a) $\Lambda = 2$, $n = 0.9$

(b) $\Lambda = 10$, $n = 0.1$

Figure 4.4: Example of inertial motion of the controlled s/c for high and low gains, expressed in the N RF (DidymosECLIPJ200). Note that the trajectory does not collide with D2 before the landing time but rather the s/c falls to D2 from its front.

4.3.2 Dealing With Chattering

In this work I have mitigated chattering using a mix between 2 of the previously presented approaches. First, a boundary layer around $s_2 = 0$ is introduced, by setting a constant ϵ s.t. that the bottom limit of the boundary layer activates the 1% of the *sigmoid* function[3]: $0.01 = sigmoid(s_{2,low})$. This makes the control profile smoother, but in exchange degrades the performance because if $s_2 \to 0$, $|\Phi \cdot sigmoid(s_2)| < \Phi$,(e.g. $|sigmoid(\approx 0)| \approx 0.5 < |sign(\approx)| = 1$), and the control cannot reject the maximum disturbance there. This has been observed to minimally affect the final performance of the algorithm. Indeed, for the fuel-optimal case of Fig. 4.7a below the relative difference on collision velocity between the use of $sigmoid(s_2)$ and $sign(s_2)$ with the proposed ϵ is of the $\approx 2\%$ on a value of $\approx 0.9\ cm/s$.

Importantly, the *sigmoid* function can also produce a very fast switching if s_{low} is relatively small. To ensure that the period of control oscillations is higher and that the SMC perturbation compensation is not activated if we are close to $s_2 = 0$, I have introduced a Schmitt Trigger to generate an oscillation between s_{high} and s_{low} The boundaries s_{high} and s_{low} have been set *ex post* simulation to:

$$s_{high} = 10^{-2}\ m/s, \qquad s_{low} = 10^{-4}\ m/s \tag{4.27}$$

Note that under the acceleration of Φ_{stab} or the common accelerations during a nominal approach[4], the period of oscillation in the band is of: $T_{band} \approx \frac{s_{high} - s_{low}}{10^{-4}} \approx 100s$. If the environmental acceleration was suddenly 10 times the max. expected, the period of oscillation would be still of $T_{band} \approx 10s$.

For illustration purposes, in Fig. 4.5 is displayed the evolution of some parameters involved in the MSSG execution (i.e. s_2, s_1, the acceleration command) for an example trajectory. The gains are $\Lambda = 2.1$ and $n = 0.6$. In Fig. 4.5 can be readily seen how the control brings s_2 to

[3] The mission Hayabusa2 also used this approximation [3].

[4] They are up to $\mathcal{O}(10^{-4})$, see Table 4.4.

0 at approximately t_r. Note that it is not at t_r precisely due to the control degradation from the *sigmoid* and the trigger. Regardless, s_2 is firmly maintained around $s_2 \approx 0$, and on the east-most plot Fig. 4.5 can be seen how the states r and ν go to zero at t_f. Note that the evolution of ν shows the "parabolic" shape of Fig. 4.1a due to $n = 0.6$ and the system being far from $s_2 = 0$ for a long time.

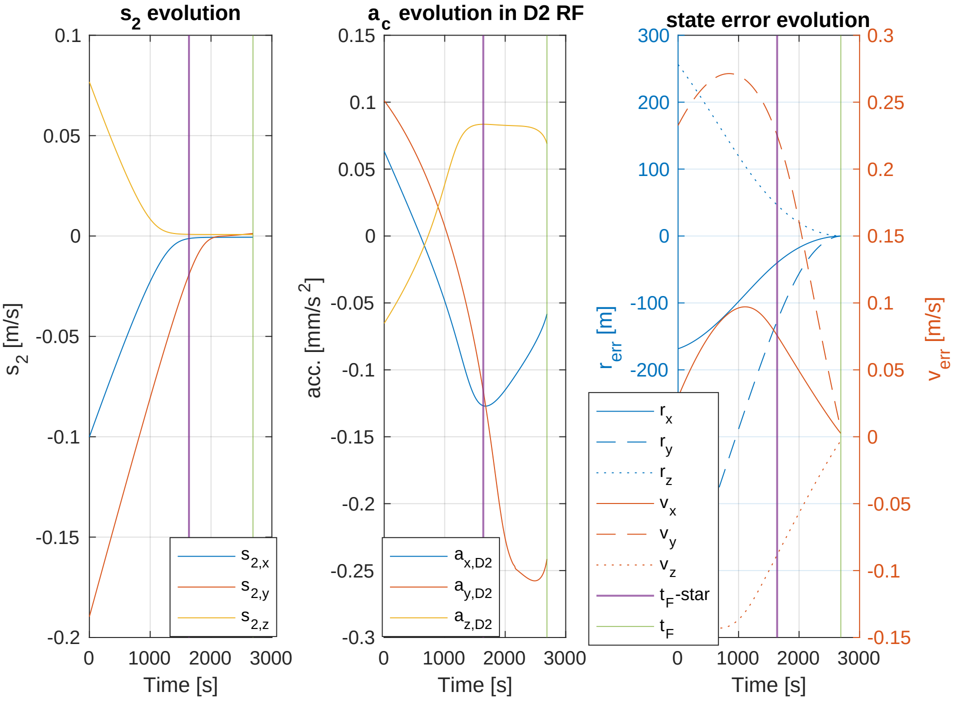


Figure 4.5: Evolution of the quantities involved in an MSSG execution. $t_F - star$ is used to label the reaching time, marked with a violet vertical line. The final time t_f is marked with a green vertical line.

4.3.3 Effect of Guidance Gain Λ and Reaching Yime t_r

Impact of Λ and t_r on Landing Performance

For some given departure states $(t_0, r(t_0), \nu(t_0))$, and a landing target $(r_L = ct.)$, the performance of the MSSG algorithm depends on the control starting epoch t_0; the wished landing time t_f; the value of Φ; and in which way the control is exerted (e.g. a Schmitt trigger, a PWPF modulator...). Usually a given trajectory is desired, which imposes a t_0 and t_f; and the value of Φ is fixed either the environmental perturbations or $s_2(0)$. Since Φ_{stab} is fixed from the dynamical environment, Φ_{conv} is the only Φ up to choice. Then, the gains of the control are Λ and $\Phi_{conv} \propto 1/\Delta t_r = 1/n(t_f - t_0) \propto 1/n$ (see section 4.1). Hence, the tuning of the MSSG will be fundamentally based on the choice of

$$\Lambda \in [2,\ \infty) \tag{4.28}$$

$$n \in (0,\ 1] \tag{4.29}$$

The parametric dependence of the MSSG's performance has been tested in MATLAB , using the model described in Appendix B. To study the MSSG itself, the thrust has been left unbounded both in magnitude and direction, and the gain matrix is spherical. For the test, the initial states have been retrieved from a reference trajectory Kernel computed by Milani Mission Analysis Team[5]. This particular reference trajectory is expected to perform a close passage about D2 at approximately 18-jun-2027-20:45:00 (UTC). The target landing point has been chosen as displayed in Table 4.2 below.

[5] Part of DAER in Politecnico di Milano

Table 4.2: Initial and final states for the parametric analysis test. The landing target has been selected as DART spacecraft's crater.

	Time (UTC)	**Position**	**Velocity**
Start	18-jun-2027-18:00:30	Retrieved Ref. Trajectory	
Landing	18-jun-2027-20:45:59	[-25.45 -74.51 17.57]	[0 0 0]

The collision velocity ($V_c \triangleq \nu(t_f)$) and the fuel spent have been captured for a number of combinations of gains Λ and Reaching Times t_r, the latter specified by as the fraction of the landing time t_f with a constant n: $\Delta t_r = n(t_f - t_0)$. For this particular case, the performance of the MSSG is displayed in Figs. 4.6a and 4.6b below.

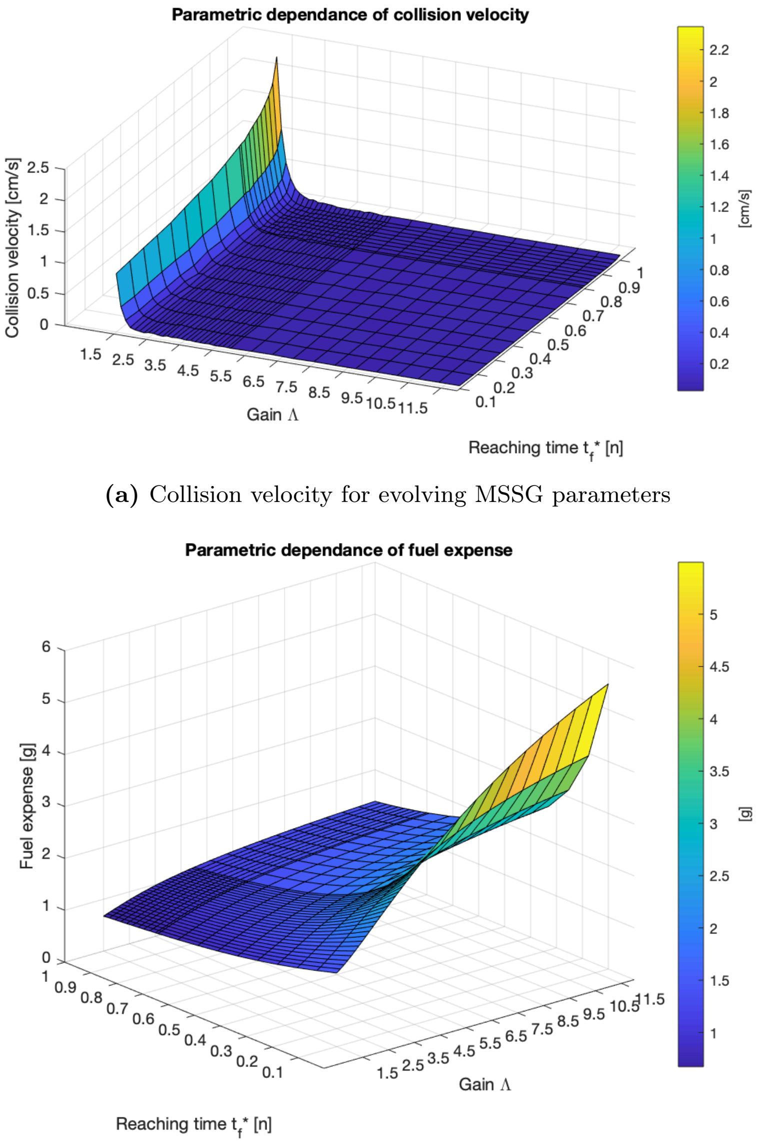
Regarding V_c, Fig. 4.6a shows that the V_c is mainly insensible to changes in Λ and t_r. Nonetheless, if $\Lambda < 2$, V_c suddenly climbs as the $\dot{s}_1$ follows Eq. (4.9) with an exponent < 1. Changes on t_r also have a minor impact on V_c, excepting when $\dot{s}$ is out of the proper convergence rate with $\Lambda < 2$. It is noteworthy that the maximum collision velocity is $< 2.5cm/s$, meaning that even with the lower performance parameters the algorithm yields an acceptable soft landing. Regarding Δm, in Fig. 4.6b can be observed how indeed higher Λ yields higher Δm and smaller t_r yields smaller Δm. Nonetheless, in contrast with the V_c dependence on Λ and t_r, is noteworthy that, the smaller is t_r, changing the parameters does have a big impact on Δm. Also, it is noteworthy that with greater t_r, Δm loses sensitivity to changes in Λ.

Selection of Λ and t_r for Continuous MSSG

The aim of this section is to discuss the selection of the gain matrix Λ and the reaching time t_r $[n]$ for continuous thrust MSSG. This is relevant as a tool to scope to understand how MSSG behaves and for Mission Analysis purposes. The Milani landing on Dimorphos imposes a set of necessities that rule the selection of the MSSG parameters. These are:

1. Necessity to keep a good landing performance in front of uncertainty
   1.1. Necessity to keep the descent velocity big enough to ensure a convergence towards $r \rightarrow r_L$ in front of perturbations. This drives for small Λ values, since the lower is Λ the less time does the trajectory shown in Fig. 4.4 spend at low relative velocities.
   1.2. Necessity to keep the collision velocity low, < 4.2 cm/s in front of perturbations. This drives for a higher Λ and a small t_r that minimize velocity error at landing.
   1.3. Necessity to keep the actual landing spot near the landing target by a defined amount. First, this drives for greater Λ and smaller t_r. Secondly, this drives for earlier firing start because the later the firing starts the greater is the uncertainty on the s/c states at that position which will degrade the thrust pointing accuracy.
   1.4. Necessity to reduce the time during which the control is switched on to avoid accumulation of thrust pointing errors. This drives for small firing time. If there is just one firing, this means a small ToF.
2. Necessity to maintain the fuel consumption low. This drives for smaller Λ and greater t_r as well as an adequate geometry of the binary system at the firing start. As mentioned in section 3.7, fuel consumption is a design parameter but landing performance is the priority.

As a preliminary approach the gain matrix Λ has been considered spherical. The Gain Matrix and reaching time have been selected as shown in Table 4.3.



(a) Collision velocity for evolving MSSG parameters

(b) Fuel expense for evolving MSSG parameters

Figure 4.6: Fuel consumption and collision velocity of the MSSG for different gains Λ and reaching times. $I_{sp} = 80\ s$

4.3.4 Effect of the Time-of-flight

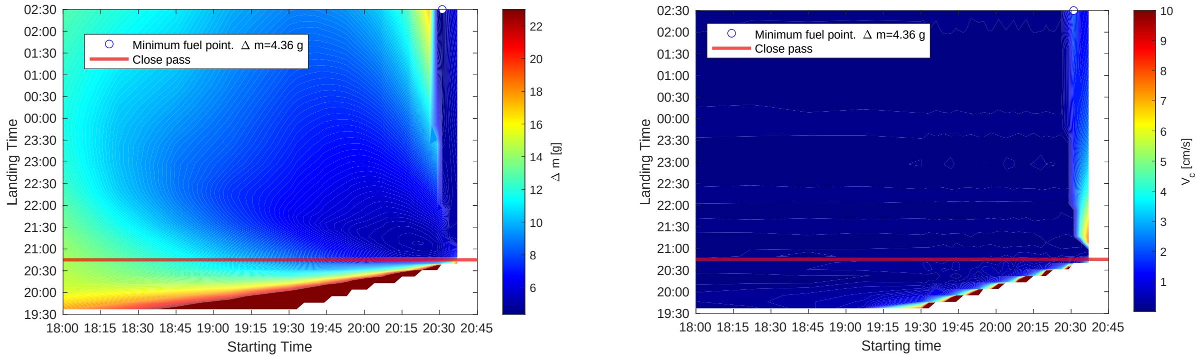
The aim of this subsection is to evaluate the influence of the Time-of-Flight (ToF) on the landing performance of the MSSG algorithm. The nominal Λ and t_r have set to $\Lambda = 2.1$ and $t_f = 0.6$ as in Table 4.3. The consumption for different combinations of Starting Times (at which the control is turned on) and Landing Times (t_f) has been surveyed. The specific impulse has been set to[6] $I_{sp} = 80\ s$. Milani is expected to approach D2 with a close passage at approximately 18-jun-2027-20:45:00 (UTC). The reference trajectory at the time of writing does not intersect D2, and passes with a significant relative velocity in front of D2. In Fig. 4.7a is displayed the mass expense for different combinations of departing times and landing times. The initial states are taken from the reference trajectory provided by Milani Mission Analysis Team and the target states in the D2 RF are the ones displayed in Table 4.2, which correspond to a point

[6] While the order of magnitude of the I_{sp} of Milani's cold thrusters can range from $40 - 120\ s$, the exact value of I_{sp} is not relevant at this stage, since the aim is to show the fuel expense evolution and not extracting particular values.

Table 4.3: MSSG parameters used for mission times analysis.

Parameter	Value	Responding to criteria	Compromising criteria
Λ	2.1	1.1; 2	1.2; 1.3
t_r $[n]$	0.6	1.2; 1.3	2

on D2's surface close to the trajectory at the closest pass.



(a) Fuel consumption for different combinations of starting times and landing times

(b) Collision velocity for different combinations of starting times and time of flights

Figure 4.7: Map of fuel consumption for different starting and landing times. $\Lambda = 2.1$ and $t_r[n] = 0.6$. $I_{sp} = 80s$ The minimum ToF computed is 1 min. The times correspond to 18-jun-2027 and 19-jun-2027 (UTC). Simulation is stopped if Milani reaches D2 surface.

In Fig. 4.7a is shown how the window for less fuel consumption is located at slightly later times than the close pass. This is because to radially descend on the landing target as imposed by the MSSG dynamics (Eq. (4.9)) at the time of close pass, the s/c needs to "search" D2 in a previous orbital location than the closest pass, for then radially descend and touch-down at the time of closest pass. This "search" on the backward orbital motion is inefficient, and it is advantageous instead to synchronise the "search" of D2 to the time similar to the close pass and then add extra time to descend on the surface. On the east side of Fig. 4.7a there is a layer of very low fuel consumption which corresponds at when the control is started at the closest passage. In that case the s/c would be made to hover for a relatively long time, and the fuel expense is greatly decreased but V_c is increased, as can be seen from the values of the same region appearing on Fig. 4.7b.

It is noteworthy that the margin to delay the landing once the control is started (the margin to extend the ToF), decreases if the starting time is closer to the close pass. Such fact drive for selecting a starting time far from the close pass. Nonetheless, greater ToF may induce to accumulated disturbances due to thrusting pointing errors and hence it would be advantageous to reduce the ToF of the controlled trajectory. Finally, in Fig. 4.7b can be seen how the collision velocity remains under 1 cm/s excepting when the control is started very close to the surface in which case the faulty convergence of $s_1 \to 0$ due to $n > 0$ is magnified.

4.3.5 Perturbations' Effect and SMC Gain Selection

The effect of perturbations has been analysed for a gain $\Lambda = 2.1$ and a $t_r = 0.6$ in order to understand what is the degradation that can be expected. To do so, a model for the perturbations along the trajectory has been introduced, explained below. To evaluate the system response to perturbations, first is needed to select a the value for Φ_{stab}.

Sliding Mode Control Gain Selection

To estimate Φ_{stab} (see Eqs. (4.20) and (4.21)) such that $\Phi_{stab} < |p_{max}|$, the maximum magnitude of the unmodeled dynamics has been estimated. To do so, the nominal case of Table 4.2 has been taken and the maximum value of the environmental accelerations has been recorded. Then, it will be assumed that

1. At least the order of magnitude of the accelerations is well modelled and hence errors in modelling are up to the 10% of the maximum acceleration. Therefore, order of magnitude of the maximum perturbation will be:

$$\mathcal{O}(p_{max}) = \mathcal{O}(|a_{max}|/10) \tag{4.30}$$

   where $|a_{max}|$ represents the maximum value of any environmental acceleration acting on the s/c.

2. The perturbations due to modelling errors are constant during the trajectory whose value follows a Gaussian distribution as:

$$p \sim \mathcal{N}\left(0\ m/s^2, [\mathcal{O}(|p_{max})|/3]^2\right) \tag{4.31}$$

   where p is the perturbing acceleration due to the poor modelling of some environmental acceleration and p_{max} its maximum value. This model accounts for an unmodeled bias in the dynamics (e.g. an error in the SRP or a constant shift of the barycenter of all gravitational forces). The fact that p is constant during all the trajectory allows the error to accumulate in a given direction, which is more conservative that allowing p to change freely. A $p = 0.1a_{max}$ is consistent with the estimation error of $3\sigma = 30\%$ in Didymos density and existing literature [7], [24].

Note that the last assumption implies that the value of p is expected to be below $|a_{max}/10|$ with a 99.7% of probability. The considered perturbations are described below and their maximum estimated value is shown in Table 4.4.

- Solar Tide: the Solar gravity has not been modeled because it holds minor importance in close operations [25]. Its value has computed using the spherical gravity model.
- Non-inertial effects: the computation of the acceleration in the D2 RF requires knowledge on its rotation state which might be poor.
- Higher resolution gravity fields: D2 is modelled using spherical harmonics and D1 as a sphere and not shape-models.
- Solar Radiation Pressure: the effect of the SRP might be poorly estimated from errors on attitude determination and modelling of the s/c surfaces' reflective properties.

Therefore, following the first assumption mentioned above and with the data of Table 4.4:

$$\mathcal{O}(p_{max}) = 10^{-5}\ m/s^2 \tag{4.32}$$

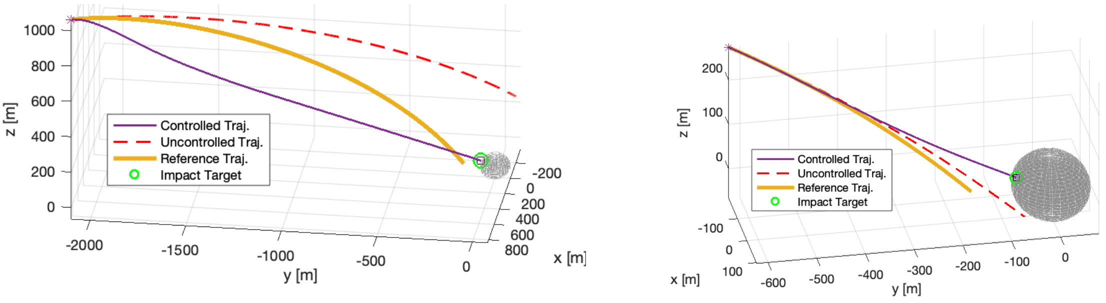
Fig. 4.8 below shows the effect of the such perturbations . There, the environment was perturbed by an instance of $\mathcal{N}(0, |p_{max}|^2)$. To protect the system from diverging while facing perturbations of $\mathcal{O}(p_{max}) = 10^{-5}\ m/s^2$ (e.g. $2 \cdot 10^{-5}, 9.9 \cdot 10^{-5}$...), the Sliding Mode Control gain Φ_{stab} will be set to:

$$\Phi_{stab} = 10^{-4}\ m/s^2 \tag{4.33}$$

and indeed with the 99.7% of probability in the Gaussian distribution, $|p| < 10^{-4}\ m/s^2$.

Table 4.4: Maximum values of environmental accelerations on the s/c. The accelerations have been retrieved from a simulation along a reference trajectory using the model proposed in Appendix B.

Perturbation	Magnitude [m/s/s]	Description
Solar Tide	10^{-9} [25]	Spherical gravity of the Sun on the s/c
Non-inertial RF effects	10^{-5}	Term $2\omega \times \nu + \omega \times \omega \times r + \ddot{R}_a$ in the dynamics with respect to the D2 RF.
Higher resolution gravity field	D1: 10^{-4} and D2: 10^{-4}	Ellipsoidal gravity pull on the s/c by D2. Near D2 its gravity primes over D1's.
SRP	10^{-8}	SRP accelerations affecting the s/c translational motion



(a) Long time of flight, greater dispersion

(b) Short time of flight, greater dispersion

Figure 4.8: Comparison between uncontrolled, unperturbed reference trajectory (yellow), uncontrolled perturbed trajectory (red), and controlled perturbed trajectory (violet). $\mathcal{O}(p_{max}) = 10^{-5}\ m/s^2$. and $p = ct. \approx [1.8,\ 6.1,\ -7.5] \cdot 10^{-6}\ m/s^2$.

Effect of Perturbations - Soft case

In this section it is investigated the effect of a soft perturbation on the performance of the control with the nominal Φ value, this is:

$$\mathcal{O}(|p_{max}|) = 10^{-5}\ m/s^2, \quad \Phi_{stab} = 10^{-4} > \mathcal{O}(|p_{max}|) \tag{4.34}$$

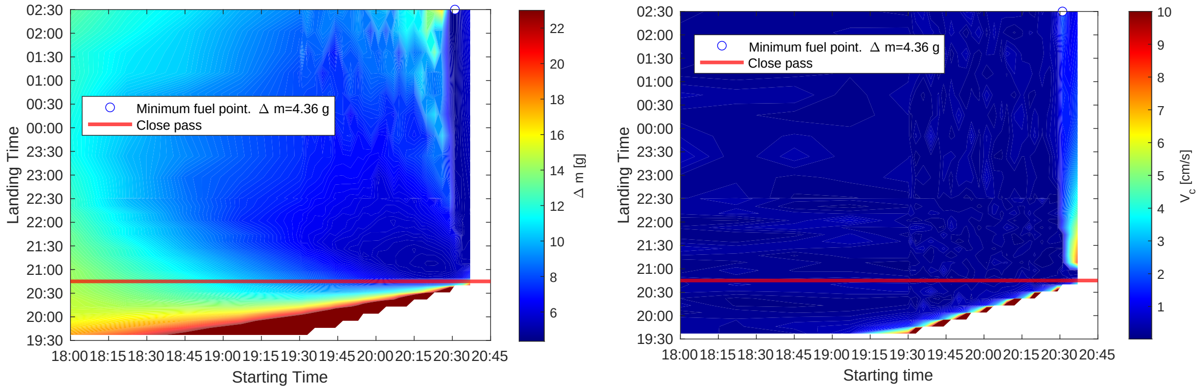
The fuel consumption and collision velocity for different combinations of control-starting epochs and landing epochs has been evaluated. In Figs. 4.9a and 4.9b below can be seen how changes in performance for different control start epochs and landing epochs with respect to the unperturbed performances (see Fig. 4.7a) are small. This was expected since the thrust is limited to $\approx 1mm/s^2$, which is higher than a_{max} and p_{max}. Hence, with enough time to converge, the thruster may always bring $s_2 \rightarrow 0$.

Effect of Perturbations - Extreme case

The aim of this section is to show how the nominal control of $\Phi_{stab} = 10^{-4}$ can deal with perturbations 10 times as big as the ones considered nominally, this is:

$$\mathcal{O}(|p_{max,extreme|}) = 10\mathcal{O}(|p_{max}|) = 10^{-4}\ m/s^2 \tag{4.35}$$

The fuel expense and collision velocity for different combinations of starting times and landing times have been evaluated, in which the environment was perturbed by an instance of $\mathcal{N}(0, |10p_{max}|^2)$. In Fig. 4.10a can be observed how the minimum consumption point is remarkably similar to that of Fig. 4.7a. Nonetheless, in the perturbed dynamics case the fuel expense increases far quicker moving away from the lobe of small fuel expense that starts after the close passage time. It is important to note that even if in the unperturbed case, greater ToF were advantageous as they didn't punish fuel expense but provided greater time margin for the

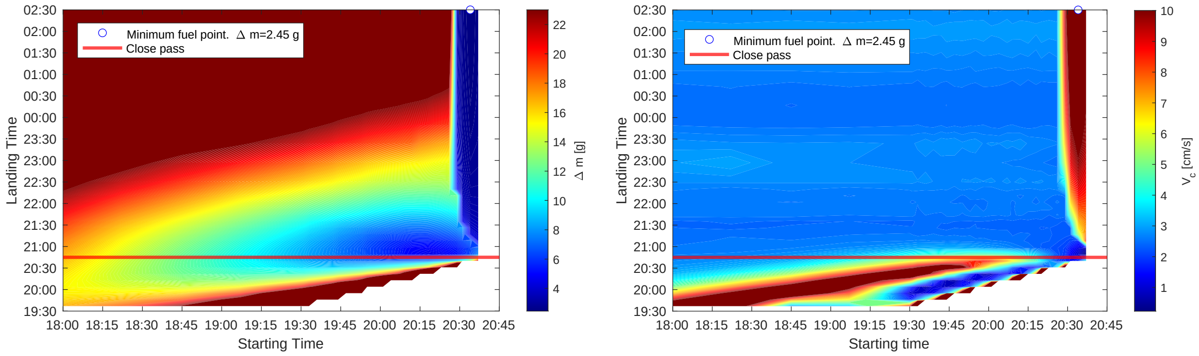


(a) Fuel consumption for different combinations of starting times and landing times

(b) Collision velocity for different combinations of starting times and time of flights

Figure 4.9: Map of fuel consumption for different starting and landing times for a nominal dynamically perturbed environment, where $p(t) \sim \mathcal{N}(0, [\mathcal{O}(|p_{max})|/3]^2)$. $\Lambda = 2.1$ and $t_r[n] = 0.6$. The minimum ToF computed is 1 min. The times correspond to 18-jun-2027 and 19-jun-2027 (UTC).

landing, in the case at hand the perturbations during a longer flight force the control to exert a greater effort, and in Fig. 4.10a can be observed how only departures near to the closing-pass time are fuel efficient. The collision velocity on different starting/finish times is displayed on Fig. 4.10b. In comparison to the unperturbed case of Fig. 4.7b, it is important to note that the extreme perturbation mainly increases the collision velocity for trajectories whose landing time is below or at the time of short passing. I.e. the extreme perturbations punish the trajectories in which the s/c is going "backwards" to "search" for D2. Nonetheless, for trajectories with greater ToF and if the landing time is greater than the close pass time, V_c remains acceptably small (approx $< 4cm/s$). In conclusion, even if under the effects of nominal perturbations are small, lower ToF and landing epochs set later than the close passage would provide further shielding towards extreme perturbations.



(a) Fuel consumption for different combinations of starting times and landing times

(b) Collision velocity for different combinations of starting times and time of flights

Figure 4.10: Map of fuel consumption for different starting and landing times for an extreme dynamically perturbed environment, where $p = ct. \sim \mathcal{N}(0, [10\mathcal{O}(|p_{max})|/3]^2)$. $\Lambda = 2.1$ and $t_r[n] = 0.6$. The minimum ToF computed is 1 min. The times correspond to 18-jun-2027 and 19-jun-2027 (UTC).

The bottom line result is that trajectories should target the blue lobe of small fuel consumption after the close passage, which requires a $ToF \gg 300s$, the current limit of firing time for Milani. Hence, a necessity to perform longer-than $300s$ flights appear, which will be tackled in the following chapters.

Chapter 5

Firing Strategy for Descent & Landing

In this chapter I introduce the design of a firing strategy for Milani CubeSat's Landing. The control algorithm is the MSSG, implemented as is shown in Appendix B. At the time of writing, the landing of Milani CubeSat is not critical to mission success and the GNC system has not been designed to optimize for descent and landing requirements. In particular, Milani is expected to perform only discrete ($300s$) firings. Nonetheless, the requirements of the landing phase (also known as "experimental phase") are still under consideration and might change. Hence it cannot be excluded that continuous thrust would be finally adopted. For this reason, in the choice of the MSSG, I first assume that continuous thrust is available. Next, with the aim to minimize the impact of the proposed MSSG on the overall current GNC design, I have developed a firing strategy that uses $300s$ firings for $\approx 96\%$ of the approach towards the asteroid. To ensure a successful and robust landing, during the last $15m$, the control has been allowed to remain open, if needed, for more than $300s$. In this chapter is presented such firing strategy.

5.1 The Two Phased Descent Strategy

From the study in Chapter 4 it is understood that the role of the MSSG is to bring s_2 to 0 and keep it thereafter. In general, if the thrust is 0, s_2 will tend to diverge from 0 and the s/c will no longer be in a sliding mode that brings $s_1 = r(t) - r_d$ and $\dot{s}_1 = \nu(t)$ to 0 in FT. Therefore, if the control is applied discontinuously, the more frequently it is applied, the more will the performance resemble that of the continuous MSSG performance (also called "ideal performance" hereafter). The firing strategy shall be based on the following needs:

1. The soft-landing needs a high performance (i.e. high accuracy) of the control system, which can be achieved only during the periods in which the control is turned on.

2. The continuous thrust periods are desired to be $\leq 300s$, but the controlled trajectories that i) spend less fuel and ii) are far from the ToF zones with high V_c risk, have time-of-flights of at least $\approx 3600s$ (see Figs. 5.5a and 5.5b below [1]).

3. The strategy needs to be suitable for different starting epochs and landing epochs.

From these needs, a 2-phase approach has been selected, in which first the s/c approaches D2 with a discontinuous thrust and below a certain height the s/c performs a fully controlled $300s$ descent. This strategy will be hereafter called Two Phased Descent Strategy (TPD) and is explained below. Let there be a boundary layer around the surface of D2 of a given height h $[m]$ above its surface (an example with $h = 10m$ is shown in Fig. 5.1). Then, the landing trajectory is divided in two phases. In each phase the control uses a different set of Λ and $\Phi_{conv} \propto 1/n$. The two phases are:

[1] This figure was already displayed and is re-presented here for clarity's sake. See Fig. 4.10a, Fig. 4.10b in section 4.3.

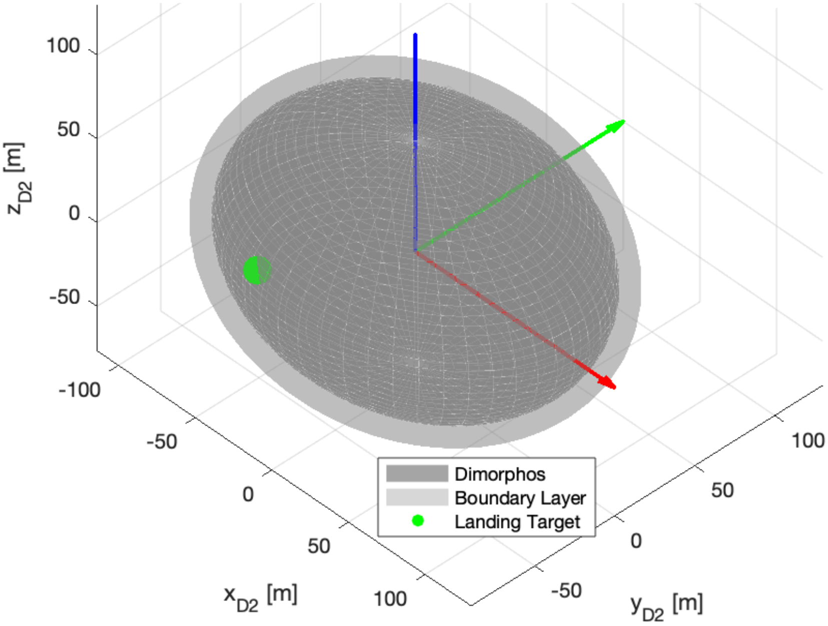


Figure 5.1: Dimorphos surrounded by a boundary layer at $10m$ from its surface. The red, green and blue arrows respectively represent the x, y and z axis of the D2 RF.

I. Approach Phase

The s/c targets the landing point $(r_L,\ \nu_L)$ at a desired time $t^*_{landing}$ and the control is turned on and off sequentially with a period of $T_{on/off} = 300s$ (see Fig. 5.2):

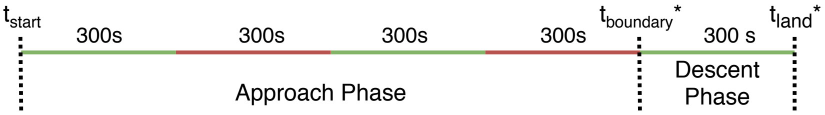


Figure 5.2: Planned sequence of the Approach Phase and Descent Phase at t_{start}. Red: control off. Green: control on. The number of displayed segments before $t^*_{boundary}$ is random.

Since $\Delta t_r = n_1 \cdot 300s$, typically $\Phi_{conv} \gg \Phi_{stab}$. For this reason, in the TPD strategy, Φ typically coincides with Φ_{conv}. At the start of each firing, Φ_{conv} is recomputed to ensure that the system brings s_2 to 0 during the firing time:

$$\Phi_{conv,t_{on}} = \frac{s_2(t_{on})}{n_1 \cdot 300s} \tag{5.1}$$

The Approach Phase finishes when the s/c reaches the boundary layer surrounding D2. The time that the s/c has to reach the boundary layer is constructed as follows. Given a desired landing time $t^*_{landing}$, the s/c shall reach the boundary layer 300s earlier (see Fig. 5.2):

$$t^*_{boundary} = t^*_{landing} - 300s \tag{5.2}$$

Given a certain starting epoch t_{start} the time stamps at which the control is to be turned on and off are computed as follows:

$$\{t_k\}_{k=0}^{N} \text{ where } t_k = \begin{cases} t_0 & k = 0 \\ t_{k-1} + 300s & t_k \leq t^*_{boundary} \end{cases} \tag{5.3}$$

The control is supposed to start at t_{start}, and the s/c shall reach the boundary layer with the thruster off to be able to perform a powered descent the subsequent 300s. It can be easily shown that this happens if there is an odd number of control time points, which an be ensured by adding another point of +300s if the length N of the sequence above is even.

II. Descent Phase

After reaching the boundary layer at a height h, the s/c performs a $300s$ powered descent towards the landing point. The actual time at which the s/c reaches the boundary layer is registered as $t_{boundary}$ and the new desired landing time at which the s/c is to land using the MSSG is defined as (see Fig. 5.3) :

$$t^{**}_{landing} = t_{boundary} + 300s \tag{5.4}$$

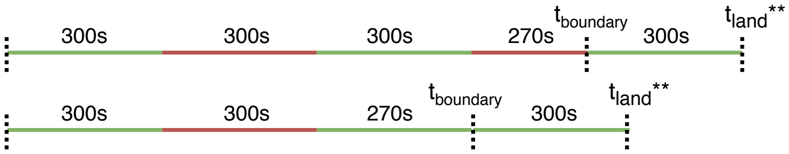


Figure 5.3: The s/c reaches the boundary at $t_{boundary}$, then targets t^{**}_{land}. The boundary can be reached with the control off (up) or on (down). Red: control off. Green: control on.

Note that the landing time $t^{**}_{landing}$ (or t_f) won't be equal to the one specified as reference in Eq. (5.2). Importantly, the perturbations and non-ideal application of the MSSG will make the s/c touch down at a time different than $t^{**}_{landing}$, probably earlier. The actual time of touchdown is $t_{landing}$ (see Fig. 5.4):

$$t_{landing} \neq t^{**}_{landing} \neq t^{*}_{landing} \tag{5.5}$$

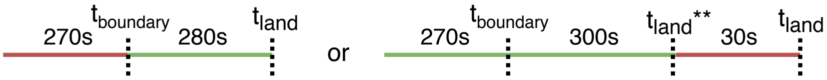


Figure 5.4: The s/c touches the surface at t_{land}. The surface can be reached before $t^{**}_{land} \equiv t_f$ (left) or after (right). Red: control off. Green: control on.

5.2 Comments & Observations on the TPD

Importantly, the landing performance of the control in the TPD depends on:

- The set of parameters with which the control is tuned during the Approach Phase:

$$\Lambda_1 \in [2,\ \infty) \tag{5.6}$$
$$n_1 \in (0, 1] \tag{5.7}$$

- The height of the boundary layer, which imposes the epoch at which the control has to start final descent. Hence it partially determines which are the initial states of the final descent for a given performance of the Approach Phase.

$$h \in (0, \infty)\ m \tag{5.8}$$

- The set of parameters with which the control is tuned during the Descent Phase:

$$\Lambda_2 \in [2,\ \infty) \tag{5.9}$$
$$n_2 \in (0, 1] \tag{5.10}$$

Due to the degradation of control authority during the Approach Phase, it is possible that the s/c reaches the boundary layer while having the control turned on. In such case, there are 3 possible control decisions that could have been taken:

1. The s/c continues firing for the rest of the $300s$, then stops firing. At the end of the firing the s/c will be close to the surface of D2 and it will free-fall to the surface. The Descent Phase is half-powered.

2. The s/c stops firing, drifts for $300s$, and starts the Descent Phase by firing $300s$ more.

3. The s/c starts a new firing batch of $300s$ without stopping, thus transgressing the requirement of $300s$ firings.

Option 1 has been discarded due to the uncertainty on when the firing would stop: depending on the control's performance, it could happen that the boundary layer was reached at $t = 1s$ of a firing, thus still having $299s$ left to actuate; or at $t = 299s$, and hence preventing any control action after $1s$. This would imply a widely varying performance of the control during the Descent Phase, which would make the scheme non-robust. Option 2 has been discarded for two reasons: first, because the s/c reaches the boundary layer with a non-zero velocity, and turning off the control at a height h would imply to descent with an uncontrolled speed close to the surface. Second, because due to uncertainty in position, the s/c could target a point set at varying altitudes from D2, again impairing robustness. Therefore, Option 3 has been retained, and I have have considered that at such range from the surface it would be in the interest of the mission to transgress the desire for maximum $300s$ firings.

Due to the degradation of control authority during all the flight, it is possible that

1. The control reaches t_f before touch-down and stops firing.

2. The control touches-down before t_f

In case 1, the s/c will start a free-fall from within the boundary layer and it has been observed to have a V_c of $\mathcal{O}(1 - 5cm/s)$. Because the s/c would perform a free fall, it is expected that its trajectory won't have enough mechanical energy to escape D2. In case 2, it is expected that the s/c will reach the surface with V_c of $\mathcal{O}(1 - 5cm/s)$, because during $t_{go} < 50s$ the velocity of descent is already small.

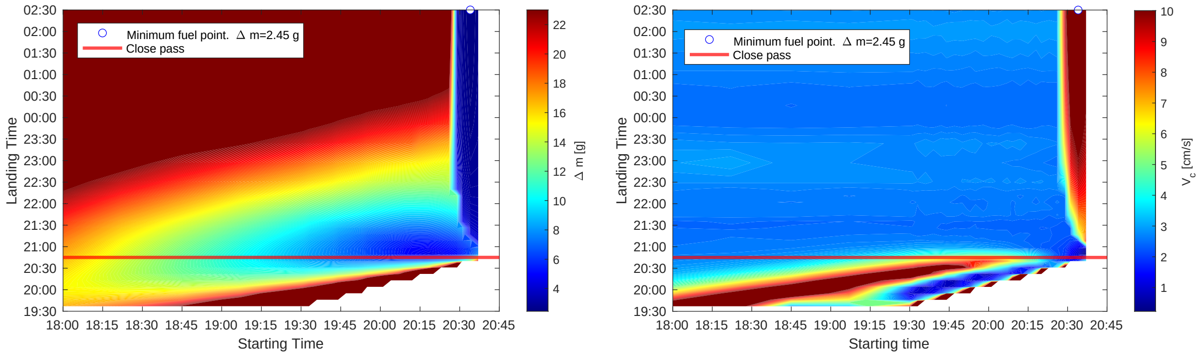
5.3 Control Tuning and Boundary Height Selection

With the TPD, the s/c can perform Approach Phases whose total Time of Flight is much greater than $300s$. As a conservative approach, the extreme perturbation case of section 4.3 can be considered. Indeed, suiting trajectories are those located in the blue lobe of Fig. 5.5a, as discussed previously. The farther the landing time from the *close pass* line, the farther will the trajectory be from the high Δm, V_c zones in the bottom-right of Figs. 5.5a and 5.5b. Simultaneously, greater-than-necessary ToF should be avoided to keep thrust pointing errors and other state estimation errors from accumulating.

The gains $\Lambda_{1,2}$, the reaching times $n_{1,2}$, and an adequate boundary layer h, have been selected by optimising the performance in a nominal trajectory, extracted from Figs. 5.5a and 5.5b. The selected trajectory has the epochs shown in Table 5.1.

Table 5.1: Start and landing times selected for the nominal trajectory. With this starting epoch the control starts at $\approx 400\ m$ of D2.

Start time (t_{start})	Desired Landing time ($t^*_{landing}$)
18-jun-2027-20:15:00 (UTC)	18-jun-2027-21:15:00 (UTC)



(a) Fuel consumption for different combinations of starting times and landing times

(b) Collision velocity for different combinations of starting times and time of flights

Figure 5.5: Map of fuel consumption for different starting and landing times for an extreme dynamically perturbed environment, where $p = ct. \sim \mathcal{N}(0, [10\mathcal{O}(|p_{max})|/3]^2)$. $\Lambda = 2.1$ and $t_r[n] = 0.6$. The minimum ToF computed is 1 min. The times correspond to 18-jun-2027 and 19-jun-2027 (UTC).

If the thrusting was continuous for the whole flight, they would leave the s/c around the minimum point of the blue lobe in Fig. 5.5a, while still being far from the east layer of high V_c appearing in Fig. 5.5b. The landing target is DART spacecraft's crater, extracted from Hera's Kernels[2]. The uncontrolled trajectory and the unbounded continuous-thrust controlled trajectory are displayed in Figs. 5.6a and 5.6b. The performance of this landing is reported in Table 5.2 . At the time of writing the Kernels provided by Hera's data repository generate a DART landing point that is some *cm* below the surface defined by the ellipsoid whose radii are given by ESA's Kernels. r_{err} is small enough so that it can be attributed to the kernel's distance and thus it can be considered negligible.

Table 5.2: Nominal trajectory landing performance for the unbounded, continuous thrust profile and $\Lambda = 2.1$, $n = 0.6$; targeting DART spacecraft's crater.

ToF $[s]$	Δm $[g]$	$\lvert V_c \rvert$ $[cm/s]$	$\lvert r_{err}(t_c) \rvert$ $[cm]$
3600	4.93	0.38	16.6

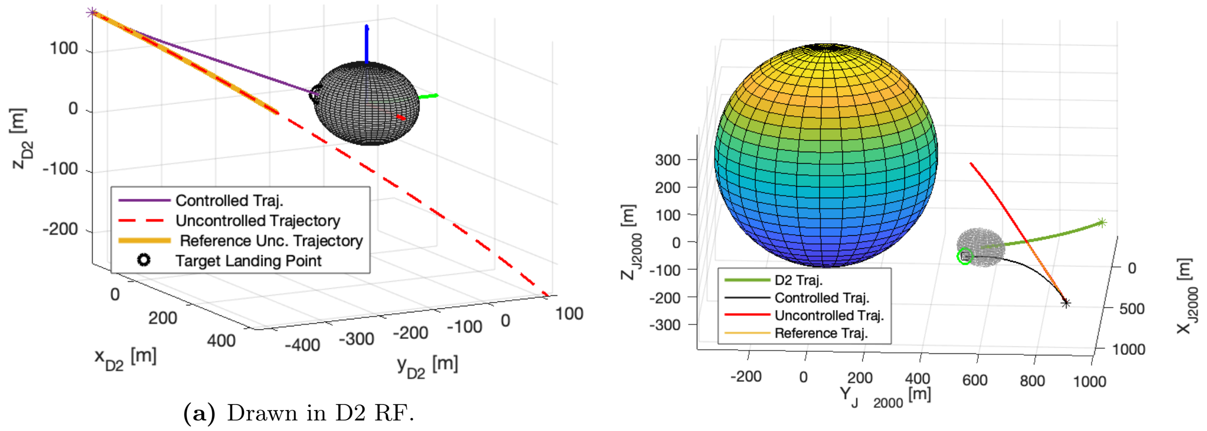
Then, the control will be tuned such that it has a good performance on the nominal trajectory and its robustness to changes on the start/landing epochs will be evaluated *ex post*. The performance has been scoped for different combinations of $\Lambda_{1,2}$, $n_{1,2}$ and h in an unperturbed environment. To simplify the analysis, I performed an iterative process, shown in Fig. 5.7. Its final iteration will be presented here. Indeed, Figs. 5.8a and 5.8b below respectively shown how V_c and Δm change with the Descent Phase parameters; while having fixed a reasonable control performance for the Approach Phase: $\Lambda_1 = 2.5$ and $n_1 = 0.2$. There, it can be seen how

1. V_c is relatively insensitive to changes on Λ_2, n_2 and h if $n_2 \in (0.2,\ 0.9]$, which would drive to select $n_2 > 0.2$. If $n_2 < 0.2$ and $\Lambda_2 < 3$, the collision velocity ramps up dramatically.

2. Δm increases monotonically with Λ_2, and at each combination of Λ_2 and height h, the smaller is n_2 the smaller is Δm. This would drive towards selecting a small Λ_2 and a small n_2

3. The height of the boundary layer plays a relatively small role, and is only significantly impacting the landing performance if $n_2 \approx 0.1$ and $\Lambda_2 < 3$. For several combinations of Λ_2 and n_2 I have observed that

[2] ESA SPICE Service, Hera Working Groups, Instrument Teams. *Hera Spice Kernel Set*. 2020. Last Accessed: aug-2021. [online]. Available: `https://www.cosmos.esa.int/web/spice/spice-for-hera`



(a) Drawn in D2 RF.

(b) Drawn in N RF.

Figure 5.6: Nominal uncontrolled, and unbounded continuous-thrust controlled trajectories. $\Lambda = 2.1$, $n = 0.6$. D1 is colored and D2 is grey. D1 and D2 are drawn at the landing time. Environment is unperturbed.

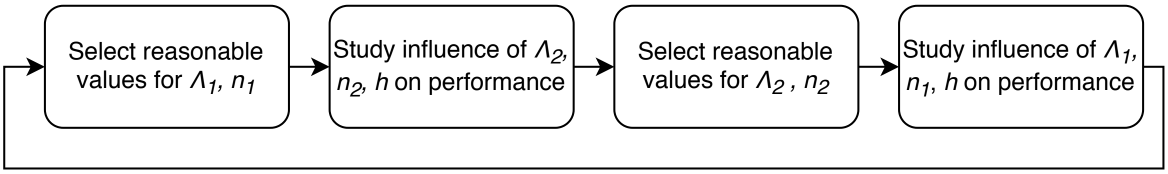

Select reasonable values for Λ_1, n_1
Study influence of Λ_2, n_2, h on performance
Select reasonable values for Λ_2 , n_2
Study influence of Λ_1, n_1, h on performance

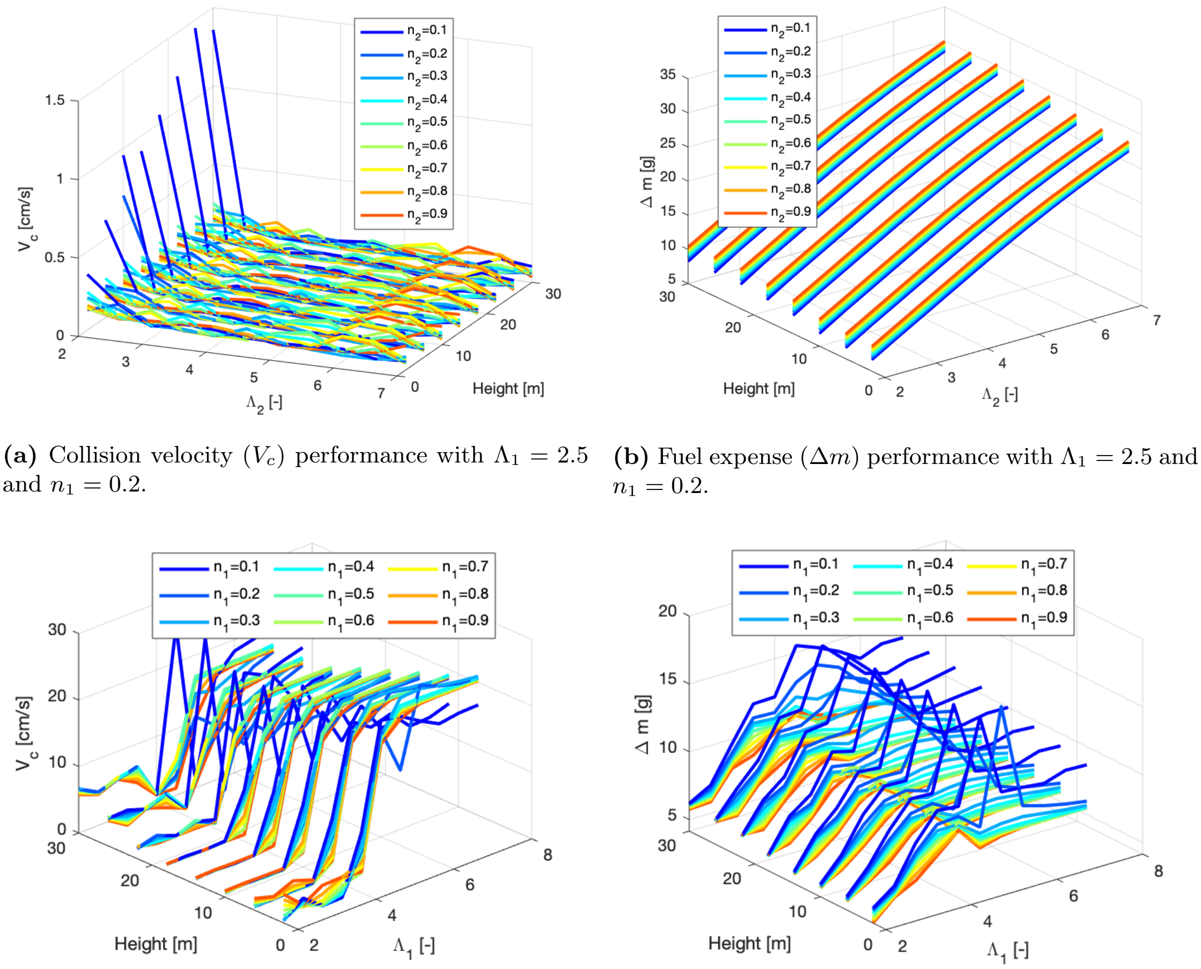
Figure 5.7: Iterative process for the study of effect of Λ_1, n_1, Λ_2, n_2 and h on the landing performance.

- A boundary layer that is too large decreases the performance because the control needs to accelerate the s/c to reach the surface in $300s$ and then it does not have time to break the speed again.
- A boundary layer that is too small decreases the performance because the control has too small amount of time to reach the sliding mode and then, inside the sliding mode, converge to $s_1 = 0$.

For the reasons stated above, the Descent Phase gains are chosen so that the control is far from the zones of risk (i.e. $n_2 \gg 0.1$) while keeping the fuel expense low ($\Lambda_1 \downarrow$). Hence, the Descent Phase gains have been set to $\Lambda_2 = 3$ and $n_2 = 0.7$. The height h is chosen to be $15m$, which I observed to generate a suitable landing performance with the set of gains that are being selected. The parameters are shown in Table 5.3.

In the other hand, Fig. 5.8c and Fig. 5.8d respectively show the influence of the Approach Phase gains Λ_1 and n_1 on the collision velocity V_c and the fuel expense Δm. Both of them shown a great sensitivity of the landing performance to Λ_1 and n_1. In particular, can be observed that:

1. From V_c performance, Λ_1 shall be $\Lambda_1 \in [2, 4)$, while the role of n_1 should be $n_1 > 0.1$.
2. From Δm performance, Λ_1 should be as small as possible inside $\Lambda_1 \in [2, 4)$. n_1 can be chosen as big as possible.
3. Greater h has a negative impact on V_c, and the performance is maximized if $h \in (5,\ 20)\ m$.



(a) Collision velocity (V_c) performance with $\Lambda_1 = 2.5$ and $n_1 = 0.2$.

(b) Fuel expense (Δm) performance with $\Lambda_1 = 2.5$ and $n_1 = 0.2$.

(c) Collision velocity (V_c) performance with $\Lambda_2 = 3$ and $n_2 = 0.7$.

(d) Fuel expense (Δm) performance with $\Lambda_2 = 3$ and $n_2 = 0.7$.

Figure 5.8: Landing performance for different values of Λ_2, n_2, Λ_1, n_1 and h. The firings are limited to $300s$, the environment is unperturbed and the thrust is bounded and quantized.

For the reasons stated above the gain Λ_1 has been set to $\Lambda_1 = 2.5$. For a boundary layer height $h = 15m$, the role of n_1 is non-important and has been set to a relatively low value of $n_1 = 0.2$ because it ensures that the s/c reaches the sliding mode earlier and hence the information on what is the shape of the trajectory, which is defined in the sliding mode, is richer. The parameters are shown in Table 5.3.

Table 5.3: Selected parameters for the control of Milani's landing.

h $[m]$	Λ_1	n_1	Λ_2	n_2
15	2.5	0.2	3	0.7

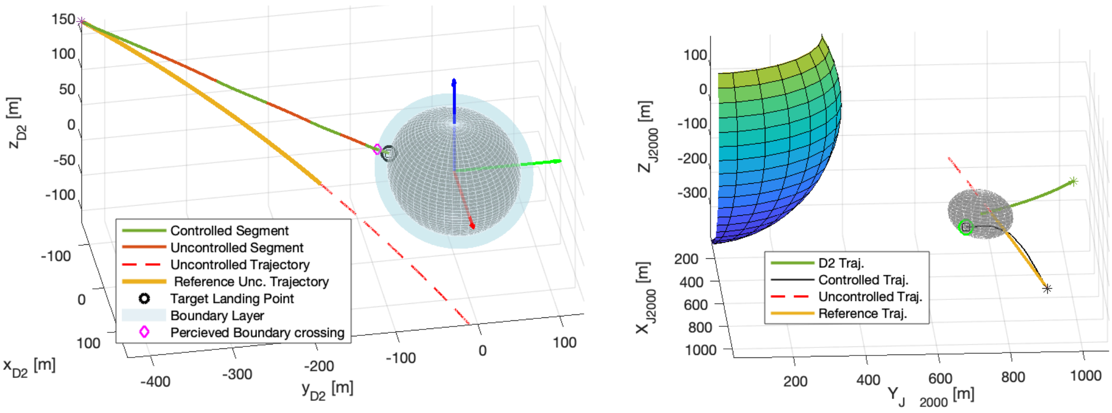
An instance of the nominal trajectory[3] performed with the TPD is shown in Figs. 5.9a and 5.9b. A zoom on its Descent Phase is displayed in Fig. 5.10. In this trajectory, the environment is unperturbed, the states are perfectly known; and the thrust is bounded and discontinuous. The performance of that landing is shown Table 5.4, which shows that is remarkably good in terms of V_c and Δm. In this instance the s/c landed at $t_{go}(t) = t_f - t \equiv t_{landing} - t = 53.3\ s > 0$,

[3] Start and landing epochs as shown in Table 5.1.

meaning that the Descent phase was completed in 244 s, and hence was fully controlled (see in Fig. 5.10 how the controlled segment reaches the surface).

Table 5.4: Performance of a nominal trajectory for the Two Phased Descent strategy. States are assumed to be known perfectly. Thrust is discontinuous and bounded. Environment is unperturbed.

ToF $[s]$	Δm $[g]$	$\|V_c\|$ $[cm/s]$	$\|r_{err}(t_c)\|$ $[cm]$
2315.2	8.42	0.73	12.1



(a) Drawn in D2 RF.

(b) Drawn in N RF.

Figure 5.9: Instance of the nominal trajectory for the Two Phased Descent strategy. States are assumed to be known perfectly. Thrust is discontinuous and bounded. Environment is unperturbed. The control parameters are those shown in Table 5.3. D1 is colored and D2 is grey. D1 and D2 are drawn at the landing time.

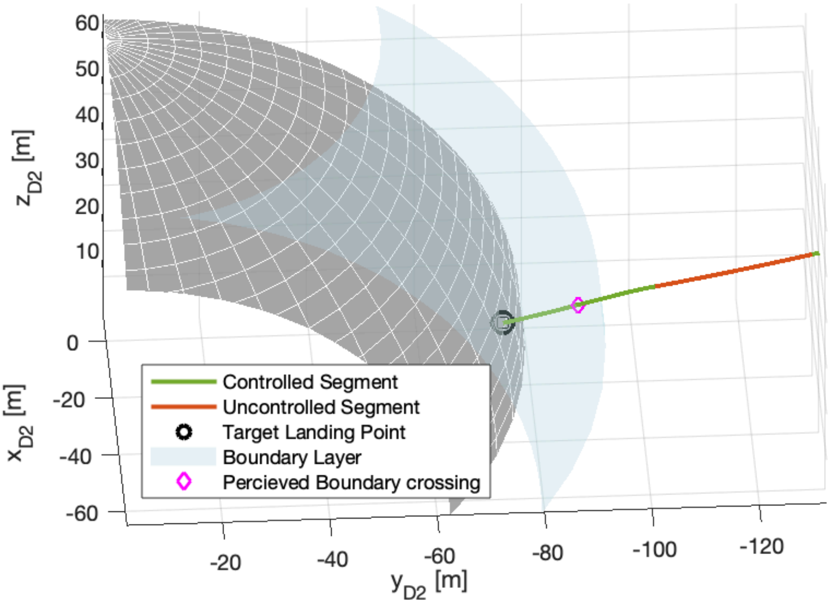


Figure 5.10: Zoom on the Descent Phase of the trajectory shown in Fig. 5.9a. Drawn in the D2 RF.

Chapter 6

Evaluation of the Two Phased Descent Strategy

In this chapter I evaluate the performance of the Two Phased Descent Strategy applied to Milani's landing. This has been done with a Monte Carlo campaign, in which the initial conditions, the state knowledge, the dynamical environment, the pointing accuracy, and the I_{sp} have been noised with Gaussian processes. The chapter starts by presenting the model adopted to degrade state knowledge and introduce dispersion in the initial conditions; then follows by presenting an instance of the nominal trajectory with navigation errors; and finally presents the statistical results of the Monte Carlo campaign.

6.1 State knowledge and initial states' dispersion

In this section is defined what is the error in state estimation during the TPD and the dispersion on the initial states at the start of the TPD. To do so, an effort has been made to use the data provided by the Milani Mission Analysis Team (available in [10] and in Milani's Mission Analysis Report (MAR) [26]). Further assumptions (detailed below) have been made in cases where the data in MAR was insufficient or incompatible with the use of the TPD.

6.1.1 State Knowledge Estimation & Implementation

State Knowledge Error Estimation

The state knowledge has been estimated using the data provided in the Milani's Mission Analysis Report [26] in the form of the 1σ value in the $(r,\ \nu)$ estimation error. The error, which increases in time, will be computed from a starting value at the start of the trajectory (t_0). It has been assumed that:

- The uncertainty on the velocity wrt. the N RF (v) is the same as wrt. the D2 RF (ν). This is reasonable because:

  $$Var[v] = Var[\nu + \omega_{D2/N} \times r]$$

  Considering $\omega_{D2/N}$ as a deterministic variable, and taking into account that $\mathcal{O}(|\omega_{D2/N}|) = 10^{-4}\ rad/s$:

  $$Var[v] \approx Var[\nu] + (10^{-4})^2 Var[r] \approx Var[\nu] \text{ and } \ \sigma_v = \sqrt{Var[v]} = \sqrt{Var[\nu]} = \sigma_\nu$$

- The σ estimation error in [26] refers to the magnitude of each component of r and ν and hence, in the implementation, each component of r and ν will be noised by σ_r and σ_ν.

From the second assumption:

$$|\tilde{r}_i| \sim \mathcal{N}(|r_i|, \sigma_r^2) \tag{6.1}$$

and similarly for ν. The evolution of σ during the trajectories without measurements increases linearly in time. From [26] it has been estimated that:

$$\sigma_r(t) = (50\ m/d) \cdot (t - t_0) + \sigma_r(t_0)\ [m] \tag{6.2}$$

$$\sigma_\nu(t) = (10^{-3} m/s/d) \cdot (t - t_0) + \sigma_\nu(t_0)\ [m/s] \tag{6.3}$$

The value of $\sigma_r(t_0)$ has been computed assuming that at the start of the control one LIDAR measurement on D2 is done, whose accuracy is [26]:

$$\sigma_r(t_0) = \sigma_{LIDAR} = \sigma_{sensor} = 1\ m \tag{6.4}$$

See Appendix C for an adaptation of the TPD strategy for $\sigma_{LIDAR} = \sqrt{\sigma_{sensor}^2 + \sigma_{shape}^2} = 10.05\ m$ and a discussion on the impact of σ_r. The value $\sigma_\nu(t_0)$ is computed by propagating $\sigma_\nu(t)$ from the Landing Orbit Acquisition Maneuver (LOAM) until t_0:

$$\sigma_\nu(t_0) = (10^{-3} m/day) \cdot (t_0 - t_{LOAM}) + \sigma_\nu(t_{LOAM})\ [m/s] \tag{6.5}$$

where $\sigma_\nu(t_{LOAM}) = 2 \cdot 10^{-4}\ m/s$ and t_{LOAM} has been taken as 17-jun-2027-22:30:00 (UTC). Hence:

$$\sigma_\nu(t_0) = 1.12\ mm/s \tag{6.6}$$

which is about 1% of the actual velocity at t_0.

State Knowledge Error Implementation

The on-board Navigation filter (i.e. the Extended Kalman Filter) is supposed to compute a set of positions and velocities that are consistent with a trajectory biased from the real one. This can be refereed to the *perceived trajectory.* The perceived trajectory will be separated by an error defined by $\sigma_{r,\nu}$ from the real trajectory. Hence, to model the Navigation error it has been assumed that:

- At the start of the controlled trajectory (t_0), the on-board Navigation filter is able to compute the real trajectory with a bias defined by $\sigma_{r,\nu}$. This is:

$$|\Delta\tilde{r}_i| \sim \mathcal{N}(0, \sigma_r(t_0)^2) \tag{6.7}$$

$$|\Delta\tilde{\nu}_i| \sim \mathcal{N}(0, \sigma_\nu(t_0)^2) \tag{6.8}$$

  Thus, during all the trajectory, the geometry perceived by the acceleration command algorithm will be:

$$\tilde{r}_i(t) = r_i(t) + \Delta\tilde{r}_i \tag{6.9}$$

$$\tilde{\nu}_i(t) = \nu_i(t) + \Delta\tilde{\nu}_i \tag{6.10}$$

- The navigation filter keeps that same bias during all the trajectory.
- The bias is also applied on computation of the relative position to the boundary layer.

6.1.2 Uncertainty On Initial States

The uncertainty during the Landing Orbit is not provided by Milani Mission Analysis Team at the time of writing (see [26]), and hence an assumption has to be performed to determine the position and landing dispersion at t_0. The σ values for the components of r and ν have been taken as:

$$\sigma_r = \frac{40}{3}\ m \tag{6.11}$$

$$\sigma_\nu = \frac{1}{3}\ cm/s \tag{6.12}$$

which with 99.7% of probability will yield errors up to $\approx 10\%$ from the nominal trajectory at t_0 and are consistent with existing literature [24]. The assumption of an error consisting of the relatively small 10% fraction is reasonable particularly for ν, since an error of $\rightarrow 100\%$ of the nominal value would mean that the trajectory could be pointing in any random direction, which is unreasonable. The 10% error in position might be optimistic, but the strong control authority of the MSSG can bring $r \rightarrow r_L$ from a wide range of positions and hence the this uncertainty is not critical to the study. The limiting factor in σ_r is that the s/c must not start the control too close to the actual surface of D2, for the TPD strategy needs time ($ToF \gg 300\ s$) to converge.

6.1.3 Instance of the Nominal Trajectory with the Noised State knowledge and Dispersed Initial Values

Here is presented an instance of the nominal trajectory (start and landing epochs set as in Table 5.1), in which the navigation information has been degraded with the model presented above. The landing performance is in Table 6.1. There it can be seen how the collision velocity is very small but the landing error is 10 times bigger than for the case with perfect state knowledge (see Table 5.4). The main reason is that in this particular instance the position navigation bias was:

$$\Delta\tilde{r} = [0.54\ 1.83\ -2.25]^T\ m \tag{6.13}$$

meaning that the perceived target point was $\approx 2m$ above the surface. This can be seen in Fig. 6.1, where the perceived landing point is shown to be above the surface of D2. Note that the s/c reaches that point in 300 s, with a relative velocity (wrt. the perceived target point) close to zero. Then, the control runs out of time and the s/c free falls towards the surface. This is one of the cases commented in section 5.1. It is here that the reason for existence of the assumption $\mathcal{O}(\sigma_r) = 1\ m$ comes to play[1]: the relatively small error in position knowledge ensures that the s/c will probably free fall a very short distance, and will soft land regardless of the navigation error in position.

Table 6.1: Performance of a nominal trajectory for the Two Phased Descent strategy. Knowledge of states is degraded. Thrust is discontinuous and bounded. Environment is unperturbed.

ToF [s]	Δm [g]	$\lvert V_c \rvert$ [cm/s]	$\lvert r_{err}(t_c) \rvert$ [m]
2693.9	8.34	1.53	1.43

The effect of the dispersion in the initial conditions can be appreciated in Fig. 6.2, where is shown the complete instance of the nominal trajectory with degraded state knowledge and initial dispersion. For this particular instance, the dispersion in initial states was:

$$\Delta r = [-5.8\ 4.6\ 47.7]^T\ m \tag{6.14}$$

$$\Delta v = [0.92\ -0.45\ 1]^T\ cm \tag{6.15}$$

[1] see Appendix C for $\sigma_r \approx 10\ m$

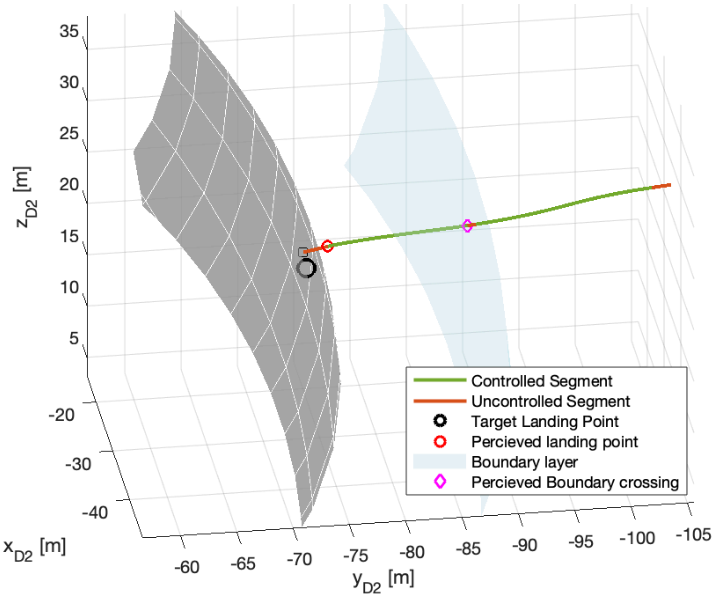


Figure 6.1: Zoom on the Descent Phase of an instance of the nominal trajectory. The states knowledge is noised, the environment is unperturbed. The thrust is bounded, discontinuous. Expressed in the D2 RF.

which slightly impacts the direction in of the the trajectory with respect to the Reference Trajectory (yellow in Fig. 6.2 but has a 40m shift from the reference starting point. Note that this is about a 10% error in distance from D2. Regardless of such error, the great control authority of the MSSG, even degraded by the discontinuity of the TDP, ensures that the s/c reaches the perceived landing point with great precision.

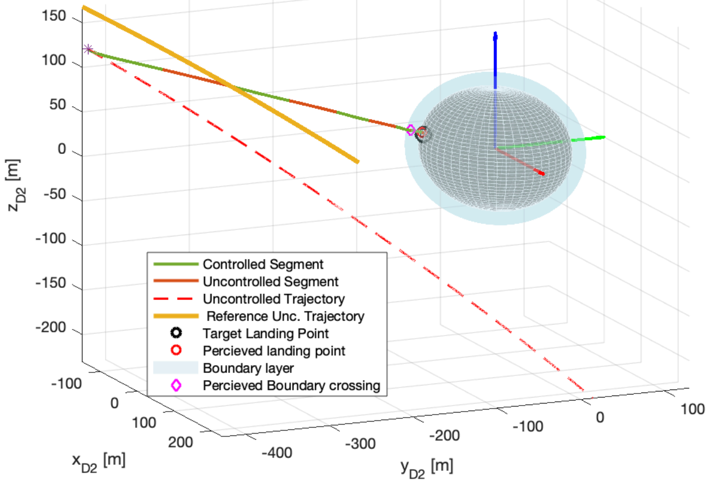


Figure 6.2: Instance of the nominal trajectory. The states knowledge is noised, the environment is unperturbed. The thrust is bounded, discontinuous. Expressed in the D2 RF.

6.2 Performance of the TDP Under Stochastic Simulations

6.2.1 Stochastic state knowledge and dynamical perturbation

This section's goal is to understand what are the statistical properties of the TPD's performance when the latter is degraded by:

- Stochastic knowledge of the states (r,ν) determined by the model presented above.

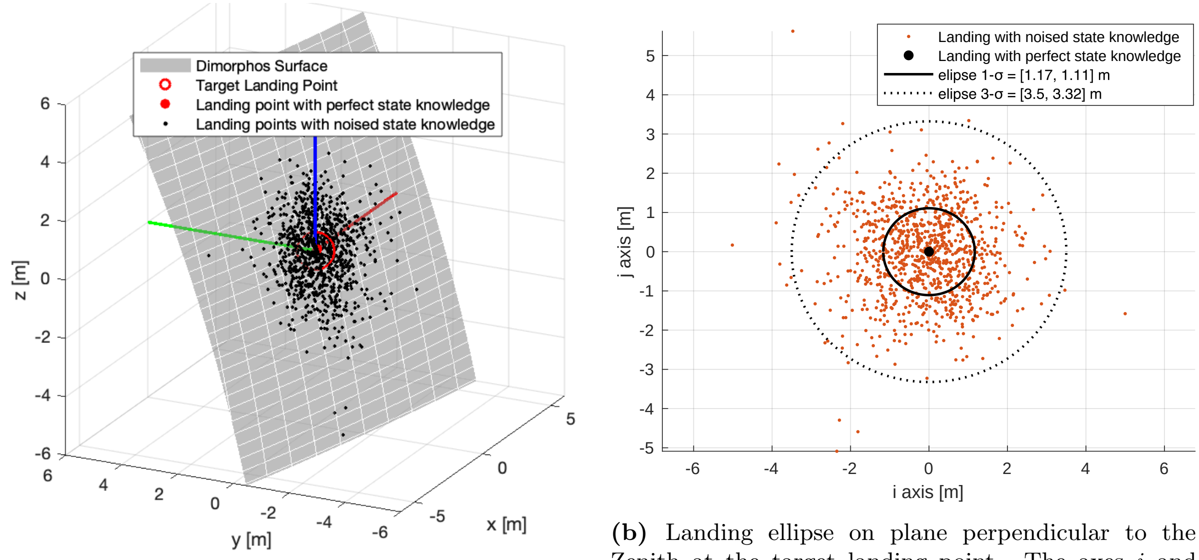
- Stochastic dispersion on the initial conditions determined by the model presented above.
- Stochastic nominal perturbations as introduced in section 4.3.5 and summarized in Appendix B.

The thrust has been bounded and quantized as summarized in Appendix B. The landing target is DART's crater location as shown by Hera's Kernels[2]. Starting and landing epochs are those of the nominal trajectory chosen in Chapter 5. 1000 Monte Carlo (MC) runs have been performed.

Landing Precision

The landing ellipse under the stochastic conditions mentioned above is shown in Figs. 6.3a and 6.3b below. From both of them can be seen how the landings are confined in the vicinity of the landing point without stochastic variables (called "ideal landing point" hereafter). In particular, the landings occur inside a maximum radius of $\approx 3.5m$ from the ideal landing point. This deviation is due to the poor state estimation error. Indeed a $\sigma_r = 1m$ error in position estimation has lead to an $\sigma_r \approx 1m$ error in landing position.

Initial conditions play a minor role due to the high control authority of the MSSG. The modelled disturbances, which are very likely to be $p_{max} < \Phi_{conv}$, have been observed to play a small role in the distributions presented here. This is noteworthy because, even if from MSSG theory (see section 4.2) it is known that when $\Phi > p_{max}$ the control is stable in FT, in the TPD strategy the control is turned off approximately half of the time. During the uncontrolled periods the system might diverge irreversively, which is helped by p being constant (see section 4.3.5). Instead, in the studied case, since $\Phi = \Phi_{conv} \gg p_{max}$ the control manages to converge the trajectory just with its turn-on time even. Notably, this is possible because $T_{max} = 10mN \Rightarrow u_{max} \approx 8.3 \cdot 10^{-4} m/s^2$, and u_{max} is big enough to allow a selection of a great enough Φ_{conv}.



(a) Landing ellipse on D2 surface. Expressed in a base aligned with D2 RF and centered on the target landing point.

(b) Landing ellipse on plane perpendicular to the Zenith at the target landing point. The axes i and j are its base, which approximately correspond to x_{D2} and y_{D2}.

Figure 6.3: Landing ellipse for a control degraded by stochastic state knowledge, stochastic dispersion on initial conditions and stochastic nominal perturbations

.

[2] ESA SPICE Service, Hera Working Groups, Instrument Teams. *Hera Spice Kernel Set.* 2020. Last Accessed: aug-2021. [online]. Available: `https://www.cosmos.esa.int/web/spice/spice-for-hera`

Collision Velocity

The magnitude of V_c along the local Nadir and the angle of V_c with respect to Nadir is shown in Figs. 6.4a and 6.4b below, in which the occurrences of each magnitude are fit to a Normal Distribution. The local Nadir points towards:

$$\hat{N}^{D2} \approx [0.19\ 0.93\ \ -0.32]^T \tag{6.16}$$

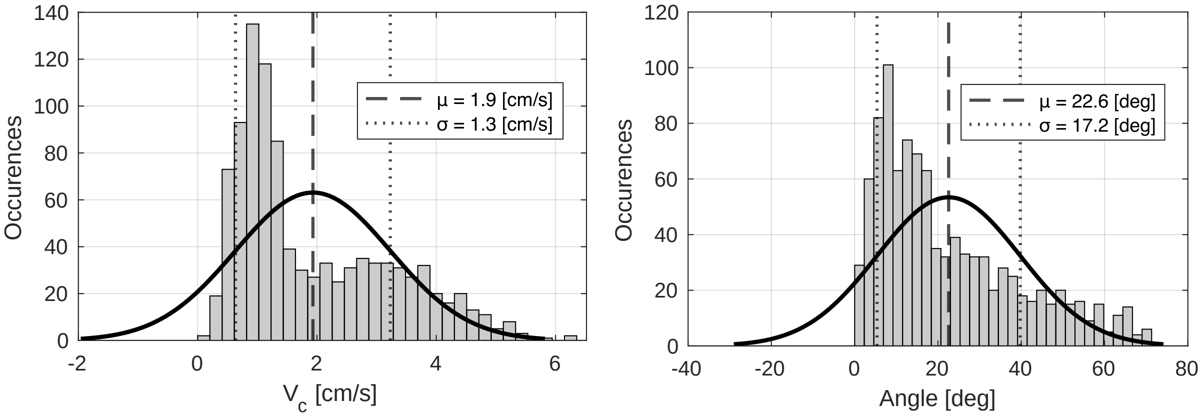
and is expressed on the D2 RF. Note that, approximately $\hat{N}^{D2} \approx [0\ 1\ 0]^T$. The distributions in Figs. 6.4a and 6.4b significantly differ from Gaussian ones, but it is still remarkable that:

- Typically the s/c collided normally to the ground at $\approx 1.9cm/s$; and the std. dev. of the normal velocity was $\sigma_{normal\ vel.} \approx 1.3cm/s$. The distribution is asymmetrical with a strong concentration of points on the interval $V_c \in [0,2]\ cm/s$.

- The typical angle of collision wrt. the local Nadir was $\approx 22.6\,\text{deg}$ and the variance of the collision angle was $\sigma_{col.angle} \approx 17.2\,\text{deg}$. The distribution is asymmetrical with a strong concentration of points on $angle \in [0,20]$ deg

Importantly, in Fig. 6.6 below can be seen how the probability to land with a velocity lower than the scape one ($\approx 4.5cm/s$) is near to 98%. For completeness' sake, in Fig. 6.5 are displayed the probability distributions for the components of V_c along D2 RF. There it can be observed that the component of V_c along y_{D2}, which is close to the local Zenith (see Eq. (6.16)) is always positive, since the s/c approaches D2 along the direction of y_{D2} and $V_c \triangleq \nu_{err} = \nu(t)$. Instead, the other components of V_c fit good over a symmetric Gaussian Distribution which is due to a Gaussian stochastic variation of the state knowledge. Finally, in Fig. 6.7 can be seen how the fuel consumption closely follows a Gaussian distribution centred at $\Delta m \approx 5.3\ g$ and with $\sigma_{\Delta m} = 0.4g$, for an $I_{sp} = 80g$.

In conclusion, the off-Nadir components of V_c and all the components of the landing error are strongly related to the navigation knowledge. For the studied conditions (i.e. navigation error and constant disturbance along the flight, with $\sigma_p = 10^{-5}/3\ m/s^2$ (see section 4.3.5)) perturbations and initial dispersion have a small impact. The landing performance is very acceptable: $V_{c-i} < 4.5cm/s$ with 98% of likely hood.



(a) Magnitude of V_c along the local Nadir at target landing point.

(b) Angle of V_c with respect to the local Nadir at target landing point.

Figure 6.4: Histograms and Normal Distribution fit of collision velocity along local Nadir and V_c direction with respect to local Nadir.

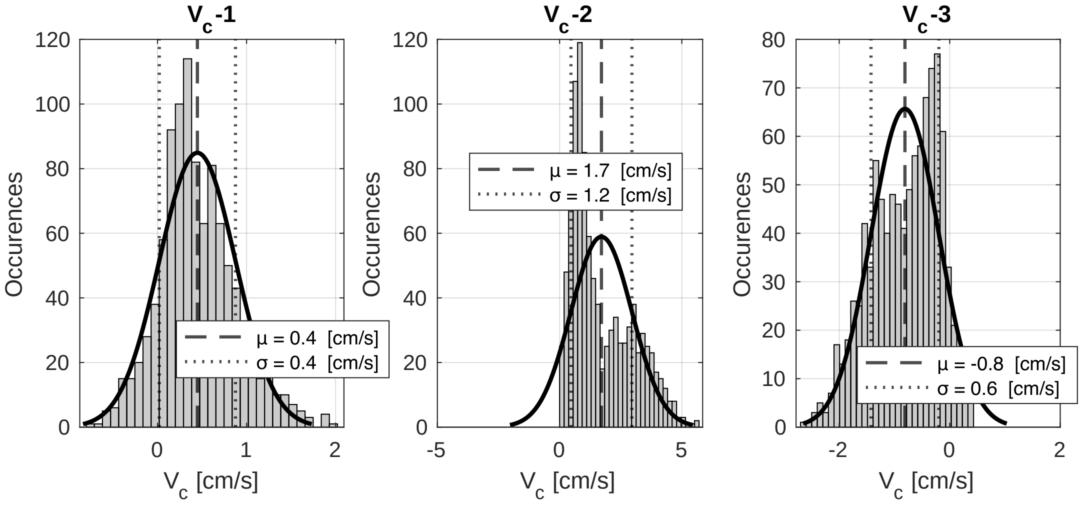


Figure 6.5: Histograms and Normal Distribution fit of the components of V_c along D2 RF.

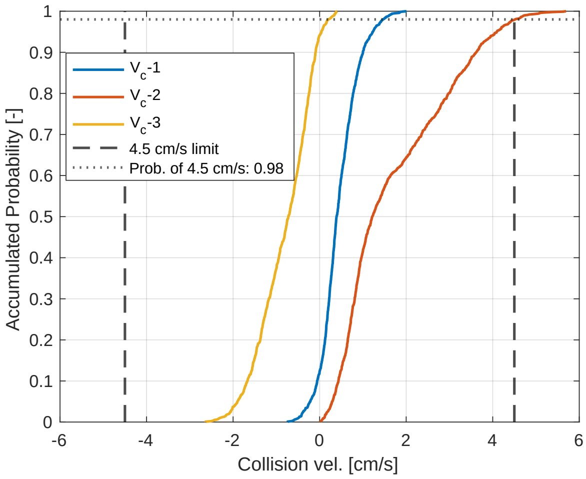


Figure 6.6: Accumulated probability of V_c components along D2 RF.

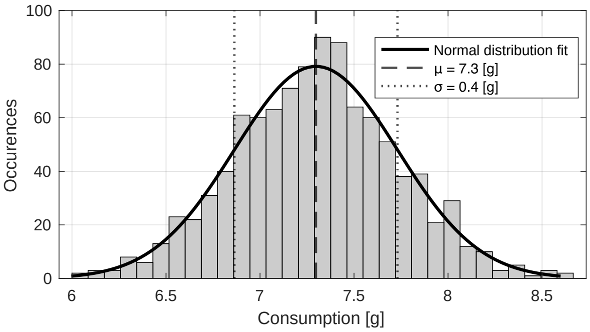


Figure 6.7: Probability distribution of fuel consumption for $I_{sp} = 80s$

6.2.2 Effect of Thruster's Performance

Stochastic Thruster Pointing

From the results of section 6.2.1 can be concluded that the control is able to reject reasonable perturbations even being turned off half of the time, and that the landing performance is mainly impacted by the poor navigation knowledge. What can also degrade performance of the control is thrust pointing errors. In this section I study its combined effect with the poor navigation knowledge.

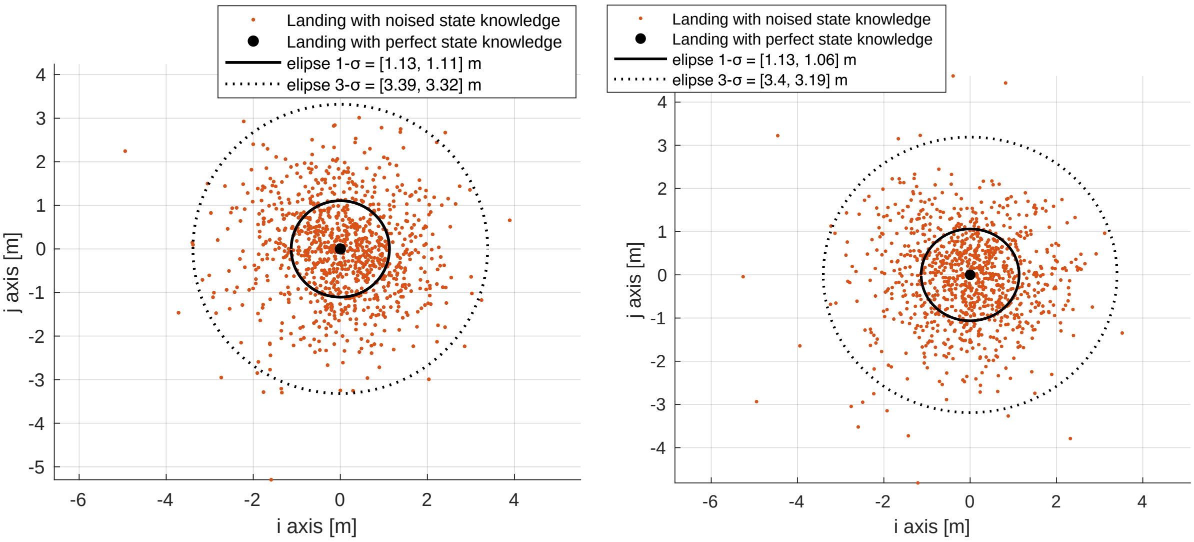
The thrust pointing error has been modeled with a random set of 3 Euler rotations along x_N, y_N, z_N. The rotation angle along each direction follows:

$$\theta_i \sim \mathcal{N}(0, \sigma_{thrust}^2) \tag{6.17}$$

The error is constant during the trajectory. The TDP has been tested with two values of σ_{thrust}:

- $\sigma_{thrust} = 1\,\text{deg}$, consistent with assumptions in [26].
- $\sigma_{thrust} = 5\,\text{deg}$, as a more conservative take.

The degradation on the landings' performance in both cases has been minimal. In Figs. 6.8a and 6.8b is shown the landing ellipse for the $\sigma_{thrust} = 1\,\text{deg}$ and $\sigma_{thrust} = 5\,\text{deg}$ respectively. It can be seen how the difference between the two is very small, as is the difference with the perfect thrust pointing of Fig. 6.3b. In Figs. 6.9a and 6.9b are shown the distribution of V_c components for both cases. Again, the difference between both thrusting accuracies is small: the worse thrust pointing has recorded only a $0.1cm/s$ increase in std. dev. of the off-Nadir components wrt. the 1 deg pointing error.



(a) 1 deg of error in thrust pointing.

(b) 5 deg of error in thrust pointing.

Figure 6.8: Landing ellipse for stochastic state knowledge and thrust pointing error.

Stochastic Specific Impulse

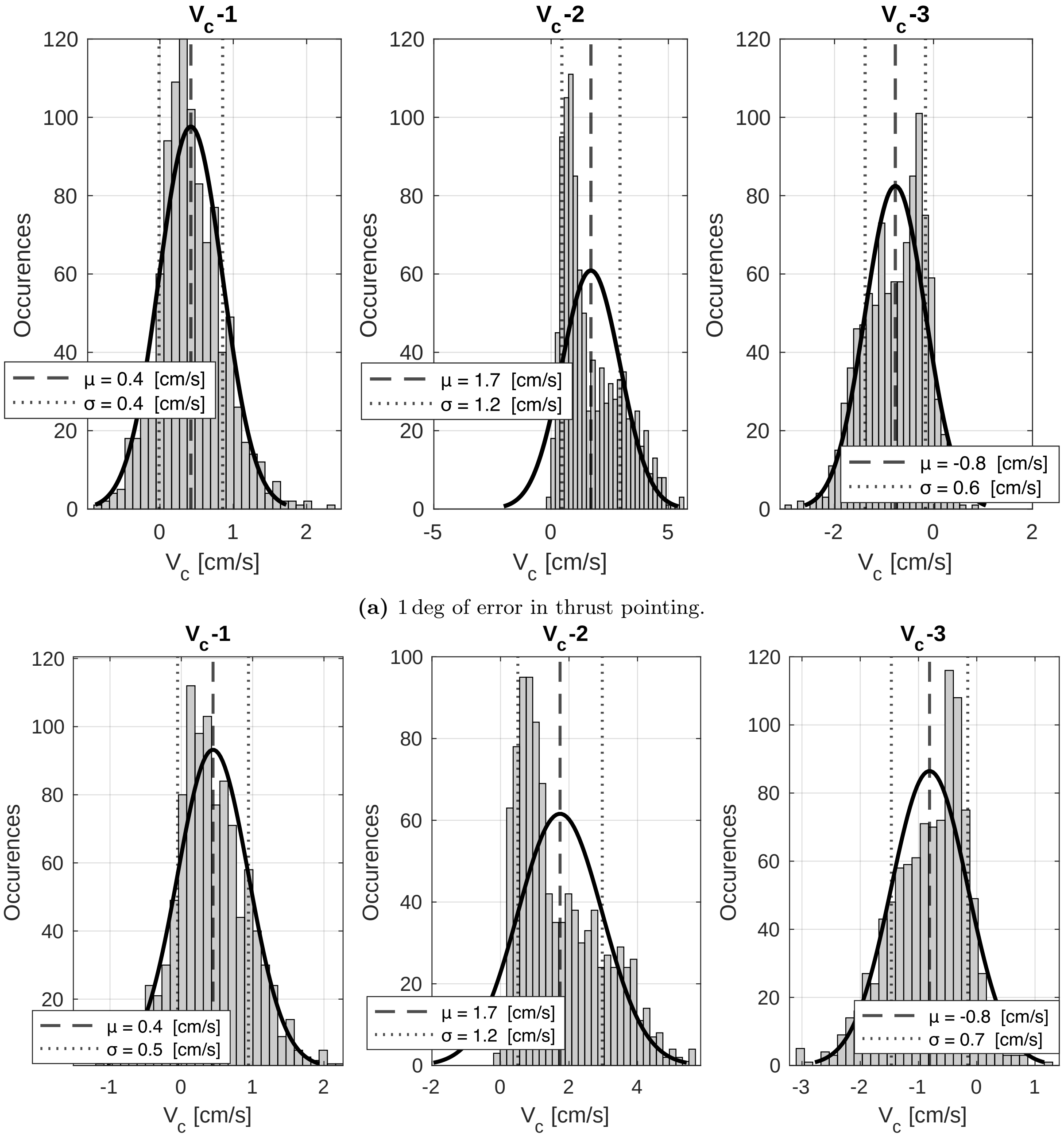
In this work an $I_{sp} = 80s$ has been assumed. It is expected that the impact of its value on the dynamics is small, since the typical change in mass is $\mathcal{O}(\Delta m) = 10g$, a 1% of the s/c mass. Nonetheless, it is expected that its magnitude proportionally affects the amount of fuel expended, since $\dot{m}$ is computed as $\dot{m} = \frac{T}{I_{sp} g_0}$. To asses the variation in Δm for uncertainty in I_{sp}, 1000 MC runs have been performed randomly extracting I_{sp} from:

$$I_{sp} \sim \mathcal{N}(60, 20/3)\ s \tag{6.18}$$

which will vary I_{sp} between an optimistic value of $80s$ and a conservative value of $40s$. The thrust pointing has been noised with a $\sigma_{thruster} = 5\,\text{deg}$ error and stochastic knowledge of the states and the dynamical environment has been introduced as proposed in the previous section.

As expected, in Figs. 6.10 and 6.11 can be observed how the landing performance for I_{sp} ranging form $(40, 80)\ s$ does not change significantly wrt. to the performance at $80s$ previously shown in Figs. 6.8b and 6.9b. Nonetheless, the consumption distribution (see Fig. 6.12) directly reflects the Gaussian character of I_{sp}: it is centered at $\mu \approx 10g$ (likely from $I_{sp} \approx 60s$, and is limited on the bottom by $\Delta m \approx 7g$, the consumption that was characteristic of $I_{sp} = 80g$ (see Fig. 6.7), which is the $3-\sigma$ limit of the I_{sp} Normal distribution. Hence, for expected values of $I_{sp} \in [40, 80]s$ the values of Δm will remain in a small interval with the order of $\mathcal{O}(\Delta m) = 10g$. Simultaneously, consumption will be strongly impacted by the value of I_{sp}. With I_{sp}, the uncertainty on the consumption is of $\sigma = 0.4g$ (see Fig. 6.7). The change in Δm with I_{sp} follows

$$\frac{\Delta(\Delta m)}{\Delta I_{sp}} \approx -0.2\ g/s \tag{6.19}$$



(a) 1 deg of error in thrust pointing.

(b) 5 deg of error in thrust pointing.

Figure 6.9: Distribution of V_c components and Gaussian fit for stochastic state knowledge and error in thrust pointing.

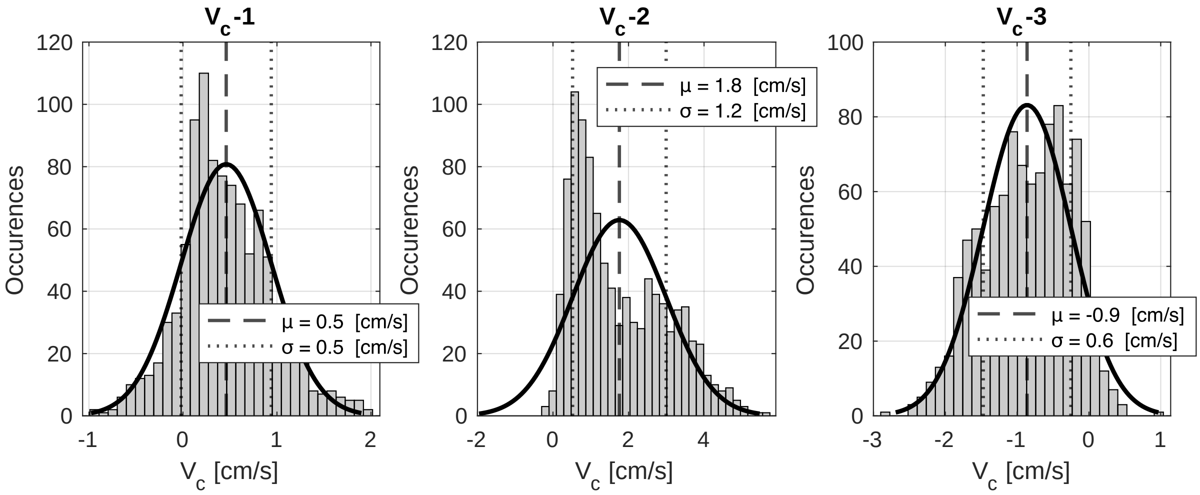


Figure 6.10: Distribution of V_c components and Gaussian fit for stochastic state knowledge, initial values, dynamical perturbations, thrust pointing error and I_{sp}.

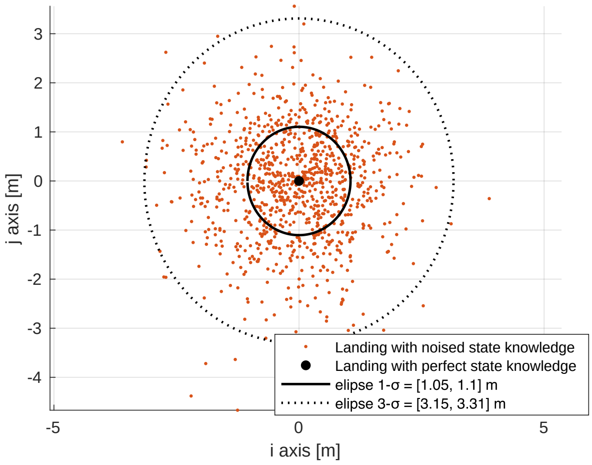


Figure 6.11: Landing ellipse for for stochastic state knowledge, initial values, dynamical perturbations, thrust pointing error and I_{sp}.

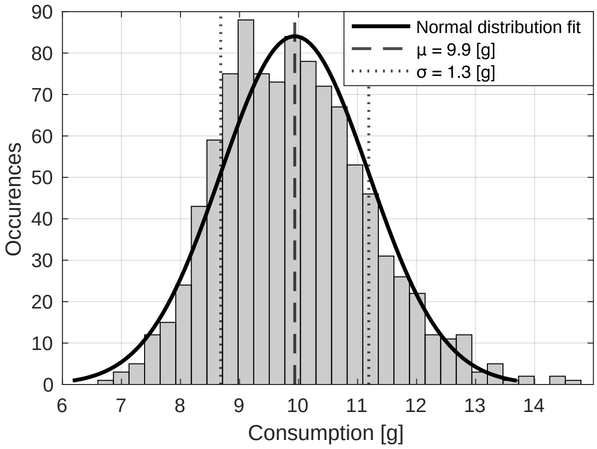


Figure 6.12: Fuel consumption for for stochastic state knowledge, initial values, dynamical perturbations, thrust pointing error and I_{sp}.

Chapter 7

Conclusions

Results

In this work I have selected the Multiple Sliding Surface Guidance (MSSG) as control algorithm for Milani CubeSat's landing on Dimorphos, the moon of the asteroid 65803 Didymos. In the selection has been assumed that its GNC system may perform close-loop navigation with continuous thrust. The choice is driven by the MSSG's robustness to bounded perturbations, simplicity and relatively low CPU consumption. The MSSG uses sliding mode control to reject bounded dynamical uncertainties in finite time, cancels the modelled environmental forces and then generates an acceleration to set the s/c on an ideally linear trajectory in Dimorphos' body-attached reference frame. Chattering has been mitigated using a Schmitt Trigger and a boundary layer around the sliding manifold.

I have studied MSSG's behaviour in Didymos' dynamical environment, and observed that i) longer time of flights than $300s$ (relevance of this limit is explained below) ii) landing times later than the close pass of a reference trajectory are beneficial in terms of fuel consumption (Δm) and collision velocity (V_c). To evaluate the MSSG's performance I first tune the control parameters (Λ and n) using a "nominal landing trajectory", defined by a control starting epoch and a landing epoch. The initial states are drawn from a the mentioned reference trajectory. The "nominal landing trajectory" lasts $1h$ and is $\approx 0.5km$ long.

Then, I design a firing strategy called the Two Phase Descent (TPD), so that the s/c performs only $300s$ firings during $\approx 95\%$ of the trajectory. In the TPD, the landing trajectory is divided in two phases: First, during the Approach Phase the control is turned on and off sequentially. Along the mentioned "nominal trajectory" this makes for $\approx 95\%$ of the distance to be traveled. Next, at $15m$ above Dimorphos' surface, the Descent Phase starts. In this period, the control is turned on for up to $300s$ while awaiting touchdown, regardless of the control's state at the start of the phase. For the TPD I have assumed that a navigation knowledge of $\sigma_r \approx 1m$ may be possible. A σ_r of this order is required to avoid targeting a point further below or far above Dimorphos' surface, which would endanger landing success. The control parameters for the Approach (1) and Descent (2) phases are different, and using an iterative design process have been set to $\Lambda_1 = 2.5$, $n_1 = 0.2$, $\Lambda_2 = 3$ and $n_2 = 0.7$.

Finally, I study the landing performance (i.e. landing ellipse and collision velocity) of the TPD using Monte Carlo analyses of 1000 runs. They show how, when perturbations are lower than the rejection term Φ_{stab}, the landing performance is mainly impacted by the navigation knowledge. With a navigation error of $\sigma_r \approx 1m$, the dispersion error of the landings with respect to the landing point using a perfect navigation knowledge is of $\sigma_{r,land} \approx 1.1m$. The trajectories typically collide with an angle of $\approx 23\,\text{deg}$ with respect to the local Nadir. The incidence angle has a dispersion of $\sigma_{angle} \approx 17\,\mathrm{deg}$. The limit of $V_c < 4.5cm/s$ (approximate Dimorphos' scape velocity) is respected with $\approx 98\%$ of confidence. Indeed the local-Nadir component of V_c has a typical value of $\approx 2cm/s$ with a dispersion of $\sigma \approx 1.3cm/s$. The off-local-Nadir components of V_c remain at lower values (order of $0.5cm/s$) and closely follow a Gaussian distribution. The control has been challenged with $1\,\mathrm{deg}$ and $5\,\mathrm{deg}$ pointing errors, which have shown to faintly impact the performance. Overall, the strong control authority of the MSSG is capable to quickly correct errors with respect to the ideal, linear trajectory. It can reject perturbations up to the spacecraft's maximum thrust, at the time of writing $10mN$. The fuel consumption of the flight is closely related to the value of I_{sp}, but always located at $\mathcal{O}(I_{sp}) = 10g$, and it has been detected that the increase in performance is $-0.2g$ per *sec.* of I_{sp}.

Discussion and Further Work

When used with continuous thrust for the complete trajectory, the MSSG reveals itself as a very good control algorithm in terms of accuracy and robustness to dynamical perturbations. Indeed, in Didymos' dynamical environment it can land Milani with collision velocities of the order of cm/s. Implementation is straightforward and, unlike e.g. the Optimal Sliding Guidance algorithm, the MSSG does not require recursive propagation of the dynamics to compute the acceleration command. This of course goes against fuel optimization but in the case at hand the environmental accelerations and s/c mass are so small that fuel consumption is always at the order of $10g$, which is acceptable.

In previous works [17] has been shown how fuel consumption during the flight can critically depend on how far is the prescribed Time-of-Flight (ToF) from the ToF of a fuel-optimal trajectory. In this work such information has not been used and a line of further research could be to compute fuel-optimal trajectories to understand whether the results of [17] appear in Didymos, and in such case to ensure that the prescribed ToF is greater than the fuel-optimal ToF.

Still in the continuous thrust framework, the performance of the MSSG in terms of V_c and Δm has been observed to be separated into regions of, rather than different ToFs, varying combinations of control-starting epochs and landing epochs. In particular, landings after the close passage of a reference, uncontrolled trajectory about Dimorphos were beneficial. Future research could shed light on the impact of varying Time of Flights and starting/landing epochs and evaluate which has the greatest effect. This could inform on whether it is more important to achieve long controlled flights or rather to reach the binary system at a given epoch to perform the flight within a predicted system's geometry.

The adaptation of the continuous thrust MSSG to mostly use $300s$ firings (in what I have called the Two Phased Descent) has shown a remarkable ability to maintain the landing performance in terms of V_c, landing ellipse and Δm. This opens the door to consider using the MSSG in a discrete-maneuver fashion: future research could determine the feasibility to use the MSSG in pre-commanded, discrete maneuvers, which has not been proposed before. Open questions are what would be the maximum allowable uncontrolled flight time between maneuvers while maintaining an acceptable performance; or how long should the firings be in terms of such time. Fundamentally, it could be further investigated what degradation in performance comes from using the MSSG in discrete firings with respect to continuous thrust.

At the current state of development, the TPD does not enforce a strict requirement of always firing for $\Delta t \leq 300s$. In some cases, during the last $15m$ the TPD may let the control be active for longer times. This is a reasonable transgression: during the final minutes of descent towards the asteroid, it could be advantageous to leave the control be less restricted in order to ensure mission success. Other alternatives that would maintain the $300s$ limit have been considered and are commented in this work. This design choice has been made to favor robustness and let the performance and decisions of the Guidance & Control logic be as independent as possible to variations on how the s/c ends up approaching the asteroid. Indeed, this particular issue could be subject of further investigation, and iterations on the design could let to slight changes on how the control behaves near the asteroid.

Hence, the TPD strategy has shown that it is possible to use simple closed-loop algorithms like the MSSG to generate acceleration commands during prescribed time batches, and that the performance of the overall trajectory using this approach is related to that of the continuous-thrust usage of the MSSG. In the case at hand, the landing performance is statistically remarkably good, which could mean that the system has not been perturbed enough. Indeed, it confirmed that the MSSG has the ability to resist bounded dynamical perturbations. For this reason, future extensions of this work could be to investigate which is the perturbation's limit (environmental, control etc.) that the TPD endure, and provide worse-case scenarios to robustify the control design. Regarding dynamical uncertainty, the limit the MSSG can correct is ruled by the value of the constant Φ, which is translated in to the maximum thrust the thruster can exert can exert. Milani thruster is currently assumed to exert $10mN$ maximum, which is greater than the accelerations received from Didymos' environment. Stronger control could induce stronger chatter, but the use of a Schmitt trigger has revealed to be an effective way to mitigate it; and the usage of $sigmoid()$ instead of $sign()$ values have been effective in softening the control without significant degradation in performance.

In particular, in this document I perturb the environment with a random but constant acceleration in a given direction, which is bound to more easily accumulate an error in a random direction when the control is turned off. Nonetheless, other approaches have been taken in previous works to perturb the environment. E.g. to randomly and slightly modify the mass of the attractor body or the magnitude of the SRP [17]. Other approaches could be to generate a random, constant, percentual change for both magnitude and direction of the environmental accelerations. Overall, to degrade the environment and experimentally confirm the limits of the control would be an interesting line of work to pursue.

Appendices

Appendix A

Navigation Schemes

Missions to small bodies have used different navigation schemes, both fully autonomous or mixed with ground intervention, [3]. The navigation part of the GNC system can be divided in the measurement generation and the state estimator, which is typically a version Kalman Filter. To obtain measurements, typically is used an IMU aided by an Optical Navigation (OpNav) camera, complemented by altitude information coming from LFR or LIDAR at lower altitudes [11]. The OpNav part is fundamental and can be done by either "Image Correlation" or "Landmark Matching". Such classification is depicted in Fig. A.1. Below are highlighted the alternative implementations for both the measurement unit and the estimation unit.

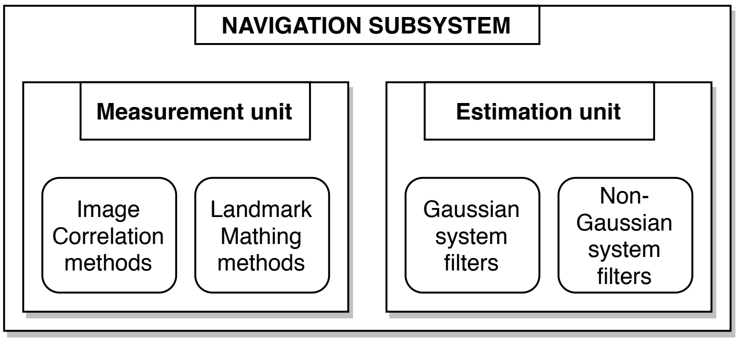


Figure A.1: Elements of the navigation part of the GNC. Shadowed boxes indicate systems, rounded boxes indicate alternative implementations.

A.1 Measurement Unit

Apart from IMU, LRF and LIDAR measurements, a fundamental part of the Measurement Unit is the OpNav part, where measurements coming from asteroid images have to be constructed. The way in which such measurements, using images, are constructed, is separated in Image Correlation and Landmark Matching. Operational principle is to compare a taken image with another one, that either is taken by the s/c at similar times or it is extracted from an on-board map.

In Landmark Matching framework, the operational principle is to identify particular features of the asteroid's surface (called landmarks or control points), whose position on the target body is known. The measurements are linked to the difference inside the picture frame between the expected position of such landmarks and the their actual position. The measurements are obtained using the following logic [27], [20]:

1. Suppose a nominal position of the landmark in the photo (which emanates from a nominal position of the s/c.
2. Take a picture and measure the difference between expected and actual change of position in the picture frame, in terms of pixel and line (Δp, Δl).
3. Apply Least Squares method to obtain a a corresponding deviation in position and attitude of the satellite ($\Delta\mathbf{P}$, $\Delta\mathbf{Q}$); where $\mathbf{P} \equiv \mathbf{r}$ and $\mathbf{Q}$ contains a set of three small Euler angles.

The logic between using Least Squares Method is that there is a link between the s/c deviation from a nominal state and a deviation of the landmark appearing in the photo with respect to an expected position. Such link can be exploited to find use the Least Squares method as shown below:

$$(p, l) = f(P, Q) \rightarrow \begin{cases} d\mathbf{p} = \frac{\partial \mathbf{p}}{\partial \mathbf{P}} dP + \frac{\partial \mathbf{p}}{\partial \mathbf{Q}} dQ \\ d\mathbf{l} = \frac{\partial \mathbf{l}}{\partial \mathbf{P}} dP + \frac{\partial \mathbf{l}}{\partial \mathbf{Q}} dQ \end{cases} \tag{A.1}$$

$$\phi \equiv [\partial(\cdot)/\partial\mathbf{P} \ \partial(\cdot)/\partial\mathbf{Q}]^T \tag{A.2}$$

$$\theta \equiv [d\mathbf{P} \ d\mathbf{Q}]^T \equiv [\Delta\mathbf{P} \ \Delta\mathbf{Q}]^T \tag{A.3}$$

$$\mathbf{y}_m = \phi^T \theta + \mathbf{e}; \ \mathbf{e} \sim G(0, \mathbf{R}^{-1}) \tag{A.4}$$

$$\hat{\theta} = \sum_{i}^{N_{pic}} \left([\phi^T \mathbf{R} \phi]^{-1} \cdot \phi \mathbf{R} \mathbf{y}_m \right) \tag{A.5}$$

The analytical expression of Eq. (A.1) can be found in [20]. There are several schemes to automatically obtain landmarks, them being craters (which not always exist) or other features of interest. For the later, the method requires images with similar illumination and resolution with moderate stereo separation, which derives in large absolute uncertainties. Recently there have been developments using machine learning that can directly detect craters in images [11]. A new technique to improve Landmark Navigation is presented by Gaskell et al. [27]. There, is proposed to first explore the asteroid with ground-based navigation in order to generate a high-resolution Global Topographic Map (GTM), which is a 3-D map of the asteroid made out of tiles whose centers are the new Landmark points. The position of these centers is precisely known, and images of them can be rendered from the GTM with different illumination conditions and viewpoints, to then be compared with their respective images taken by the OpNav cameras and to follow the process aforementioned.

This technique allows an autonomous, reliable navigation on the asteroid but requires an *a priori* exploration of it and the capability to compute (either on-ground or on-board) a GTM.

Regarding Image Correlation Methods, they do not require of an identification of geographic feature points in the photos. In this framework, image taken are compared to an *a priori* loaded on board map. Then the sensed image is correlated with the stored one, and this information is used to correct the IMU estimation on the s/c position [28].

Landmark Matching techniques are the most resistant to noise in the data, distortions, changes in viewpoint and illumination [11], but require an *a priori* knowledge of the terrain if large absolute uncertainties are to be avoided. To tackle this dilemma, Hayabusa missions brought their own artificial Landmark, which enabled to perform the landing with great relative position accuracy [3]. Image Correlation techniques are easy to implement but they are sensitive to scale and perspective differences between on-board map and sensed images.

A.2 Estimation Unit

The Estimation Unit takes measurements to filter the propagation of the system dynamics which yields the current states of the system. Such procedure is done by taking into account the propagation of uncertainty of the system. In [11] is noted how in the state-of-the art there are two main approaches to construct the filterers.

In one hand, there are the group of schemes that consider the system Gaussian. To this group belong the classical Kalman Filter and its augmentations, such as the Extended, where the dynamics of the system are linearized around a nominal state trajectory. To increase the convergence accuracy that is obtained with such filter, the Uscented Kalman Filter is available, where the dynamics are approximated with an unscented transformation instead of a linearization.

On the other, the Gaussian stochasticity assumption can be avoided using a Particle Filter (based on Monte-Carlo simulations) or using Polynomial Chaos to approximate the stochastic properties of the system by an orthogonal polynomial expansion. The Polynomial Chaos approach has been shown to provide the same accuracy as the Particle Filter but with more accuracy. To further account for modeling errors, developments that use Minimum Model Error approach or mixed H_2/H_{inf} approaches estimate the states.

Appendix B

Modelling & Implementation

The aim of this appendix is to detail what is the dynamical model and what is the control algorithms used for the s/c simulations. The control algorithm is analysed in detail in Chapter 4 and hence here it will only be summarized the way in which it has been applied in the simulations. This appendix is divided as follows: Sections B.1 and B.2 present the dynamical equations used to model the problem and its validation. Section B.3 summarizes what is the control algorithm and what are the dynamical perturbations.

B.1 Dynamical Modelling

B.1.1 Mathematical notation and reference frames

In the following derivation, vectors and matrices are noted in bold lowercase and uppercase respectively. Transposed tensors are noted with the superscript $\mathbf{x}^T$. A vector that goes to point P from point Q is noted as: $\mathbf{x}_{P-Q}$.

An item expressed in a given reference frame is noted with a superscript $\mathbf{x}^Y$, where $\mathbf{x}$ is a generic quantity and Y is the base on which it is expressed. The rotation matrices that go to base B from base A are defined as:

$$\mathbf{C}^{B/A} = \begin{bmatrix} \mathbf{b}_1^{A,T} \\ \mathbf{b}_2^{A,T} \\ \mathbf{b}_2^{A,T} \end{bmatrix} = \begin{bmatrix} \mathbf{a}_1^B & \mathbf{a}_2^B & \mathbf{a}_2^B \end{bmatrix} \tag{B.1}$$

where the vectors $\mathbf{a}_i$ and $\mathbf{b}_i$ represent the unit vectors defining the bases A and B respectively. The reference frames (RF) that are going to be used are displayed in Table B.1 below.

A derivative with respect to a reference frame A is noted as: $\left.\frac{d\mathbf{y}}{d\mathbf{x}}\right|_A$
The upper dot symbol is analogous to the derivative of a tensor in the frame it is expressed. I.e. $\dot{\mathbf{r}}^N = \left.\frac{d\mathbf{r}^N}{dt}\right|_N$ or $\dot{\omega}^B = \left.\frac{d\omega^B}{dt}\right|_B$.
Any of the notation described above may be replaced with an in-text definition, e.g.: "The vector r points to the s/c CoM from Barycenter." Which means $r = \mathbf{r}_{c-Bar}$. Also, the term "D RF" will be used to generally refer to either D1 or D2 RF.

B.1.2 Translational EOM

The spacecraft is assumed to be a rigid container that loses mass during operations. The change of linear momentum of the solid can be evaluated using the Navier-Stokes equations expressed with respect to an inertial RF (e.g. N RF); and considering a control volume (CV) attached in

Table B.1: Reference frames being used in the derivation.

Abbreviation	Full name	Inertiallity	Comment
N RF	DidymosECLIPJ2000	Yes	Centered in Didymos Barycenter (Bar), oriented as ECLIPJ2000. Defined by ESA's Kernels.
B RF	Body Reference Frame	No	Centered in Milani CoM, fixed on the body.
D1 RF	Didymos Body-attached RF	No	Centered in D1 CoM, fixed on the body. Defined by ESA's Kernels.
D2 RF	Dimorphos Body-attached RF	No	Centered in D2 CoM, fixed on the body. Defined by ESA's Kernels.

the interior of the s/c walls. The Navier-Stokes momentum conservation equation is:

$$\frac{d}{dt}\Big|_N \int_V \mathbf{u}^N \rho dV + \int_S \mathbf{u}^N \rho(\mathbf{u}^N - \mathbf{v}^N_{CV}) \cdot \mathbf{n} dS = \int_S \mathbf{f_{(n)}} dS + \int_V \mathbf{g}^N \rho dV + \sum \mathbf{F}^N_i \tag{B.2}$$

where $\mathbf{u}^N$ is the inertial velocity of all particles in the CV with respect to Didymos Barycenter. $\mathbf{v}^N_{CV}$ is the inertial velocity of the CV walls with respect to Didymos Barycenter, both expressed on the N RF. $\mathbf{f_{(n)}} = \sigma \cdot \mathbf{n}$ are the forces acting on the CV's walls as result of a stress tensor. The term $\sum \mathbf{F}_i$ contains all external forces that are not gravitational. Note that the second term on the LHS is zero except for the point where the mass exits the control volume and leaves to space. The terms of Eq. (B.2) can be further developed as:

$$\frac{d}{dt}\Big|_N \int_V \mathbf{u}^N \rho dV + \dot{m}\mathbf{v}_{exhaust} = A_{e\ nozzle}(P_{ambient} - P_{e\ nozzle})\mathbf{n}_{outlet} + \sum F_{external} \tag{B.3}$$

The first term on the LHS can be integrated over the body to refer to the CoM quantities:

$$\frac{d(m\mathbf{v}^N_{CoM})}{dt}\Big|_N = -\dot{m}\mathbf{v}_{exhaust} + A_{e\ nozzle}(P_{ambient} - P_{e\ nozzle})\mathbf{n}_{outlet} + \sum_i^{N_{att.}} \mathbf{F}^N_{g,i} + \mathbf{F}_{SRP} \tag{B.4}$$

The term $-\dot{m}\mathbf{v}_{exhaust} + A_{e\ nozzle}(P_{ambient} - P_{e\ nozzle})\mathbf{n}_{outlet}$ can be identified as the thrust exerted by the propulsion device, and be renamed $\mathbf{T}$:

$$\mathbf{T} \triangleq -\dot{m}\mathbf{v}_{exhaust} + A_{e\ nozzle}(P_{ambient} - P_{e\ nozzle})\mathbf{n}_{outlet} \tag{B.5}$$

which allows to compress the expression of the change in momentum to:

$$\frac{d}{dt}(m\mathbf{v}^N)\Big|_N = \sum_i^{N_{att.}} \mathbf{F}^N_{g,i} + \mathbf{F}^N_{SRP} + \mathbf{T}^N \tag{B.6}$$

where it has been noted that the external non-gravitational forces acting on the s/c is just the SRP and $\mathbf{v}_{CoM}$ has been renamed $\mathbf{v}$. The following assumption can be done:

- Change of momentum of the CV due to change in its mass ($\dot{m}\mathbf{v}$) is negligible

which allows to develop Eq. (B.5) as follows:

$$\frac{d}{dt}(m\mathbf{v}^N)\Big|_N = \sum_i^{N_{att.}} \mathbf{F}^N_{g,i} + \mathbf{F}^N_{SRP} + \mathbf{T}^N$$

$$\frac{dm}{dt}\Big|_N \mathbf{v}^N + \frac{d\mathbf{v}^N}{dt}\Big|_N m = \sum_i^{N_{att.}} \mathbf{F}^N_{g,i} + \mathbf{F}^N_{SRP} + \mathbf{T}^N$$

$$\frac{d\mathbf{v}^N}{dt}\Big|_N m \approx + \sum_i^{N_{att.}} \mathbf{F}^N_{g,i} + \mathbf{F}^N_{SRP} + \mathbf{T}^N$$

where $\mathbf{v}$ is the inertial velocity of the s/c CoM. $\sum_i^{N_{att.}} \mathbf{F}_{g,i}$ is the sum of gravitational forces due to the presence of several attractors. The gravitational environment about the Didymos system can be divided into different regions. In [25] is shown how the Solar Tide can be neglected for close-proximity operations around Dimorphos. In such case:

$$\sum_i^{N_{att.}} \mathbf{F}_{g,i}^N = \mathbf{F}_{g,D1}^N + \mathbf{F}_{g,D2}^N \tag{B.7}$$

Let $\mathbf{r}^N$ be the vector that goes from Didymos Barycenter to the s/c CoM expressed in the N RF (i.e. $\mathbf{r}_{c-Bar}^N$). Since $d(\mathbf{r}^N)/dt|_N = \dot{\mathbf{r}}^N = \mathbf{v}^N$ the system can be expressed in terms of $\mathbf{r}$ as a combination of two first-order PDE:

$$\begin{cases} \dot{\mathbf{r}}^N = \mathbf{v}^N \\ \dot{\mathbf{v}}^N = \mathbf{g}_{D1}^N(\mathbf{r}_{C-D1}^N, \mathbf{C}^{N/B}) + \mathbf{g}_{D2}^N(\mathbf{r}_{C-D2}^N, \mathbf{C}^{N/B}) + \mathbf{C}^{N/B}\mathbf{a}_{SRP}^B + \mathbf{C}^{N/B}\frac{\mathbf{T}^B}{m} \end{cases} \tag{B.8}$$

where the terms $\mathbf{g}_D^N$ are the force over mass executed by the gravity of the asteroid D-i upon the s/c. $\mathbf{a}_{SRP}$ is the analogous quantity exerted by the SRP.

B.1.3 Rotational EOM

The rotational dynamics are described by the so-called Euler Equations, which express the rotational state of the asteroid in terms of its instantaneous angular velocity $\omega^B = [\omega_x\ \omega_y\ \omega_z]^T$ wrt. N RF expressed in the B RF. Considering the s/c as a rigid asteroid with inertia $\mathbf{J}^B$ and a set of reaction wheels whose inertia is $\mathbf{J}_r^B$ ($\mathbf{J}_r^B \in \mathbf{J}^B$), it can be shown how the Conservation Of Angular Momentum yields [29]:

$$\left.\frac{d\mathbf{H}^B}{dt}\right|_N = \left.\frac{d\mathbf{H}^B}{dt}\right|_B + \Omega_{B/N}^B \times \mathbf{H}^B = \sum \mathbf{M^B}; \qquad \Omega_{B/N}^B = \omega^B \tag{B.9}$$

where:

$$\begin{aligned} \mathbf{H}^B &= \mathbf{J}_b^B\omega^B + \mathbf{J}_r^B(\omega^B + \omega_r^B) = (\mathbf{J}_b^B + \mathbf{J}_r)\omega^B + \mathbf{J}_r^B\omega_r^B; \qquad \mathbf{J} = \mathbf{J}_b^B + \mathbf{J}_r^B \\ &= \mathbf{J}^B\omega^B + \mathbf{J}_r^B\omega_r^B \end{aligned}$$

and therefore:

$$\mathbf{J}^B\dot{\omega}^B + \omega^B \times \mathbf{J}^B\omega^B + \dot{\mathbf{J}}^B\omega^B + \mathbf{J}_r^B\dot{\omega}_r^B + \omega^B \times \mathbf{J}_r^B\omega_r^B + \dot{\mathbf{J}}_r^B\omega_r^B = \sum \mathbf{M}^B \tag{B.10}$$

where ω^B is the ang. vel. of the s/c body in the B RF; ω_r^B is the relative ang. vel. of the RWs in the B RF; $\mathbf{J}_r$ is the inertia of the body excluding RWs and $\mathbf{M}^B$ are the external moments acting on the s/c in the B RF. The moments $\mathbf{M}$ are the disturbances and the control action: $\sum \mathbf{M} = \sum \mathbf{M_d} + \sum \mathbf{M_c}$. In this model: $\sum \mathbf{M_c} = \mathbf{M}_{thrusters}$ and $\sum \mathbf{M_d} = \mathbf{M}_{g,D1} + \mathbf{M}_{g,D2} + \mathbf{M}_{SRP}$. The dynamics of the RWs are described by:

$$\mathbf{J}_r^B\dot{\omega}_r^B = \mathbf{M}_r^B \tag{B.11}$$

where $\mathbf{M}_r^B$ is a moment applied by the control system on the wheel to control its spin rate.

B.1.4 Kinematic link

The attitude can be represented by a Quaternion, which is equivalent to a corresponding Cosine Direction Matrix but uses less parameters and thus its advantageous for numerical computation. The evolution in time of the attitude Quaternion is dependant on the angular velocity ω and can be shown to be [29]:

$$\begin{cases} \dot{q}^{B/N}(t) = \frac{1}{2}\mathbf{\Omega}(\omega(t)^B_{B/N})q(t)^{B/N} \\ \mathbf{\Omega} \triangleq \begin{bmatrix} [\omega^B]_\times & \omega^B \\ -\omega^{B,T} & 0 \end{bmatrix} \end{cases} \tag{B.12}$$

where $[\cdot]_\times$ is the operator that yields the skew matrix. The Attitude Quaternion can be transformed back to an Cosine Direction Matrix $\mathbf{C}^{X/Y}$ following the transformation in Eq. (B.13):

$$\begin{cases} \mathbf{C} = \left(q_4^2 - \mathbf{q}^T\mathbf{q}\right)\mathbf{I}_3 + 2\mathbf{q}\mathbf{q}^T - 2q_4[\mathbf{q}]_\times \\ \mathbf{q}, q_4 := \quad q = [\mathbf{q} \quad q_4]^T \end{cases} \tag{B.13}$$

Note that the Translational and Rotational EOM are linked through the attitude matrix $\mathbf{C}(t)$: the projections of the forces acting on the s/c CoM depend on the attitude, which in turn depends on the angular velocity. Simultaneously, the transnational motion of the s/c implies a constant change in the environment, which in its turn affects the Rotational dynamics through the torques that the s/c experiences.

B.1.5 Variation of mass

The mass variation is given by the fuel spending during operations. Such change of mass also implies a change on its distribution and therefore on the inertia tensor of the body. The variation of mass and inertia is modeled as:

$$\dot{m} = -\frac{|\mathbf{T}|}{I_{sp}g_0} \tag{B.14}$$

$$\dot{\mathbf{J}} = 0 \tag{B.15}$$

B.1.6 Full 6DOF system

The states of the system consist in:

$$\mathbf{x} = \begin{bmatrix} \mathbf{r}^N & \mathbf{v}^N & \omega^B & q^{B/N} & m & \mathbf{J}^B \end{bmatrix}^T \tag{B.16}$$

where $\mathbf{r} \in \mathbb{R}^{3x1}$, $\mathbf{v} \in \mathbb{R}^{3x1}$, $\omega \in \mathbb{R}^{3x1}$, $q \in \mathbb{R}^{4x1}$, $m \in \mathbb{R}^{1x1}$ and $\mathbf{J} \in \mathbb{R}^{3x3}$ (stacked in a $\mathbb{R}^{6x1}$ vector). In the sections have been developed the equations that yield the states derivatives. Below are displayed together:

$$\begin{cases} \dot{\mathbf{r}}^N = \mathbf{v}^N \\ \dot{\mathbf{v}}^N = \mathbf{g}^N_{D1}(\mathbf{r}^N_{C-D1}, \mathbf{C}^{N/B}) + \mathbf{g}^N_{D2}(\mathbf{r}^N_{C-D2}, \mathbf{C}^{N/B}) + \mathbf{C}^{N/B}\mathbf{a}^B_{SRP} + \mathbf{C}^{N/B}\frac{\mathbf{T}^B}{m} \\ \dot{\omega}^B = \mathbf{J}^{-1}\Big[- \Big(\omega^B \times \mathbf{J}^B\omega^B + \dot{\mathbf{J}}^B\omega^B + \mathbf{J}^B_r\dot{\omega}^B_r + \omega^B \times \mathbf{J}^B_r\omega^B_r + \dot{\mathbf{J}}^B_r\omega^B_r\Big) \\ \qquad + \quad \mathbf{M}^B_{g,D1} + \mathbf{M}^B_{g,D2} + \mathbf{M}^B_{SRP} + \mathbf{M}^B_c\Big] \\ \dot{q}^{B/N}(t) = \frac{1}{2}\mathbf{\Omega}(\omega^B(t))q^{B/N}(t) \\ \dot{m} = -\frac{|\mathbf{T}|}{I_{sp}g_0} \\ \dot{\mathbf{J}} = 0 \end{cases} \tag{B.17}$$

B.1.7 Reduced Orbit-Attitude Coupled Dynamics

The principal aim of this work is to introduce and implement a control action that can rule the translational motion of the s/c so that it lands on an asteroid with a certain performance. As it has been discussed in Chapter 2, the orbit-attitude coupling and the SRP are most important in an asteroid dynamical environment, and hence it is fundamental that the translation motion described by the equation $\dot{\mathbf{v}}^N$ above is fed by the attitude $\mathbf{C}^{N/B}$ information.

At the time of writing, the attitude control $\mathbf{M}_c \rightarrow \dot{\omega}^B$ is being designed by DART Group, and hence in this project the attitude $\mathbf{C}^{N/B}$ will be considered as given by the attitude control system, and to be independent from the translational control system. In this framework $\mathbf{C}^{N/B}(t)$ has been set so that

- The navigation camera in Milani points constantly towards the center of D2
- The star tracker points along the direction to-s/c-from sun, setting the solar panels in the perpendicular plane of the solar vector and protecting the star tracker of direct solar exposure

For this reason, the propagated dynamics correspond to the system:

$$\begin{cases} \dot{\mathbf{r}}^N = \mathbf{v}^N \\ \dot{\mathbf{v}}^N = \mathbf{g}_{D1}^N(\mathbf{r}_{C-D1}^N, \mathbf{C}^{N/B}) + \mathbf{g}_{D2}^N(\mathbf{r}_{C-D2}^N, \mathbf{C}^{N/B}) + \mathbf{C}^{N/B}\mathbf{a}_{SRP}^B + \mathbf{C}^{N/B}\frac{\mathbf{T}^B}{m} \\ \dot{m} = -\frac{|\mathbf{T}|}{I_{sp}g_0} \\ \mathbf{C}^{N/B} = f(\mathbf{r}) \end{cases} \tag{B.18}$$

B.2 Environmental Modelling

B.2.1 Gravitational Pull

The gravity field of the bodies D1 and D2 is modeled with spherical harmonic potential expansion. The potential of each is expressed in their respective body-attached RF. If D1, D2 RF are centered at the asteroids' CoM and oriented along their principal inertia axis, the first non-trivial function on the potential expansion is [1]:

$$U(\mathbf{r}_{c-D}^D) = \frac{\mu}{r}\left[1 + \left(\frac{1}{r}\right)^2 \left\{C_{20}\left(1 - \frac{3}{2}cos(\delta)^2\right) + 3C_{22}cos(\delta)^2 cos(2\lambda)\right\}\right] \tag{B.19}$$

hereafter called J2 Expansion, where λ, δ and r are the spherical coordinates used to represent the target point $\mathbf{r}$ in the Cartesian D RF. The C_{ij} noted in this work correspond to [1]:

$$C_{20} = \frac{1}{5}\left(\gamma^2 - \frac{\alpha^2 + \beta^2}{2}\right) \tag{B.20}$$

$$C_{22} = \frac{1}{20}\left(\alpha^2 - \beta^2\right) \tag{B.21}$$

where α, β and γ are the radii of the triaxial ellipsoid. Note that here the potential is defined as a positive quantity and therefore the resultant gravitational gorce acting over the s/c body $\mathcal{B}$ is defined as:

$$\mathbf{F}_g^D \triangleq \int_{\mathcal{B}} +\nabla^D U dm \tag{B.22}$$

where $\nabla^D(\cdot)$ is the gradient operator over the coordinates of the D RF. Using the J2 Expansion allows to analytically develop Eq. (B.22) and leave $\mathbf{F}_g$ in terms of the inertia tensor J. Using the transformation from spherical to Cartesian coordinates the potential can be expressed in Cartesian coordinates by the following developments:

$$\mathbf{r}^D_{c-D} = [x\ y\ z]^T \qquad\qquad cos(\delta)^2 = ... = \frac{r^2 - z^2}{r^2}$$

$$\begin{cases} rcos(\lambda)cos(\delta) = x \\ rsin(\lambda)cos(\delta) = y \\ rsin(\delta) = z \end{cases} \qquad \begin{aligned} &cos(2\lambda) = cos(\lambda)^2 - sin(\lambda^2) \\ &= sin(\lambda)^2 \left((\frac{x}{y})^2 - 1) \right) = ... = \frac{x^2 - y^2}{r^2 - z^2} \end{aligned}$$

and therefore:

$$U(\mathbf{r}^D_{c-D}) = \frac{\mu}{r} - \frac{\mu C_{20}(x^2 + y^2 - 2z^2)}{2r^5} + \frac{3\mu C_{22}(x^2 - y^2)}{r^5} = U_s + U_{NSG} \tag{B.23}$$

where the vector $\mathbf{r}$ points to a target point from the asteroid's CoM. Note that the first term is the classical gravitational potential of a spherically-distributed mass, and thus the orientation of the s/c position wrt. the body does not affect the potential's magnitude, only the distance to its center. The other two terms contain non-spherically-symmetric terms and therefore the position of the s/c wrt. to the body comes into play through the coordinates {x, y, z}. Both the spherical term and the non-spherical gravity (NSG) terms will yield Attitude-Orbit Coupling (see Chapter 2).

Gravitational Force

The spherical term integration in Eq. (B.22) can be developed as shown in [30]. This section walks through that work. Let now $R^D = [x\ y\ z]^T$ be the vector to a target-point from the asteroid's CoM; r be the vector to the same point from the s/c CoM; and R_c the vector to the s/c CoM from the asteroid CoM:

$$R = R_c + r \tag{B.24}$$

For clarity's sake the following notation simplification will be used:

- Vectors are notated in non-bold font. I.e. $\mathbf{r}^B_{c-D} \rightarrow r^B$
- A vector appearing in a denominator refers instead to the modulus of that vector. I.e. $1/|\mathbf{r}| \rightarrow 1/r$.

Consider first the spherical gravity term of Eq. (B.23) alone. Then, the Gravitational force that the whole body experiences is:

$$\mathbf{F}^D_{g,s} = \int_{\mathcal{B}} \frac{\partial U_s}{\partial R^D} dm = \int_{\mathcal{B}} -\frac{\mu}{R}(R^D_c + r^D)dm \rightarrow \mathbf{F}^B_{g,s} = \int_{\mathcal{B}} -\frac{\mu}{R}(R^B_c + r^B)dm$$

Let R^B_c, r^B and R^B be hereafter noted as R_c, r and R respectively. If the integral is performed on the B RF, by definition $\int_{\mathcal{B}} r^B dm = 0$ and thus:

$$\mathbf{F}^B_{g,s} = -\mu R_c \int_{\mathcal{B}} \frac{1}{|R_c + r|^3} dm$$

the term inside the integral can be approximated with a Second Order Taylor Expansion on the vicinity of the point $\mathbf{R}_c$:

$$|R_c + r|^{-3} = R_c^{-3} \left[1 + \frac{2R_c r}{R_c^2} + (\frac{r}{R_c})^2 \right]^{-3/2} = R_c^{-3} \cdot f(r)$$

$$\begin{cases} \frac{\partial f}{\partial r} = -\frac{3}{2} \left[1 + \frac{2R_c r}{R_c^2} + (\frac{r}{R_c})^2 \right]^{-5/2} \cdot \left(\frac{2R_c}{R_c^2} + \frac{2r}{R_c^2} \right) \\ \frac{\partial^2 f}{\partial r^2} = \frac{15}{4} \left[1 + \frac{2R_c r}{R_c^2} + (\frac{r}{R_c})^2 \right]^{-7/2} \cdot \left(\frac{2R_c}{R_c^2} + \frac{2r}{R_c^2} \right)^2 - \frac{3}{2} \left[1 + \frac{2R_c r}{R_c^2} + (\frac{r}{R_c})^2 \right]^{-5/2} \frac{2}{R_c^2} \end{cases}$$

$$f(r) \approx f(0) + \frac{\partial f}{\partial r}\Big|_{r=0} r + \frac{1}{2}\frac{\partial^2 f}{\partial r^2}\Big|_{r=0} (r \cdot r) = 1 - 3\frac{R_c r}{R_c^2} - \frac{3}{2}\frac{r \cdot r}{R_c^2} + \frac{15}{2}\left(\frac{R_c \cdot r}{R_c^2} \right)^2$$

And therefore:

$$\mathbf{F}_{g,s}^{B} = -\mu R_c \int_{\mathcal{B}} \frac{1}{|R_c + r|^3} dm \approx -\mu R_c \int_{\mathcal{B}} \frac{1}{R_c^3} \left[1 - 3\frac{R_c r}{R_c^2} - \frac{3}{2}\frac{r \cdot r}{R_c^2} + \frac{15}{2}\left(\frac{R_c \cdot r}{R_c^2}\right)^2 \right] dm$$

Note that all the terms above are of order $\mathbf{O}(R_c^{-4})$ or greater. Higher order terms have been implicitly discarded by using the Taylor Expansion up to Second Order. Using the identity $a \times (b \times c) = (a \cdot c)b - (a \cdot b)c$ allows to re-write the expression as:

$$\begin{aligned}\mathbf{F}_{g,s}^{B} = -\frac{\mu}{R_c^3}\Bigg\{ & mR_c - \frac{3}{R_c^2}\int \left[r \times (r \times R_c) + r^2 R_c \right] \mathrm{d}m - \frac{3}{2R_c^2}\int_{\mathcal{B}} r^2 R_c \, \mathrm{d}m \\ & + \frac{15}{2R_c^4}\int_{\mathcal{B}} R_c \cdot \left[r \times (r \times R_c) + r^2 R_c \right] R_c \, \mathrm{d}m \Bigg\}\end{aligned}$$

Realizing that $\int_{\mathcal{B}} r^2 dm = \frac{1}{2} tr(\mathbf{J}^B)$ and[1] $\mathbf{J} = \int_{\mathcal{B}} -[r]_\times [r]_\times dm$, the expression above can be re-expressed as:

$$\mathbf{g}_{D,s}^{B} = \mathbf{F}_{g,s}^{B}/m = -\frac{\mu}{|\mathbf{r}_{c-D}|^3}\mathbf{r}_{c-D}^{B} - 3\frac{\mu}{|\mathbf{r}_{c-D}|^5}\mathbf{\Gamma}^B r_{c-D}^{B} + \frac{15}{2}\frac{\mu(\mathbf{r}_{c-D}^{B,T}\mathbf{J}^B\mathbf{r}_{c-D}^{B})}{|\mathbf{r}_{c-D}|^7}\mathbf{r}_{c-D}^{B} \qquad \text{(B.25)}$$

where $\mathbf{\Gamma}^B = \frac{1}{2} tr(\mathbf{J}^B)\mathbf{I} + \mathbf{J}^B$ and the notation of $\mathbf{r}_{c-D} \equiv R_c$ has been recovered. Note that Eq. (B.25) reports the gravity acceleration on the s/c CoM exerted by only the spherical gravity field term. The first term in the RHS is the classical central force, while the other two terms are responsible for the spherical gravity AOC. The attitude of the s/c is introduced through the projection of $\mathbf{r}_{c-D}$ in the B RF. In order to reduce the computational burden the gravitational acceleration will hereafter be evaluated in the N RF directly, and therefore the information on the s/c attitude will enter through the projection of the tensor J on the N RF: $\mathbf{J}^N = \mathbf{C}^{N/B}\mathbf{J}^B(C^{N/B})^T$.

The gravity acceleration terms coming from the NSG part of the potential expansion can be retrieved from [12] and are:

$$\mathbf{g}_{NSG,D}^{D} = \begin{bmatrix} -\frac{\mu C_{20} x_c}{|r_{c-D}|^5} + \frac{5\mu C_{20} x_c \left(x_c^2 + y_c^2 - 2z_c^2\right)}{2|r_{c-D}|^7} + \frac{6\mu C_{22} x_c}{|r_{c-D}|^5} - \frac{15\mu C_{22} x_c \left(x_c^2 - y_c^2\right)}{|r_{c-D}|^7} \\ -\frac{\mu C_{20} y_c}{|r_{c-D}|^5} + \frac{5\mu C_{20} y_c \left(x_c^2 + y_c^2 - 2z_c^2\right)}{2|r_{c-D}|^7} - \frac{6\mu_n C_{22} y_c}{|r_{c-D}|^5} - \frac{15\mu C_{22} y_c \left(x_c^2 - y_c^2\right)}{|r_{c-D}|^7} \\ \frac{2\mu C_{20} z_c}{|r_{c-D}|^5} + \frac{5\mu C_{20} z_c \left(x_c^2 + y_c^2 - 2z^2\right)}{2|r_{c-D}|^7} - \frac{15\mu C_{22} z_c \left(x_c^2 - y_c^2\right)}{|r_{c-D}|^7} \end{bmatrix} \qquad \text{(B.26)}$$

where $r_{c-D}^D = [x_c \ y_c \ z_c]^T$. Note that in the expression above the integral hasn't been re-expressed with $\mathbf{J}^B$ because it has been performed on the D RF, retaining again terms up to order $\mathbf{O}(|r_{c-D}|^{-4})$.

Finally, the total gravitational acceleration exerted by the asteroid D on the s/c CoM, expressed in the N RF is:

$$\begin{aligned}\mathbf{g}_D^N &= \mathbf{g}_{s,D}^N + \mathbf{C}^{N/D}\mathbf{g}_{NSG,D}^D \\ &= -\frac{\mu}{|\mathbf{r}_{c-D}|^3}\mathbf{r}_{c-D}^N - 3\frac{\mu}{|\mathbf{r}_{c-D}|^5}\mathbf{\Gamma}^N r_{c-D}^N + \frac{15}{2}\frac{\mu(\mathbf{r}_{c-D}^{N,T}\mathbf{J}^N\mathbf{r}_{c-D}^N)}{|\mathbf{r}_{c-D}|^7}\mathbf{r}_{c-D}^N + \mathbf{C}^{N/D}\mathbf{g}_{NSG,D}^D\end{aligned}$$

where $\mathbf{J}^N = \mathbf{C}^{N/B}\mathbf{J}^B(C^{N/B})^T$; $\mathbf{\Gamma}^N = \mathbf{C}^{N/B}\left[\frac{1}{2} tr(\mathbf{J}^B)\mathbf{I} + \mathbf{J}^B\right](C^{N/B})^T$; and $\mathbf{g}_{NSG,D}^D$ is available from Eq. (B.26).

[1] The operator $[\cdot]_\times$ yields the skew-matrix.

Gravitational Torque

Let again $R = [x\ y\ z]^T$ be the vector to a target-point from the asteroid's CoM expressed in the D RF; r be the vector to the same point from the s/c CoM; and R_c the vector to the s/c CoM from the asteroid CoM:

$$R = R_c + r \tag{B.27}$$

For the moment, consider only the spherical term of U^D. The gravitational torque with respect to the s/c CoM, created by the spherical gravity of D, in the D RF is:

$$\begin{aligned}\mathbf{M}^D_{g,s} &= \int_{\mathcal{B}} r \times \frac{\partial U^D}{\partial R^D} dm = \int_{\mathcal{B}} r \times -\frac{\mu R}{|R|^3} dm = \int_{\mathcal{B}} r \times -\frac{\mu (R_c + r)}{|R_c + r|^3} dm \\ &= \int_{\mathcal{B}} r \times -\frac{\mu R_c}{|R_c + r|^3} dm = R_c \mu \times \int_{\mathcal{B}} \frac{r}{|R_c + r|^3} dm\end{aligned}$$

The denominator can be approximated with a First Order Taylor Expansion using the same procedure proposed for the gravitational force section. This yields:

$$\mathbf{M}^D_{g,s} \approx R_c \mu \times \int_{\mathcal{B}} \frac{r}{R_c^3} \left[1 - 3\frac{R_c r}{R_c^2} \right] dm = \frac{3\mu}{R_c^5} R_c \times \mathbf{J}^D R_c \tag{B.28}$$

And, expressing the moment in the B RF while recuperating the general vector notation:

$$\mathbf{M}^B_{g,s} = \frac{3\mu}{|r_{c-D}|^5} \mathbf{r}^B_{c-D} \times \mathbf{J}^B \mathbf{r}^B_{c-D} \tag{B.29}$$

Hereafter, as in the gravitational force section, only the terms of order $\geq \mathbf{O}(|\mathbf{r}_{c-D}|^{-4})$ are retained. This implies that the non-spherical gravity terms do not contribute to the gravitational torque [12] and therefore:

$$\mathbf{M}^B_{g,D} = \frac{3\mu}{|r_{c-D}|^5} \mathbf{r}^B_{c-D} \times \mathbf{J}^B \mathbf{r}^B_{c-D} \tag{B.30}$$

B.2.2 SRP acceleration and torque

The SRP acceleration has been modeled using the cuboid model, available in [29] and tested in [12]:

$$\mathbf{F}_i = -PA_i(\hat{\mathbf{S}}^B \cdot \mathbf{n}_i) \left[(1-\rho_s)\hat{\mathbf{S}}^B_i + (2\rho_s(\hat{\mathbf{S}}^B \cdot \mathbf{n}^B_i) + \frac{2}{3}\rho_d)\mathbf{n}^B_i \right] \tag{B.31}$$

$$\mathbf{F}^B_{SRP} = \sum_{i=1}^{N} \mathbf{F}_i \tag{B.32}$$

$$\mathbf{M}^B_{SRP} = \sum_{i=1}^{N} \mathbf{r}^B_i \times \mathbf{F}_i \tag{B.33}$$

Where $\mathbf{n}^B_i$ is the unit vector normal to the each lighted surface; and r^B_i is the vector $\overline{\mathrm{CoM_{sc}}, \mathrm{GC_i}}$. In the summation of Eq. (B.31) only the forces with $(\hat{\mathbf{S}}^B \cdot \mathbf{n}_i) \geq 0$ are retained. The following assumptions have been done:

- The direct solar radiation pressure has a value of $4.539807335646850e-06\ Pa$.
- The Asteroid's radiation and its Albedo have been neglected.
- The shadow of the s/c upon its own surfaces has been neglected.
- The Absorption, Diffuse Reflection and Specular reflection coefficients are homogeneous through the surface of each element.
- The solar panels' thickness-wise area is negligible.
- The center of pressures (CoP) of each element coincides with its CoM.

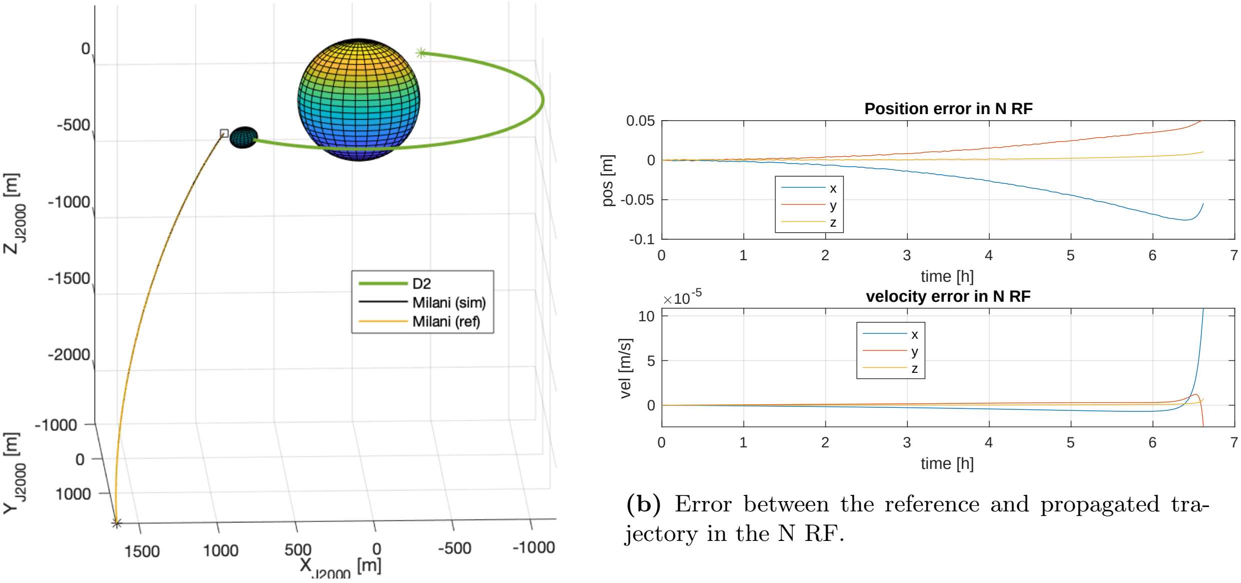
The diffuse (d) and specular (s) emission coefficients for the have been set to:

Table B.2: Radiation emission coefficients

	Body	Yoke	Panel
ρ_s	0.75	0.8	0.09
ρ_d	0.15	0.1	0.01

B.2.3 Validation of environmental forces

The modelled environment has been validated by comparing the propagated trajectory and the forces acting on the s/c to the values provided by a reference trajectory and the dynamical study in [25]. Importantly, in [25] is used the Cannonball model to evaluate the SRP. In order to approximate the results to those of [25] as much as possible, for the validation process the SRP has been modeled using the Cannon Ball Model too. Also, no control has been applied on the s/c; the attitude has been left to change freely; and no perturbation have been introduced. The reference trajectory as well as the propagated one can be observed in Fig. B.1a. There, it can be observed that the propagation error is remarkably small. The latter can be better appreciated in Fig. B.1b, where is shown how over $\approx 7\ h$ of simulation the error in position has the order of $\mathcal{O}(cm)$ and the error in velocity the order of $\mathcal{O}(10^{-2}\ mm/s)$.In Fig. B.2 can be observed the evolution of the acceleration exerted by the different perturbations upon the s/c. It is consistent with the dynamical study of [25] and therefore the model is accepted as validated.



(b) Error between the reference and propagated trajectory in the N RF.

(a) Reference and propagated trajectory expressed in the N RF frame.

Figure B.1: Comparison between reference and propagated trajectory for the validation process. SRP model: Cannon Ball. Without control on translational and rotational dynamics.

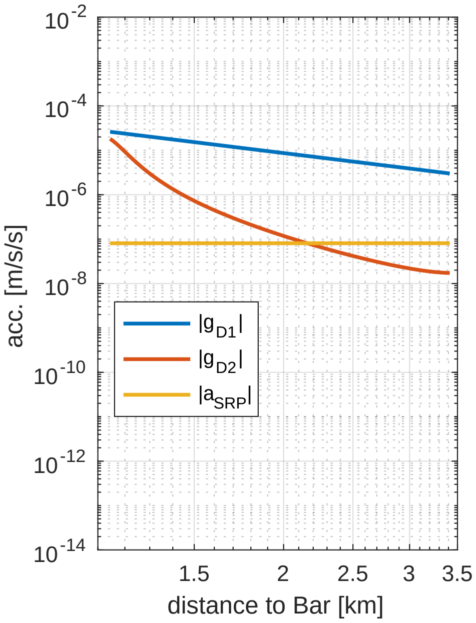


Figure B.2: Evolution of the perturbations over distance. SRP model: Cannon Ball.

B.3 Control implementation

The objective of this section is to synthesize the control implementation, the rationale of which has been introduced and discussed in section 4.3. A different notation wrt. environmental modelling has been adopted for clarity's sake. Notation changes as follows:

- Vectors and matrices are expressed in non-bold font.
- All vectors are expressed in the D2 RF unless stated otherwise.

$$\mathbf{r}^{D2}_{c-D2} \to r, \qquad \mathbf{r}^{D2}_{D2-D} \to R_a, \qquad \mathbf{a}^{D2}_{c} \to u$$

and note that:

$$\nu = \dot{r} = \left.\frac{dr}{dt}\right|_{D2} = \left.\frac{dr}{dt}\right|_{N} - \omega_{D2/N} \times r = \mathbf{v}^{D2}_{c-D2} - \omega^{D2}_{D2/N} \times r^{D2}$$

B.3.1 MSSG formula

The Multiple Sliding Surface Guidance (MSSG) algorithm has been implemented by evaluating the function below at every time step:

$$\boxed{u = -\Phi sigmoid(s_2) - \frac{\Lambda}{t_{go}}\dot{s}_1 - \frac{\Lambda}{t_{go}^2}s_1 + g_{NI}(r,t) \quad \text{(B.34)}}$$

The definition of the variables appearing on the MSSG expression are:

$$\begin{aligned} t_{go} &= t_f - t & \dot{s}_1 &= \nu - \nu_L \\ s_1 &= r - r_L & s_2 &= \dot{s}_1 + \frac{\Lambda}{t_{go}}s_1 \end{aligned} \tag{B.35}$$

And, since the problem is vectorial (i.e. $s_1 \in \mathbb{R}^{3\times1}$):

$$g_{NI}(t,r) = 2\omega \times \nu + \omega \times \omega \times r + \ddot{R}_a - g(r,t) \tag{B.36}$$

$$\Phi = \Phi I_{3\times3}, \quad \Phi = max\left(\frac{|s_{2,i}(0)|}{\Delta t_r}, |p_{max}|\right) \tag{B.37}$$

$$\Lambda = \Lambda I_{3\times3} \tag{B.38}$$

$$sigmoid(s_2) = \frac{s_2}{|s_2| + \epsilon}, \quad \epsilon =: sigmoid(s_{2,low}) = 0.01 \tag{B.39}$$

Note that:

- For all simulations the control is completely turned off 5 s before touch-down to avoid a burst in the control when $t_{go} \to 0$.
- The gains of the control are Λ and $\Phi \propto 1/\Delta t_r = 1/n(t_f - t_0) \propto 1/n$. Hence, the tuning of the MSSG will be fundamentally based on the choice of

$$\Lambda \in [2,\ \infty)] \tag{B.40}$$

$$n \in (0,\ 1] \tag{B.41}$$

B.3.2 Chattering-avoidance

The perturbation-rejection term $a_{smc} = \Phi sigmoid(s_2)$ is activated using a Schmitt trigger with:

$$s_{2,high} = 10^{-2}\ m/s$$
$$s_{s,low} = 10^{-4}\ m/s$$

B.3.3 Perturbing accelerations' model

The dynamics described by Eq. (B.18) have been artificially perturbed using randomly-generated perturbation vectors $a_p(t)$, whose components' magnitude follows a normal distribution as:

$$a_p \in \mathbb{R}^{3\times 1} \sim \mathcal{N}\left(0\ m/s^2, (|p_{max}|/3)^2\right)$$

and $|p_{max}|$ has been assumed to be

$$|p_{max}| = 10^{-5}\ m/s^2$$

For a given trajectory simulation $a_p = c.t.$

B.3.4 Thrust limitation and quantization

For the implementation of Milani's thrusters (i.e. not during the MSSG analysis), in this work the thrust has been saturated and quantized in the B RF. Central to this process is the fact that the GNC system has a command frequency of $f_{gnc} = 1Hz$. Some properties of the thrusters are yet to be defined at the time of writing. Indeed, the Minimum Impulse Bit of Milani's thruster has been assumed to be equal to the one of a commercial 6DOF control thruster with equal maximum thrust. The properties are the following:

- Minimum Impulse Bit (MIB): $25\mu Ns$. Assumed to be equal to a 6DOF cold gas propulsion system from GOMSpace [31].
- Minimum Increase in Thrust: $MIB \cdot f_{gnc} = 25\mu N$
- Minimum Thrust: $25\mu N$
- Maximum Thrust: $10mN$. Extracted from Milani's Mission Analysis Report [26].

Appendix C

Landing Greater Navigation Error

In this appendix is discussed the effect of using a greater navigation error in position, which is closer to what can be extracted from from [26] at the time of writing. From [26] it is known that LIDAR measurements on D1 have an error of

$$\sigma_{LIDAR} \approx 10m \tag{C.1}$$

An effort to consider this value for σ_r while applying the TPD strategy presented in sec. 5.1 has been made. To use $\sigma_r \approx 10m$ induces:

- The perceived landing point (PLP) by the navigation system will fall inside a sphere with radius $\approx 30m$ with a 99.7% of probability (3-σ property of Gaussian distributions).

This means that the MSSG would leave the s/c with high performance (i.e. $r,\ \nu \approx 0$) at a point up to $30m$ above the surface. This implies that

- The position that the Control system targets can diverge significantly from the actual target, and with relatively great magnitude.
- The s/c can then perform a free fall of $30m$ towards the surface.
- The s/c might be left with an out-ward velocity. This happens if, simultaneously:
    1. The PLP is above the surface.
    2. The s/c surpasses the PLP and has to go back.
    3. The control has still firing time to push the s/c up towards the surpassed PLP.

An example of this problem is shown in Fig. C.1a. To tackle these issues, the TBD has been modified:

1. The boundary layer is to be set at $30m$, so that the landing point is perceived to be within the boundary with 99.7% probability.
2. Once the s/c perceives that the boundary layer has been crossed, a new measurement of the LIDAR is to be done on D2 with:

   $$\sigma_{LIDAR} = 1m \tag{C.2}$$

   This allows the MSSG to re-target to a point closer to the actual landing target.
3. Since $h \uparrow$ to $h = 30$, the descent in $300s$ would need an average velocity of $\bar{\nu} = 30m/300s = 10cm/s$. The s/c typically reaches the boundary layer at $\mathcal{O}(5cm/s)$, and hence to reach r_L in $300s$ the control needs to first accelerate greatly the s/c and then quickly break

its speed to reach r_L with $\nu = 0$ (see v_y in Fig. C.1b). This rises V_c significantly, up to $10cm/s$ in the case of Fig. C.1a. To tackle this, the Descent Phase firing is still limited to $300s$, but the t_f that the MSSG uses is extended to $1200s$, so that $t_{go} \uparrow$ and minimization of ν primes over r while minimizing[1]

$$s_2 = \dot{s}_1 + \frac{\Lambda}{t_{go}} s_1 = \nu(t) + \frac{\Lambda}{t_{go}}(r(t) - r_L) \tag{C.3}$$

Note that the increase on t_f to $1200s$ has been done because $\sigma_r = 30m \Rightarrow h = 30m \Rightarrow \bar{\nu} \uparrow \Rightarrow t_f \uparrow$.

This modified TPD performs reasonably well in front of such strong error in Navigation knowledge, the changed version of Fig. C.1a is shown in Fig. C.2. In that case $V_c = 4.3cm/s$. There, the control is limited to greately stop the s/c during the $300s$ after the perceived boundary layer crossing, and then the s/c free falls towards the surface.

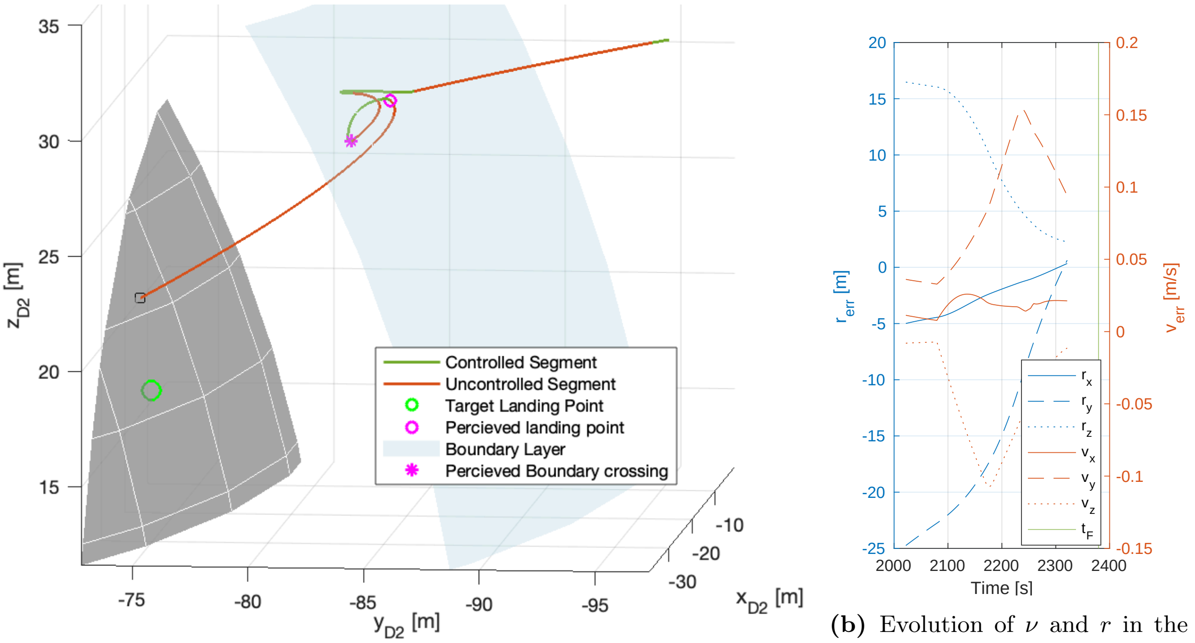
From this study it can be concluded that:

- The value of σ_r determines how close to the surface is the PLP. After reaching the PLP, the s/c performs a free fall towards the surface. High values of σ_r may imply that:
  - The s/c performs a free fall from a great height, increasing V_c.
  - The s/c reaches the PLP with an acceleration out-ward from D2.
- Requirements on the maximum free-fall distance may impose a requirement on σ_r. In this work, the MSSG parameters $\Lambda_2 = 3$ and $n = 0.7$, as well as the boundary layer height $h = 15m$ were selected while considering perfect state knowledge. To add the state knowledge *a posteriori*, a σ_r has to be enforced s.t. the PLP is within the boundary layer of $h = 15m$.
- The TBD strategy can be adapted to use greater values of σ_r (e.g $\sigma_r \geq 10\ m$), which needs to use a low σ_r close to the surface anyway, and an artificial increase of t_f in the MSSG to avoid a peak in ν.

To use the modified TBD strategy has finally been discarded because:

- The choice of $\sigma_r = 1m$ for the whole trajectory would allow the choice of $h = 15m$ (as in sec. 5.1).
- For the reasons explained above, land with $\sigma_r = 10m$, during the descent phase σ_r has to be $\mathcal{O}(1m)$ anyway, and hence $\sigma_r = 1m$ has to be imposed at some point. Then, it reasonable to already require $\sigma_r = 0$ for the whole descent and avoid using an h of $h = 50m$.

[1] See 4.3



(a) Last moments of the trajectory in the D2 RF.

(b) Evolution of ν and r in the last stages of Fig. C.1a.

Figure C.1: TBD used with a Navigation knowledge of $\sigma_r = 10m$. The simulation stops at collision.

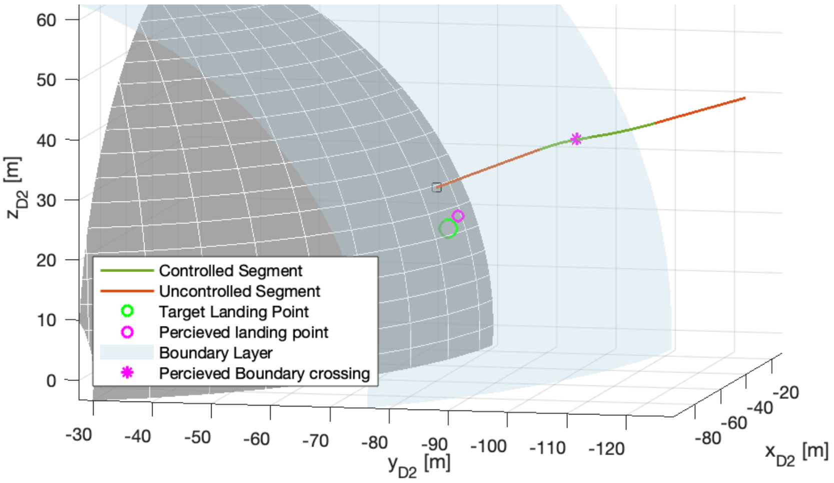


Figure C.2: TBD starting with a Navigation knowledge of $\sigma_r = 10m$, and setting it to $\sigma_r = 1m$ after the boundary layer, with $t_f = 1200s$ during the descent phase, but a firing time of $300s$.

List of Figures

List of Tables

Bibliography

[1] D. J. Scheeres, *Orbital Motion in Strongly Perturbed Environments*, 1st ed. New York: Springer, 2012 (cit. on pp. 1, 4, 5, 7, 24, 67).

[2] D. Brownlee, “The Stardust Mission: Analyzing Samples from the Edge of the Solar System,” *http://dx.doi.org/10.1146/annurev-earth-050212-124203*, vol. 42, pp. 179–205, Jun. 2014. DOI: 10.1146/ANNUREV-EARTH-050212-124203. [Online]. Available: https://www.annualreviews.org/doi/abs/10.1146/annurev-earth-050212-124203 (cit. on p. 1).

[3] G. Ono, F. Terui, N. Ogawa, *et al.*, “GNC strategies and flight results of Hayabusa2 first touchdown operation,” *Acta Astronautica*, vol. 174, no. January, pp. 131–147, 2020. DOI: 10.1016/j.actaastro.2020.04.029 (cit. on pp. 1, 16, 29, 60, 61).

[4] D. S. Lauretta, S. S. Balram-Knutson, E. Beshore, *et al.*, “OSIRIS-REx: Sample Return from Asteroid (101955) Bennu,” *Space Science Reviews*, vol. 212, no. 1-2, pp. 925–984, Oct. 2017. DOI: 10.1007/S11214-017-0405-1 (cit. on p. 1).

[5] J. P. Sanchez and C. R. McInnes, “Assessment on the feasibility of future shepherding of asteroid resources,” *Acta Astronautica*, vol. 73, pp. 49–66, Apr. 2012. DOI: 10.1016/J.ACTAASTRO.2011.12.010 (cit. on p. 1).

[6] C. R. Chapman, “The hazard of near-Earth asteroid impacts on earth,” *Earth and Planetary Science Letters*, vol. 222, no. 1, pp. 1–15, May 2004. DOI: 10.1016/J.EPSL.2004.03.004 (cit. on p. 1).

[7] D. C. Richardson, O. S. Barnouin, L. A. M. Benner, *et al.*, “Dynamical and Physical Properties of 65803 Didymos, the Proposed AIDA Mission Target,” in *AAS/Division for Planetary Sciences Meeting Abstracts #48*, ser. AAS/Division for Planetary Sciences Meeting Abstracts, vol. 48, Oct. 2016 (cit. on pp. 1, 6, 7, 34).

[8] The Johns Hopkins University Applied Physics Laboratory, *Double Asteroid Redirection Test*, 2021. [Online]. Available: https://dart.jhuapl.edu/Mission/index.php (cit. on p. 1).

[9] ESA Earth Observation Portal, *Hera - the asteroid deflection mission of ESA*, 2021. [Online]. Available: https://directory.eoportal.org/web/eoportal/satellite-missions/content/-/article/hera (cit. on p. 1).

[10] F. Ferrari, V. Franzese, M. Pugliatti, *et al.*, “Preliminary mission profile of Hera’s Milani CubeSat,” *Advances in Space Research*, vol. 67, no. 6, pp. 2010–2029, 2021. DOI: 10.1016/j.asr.2020.12.034. [Online]. Available: https://doi.org/10.1016/j.asr.2020.12.034 (cit. on pp. 1, 46).

[11] D. Ge, P. Cui, and S. Zhu, “Recent development of autonomous GNC technologies for small celestial body descent and landing,” *Progress in Aerospace Sciences*, vol. 110, no. July, p. 100 551, 2019. DOI: 10.1016/j.paerosci.2019.06.002. [Online]. Available: https://doi.org/10.1016/j.paerosci.2019.06.002 (cit. on pp. 2, 8, 9, 15, 16, 60–62).

[12] G. Misra, M. Izadi, A. Sanyal, *et al.*, "Coupled orbit-attitude dynamics and relative state estimation of spacecraft near small Solar System bodies," *Advances in Space Research*, vol. 57, no. 8, pp. 1747–1761, 2016. DOI: `10.1016/j.asr.2015.05.023`. [Online]. Available: `http://dx.doi.org/10.1016/j.asr.2015.05.023` (cit. on pp. 2, 6, 69, 70).

[13] F. Ferrari and M. Lavagna, "Consolidated phase a design of Asteroid Impact Mission: MASCOT-2 landing on binary asteroid didymos," in *26th AAS/AIAA Space Flight Mechanics Meeting*, vol. 158, 2016, pp. 3759–3769 (cit. on p. 3).

[14] G. B. Sincarsin and P. Hughes, "Gravitational orbit-attitude Coupling for very large spacecraft," *Celestial Mechanics*, vol. 31, no. August, pp. 143–161, 1983 (cit. on p. 6).

[15] P. Simplício, A. Marcos, E. Joffre, *et al.*, "Review of guidance techniques for landing on small bodies," *Progress in Aerospace Sciences*, vol. 103, no. January, pp. 69–83, 2018. DOI: `10.1016/j.paerosci.2018.10.005`. [Online]. Available: `https://doi.org/10.1016/j.paerosci.2018.10.005` (cit. on pp. 8, 11–17, 19, 20).

[16] R. Furfaro, B. Gaudet, D. R. Wibben, *et al.*, "Development of non-linear guidance algorithms for asteroids close-proximity operations," *AIAA Guidance, Navigation, and Control (GNC) Conference*, pp. 1–18, 2013. DOI: `10.2514/6.2013-4711` (cit. on pp. 13, 14, 19).

[17] R. Furfaro, D. Cersosimo, and D. R. Wibben, "Asteroid precision landing via multiple sliding surfaces guidance techniques," *Journal of Guidance, Control, and Dynamics*, vol. 36, no. 4, pp. 1075–1092, 2013. DOI: `10.2514/1.58246` (cit. on pp. 14, 15, 19, 22, 25, 26, 57, 58).

[18] D. Cersosimo, J. Bellerose, and R. Furfaro, "Sliding guidance techniques for close proximity operations at multiple asteroid systems," *AIAA Guidance, Navigation, and Control (GNC) Conference*, pp. 1–16, 2013. DOI: `10.2514/6.2013-4712` (cit. on p. 15).

[19] R. Furfaro, D. R. Wibben, B. Gaudet, *et al.*, "Terminal multiple surface sliding guidance for planetary landing: Development, tuning and optimization via reinforcement learning," *Journal of the Astronautical Sciences*, vol. 62, no. 1, pp. 73–99, 2015. DOI: `10.1007/s40295-015-0045-1` (cit. on pp. 15, 19).

[20] S. Bhaskaran, S. Nandi, S. Broschart, *et al.*, "Small body landings using autonomous onboard optical navigation," *Journal of the Astronautical Sciences*, vol. 58, no. 3 PART 1, pp. 409–427, 2011. DOI: `10.1007/bf03321177` (cit. on pp. 16, 60, 61).

[21] R. Furfaro, private communication, Jul. 2021 (cit. on p. 19).

[22] Y. Shtessel, C. Edwards, L. Fridman, *et al.*, *Sliding mode control and observation.* 2014, pp. 1–356. DOI: `10.1007/978-0-8176-4893-0` (cit. on pp. 23, 24).

[23] J. Guldner and V. I. Utkin, "The chattering problem in sliding mode systems," *Fourteenth intenational symposium of mathematical theory of networks and systems, MTNS2000*, 2000 (cit. on p. 24).

[24] O. Çelik, O. Karatekin, B. Ritter, *et al.*, "Reliability Analysis of Ballistic Landing in Binary Asteroid 65803 (1996GT) Didymos under Uncertainty and GNC Error Considerations," *26th International Symposium on Spaceflight Dynamics*, 2017 (cit. on pp. 34, 48).

[25] F. Ferrari, V. Franzese, M. Pugliatti, *et al.*, "Trajectory Options for Hera ' s Milani CubeSat Around (65803) Didymos," *The Journal of the Astronautical Sciences*, 2021. [Online]. Available: `https://doi.org/10.1007/s40295-021-00282-z` (cit. on pp. 34, 35, 65, 71).

[26] Politecnico di Milano and Tyvak International, "Milani CubeSat Development: Mission Analysis Report," Politecnico di Milano, Tyvak International, Tech. Rep. 3, 2021, pp. 73–80 (cit. on pp. 46–48, 52, 74, 75).

[27] R. W. Gaskell, O. S. Barnouin-Jha, D. J. Scheeres, *et al.*, "Characterizing and navigating small bodies with imaging data," *Meteoritics and Planetary Science*, vol. 43, no. 6, pp. 1049–1061, 2008. DOI: 10.1111/j.1945-5100.2008.tb00692.x (cit. on pp. 60, 61).

[28] D. Adams, T. B. Criss, and U. J. Shankar, "Passive optical Terrain Relative Navigation using APLNav," *IEEE Aerospace Conference Proceedings*, 2008. DOI: 10.1109/AERO.2008.4526303 (cit. on p. 61).

[29] B. Wie, *Space Vehicle Dynamics and Control*, 2nd, J. A. Schetz, Ed. American Institute of Aeronautics and Astronautics, Inc., 2008, pp. 336, 345, 749. DOI: 10.1108/aeat.1998.12770eae.008 (cit. on pp. 65, 70).

[30] H. Schaub and J. L. Junkins, *Analytical Mechanics of Space Systems*, 2nd, J. A. Schetz, Ed. AIAA Education Series Editor-in-Chief, 2009, pp. 188–193 (cit. on p. 68).

[31] GomSpace, *NanoProp 6DOF Technical Sheet.* [Online]. Available: gomspace.com (cit. on p. 74).