

Design of Convex Guidance for the Final Phase of Satellite Rendezvous

Tested using Hardware-in-the-Loop
Simulations

S. Vromen

Technische Universiteit Delft

Design of Convex Guidance for the Final Phase of Satellite Rendezvous

Tested using Hardware-in-the-Loop Simulations

by

S. Vromen

in partial fulfillment of the requirements for the degree of

Master of Science

in Aerospace Engineering

at the Delft University of Technology,

to be defended publicly on Thursday August 27, 2015 at 10:30 AM.

Supervisor:	Dr. ir. E. Mooij,	TU Delft
Thesis committee:	Prof. dr. ir. P. N. A. M. Visser,	TU Delft
	Dr. Q. Chu,	TU Delft
	Ir. F. de Bruijn,	DLR Bremen

An electronic version of this thesis is available at <http://repository.tudelft.nl/>.

"Winners never quit and quitters never win"

- Vince Lombardi

Acknowledgements

A great deal of effort has gone into the realization of this thesis, especially when all the years of education that led up to this final project are considered. This thesis could not have been completed without the support and assistance of many people. I would like to start by expressing my gratitude to the people who have had a direct influence on my work. First, and foremost, I would like to thank my direct supervisor at the TU Delft, Erwin Mooij, for being a great advisor and for always being readily available for discussions. I would also like to thank Ferdi de Bruijn, at DLR, for bringing me on board at the DLR Institute of Space Systems in Bremen and for taking the time to guide me through the worlds of convex optimization, model predictive control, and TEAMS. Your great feedback and tutoring skills are very much appreciated. Furthermore, I would like to thank Markus Schlotterer, at DLR, for teaching me the ropes of TEAMS and assisting me at times when the facility seemed to have decided to live a life of its own.

For me being a student is not just about academics, but also about seizing as many opportunities as you can to expand your horizon. Over the past years, the lessons I have learned from the projects I was involved in and the people I have worked with ended up being, in many ways, even more important than what I have learned from my classes. For these shaping times and most of all for the amazing fun we have had, I want to thank all my friends from the VSV, the 66th board, Orient, RVD24, JC Stuk, Dopie, Tenniphil, the VSVrouwenboot, Mango and my study buddies from the 9th floor (especially Harry, my personal professor). Being a part of all these groups have truly made my years in Delft and I will cherish all the memories we have made.

Finally, I want to thank my family. Thank you Laura and Rick, for your friendship, you will always be my big sister and baby brother. And most importantly, I want to thank my parents, for being such great people. Mom and dad, without your continuing love, support, and encouragement I would not have been able to do any of this and I am forever thankful for the loving home you provide.

Summary

Over the past decades the steep increase in the number of space-debris objects have sparked many studies into active space-debris removal. Past collisions have shown that space debris is a very real threat to operational satellites and active debris removal is necessary to stabilize the situation. ESA has, therefore, performed multiple studies on the removal of the no longer operational satellite ENVISAT from orbit. Since ENVISAT lost functionality, it has acquired a tumbling motion, which poses many challenges for a removal mission. A scenario was proposed in which a chaser spacecraft performs a rendezvous with ENVISAT, to actively de-orbit the satellite. To increase the robustness to unexpected events and reduce operational costs it is highly desirable that these operations can be performed autonomously.

In this research an autonomous guidance and orbit control system was developed that enables the final phase of the rendezvous with ENVISAT. An approach strategy was adopted that aims to maintain alignment with the spin axis of ENVISAT throughout the approach. The designed algorithms use convex guidance to ensure globally-optimal solutions, while at the same time constraining the trajectory. The guidance algorithm minimizes a weighed combination of the thrust and the state error. Furthermore, the concept of model predictive control is applied to allow for an unconstrained time-to-go. Two guidance and control strategies were examined in this research, one that is solely based on model predictive control and another that employs an additional LQR-controller. The functional simulations were complemented by hardware-in-the-loop simulations using the DLR flat-floor test facility, TEAMS. These real-time tests also include the docking phase.

Both the functional and hardware-in-the-loop simulation results show that the baseline model predictive control method can successfully perform the operations with an accuracy well above the requirement. It was found that implementing an extra LQR controller resulted in a similar accuracy, but an increase in ΔV of 21%. The functional simulation results also revealed that the required ΔV , compared to the baseline, could be decreased further by 18%, while maintaining the baseline accuracy. This is achieved by optimizing the weight parameters used in the convex optimization. It does, however, lead to an increased mission duration. The designed algorithms were successfully implemented on TEAMS, where it was shown that the addition of an LQR controller again leads to a higher use of propellant and on top of that it results in a longer operation time.

Table of Contents

List of Abbreviations	xiii
List of Symbols	xiii
1 Introduction	1
1-1 Motivation	1
1-2 Research Question and Scope	3
1-3 Research Methodology	3
1-4 Thesis Roadmap	4
2 Mission Heritage and Reference Scenario	5
2-1 Mission Heritage	5
2-1-1 Active Debris Removal	5
2-1-2 Rendezvous and Docking	6
Early Rendezvous and Docking	6
Engineering Test Satellite VII	7
XSS-10 and XSS-11	8
DART	9
Orbital Express	9
ATV	10
2-2 Reference Mission Scenario	10
2-2-1 Target and Chaser Satellite	11

2-2-2	Approach Overview and Requirements	12
2-3	HIL-Simulation Environment	13
2-3-1	Examples of Flat-floor Testbeds	14
	Flight Robotics Laboratory of NASA	14
	Space Systems Laboratory of MIT	14
	Spacecraft Robotics Laboratory of NRL	15
3	Orbital Mechanics	17
3-1	Reference Frames	17
3-1-1	ECI Frame	18
3-1-2	Rotating Hill Frame	18
3-1-3	Transformation between ECI and Rotating Hill Frame	18
3-2	State Representations	19
3-2-1	Cartesian Coordinates	20
3-2-2	Keplerian Elements	20
3-3	Equations of Motion	21
3-3-1	Absolute Motion	21
3-3-2	Relative Motion	23
3-4	Relative Perturbations	24
3-4-1	Perturbations Models	24
3-4-2	Magnitude of Relative Perturbations	26
4	GNC for Rendezvous	27
4-1	Tasks and Functions of the GNC System	28
4-1-1	Definitions	28
4-1-2	Architecture and System Interaction	28
4-2	Guidance and Control System	29
4-3	Guidance and Control Algorithms	30
4-4	Guidance Problem Formulation for the Reference Scenario	33
4-5	Implemented Guidance and Control Strategies	33

5	Convex Guidance	37
5-1	Convex Optimization	37
5-1-1	Convex Sets	38
5-1-2	Convex Functions	40
5-1-3	Convex Optimization Problems and Second-Order Cone Programs	41
5-2	Formulation Convex Optimization Problem for the Reference Scenario	43
5-2-1	Discretization	43
	Different Discretization Time Steps	44
	Discrete Problem Formulation	45
5-2-2	Formulation Safety Constraint	46
6	Functional Simulator Development	49
6-1	Software Architecture	49
6-2	Guidance and Control System Development and Implementation	50
6-2-1	Guidance System	50
6-2-2	Generation of the Analytical Reference Trajectory	50
6-2-3	Implementation of the Convex Optimizer	52
6-2-4	Control System	53
6-3	State Propagation	54
6-4	Software Verification	54
6-4-1	Dynamics Model	55
6-4-2	Guidance System	55
6-4-3	Control System	56
7	TEAMS	59
7-1	Specifications and Previous Projects	59
7-2	Adaptations required for TEAMS	61
7-2-1	Reference Scenario	61
7-2-2	Dynamics	62
7-2-3	Optimization Problem Formulation	62

8	TEAMS Simulator Development	63
8-1	Software Architecture	63
8-2	Guidance and Control System Development and Implementation	64
8-2-1	Guidance System	64
	Generation of Analytical Reference Trajectories	64
	Implementation of the Convex Optimizer	67
	Guidance for the Attitude	67
8-2-2	Control System	68
8-2-3	Target Guidance and Control System	68
8-3	Verification	68
9	Test Campaigns and Results	71
9-1	Functional Simulations Test Campaign	71
9-1-1	Optimization Problem Size Reduction	72
9-1-2	General Simulation Parameters	73
9-2	Functional Simulations Results	73
9-2-1	Performance of the Guidance and Control Strategies	73
9-2-2	Optimization Analysis of the MPC Method	76
	Trade-off between Minimum State Error and Minimum Thrust	76
	Thrust Minimization using the 1-Norm	78
	Combined Approach Analysis	80
9-2-3	Validation of the Results	82
9-3	TEAMS Test Campaign	84
9-4	TEAMS Results	84
9-4-1	Analytical Reference Trajectories	85
9-4-2	Performance of the Guidance and Control System	87
10	Conclusion and Recommendations	91
10-1	Conclusions	92
10-1-1	Functional Simulations	92
	Safety and Reliability	92
	Propellant-efficiency	92
10-1-2	TEAMS Simulations	93
	Safety and Reliability	93

Propellant-efficiency	93
10-2 Recommendations	94
10-2-1 Guidance and Control Design for both Simulators	94
10-2-2 Approach Procedures for ENVISAT	95
10-2-3 TEAMS	95

List of Abbreviations

ADR	Active Debris Removal
AFRL	Air Force Research Laboratory
ATV	Automated Transfer Vehicle
CAM	Collision Avoidance Manoeuvres
CDF	Concurrent Design Facility
CoM	Centre of Mass
DARPA	Defense Advanced Research Projects Agency
DART	Demonstration of Autonomous Rendezvous Technology
DLR	Deutsches Zentrum für Luft- und Raumfahrt
DoF	Degree(s) of Freedom
ECI	Earth-Centred Inertial
ESA	European Space Agency
FDIR	Fault Identification, Detection, and Recovery
GEO	Geostationary Earth Orbit
GNC	Guidance Navigation and Control
GPS	Global Positioning System
HCW	Hill-Clohessy-Wiltshire
HIL	Hardware-In-the-Loop
HPS	High Performance Simulator

IADC	Inter-Agency Space Debris Co-ordination Committee
KOS	Keep-Out-Sphere
ISS	International Space Station
LEO	Low Earth Orbit
LOS	Line-Of-Sight
LQR	Linear Quadratic Regulator
MIT	Massachusetts Institute of Technology
MPC	Model Predictive Control
MRD	Mission Requirements Document
NASA	National Aeronautics and Space Administration
PMD	Post Mission Disposal
RAAN	Right Ascension of the Ascending Node
SOCP	Second Order Cone Program
SPHERES	Synchronized Position Hold, Engage, Reorient Experimental Satellites
SSO	Sun Synchronous Orbit
SRP	Solar Radiation Pressure
TEAMS	Test Environment for Applications of Multiple Spacecraft
TLE	Two-Line Element

List of Symbols

Symbol	Description	Unit
a	semi-major axis	[m]
a	acceleration	[m/s ²]
\mathbf{a}	acceleration vector	[m/s ²]
A	frontal area	[m ²]
\mathbf{A}	continuous system matrix	[varies]
\mathbf{A}_d	discrete system matrix	[varies]
\mathbf{B}	continuous input matrix	[varies]
\mathbf{B}_d	discrete input matrix	[varies]
c	speed of light	[m/s]
C_d	drag coefficient	[-]
e	orbit eccentricity	[-]
$\hat{\mathbf{e}}$	unit vector	[-]
\mathbf{F}	force vector	[N]
F_s	solar flux	[W/m ²]
G	gravitational constant	[m ³ /(kgs ²)]
h	specific angular momentum	[m ² /s]
h	integration step size	[s]
i	inclination	[rad]
J	performance index	[varies]
J	mass moment of inertia	[kgm ²]
J_n	zonal harmonic coefficient	[-]
\mathbf{K}	gain matrix	[varies]
m	mass	[kg]
M	mass of the central body	[kg]
N	number of discretization steps	[-]
n	mean motion	[rad/s]
\mathbf{n}	mean motion vector	[rad/s]
q	reflectance factor	[-]
\mathbf{Q}	weight matrix	[varies]

R	weight matrix	[varies]
R	radius	[m]
r	position radius	[m]
\mathbf{r}	position vector	[m]
$\dot{\mathbf{r}}$	velocity vector	[m/s]
$\ddot{\mathbf{r}}$	acceleration vector	[m/s ²]
\mathbf{s}	unit vector towards the sun	[-]
\mathbf{s}	slack variable vector	[varies]
t	time	[s]
T	orbital period	[s]
T	torque	[Nm]
T	thrust vector	[N]
T	Stacked solution vector	[N]
T	thrust magnitude	[N]
\mathbf{u}	control vector	[N]
v	velocity	[m/s]
\mathbf{v}	velocity vector	[m/s]
X	state vector	[varies]
\mathbf{x}^*	optimal solution vector	[varies]
x	x-coordinate	[m]
y	y-coordinate	[m]
z	z-coordinate	[m]
Z	discrete state vector	[varies]

α	rotational velocity vector	[rad/s]
β	spin axis offset angle	[rad]
γ	weight factor	[-]
δ	precession rate spin axis	[rad/s]
μ	gravitational parameter	[m ³ /s ²]
ρ	atmospheric density	[kg/m ³]
θ	true anomaly	[rad]
θ	safety angle	[rad]
$\ddot{\theta}$	rotational acceleration	[rad/s ²]
ω	argument of periapsis	[rad]
ω	rotational velocity	[rad/s]
Ω	RAAN	[rad]

Subscript	Description
<i>0</i>	initial
<i>approach</i>	approach phase
<i>c</i>	chaser
<i>c</i>	circular
<i>cb</i>	ECEF frame
<i>drag</i>	atmospheric drag
<i>E</i>	Earth
<i>f</i>	final
<i>h</i>	rotating Hill reference frame
<i>i</i>	ECI frame
<i>n</i>	normal direction
<i>r</i>	radial direction
<i>ref</i>	reference
<i>rel</i>	relative
<i>sa</i>	spin axis reference frame
<i>SRP</i>	solar radiation pressure
<i>t</i>	target
<i>t</i>	tangential direction

Chapter 1

Introduction

1-1 Motivation

Since the beginning of space flight the collision hazard has increased substantially due to the large growth of artificial objects in space. Even if no future launches would be executed there would be a net increase in the overall satellite population due to collisions that will inevitably occur (Rodriguez and Liou, 2008). As it is highly unlikely that there will be no more space launches in the near future, passive and active debris removal is the subject of many studies that aim to limit the increase of orbital debris. The policy adapted by most space agencies is to implement methods to limit the growth of space debris. These practices are intended to prevent runaway growth in what is known as the Kessler syndrome: a case in which collisions between orbiting objects create a cascade of debris particles and further collisions, effectively rendering an orbital belt unusable (Kessler et al., 2010). The difference between passive and active debris removal is that passive systems are often mounted onto a spacecraft in advance and are used after the mission is complete, whereas with active debris removal a dedicated spacecraft will de-orbit debris objects or put them in a graveyard orbit. A study by Liou et al. (2010) shows that the Low Earth Orbit (LEO) environment could be stabilized if at least five objects with the highest probability of collision and mass product are actively removed each year. Wormnes et al. (2013) even states that the number of objects that should be removed, to achieve this, is in the order of 10 each year. In the current economical, social, and political environment, it is essential that this happens in a reliable, safe, and cost-effective manner.

Numerous technical concepts have been or are being developed to support Active Debris Removal (ADR). Many technical, legal, and financial challenges have been identified over the past decades. Wormnes et al. (2013) established that it might be financially more attractive to de-orbit multiple targets during one mission, but such a mission is significantly more challenging from a technical point of view. Even for a mission that attempts to remove only a single target it is likely that the removal spacecraft has to rendezvous with, and control, the target debris object. Docking to well-known and cooperative objects in space has only been accomplished by few space-faring nations. It becomes even more challenging when the characteristics of the target object are not fully known and the target itself is no longer able

to cooperate. To perform such operations autonomously allows for a higher chance of mission success when failures occur or when communication time with ground stations is limited. Performing an active debris removal mission autonomously will thus increase the robustness to unexpected events and reduce operational costs by eliminating the need for ground-based orbit control.

Anomalies encountered during previous missions, such as DART (NASA, 2006) and Orbital Express (Friend, 2008), have shown that to achieve this, current satellite capabilities have to be enhanced. One of the key technologies that is required to successfully realize such systems is robust and optimal guidance and control. The guidance and control system should ensure that an active debris removal mission is executed in a safe manner, preferably using a minimum amount of propellant. The computational power of on-board computers has increased significantly over recent years. Together with the steps made in optimization theory this has led to new guidance algorithms for rendezvous that are rooted in numerical optimization. See, for example, Gerth (2014) and Betts (1998), and the references therein. Convex guidance, that is based on convex optimization theory (Boyd and Vandenberghe, 2009) is such a technique and has shown promising results in research into planetary landers and formation flying (Acikmese and Ploen, 2007), (Blackmore et al., 2010), (Tillerson et al., 2002). It is also becoming a topic of interest in satellite rendezvous research (Lu and Liu, 2013), (Gao et al., 2009). Convex optimization is capable of delivering globally-optimal solutions, while at the same time constraining the trajectory. The main advantages of guidance based on convex optimization are that 1) convex optimization theory proves that a well-posed convex problem is guaranteed to converge, 2) the obtained solution will be the global optimum, 3) there are a number of different, very efficient solvers for this kind of problem, and 4) constraints and penalties can be imposed.

A key ingredient of the guidance and control system development is the capability to test and evaluate the designed algorithms. A first step in the testing sequence is to evaluate the algorithms in a functional simulation environment. These simulations can then be complemented by Hardware-In-the-Loop (HIL) simulations to evaluate the interaction of the guidance and control system with hardware elements, such as sensors and actuators and the corresponding real-time delays. There are multiple methods that exist for HIL testing of guidance and control algorithms. To reproduce the six Degree(s) of Freedom (DoF)s kinematics and vehicle dynamics of a spacecraft in a micro-gravity environment to a high degree, neutral buoyancy facilities can be used (Romano et al., 2007). These provide an adequate representation of the frictionless environment in orbit (Carignan and Akin, 2000). The kinematics of the six DoFs of the relative motion of two spacecraft, can be reproduced by using robotic simulators or cranes (Benninghoff et al., 2012). The three DoFs attitude kinematics and torque-free motion can be reproduced by suspending the spacecraft simulator on a hemispherical air bearing, where the center of mass coincides with the center of rotation of the bearing (Schwartz et al., 2003). Another option is a flat-floor test facility that enables the reproduction of the kinematics and vehicle dynamics for three DoFs (one rotational and two translational).

1-2 Research Question and Scope

The thesis research will aim to answer the following research question:

How can a satellite perform the final phase of a rendezvous mission autonomously with an uncooperative and tumbling target satellite in a low-Earth orbit in a safe, reliable, and propellant-efficient manner?

The research limits itself to the development of the guidance and orbit control of a chaser satellite to execute these operations. The design of a navigation system is omitted from this thesis work, though it will certainly be an important step in the next design phase, together with an assessment of the lighting and communication conditions. The target satellite in this research is ENVISAT, which is an ESA owned object that is no longer operational (ESA, 2012). ENVISAT suffered a major anomaly on 8 April 2012, resulting in a loss of communication links. It is a large satellite in a near-polar, near-circular orbit at an altitude of approximately 770 km, which makes it a high-profile target for ADR. A challenging aspect of a rendezvous and docking mission with ENVISAT is the tumbling motion it has acquired since the loss of communication. An approach strategy will be designed in this research that aims to handle this characteristic under the constraints posed by the research question. The tumbling motion of the target puts the most stringent requirements on the guidance and control system during the final phase of the rendezvous and docking operations and this stage will therefore be the focus of this research.

1-3 Research Methodology

To answer the proposed research question, within the defined scope, the following research methodology is adopted. A detailed reference scenario is defined, for which different guidance and control strategies are developed. The developed guidance and control algorithms should enable operations with sufficient accuracy to perform a successful rendezvous operation up to the point where the docking mechanism can be deployed. To test the guidance and control system a functional simulator is developed. The functional simulations results are validated with HIL simulations. The Test Environment for Applications of Multiple Spacecraft (TEAMS) facility of the Deutsches Zentrum für Luft- und Raumfahrt (DLR), a flat-floor test facility, is used in this research. It enables the reproduction of the kinematics and vehicle dynamics for three DoFs (one rotational and two translational). TEAMS emulates the force and momentum-free dynamics of satellites in orbit, i.e., it reproduces the weightlessness and frictionless environment (Schlotterer and Theil, 2010). The experiments are performed using two free-floating, air-cushion vehicles that move over a highly smooth surface. Limitations are, that the vehicle dynamics are reduced from three-dimensional orbit dynamics in the functional simulator to a basic double integrator on TEAMS and that to represent the ENVISAT reference scenario on TEAMS it has to be adapted to a 2D scenario. A large advantage, however, that is obtained by tests performed with TEAMS, is that it captures the interaction of the Guidance Navigation and Control (GNC) system with actual sensors and actuators and the real-time delays of such an environment.

A test campaign is designed, for both the functional simulations and the HIL simulations, to assess the performance of the different guidance and control strategies. Based on the

development challenges of the simulators and the results obtained in these test campaigns, conclusions and recommendation can then be formulated.

1-4 Thesis Roadmap

The remainder of this report has adapted the following structure: Chapter 2 will discuss other research and missions that provide a contextual framework for the reference scenario, which will also be presented in this chapter. The mathematical models used to describe the absolute and relative motion of the target and chaser satellite will be treated in Chapter 3. The functions and tasks of a guidance and control system will be covered in Chapter 4. This chapter will provide a brief introduction to GNC as a whole before delving deeper into the workings of guidance and control algorithms and covering the implemented guidance and control strategies. The theory behind and the construction of the convex guidance algorithms are presented in Chapter 5. The development, implementation, and verification of the functional simulator is discussed in Chapter 6. An introduction to the TEAMS facility is given in Chapter 7. This chapter will also discuss the necessary adaptations to assess the developed guidance and control algorithms on TEAMS. The developed software for TEAMS will be examined in Chapter 8. The test campaigns and corresponding results for both the functional simulations and HIL simulations are covered in Chapter 9. The conclusions that are drawn based on this thesis research are presented in Chapter 10, which will also give a number of recommendations for future work.

Mission Heritage and Reference Scenario

Many technical concepts have been or are being developed to support ADR using autonomous rendezvous and docking. The mission heritage of these two topics will be treated in Section 2-1. This section provides a contextual framework for the reference scenario, which will be presented in Section 2-2.

2-1 Mission Heritage

ADR has been the focus of many studies and missions over the past years. This section will first discuss the target selection in Section 2-1-1. The aim of this thesis research is to design a guidance and control system that enables autonomous rendezvous and docking operations. Past missions have shown that this kind of operations poses many challenges. A brief overview, from the early rendezvous and docking missions to the more recent missions that performed autonomous rendezvous and docking operations, will therefore be provided in Section 2-1-2.

2-1-1 Active Debris Removal

The clear need for ADR was already established in Chapter 1. This chapter stated that it is necessary to remove 5-10 objects with the highest probability of collision and mass product to stabilize the LEO environment (Liou et al., 2010), (Wormnes et al., 2013). Note that the Post Mission Disposal (PMD) regulation for LEO, which states that a spacecraft has to de-orbit in less than 25 years (IADC, 2013), is already taken into account in this case and thus only sufficient ADR measures might mediate the situation. In ESA (2012), it was determined that in terms of number of collisions prevented per object removed the best strategy is to select targets based on the following principles:

- The selected objects should have a high mass (they have the largest environmental impact in case of collisions).
- The objects should have high collision probabilities (they should be in densely populated areas).
- The objects should be at high altitudes (where the orbital lifetime of resulting fragments is long).

In LEO the majority of these objects is at altitudes between 800 and 1000 km, in near-circular, highly inclined orbits. This orbital belt is therefore often the focus of ADR studies. Several technical concepts have been or are being developed at this time to support ADR. Many challenges have been identified over the past decades, technically, legally and financially. From a financial point of view it might be more attractive to de-orbit multiple targets during one mission, but this will increase the technical challenges of such a mission (Wormnes et al., 2013). Even for just a single target it is likely that the removal spacecraft has to rendezvous with, and control, the target debris object. Another challenging aspect could be uncertain mass and attitude properties of the target satellite. Depending on the strategy and the corresponding removal mechanism the removal spacecraft might have to perform a docking operation. Successfully docking with an uncooperative target has never been achieved without involving human astronauts. Another challenge lies in the fact that there may not be a convenient place to grab the object, dock with it, or attach a de-orbitation device to it (Jakhu, 2012). Wormnes et al. (2013) identified that approaching a target, flying around it and connecting to it, puts stringent requirements on the rotational and translational capabilities of the chaser spacecraft. This will most likely require more complex propulsion and GNC systems than the systems usually installed in LEO.

2-1-2 Rendezvous and Docking

Several of these ADR methods under investigation require docking with the target debris by making use of, for example, a robot arm or clamping device. Docking to well-known and cooperative objects in space has only been accomplished by few spacefaring nations. To increase the robustness to unexpected events it is therefore desirable to have the capability to perform the rendezvous and docking operations autonomously. This section will explore past developments related to (autonomous) rendezvous and docking.

Early Rendezvous and Docking

The first rendezvous and docking missions performed by the United States relied heavily on human control in the final approach phase. The very first rendezvous and docking was achieved as part of the Gemini program, where the Gemini 8 capsule docked with the modified Agena booster second stage target. The astronaut crew manually manoeuvred their spacecraft during the last tens of meters. This reliance on manual control required a unique scenario for each mission, which is rather expensive and labour intensive (Polites, 1999). The experience gained by the United States during the Gemini program was later used in the Apollo program, which performed rendezvous and docking operations in a lunar orbit. The lunar excursion



Figure 2-1: An early Soviet stamp illustrating the first automatic docking performed by Kosmos-186 and Kosmos-188 (Matsievsky, 2006).

module docked with the command/service module after ascending from the lunar surface. The crew on-board the lunar excursion module performed a multitude of manual tasks throughout these operations. More recent, the developed rendezvous and docking approach was applied to rendezvous and docking tasks of the Space Shuttle Orbiter. An important difference between the Space Shuttle and the Gemini and Apollo programs is that next to regular docking operations the Space Shuttle, made use of a robotic arm to retrieve and berth satellites (Goodman, 2006). Although the complexity of the Space Shuttle missions was much larger than for Gemini and Apollo, a substantial part of the operations were still performed manually by the crew (Polites, 1999).

The Russian space program, on the other hand, incorporated automated and standardized rendezvous and docking manoeuvres from early on. The first automated docking manoeuvre was performed in 1967, when the Soviet experimental unmanned spacecraft Kosmos-186 docked with Kosmos-188 (Legostaev and Raushenbach, 1969). It is important to consider that they had to use very simple control algorithms due to the limited available computational abilities. The Soyuz program furthermore accomplished the first docking of two piloted vehicles and transferred crew from one spacecraft to another in orbit for the first time. The spacecraft in the Soyuz program were designed primarily for automated orbital rendezvous with piloted capabilities generally reserved for contingency operations (Woffinden and Geller, 2007). This led to a longer development time but improved the efficiency and reliability of the vehicles.

Engineering Test Satellite VII

Throughout the last two decades a considerable number of missions have flown that demonstrated different levels of autonomous rendezvous and docking capabilities. Just before the turn of the millennium the Japanese launched the Engineering Test Satellite VII, which consisted of both a chaser and a target satellite, illustrated in Figure 2-2. The chaser satellite was equipped with robot arms and had the objective to perform two rendezvous and docking experiments. The first experiment involved the separation of the chaser, which then moved



Figure 2-2: The chaser and target modules of the Engineering Test Satellite VII (JAXA, 2003).

to a hold point at a distance of two meters where it remained for 15 minutes with close to zero relative velocity. After receiving an approach command it successfully docked a couple of minutes later to the target satellite. During the second approach phase experiment, the chaser encountered a contingency after which it autonomously activated an abort command and flew to a retreat point. After failure of two more approach attempts ground investigation established that there was a thruster valve malfunctioning, but after system reconfiguration the second rendezvous and docking experiment was also successful (Ohkami and Kawano, 2003). Even though the cause had to be established by ground investigation, the sophisticated rendezvous software autonomously detected a failure, triggered an abort command and safely performed Collision Avoidance Manoeuvres (CAM).

XSS-10 and XSS-11

In 2003 the Experimental Satellite System series conducted by the US Air Force demonstrated capabilities necessary to perform autonomous rendezvous and docking in orbit using micro-satellites. The first satellite, the XSS-10, had the objective to demonstrate autonomous navigation, proximity operations, and inspection of a resident space object. A problem occurred at the fourth inspection point where it planned to, and apparently did, perform an approach manoeuvre to decrease its distance to the resident space object to 50 meters. A telemetry drop-out occurred and the ground station lost the signal, therefore it cannot be verified how close the satellite was to the object (Davis and Melanson, 2004). It achieved all its other objectives without running into major difficulties. Two years later the successor of the XSS-10 was launched, the XSS-11. Its objective was to further demonstrate capabilities for performing autonomous proximity operations with the upper stage of its launch vehicle (AFRL, 2005). It successfully executed rendezvous and 75 natural motion circumnavigations. The XSS-11 also performed rendezvous operations with a number of US-owned decommissioned satellites, however, these results are not readily available to the public (Munoz, 2011).

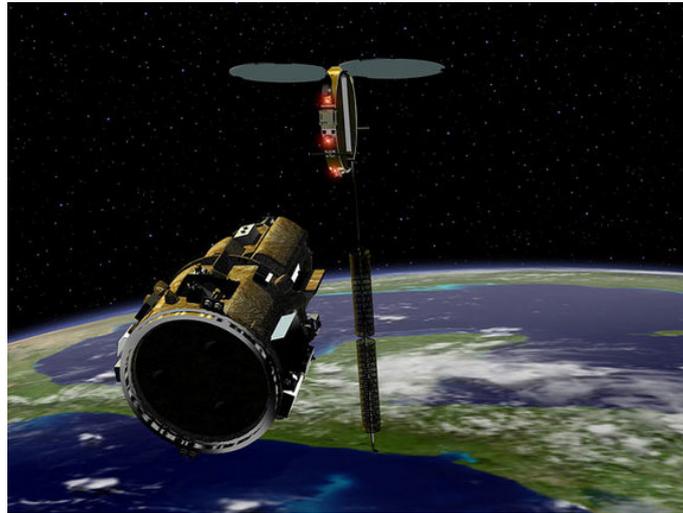


Figure 2-3: An artist impression of the DART spacecraft as it approaches the target satellite in orbit (Malik, T., 2006).

DART

Around the same time as the XSS-11 mission the National Aeronautics and Space Administration (NASA) launched the Demonstration of Autonomous Rendezvous Technology (DART) spacecraft, which was designed to demonstrate, in orbit, the hardware and software necessary for autonomous rendezvous to a separation distance of 5 meters. As stated in NASA (2006), the navigation system already malfunctioned slightly during the first phases of the rendezvous but operations could be carried out as planned. During the last part of the approach the navigation system bias error resulted in wrong thruster commands and excessive use of propellant. DART also failed in autonomously avoiding a collision as it eventually collided with the target. It was suggested that these biased measurements would not have doomed the flight had the preprogrammed gain matrix in the navigation filter been properly tuned. The mishap investigation board further concluded that there was an insufficient system-level understanding of the potential effects of complete or partial loss of functionality of relevant subsystems. The lack of thorough validation of math models and testing also contributed substantially to the mission failure.

Orbital Express

Orbital Express was a mission funded by Defense Advanced Research Projects Agency (DARPA), that was launched in 2007. Its goal was to validate the technical feasibility of robotic, autonomous on-orbit refuelling and reconfiguration of satellites (Friend, 2008). The mission consisted of two spacecraft, a service satellite called ASTRO and a client satellite named NextSat. It completed several on orbit transfers of propellant and batteries successfully. During another experiment ASTRO autonomously undocked from NextSat and moved to a distance of 10 meters. It then autonomously returned down the corridor to capture the NextSat again within two minutes of predicted time of capture. Throughout the third scenario ASTRO experienced a failure of the primary sensor computer. It then attempted to

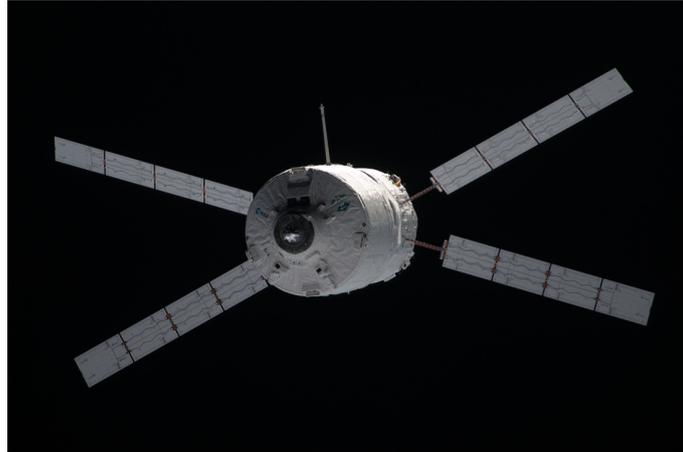


Figure 2-4: In-orbit photo of the ATV (Blau, P., 2015).

reboot it twice, without success. This resulted in an autonomous abort to 120 meters and ground control had to step in because the relative navigation state at that point had decayed significantly. The mission included a total of nine scenarios during which six world or US firsts in space were achieved (Friend, 2008).

ATV

From April 2008 the European Automated Transfer Vehicle (ATV) program has performed regular deliveries to the International Space Station (ISS) up to July 2014 when it was launched to perform its final mission. The ATV was the first European vehicle to successfully perform automated rendezvous and docking, and the largest ever to do so in the complex context of human spaceflight (Strandmoe et al., 2008). Furthermore, it was the first to use relative GPS technology for autonomous far rendezvous and the first to perform a fully automated final approach up to docking entirely based on optical sensors (Baize and Novelli, 2010). The flight is autonomously controlled by GNC algorithms of the flight control system. The vehicle is not fully autonomous, because the ATV Control Centre on the ground uploads mission plans at certain stages. The ATV was required to dock to the ISS with an accuracy of less than 10 cm, and during the first flight approximately 1.5 cm accuracy was achieved (Baize and Novelli, 2010). Collision avoidance is performed during the rendezvous and the departure phases through a segregated but fault tolerant safety chain with dedicated computers and sensors. The flight control system has proved to function as required in orbit, and no anomalies occurred onboard (Strandmoe et al., 2008).

2-2 Reference Mission Scenario

The reference mission for this research is based on the e.deorbit mission (ESA, 2012). The e.deorbit Concurrent Design Facility (CDF) study was the first system level study on ADR by the European Space Agency (ESA). After completion of that study, contracts were awarded to industry and three consortia submitted proposals. At the moment of writing the mission is in phase B.

Table 2-1: Orbital parameters of Envisat, based on TLE of September 28, 2014 (N2YO, 2014)

Orbital elements	Current value
semi-major axis [km]	7146
Eccentricity [-]	0,0001227
Inclination [deg]	98,3724
RAAN [deg]	334,5377
Argument of perigee [deg]	103,8499
True anomaly [deg]	333,8374

The e.deorbit mission objective is to safely de-orbit the target object ENVISAT, an ESA owned object that is no longer operational (ESA, 2012). ENVISAT suffered a major anomaly on 8 April 2012 resulting in a loss of communication links. It is in a near-polar, near-circular orbit at an altitude of approximately 770 km. The mission consists of a chaser satellite that is launched by a small or medium launcher, then performs a safe and propellant-efficient rendezvous and docking with the target satellite, and removes it from the LEO protected zone. Several capture techniques are investigated in the e.deorbit assessment study. This thesis is not concerned with the specific capturing mechanism, but it will be assumed that a rigid connection has to be made with the target satellite. The capturing mechanism will grasp and/or clamp ENVISAT at the Centre of Mass (CoM) on the upper side where no instruments are situated. This is defined by ESA as the most suitable location to make a connection.

2-2-1 Target and Chaser Satellite

This section will present aspects of ENVISAT, which are important for the development of the guidance and control system. ENVISAT has a mass of 7828 kg and will be modelled as a point mass with reference surfaces. The initial orbital parameters of Envisat for the scenario are given in Table 2-1.

Figure 2-5 shows the top view and side view of ENVISAT and the relevant dimensions. From this figure it immediately becomes clear that one of the difficulties of rendezvous and docking with ENVISAT is the large solar array. This becomes even more challenging when the tumbling motion of ENVISAT is considered. Radar measurements of Envisat performed at the end of 2013 showed that the main motion of Envisat is a rotation of approximately 3.5 deg/s around its orbital angular momentum vector Bastida Virgili (2014). On top of this rotation, ENVISAT is slightly tumbling around its other body axes. There is uncertainty on the future evolution of ENVISAT's motion and currently ESA is analysing the satellite to enable better predictions (Deloo, 2015). Because of this uncertainty several assumptions are made for this research concerning the attitude of the target:

- ENVISAT is assumed to be rotating with its spin axis under an angle of 30 deg offset from its orbital angular momentum vector.
- The spin axis itself is also precessing with a rotational rate of 0.2 deg/s around the orbital angular momentum vector.

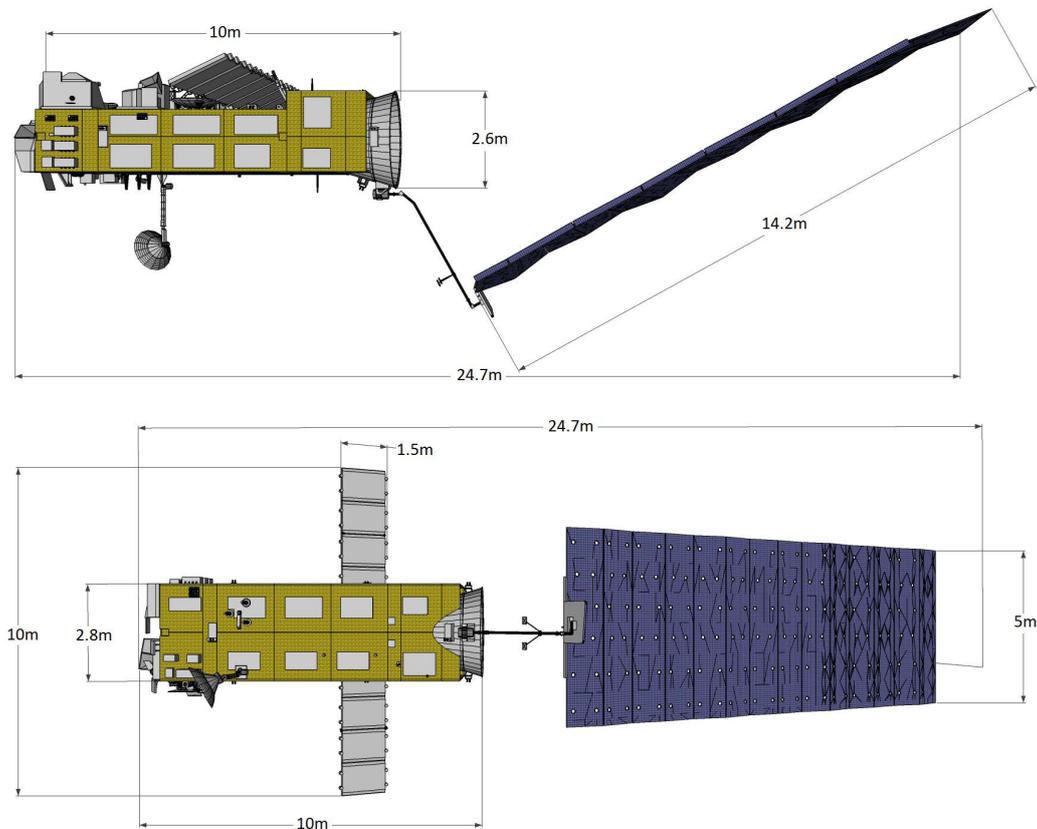


Figure 2-5: ENVISAT model and dimensions. Upper: side view. Lower: Top view (ESA, 2012).

The chaser satellite is also based on the chaser presented in the e.deorbit study (ESA, 2012). A model of this satellite is shown in Figure 2-6. For this research it is important to state that the chaser has 12 thrusters, which are fired in pairs, and are capable of delivering a thrust of 44 N along each axis of the chaser body frame.

2-2-2 Approach Overview and Requirements

The reference scenario is concerned with the final part of the rendezvous, The chaser starts at 50 m from the target and is initially aligned with the spin axis. This distance is chosen, because the Keep-Out-Sphere (KOS) defined by ESA has a radius of 50 m (Deloo, 2015). During the approach the alignment with the spin axis shall be maintained. The scenario ends at 3 m from the target's CoM. It is assumed that the clamping/grasping mechanism will be deployed at this point. The reference scenario ends at 3 m from the target, where a stable motion with respect to the target should be kept. It is assumed that the clamping/grasping will be deployed at this point.

To ensure that the chaser satellite will not collide during the approach with any part of ENVISAT a safety cone constraint is formulated. This constraint defines a cone in which the chaser can move throughout the approach. The cone is fixed with respect to the spin axis, i.e., it moves along with the movement of the spin axis. To clarify this a side view orientation

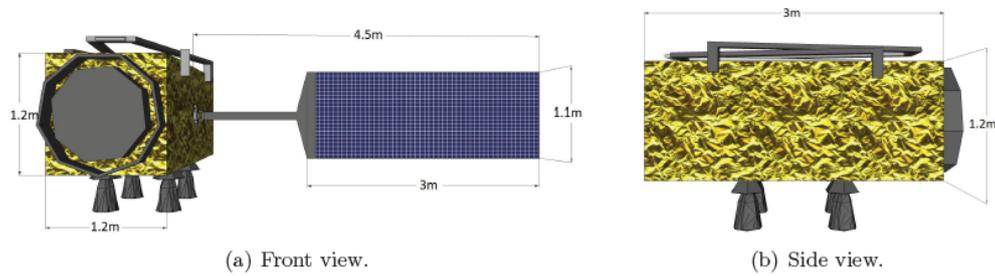


Figure 2-6: Chaser model and dimensions. Upper: side view. Lower: Top view (ESA, 2012).

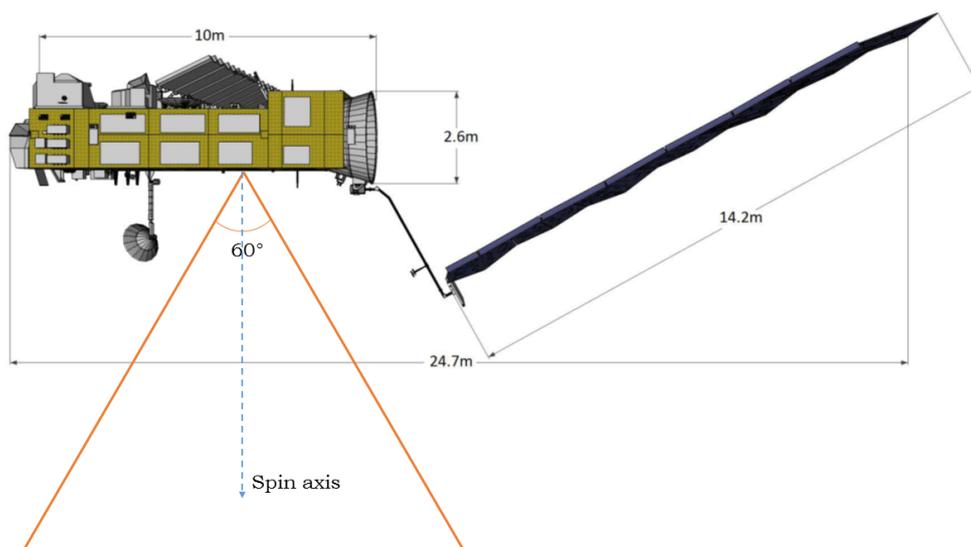


Figure 2-7: 2D view of the safe approach area of the chaser, which is indicated by the area inside the triangle. It is a cone with a half-angle of 30 degrees, that moves along with the spin axis movement. (Adapted from ESA (2012).)

of this cone is shown in Figure 2-7. The area inside the triangle portrays the area in which the chaser is allowed to move. The exact implementation of this constraint in the guidance algorithm is discussed in Section 5-2-2.

Based on Fehse (2003), it is assumed that the control error, at the stage where the chaser is at 3 m relative distance from the target and has obtained a stable circular in-plane motion with respect to the target, is required to be in the order of a few centimetres or less for the position and no larger than 1 cm/s for the velocity.

2-3 HIL-Simulation Environment

On-the-ground experimentation is a low-risk, relatively low-cost, and potentially high-return method for validating guidance and control systems (Romano et al., 2007). There are a number of different types of HIL-simulation environments like flat floor testbeds, blimp units

(lighter-than-air vehicles), underwater vehicles, and robotic facilities (Nolet, 2007). A brief introduction to the different HIL-simulation environments was already provided in Chapter 1. Each of these testbeds have their own inherent advantages and drawbacks.

The reference scenario described in Section 2-2 shows that the main requirement to test this scenario is that the testbed should be able to represent a chaser vehicle that can approach, and preferably dock with, a rotating target. Next to this, it should allow for easy implementation of the designed guidance and control system in the HIL-simulation environment. An attractive low-cost solution is a flat-floor testbed. Such a testbed usually consist of a table with a highly smoothed surface with areas ranging from approximately 15 m² up to about 200 m². A number of physical satellite model vehicles float on air bearings in a near-frictionless environment to practice different aspects of spacecraft missions. Such a test facility provides at least three DoF, two translational and one rotational. Extra degrees of freedom can be obtained by enhancing the satellite vehicles with for example attitude platforms and vertical actuators. Flat-floor testing facilities are often an intermediate step in the testing process, it allows for long duration testing and validating of GNC algorithms before tests on more expensive facilities are performed (Romano et al., 2007). The use of a flat floor test bed provide a good platform to test several critical aspects of actual autonomous rendezvous and docking manoeuvres.

The TEAMS flat-floor test facility of DLR is therefore used in this research. The main adaptation to the reference scenario is the reduction of the 3D rotational motion of the target to a 2D rotation. It is also equipped with docking adapters which allow for the testing of the docking phase. A detailed description of the TEAMS facility is provided in Chapter 7.

2-3-1 Examples of Flat-floor Testbeds

Several examples of flat-floor test facilities will be described in this section to provide a contextual framework for the TEAMS facility of DLR that will be used in this thesis research. This is by no means a full overview of the existing flat-floor test facilities, but this section will give a good indication of the different specifications and capabilities of such facilities.

Flight Robotics Laboratory of NASA

The Flight Robotics Laboratory at the Marshall Space Flight Center has one of the largest flat floor facilities in the United States with an impressive area of 360 m² (Wiegmann et al., 2012). The facility compromises two air-bearing spacecraft simulators capable of carrying 180 kg payload each, and a target simulator on a crane capable of carrying a 200 kg payload, shown in 2-8. The Spacraft Simulators provide 3 DoF and the target vehicle is mounted on an overhead gantry providing 6 DoF. The spacecraft simulators are closed-loop computer controlled for sensor/docking testing. The facility has been used for testing of a.o. systems of the DART mission and the Orbital Express program (Roe et al., 2004).

Space Systems Laboratory of MIT

The Massachusetts Institute of Technology (MIT) Space Systems Laboratory has a flat floor test facility that is mainly used for testing the Synchronized Position Hold, Engage, Reorient

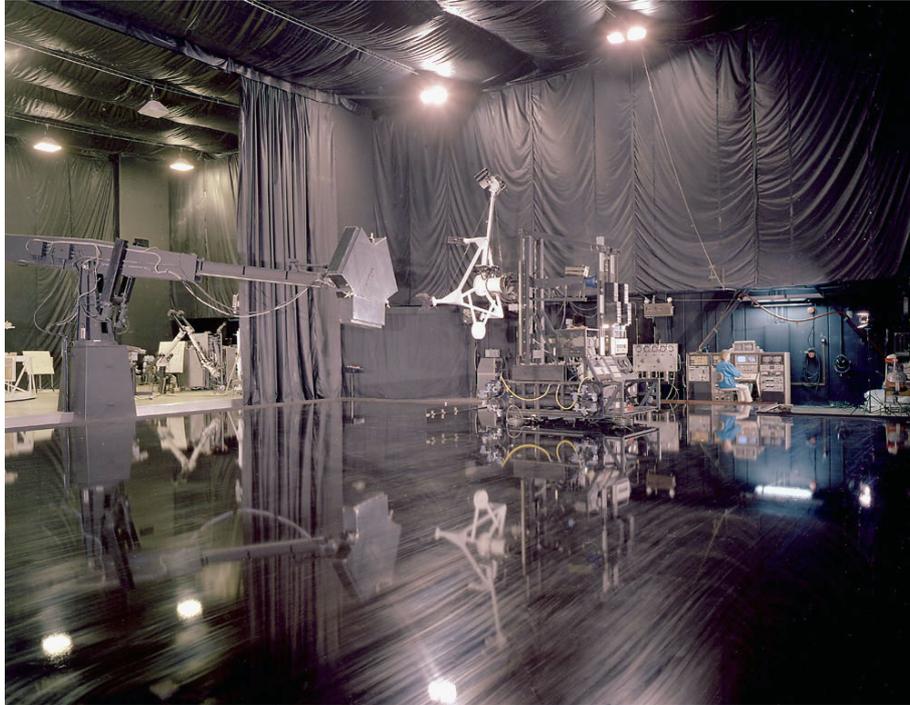


Figure 2-8: The spacecraft (right) and target (left) simulators at the Flight Robotics Laboratory (NASA, 1985).

Experimental Satellites (SPHERES) system (Miller et al., 2000). It has a smoothed glass surface with an area of 25 m² and 3 DoF. Next to this, the lab frame rotates once over every day with respect to the inertial frame. The vehicles are kept afloat by expelling compressed gas (CO₂) downward onto the surface of the testbed. The facility has been used to successfully demonstrate maneuvering and docking of the SPHERES system using a flexible beam in a 2D-environment.

Spacecraft Robotics Laboratory of NRL

The Spacecraft Robotics Laboratory of the Naval Postgraduate School has a flat floor facility constituting an epoxy floor surface of 20 m², a spacecraft simulator, and a target simulator, shown in 2-9 (Romano et al., 2007). The chaser spacecraft simulator can independently control its translational movements through the use of eight cold-gas thrusters and it controls its attitude with a reaction wheel. The use of the reaction wheel results in significant propellant conservation and extends the time endurance of single experimental runs. The spacecraft simulators are capable of autonomously performing operations during the experimental tests. In particular, the chaser spacecraft simulator only uses three light emitting diodes mounted on the target vehicle simulator external reference for its navigation.

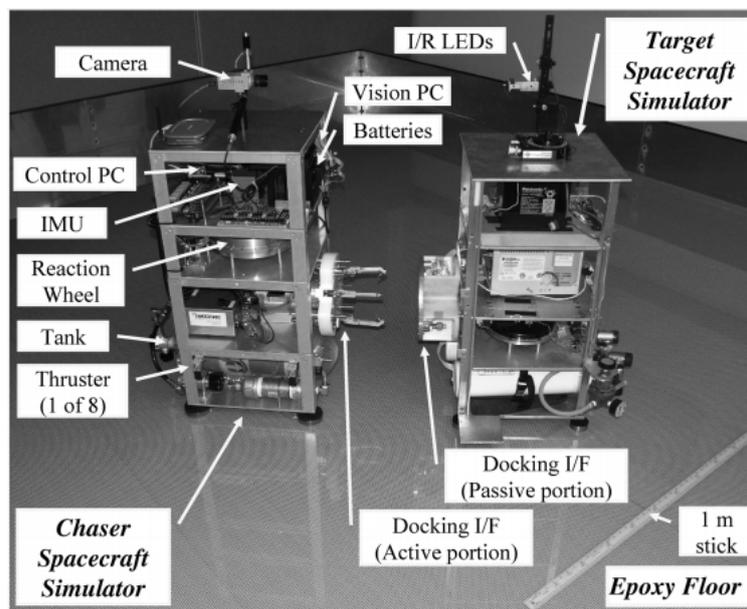


Figure 2-9: The autonomous docking flat floor test facility at the Spacecraft Robotics Laboratory of the Naval Postgraduate School (Romano et al., 2007)

Chapter 3

Orbital Mechanics

This chapter will cover the important elements of constructing models for the absolute and relative motion of the chaser and target satellite. Section 3-1 will cover the reference frames that are important for the construction of both models and the conversion between them. State representations that can express the absolute and relative motion are treated in Section 3-2. The equations of motion needed to model the absolute and relative motion are derived in Section 3-3. The perturbations that need to be included in the absolute motion models, because they have a significant impact on the relative dynamics of the chaser and target satellite, are discussed in Section 3-4.

3-1 Reference Frames

Absolute and relative motion models can be used to describe the absolute and relative trajectories of the chaser and target spacecraft. The reference frames used to describe these models are discussed in this section. A reference frame is a coordinate system in which different properties of an object, such as position, velocity, and orientation, can be determined. A reference frame is generally specified by its origin O and a set of three orthogonal vectors. All reference frames that are used in this report are right-handed. For the mission under consideration two types of reference frames are of interest, the Earth-Centred Inertial (ECI) frame and the rotating Hill frame. The former is used to describe the orientation of the orbit in an inertial frame and to describe the motion of a spacecraft in an orbit. The latter will be used to describe the motion of the chaser relative to the target.

Furthermore, note that only translational motion is considered for the reference scenario, and thus the three rotational degrees of freedom are ignored. This is a common assumption in the development of guidance laws, mainly caused by the fact that the rotational dynamics are much faster (Acikmese and Ploen, 2007). This allows for the decoupling of the translational and rotational dynamics. This assumption basically translates into the concept of "perfect attitude control", because it is assumed that the required attitude is realized instantaneously.

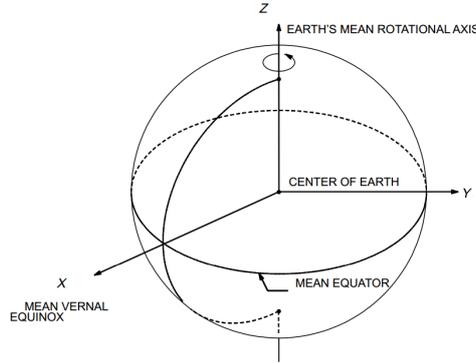


Figure 3-1: Definition of Earth-Centred Inertial reference frame (Sauceda, 2001).

3-1-1 ECI Frame

The ECI reference frame will be used to describe the absolute motion models of the target and chaser spacecraft. The ECI frame has its origin at the CoM of the Earth and the +X-axis is directed towards the J2000 vernal equinox. The +Z-axis is collinear with the rotation axis, neglecting nutation and precession, of the Earth and the +Y-axis completes the right-handed system. The OXY plane lies in the equatorial plane of the Earth. The ECI frame is depicted in Figure 3-1.

3-1-2 Rotating Hill Frame

To describe the relative motion the rotating Hill frame is used, which is depicted in Figure 3-2. The origin is fixed to the CoM of the target spacecraft. The +X-axis (R-bar) is aligned with the radius vector of the target, in the direction away from the Earth. The +Y-axis (V-bar) lies in the orbital plane, in the direction of the velocity and the +Z-axis (H-bar) completes the right-handed system and lies in the direction of the orbital angular momentum.

3-1-3 Transformation between ECI and Rotating Hill Frame

The absolute motion models of the target and chaser spacecraft are both described in the ECI frame. To enable the description of the state of the chaser in the rotating Hill frame a transformation between the two reference frames is required. This can be achieved using a transformation matrix. For this particular case the transformation matrix is formed by the unit vectors of the target in the rotating Hill frame. The unit vectors in radial, normal, and tangential direction are obtained from the target's inertial position and velocity by:

$$\begin{aligned}
 \hat{\mathbf{e}}_r &= \frac{\mathbf{r}_{t_i}}{|\mathbf{r}_{t_i}|} \\
 \hat{\mathbf{e}}_n &= \frac{\mathbf{r}_{t_i} \times \dot{\mathbf{r}}_{t_i}}{|\mathbf{r}_{t_i} \times \dot{\mathbf{r}}_{t_i}|} \\
 \hat{\mathbf{e}}_t &= \hat{\mathbf{e}}_n \times \hat{\mathbf{e}}_r
 \end{aligned} \tag{3-1}$$

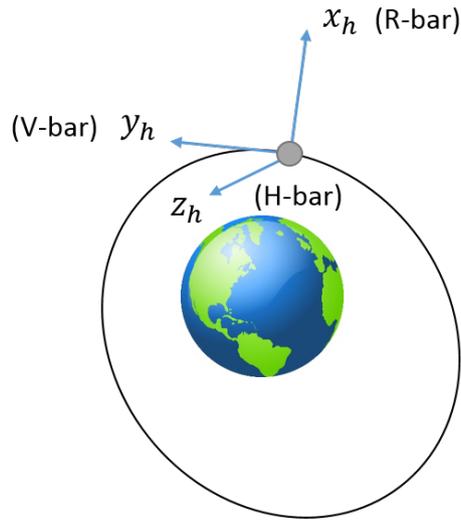


Figure 3-2: Definition of the rotating Hill frame.

The transformation matrix is then given by:

$$\mathbf{T}_h = [\hat{\mathbf{e}}_r \quad \hat{\mathbf{e}}_n \quad \hat{\mathbf{e}}_t]^T \quad (3-2)$$

The position and velocity of the chaser spacecraft in the rotating Hill frame are then obtained using Equations (3-3) and (3-4), where \mathbf{n} is the mean motion of the target in the inertial frame (De Bruijn et al., 2011).

$$\mathbf{r}_{c_h} = \mathbf{T}_h(\mathbf{r}_{c_i} - \mathbf{r}_{t_i}) \quad (3-3)$$

$$\dot{\mathbf{r}}_{c_h} = \mathbf{T}_h(\dot{\mathbf{r}}_{c_i} - \dot{\mathbf{r}}_{t_i}) - \mathbf{n} \times \mathbf{r}_{c_h} \quad (3-4)$$

The transformation matrix used to convert the state from the rotating Hill frame back to the ECI frame is simply the transpose of \mathbf{T}_h .

3-2 State Representations

A state vector can be used to mathematically describe the state of an object. For instance, let \mathbf{r} and $\dot{\mathbf{r}}$ represent the position and velocity of an object, respectively. In Cartesian coordinates this would be:

$$\mathbf{r} = [x \quad y \quad z]^T \quad \dot{\mathbf{r}} = [\dot{x} \quad \dot{y} \quad \dot{z}]^T \quad (3-5)$$

and a possible state vector could be chosen as:

$$\mathbf{X} = [\mathbf{r}^T \quad \dot{\mathbf{r}}^T]^T \quad (3-6)$$

There are many coordinate systems used in spacecraft and mission design. The two coordinate systems that are used in this thesis research will be discussed in more detail in the remainder of this section.

3-2-1 Cartesian Coordinates

Coordinate systems provide means to describe vectors with respect to some set of axes defined by a reference frame. A widely used system in orbital mechanics is the Cartesian coordinate system, where the position of any point in three-dimensional space is specified by three Cartesian coordinates. The three corresponding vectors are described by forming scaled linear-combinations along the unit vectors of the reference frame. This basically means that each unit vector of the coordinate axes is multiplied with a scalar and added together to obtain the vector that corresponds to the desired point in space.

3-2-2 Keplerian Elements

The Cartesian system can be used to describe orbital motion, but is not particularly intuitive. More convenient alternatives exist to do this, such as the Keplerian system. A body in an inertial reference frame can be defined in the Keplerian system by the following six parameters:

- semi-major axis, a
- eccentricity, e
- inclination, i
- argument of periapsis, ω
- Right Ascension of the Ascending Node (RAAN), Ω
- true anomaly, θ

The Kepler system is visualized in Figure 3-3. The semi-major axis and eccentricity describe the shape of the orbit, where the semi-major axis is defined as the distance between perigee and apogee and the eccentricity indicates the elongation of the orbit.

The four remaining angles describe the orientation of the orbit and the position of the body in the orbit. The inclination is defined as the angle between the orbital plane and the xy -plane of the ECI frame, measured at the ascending node. The ascending node is the point where the satellite crosses the xy -plane, in the direction of $+Z$. The inclination can thus be between 0 and 180 degrees, where an inclination of 0 degrees corresponds to a prograde equatorial orbit, and an inclination of 180 corresponds to a retrograde equatorial orbit. Evidently, an inclination of 90 degrees describes a polar orbit. The RAAN is defined as the angle between the x -axis of the ECI frame and the ascending node. This angle can thus vary from 0 to 360 degrees. The RAAN is undefined for orbits which have an inclination of 0 degrees, because in this case the ascending node can not be defined. The argument of periapsis is defined as the angle in the orbital plane between the ascending node and the periapsis and can be

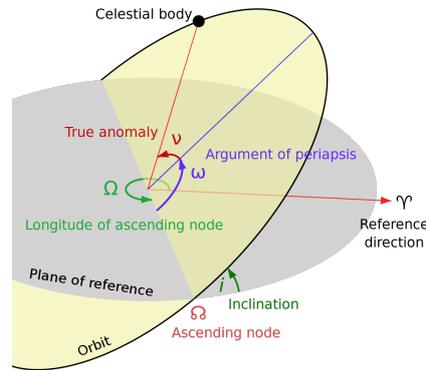


Figure 3-3: Orbital elements as defined in the Keplerian system (Vivarad, 2013).

between 0 and 360 degrees. Note, though, that for zero eccentricity (no definable periaapsis) and zero inclination (no definable ascending node) this angle is undefined. The true anomaly describes the position of the satellite in the orbital plane. The true anomaly is defined as the angle between the periaapsis and the position of the satellite, measured from the CoM of the attracting body and can thus be between 0 and 360 degrees.

3-3 Equations of Motion

The equations to describe the orbital motion around a central body can be derived from Kepler's and Newton's laws. These derivations are described in many reference books (Wertz et al., 2011; Curtis, 2013; Prussing and Conway, 1993). The main assumptions and relations, important to the understanding of rendezvous trajectories, will be reiterated here. First, a short overview of the absolute equations of motion will be provided. After that the equations to describe the relative motion between the two spacecraft are discussed.

3-3-1 Absolute Motion

As stated by Alfriend et al. (2007), the assumptions relating to the Keplerian two-body problem are:

- There are no external or internal forces except gravity.
- The gravitating bodies are spherical and have constant density.
- There are no tidal forces.
- The mass of the orbiting body is negligible compared to the primary body's mass.
- The gravitational force is Newtonian.

Newton's law of gravitation and his second law, which relates force and acceleration, are depicted in Equations (3-7) and (3-8), respectively. By combining these two equations and

using the assumption that $M \gg m$ Equation (3-9) can be formulated, which describes the orbital motion of a satellite with mass m at a distance r from the primary body, where μ is equal to the gravitational constant G times the mass of the central body M (Bate et al., 1971; Fehse, 2003).

$$\mathbf{F}_g = -\frac{GMm}{r^3}\mathbf{r} \quad (3-7)$$

$$\Sigma\mathbf{F} = m\ddot{\mathbf{r}} \quad (3-8)$$

$$\ddot{\mathbf{r}} = -\frac{\mu}{r^3}\mathbf{r} \quad (3-9)$$

A solution to Equation (3-9) is a polar equation of a conic section given by:

$$r = \frac{p}{1 + e \cos \theta} \quad (3-10)$$

where p is the semi-latus rectum. According to Wakker (2010), the type of conic section that is described by Equation (3-10) depends on the value of the eccentricity e . The following cases can be identified:

- $0 \leq e < 1$: ellipse
- $e = 1$: parabola
- $e > 1$: hyperbola

This study will limit itself to the application of elliptical and circular orbits, which is sufficient for applications to the test case. For an elliptical orbit the radius of apogee is given by:

$$r_a = a(1 + e) \quad (3-11)$$

and the radius of perigee by:

$$r_p = a(1 - e) \quad (3-12)$$

with $r_a + r_p = 2a$, Equation (3-10) can be rewritten into:

$$r = \frac{a(1 - e^2)}{1 + e \cos \theta} \quad (3-13)$$

Orbital perturbations are neglected in these formulations. Since the main interest will lie in the relative orbital perturbations, i.e., the effect of perturbations on the relative orbits, these will be discussed in more depth in section 3-4.

3-3-2 Relative Motion

There are many existing models that describe relative motion, which can be classified according to their nature. Models can, for example, be linear or non-linear. The relative state variables can be, for instance, local Cartesian, or curvilinear variables. Perturbations, such as drag and J_2 , can be taken into account and other assumptions on, for instance, eccentricity of the reference satellite can also be made.

A linear relative motion model that is often used to describe the chaser satellite's relative motion in rendezvous scenarios is the Hill-Clohessy-Wiltshire (HCW) model, developed by Clohessy and Wiltshire (Clohessy and Wiltshire, 1960). The equations of motion derived and solved in the HCW model are similar to those used by the mathematician George Hill in 1878 to describe the Moon's motion relative to the Earth (Hill, 1878). The HCW model has typically been used in the past for rendezvous missions, where the two spacecraft are in close proximity and the mission time is relatively short. The HCW model assumes that the chief's orbit has zero eccentricity (and the deputy has zero or small eccentricity), and no perturbations are taken into account, just two-body gravitational dynamics.

Since the orbit of ENVISAT is near-circular and the distance between the chaser and target satellite is sufficiently small in the last phases of the rendezvous operations the HCW equations fit the reference scenario well. Next to that, the guidance laws should be simple and robust to be applied online. The simplifying assumptions of the HCW model result in an error of about 10^{-5} m/s², as can be seen in Figure 3-4, which is deemed to be sufficiently accurate to be implemented in the guidance laws.

As given by Curtis (2013), the HCW equations are denoted:

$$\begin{aligned}\ddot{x} - 2n\dot{y} - 3n^2x &= 0 \\ \ddot{y} + 2n\dot{x} &= 0 \\ \ddot{z} + n^2z &= 0\end{aligned}\tag{3-14}$$

where n is the mean motion of the target satellite. The equations are defined in the rotating Hill frame. For the full derivation of these equation the author refers to Alfriend et al. (2007) or Curtis (2013).

From these equations it can be concluded that the motion in the z-direction is a pure harmonic oscillation that is uncoupled from the motion in the x and y directions. Straightforward, time-explicit, closed form analytical solutions exists for these equations, given in Equation (3-15) (Curtis, 2013). These solutions yield the trajectory of the chaser in the rotating Hill frame. The solutions are obtained using the relative position and velocity at epoch time t_0 as initial conditions, denoted by the subscript 0.

$$\begin{aligned}x &= 4x_0 + \frac{2}{n}\dot{y}_0 + \frac{\dot{x}_0}{n}\sin(nt) - \left(3x_0 + \frac{2}{n}\dot{y}_0\right)\cos(nt) \\ y &= y_0 - \frac{2}{n}\dot{x}_0 - 3(2nx_0 + \dot{y}_0)t + 2\left(3x_0 + \frac{2}{n}\dot{y}_0\right)\sin(nt) + \frac{2}{n}\dot{x}_0\cos(nt) \\ z &= \frac{1}{n}\dot{z}_0\sin(nt) + z_0\cos(nt)\end{aligned}\tag{3-15}$$

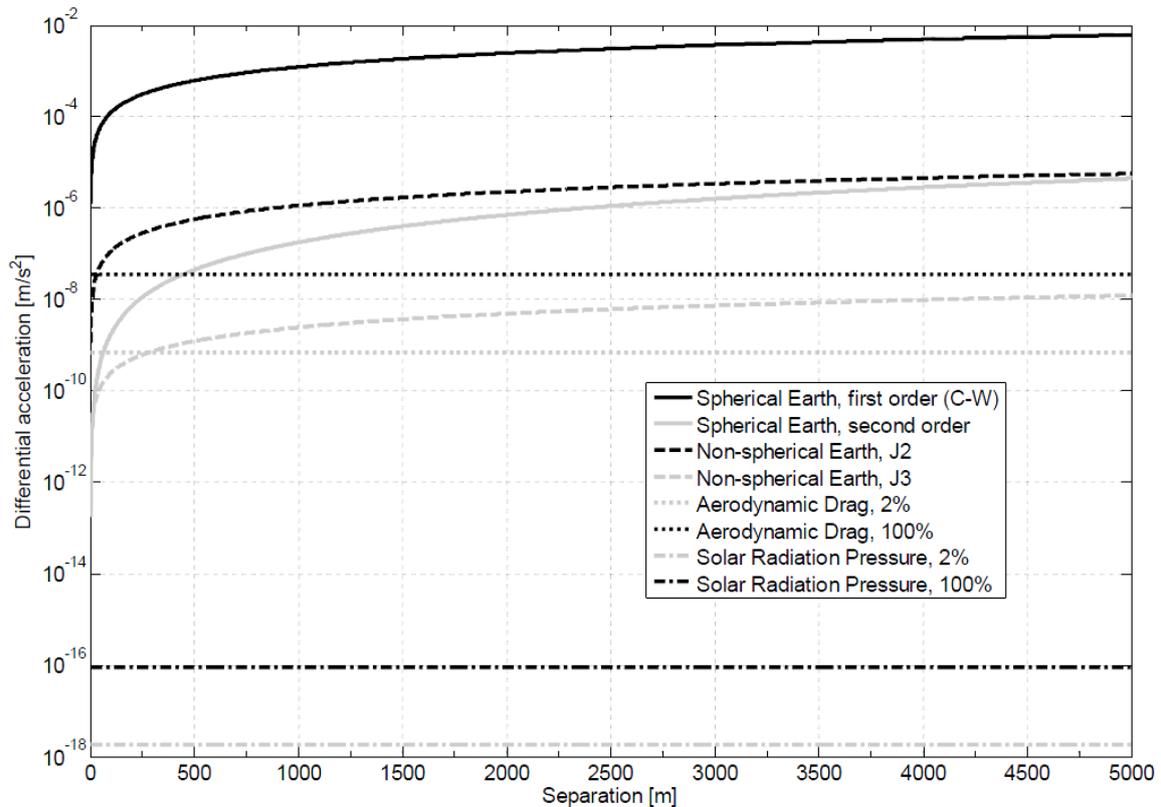


Figure 3-4: Magnitude of relative accelerations for close near-circular formations as a function of the spacecraft separation (D'amico, 2010).

3-4 Relative Perturbations

The absolute motion models should include the perturbations that have the largest impact on the relative dynamics of the satellites. Which orbital perturbations to include is based on an examination of perturbed satellite orbits below altitudes of 1500 km by (D'amico, 2010). The author analyzed the relative accelerations for close near-circular formations as a function of the spacecraft separation, for which the results are presented in Figure 3-4. It is clear from this figure that for a separation distance below 50 m the most significant relative environmental perturbations are J_2 and atmospheric drag. Deloo (2015) showed that for this reference scenario Solar Radiation Pressure (SRP) is also a significant relative perturbation and SRP is therefore also included. How these perturbations are modeled and the magnitude of effects is discussed in the remainder of this section.

3-4-1 Perturbations Models

As discussed, the main perturbation for a formation in LEO stems from the J_2 gravity term, which results from the flattening of the Earth. The absolute acceleration due to the J_2 effect is modelled according to:

$$\begin{aligned}
a_x &= -\frac{3}{2}\mu J_2 \frac{R^2}{r^5} x \left(1 - 5\frac{z^2}{r^2}\right) \\
a_y &= -\frac{3}{2}\mu J_2 \frac{R^2}{r^5} y \left(1 - 5\frac{z^2}{r^2}\right) \\
a_z &= -\frac{3}{2}\mu J_2 \frac{R^2}{r^5} z \left(3 - 5\frac{z^2}{r^2}\right)
\end{aligned} \tag{3-16}$$

The J_2 coefficient is assumed to be equal to 1082.63×10^{-6} (Braeunig, 2014).

The acceleration due to the atmospheric drag is dependent on the relative velocity, the ballistic coefficient. It is given by:

$$\mathbf{a}_{drag} = -\frac{1}{2}\rho v_{rel} \left(\frac{C_D A}{m}\right) \mathbf{v}_{rel} \tag{3-17}$$

where m is the mass of the spacecraft, ρ is the atmospheric density, A is the average frontal area of the spacecraft (the area normal to the relative velocity vector), and C_D is the dimensionless drag coefficient (Curtis, 2013). The relative velocity \mathbf{v}_{rel} is defined as in Equation (3-18),

$$\mathbf{v}_{rel} = \mathbf{v} - \boldsymbol{\omega}_E \times \mathbf{r} \tag{3-18}$$

where $\boldsymbol{\omega}_E$ is the angular velocity of the Earth, which is assumed to be equal to 7.2921150×10^{-5} m/s (Bizouard, C., 2014). The atmospheric density is based on the NRLMSISE-00 model, which is the most recent atmosphere model, and is based on mass spectrometer and radar measurements of the upper atmosphere, along with additional databases from drag measurements (COSPAR, 2012). A constant density of 1.94×10^{-13} is assumed which is based on an altitude of about 770 km and high solar activity. An average frontal area of 38.14 m^2 and 5.5 m^2 are assumed for ENVISAT and the chaser satellite, respectively. The mass of ENVISAT is assumed to be equal to 7828 kg and that of the chaser to 1444 kg. The drag coefficient is difficult to determine at this stage and is assumed to be equal to 2.2, for both satellites. These values are based on ESA (2012) and an analysis performed by Deloo (2015).

The acceleration due to the SRP is given by:

$$\mathbf{a}_{SRP} = -\frac{F_s(1+q)}{c} \frac{A}{m} \mathbf{s}_i \tag{3-19}$$

where F_s is the solar flux, q is the reflectance factor, c is the speed of light, A is the average frontal area of the spacecraft, and m is its mass. The vector \mathbf{s}_i is the unit vector towards the Sun in the ECI frame (Curtis, 2013). A constant solar flux of 1365 W/m^2 and a reflectance factor of 0.3 (for both satellites) are assumed. The speed of light is assumed to be equal to 2.99792458×10^8 m/s (Bizouard, C., 2014).

Table 3-1: Order of magnitude of the relative perturbations acting on the chaser satellite in the final rendezvous phase.

Perturbing acceleration	Order of magnitude (m/s²)
Non-spherical Earth (J_2)	10^{-7}
Solar Radiation Pressure	10^{-8}
Atmospheric drag	10^{-11}

3-4-2 Magnitude of Relative Perturbations

The perturbations are included in the absolute motion models of both satellites and the relative accelerations are obtained by simply subtracting the corresponding absolute accelerations from each other. The perturbations described in the previous section are implemented in the chaser and target absolute motion models. The resulting order of magnitude of the relative perturbations are summarized in Table 3-1. These values vary somewhat from the values presented by D'amico (2010), which can be attributed to differences in the altitude of the satellites, the atmospheric and SRP models used, and the assumed satellite parameters.

Chapter 4

GNC for Rendezvous

Many satellite subsystems interact during the process of rendezvous and docking. Sensor data, GNC information and physical docking functions have to be harmonized to accomplish the mission. Past missions, discussed in Section 2-1-2, have shown that the current technologies and development procedures need to be enhanced to improve the safety and mission success of autonomous rendezvous and docking mission operations.

To enable autonomous rendezvous and docking, i.e., without the intervention of flight controllers or crew, with an uncooperative and tumbling target in a safe manner poses demanding requirements on the GNC system. There is a need for reliable and efficient autonomous rendezvous optimal path planning and control algorithms, that drive the rendezvous and docking with the target under safety and practical constraints.

The GNC system lies at the heart of the mission planning and execution of rendezvous and docking operations. The guidance system generates the desired state values, the navigation system determines the actual state, and the control system generates the force and torque commands to achieve the desired state. In simpler terms the navigation system answers the question "Where am I?", the guidance system "Where do I need to go?", and the control system "How do I get there?". These simple terms do not always directly apply, but provide a good intuitive explanation of the GNC system.

This chapter will discuss the GNC system in relation to rendezvous and docking operations, specifically focussing on the guidance and control system. Section 4-1 will introduce the main tasks and functions of a general satellite GNC system. A more in-depth discussion on the guidance and control system and the algorithms that lie at the base of these systems are provided in Sections 4-2 and 4-3. The basic guidance problem for the reference scenario is formulated in Section 4-4. Finally, Section 4-5 will provide an overview of the different guidance and control strategies that are considered in this thesis.

4-1 Tasks and Functions of the GNC System

4-1-1 Definitions

The introduction of this chapter explained the functions of the GNC system in simple terms. Before delving deeper into the concepts and algorithms used in GNC system development, it is useful to provide a set of exact definitions of GNC that are used throughout this report. The following definitions are adapted from Fehse (2003) and Pfeiffer (1968).

Guidance "has the task of calculating and executing a realizable acceleration profile which will cause the trajectory of the space vehicle to attain desired conditions."

Navigation "has the task to provide the controller and the guidance function with the necessary information on the present state of the vehicle. "

Control "has the task to provide the force and torque commands which will be executed by the actuators of the spacecraft to correct the deviations of the estimated state vector from the desired state vector."

The attentive reader will have noticed that both the guidance and control system provide commands to achieve the desired trajectory. The difference between these commands will be explained further in Section 4-2.

From these definitions it becomes clear that the heart of the GNC system is an onboard computer. This computer processes the implemented guidance algorithms, control laws, and navigation filters. The hardware that provides the necessary input information for the navigation system are sensors. These can be, for example, star trackers, inertial measurement units, or cameras. The control commands are in the end executed by, for example, thrusters or reaction wheels.

4-1-2 Architecture and System Interaction

To execute the tasks defined in the previous sections the guidance, navigation, and control system have to interact with each other and with other subsystems. A generic architecture of a GNC system and the systems it interacts with is provided in Figure 4-1.

The guidance system needs the actual state/trajectory of the satellite to generate the desired state/trajectory and corresponding acceleration profile. This input is provided by the navigation system, which uses sensors and predefined theoretical models to supply the required information. The task of the control system is to produce the force commands necessary to eliminate the error between the desired state and the estimated state. It receives input from the guidance system on the desired trajectory and attitude profiles, and from the navigation system on the estimated state. Once the actuators have executed these commands the state of the spacecraft will change. The state is not only effected by the control forces and torques on the spacecraft, but also by the environmental disturbances, such as atmospheric drag and solar radiation pressure.

It is unavoidable that in this sequence of operations errors will arise. There will be, for example, errors in the estimation of the state, or the thrusters will not execute the force exactly as modelled. It is the task of the GNC system to be sufficiently robust to achieve the

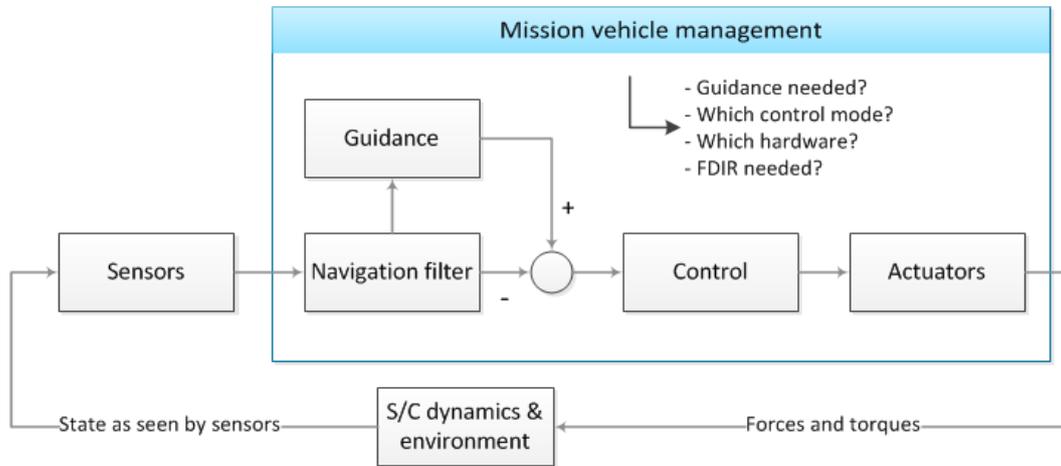


Figure 4-1: Overview of a generic GNC system interaction and architecture.

desired state within the required accuracy, as defined in Section 2-2, in the presence of such errors.

The overarching system is the mission vehicle management system. Its primary function is the selection of the operations mode management, i.e., it selects the GNC mode and corresponding hardware according to the different requirements for different mission phases and circumstances. It is further responsible for safe spacecraft operations through Fault Identification, Detection, and Recovery (FDIR) and thus also for the initiation of CAMs when necessary.

4-2 Guidance and Control System

Having established a basic understanding of GNC, the guidance and control systems can now be examined further. Figure 4-2 provides a more detailed overview of the tasks and functions of these systems. Recall, that only translational dynamics are considered for the reference scenario. Therefore, control in this case refers to orbit control and corresponds to the path tracking part. The path planning part is thus essentially the guidance function.

The path planning system consists of an algorithm that provides the trajectory from the current location to the desired location and the corresponding acceleration profile, i.e., the feed-forward commands, using a predefined relative motion model. The algorithm solves the optimization problem using a predefined performance index and constraints. According to Jacobsen et al. (2002), optimization based path planning comprises three fundamental steps. In the first step, the path is expressed in terms of parameters that describe its kinematic shape and velocity profile along the path. The second step consists of defining a cost function that evaluates the performance of the path for each set of parameters based on performance metrics and applies a penalty for the violation of constraints. Finally, an optimization routine

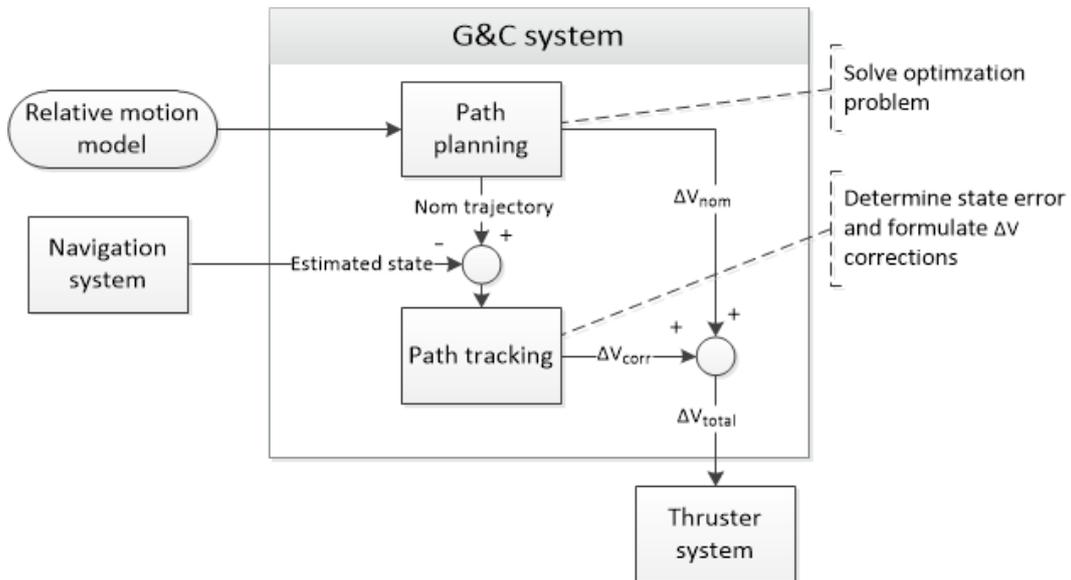


Figure 4-2: Overview of the guidance and control system tasks and functions.

is used to determine the values of the path parameters that result in the lowest total cost. The trajectories can, for example, be optimized for minimal fuel consumption or safety. Other factors that can play a role in the final approach phase are plume impingement and Line-Of-Sight (LOS) requirements.

The reference trajectory, provided by the guidance system, and the estimated state, provided by the navigation system, are the input for the path tracking subsystem. The path planning system is computationally intensive and therefore updated at a frequency that will often have the result that the feed-forward commands are insufficient to achieve the desired state in the presence of unmodeled perturbations and errors. The path tracking system is used to correct for this, which is a computationally less intensive system and can therefore function at a higher update frequency. The path tracking subsystem employs a predefined type of control algorithm to compute the force commands required to eliminate the error between the desired and estimated state.

4-3 Guidance and Control Algorithms

Three classes of top-level guidance algorithms can be identified: open-loop guidance, explicit algorithms based on closed-form expressions, and algorithms rooted in numerical techniques. Open-loop algorithms are those that do not incorporate state (or output) feedback. An example would be a trajectory based on pre-computed guidance commands that is generated at mission control and then uploaded to a spacecraft. This is not suitable for the final phases of a rendezvous and docking mission, because errors in, for example, the predicted future states of the target cannot be corrected for and re-targetings are not an option. Such a method would overall provide very low robustness. The second class, explicit methods, make up the most-used guidance algorithms. In these methods, simplifying assumptions are made that yield analytical expressions for the acceleration commands. Such laws are computationally

efficient, as they provide closed-form expressions. A disadvantage is that they do not deliver globally optimal solutions (Rea, 2009).

Guidance algorithms rooted in numerical optimization form the last class. As discussed in Chapter ?? the computational power of on-board computers has increased significantly over recent years. Together with the steps made in optimization theory this has led to new guidance algorithms for rendezvous that are rooted in numerical optimization. Convex guidance, that is based on convex optimization theory (Boyd and Vandenberghe, 2009) is capable of delivering globally-optimal solutions, while at the same time constraining the trajectory. The main advantages of guidance based on convex optimization are that 1) Convex optimization theory proves that a well-posed convex problem is guaranteed to converge, 2) the obtained solution will be the global optimum, 3) there are a number of different, very efficient solvers for this kind of problem, and 4) constraints and penalties can be imposed.

The guidance algorithm to be developed will also be implemented on TEAMS. Successful rendezvous and docking operations have been achieved on TEAMS using convex optimization (private communication F. de Bruijn, 2014). Together with the evident benefits of convex optimization, this is the reason that convex guidance has been selected as the basis for the path planning algorithm of the guidance system. The theory and construction of the path planning algorithm will be discussed in-depth in Chapter 5.

The path-tracking subsystem employs a control algorithm to determine how to correct for the error between the desired state and the estimated state. The general control law in the case of state feedback control is given by Equation (4-1), where \mathbf{K} is the gain matrix which can be tuned to alter the behaviour of the system. The system is mathematically depicted by Equation (4-2), where $\mathbf{x}(t)$ is an n -dimensional state vector of the system, $\mathbf{u}(t)$ is an m -dimensional input (or control) vector that is linear in the system state (Paraskevopoulos, 2001). The coefficient matrices \mathbf{A} and \mathbf{B} represent the system and input matrices, and are an $n \times n$ and $n \times m$ matrix, respectively.

$$\mathbf{u}(t) = -\mathbf{K}\mathbf{x}(t) \quad (4-1)$$

$$\dot{\mathbf{x}}(t) = \mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{u}(t) \quad (4-2)$$

Inserting the control law into this system gives the relation in Equation (4-3), for which the eigenvalues can be determined using Equation (4-4). This equation shows that the eigenvalues of the closed-loop system can be changed by varying the gain matrix \mathbf{K} (Paraskevopoulos, 2001).

$$\dot{\mathbf{x}} = (\mathbf{A} - \mathbf{B}\mathbf{K})\mathbf{x} \quad (4-3)$$

$$\det(\mathbf{A} - \mathbf{B}\mathbf{K} - \lambda\mathbf{I}) = 0 \quad (4-4)$$

The gain matrix \mathbf{K} can be computed in a mathematically closed form (Paraskevopoulos, 2001). An indirect method is Quadratic Optimal Control, which minimizes a mathematically defined cost criterion. A direct method is the pole placement, this method solves for \mathbf{K} on

basis of the specified poles of the closed-loop system. In the case where the cost function is described utilizing a quadratic criterion the cost function takes the form of given in Equation (4-5).

$$J = \int_0^{\infty} (\mathbf{x}^T \mathbf{Q} \mathbf{x} + \mathbf{u}^T \mathbf{R} \mathbf{u}) dt \quad (4-5)$$

In this equation the term $\mathbf{x}^T \mathbf{Q} \mathbf{x}$ represents the weighed state error and the term $\mathbf{u}^T \mathbf{R} \mathbf{u}$ the weighed control effort (Paraskevopoulos, 2001). \mathbf{Q} is a real positive semi-definite matrix, whereas \mathbf{R} is a real symmetric positive definite matrix. Varying \mathbf{Q} and \mathbf{R} thus changes the weights on the state error and control effort, respectively. By varying each of the elements of \mathbf{Q} and \mathbf{R} , each of the corresponding elements of \mathbf{x} and \mathbf{u} can be addressed. \mathbf{Q} and \mathbf{R} can be defined using, for example, using 'Bryson's Rule':

$$\mathbf{Q} = \text{diag} \left\{ \frac{1}{\Delta x_{1 \max}^2} \frac{1}{\Delta x_{2 \max}^2} \frac{1}{\Delta x_{n \max}^2} \right\} \quad (4-6)$$

with $\Delta x_{i \max}$ the maximum allowable magnitude of the i -th element of the state error, and

$$\mathbf{R} = \text{diag} \left\{ \frac{1}{\Delta u_{1 \max}^2} \frac{1}{\Delta u_{2 \max}^2} \frac{1}{\Delta u_{n \max}^2} \right\} \quad (4-7)$$

where $\Delta u_{j \max}$ is the maximum allowable value of the j -th control. Based on empirical results these matrices can then be tuned further if necessary.

Using Equation (4-1), Equation (4-5) can be rewritten to Equation (4-8). The gain matrix \mathbf{K} is then found using Equation (4-9), in which the matrix \mathbf{P} is positive definite and can be solved from the Ricatti equation in (4-10).

$$J = \int_0^{\infty} \mathbf{x}^T (\mathbf{Q} + \mathbf{K}^T \mathbf{R} \mathbf{K}) \mathbf{x} dt \quad (4-8)$$

$$\mathbf{K} = \mathbf{R}^{-1} \mathbf{B}^T \mathbf{P} \quad (4-9)$$

$$\mathbf{A}^T \mathbf{P} + \mathbf{P} \mathbf{A} - \mathbf{P} \mathbf{B} \mathbf{R}^{-1} \mathbf{B}^T \mathbf{P} + \mathbf{Q} = \mathbf{0} \quad (4-10)$$

The controller obtained using the preceding analysis is known as the Linear Quadratic Regulator (LQR), which is a well-understood controller applied in many applications, often to track some desired trajectory. This type of controller is embedded in the orbit control function of the functional simulator as well as in the the orbit and attitude control functions on TEAMS, because of its low computational needs and its easy implementation in the used simulation environment, which will be discussed in Chapter 6.

4-4 Guidance Problem Formulation for the Reference Scenario

The previous sections of this chapter provided a good understanding of the guidance system and its main functions. Together with the description of the mission reference scenario in Section 2-2 and the HCW equations in Section 3-3-2, this provides the basis for the guidance problem formulation, as depicted below. Recall, that the mission goal is to have the chaser satellite perform a safe, reliable, and propellant-efficient rendezvous and docking with the target satellite. Let $\mathbf{X} = (\mathbf{r}^T \dot{\mathbf{r}}^T)^T$ be the trajectory state of the chaser in the rotating Hill frame, T_x , T_y , and T_z the thrust of the chaser in the x, y and, z direction, respectively, T_{max} the maximum thrust that can be applied in the x, y, and z direction and t_0 and t_f the specified initial time and final time, respectively. The basic optimization problem for this scenario is then given by:

$$\begin{aligned}
 & \text{minimize } \int_0^{t_f} \|\mathbf{T}\|_2 / dt \\
 & \text{subject to} \\
 & \text{HCW equations of motion} \\
 & \mathbf{r}(0) = \mathbf{r}_0, \quad \dot{\mathbf{r}}(0) = \dot{\mathbf{r}}_0, \quad \mathbf{r}(t_f) = \mathbf{r}_{t_f}, \quad \dot{\mathbf{r}}(t_f) = \dot{\mathbf{r}}_{t_f} \\
 & T_i \leq T_{max} \quad i = x, y, z
 \end{aligned} \tag{4-11}$$

This problem now includes constraints on the maximum thrust and specifies the initial and final relative states. The implementation of constraints on the safety and the adaptations needed to this basic problem formulation to implement it in the functional simulator will be discussed further in Chapter 5.

4-5 Implemented Guidance and Control Strategies

The guidance system usually functions at a lower update frequency than the control system, as explained in Section 4-2. In theory, if the guidance system is updated at a high enough rate, the feed-forward commands are sufficient to achieve the desired trajectory and in that case the the guidance system essentially includes both the guidance and control functions. This notion forms the basis of the philosophy behind Model Predictive Control (MPC).

Figure 4-3 depicts the concept of model predictive control. A model predictive controller uses information on the current state and the system dynamics to predict how future control accelerations change the system state De Bruijn and Gill (2014). Using this information, a trajectory will be planned (open-loop) for a certain specified time, i.e., the planning horizon or prediction horizon. This planned trajectory will deviate from the actual trajectory due to modeling errors and unmodeled perturbations. To reduce the effect of these errors only a fraction of the plan is executed. At the next guidance update step, the system state is again obtained and the guidance function re-initiates the planning algorithm and the process repeats itself. This process is therefore also known as receding horizon control. One of the main advantages, which is of great importance for this research, is thus that it allows for an unconstrained time-to-go, i.e., an unconstrained time to reach the final desired state.

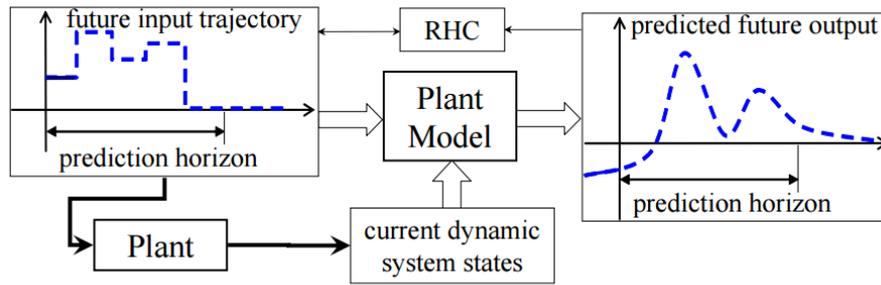


Figure 4-3: Conceptual visualization of MPC (Gorinevsky, 2005).

MPC has been applied multiple times before in spacecraft rendezvous research (Hartley et al., 2012), (Saponara et al., 2013), (Di Cairano et al., 2012). Richards and How (2003) has shown, for example, that employing a model predictive controller in spacecraft rendezvous results in less use of propellant than a more traditional glideslope controller. Park et al. (2011) applied MPC in the rendezvous with a tumbling target and showed that predicting the motion of the docking port and the changes in the LOS constraints allows for satisfactory performance of maneuvers initiated when the spacecraft is further away from the platform and when the platform is rotating at a higher rate ($\omega = 7$ deg/s). The prediction also contributed to the reduction of fuel consumption.

A main advantage that can be exploited in the reference scenario is that MPC allows for an unconstrained time-to-go, because of the "moving" planning horizon. Three guidance and control strategies are constructed to assess the performance of MPC for the reference scenario:

1. NUM-OL + LQR: In this method the desired path is a precomputed open-loop numerical reference trajectory including the corresponding feed-forward acceleration commands. This trajectory is then tracked by an LQR to account for disturbances that are not taken into account in the generation of the numerical reference.
2. MPC+LQR: This method uses the MPC method to compute the numerical reference trajectory and corresponding acceleration profile at each guidance update step. This trajectory is then also tracked by an LQR at a higher sampling rate than the guidance function to account for disturbances that are not taken into account in the generation of the numerical reference.
3. MPC: This method solely applies MPC, where the guidance and control function are essentially combined into one. It runs at the same sampling rate as the LQR in the MPC+LQR method and the control is thus solely based on the feedforward commands generated by the MPC.

The first strategy does not employ MPC but computes the numerical reference trajectory in an open-loop manner. A similar but computationally more efficient option would be to track an analytical reference trajectory using an LQR. The main disadvantage with this method is that no constraints can be applied to avoid collisions with the target, and will therefore not be considered in this thesis research. The second strategy, MPC+LQR, is computationally

less intensive than the MPC strategy, because of the lower guidance update rate. The third strategy, MPC, although computationally the most intensive, is expected to achieve a better accuracy due to this higher update rate.

All the proposed strategies use the solution provided by the convex optimization solver in the path planning algorithm to provide the feed-forward acceleration commands at each update step.

Convex Guidance

Optimality plays an important role in spacecraft guidance, this most often means finding optimal trajectories within a set of constraints. The quantity to be minimized is typically the overall thrust, to decrease the required propellant mass. Constraints can be placed on, for example, final conditions, but can also ensure safety or avoidance of plume impingement. Section 5-1 will provide background information on convex optimization theory, which is needed to understand the guidance algorithm to be developed. The basic formulation of the optimization problem for the reference scenario was already provided in Section 4-4. Section 5-2 will develop this optimization problem further to address the requirements on safety and convert it into a format that is implementable in the functional simulator.

5-1 Convex Optimization

Mathematical optimization in general is the task of finding the subset of extreme values, either maxima or minima, in a given set. The first step in solving an optimization problem is to formulate it correctly. The standard form of an optimization problem is given in Equation (5-1), where $\mathbf{x} \in \mathbb{R}^n$ holds the optimization variables and the function $f_0 : \mathbb{R}^n \rightarrow \mathbb{R}$ is the objective function.

$$\begin{aligned} & \text{minimize } f_0(\mathbf{x}) \\ & \text{subject to } g_i(\mathbf{x}) \leq 0 \quad i = 1, \dots, m \\ & \quad \quad \quad h_j(\mathbf{x}) = 0 \quad j = 1, \dots, p \end{aligned} \tag{5-1}$$

This formulation thus states the problem of finding a solution \mathbf{x} that minimizes the function $f_0(\mathbf{x})$, while satisfying the constraints listed under "subject to". $g_i : \mathbb{R}^n \rightarrow \mathbb{R}, i = 1, \dots, m$ are the inequality constraint functions and $h_j : \mathbb{R}^n \rightarrow \mathbb{R}, j = 1, \dots, p$ form the equality constraint functions. Thus, the problem is subject to n variables and $m + p$ constraints. A maximization problem is formulated by simply negating the objective function. All problems in this thesis

report will be minimization problems. The vector \mathbf{x}^* is defined as an optimal solution to the optimization problem, i.e., it has the smallest objective value of all possible vectors in the set. Mathematically this translates into, \mathbf{x}^* is the optimum if for any vector \mathbf{z} the following holds:

$$f_0(\mathbf{z}) \geq f_0(\mathbf{x}^*) \quad (5-2)$$

The types of functions that specify the objective f_0 and the constraints g_i and h_j determine what kind of optimization problem it is. Different types of optimization problems come with different solution methods for finding the minimum. If all functions and the search space D are convex, convex optimization theory can be applied to find the solution to the optimization problem. Even though problems cannot always be solved analytically and are often non-linear, very efficient and robust methods for finding solutions exist in the field of convex optimization theory.

The purpose of this section is to provide the background information on convex optimization, needed to understand the guidance algorithms developed in the remainder of this chapter. Convex sets and functions, and convex optimization problems will be discussed in this section. The concepts discussed in this section are mainly based on Boyd and Vandenberghe (2009). For more detailed and in-depth information on convex optimization theory, the reader is referred to this reference.

5-1-1 Convex Sets

In the case that the objective function is linear and the constrained space is also represented by linear equalities and linear inequalities, the problem is a linear programming problem (Luenberger and Ye, 2008). For a linear program the objective function and the constraints thus satisfy the general condition in Equation (5-3) (Boyd and Vandenberghe, 2009).

$$f_i(\alpha\mathbf{x} + \beta\mathbf{y}) = \alpha f_i(\mathbf{x}) + \beta f_i(\mathbf{y}) \quad (5-3)$$

Linear programming can be considered to be a subset of convex programming. In a convex optimization problem the objective is convex, the inequality constraints are convex and the equality constraints are affine (Boyd and Vandenberghe, 2009). A set C is affine if the line through any two distinct points in C lies in C . The convex condition is mathematically depicted as:

$$f_i(\alpha\mathbf{x} + \beta\mathbf{y}) \leq \alpha f_i(\mathbf{x}) + \beta f_i(\mathbf{y}) \quad (5-4)$$

with $\alpha + \beta = 1$, $\alpha \geq 0$ and $\beta \geq 0$.

In general, a set C is considered to be convex if the line segment between any two points in C lies in C , which means roughly that every point in the set can be "seen" by every other point in the set, along an unobstructed straight path between them (Boyd and Vandenberghe, 2009). Several examples of convex and non-convex sets are given in Figure 5-1. The hexagon at the left of this figure, which includes its boundary, is convex. The kidney shaped set, in the middle, is not convex, because the line segment between the two points in the set does not

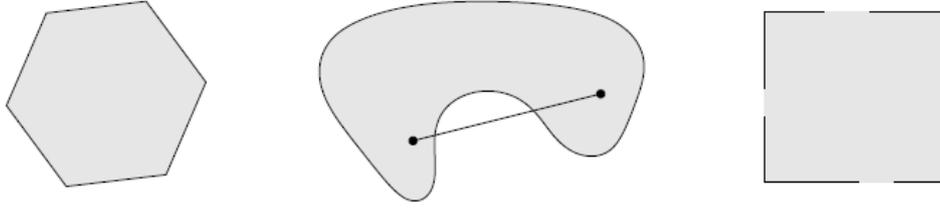


Figure 5-1: Examples of convex and nonconvex sets (Boyd and Vandenberghe, 2009).

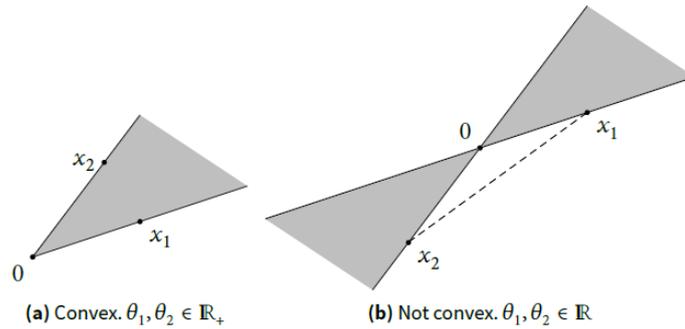


Figure 5-2: Examples of convex cones (Boyd and Vandenberghe, 2009).

lie completely in the set. On the right, a square is depicted, which contains some boundary points but not others, and is therefore not convex.

The most commonly known characterization of convexity is that if a function f is differentiable twice, then it is convex if the associated Hessian matrix $\nabla^2 f$ is positive semi-definite (all entries are non-negative) on the entire domain of f . An advantageous characteristic of convex problems is that, any locally optimal point is globally optimal.

A special type of convex sets are convex cones. A set C is defined as a convex cone if for any $\mathbf{x}_1, \mathbf{x}_2 \in C$ and any scalars $\theta_1 \geq 0$ and $\theta_2 \geq 0$, $\theta_1 \mathbf{x}_1 + \theta_2 \mathbf{x}_2 \in C$. The cone is thus the set spanned by an origin and the lines going through \mathbf{x}_1 and \mathbf{x}_2 , as shown on the left-hand side of Figure 5-2. From this figure it is clear that any two points in this set can be connected without the connecting line going outside of the set, intuitively explaining the convexity of cones. An example for a cone that is not convex is shown on the right-hand side of Figure 5-2.

A specific type of convex cone, the second-order cone, will prove to be important in the formulation of the safety constraint for the reference mission scenario. The second-order cone, corresponding to the Euclidean norm, is described by:

$$\begin{aligned}
 C &= \{(\mathbf{x}, t) \mid \|\mathbf{x}\| \leq t\} \subseteq \mathbb{R}^{n+1} \\
 &= \left\{ \begin{bmatrix} \mathbf{x} \\ t \end{bmatrix} \begin{bmatrix} \mathbf{x} \\ t \end{bmatrix}^T \begin{bmatrix} I_{n \times n} & 0 \\ 0 & -1 \end{bmatrix} \begin{bmatrix} \mathbf{x} \\ t \end{bmatrix} \leq 0, t \geq 0 \right\}
 \end{aligned} \tag{5-5}$$

The physical shape of the second-order cone and its intuitive convexity is portrayed in Figure

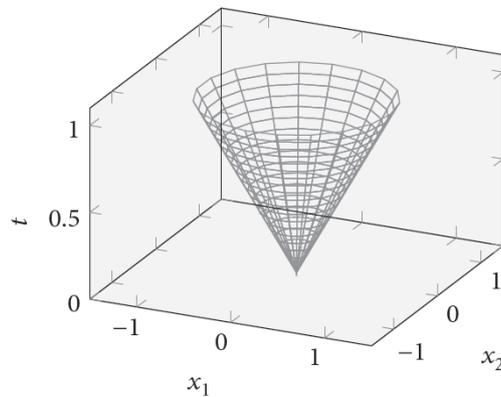


Figure 5-3: Boundary of the second-order cone in \mathbb{R}^3 , the inside of the cone is also included in the convex space (Boyd and Vandenberghe, 2009).

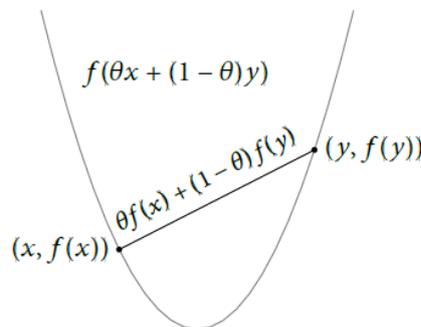


Figure 5-4: Graph of a convex function. The line between any two points on this graph lies above the graph (Boyd and Vandenberghe, 2009)

5-3. Note that the cone is filled, thus the convex space includes the inside of the cone.

5-1-2 Convex Functions

Optimization problems are only convex when all functions involved are convex. But how can it be determined if a function is convex? It is not always intuitive and therefore techniques have been developed to assess the convexity of functions. The basic three conditions that have to be fulfilled for a function $f: \mathbb{R}^n \rightarrow \mathbb{R}$ to be convex are: (1) the domain $\text{dom } f$ is a convex set; (2) all $\mathbf{x}, \mathbf{y} \in \text{dom } f$; and (3) $0 \leq \theta \leq 1$. The following must then hold:

$$f(\theta \mathbf{x} + (1 - \theta) \mathbf{y}) \leq \theta f(\mathbf{x}) + (1 - \theta) f(\mathbf{y}). \quad (5-6)$$

This inequality can be understood geometrically, as depicted in Figure 5-4. For a function to be convex, the line that connects any two points of a function must lie above the function's graph.

For a function to be convex, it must also fulfil first-order and second-order necessary conditions. The first-order condition states that: if f is differentiable, then f is convex if $\text{dom } f$ is

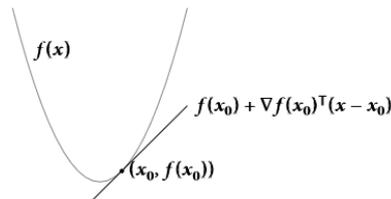


Figure 5-5: The first-order condition for convex functions (Boyd and Vandenberghe, 2009).

convex and

$$f(\mathbf{x}) \geq f(\mathbf{x}_0) + \nabla f(\mathbf{x}_0)^T(\mathbf{x} - \mathbf{x}_0) \quad (5-7)$$

holds for all $\mathbf{x}, \mathbf{y} \in \text{dom } f$. The right-hand side of the inequality is not only an affine function, but also a first-order Taylor approximation of the function $f(\mathbf{x})$, this is depicted in Figure 5-5. The inequality means that the Taylor approximation is a global underestimator if the function is convex: the approximated line always lies below the graph of the convex function. This is an important conclusion, seeing as it shows that from local information, global information can be derived.

The second-order condition states that the Hessian must be positive semi-definite. If it is assumed that f is twice differentiable, i.e., the Hessian $\nabla^2 f$ exists at each point in $\text{dom } f$. Then f is convex and only if $\text{dom } f$ is convex and $\nabla^2 f$ is positive semi-definite: for all $\mathbf{x} \in \text{dom } f$

$$\nabla^2 f \succeq 0. \quad (5-8)$$

This entails that the derivative must be non-decreasing, i.e., it must have upward curvature at x . There are also convex functions that are not differentiable, it is thus important to note that these conditions are not requirements. An example of such a function is $f(x) = |x|$: although it is not differentiable at $x = 0$ it is still a convex function.

5-1-3 Convex Optimization Problems and Second-Order Cone Programs

Recall the general form of optimization problems, as given in Equation (5-1). In Section 5-1 it was discussed that convex optimization problems possess certain attractive properties, that make their solution simple, efficient, and robust.

This section will discuss the formulation of convex optimization problem and explain on a high-level the reasons behind the benefits that come with convex optimization problems.

The standard form of convex optimization problems is given in Equation (5-9), in which f_0 and g_1, \dots, g_m are all convex functions.

$$\begin{aligned} & \text{minimize} && f_0(\mathbf{x}) \\ & \text{subject to} && g_i(\mathbf{x}) \leq 0 \quad i = 1, \dots, m \\ & && \mathbf{a}_j^T \mathbf{x} = b_j \quad j = 1, \dots, p \end{aligned} \quad (5-9)$$

There are three additional requirements that must be fulfilled for convex optimization problems in comparison to general optimization problems:

- The objective function f_0 must be convex;
- The inequality constraint functions must be convex; and
- The equality constraint functions $h_i(\mathbf{x}) = \mathbf{a}_i^T \mathbf{x} - b_i$ must be affine.

The result of these requirements is that the feasible set of the problem (its domain or solution space) is also a convex set (the intersection of convex sets is a convex set again).

A special class of convex optimization problems are in the format of a Second Order Cone Program (SOCP). These are optimization problems, which fulfil some additional requirements:

- The objective function must be affine (linear); and
- The inequality constraints are convex cones; and
- The equality constraints are affine.

Mathematically this is described as:

$$\begin{aligned}
 & \text{minimize } f(\mathbf{x}) \\
 & \text{subject to} \\
 & \|\mathbf{A}_i \mathbf{x} + \mathbf{b}_i\| \leq \mathbf{c}_i^T \mathbf{x} + \mathbf{d}_i, \quad i = 1, \dots, m \\
 & \mathbf{F} \mathbf{x} = \mathbf{g}
 \end{aligned} \tag{5-10}$$

where $A_i \in \mathbb{R}^{k \times n}$ and $F \in \mathbb{R}^{b \times n}$. To understand why this is called a "cone problem", recall Equation (5-5), which represented a set described by a second-order cone, and the visual representation of this cone in Figure 5-7. By analogy it is clear that the inequality constraint in Equation (5-10) also describes a cone:

$$C = \{(\mathbf{A}_i \mathbf{x} + \mathbf{b}_i, \mathbf{c}_i^T \mathbf{x} + \mathbf{d}_i) \mid \|\mathbf{A}_i \mathbf{x} + \mathbf{b}_i\| \leq \mathbf{c}_i^T \mathbf{x} + \mathbf{d}_i\} \subseteq \mathbb{R}^{k+1} \tag{5-11}$$

Note that SOCPs are equivalent to other types of optimization problems, such as quadratically-constrained quadratic-programs (by setting $\mathbf{c}_i = 0$ and squaring the constraints) or linear programs (by setting $\mathbf{A}_i = 0$) (Gerth, 2014).

A simple example will illustrate why the solution for convex problems are always global minima. Figure 5-6 illustrates that if a function is convex over its domain $\text{dom } f$, then there can necessarily be only one global optimum. A local minimum will therefore never be the solution to a convex optimization problem. Another advantage is that the minimum can be found rather easily because the negative gradient will always lead to the minimum.

It must be noted that the preceding analogy should be considered with caution. It is important to realize that not every convex set is as easy to visualize as the example in Figure 5-6, it might even be impossible. This example is merely treated here because it intuitively shows this aspect of convex optimization problems.

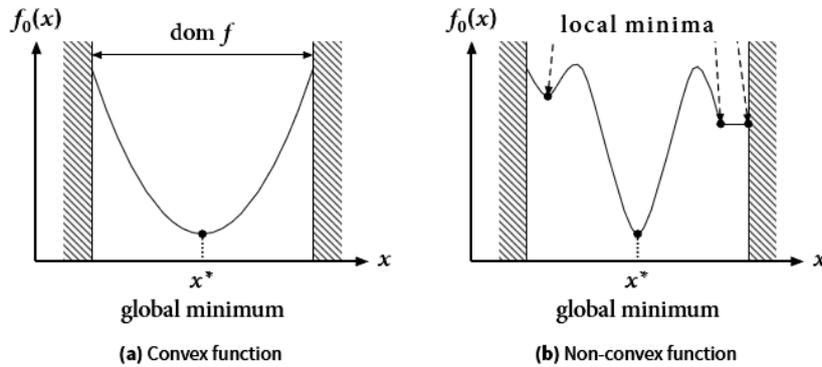


Figure 5-6: Optimal solutions in convex and non-convex functions. Convex functions can only have one single, globally optimal solution. Note that in (b) the entire straight line is a local minimum (Boyd and Vandenberghe, 2009).

5-2 Formulation Convex Optimization Problem for the Reference Scenario

The basic optimization problem formulation was treated in Section 4-4. When this problem is tested against the conditions for a convex optimization problem it is evident that this problem formulation is already in a convex format. The objective function and inequality constraint are norm functions which are proven to be convex functions (Boyd and Vandenberghe, 2009) and the HCW equations are a set of linear, time-invariant differential equations, which are affine.

5-2-1 Discretization

Having established that the optimization problem is convex, it now has to be discretized to obtain a problem that can be solved using convex programming. The given time interval $t \in [t_0, t_f]$ is divided into a number of steps N . Let each time step be denoted by Δt , then the time at node k is simply described by:

$$t_k = k\Delta t + t_0, \quad k = 0, 1, \dots, N \quad (5-12)$$

The relative motion model is discretized using the first order forward method and is given by Equation (5-13).

$$\mathbf{x}(k+1) = \mathbf{A}_d \mathbf{x}(k) + \mathbf{B}_d \mathbf{u}(k) \quad (5-13)$$

The discrete matrices \mathbf{A}_d and \mathbf{B}_d are related to the continuous state-space matrices A and B by:

$$\mathbf{A}_d = I_{6 \times 6} + \mathbf{A}\Delta t, \quad \mathbf{B}_d = \mathbf{B}\Delta t \quad (5-14)$$

which is a simple Euler integration. The integration can also be done using a higher-order analysis, but this would lead to a more complex discretization procedure, which is deemed unnecessary at this point in the development.

The dynamics for subsequent time steps depend on the initial state and the control inputs at all subsequent time steps (De Bruijn and Gill, 2014). This is illustrated by the sequence in Equation (5-15).

$$\begin{aligned}
\mathbf{x}(k+1) &= \mathbf{A}\mathbf{x}(k) + \mathbf{B}\mathbf{u}(k) \\
\mathbf{x}(k+2) &= \mathbf{A}\mathbf{x}(k+1) + \mathbf{B}\mathbf{u}(k+1) \\
&= \mathbf{A}^2\mathbf{x}(k) + \mathbf{A}\mathbf{B}\mathbf{u}(k) + \mathbf{B}\mathbf{u}(k+1) \\
\mathbf{x}(k+N) &= \mathbf{A}^N\mathbf{x}(k) + \mathbf{A}^{N-1}\mathbf{B}\mathbf{u}(k) + \dots + \mathbf{B}\mathbf{u}(k+N-1)
\end{aligned} \tag{5-15}$$

These equations can be combined into one matrix equation:

$$\mathbf{Z}(k) = \mathbf{F}\mathbf{x}_0 + \mathbf{H}\mathbf{U}(k) \tag{5-16}$$

where \mathbf{x}_k has been denoted by \mathbf{x}_0 , which is the state at the current time step and

$$\begin{aligned}
\mathbf{Z}(k) &= \begin{bmatrix} \mathbf{x}(k+1) \\ \mathbf{x}(k+2) \\ \vdots \\ \mathbf{x}(k+N) \end{bmatrix}, \quad \mathbf{U}(k) = \begin{bmatrix} \mathbf{u}(k) \\ \mathbf{u}(k+1) \\ \vdots \\ \mathbf{u}(k+N-1) \end{bmatrix}, \quad \mathbf{F} = \begin{bmatrix} \mathbf{A} \\ \mathbf{A}^2 \\ \vdots \\ \mathbf{A}^N \end{bmatrix} \\
\mathbf{H} &= \begin{bmatrix} \mathbf{B} & 0 & & \\ \mathbf{A}\mathbf{B} & \mathbf{B} & 0 & \\ \vdots & \vdots & \ddots & \\ \mathbf{A}^{N-1}\mathbf{B} & \mathbf{A}^{N-2}\mathbf{B} & \dots & \mathbf{B} \end{bmatrix}
\end{aligned}$$

Different Discretization Time Steps

As stated before, the Euler method is used in the discretization. To reduce the error induced by applying the Euler method, it is therefore desirable to use a small time step Δt . A setback with such a small time step is that it significantly reduces the planning horizon, i.e., $N\Delta t$, for a fixed value of N . To compensate for this, two different sizes for Δt are implemented. The first several time steps of the planning horizon will be equal to Δt_1 , which have a smaller size and the remaining time steps are equal to Δt_2 . The \mathbf{F} and \mathbf{H} matrix in Equation (5-16) are then adjusted according to:

$$\mathbf{F} = \begin{bmatrix} \mathbf{A}_1 \\ \mathbf{A}_1^2 \\ \vdots \\ \mathbf{A}_1^{N_s} \\ \mathbf{A}_2\mathbf{A}_1^{N_s} \\ \vdots \\ \mathbf{A}_2^{N-N_s}\mathbf{A}_1^{N_s} \end{bmatrix}$$

for $k = 1, \dots, N$ and \mathbf{Z}_{ref} is the discrete reference trajectory and \mathbf{U} is substituted by \mathbf{T} .

5-2-2 Formulation Safety Constraint

It is essential to the mission success that throughout the approach the chaser satellite does not intersect with the target satellite. To this end a safety cone-constraint is formulated, which was already briefly discussed in Section 2-2. This constraint ensures that the chaser stays within the space defined by the safety cone throughout the approach. To implement this constraint in the convex optimization problem, first a mathematical description is derived in continuous time in the form of a second-order cone-constraint (as presented in Equation (5-10)):

$$\|A_i \mathbf{x} + \mathbf{b}_i\| \leq \mathbf{c}_i^T \mathbf{x} + \mathbf{d}_i, \quad i = 1, \dots, m \quad (5-18)$$

This constraint then has to be reformulated to apply it in discrete time, this procedure is based on work presented in Gerth (2014).

The safety cone has its vertex at the target's CoM and extends outwards in the direction of the spin axis, as can be seen in Figure 5-7, where the z-axis is at all times aligned with the spin axis of the target. The three-dimensional geometry of the safety cone is defined by the safety angle θ . The angle between the position vector of the chaser (origin at the CoM of the target) and the spin axis is denoted by ϵ . It can then be said that at time t :

$$\tan \epsilon \, r_z = \sqrt{r_x^2 + r_y^2} \quad (5-19)$$

This equation represents a cone of a given angle ϵ . If the angle ϵ is replaced by the safety angle θ , the constraint can be written as:

$$\sqrt{r_x^2(t) + r_y^2(t)} \leq \tan \theta \, r_z(t) \quad \forall t \in (t_0, t_f) \quad (5-20)$$

which is then clearly formulated as a second-order cone-constraint.

Figure 5-7 clarifies the geometrical interpretation of this constraint. The line extending from the tip of the z-axis to the outer boundary of the cone represents the right-hand-side of Equation (5-20). The left-hand-side of this equation is represented by the orange line extending from the tip of the z-axis to the tip of the position vector (red arrow). The constraint enforces that the dashed red line in the xy-plane must always be smaller than the local cone radius, which means that the position vector is at all times inside of the cone.

To discretize this constraint the respective elements have to be selected from the solution vector \mathbf{Z} (which was given in Equation (5-17)). The matrix to extract r_z from the solution vector and multiply it by $\tan(\theta)$ at a particular time step $t(k)$ is:

$$\mathbf{E}_{r_z k} = \text{diag}\{0 \ 0 \ \tan \theta \ 0 \ 0 \ 0\} \quad (5-21)$$

The matrix to select r_x and r_y from \mathbf{Z} at a particular time step $t(k)$ is given by:

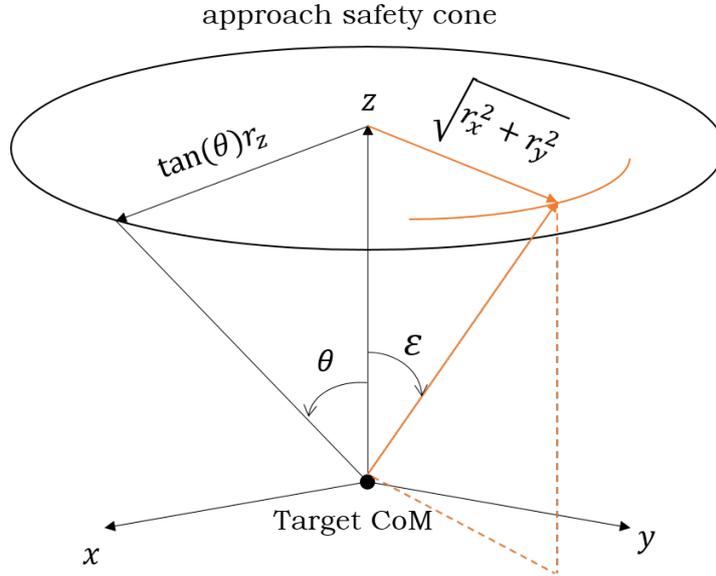


Figure 5-7: Geometry of the safety cone-constraint: the position vector of the chaser in the spin axis reference frame must at all times remain within the approach cone, defined by the safety angle θ .

$$\mathbf{E}_{r_{xy}k} = \text{diag}\{1 \ 1 \ 0 \ 0 \ 0 \ 0\} \quad (5-22)$$

In discrete time the safety cone constraint can then be formulated as:

$$\|\mathbf{E}_{r_{xy}k}\mathbf{E}_{k\mathbf{Z}}\mathbf{Z}\| \leq \mathbf{E}_{r_zk}\mathbf{E}_{k\mathbf{Z}}\mathbf{Z}, \quad k = 1, 2, \dots, N \quad (5-23)$$

where $\mathbf{E}_{k\mathbf{Z}}$ is the matrix that selects the state vector of the current time step out of the stacked solution vector \mathbf{Z} .

The discretized convex problem including all the constraints, for $k = 1, \dots, N$, then reads:

$$\begin{aligned} & \text{minimize} \quad \sum_{k=0}^N \|\mathbf{Q}\mathbf{T}\|_2 + \gamma\|\mathbf{s}\|_2 \\ & \text{subject to} \\ & \mathbf{Z} = \mathbf{F}\mathbf{x}_0 + \mathbf{H}\mathbf{T} \\ & \mathbf{Z}_{ref} = \mathbf{Z} + \mathbf{s} \\ & \|\mathbf{E}_{xyk}\mathbf{E}_{k\mathbf{Z}}\mathbf{Z}\| \leq \mathbf{E}_{r_zk}\mathbf{E}_{k\mathbf{Z}}\mathbf{Z} \\ & T_{x_k} \leq T_{max} \\ & T_{y_k} \leq T_{max} \\ & T_{z_k} \leq T_{max} \end{aligned} \quad (5-24)$$

Chapter 6

Functional Simulator Development

A key ingredient of the guidance and control system development is the capability to test and evaluate the designed algorithms. The first step in this testing sequence is to test the system using a functional simulator. A dedicated functional simulator is developed and verified using Matlab and Simulink. This chapter will treat the different steps in the development of this simulator. Section 6-1 will provide an overview of the software architecture. The implementation of the designed guidance and control algorithms and the necessary developments to achieve this are treated in Section 6-2. The propagation of the state of both the chaser and target satellite in continuous time is detailed in Section 6-3. The verification of the systems and models implemented in the functional simulator is covered in Section 6-4.

6-1 Software Architecture

The functional simulator is developed using a combination of Matlab and Simulink. The top-level architecture of this simulator is presented in Figure 6-1. The figure shows that the path-planning block uses information on the current relative state and produces a set of feed-forward commands and the corresponding numerical reference trajectory. This block thus fulfils the guidance function. The path tracking block consists of an LQR controller, which was discussed in Section 4-3, which uses the current relative state and the numerical reference state as input. It produces a set of feedback commands that eliminates the error between the two inputs at that update step. This block thus fulfils the orbit control function. The target block includes the absolute satellite dynamics, including the J_2 effect, SRP, and atmospheric drag, which are propagated to obtain the new target state. The chaser satellite block uses the feed-forward and feedback accelerations as inputs and together with the satellite dynamics these are used to obtain the new state.

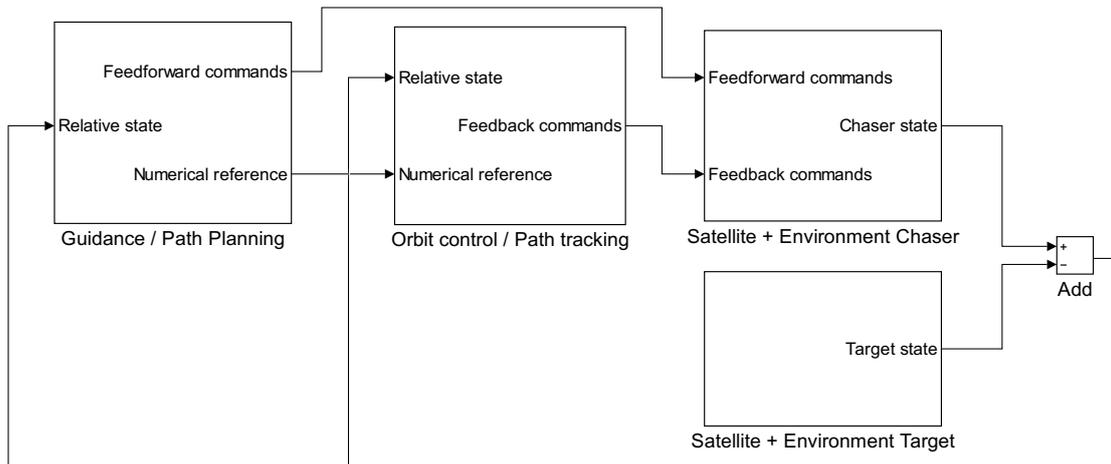


Figure 6-1: Top-level software architecture of the functional simulator.

6-2 Guidance and Control System Development and Implementation

6-2-1 Guidance System

In Section 5-2 the complete convex optimization problem for the reference scenario was presented. Solving this problem is a task of the guidance function. An overview of the tasks of the developed guidance function is presented in Figure 6-2. Based on the current relative state and the orientation of the target, an analytical reference trajectory is computed by the guidance function. The discrete reference states are denoted by \mathbf{Z}_{ref} in the optimization problem. This reference trajectory is generally infeasible. The trajectory is then fed into the convex optimizer, along with the current state, which solves the optimization problem presented in Equation (5-24). The convex optimizer produces a feasible trajectory, and the force commands to achieve this trajectory, are referred to as the feed-forward commands.

6-2-2 Generation of the Analytical Reference Trajectory

As discussed in the reference scenario description, ENVISAT is an uncooperative, rotating target. The approach strategy is defined such that throughout the approach the docking axis of the chaser satellite maintains alignment with the spin axis of the target satellite.

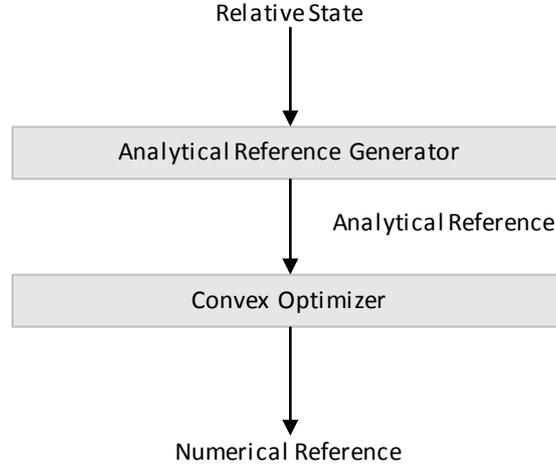


Figure 6-2: Overview of the main steps in the developed guidance algorithm.

The reference trajectory for the approach of the chaser towards the target is thus a straight trajectory along the spin axis, which will be followed with a constant velocity until the chaser is at distance of 3 m from the target. At this point, the chaser will no longer move towards the target but it will keep tracking the motion of the spin axis of the target. This description of the reference trajectory that can be described mathematically in quite simple terms. To maintain this it is desirable to introduce a new reference frame, which will be referred to as the spin axis reference frame, denoted by the subscript sa . It's definition is most easily explained, starting from the definition of the rotating Hill frame, depicted by the black axes frame in Figure 6-3. The spin axis reference frame, at $t = t_0$, is depicted by the blue axes frame and also has its origin at the CoM of the target. At $t = t_0$ the z_{sa} -axis and the x_{sa} -axis are constructed by rotating the z_h -axis and the x_h -axis around the y_h -axis over an angle β . At $t = t_1$, the z_{sa} -axis has moved over an angle δ_{t_1} , which is equal to ωt_1 . The z_{sa} -axis thus describes a circular motion around the z_h -axis over time, which resembles the precession of the spin axis, shown by the red dashed circle.

The HCW equations are defined in the rotating Hill frame and therefore the transformation matrix in Equation (6-1) is constructed to convert the trajectory from the spin axis frame to the rotating Hill frame.

$$\mathbf{T}_{sh} = \begin{bmatrix} \cos \beta \cos \delta(t) & -\sin \delta(t) & -\sin \beta \cos \delta(t) \\ \cos \beta \sin \delta(t) & \cos \delta(t) & -\sin \beta \sin \delta(t) \\ \sin \beta & 0 & \cos \beta \end{bmatrix} \quad (6-1)$$

Here the angle β is the constant offset of the spin axis from the H-bar and δ is the precession rate of the spin axis, which is equal to $\delta = \delta_0 + \omega t$.

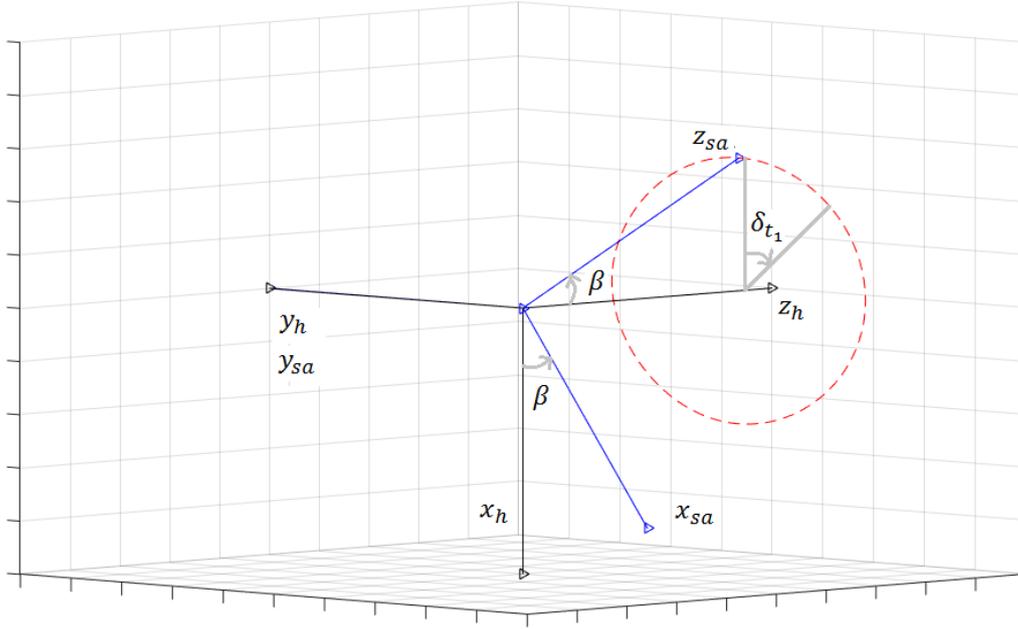


Figure 6-3: Definition of the spin axis reference frame (blue), starting from the rotating Hill frame (black).

The reference trajectory state $\mathbf{Z}_{\text{ref}_h} = [\mathbf{r}_h \dot{\mathbf{r}}_h]^T$ in the rotating Hill frame is then given by:

$$\mathbf{r}_h = \mathbf{T}_{sh} \mathbf{r}_{sa}, \quad \dot{\mathbf{r}}_h = \mathbf{T}_{sh} \dot{\mathbf{r}}_{sa} + \boldsymbol{\omega} \times \mathbf{r}_h \quad (6-2)$$

where $\boldsymbol{\omega} = [0 \ 0 \ \omega]^T$, and $\mathbf{Z}_{\text{ref}_{sa}} = [\mathbf{r}_{sa} \dot{\mathbf{r}}_{sa}]^T$ is the reference trajectory state in the spin axis frame. To clarify what the resulting path looks like in the rotating Hill frame, it is depicted in Figure 6-4.

6-2-3 Implementation of the Convex Optimizer

The convex optimization problem is solved at each update step of the guidance system. The concept of model predictive control is implemented by only executing the first time step of the feed-forward commands.

Multiple methods exist to solve convex optimization problems. For this thesis research these methods are not assessed, because it is decided to make use of an existing solver. This is a desirable option, because of the high level of maturity that convex-optimization solvers have reached over the last decade (Boyd and Vandenberghe, 2009). To solve the guidance problem numerically it has to be transcribed into a format that can be read by a solver. To avoid having to go through the cumbersome process of doing this by hand a modeling language is used for this.

For the functional simulator the modeling language CVX is used. This is a Matlab-based modeling system for convex optimization (Grant et al., 2008). CVX turns Matlab into a modeling language, which allows constraints and objectives to be specified using standard Matlab expression syntax. A number of standard problem types are supported by CVX,

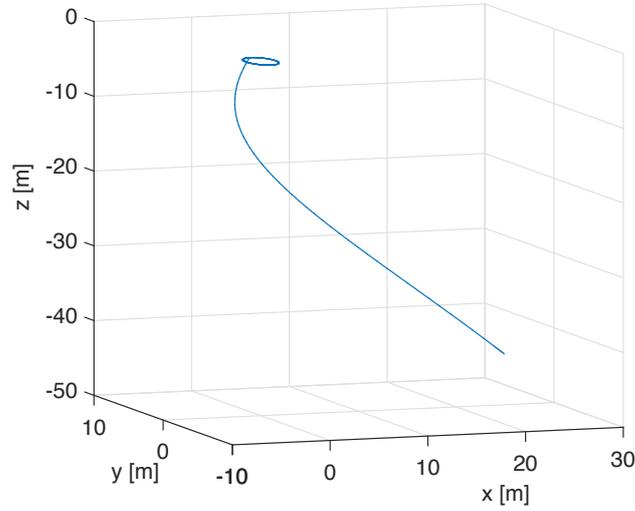


Figure 6-4: Reference trajectory in the rotating Hill frame.

including linear and quadratic programs, second-order cone programs, and semidefinite programs (SDPs). CVX can also solve much more complex convex optimization problems, but this is not of interest for the optimization problem applied in this research. The default solver in CVX is SDPT3. Its performance is sufficient to obtain satisfactory results in this research. SDPT3 is a Matlab implementation of infeasible primal-dual path-following algorithms (Tütüncü et al., 2003). For extended documentation on how to use CVX, the different solvers, and the limitations of the program, the reader is referred to Grant and Boyd (2014).

6-2-4 Control System

The control system employs a basic LQR controller, for which the concept and mathematical background was described in Section 4-3. The `dlqr` function in matlab is used to obtain the optimal gain matrix K . The function essentially computes the optimal gain matrix \mathbf{K} such that the state-feedback law given in Equation 4-1 minimizes the quadratic cost function given by Equation (4-8) for the discrete-time system. The inputs for this function are the continuous state matrices \mathbf{A}_c and \mathbf{B}_c given in Equation (6-3), which are based on the HCW equations (Equation (3-14)), and the weighing matrices \mathbf{Q} and \mathbf{R} .

$$A_c = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 3n^2 & 0 & 0 & 0 & 2n & 0 \\ 0 & 0 & 0 & -2n & 0 & 0 \\ 0 & 0 & -n^2 & 0 & 0 & 0 \end{bmatrix} \quad B_c = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 1/m_T & 0 & 0 \\ 0 & 1/m_T & 0 \\ 0 & 0 & 1/m_T \end{bmatrix} \quad (6-3)$$

The weighing matrices \mathbf{Q} and \mathbf{R} are based on Bryson's rule. The values on the diagonal of \mathbf{Q} correspond to the inverse of the squared maximum allowable position and velocity error and

the values on the diagonal of \mathbf{R} correspond to the inverse of the squared maximum control force that can be applied by the chaser. \mathbf{Q} and \mathbf{R} are then given by:

$$Q = \begin{bmatrix} 1/x_{err}^2 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1/y_{err}^2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1/z_{err}^2 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1/\dot{x}_{err}^2 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1/\dot{y}_{err}^2 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1/\dot{z}_{err}^2 \end{bmatrix} \quad R = \begin{bmatrix} 1/T_{max}^2 & 0 & 0 \\ 0 & 1/T_{max}^2 & 0 \\ 0 & 0 & 1/T_{max}^2 \end{bmatrix} \quad (6-4)$$

The maximum position error is set to 1 cm and the maximum velocity error is set to 1 mm. The maximum thrust that can be applied is equal to 44 N. The LQR controller is not tuned further, because this research does not focus on optimizing the LQR design.

6-3 State Propagation

In the scope of this research, numerical integration only needs to be applied in the state propagation of the chaser and target satellites. The Euler method is used in the discretization of the optimization problem, which is a first-order method and therefore there is no need for an integrator based on a model of a very high-order. The relatively simple, but sufficient, fourth-order Runge Kutta integrator is chosen to propagate the states of the satellites.

This method uses four function calls, in which the rate of change of the state at four points in time within the interval h are determined to obtain the new state at time $t_0 + h$. The general equations of the RK4 scheme are given in Equation (6-5) (Montenbruck and Gill, 2000).

$$\begin{aligned} \dot{x}_1 &= f(t_0, x_0) \\ \dot{x}_2 &= f\left(t_0 + \frac{h}{2}, x_0 + \frac{h}{2}\dot{x}_1\right) \\ \dot{x}_3 &= f\left(t_0 + \frac{h}{2}, x_0 + \frac{h}{2}\dot{x}_2\right) \\ \dot{x}_4 &= f(t_0 + h, x_0 + h\dot{x}_3) \\ x(t) &= x(t_0) + \frac{h}{6}(\dot{x}_1 + 2\dot{x}_2 + 2\dot{x}_3 + \dot{x}_4) \end{aligned} \quad (6-5)$$

The state propagation takes place in the Simulink model and the RK4 model is implemented by simply setting the Simulink solver to the ode4 (Runge-Kutta) mode. A fixed step-size of 0.1 s is used.

6-4 Software Verification

The software developed for the functional simulator should be verified to ensure that the system is reliable and functions correctly, i.e., to ascertain that the simulator accurately represents the conceptual description and specifications of the design.

6-4-1 Dynamics Model

The developed dynamical models are the same for the chaser and target satellite. The central gravity field and J_2 implementation are verified using the High Performance Simulator (HPS), a satellite simulator designed by DLR. The HPS consists of a set of MATLAB/Simulink libraries that include, o.a., simple and multi-body dynamics, and environmental models for the calculation of gravity, atmosphere, magnetic field, solar pressure, eclipse, and ephemerides. Most models are coded in (embedded) MATLAB/Simulink or C/C++ called via Simulink S-functions. The HPS has already undergone an extensive verification and validation process to ensure that only correctly documented and validated models are integrated in the library (Pelivan et al., 2012).

The gravity models of the HPS are of a format that allows for direct comparison with the gravitational model implemented in the functional simulator (central gravity field and J_2 effect). The difference between the resulting chaser/target state computed by the HPS and the one computed by the functional simulator, after approximately one orbit, is in the order of 10^{-8} m for the position and 10^{-8} m/s for the velocity. This difference is deemed to be sufficiently small to conclude that the gravity model is implemented correctly.

6-4-2 Guidance System

To verify the guidance system three test cases are constructed, which are described in Table 6-1. The numerical reference path constructed by the guidance system is compared to the analytical reference for these three test cases, for which the results are shown in Figure 6-5. It is clear from these figures that the guidance system produces trajectories that are very similar to the analytical reference trajectories, which shows that the guidance system functions correctly. The deviations from the analytical reference trajectory are caused by the fact that the optimizer minimizes a weighed combination of thrust and the error with respect to the analytical reference trajectory.

Table 6-1: Verification tests for the guidance system in the functional simulator.

Test	Description	Expected result	Prms	Result
1	Spin axis is aligned with the H-bar and there is no precession of the spin axis	Straight path towards the target along H-bar	$\beta = 0$ deg $\omega = 0$ deg/s	Fig 6-5 left
2	Spin axis under 30° angle with H-bar and there is no precession of the spin axis	Straight path towards the target under 30° angle with H-bar	$\beta = 30$ deg $\omega = 0$ deg/s	Fig 6-5 middle
3	Spin axis under 30° angle with H-bar and the spin axis precessed with 0.2 deg/s	Spiral towards the target ending in a circular in-plane motion	$\beta = 30$ deg $\omega = 0.2$ deg/s	Fig 6-5 right

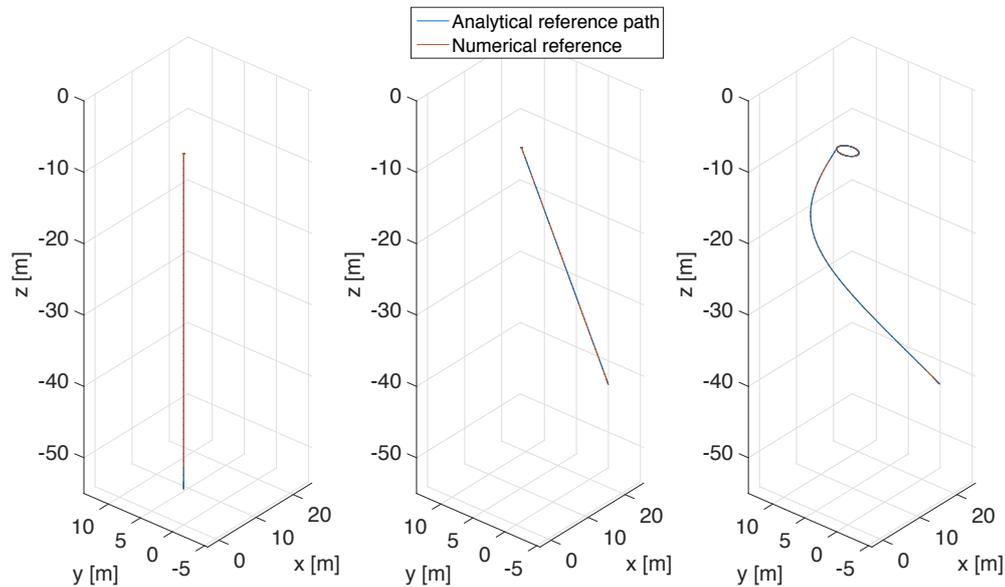


Figure 6-5: Guidance system verification results for test 1 (left), test 2 (middle), and test 3 (right). The error between the analytical reference and the numerical reference is insignificant for all test cases.

The implementation of the safety cone constraint is tested by applying several sets of initial states for the chaser satellite that lie outside of the safety cone. The optimization problem solver should then produce infeasible commands (due to the infeasibility of the problem), which was indeed the case for all tests.

6-4-3 Control System

The control system consists of an LQR that regulates the orbit control. To verify that it functions correctly a test case is designed in which the analytical reference trajectory is solely tracked by the LQR with an update rate of 1 Hz and the plant model runs at 10 Hz. The analytical reference and the actual path followed are given in Figure 6-6. The error between the analytical reference states and the actual states is given in Figure 6-7. It is clear from these figures that the LQR is implemented such that it is able to accurately track the analytical reference trajectory.

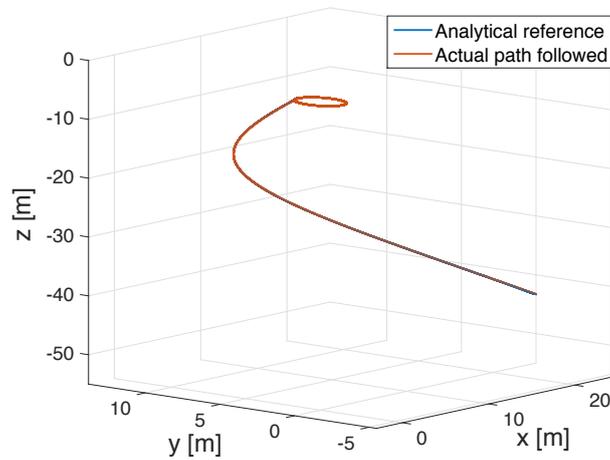


Figure 6-6: Verification of the LQR orbit controller.

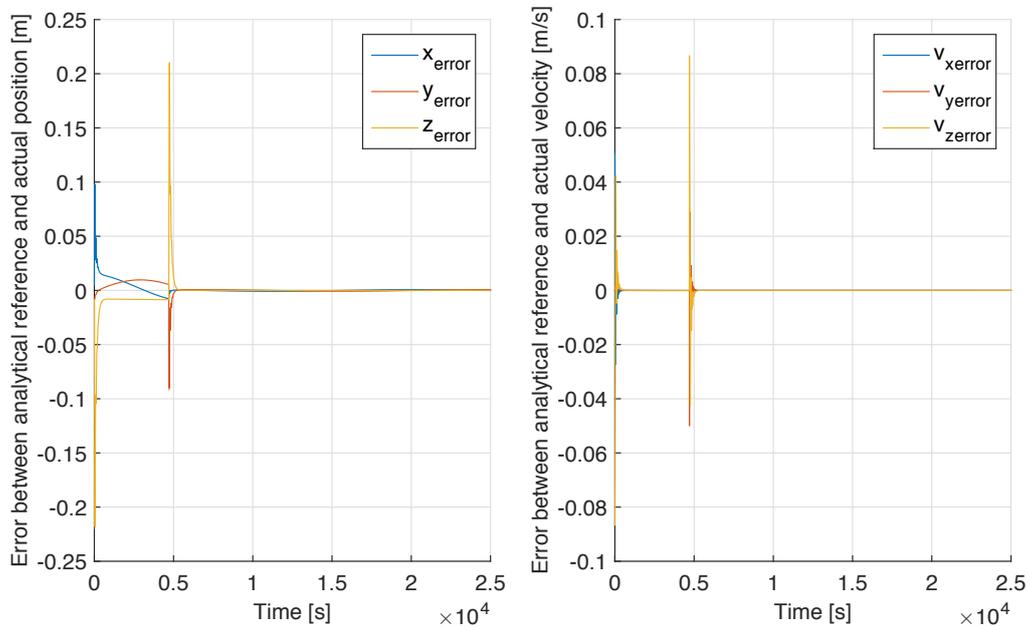


Figure 6-7: Error between the analytical reference position and actual path followed (left) and the error between the analytical reference velocities and actual velocities (right).

Chapter 7

TEAMS

There are many methods that exist for HIL testing of GNC algorithms. As discussed in Chapter 1 the method used in this research employs the TEAMS facility of DLR, the German aerospace center, in Bremen. This facility emulates the force and momentum-free dynamics of satellites in orbit, i.e., it reproduces the weightlessness and frictionless environment (Schlotterer and Theil, 2010). The experiments are performed using two free-floating, air-cushion vehicles that move over a highly smooth surface. A limitation is that the vehicle dynamics are reduced from orbit dynamics in the functional simulator to a basic double integrator on TEAMS, which neglects orbital perturbations. A large advantage, however, that is obtained by tests performed with TEAMS, is that it captures the interaction of the GNC system with actual sensors and actuators and the inherent real-time delays and errors induced by these systems.

The remainder of this chapter will present the TEAMS facility in more detail in Section 7-1. Adaptations had to be made to the functional simulator to enable testing with TEAMS. The required changes made to the reference scenario and the dynamics of the TEAMS vehicles is treated in Section 7-2.

7-1 Specifications and Previous Projects

The TEAMS facility emulates the force and torque-free dynamics of satellites in orbit. The surface over which the vehicles move consists of two granite tables with a total test area of 5 m by 4 m. The tables have been levelled with an accuracy of less than 20 μm from one edge to the other. All the experiments performed for this thesis research could only make use of one table, thus only an area of 2.5 m by 2 m was accessible.

Two TEAMS 3D air cushion vehicles represent the target and chaser spacecraft. These vehicles are able to emulate three degrees of freedom, two translational degrees and one rotational. Beneath each vehicle are three air cushion pads, which create a thin air film on which the vehicle can frictionlessly float. Air for the air bearings and thrusters is stored in four 300 bar air tanks. Pressure regulators regulate the air pressure down to 6-8 bar. Each

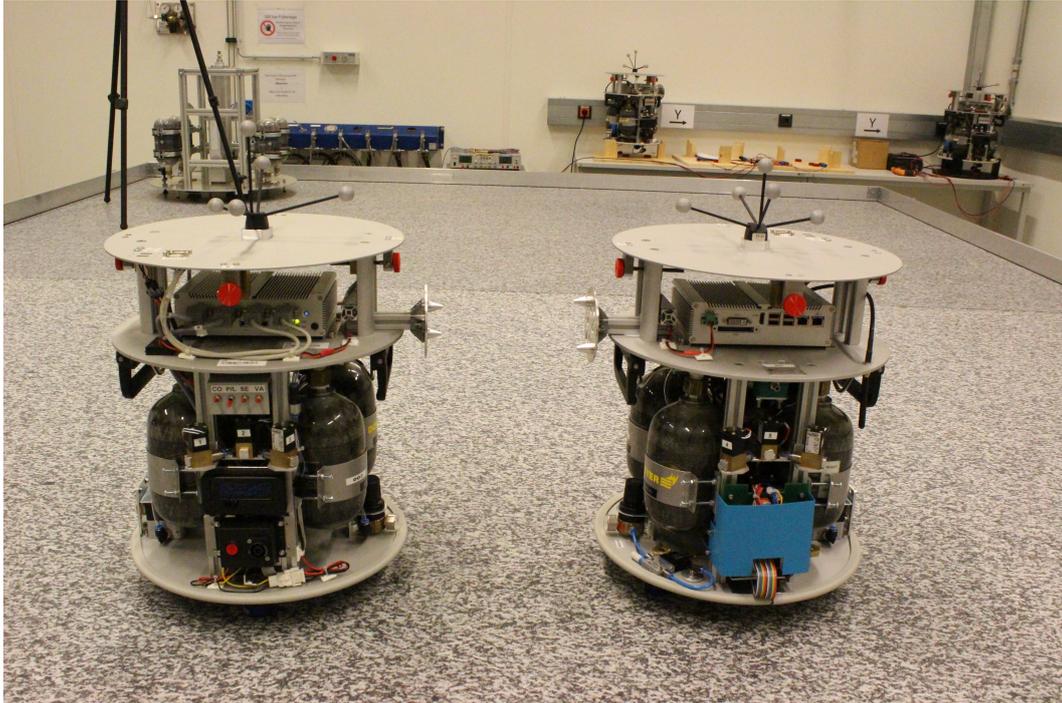


Figure 7-1: TEAMS facility with two TEAMS 3D air cushion vehicles.

vehicle is also equipped with a docking adapter, as shown in Figure 7-2. These adapters are mounted so that the two pins of the adapter of the chaser vehicle can slide into the two holes of the docking adapter of the target vehicle, resulting in a tight fit.

A DTrack infrared tracking system forms the sensor suite that performs measurements on the position and attitude of both vehicles. Five reflective balls are mounted on each vehicle, which are tracked by six infra-red cameras. The system computes the position and attitude of each vehicle by combining the images obtained with these cameras. The two vehicles have a different configuration of balls to enable the system to distinguish them from one another. The position and attitude information is sent over the local wireless network and is available to each onboard computer.

To control the position and attitude of the vehicles they are each equipped with proportional cold gas thrusters supported by 6-8 bar pressurized air. The maximum thrust of the chaser vehicle is about 33 mN.

The onboard computer is an embedded x86 Atom Z530 on a PC/104 stack and runs the QNX real-time operating system. Software can be uploaded and real-time data can be downloaded, displayed, or saved via a WLAN connection. The GNC algorithms developed for the TEAMS experiments are developed using Matlab/Simulink together with Simulink coder for automatic generation of C code.

TEAMS has mainly been used to test and validate GNC algorithms for formation flying and swarm operations. Schlotterer (2012) has used TEAMS to successfully validate the performance and robustness of a path-planning and collision avoidance method for satellite formations and swarms. The results obtained by Di Mauro et al. (2013), using TEAMS, show the effectiveness of a nonlinear controller technique for satellite proximity operations.

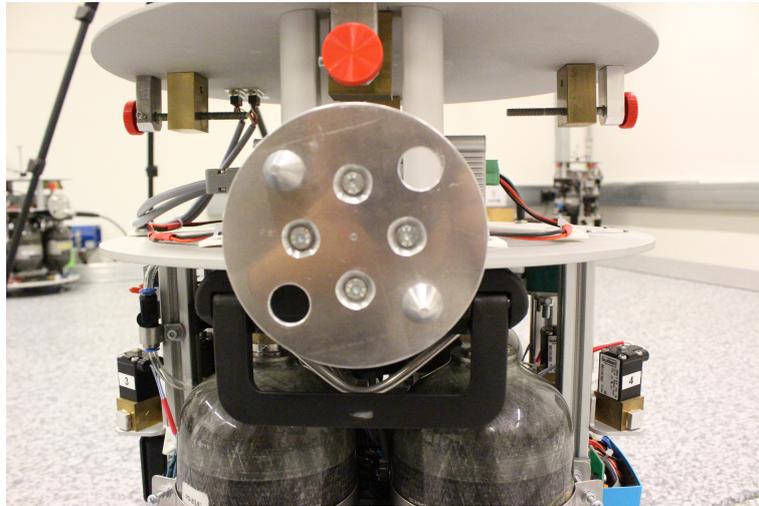


Figure 7-2: Docking adapter of a TEAMS 3D vehicle.

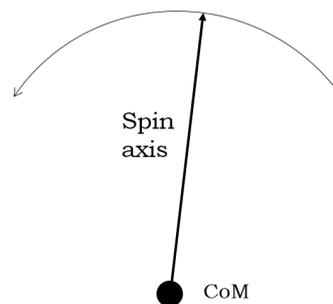


Figure 7-3: Motion of the spin axis of ENVISAT in the yz -plane (\bar{V} - \bar{H}).

7-2 Adaptations required for TEAMS

In Section 2-2 a reference scenario was constructed to rendezvous with ENVISAT. The corresponding dynamics were presented in Chapter 3. This section will present the changes that were made to represent this scenario on TEAMS and it will also discuss the dynamics of the TEAMS vehicles.

7-2-1 Reference Scenario

The spin axis of ENVISAT was assumed to rotate around its orbital angular momentum axis under an angle of 30 degrees. This 3D motion has to be converted to a 2D motion to represent the scenario on TEAMS. The docking adapters of the vehicles are mounted on the side of the vehicles. The movement of the adapter on the target vehicle will represent the movement of the spin axis of ENVISAT. The motion of the spin axis of ENVISAT in the yz -plane of the rotating Hill frame (\bar{V} - \bar{H}) is represented in Figure 7-3. As can be seen in this figure, the spin axis moves back and forth over an arc of 60 degrees.

The target vehicle of TEAMS will follow the motion presented in Figure 7-3. The velocity of this rotation is equal to the precession rate of the spin axis, which was assumed to be 0.2

deg/s. A full arc movement would thus be achieved in 300 s. The rendezvous and docking tests performed with TEAMS will be well below this time, therefore, the target vehicle of TEAMS will only rotate in one direction with a rotational velocity of 0.2 deg/s around its CoM.

The initial relative distance between the two vehicles was limited by the area of the tables. Only one table could be used, which has an area of 2.5 x 2 m. To guarantee that there was enough room for the chaser to move around the target, the initial relative distance was set to about 1.0 m.

7-2-2 Dynamics

The inertial reference frame of the test area used to represent the dynamics of the TEAMS vehicles has its origin at the center of the table. The y-axis is directed along the short side of the table and the x-axis along the long side. The dynamics of the vehicles are represented by:

$$\ddot{x} = \frac{T_x}{m}, \quad \ddot{y} = \frac{T_y}{m}, \quad \ddot{\theta} = \frac{T_z}{J} \quad (7-1)$$

where T_x and T_y are the thrust forces in x, and y direction, respectively, m is the mass of the vehicle, T_z is the torque exerted on the vehicle, and J is the mass moment of inertia of the vehicle.

The relative state that will be used in the guidance algorithms is obtained by simply subtracting the absolute state of the target vehicle from that of the chaser vehicle.

7-2-3 Optimization Problem Formulation

In Section 5-2 the optimization problem for the reference scenario was presented. To implement it on TEAMS this optimization problem has to be adapted to include the dynamics on TEAMS. The safety constraint is also removed for the TEAMS simulator, because of two reasons. Firstly, the CVXGEN program has a limited problem size capability, which is exceeded when the safety cone constraint is included. Secondly, CVXGEN can only handle quadratic problems and the safety cone constraint is formulated as a second-order cone constraint. The optimization problem used on TEAMS is thus formulated as:

$$\begin{aligned} & \text{minimize} \quad \sum_{k=0}^N \|Q\mathbf{T}\|_2 + \gamma \|\mathbf{s}\|_2 \\ & \text{subject to} \\ & \quad \mathbf{Z}(k) = F\mathbf{x}_0 + H\mathbf{T}(k) \\ & \quad \mathbf{Z}_{ref} = \mathbf{Z} + \mathbf{s} \\ & \quad \|\mathbf{T}_i\| \leq T_{max} \quad i = x, y \end{aligned} \quad (7-2)$$

The \mathbf{F} and \mathbf{H} matrix are constructed in the same manner as for the reference scenario (discussed in Section 5-2), the only difference is that for TEAMS they are based on the dynamics of the TEAMS vehicles.

TEAMS Simulator Development

To validate the designed guidance and control system further this thesis research uses the TEAMS facility, discussed in Chapter 7. The functional simulator, developed in Chapter 6, has to be replaced by a dedicated simulator for TEAMS, in which the developed guidance and control algorithms are implemented. An overview of the software architecture of the TEAMS simulator will be given in 8-1. The implementation of the designed guidance and control algorithms and the necessary adaptations and developments to achieve this are treated in Section 6-2. The verification tests that are performed to ensure that the TEAMS simulator, including the HIL elements, functions correctly will be presented in Section 8-3.

8-1 Software Architecture

The high-level architecture of the TEAMS Matlab/Simulink model of the chaser vehicle is depicted in Figure 8-1. The figure shows that the D-Track measurement system block provides information on the state of the chaser and the target, which is based on the measurements by the infra-red cameras. The navigation filter uses this noisy input data to produce a statistically optimal estimate of the state of the target/chaser vehicle. Based on this information the guidance system then computes a desired trajectory and the corresponding feed-forward commands, which are fed into the plant system via the control system, which incorporates the thruster model. The desired trajectory is then used by the control system, which compares it to the estimated state obtained by the navigation system, and computes the control commands to eliminate the error between the two desired and estimated state. As stated, the thruster system is embedded in the control system and the control outputs are thus the thruster commands, which are relayed to the onboard computer of the vehicle.

The high-level architecture of the TEAMS Matlab/Simulink model of the target vehicle is constructed in a similar manner as that of the chaser vehicle, the most important difference is that the guidance and control of the target is not based on the relative state but is merely based on a simple preprogrammed rotation. This represents the uncooperative nature of ENVISAT.

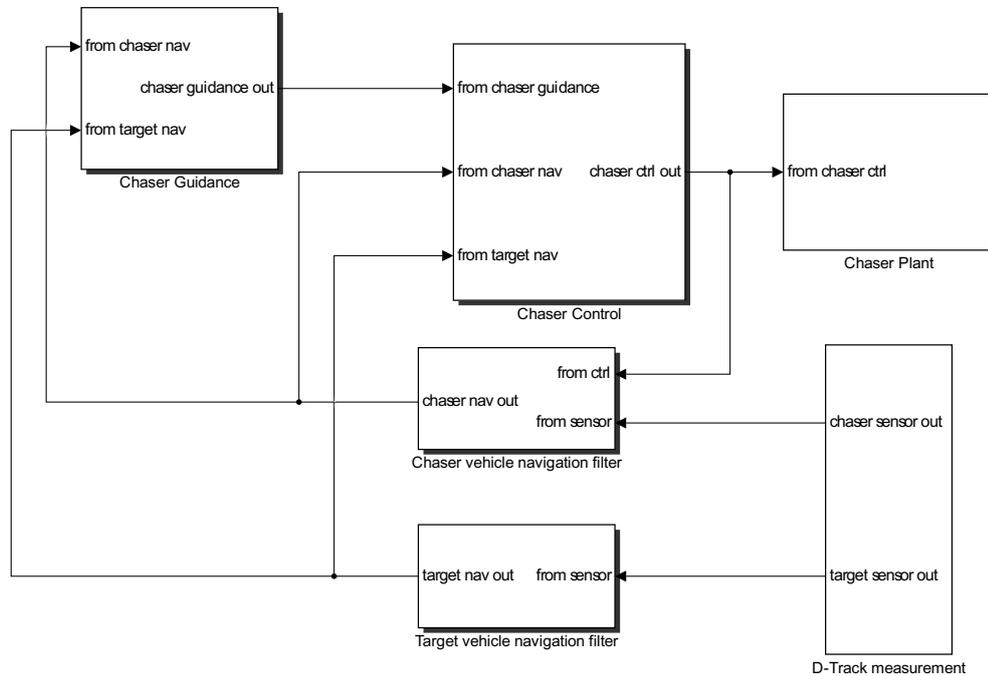


Figure 8-1: High-level architecture of the TEAMS Matlab/Simulink model including the sensor measurements, navigation filters, guidance, control, and plant model.

8-2 Guidance and Control System Development and Implementation

This section will treat the development and implementation of the guidance and control system and algorithms for the chaser vehicle. These topics will be covered for the target vehicle in the next section.

8-2-1 Guidance System

The guidance system on TEAMS consists of the same two main functions as for the functional simulator, as presented in Figure 6-2. The construction and implementation of these two functions are adapted for the TEAMS simulator to fit the redefined scenario as presented in Section 7-2 and the nature of the test environment. These adaptations are discussed in the remainder of this section.

Generation of Analytical Reference Trajectories

The analytical reference trajectory for the functional simulator is a predefined trajectory that is time-invariant, representing the final phase of the approach. This trajectory depends on

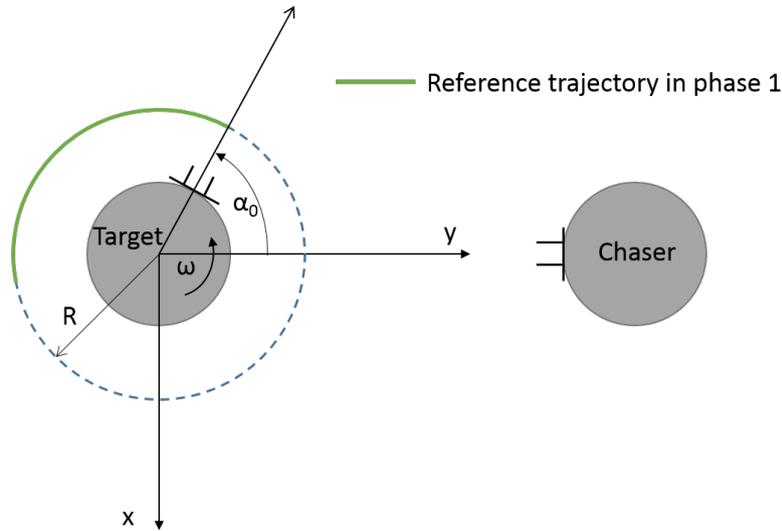


Figure 8-2: Reference trajectory of the chaser in phase 1 of the approach.

the translational state of the chaser satellite and has a size that corresponds to the planning horizon. This implementation assumes full knowledge of the current and future motion of the target satellite. One can imagine that for the free-floating vehicles on TEAMS, with a motion of the target vehicle that is not purely theoretical, this is not a realizable procedure. The analytical reference trajectory in the guidance function of the chaser vehicle is therefore based on real-time information on the state of the target vehicle.

Another difference lies in the fact that only translational dynamics were considered in the functional simulator. Recall that the TEAMS vehicles are both equipped with a docking adapter, before the final approach can commence these have to be aligned with each other. Therefore, rotational dynamics do need to be taken into account for the TEAMS simulator. The construction of the analytical reference trajectory is therefore separated into two phases:

- In *phase 1* the chaser moves towards a point at a specific radial distance from the target. The attitude will be altered to ensure alignment of the docking adapters before the second phase commences. The corresponding analytical reference trajectory is depicted in Figure 8-2.
- In *phase 2* the chaser moves in a straight line, in the target-centered, rotating reference frame, towards the target, with a constant velocity, until docking is achieved. The corresponding analytical reference trajectory is shown on the left-side of Figure 8-3. In the target-centered, non-rotating reference frame this will be a trajectory that spirals inward towards the target. The corresponding analytical reference trajectory is shown on the right-side of Figure 8-3.

The reference state trajectory in the first phase is then mathematically given by:

$$\mathbf{r}_{ref} = \begin{bmatrix} \mathbf{x}_{ref} \\ \mathbf{y}_{ref} \end{bmatrix}, \quad \dot{\mathbf{r}}_{ref} = \begin{bmatrix} -\omega \mathbf{y}_{ref} \\ \omega \mathbf{x}_{ref} \end{bmatrix} \quad (8-1)$$

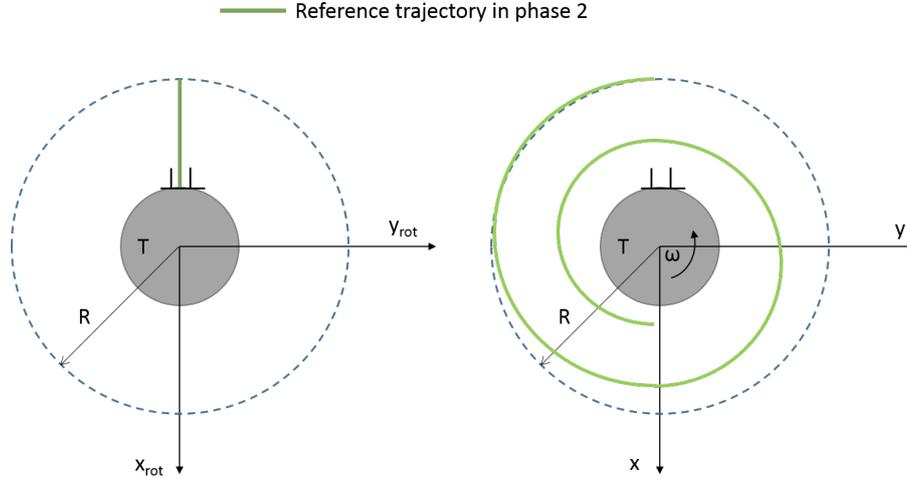


Figure 8-3: Reference trajectory of the chaser in phase 2 of the approach. Left: target-centered, rotating reference frame. Right: target-centered, non-rotating reference frame.

where ω is the rotational velocity. The angle α and the reference states in x and y direction, denoted by x_{ref} and y_{ref} , respectively, are given, in continuous time, by:

$$\begin{aligned}\alpha &= \alpha_0 + \omega t \\ x_{ref} &= -R \sin \alpha \\ y_{ref} &= R \cos \alpha\end{aligned}\tag{8-2}$$

The radial distance R and the angle α_0 representing the attitude are depicted in Figure 8-2. Phase 2 will commence if the the following two conditions are met:

$$\begin{aligned}(x_{ref_0} - x_{rel})^2 + (y_{ref_0} - y_{rel})^2 &\leq r_{ph2} \\ (\dot{x}_{ref_0} - \dot{x}_{rel})^2 + (\dot{y}_{ref_0} - \dot{y}_{rel})^2 &\leq \dot{r}_{ph2}\end{aligned}\tag{8-3}$$

where x_{ref_0} and y_{ref_0} are the coordinates of the starting point of the reference trajectory and \dot{x}_{ref_0} and \dot{y}_{ref_0} the velocities in x and y direction at this point. This means that at every guidance update step it is checked if 1) the chaser is within a radius of r_{ph2} m to the starting point of the analytical reference trajectory of that update step and 2) if the relative velocity of the chaser to the target is less than \dot{r}_{ph2} m/s.

In the rotating, target-centered reference frame the docking adapter is always aligned with the y-axis. The analytical reference trajectory in phase 2 is thus a straight line along this axis, denoted by $\mathbf{r}_{ref_{rot}}$. The conversion of the analytical reference trajectory in the rotating reference frame to the non-rotating frame is given by:

$$\begin{aligned}
\mathbf{r}_{refinert} &= \begin{bmatrix} x_{refinert} \\ y_{refinert} \end{bmatrix} \\
\mathbf{r}_{refinert} &= \mathbf{T}_{rn} \mathbf{r}_{refrot} \\
\dot{\mathbf{r}}_{refinert} &= \mathbf{T}_{rn} \dot{\mathbf{r}}_{refrot} + \begin{bmatrix} -\omega y_{refinert} \\ \omega x_{refinert} \end{bmatrix}
\end{aligned} \tag{8-4}$$

where the transformation matrix \mathbf{T}_{rn} is given by:

$$\mathbf{T}_{rn} = \begin{bmatrix} \cos \alpha & -\sin \alpha \\ \sin \alpha & \cos \alpha \end{bmatrix} \tag{8-5}$$

Implementation of the Convex Optimizer

The convex optimizer in the functional simulator employed the modeling language CVX to formulate the optimization problem, which had an embedded solver available. A large disadvantage of using a modeling language like CVX is that it decreases the performance of the optimization. Next to that, CVX cannot be implemented as an S-function in Simulink, which is necessary to run the algorithms on TEAMS. For these reasons, the real-time application on TEAMS makes use of CVXGEN. CVXGEN is a software tool that takes a high level description of the convex optimization problem, and automatically generates custom C code that compiles into a reliable, high speed solver for the defined problem Mattingley and Boyd (2012). CVXGEN generates simple code that is especially suitable for embedding in real-time applications.

The convex optimizer on TEAMS works as follows: the analytical reference trajectory in the non-rotating, target-centered reference frame is fed into the CVXGEN solver, together with the estimated relative state, which then solves the optimization problem in real-time and produces a feasible trajectory, together with the feed-forward commands to achieve this trajectory.

The downside of CVXGEN is that it has a limited problem size capability. To comply with this limitation the number of discretization steps in the optimization problem had to be decreased to $N = 15$. The CVXGEN solver was implemented as an S-function in Matlab/Simulink to enable use on TEAMS. To create the S-function the S-function builder in Simulink was used.

Guidance for the Attitude

The attitude guidance of the chaser is based on the attitude and rotational velocity of the target vehicle. The desired attitude and rotational velocity of the chaser vehicle is given by:

$$\begin{aligned}
\alpha_{ch} &= \alpha + \pi \\
\omega_{ch} &= \omega
\end{aligned} \tag{8-6}$$

where the information on the attitude and angular velocity of the target are obtained from sensor measurements of the target vehicle.

8-2-2 Control System

The control block of the TEAMS simulink model incorporates both the orbit control and the attitude control of the chaser vehicle. The basis of both the path tracking (orbit control) and attitude control is an LQR controller, which are designed specifically for the vehicle (private communication F. de Bruijn, 2014).

The rotational movement of the target is not perfectly smooth, as will be discussed further in Section 9-4. To improve the smoothness an integral term in the LQR control was added for both the chaser and the target vehicle. Let the rotational state space system of the rotation of the vehicles be given by:

$$\begin{bmatrix} \dot{\theta} \\ \ddot{\theta} \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \theta \\ \dot{\theta} \end{bmatrix} + \begin{bmatrix} 0 \\ \frac{1}{J} \end{bmatrix} \mathbf{T} \quad (8-7)$$

where J is the moment of inertia. To include the integral term an extra state is included in the rotational state-space system which is defined as:

$$\frac{d\Theta}{dt} = \theta \quad (8-8)$$

resulting in the state-space system in Equation (8-9).

$$\begin{bmatrix} \dot{\Theta} \\ \dot{\theta} \\ \ddot{\theta} \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} \Theta \\ \theta \\ \dot{\theta} \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ \frac{1}{J} \end{bmatrix} \mathbf{T} \quad (8-9)$$

8-2-3 Target Guidance and Control System

The TEAMS Matlab/Simulink model for the target vehicle has the same structure as the one for the chaser vehicle. The main difference lies in the guidance system, which is, evidently, a lot simpler for the target vehicle.

The guidance system uses predefined analytical reference states, in which the target has a fixed position and a constant rotational velocity of 0.2 deg/s. The control of the target vehicle, which is also an LQR controller, then produces the force and torque commands to achieve this trajectory as accurately as possible.

8-3 Verification

To verify that the guidance and control algorithms designed for TEAMS worked correctly, use was made of a pre-designed, dedicated functional simulator, that incorporates the same Simulink architecture used on TEAMS. Evidently, the major difference is that this model makes use of a plant model instead of the actual TEAMS vehicles. This functional simulator was used to implement the developed guidance and control algorithms and adapt them to the TEAMS format.

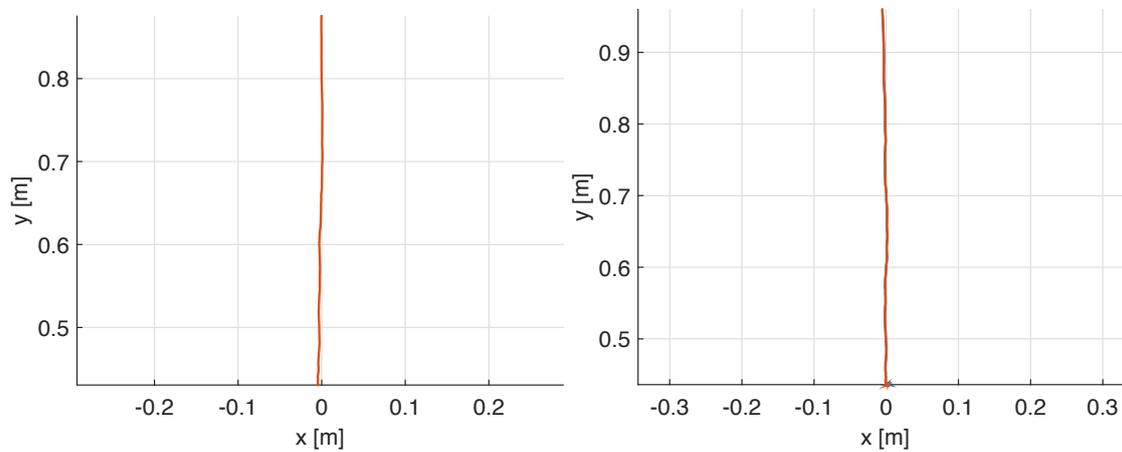


Figure 8-4: Verification tests for the TEAMS simulator for a non-rotating target. On the left test 1 and on the right test 2, which differ somewhat in the initial position of the chaser vehicle.

The most important verification, however, was done for the TEAMS simulator including all the hardware components. This includes the developed guidance and control algorithms, but also the navigation, sensor models and the data transmission. This part of the process was the most time-consuming in the development and mainly consisted of performing a test, inspecting the process visually and inspecting the data obtained during the tests. Then making changes to the model based on the presented deviations or errors from the conceptual design. This was done in cooperation with TEAMS experts. Examples include, adapting the thruster direction matrix of the chaser vehicle to reduce thruster errors and experimenting with the maximum data transmission capabilities of the system to decide upon the sampling rates.

To enable a better understanding of the causes of other errors tests were first performed with a non-rotating target. In this case the chaser vehicle should align the docking adapters and then just move in a straight line towards the target. This scenario makes for a good test case because deviations from the expected behaviour (a straight line towards the target) are readily inspected visually and therefore the causes of errors are deducted more easily than for the rotating target scenario. During the tests performed for this test case errors were eliminated from the sensor model and the navigation filter of the target vehicle was tuned to ensure a fixed position of the target.

The influence of the weighing factor γ on the performance of the system was also analysed. Recall, this parameter determines how much the solution obtained by the guidance system is allowed to deviate from the analytical reference trajectory. One can easily understand that if the holes of the docking adapter are about 1.5 cm it is important that this parameter is high enough to ensure a successful docking operation. However, this parameter should be low enough to ensure that the cost function puts enough emphasis on minimizing the propellant

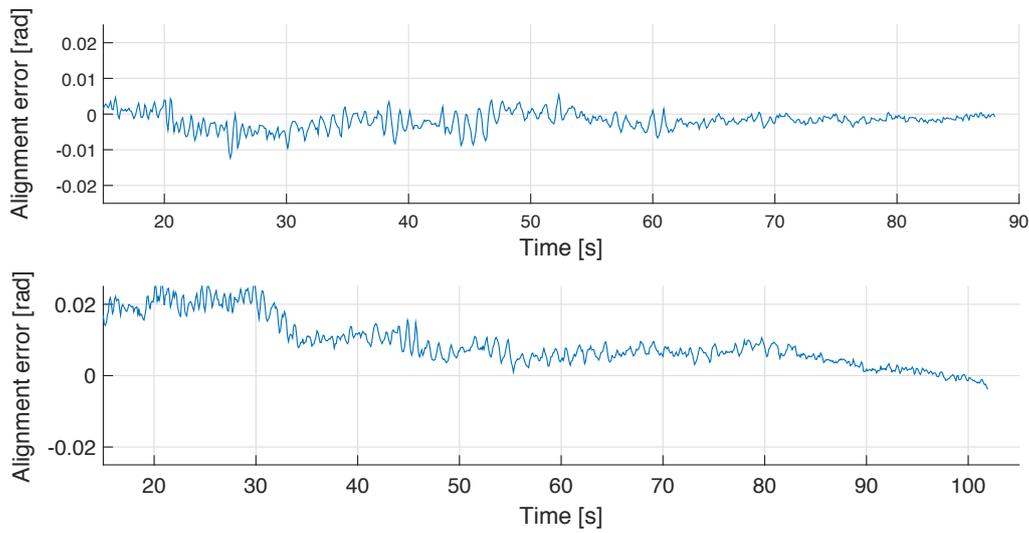


Figure 8-5: Verification of the TEAMS simulator for a non-rotating target. The alignment error over time for test 1 (top) and test 2 (bottom).

consumption, to obtain a propellant-efficient approach. It was also shown that increasing the weight too much resulted in unsuccessful docking operations, indicating that numerical reference trajectory was in those cases not realizable in the presence of errors. The results of this analysis also showed that successful docking operations were achieved when the penalty on the position error was about 1.5 times higher than the penalty on the velocity error.

After a good result was obtained the test case was repeated multiple times to ensure consistent behaviour of the system. For two of the performed tests the actual path followed by the chaser vehicle is shown in Figure 8-4. The initial conditions of both tests were the same, except for a difference in the initial position of the chaser of about 10 cm in the y-direction. It is clear from these figures that the chaser follows a path that is approximately a straight line. Figure 8-5 shows the corresponding error in alignment of the docking adapters for both tests. It is clear from this figure that during the final phase of the approach the alignment error is sufficiently small to ensure a successful docking.

Test Campaigns and Results

To assess the performance of the guidance and control strategies, presented in Section 4-5, a test campaign is designed for the functional simulator, which will be presented in Section 9-1. To validate these results further a test campaign is also constructed for the TEAMS simulator, which will be discussed in Section 9-3. The obtained results for the functional simulations are presented in Section 9-2 and for the tests performed with TEAMS in Section 9-4.

9-1 Functional Simulations Test Campaign

Two of the guidance and control strategies presented in Section 4-5 are evaluated using the functional simulator developed in Chapter 6. One of the strategies, the NUM-OL + LQR method, could unfortunately not be tested, because the developed software is not efficient enough. This results from the fact that for this strategy the planning horizon should include the entire reference scenario. This becomes an issue because the Euler method is used in the discretization of the optimization problem, which means that the size of the time steps is limited due to the error induced by the method. With a Δt_1 of 1 s and a Δt_2 of 10 s, this means that the number of discretization steps N should be at least equal to 800. CVX requires a very large amount of computing time for the designed optimization problem, with these parameters. The computational efficiency of the program should be increased significantly to reduce the corresponding computation time. The implementation of the convex optimizer in CVXGEN is unfortunately not an option due to the large problem size. The most self-evident solution would be to write a program that does not use CVX. This would be very time-consuming and is therefore not considered in this thesis research. The choice is therefore made to exclude this strategy from the test campaign. The test campaign parameters to assess the remaining two methods (MPC+LQR and MPC) are presented in Table 9-6.

The values of the sampling rates are based on the parameters used in the e.deorbit CDF study where a position controller of 5 Hz is used. (ESA, 2012).

Table 9-1: Functional simulator test campaign parameters.

Parameter	MPC+LQR	MPC
Guidance sampling rate [Hz]	1	5
Control sampling rate [Hz]	5	-

These strategies are assessed on the following performance parameters:

- Amount of ΔV required to complete the rendezvous operation
- State error, i.e., the error between the analytical reference states and the actual states
- Time to complete the rendezvous operation

Recall, from Section 2-2 that at the stage where the chaser is at 3 m relative distance from the target and has obtained a stable circular in-plane motion with respect to the target, the state error is required to be in the order of a few centimetres or less for the position and no larger than 1 cm/s for the velocity.

9-1-1 Optimization Problem Size Reduction

It would be desirable to have an optimization problem size that improves the computational efficiency while still being sufficiently accurate. The starting point of the functional simulator design assumes a problem with $N = 50$, $N_s = 10$, $\Delta t_1 = 1$ s and $\Delta t_2 = 10$ s, this corresponds to a planning horizon of 410 s. To assess if this can be reduced further without losing significant performance, a short analysis on the optimization problem size was performed. In this analysis only the MPC method is considered, with a constant Δt_1 of 1 s (thus a guidance update rate of 1 Hz) and a Δt_2 of 10 s. These values are deemed to be sufficient to indicate the difference in performance for different problem sizes. The results for this analysis are shown in Table 9-2. The minimum and maximum state error relate to both the position and the velocity entries of the state. The state error is assessed at the final part of the rendezvous, which is at the point where a stable circular in-plane motion, at a distance of 3 m from the target, is achieved.

The results show that the planning horizon should be sufficiently large to include enough information on the predicted future states to achieve the required accuracy, i.e., the state error should be sufficiently small to meet the requirements. Once this threshold is reached ($N = 15$, $N_s = 3$) the performance does not increase any further for larger problem sizes. The planning horizon, however is relatively small for these settings. However, the analysis is performed without any measurement noise taken into account, i.e., it assumed perfect knowledge on the future states of the target. To compensate for this assumption settings are chosen that result in a larger planning horizon, which should enable handling uncertainties better. The analyses performed in the remainder of this section will therefore use a problem size of $N = 25$ with $N_s = 10$, which corresponds to a planning horizon of 160 s.

Table 9-2: Results on the optimization problem size analysis.

Problem size [-]	ΔV [m/s]	Min and max state error [m] (order of magnitude)
$N = 5, N_s = 3$	0.7765	$10^0 - 10^1$
$N = 15, N_s = 10$	0.5013	$10^{-5} - 10^{-1}$
$N = 15, N_s = 3$	0.5002	$10^{-5} - 10^{-1}$
$N = 25, N_s = 10$	0.5022	$10^{-5} - 10^{-1}$
$N = 25, N_s = 3$	0.5013	$10^{-5} - 10^{-1}$
$N = 50, N_s = 10$	0.5026	$10^{-5} - 10^{-1}$

9-1-2 General Simulation Parameters

For every test case the chaser vehicle starts at 50 m relative distance from the target on the spin axis with zero relative velocity. The general simulation parameters that are kept constant for all test cases are given in Table 9-3. The spacecraft specific parameters, for the target and chaser spacecraft, used in the subsequent analyses can be found in Table 9-4. These values are based on information provided by Deloo (2015) and ESA (2012).

9-2 Functional Simulations Results

This section will present the results obtained with the simulations performed using the functional simulator. It will discuss the results of the test campaign presented in the previous section. Furthermore, it will also present the results of an analysis on the influence of the optimization parameters and settings on the performance of the MPC method.

9-2-1 Performance of the Guidance and Control Strategies

The performance of the guidance and control strategies is based on how well the error with respect to the analytical reference and the propellant use are minimized. The reference scenario is simulated for both the MPC+LQR and MPC method. The results on the analytical reference path and the actual path are shown for both methods in Figure 9-1. The figure also

Table 9-3: General simulation parameters.

Description	Parameter	Value
State propagation rate	f_{plant} [Hz]	10
Discretization time step 2	Δt_2 [s]	10
Nr. of steps in the discretization	N [-]	25
Nr. of discretization time steps of size Δt_1	N_s [-]	10
Weight on slack variable in opt	γ [-]	1000
Spin axis angle with respect to H-bar	β [deg]	30
Spin axis precession rate	ω [deg/s]	0.2
Desired approach velocity along spin axis	$v_{approach}$ [m/s]	0.10
Maximum thrust in x, y and z direction	T_{max} [N]	44

Table 9-4: Target and chaser specific parameters used in the functional simulations.

Description	Parameter	Value	
		Target	Chaser
Mass	m [kg]	7828	1444
Frontal area	A [m ²]	38.14	4.5
Drag coefficient	C_D [-]	2.2	2.2
Reflectance factor	q [-]	0.3	0.3

shows the projections of the actual path on the three planes. This figure shows that both methods succeed in following the analytical reference trajectory. To assess the performance the error between the analytical reference states and the actual states is also computed. These results are shown in Figure 9-2.

The peaks for the velocity errors in Figure 9-2 correspond to the point in time when the relative distance of 3 m is reached and the inwards spiralling motion transforms into a circular in-plane motion. These peaks are relatively high because of the abrupt change in the analytical reference velocity from 0.10 m/s to 0 m/s at this point. After the circular in-plane motion stabilizes both methods are able to keep the state error sufficiently small. At this stage the maximum position error is in the order of 10^{-6} m for the MPC method and 10^{-5} m for the MPC+LQR method. The maximum velocity error at this stage is for both methods in the order of 10^{-6} m/s.

The ΔV spent up to the point that a stable motion is achieved at 3 m relative distance from the target and the time that it takes to achieve this is taken as the comparison standard for the analyses in the remainder of the chapter. This point will also be the reference point for the time to complete the rendezvous operation. This stable motion is represented by the circular paths in Figure 9-1. The remainder of this circular motion is less interesting in the ΔV computations, because it is assumed that from this point on the docking mechanism can be deployed. For that stage it is most likely that different guidance and control modes will be implemented.

The ΔV applied to achieve this stable final motion is equal to 0.5888 m/s for the MPC+LQR method. The feedforward and feedback commands contribute 0.5276 m/s, and 0.0612 m/s to this total, respectively. Clearly, the feed-forward commands are the main driver for the applied ΔV . The ΔV required for the MPC method is 0.4865 m/s. The time to achieve this is equal to about 485 s for the MPC method and about 490 s for the MPC+LQR method. The overall results thus show that the addition of an LQR controller results in a similar state error and time to complete the rendezvous operations but it leads to an increase in ΔV of 21%.

To assess where this difference comes from, the control force profiles for the MPC+LQR method, corresponding to the path followed in Figure 9-1 up to the point when the stable motion is achieved, are depicted in Figure 9-3. From the zoomed view on the right side of this figure it is clear that the feedback commands counteract the feed-forward commands at times, which causes the higher amount of required ΔV for this method. Further analysis is necessary to deduce the cause of these counteracting profiles.

For the MPC method, the control profiles corresponding to the path followed in Figure 9-1 up to the point when the stable motion is achieved, are shown in Figure 9-4. The right

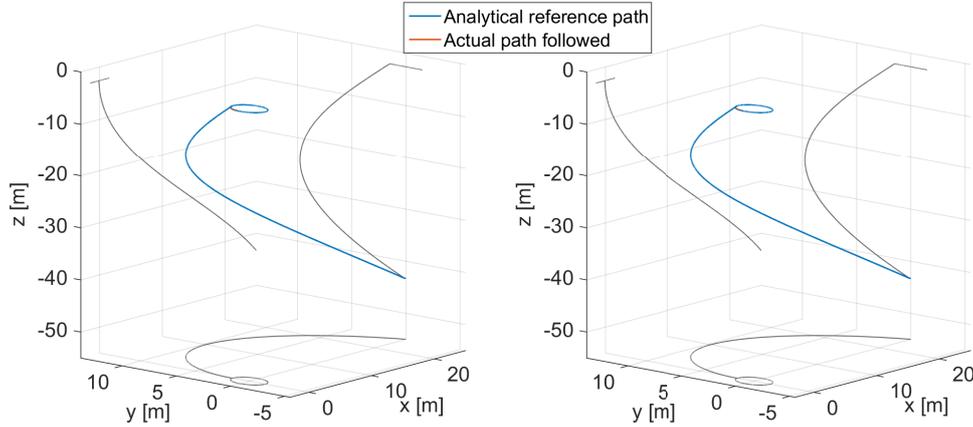


Figure 9-1: Analytical reference and actual path followed for MPC (left) and MPC+LQR (right).

side of this figure also shows a zoomed view. When this is compared to the feedforward commands of the MPC+LQR, it is clear that the feedforward control behaviour is similar for both methods. The control profiles for both methods show oscillating behaviour from the point where the planning horizon includes the final circular motion, this could be caused by the used discretization method and size of time step Δt_2 but to draw definite conclusions this behaviour should be analysed further in-depth. If it turns out these oscillations cannot be diminished, then there is a need for throttleable thrusters.

Furthermore, both control profiles show a large control input at the beginning of the operation. This is the result from the fact that the spin axis has an initial relative velocity with respect to the target's CoM whereas the chaser does not. Therefore at the start of the operation the chaser immediately lags behind the spin axis, which has to be corrected for.

To provide an indication of the propellant use related to these ΔV values, Equation (9-1) is used.

$$m_{prop} = m_{dry} \left(e^{\left(\frac{\Delta V}{I_{sp} g_0} \right)} - 1 \right) \quad (9-1)$$

Based on Deloo (2015), an I_{sp} of 300 s is assumed and $g_0 = 9.81 \text{ m/s}^2$. The propellant mass for the MPC and MPC+LQR method are then equal to 0.24 kg and 0.29 kg, respectively.

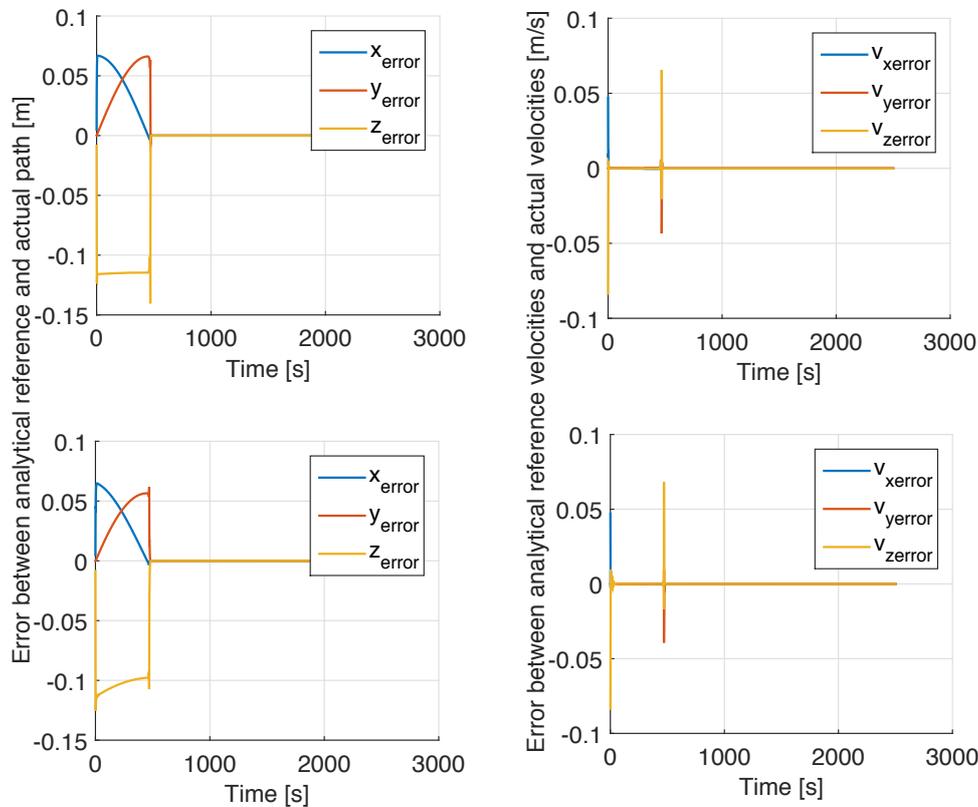


Figure 9-2: Error between the analytical reference position and actual path followed (left) and the error between the analytical reference velocities and actual velocities (right) for the MPC method (top) and the MPC+LQR method (bottom).

9-2-2 Optimization Analysis of the MPC Method

Trade-off between Minimum State Error and Minimum Thrust

As stated before, the performance of the MPC method is based on how well the error with respect to the analytical reference, i.e., the state error, and the propellant use are minimized. In Section 5-2, it was shown that the minimization of these two parameters is a trade-off which is determined by the magnitude of the weight put on the slack variable in the convex minimization. Recall, that there is already a fixed weight on the thrust related to the differently sized time steps, which was discussed in Section 5-2-1. In this section the results of a preliminary analysis will be presented that assesses the influence of the weight parameter on the slack variable to assess how much the required mission propellant can be decreased while staying below the required state error. The preceding analyses used a value for γ of 1000. This weight is now varied to assess the influence on the performance parameters for $\gamma = 500$, $\gamma = 100$, $\gamma = 10$, and $\gamma = 1$.

It is apparent, that applying a lower weight on the slack variable in the optimization will result in larger deviations from the reference trajectory. Recall, that the implementation of

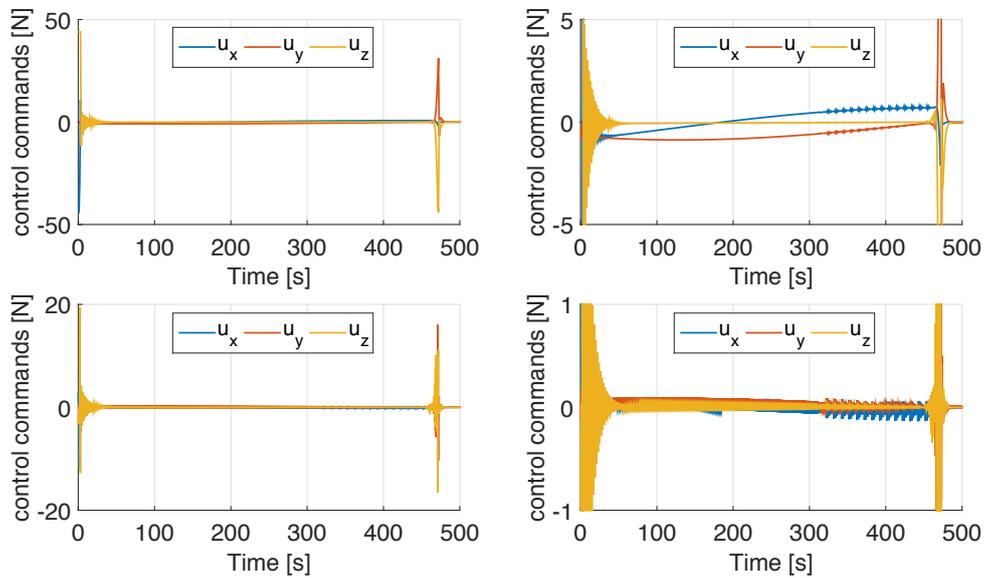


Figure 9-3: Control force profiles for the MPC+LQR method. The feed-forward commands are shown at the top left and a zoomed in view on the top right. The feedback commands are shown at the bottom left and a zoomed in view on the bottom right.

the safety-cone constraint in the convex optimizer will ensure that the resulting deviations are not allowed outside the safety cone.

To assess the influence of the weight on the performance, first the analytical reference and the resulting actual path followed are shown in Figure 9-5. Figure 9-6 shows the state error for these three test cases. It is clear that for $\gamma=1$ the operation is unsuccessful. The corresponding ΔV values and times to complete the operation for the other values of γ are given in Table 9-5.

For $\gamma=10$ the state error is relatively large throughout parts of the approach but the safety cone constraint will always ensure that the chaser does not collide with ENVISAT and it does results in a significant decrease in ΔV even though the time to complete the operation

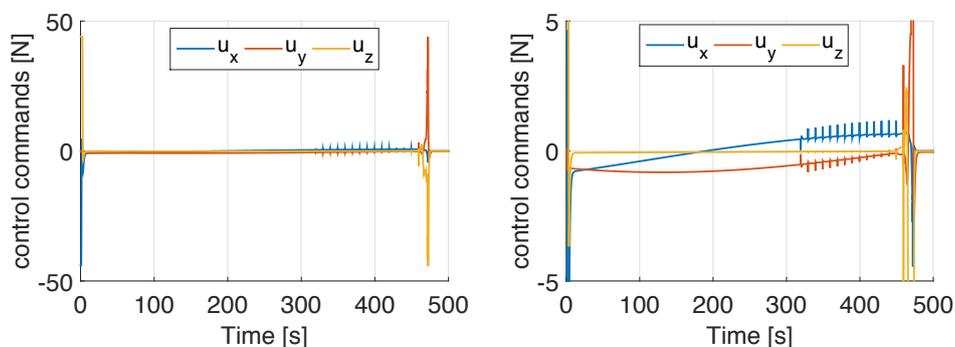


Figure 9-4: Control force profiles for the MPC method, which only consist of feed-forward commands. A zoomed-in view is depicted on the right-side.

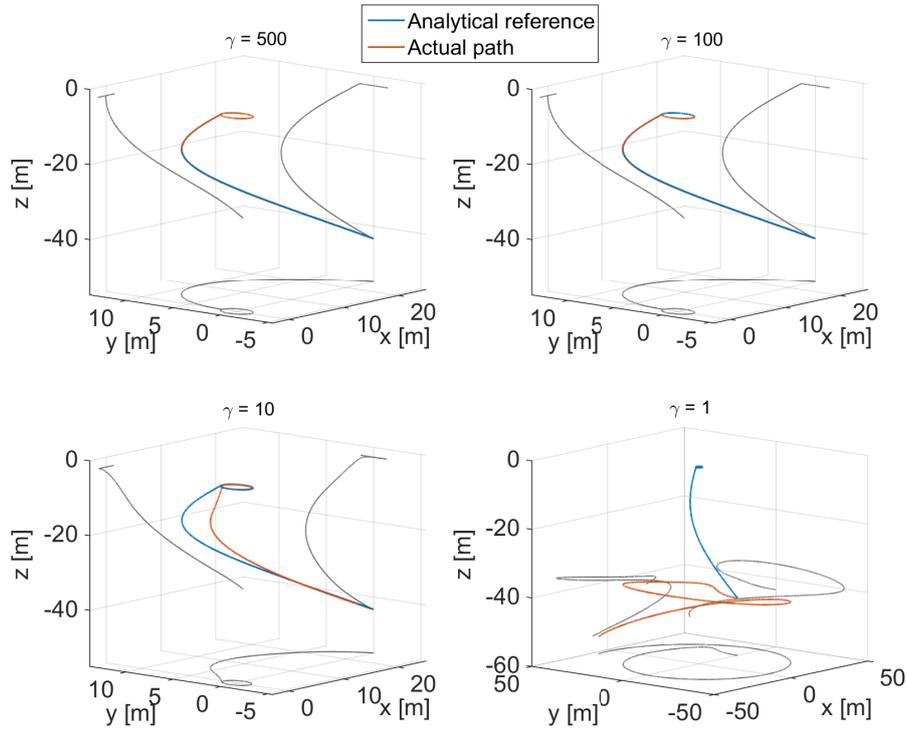


Figure 9-5: Analytical reference states and actual states for different values of γ .

is substantial. However, at the final circular motion the position error does not meet the requirement. The maximum order of magnitude of the position and velocity errors between the analytical reference and actual states, for all these cases, are given in Table 9-5. The error for $\gamma=1$ is not given there because this simulation never reaches this final stage. From this table it can be concluded that for $\gamma = 100$, $\gamma = 500$, and $\gamma = 1000$ the maximum state error fulfils the requirements and the time to complete the operation is comparable, while the ΔV required decreases moderately.

Thrust Minimization using the 1-Norm

To reduce the ΔV required to perform the rendezvous further it is hypothesized that this could be achieved by replacing the 2-norm in the minimization of the thrust by the 1-norm minimization. The 1-norm is namely directly a measure of the propellant use for axially aligned thrusters (Ross, 2006). To clarify the difference between the computation of both norms, they will be mathematically described. Let $\mathbf{T} = [T_1 \ T_2 \ \dots \ T_n]^T$, then the 1-norm minimization is given by:

$$\|\mathbf{T}\|_1 = \sum_{k=1}^n |T_k| \quad (9-2)$$

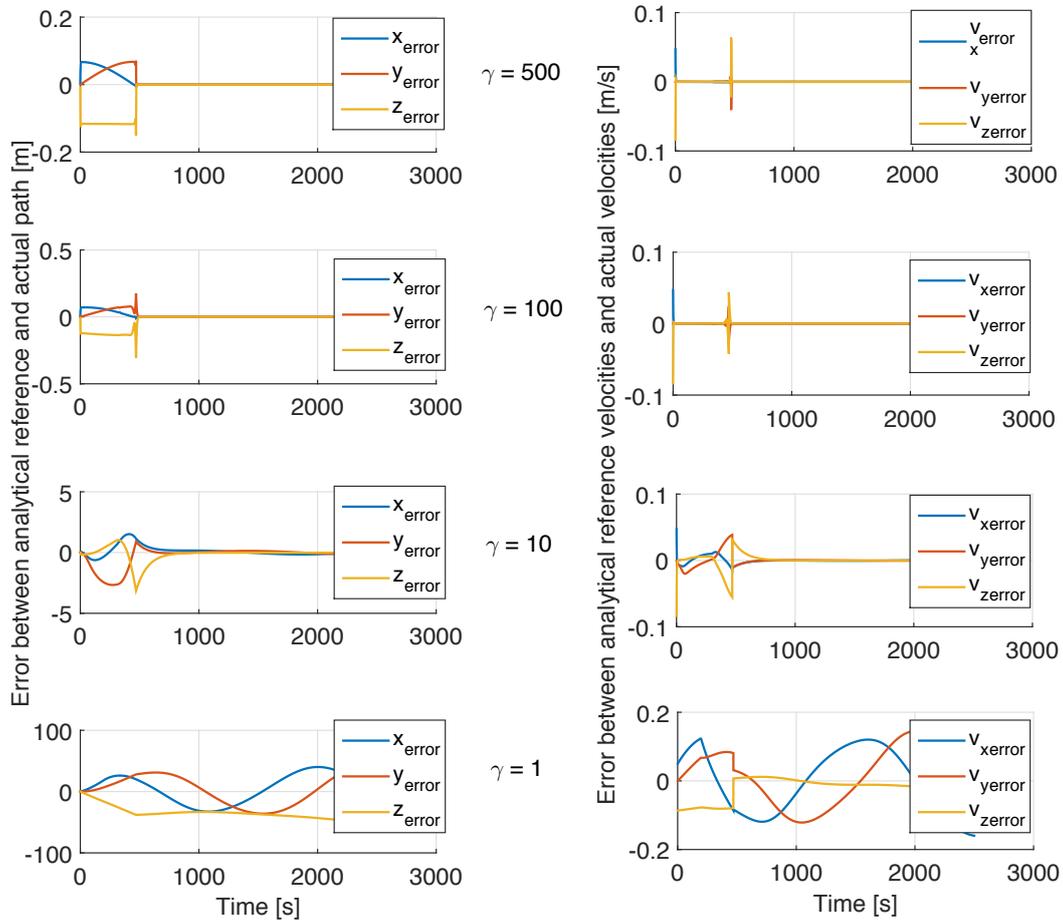


Figure 9-6: Error between the analytical reference position and actual path followed (left) and the error between the analytical reference velocities and actual velocities (right) for different values of γ .

whereas the 2-norm is denoted by:

$$\|T\|_2 = \sqrt{\sum_{k=1}^n T_k^2} \tag{9-3}$$

To assess the difference in performance the preceding analysis is repeated with the 1-norm in the thrust minimization. The test case with $\gamma = 1$ is excluded in this analysis, because of the infeasible results that were previously obtained.

To assess the performance, first the analytical reference and the actual path followed for this 1-norm minimization are shown for $\gamma = 1000$, $\gamma = 500$ and $\gamma = 100$, $\gamma = 10$ in Figure 9-5. The corresponding state errors for these cases are presented in Figure 9-8. The corresponding ΔV values and times to complete the rendezvous operations are given in Table 9-5.

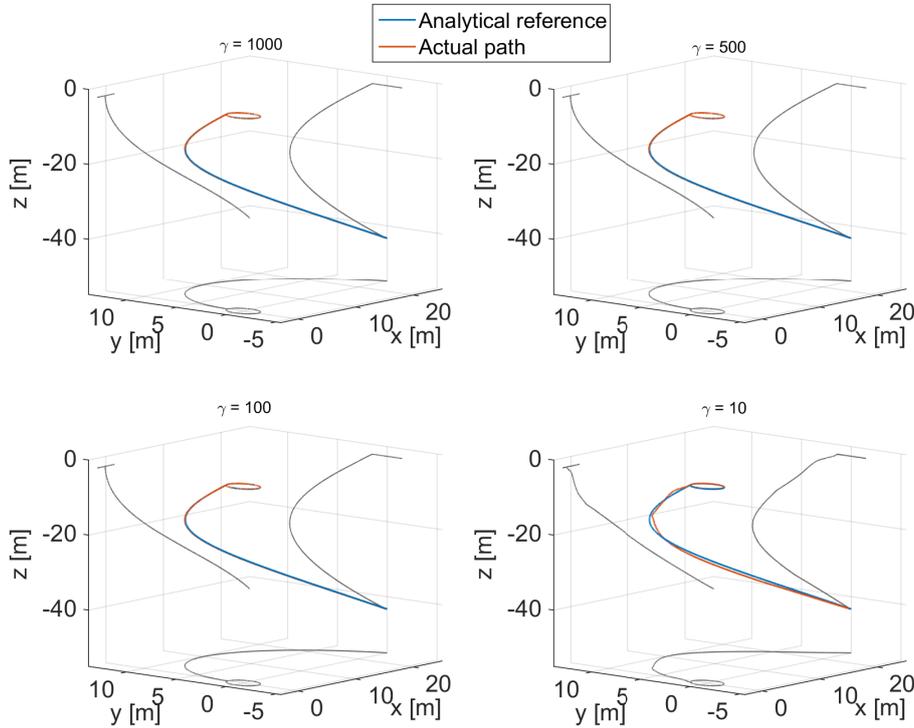


Figure 9-7: Analytical reference states and actual states for different values of γ . These values are obtained using a 1-norm thrust minimization.

The resulting state errors for $\gamma = 1000$ and $\gamma = 500$ at the final part of the approach obtained with the 1-norm minimization are similar to the results obtained with the 2-norm minimization. The 1-norm minimization does, however, lead to a decrease of about 2% in ΔV for these values of γ .

For $\gamma = 100$ and $\gamma = 10$, the resulting state errors at the final part of the approach obtained with the 1-norm minimization are also similar to the results obtained with the 2-norm minimization. The required ΔV , however, increases by 2% and 17%, respectively. The cause of this difference can be clarified by the paths and errors shown in Figures 9-7 and 9-8. The 1-norm minimization results in a path that is significantly less smooth, which is especially clear for the results obtained for $\gamma = 10$. The sharp angles in the followed path indicate that at these points higher thrust levels are applied. This is also shown in Figure 9-9, where the control profiles for $\gamma = 10$ are displayed for the 1-norm minimization and the 2-norm minimization.

Combined Approach Analysis

The previous sections showed that the lowest ΔV value was obtained for the simulation that applied the 2-norm for the thrust and used a weight of $\gamma = 10$. The resulting state error in the final stage was, however, not within the requirements. The lowest ΔV value that was obtained for a set of simulation parameters that did meet the requirements was also with a

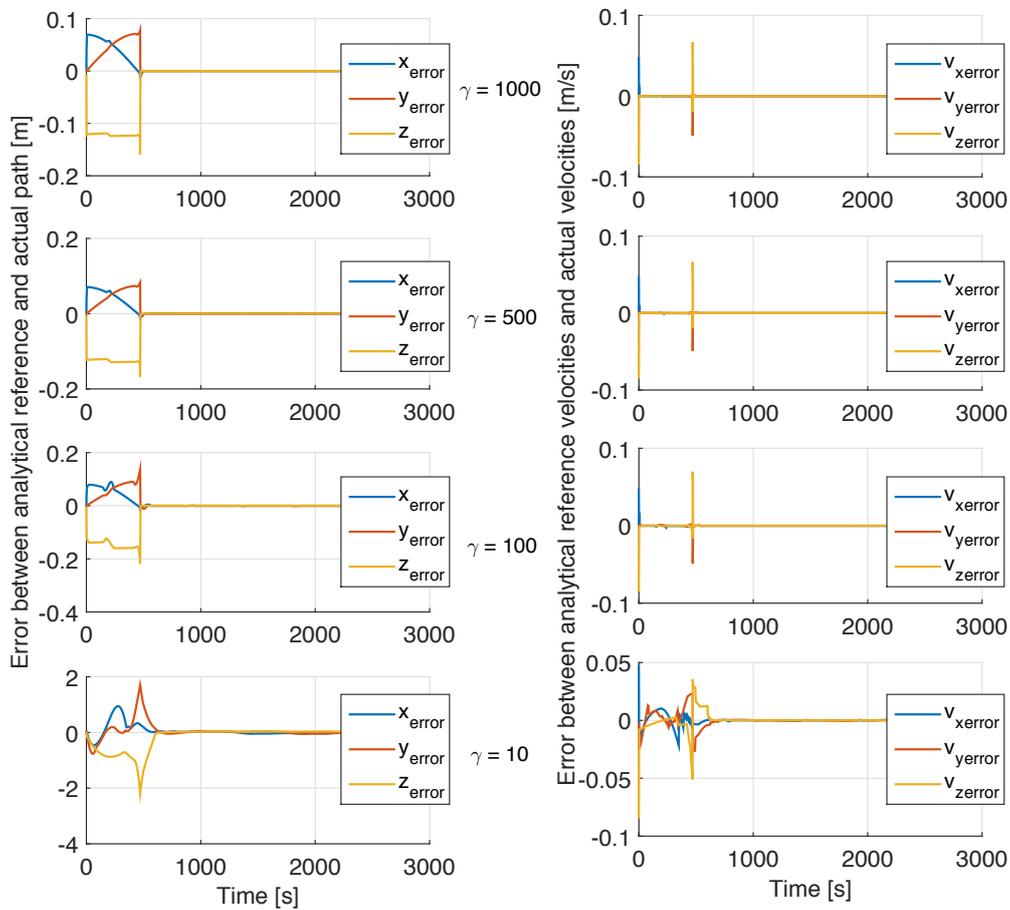


Figure 9-8: Error between the analytical reference position and actual path followed (left) and the error between the analytical reference velocities and actual velocities (right) for different values of γ . These results are obtained with a 1-norm thrust minimization.

2-norm minimization, but a weight of $\gamma=100$. To combine the best of these two approaches, it is proposed to use a value of $\gamma=10$ for the first 800 seconds of the approach (this is the point where the motion is stabilized for the simulation with $\gamma=10$) and a value of $\gamma=100$ for the remainder of the approach.

The resulting path for this approach together with the analytical reference is shown in Figure 9-10. The corresponding state error is shown in Figure 9-11. After 845 s the final circular motion is stabilized and at this point the maximum position error is in the order of 10^{-5} m and the velocity error is in the order of 10^{-6} m/s. The final state error thus meets the requirements for this case. The ΔV required to complete this operation is equal to 0.3974 m/s. These results show that a combined approach results in an approach that can fulfil the requirements while decreasing the ΔV required to complete this operation. To improve this result further, γ should be optimized over time. It could also be beneficial to implement different values for the weight on the position and velocity entries in the slack variable.

Another optimization strategy could be to only put constraints on the states related to the

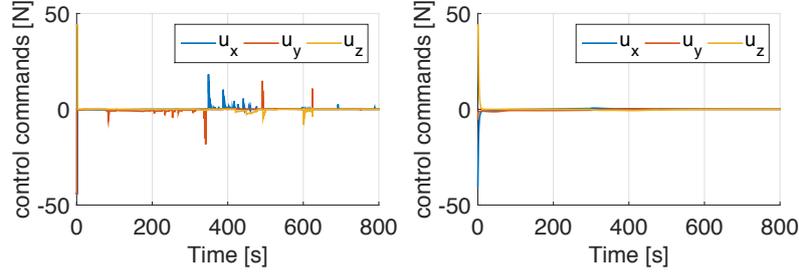


Figure 9-9: Control force profiles for $\gamma=10$. The thrust minimization using the 1 norm is shown on the left and the 2-norm on the right.

Table 9-5: MPC method: Maximum order of magnitude of the state error at the final stage of the approach for different optimization parameters and the time and ΔV required to complete the entire approach.

Opt parameters	Position error [m]	Velocity error [m/s]	ΔV [m/s]	Time [s]
2-norm				
$\gamma=1000$	10^{-6}	10^{-6}	0.4865	485
$\gamma=500$	10^{-6}	10^{-6}	0.4824	480
$\gamma=100$	10^{-5}	10^{-6}	0.4636	500
$\gamma=10$	10^{-1}	10^{-4}	0.3774	800
1-norm				
$\gamma=1000$	10^{-6}	10^{-6}	0.4787	500
$\gamma=500$	10^{-6}	10^{-6}	0.4748	530
$\gamma=100$	10^{-6}	10^{-6}	0.4738	570
$\gamma=10$	10^{-1}	10^{-4}	0.4526	800

final circular motion and just put a constraint on the maximum velocity throughout the rest of the approach. The safety cone would then ensure that the chaser stays within the designated safe area. It is recommended that further research considers these adaptations.

9-2-3 Validation of the Results

To validate the results obtained in the previous section they are compared to the results of similar research by Deloo (2015), which was done in corporation with ESA. The research further analyses the e.deorbit mission scenario, presented in ESA (2012). The major differences that are important when making a comparison with this thesis research are that in Deloo (2015) the following is assumed/used:

- The dynamics models used are solely based on the HCW equations.
- The spin axis offset angle β is equal to 45 degrees.
- The final approach assumes a constant approach velocity of 5 cm/s.
- The applied manoeuvres are all impulsive shots.
- There is no initial velocity difference between the spin axis and the chaser satellite.

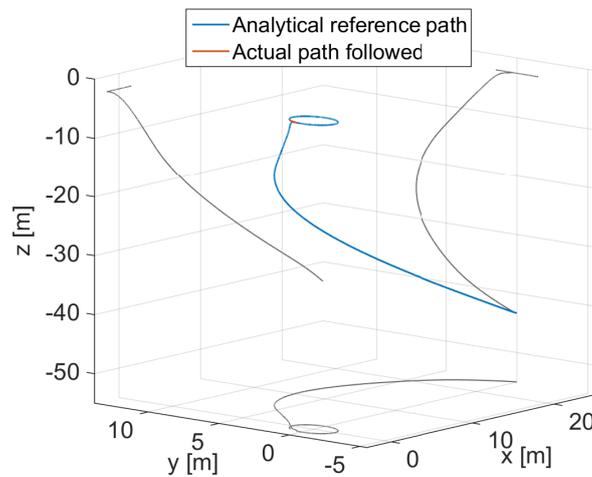


Figure 9-10: Analytical reference states and actual states for the combined approach strategy.

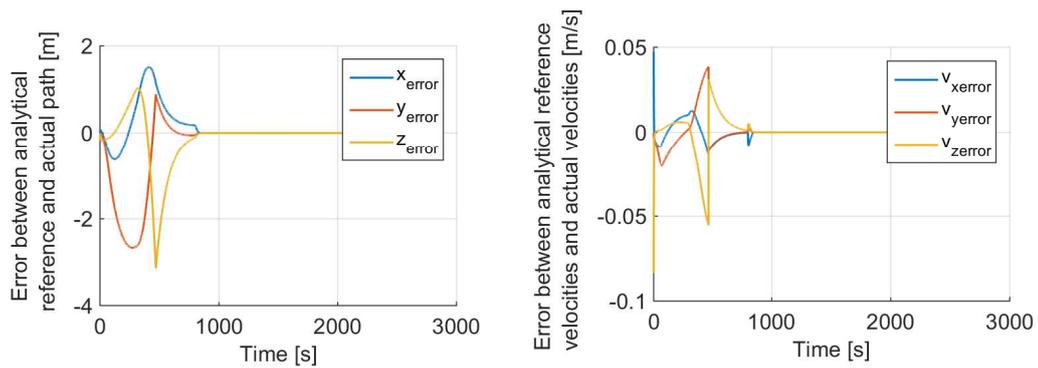


Figure 9-11: Error between the analytical reference position and actual path followed (left) and the error between the analytical reference velocities and actual velocities (right) for the combined approach strategy.

The results on ΔV and time to perform the rendezvous operation are shown in Figure 9-12. The CFMFA 1 phase corresponds to the phase that is analysed in this thesis research. The ΔV required for that phase is about 0.35 m/s, which is slightly lower than the amount required for the combined approach. This still indicates that the combined approach delivers a fairly good result since in Deloo (2015) a zero initial velocity difference between the spin axis and the chaser spacecraft is assumed at the start of the CFMFA 1 phase. Furthermore, a constant approach velocity of 5 cm/s is assumed, which is twice as low as the approach velocity in this thesis research. On top of that, the difference in used dynamical models will also contribute to a higher value for the required propellant. Recall, that D'amico (2010) assessed the error induced by the HCW equations is in the order of 10^{-5} m/s² (see Figure 3-4). The figure also shows that it takes about 1000 s to complete the CFMFA operation. This is twice as long as the time for the baseline scenario in this research, which corresponds to the factor two difference in approach velocity.

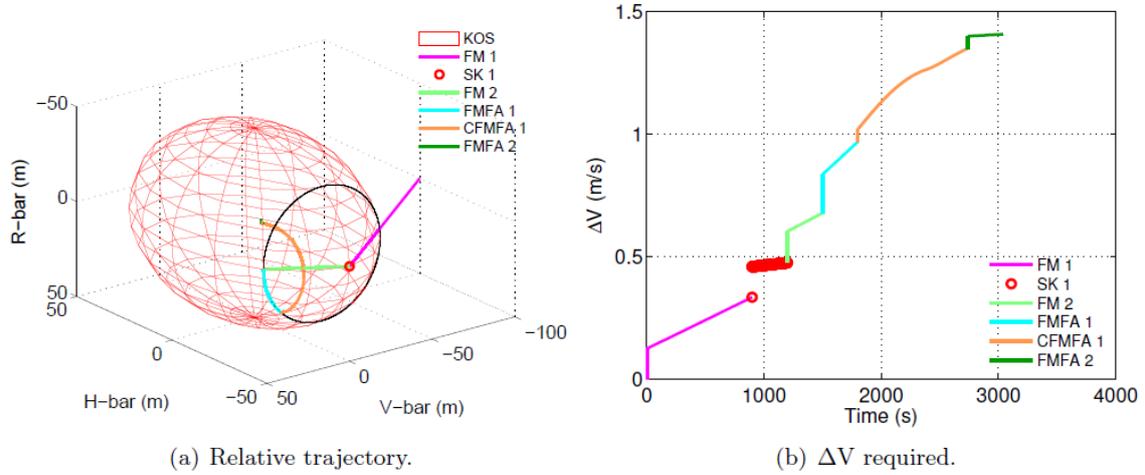


Figure 9-12: The results by Deloo (2015) on ΔV and time to perform the rendezvous operation.

9-3 TEAMS Test Campaign

The conclusions that can be drawn from the results obtained for the MPC+LQR and MPC method with the functional simulations can be validated further the TEAMS simulations. The TEAMS facility, described in Chapter 7, is used to this end. The test campaign parameters for the tests performed with TEAMS are presented in Table 9-6.

These two strategies are compared on the following performance parameters:

- Applied thrust force
- Docking adapters alignment error
- Time to achieve docking

For all the tests in this campaign the target has a fixed position and a rotational velocity of 0.2 deg/s around its CoM. The chaser starts at about 1.0 m relative distance. The general simulation parameters that are the same for every test are given in Table 9-7.

9-4 TEAMS Results

In this section the results obtained with the tests performed on TEAMS will be presented. It should be noted that for all test results the first 15 seconds of each test run is excluded. This

Table 9-6: TEAMS test campaign parameters

Parameter	MPC+LQR	MPC
Guidance sampling rate [Hz]	1	10
Control sampling rate [Hz]	10	-

Table 9-7: General TEAMS simulation parameters.

Description	Parameter	Value
Discretization time step 2	Δt_2 [s]	10
Nr. of steps in the discretization	N [-]	15
Nr. of discretization time steps of size Δt_1	Ns [-]	2
Radius of phase 2 constraint circle	\dot{r}_{ph2} [m]	0.05
Relative velocity constraint to commence phase 2	\dot{r}_{ph2} [m/s]	0.01
Radius of phase 1 arc	R [m]	0.75
Rotational velocity target	ω [deg/s]	0.2
Desired radial approach velocity during phase 2	$v_{rad_{ph2}}$ [m/s]	0.005
Mass of vehicle	M_{TEAMS} [kg]	18.77
MoI of vehicle	J [kg·m ²]	0.29
Maximum thrust in x and y direction	T_{max} [N]	0.0329

is done to eliminate the errors induced by bad estimations obtained by the navigation system, that would otherwise result in biased information on the performance of the strategies. After this period of time the estimator provided sufficiently good results.

9-4-1 Analytical Reference Trajectories

The theoretical analytical reference trajectories for TEAMS were discussed in Section 8-2-1. Tests on TEAMS showed that the reference trajectories generated during the tests differ from the theoretical ones. This is quite logical, when one considers that these trajectories are based on real-time information on the state of the target vehicle. The significance and magnitude of the deviations deserve further explanation. To this end, several reference trajectories that are planned at different times for case 1 and case 2 are shown in Figures 9-13 and 9-14, respectively. They are computed at 1 Hz and 10 Hz (case 1 and case 2), but for clarity only a subset is shown in this figure. Each line represents the analytical reference trajectory at a certain time step, with a size that corresponds to the planning horizon.

It is clear from these figures that they deviate from the theoretical analytical reference trajectory. To explain this, recall, that the trajectory depends on the measured rotational velocity of the target vehicle. This figure shows that the estimator output has a maximum deviation from the theoretical value of about 0.1 rad/s. The reference trajectories are based on the attitude and rotational velocity of the target. The rotational velocity should theoretically remain constant over time, but as can be seen in Figure 9-15, the estimated rotational velocity deviates from the theoretical value. A higher measured rotational velocity thus results in a reference trajectory that encompasses a longer path in total length. Figure 9-14 shows at $t = 74.9$ s that the computed reference trajectory even lies in the opposite direction of the theoretical target motion, which is a result from a measured rotational velocity that has a negative value. To improve output of the estimator of the target vehicle, it should be tuned further and the thruster settings should also be analysed to assess if the current settings negatively influence the target vehicle behavior.

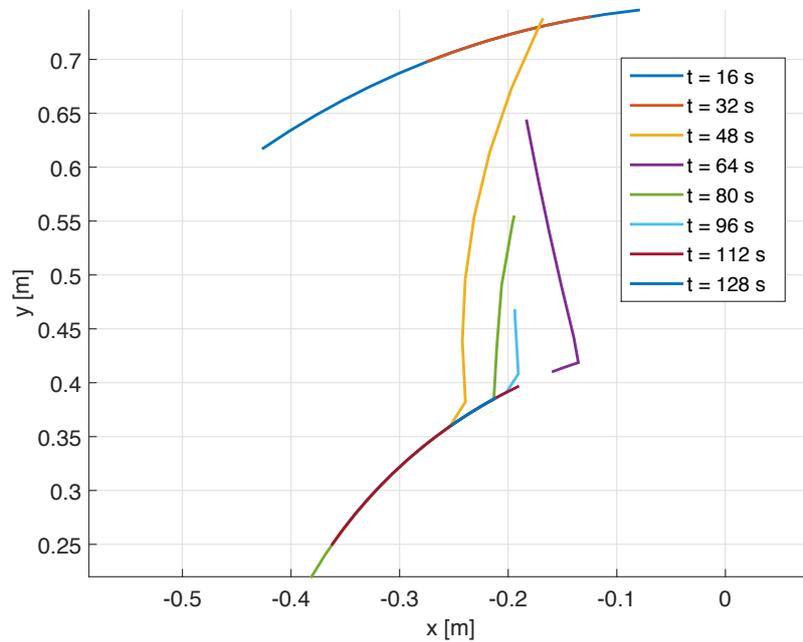


Figure 9-13: Chaser reference trajectories planned at different time steps for case 1

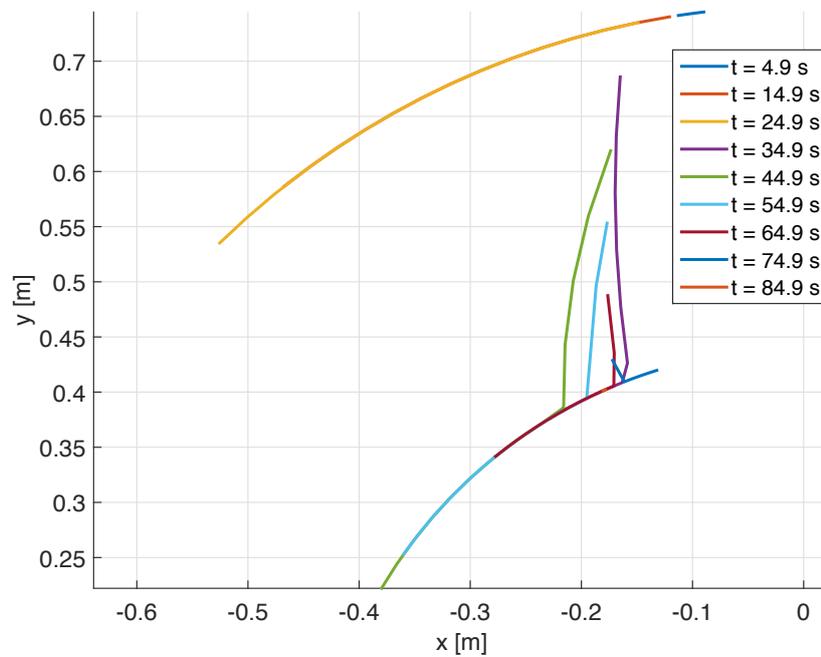


Figure 9-14: Chaser reference trajectories planned at different time steps for case 2

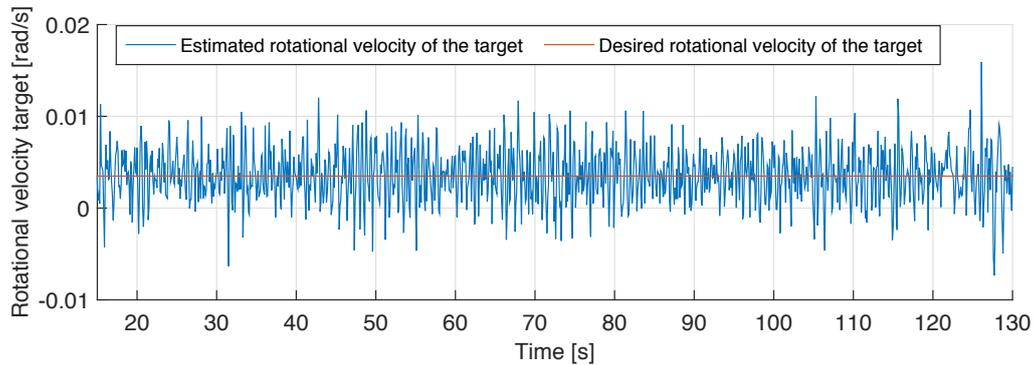


Figure 9-15: The estimated and commanded rotational velocity of the target vehicle for case 1.

9-4-2 Performance of the Guidance and Control System

The resulting paths for the tests performed with the MPC+LQR method (case 1) and MPC method (case 2) are shown in Figures 9-16 and 9-17. The figures show the actual path followed by the CoM of the chaser vehicle. They also depict the path that the virtual point at a distance of twice the adapter length (measured from the CoM of the target vehicle) from the CoM from the target vehicle follows over time. This virtual point thus essentially describes the position that the CoM of the chaser should obtain once the docking conditions are met. It is clear from the figures that for both cases, at the point just after the docking conditions are met, there is a discrepancy between the path of the virtual point and the actual path of the chaser. This is the result from the final sliding motion of the docking pins of the chaser into the docking holes of the target. It can be seen that thereafter the vehicles rotate together for a short period of time before the test is terminated. At the stage when the docking operation was performed the relative velocity was about 2 mm/s, which resulted in a "soft" docking. The figures also shows the analytical reference trajectory that is generated at the start of phase 2. Recall, that the analytical reference for the TEAMS changes over time, and therefore it should be noted that the reference trajectory shown only holds for that specific point in time.

The alignment error of the docking adapters during the rendezvous and docking operation is shown for both cases in Figure 9-18. In both cases the error was small enough to ensure a successful docking operation.

To gain more insight into the robustness of the guidance and control algorithms three tests are performed for each test case. Each test resulted in a successful docking operation. Table 9-8 shows the attitude error at docking, the total magnitude of the applied forces, and the time until docking is achieved, for each test. The initial states of the vehicles vary somewhat from each other for each test (order of 0.1 m) and therefore the time until docking and the applied thrust forces can not be compared directly to one another. The multiple tests do, however, provide some insight into the difference in performance between the MPC+LQR (case 1) and MPC (case 2) methods. It shows, for example, that the MPC method consistently requires less propellant and time to achieve docking than the MPC+LQR method. To verify that this is generally the case a physical test set-up should be designed that ensures that the initial conditions are equal for each test. i.e., constrain the free-floating vehicles before the start of the simulation.

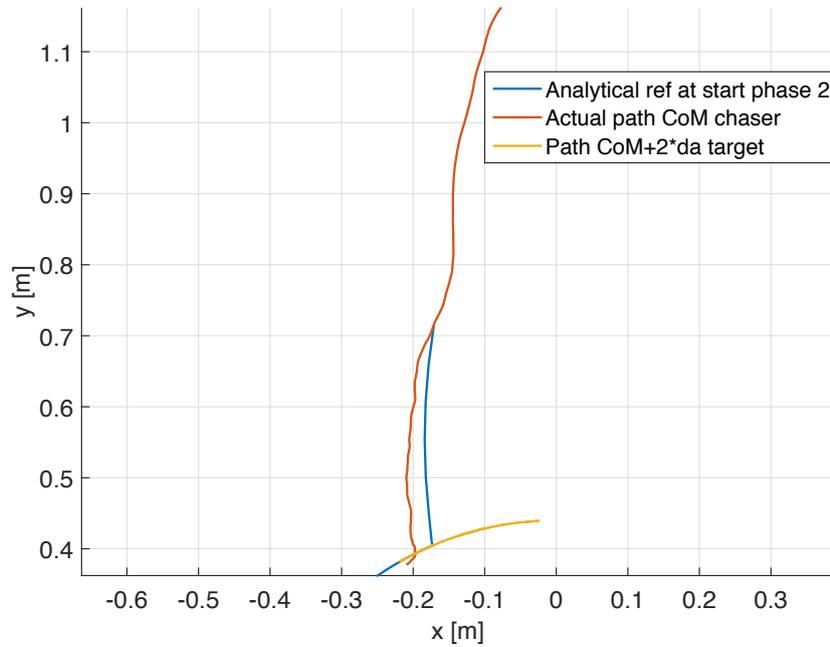


Figure 9-16: Case 1: Analytical reference, actual path of the CoM of the chaser vehicle, and the path of the point that is at two times the adapter distance from the CoM of the target.

Table 9-8: Performance analysis of both cases at docking. Each cases is tested three times.

Parameter	Case 1			Case 2		
	Test 1	Test 2	Test 3	Test 1	Test 2	Test 3
att_{error} [rad]	0.0137	0.0062	0.0083	0.00094	0.0051	-0.0084
T_{totLQR_x} [Ns]	1.65	2.36	1.78	-	-	-
T_{totLQR_y} [Ns]	1.66	1.75	1.58	-	-	-
T_{tot_x} [Ns]	3.55	4.53	3.58	2.50	2.63	2.43
T_{tot_y} [Ns]	3.30	3.64	2.96	2.25	2.19	2.17
t [s]	131.3	162.8	127.3	97.4	107.3	93.8

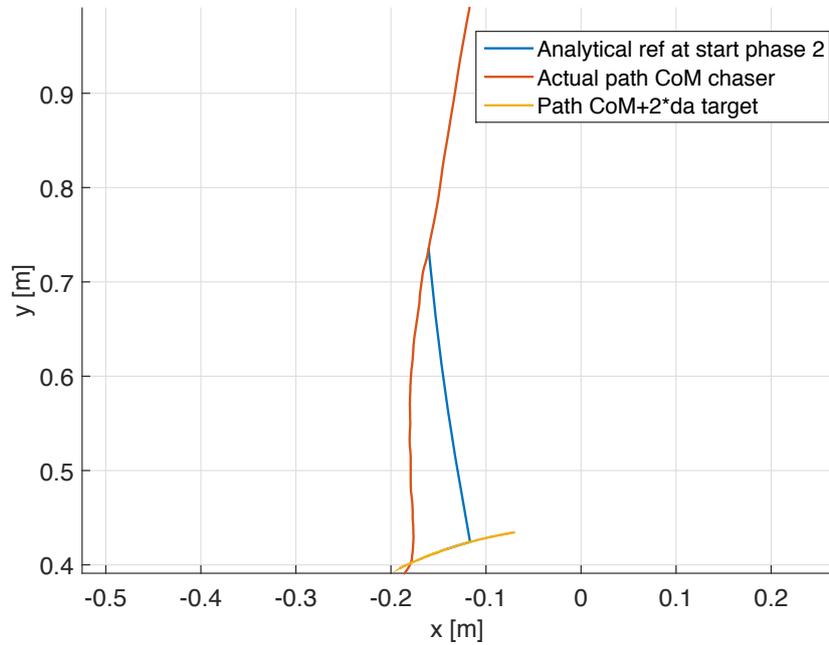


Figure 9-17: Case 2: Analytical reference, actual path of the CoM of the chaser vehicle, and the path of the point that is at two times the adapter distance from the CoM of the target.

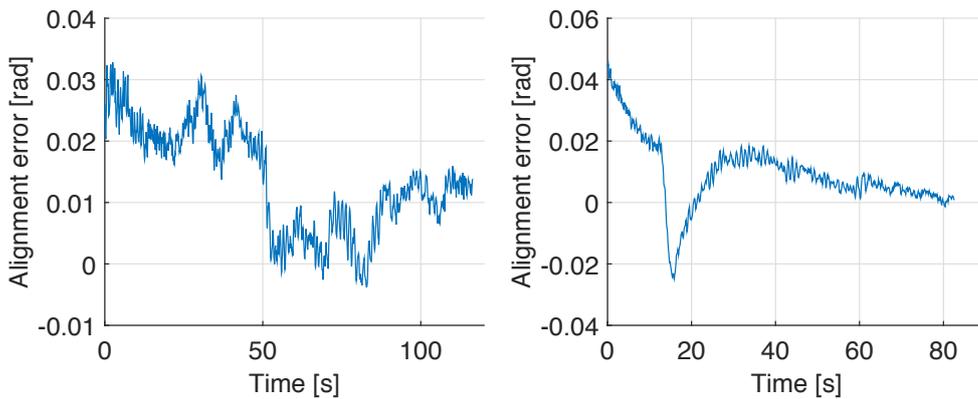


Figure 9-18: The docking adapter alignment error during phase 2 of the approach for case 1(left) and case 2 (right).

Conclusion and Recommendations

This thesis set out to answer the following research question:

How can a satellite rendezvous and dock autonomously with a large uncooperative and tumbling target satellite in a low-Earth orbit in a safe, reliable, and propellant-efficient manner?

The research question is broadly formulated. To scope the research it was decided to design, implement, and evaluate a guidance and control system that enables the final phase of rendezvous and docking operations with the no longer operational satellite ENVISAT. A challenging aspect of the design lies in the tumbling motion ENVISAT has acquired after communication links were lost. Therefore, an approach strategy was adopted that aims to maintain alignment with the spin axis throughout the approach.

The designed guidance and control system uses convex guidance to ensure globally-optimal solutions, while at the same time constraining the trajectory. Furthermore, the concept of MPC is applied to allow for an unconstrained time-to-go. Two guidance and control strategies were constructed to assess the performance of MPC for the reference scenario:

- **MPC+LQR:** This method uses the MPC method to compute the numerical reference trajectory and corresponding acceleration profile at each guidance update step. This trajectory is then also tracked by an LQR at a higher sampling rate than the guidance function to account for disturbances that are not taken into account in the generation of the numerical reference.
- **MPC:** This method solely applies MPC, where the guidance and control function are essentially combined into one. It runs at the same sampling rate as the LQR in the MPC+LQR method and the control is thus solely based on the feedforward commands generated using convex optimization.

The research question states that the operations have to be performed in a safe, reliable and propellant efficient-manner. Thus, these three elements are the primary drivers for the selection of the most appropriate model. The methods were first assessed using functional

simulations. These were then complemented by HIL simulations that made use of the TEAMS facility of DLR in Bremen. This facility emulates the force and momentum-free dynamics of satellites in orbit, i.e., it reproduces the weightlessness and frictionless environment.

Based on the results presented in Chapter 9, it is possible to draw a number of conclusions from the performed research, which will be presented in Section 10-1. And, since no research is entirely conclusive, recommendations for future research towards the topic of autonomous rendezvous and docking with uncooperative, tumbling targets will be given in Section 10-2.

10-1 Conclusions

10-1-1 Functional Simulations

To assess the performance of the two developed guidance and control strategies a dedicated functional simulator was built that only considers translational dynamics. In the simulation scenario the chaser is initially aligned with the spin axis of the target at a distance of 50 m. It approaches the target up to a distance of 3 m, where it stays at this distance while tracking the motion of the spin axis. It is assumed that at this point the docking mechanism will be deployed, which is outside of the scope of this research. The tests for the MPC+LQR method employed the MPC function at an update frequency of 1 Hz and the LQR function at an update frequency of 5 Hz. The tests for the MPC method employed the MPC function at an update frequency of 5 Hz.

Safety and Reliability

Once the relative distance of 3 m from the target was reached, both the MPC and MPC+LQR methods succeeded in tracking the motion of the spin axis, in the presence of the most prominent relative orbital perturbations, with an error that was well below the requirements. The MPC method achieved an accuracy of 10^{-6} m for the position and 10^{-6} m/s for the velocity. The MPC+LQR method achieved an accuracy of 10^{-5} m for the position and 10^{-6} m/s for the velocity. It should be noted though that measurement noise or thruster errors were not included in this analysis.

To ensure that the chaser satellite will not collide during the approach with any part of ENVISAT a safety cone constraint was implemented for both methods in the convex optimizer as a convex second-order cone constraint. This constraint defines a cone in which the chaser is allowed to move throughout the approach. The cone is fixed with respect to the spin axis, i.e., it moves along with the rotation of the spin axis. The results showed that the chaser satellite stays within this cone at all times, and thereby it is guaranteed that it does not collide with ENVISAT or any of its appendages.

Propellant-efficiency

To obtain a guidance and control system that produces propellant-efficient commands within the given constraints the planned trajectory is optimized using convex optimization. The optimizer minimizes a weighed combination of the thrust and the state error.

Based on the obtained results it can be concluded that the MPC was the main driver for the required propellant and that the addition of the LQR leads to an increase of the overall required propellant of 21%. The corresponding control profile of the MPC+LQR method shows that this is the result from the fact that the feedforward commands counteract the feedback commands at times. Further analysis is necessary to determine which factors cause this behavior.

A preliminary analysis to assess the influence of the weight parameter that dictates the weighed minimization of the thrust and the state error was also performed. This was done to assess how much the required propellant could be decreased while keeping the state error sufficiently small. This analysis was then repeated with a 1-norm minimization of the thrust, instead of the conventional 2-norm, to assess the influence of this adaptation on the required propellant. It was found that for the 1-norm minimization there is a break-even point for which a further decrease in the weight on the state error will no longer lead to a reduction in required propellant. A combined approach using the settings that provided the most promising results was then designed and evaluated. From the obtained results it can be concluded that this combined approach leads to a significant decrease in required propellant with regards to the baseline scenario.

10-1-2 TEAMS Simulations

The reference scenario was adapted to enable implementation on TEAMS and the guidance and control algorithms were converted accordingly. The TEAMS vehicles were each equipped with a docking adapter to simulate the actual docking operation, which was omitted in the functional simulations. To enable this, attitude guidance and control was included in the TEAMS simulator. To permit implementation in the real-time environment the computational efficiency of the optimization problem had to be improved. To reduce the optimization problem size the safety cone constraint was omitted and the number of steps in the discretization of the optimization problem had to be reduced.

Safety and Reliability

The functional simulator did not include any external errors, whereas measurement errors, thruster errors, and real-time delays are inherent to HIL simulations. With TEAMS it was shown that a successful rendezvous and docking operation can be performed in the presence of these errors and delays. To ensure a soft docking of the two vehicles a requirement was placed on the relative velocity of 5 mm/s after alignment of the docking adapters is achieved and the final phase of the approach commences.

Propellant-efficiency

The tests performed with TEAMS confirmed the findings of the functional simulation results. For these tests the MPC function was also the main driver for the required propellant and the addition of an LQR resulted in an increase in required propellant as well. Next to this, the tests performed with the MPC method consistently achieved the docking in less time than the MPC+LQR method. It should be noted though that the initial states of the vehicles

vary somewhat from each other for each test and therefore, more tests should be performed to confirm these findings.

10-2 Recommendations

The results of the performed research also provide ample opportunities for recommendations for future research. The recommendations focus on the development of the guidance and control design and the test campaigns. The shortcomings of the models used for the performed research will also be discussed.

10-2-1 Guidance and Control Design for both Simulators

The developed guidance and control algorithms proved to be effective for the reference scenario, but provide plenty of room for improvement.

The designed convex optimizer, used the Euler method to discretize the optimization problem. The errors induced by this method become more significant as the size of the applied time steps increases. To increase the planning horizon, while maintaining performance, it is necessary to apply a method that allows for a larger step size without an increased error. For future research, the main recommendation is therefore, to select a higher-order method to discretize the optimization problem to reduce the error induced by the discretization. It should be noted though that a higher-order method will most likely be computationally less efficient, which could become an issue when implemented on TEAMS. It is therefore important to evaluate which method can achieve these benefits while still enabling implementation in a HIL simulator.

Furthermore, the navigation system is not included in the design of the functional simulator. Including a navigation system provides a more accurate simulation environment. The state of the target would then not be perfectly known. Testing the guidance and control system in the presence of the resulting errors would further increase the reliability of the system. Further analysis should also include other external errors, such as thruster errors.

The development of the simulator for TEAMS was mainly limited by the problem size capabilities of CVXGEN. To enable real-time testing of the entire optimization problem, including all the constraints, it is necessary to increase the computational efficiency of the program. This can be achieved by developing a dedicated solver that allows for larger problem sizes, while maintaining the computational efficiency needed to solve the problem in real-time.

Both test environments showed that the MPC+LQR method delivered an overall worse performance than the MPC method. It should be noted though, that the LQR is based on a simple design and is not tuned to assess if this would increase the performance of the MPC+LQR method. It is therefore unknown if the addition of an extra controller is in general undesirable. Furthermore, both methods show strong oscillations in their thrust profiles, which could be induced by the chosen discretization method. Future research, should assess if this is indeed the case and if not, these results indicate that throttleable thrusters are needed and the feasibility of these thrust profiles should be evaluated.

10-2-2 Approach Procedures for ENVISAT

The approach strategy assumed a specific perfectly known motion of ENVISAT. Since the actual motion is still uncertain, it would be wise to test the test developed guidance and control algorithms for different motion characteristics and thus different approach strategies. This will further validate the robustness of the developed system. To increase the safety of the system it also recommended to include the autonomous activation of CAM procedures in case of failures. Next to this, it is also essential to assess the passive safety of the followed trajectories in case of, for example, engine failure.

The approach strategy is currently based on a reference trajectory. Another option could be to only put constraints on the final circular motion and the maximum velocity throughout the approach, which could lead to a further decrease in required propellant use.

10-2-3 TEAMS

The time to perform tests using the TEAMS facility was limited and that naturally results in plenty of room for recommendations for future testing with TEAMS. First and foremost, it is recommended that when tests are performed with TEAMS, the guidance, navigation, control, and thruster systems are considered as a whole because of the important interactions between all the included subsystems that influence the performance on any one of these systems. To increase the robustness of the developed TEAMS simulator it would be useful to tune the navigation filter further to enable better estimates on the state of the target. The thruster system of the target and chaser vehicles can also be optimized further, to decrease the errors induced by this system.

Finally, the initial states of the vehicles vary somewhat from each other for each performed test. To confirm the drawn conclusions that were based on the results obtained with TEAMS, a physical test set-up should be designed that ensures that the initial conditions are equal for each test, i.e., that the the free-floating vehicles are constrained before the start of each simulation.

Bibliography

- Acikmese, B. and Ploen, S. R. (2007). Convex programming approach to powered descent guidance for mars landing. *Journal of Guidance, Control, and Dynamics*, 30(5):1353–1366.
- AFRL (2005). Xss-11 micro satellite. <http://www.kirtland.af.mil/shared/media/document/AFD-070404-108.pdf>. Accessed: September 3, 2014.
- Alfriend, K., Vadali, S., Gurfil, P., How, J., and Breger, L. (2007). *Spacecraft Formation Flying*. Butterworth-Heinemann, Oxford.
- Baize, L. and Novelli, A. (2010). ATV Jules Verne control centre: from challenges to success. In *Huntsville, Alabama: Proc. of SpaceOps 2010 Conference. AIAA*.
- Bastida Virgili, B. (2014). Investigation on Envisat attitude motion. <https://indico.esa.int/indico/event/46/material/slides/6.pdf>.
- Bate, R., Mueller, D., and J.E., W. (1971). *Fundamentals of astrodynamics*. Dover Publications.
- Benninghoff, H., Boge, T., and Tzschichholz, T. (2012). Hardware-in-the-loop rendezvous simulation involving an autonomous guidance, navigation and control system. *Advances in the Astronautical Sciences*, 145:953–972.
- Betts, J. T. (1998). Survey of numerical methods for trajectory optimization. *Journal of guidance, control, and dynamics*, 21(2):193–207.
- Bizouard, C. (2014). Useful Constants. <http://hpiers.obspm.fr/eop-pc/models/constants.html>. Accessed: 24 July, 2015.
- Blackmore, L., Acikmese, B., and Scharf, D. P. (2010). Minimum-landing-error powered-descent guidance for mars landing using convex optimization. *Journal of guidance, control, and dynamics*, 33(4):1161–1171.
- Blau, P. (2015). Automated Transfer Vehicle - Spacecraft Information. <http://www.spaceflight101.com/atv-spacecraft-information.html>. Accessed: 23 July, 2015.
-

- Boyd, S. and Vandenberghe, L. (2009). *Convex optimization*. Cambridge university press.
- Braeunig (2014). Basic constants. <http://www.braeunig.us/space/constant.htm>. Accessed: November 23, 2014.
- Carignan, C. R. and Akin, D. L. (2000). The reaction stabilization of on-orbit robots. *Control Systems, IEEE*, 20(6):19–33.
- Clohessy, W. and Wiltshire, R. (1960). Terminal guidance system for satellite rendezvous. *Journal of the Aerospace Sciences*, 27:653–658.
- COSPAR (2012). Models of the Earth’s Upper Atmosphere. COSPAR International Reference Atmosphere CIRA-2012.
- Curtis, H. (2013). *Orbital Mechanics for Engineering Students 3rd edition*. Butterworth-Heinemann, Boston, MA.
- D’amico, S. (2010). Autonomous formation flying in low earth orbit. http://repository.tudelft.nl/assets/uuid...399d.../Damico_PhD_15012010.pdf. PhD thesis.
- Davis, T. and Melanson, D. (2004). Xss-10 microsatellite flight demonstration program results. In *Defense and Security*, pages 16–25. International Society for Optics and Photonics.
- De Bruijn, F. and Gill, E. (2014). Influence of sensor and actuator errors on impulsive satellite formation control methods. *Acta Astronautica*, 94(2):608–618.
- De Bruijn, F., Gill, E., and How, J. (2011). Comparative analysis of cartesian and curvilinear clohessy-wiltshire equations. Proceedings of the 22nd International Symposium on Space Flight Dynamics.
- Deloo, J. (2015). Analysis of the rendezvous phase of e.deorbit. http://repository.tudelft.nl/assets/uuid:7f445d54-c758-45a8-b87e-a9c0eec61617/Thesis_JDELOO.pdf. Master thesis.
- Di Cairano, S., Park, H., and Kolmanovsky, I. (2012). Model predictive control approach for guidance of spacecraft rendezvous and proximity maneuvering. *International Journal of Robust and Nonlinear Control*, 22(12):1398–1427.
- Di Mauro, G., Schlotterer, M., Theil, S., and Lavagna, M. (2013). Experimental implementation of SDRE method for autonomous rendezvous and docking maneuvering. In *The International Conference on Spacecraft Formation Flying Missions and Technologies (SFFMT), 2013*. German Space Operations Center of DLR.
- ESA (2012). e.deorbit assessment. Private communication Deloo, 2015. CDF Study Report: CDF-135(C).
- Fehse, W. (2003). *Automated Rendezvous and Docking of Spacecraft*. Cambridge University Press.
- Friend, R. (2008). Orbital express program summary and mission overview. In *SPIE Defense and Security Symposium*, pages 695803–695803. International Society for Optics and Photonics.
-

- Gao, H., Yang, X., and Shi, P. (2009). Multi-objective robust control of spacecraft rendezvous. *Control Systems Technology, IEEE Transactions on*, 17(4):794–802.
- Gerth, I. (2014). Convex optimization for constrained and unified landing guidance. Master thesis.
- Goodman, J. (2006). History of space shuttle rendezvous and proximity operations. *Journal of Spacecraft and Rockets*, 43(5):944–959.
- Gorinevsky (2005). Lecture 14 - Model Predictive Control Part 1: The Concept. [Lecture series and slides].
- Grant, M. and Boyd, S. (2014). CVX: Matlab software for disciplined convex programming, version 2.1. <http://cvxr.com/cvx>. Accessed: 10 December, 2014.
- Grant, M., Boyd, S., and Ye, Y. (2008). Cvx: Matlab software for disciplined convex programming.
- Hartley, E. N., Trodden, P. A., Richards, A. G., and Maciejowski, J. M. (2012). Model predictive control system design and implementation for spacecraft rendezvous. *Control Engineering Practice*, 20(7):695–713.
- Hill, G. (1878). Researches in the lunar theory. *American Journal of Mathematics*, 1(1):5–26, 129–147, 245–260.
- IADC (2013). Stability of the future leo environment. Technical report, Inter-Agency Space Debris Co-ordination Committee.
- Jacobsen, S., Lee, C., Zhu, C., and Dubowsky, S. (2002). Planning of safe kinematic trajectories for free flying robots approaching an uncontrolled spinning satellite. In *ASME 2002 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, pages 1145–1151. American Society of Mechanical Engineers.
- Jakhu, R. (2012). Active debris removal - an essential mechanism for ensuring the safety and sustainability of outer space. In *49th Session of the Scientific and Technical Subcommittee Committee on the Peaceful Uses of Outer Space, United Nations*.
- JAXA (2003). Engineering test satellite vii "kiku-7" (ets-vii). <http://global.jaxa.jp/projects/sat/ets7/index.html>. Accessed: July 23, 2015.
- Kessler, D., Johnson, N., Liou, J., and Matney, M. (2010). The kessler syndrome: implications to future space operations. *Advances in the Astronautical Sciences*, 137(8):2010.
- Legostaev, V. and Raushenbach, B. (1969). Automatic assembly in space. *Cosmic Research*, 7:723.
- Liou, J., Johnson, N., and Hill, N. (2010). Controlling the growth of future leo debris populations with active debris removal. *Acta Astronautica*, 66(56):648 – 653.
- Lu, P. and Liu, X. (2013). Autonomous trajectory planning for rendezvous and proximity operations by conic optimization. *Journal of Guidance, Control, and Dynamics*, 36(2):375–389.
-

- Luenberger, D. and Ye, Y. (2008). *Linear and nonlinear programming*, volume 116. Springer.
- Malik, T. (2006). NASA Releases Summary of DART Mission Mishap. <http://www.space.com/2408-nasa-releases-summary-dart-mission-mishap.html>. Accessed: 23 July, 2015.
- Matsievsky (2006). [https://commons.wikimedia.org/wiki/File:The_Soviet_Union_1968_CPA_3622_sta,mp_\(Kosmos_186_and_Kosmos_188_linking_in_Space\).png](https://commons.wikimedia.org/wiki/File:The_Soviet_Union_1968_CPA_3622_sta,mp_(Kosmos_186_and_Kosmos_188_linking_in_Space).png). From collection of Matsievsky.
- Mattingley, J. and Boyd, S. (2012). Cvxgen: a code generator for embedded convex optimization. *Optimization and Engineering*, 13(1):1–27.
- Miller, D., Saenz-Otero, A., Wertz, J., Chen, A., Berkowski, G., Brodel, C., Carlson, S., Carpenter, D., Chen, S., Cheng, S., et al. (2000). Spheres: a testbed for long duration satellite formation flying in micro-gravity conditions. In *Proceedings of the AAS/AIAA Space Flight Mechanics Meeting, Clearwater, FL, Paper No. AAS 00-110*.
- Montenbruck, O. and Gill, E. (2000). *Satellite Orbits: Models, Methods, and Applications*. Springer.
- Munoz, J. (2011). *Rapid path-planning algorithms for autonomous proximity operations of satellites*. PhD thesis, University of Florida.
- N2YO (2014). Real time satellite tracking and predictions. <http://www.n2yo.com/satellite/?s=27386>. Accessed: 29 September, 2014.
- NASA (1985). Flat floor area building 4619. <http://mix.msfc.nasa.gov/abstracts.php?p=468>. Accessed: September 19, 2014.
- NASA (2006). Overview of the dart mishap investigation results, for public release. http://www.nasa.gov/pdf/148072main_DART_mishap_overview.pdf. Accessed: August 27, 2014.
- Nolet, S. (2007). *Development of a guidance, navigation and control architecture and validation process enabling autonomous docking to a tumbling satellite*. PhD thesis, Massachusetts Institute of Technology.
- Ohkami, Y. and Kawano, I. (2003). Autonomous rendezvous and docking by engineering test satellite vii: a challenge of japan in guidance, navigation and control breakwell memorial lecture. *Acta Astronautica*, 53(1):1–8.
- Paraskevopoulos, P. (2001). *Modern control engineering*. CRC Press.
- Park, H., Di Cairano, S., and Kolmanovsky, I. (2011). Model predictive control for spacecraft rendezvous and docking with a rotating/tumbling platform and for debris avoidance. In *American Control Conference (ACC), 2011*, pages 1922–1927. IEEE.
- Pelivan, I., Heidecker, A., and Theil, S. (2012). High performance satellite dynamics and control simulation for multi-purpose application. *Journal of Aerospace Engineering*, 4(3):120.
-

- Pfeiffer, C. (1968). An analysis of guidance modes. Technical report Electronics Research Center, NASA. <http://ntrs.nasa.gov/archive/nasa/casi.ntrs.nasa.gov/19680027313.pdf#page=13>.
- Polites, M. (1999). Technology of automated rendezvous and capture in space. *Journal of Spacecraft and Rockets*, 36(2):280–291.
- Prussing, J. and Conway, B. (1993). *Orbital Mechanics*. Oxford University Press.
- Rea, J. R. (2009). *An investigation of fuel optimal terminal descent*. PhD thesis, University of Texas. <http://repositories.lib.utexas.edu/handle/2152/18393>.
- Richards, A. and How, J. (2003). Performance evaluation of rendezvous using model predictive control. In *AIAA Guidance, Navigation, and Control Conference and Exhibit*, page 5507.
- Rodriguez, H. and Liou, J. (2008). orbital debris : Past, present, and future. In *Proceedings of American Institute of Aeronautics and Astronautics (AIAA) Annual Technical Symposium*. American Institute of Aeronautics and Astronautics.
- Roe, F., Howard, R., and Murphy, L. (2004). Automated rendezvous and capture system development and simulation for nasa. In *Defense and Security*, pages 118–125. International Society for Optics and Photonics.
- Romano, M., Friedman, D., and Shay, T. (2007). Laboratory experimentation of autonomous spacecraft approach and docking to a collaborative target. *Journal of Spacecraft and Rockets*, 44(1):164–173.
- Ross, I. M. (2006). Space trajectory optimization and l1-optimal control problems. *Modern astrodynamics*, 1:155–188.
- Saponara, M., Barrena, V., Bemporad, A., Hartley, E., Maciejowski, J., Richards, A., Tramutola, A., and Trodden, P. (2013). Model predictive control application to spacecraft rendezvous in mars sample return scenario. In *Progress in Flight Dynamics, Guidance, Navigation, Control, Fault Detection, and Avionics*, volume 6, pages 137–158. EDP Sciences.
- Sauceda, F. (2001). International Space Station Program: Space Station Reference Coordinate Systems. Technical report, NASA.
- Schlotterer, M. and Theil, S. (2010). Testbed for on-orbit servicing and formation flying dynamics emulation. In *AIAA, editor, "AIAA Guidance, Navigation and Control Conference and Exhibit," AIAA-2010-8108*.
- Schlotterer, M. and Novoschilov, S. (2012). On-ground path planning experiments for multiple satellites. In *International Symposium on Space Flight Dynamics (ISSFD), 2012*. Jet Propulsion Laboratory.
- Schwartz, J. L., Peck, M. A., and Hall, C. D. (2003). Historical review of air-bearing spacecraft simulators. *Journal of Guidance, Control, and Dynamics*, 26(4):513–522.
-

- Strandmoe, S., DePasquale, E., Escane, I., Augelli, M., Personne, G., Cavrois, B., Fau, N., Yu, M., Zink, M., Clerc, X., et al. (2008). Automated transfer vehicle (atv) flight control achievements. In *7th International ESA Conference on Guidance, Navigation and Control Systems*.
- Tillerson, M., Inalhan, G., and How, J. P. (2002). Co-ordination and control of distributed spacecraft systems using convex optimization techniques. *International Journal of robust and nonlinear control*, 12(2-3):207–242.
- Tütüncü, R., Toh, K. C., and Todd, M. J. (2003). Solving semidefinite-quadratic-linear programs using sdpt3. *MATHEMATICAL PROGRAMMING*, 95:189–217.
- Vivarad, P. (2013). Orbital elements. <http://www.vivarad.info/mteaching4.php>. Accessed: July 23, 2015.
- Wakker, K. (2010). *Astrodynamics I (lecture notes) AE4-874 part I*. Delft University of Technology.
- Wertz, J., Everett, D., and Puschell, J. (2011). *Space Mission Engineering: The New SMAD*. Microcosm Press.
- Wiegmann, B., Hovater, M., and Kos, L. (2012). Nasa’s marshall space flight center recent studies and technology developments in the area of ssa/orbital debris. In *Advanced Maui Optical and Space Surveillance Technologies Conference*.
- Woffinden, D. and Geller, D. K. (2007). Navigating the road to autonomous orbital rendezvous. *Journal of Spacecraft and Rockets*, 44(4):898–909.
- Wormnes, K., Le Letty, R., Summerer, L., Schonenborg, R., Dubois-Matra, O., Luraschi, E., Cropp, A., Krag, H., and Delaval, J. (2013). Esa technologies for space debris remediation. In *6th European Conference on Space Debris, Darmstadt, Germany*, pages 22–25.
-