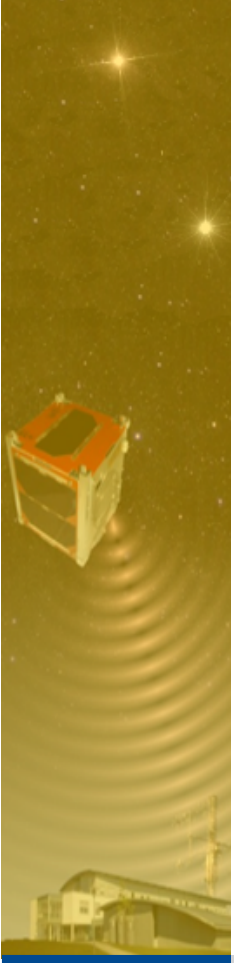


Institut für Informatik
Lehrstuhl für Robotik und Telematik
Prof. Dr. K. Schilling
Prof. Dr. A. Nüchter

Würzburger Forschungsberichte in Robotik und Telematik

Uni Wuerzburg Research Notes in Robotics and Telematics

Julius-Maximilians-
UNIVERSITÄT WÜRZBURG

Band 26

Dissertation an der Graduate School of Science and Technology

Julian Scharnagl

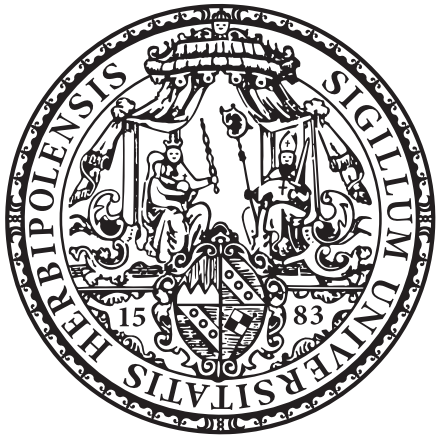
Distributed Guidance, Navigation and Control for Satellite Formation Flying Missions

Julius-Maximilians-Universität Würzburg
Graduate School of Science and Technology

Doctoral thesis / *Dissertation*
for the doctoral degree / *zur Erlangung des Doktorgrads*
Doctor rerum naturalium (Dr. rer. nat.)

Distributed Guidance, Navigation and Control for Satellite Formation Flying Missions

Verteilte Leit- und Regelungsverfahren für Satellitenformationen



Submitted by / *Vorgelegt von*
Julian Scharnagl
from / aus
Schweinfurt
Würzburg, 2021

Submitted on / *Eingereicht am:* December 9th, 2021

Members of thesis committee / *Mitglieder des Promotionskomitees*

Chairperson / Vorsitz: Prof. Dr. Christian Klingenberg

1. Reviewer and Examiner / 1. Gutachter und Prüfer: Prof. Dr. Klaus Schilling

2. Reviewer and Examiner / 2. Gutachter und Prüfer: Prof. Dr. Sergio Montenegro

3. Examiner / 3. Prüfer: Prof. Dr. Dusan Stipanovic

Day of thesis defense / *Tag des Promotionskolloquiums:* September 1st, 2022

Abstract

Ongoing changes in spaceflight – continuing miniaturization, declining costs of rocket launches and satellite components, and improved satellite computing and control capabilities – are advancing Satellite Formation Flying (SFF) as a research and application area. SFF enables new applications that cannot be realized (or cannot be realized at a reasonable cost) with conventional single-satellite missions. In particular, distributed Earth observation applications such as photogrammetry and tomography or distributed space telescopes require precisely placed and controlled satellites in orbit.
Several enabling technologies are required for SFF, such as inter-satellite communication, precise attitude control, and in-orbit maneuverability. However, one of the most important requirements is a reliable distributed Guidance, Navigation and Control (GNC) strategy. This work addresses the issue of distributed GNC for SFF in 3D with a focus on Continuous Low-Thrust (CLT) propulsion satellites (e.g., with electric thrusters) and concentrates on circular low Earth orbits. However, the focus of this work is not only on control theory, but control is considered as part of the system engineering process of typical small satellite missions. Thus, common sensor and actuator systems are analyzed to derive their characteristics and their impacts on formation control. This serves as the basis for the design, implementation, and evaluation of the following control approaches: First, a Model Predictive Control (MPC) method with specific adaptations to SFF and its requirements and constraints; second, a distributed robust controller that combines consensus methods for distributed system control and H_∞ robust control; and finally, a controller that uses plant inversion for control and combines it with a reference governor to steer the controller to the target on an optimal trajectory considering several constraints. The developed controllers are validated and compared based on extensive software simulations. Realistic 3D formation flight scenarios were taken from the Networked Pico-Satellite Distributed System Control (NetSat) cubesat formation flight mission. The three compared methods show different advantages and disadvantages in the different application scenarios. The distributed robust consensus-based controller for example lacks the ability to limit the maximum thrust, so it is not suitable for satellites with CLT. But both the MPC-based approach and the plant inversion-based controller are suitable for CLT SFF applications, while showing again distinct advantages and disadvantages in different scenarios.
The scientific contribution of this work may be summarized as the creation of novel and specific control approaches for the class of CLT SFF applications, which is still lacking methods withstanding the application in real space missions, as well as the scientific evaluation and comparison of the developed methods.

Zusammenfassung

Die anhaltenden Veränderungen in der Raumfahrt – die fortschreitende Miniaturisierung, die sinkenden Kosten für Raketenstarts und Satellitenkomponenten sowie die verbesserten Rechen- und Steuerungsmöglichkeiten von Satelliten – fördern den Satelliten-Formationsflug (SFF) als Forschungs- und Anwendungsgebiet. SFF ermöglicht neue Anwendungen, die mit herkömmlichen Einzelsatellitenmissionen nicht (oder nicht mit vertretbarem Aufwand) realisiert werden können. Insbesondere verteilte Erdbeobachtungsanwendungen wie Photogrammetrie und Tomographie oder verteilte Weltraumteleskope erfordern präzise positionierte und kontrollierte Satelliten in der Umlaufbahn.

Für den SFF sind verschiedene Basistechnologien erforderlich, z. B. Kommunikation zwischen den Satelliten, präzise Lageregelung und Manövrierfähigkeit. Eine der wichtigsten Anforderungen sind jedoch zuverlässige verteilte Leit- und Regelungsverfahren (Guidance, Navigation and Control, GNC). Diese Arbeit befasst sich mit dem Thema der verteilten GNC für SFF in 3D mit dem Schwerpunkt auf Satelliten mit kontinuierlichem, niedrigen Schub (Continuous Low-Thrust, CLT) z.B. mit elektrischen Triebwerken und legt den Fokus hier zusÃ¤tzlich auf niedrige kreisförmige Erdumlaufbahnen. Der Schwerpunkt dieser Arbeit liegt jedoch nicht nur auf der Regelungstheorie, vielmehr wird Regelung als Teil des Systementwicklungsprozesses typischer Kleinsatellitenmissionen betrachtet. So werden gängige Sensor- und Aktuatorsysteme analysiert, um ihre Eigenschaften und ihre Auswirkungen auf die Formationskontrolle abzuleiten. Dies dient als Grundlage für den Entwurf, die Implementierung und die Bewertung der folgenden Regelungsansätze: Erstens eine Modellprädiktive Regelung (Model-Predictive Control, MPC) mit spezifischen Anpassungen an die Anforderungen und Beschränkungen des SFFs, zweitens ein robuster Regler, der Konsensmethoden für die Steuerung verteilter Systeme mit robuster H_∞-Regelung kombiniert, und schließlich ein kaskadierter Regler, der zur Steuerung die Regelstrecke invertiert und dessen Referenz von einem Referenzregler auf einer optimalen Trajektorie unter Berücksichtigung verschiedener Beschränkungen zum Ziel gesteuert wird. Die entwickelten Regler werden auf der Grundlage umfangreicher Softwaresimulationen validiert und miteinander verglichen. Realistische 3D-Formationsflug-Szenarien wurden der NetSat-Formationsflug-Mission entnommen. Die drei verglichenen Methoden zeigen unterschiedliche Vor- und Nachteile in den verschiedenen Anwendungsszenarien. Der verteilten robusten konsensbasierten Regelung fehlt bspw. die Fähigkeit, den maximalen Schub zu begrenzen, sodass sie nicht für Satelliten mit CLT geeignet ist. Aber sowohl der MPC-basierte Ansatz als auch der auf der Invertierung der Regelstrecke basierende Ansatz sind für CLT-

SFF-Anwendungen geeignet und weisen wiederum ander Vor- und Nachteile in unterschiedlichen Szenarien auf.
Der wissenschaftliche Beitrag dieser Arbeit besteht in der Entwicklung neuartiger und spezifischer Regelungsansätze für die Klasse der CLT-SFF-Anwendungen, für die es noch keine Methoden gibt, die der Anwendung in realen Weltraummissionen standhalten, sowie in der wissenschaftlichen Bewertung und dem Vergleich der entwickelten Methoden.

Acknowledgments

Various people have supported me in many ways during my PhD thesis and made this work possible – in the following I would like to thank these people.

I would like to express my sincere thanks to Prof. Dr. Klaus Schilling, Chair of Robotics and Telematics at the University of Würzburg, who gave me the opportunity to be part of the space team and to conduct this dissertation. I would also like to thank Prof. Dr. Sergio Montenegro for supporting my thesis and Prof. Dr. Dusan Stipanovic, who provided me with moral support and helpful advice during the long years of this PhD. I would also like to thank Dr. Finn Ankersen for his support and the opportunity to work with him at ESA's European Space Research and Technology Centre (ESTEC). Many thanks to Dr. Stephan Schröder-Köhne for his help in administrative matters and for organizing wonderful and informative events.

My special thanks go to my colleague and good friend Florian Kempf for the many discussions and helpful thoughts on control theory topics and the endless effort he spent to support me. I would also like to thank my dedicated students, Lukas Steindorf, Eric Reinthal, Panagiotis Kremmydas, Ursula Scherm, Johannes Dauner, Florian Kunzi, Ivaylo Angelov and all others for the fruitful joint work on formation control. I enjoyed every hour. Special thanks also go to the ZfT family – Slavi Dombrovski, Ilham Mammadov, Alexander Kleinschrodt, Roland Haber, Oliver Ruf and the other wonderful colleagues who supported me during this work.

Last but not least, I would like to thank my family who has supported me not only during this work, but throughout my life. A very special word of thanks goes to you, my love, for your unconditional support and for always allowing me to share our time together with my work.

Contents

List of Abbreviations

The following abbreviations are used in this work. They are listed in alphabetical order together with the page of their first appearance.

ACS	Attitude Control System	166
ADCS	Attitude Determination and Control System	155
AFRL	Air Force Research Laboratory	16
AIT	Assembly, Integration and Test	8
AOCS	Attitude and Orbit Control System	23
AODS	Attitude and Orbit Determination System	182
ARE	Algebraic Riccati Equation	84
ATF	Along Track Formation	6
ATV	Automated Transfer Vehicle	7
BNSC	British National Space Centre	16
CAD	Computer-Aided Design	24
CLF	Control Lyapunov Function	84
CLT	Continuous Low-Thrust	10
CLTF	Closed-Loop Transfer Function	92
CNES	French National Centre for Space Studies	16
CO	Circular Orbit	61
COE	Classical Orbital Element	33
CWF	Cartwheel Formation	61
CSA	Canadian Space Agency	16
CSV	Comma-Separated Values	180
DEM	Digital Elevation Model	61
DLR	German Aerospace Center	6
DOE	Differential Orbital Element	36
DoF	Degrees of Freedom	24
DOT	Design Option Tree	159
ECI	Earth-Centered Inertial	28
EH	Euler-Hill	29
ELISA	ELectronic Intelligence by SAtellite	18

EOE	Equinoctial Orbital Element	36
ERC	European Research Council	21
ESA	European Space Agency	5
ESTEC	European Space Research and Technology Centre	157
FEEP	Field-Emission Electric Propulsion	173
FPGA	Field Programmable Gate Array	21
FSF	Formation Simulation Framework	179
GA	Gim-Alfriend	56
GNC	Guidance, Navigation and Control	9
GNSS	Global Navigation Satellite System	8
GPS	Global Positioning System	5
GRACE	Gravity Recovery and Climate Experiment	6
GRACE-FO	GRACE Follow-On	18
GSFC	Goddard Space Flight Center	16
GTO	Geostationary Transfer Orbit	104
GVE	Gauss's Variational Equations	57
HEO	Highly Elliptical Orbit	18
HCW	Hill-Clohessy-Wiltshire	29
IMU	Inertial Measurement Unit	24
ISL	Inter-Satellite Link	21
ISS	International Space Station	28
JAXA	Japan Aerospace Exploration Agency	16
JPL	Jet Propulsion Laboratory	16
LEO	Low Earth Orbit	9
LEOP	Launch and Early Orbit phase	7
LFT	Linear Fractional Transformation	92
LC	Lyapunov Controller	127
LQR	Linear Quadratic Regulator	7
LTI	Linear Time-Invariant	50
LTV	Linear Time-Varying	81
LVLH	Local-Vertical, Local-Horizontal	8
MEMS	Micro-Electro-Mechanical Systems	154
MEO	Medium Earth Orbit	161
MH	Mission Handler	182
MIMO	Multiple-Input Multiple-Output	105
MMS	Magnetospheric Multiscale Mission	6

MOEA	Multi-Objective Evolutionary Algorithm	104
MPC	Model Predictive Control	7
NASA	National Aeronautics and Space Administration	6
NanoFEEP	Nano Field-Emission Electric Propulsion	21
NetSat	Networked Pico-Satellite Distributed System Control	7
NOE	Non-Singular Orbital Element	35
NPI	Networking/Partnering Initiative	291
OBC	On-Board Computer	110
ODE	Ordinary Differential Equation	48
ONERA	Office National d'Etudes et de Recherches AÃ©rospatiales	16
Orekit	Orbit Extrapolation Kit	44
PCB	Printed Circuit Board	155
PCDU	Power Control and Distribution Unit	183
PCO	Projected Circular Orbit	6
Prisma	Prototype Research Instruments and Space Mission Technology Advancement	8
QP	Quadratic Programming	87
RG	Reference Governor	126
RK	Runge-Kutta	186
RMS	Root Mean Square	52
ROE	Relative Orbit Element	8
RTN	Radial-Tangential-Normal	29
RvD	Rendezvous and Docking	15
SAR	Synthetic Aperture Radar	8
SEZ	South-East-Normal (topocentric-horizon)	32
SFF	Satellite Formation Flying	5
STM	State-Transition Matrix	41
STS	Space Transportation System	110
TAI	International Atomic Time (Temps Atomique International)	186
TF	Transfer Function	92
TH	Tschauner-Hempel	51
THF	Tetrahedron Formation	61
TLE	Two Line Elements	24
TOM	Telematics Earth Observation Mission	26
TT	Terrestrial Time	28
UDP	User Datagram Protocol	181

UHF	Ultra High Frequency	24
UKSA	United Kingdom Space Agency	16
UTC	Universal Time Coordinated	186
UWE	University of Würzburg Experimental	21
YA	Yamanaka-Ankersen	52
ZfT	Zentrum für Telematik e.V.	7
ZOH	Zero-Order Hold	88

Chapter 1

Introduction

1.1 Overview

A paradigm shift in spacecraft engineering emerges from single and large multi-functional satellites towards cooperating groups of small satellites, forming a constellation, cluster or formation. This will enable innovative approaches in areas like Earth observation, scientific exploration or telecommunication.

Walker (1984) provided the first mathematical description for placing spacecraft in different orbits around the Earth. The so called Walker-constellation places multiple satellites in each orbit, while the different orbital planes are inclined against each other. By this, an almost global coverage (except of the pole regions) can be obtained with a minimal number of satellites. Early use of the Walker-constellation was made by the first global navigation satellite system Global Positioning System (GPS), which consisted of ten spacecraft in its first phase (Green et al., 1989). Other applications are satellite-based voice and data transfer around the world, which is provided by the constellations Iridium (Maine et al., 1995) and Globalstar (Smith, 1996). In contrast to constellations, the satellites in a cluster or formation are closer together. One of the first mission concepts (including an early controller design) that makes relevant use of a satellite formation in orbit is a proposed infrared space interferometer by Sholomitsky (1977). A satellite cluster consists of multiple spacecraft, which are placed in similar orbits, maintaining a bounded relative motion. A notable mission in this field is European Space Agency (ESA)'s first cornerstone project Cluster-II, which was finally launched in the year 2000 after a failed start of Cluster in 1996 (Escoubet et al., 2001). The tetrahedron cluster consists of four spacecraft with inter-satellite distances between 100 and 2000 km and measures the magnetic field of the Earth. The use of a cluster in this field of research enables spatially distributed measurements at the same time, since the magnetic field can only be measured in-situ. The distinction to a satellite formation lies in the nature of orbit control. While within a cluster (like within a constellation, only in closer proximity) every spacecraft is controlled individually to a certain absolute orbit, the satellites of a formation are controlled relative to each other. Although the whole formation may be controlled to follow a certain absolute orbit, main focus is directed to autonomously maintain or reconfigure the relative states between the satellites.

Satellite Formation Flying (SFF) is an evolving research area that has developed in the last years from concepts and theoretical considerations to real-world applications in space. Various attempts have been undertaken to achieve distributed mission architectures reducing costs, development time, increasing failure safety and expanding possibilities for further mission concepts. Often the relative motion of cooperating spacecraft has to be reconfigured while collisions have to be precluded. Contributing to the improvement of telecommunications and Earth and deep space observation missions and others, the distribution of satellite systems flying in proximity using 3D force free formations has yet to prove itself valuable in real world applications. Since formation flying follows the principle to divide a common task among several agents, small satellites are especially suitable due to their low costs, although mainly other formation flying missions have been performed so far. The two satellites of Gravity Recovery and Climate Experiment (GRACE), a joint mission by German Aerospace Center (DLR) and National Aeronautics and Space Administration (NASA), formed a satellite formation to measure Earth's gravity field in 2002 (D'Errico, 2013, p. 547ff.). They maintained a relative distance of 220 km in the same orbit and measured the gravity field of the Earth. However, no relative position (formation) control has been performed in this mission (Kirschner et al., 2013). A further noteworthy formation is DLR's TanDEM-X mission, which extended the single-satellite TerraSAR-X mission to a two-satellite formation in 2010. The two satellites maintain a relative distance of 200 m to sub-meter accuracy and created a digital elevation model of the Earth with unprecedented resolution and precision. It is a leader-follower formation, where only the second satellite actively maintains the relative state (Ardaens et al., 2008). The two satellites Mango and Tango of the Prisma mission launched in 2010 present one of the first examples of closed-loop controlled formation flying (Alfriend et al., 2010, p. 9, 271f.), (D'Errico, 2013, p. 599ff.). The technology demonstrator mission was the first to use small satellites for formation flying (D'Amico et al., 2006). Further, the Magnetospheric Multiscale Mission (MMS) from NASA demonstrated in 2015 satellite formation flying of four satellites in a force-free tetrahedron formation (Alfriend et al., 2010, p. 10), (Sharma and Curtis, 2005, Williams et al., 2016). One that stands out is the CanX-4 & CanX-5 mission performed in 2014 for demonstrating autonomous acquisition and maintenance of a force-free formation. The mission from University of Toronto was the first research mission to form a Projected Circular Orbit (PCO) formation, specifically for two satellites, at about 50 m and100 m semi-major axis. Further, it was the first CubeSat formation performed in space, namely by using two 8 unit CubeSats with size of 20 x 20 x 20 cm (Bonin et al., 2015). From the same team in 2018 the HawkEye 360 Pathfinder mission was launched, consisting of three microsatellites with a size of 20 x 20 x 44 cm and a mass of 13.4 kg each that successfully acquired and maintained an Along Track Formation (ATF) formation with approx. 125 km inter-satellite distance (Sarda et al., 2018).

So far all flown SFF missions have only consisted of maximum three spacecraft, while performing formations in only two dimensions (except for MMS that flew four space-

craft using ground-based control in a tetrahedral formation with distances above 10 km). However, the German research institute Zentrum für Telematik e.V. (ZfT) developed a formation flying mission consisting of four CubeSats that aims at autonomous close range formation flying in three dimensions with relative distances in the order of 1 km. The Networked Pico-Satellite Distributed System Control (NetSat) mission is a technology demonstration mission that aims at developing and demonstrating efficient guidance and control to dynamically evolving formation topologies. More specifically, force free formations in three dimensions that are defined from geometric configurations like cartwheel or tetrahedron formations. The mission consists of four identical nano-satellites equipped with 350 µN electric continuous low-thrust propulsion system, a mass of approximately 3.9 kg and a size of 10 x 10 x 34 cm (Haber et al., 2020, Schilling et al., 2015). The satellites have been launched on September 28, 2020 and are currently in the Launch and Early Orbit phase (LEOP) phase.

Different control strategies for satellite formation control have been proposed including proportional derivative control (Robertson et al., 1999), Lyapunov-based control (Schaub et al., 2000a), Linear Quadratic Regulator (LQR) control (Mathavaraj and Padhi, 2021) and Adaptive Control (Queiroz et al., 2000). For continuous control applications with higher planning demands e.g. due to limited resources like power or propellant, Model Predictive Control (MPC) is of advantage due to its optimality and capability to plan a distinct time into the future. Further, its ability to consider both cost functions with flexible weights (e.g. on propellant use) and constraints is beneficial for constrained real-world systems (Dunbar and Murray, 2002). LQR has been used e.g. in the CanX-4 & CanX-5 mission (Bonin et al., 2015, Pluym and Damaren, 2006) or robust control using H_∞ synthesis has been employed for the rendezvous phase of ESA's Automated Transfer Vehicle (ATV) (Ankersen, 2010). The different control approaches show advantages and disadvantages for specific application scenarios. However, there is no suitable solution for scalable distributed space systems that is also applicable to small satellites only providing continuous low thrust.

A key element of successful formation flying missions is the ability to accurately maintain or alter inter-satellite distances. Relative orbit control was first developed for automated on-orbit rendezvous. Substantial progress in docking maneuvers was accomplished by Soyuz (Polites, 1999) and the Space Shuttle (Goodman, 2006) approaching a space station. The mathematical distinction to satellite formations is the ultimate relative state vector, which evaluates to zero for rendezvous maneuvers. This terminal approach is similar, but not equal, to formation acquisition, maintenance and reconfiguration maneuvers. As rendezvous is a well-studied field with more sophisticated analyses compared to satellite formations, some conclusions turn out to be helpful for formation flying as well. Sengupta (2007) provides an analysis of rendezvous under the influence of perturbations. In this kind of maneuvers, the relative non-spherical influence of Earth vanishes in the terminal maneuver phase

due to (almost) equal state vectors of the spacecraft. Results of relative orbit control methods used in the early formation mission Prototype Research Instruments and Space Mission Technology Advancement (Prisma) give insights into real mission data (D'Amico, 2010, D'Amico et al., 2009a). Although a double impulsive thrust scheme was used there, one can draw conclusions about the order of magnitude of thrust needed to perform reconfiguration maneuvers in a real environment. This control scheme, which initially used the Local-Vertical, Local-Horizontal (LVLH) frame, was later extended to use state vectors consisting of Relative Orbit Element (ROE) by D'Amico et al. (2011).

Difficulties of satellite formation flying emerge from the need for very accurate relative position and velocity measurements. For close-range formations, radar-based tracking and position estimation for the individual satellites becomes infeasible, since all spacecraft may return a single echo. This can be overcome with Global Navigation Satellite System (GNSS) receivers on-board the satellites. Furthermore, inter-satellite communication has to be provided to exchange relative state vectors, when the formation should maintain its configuration autonomously. It is especially challenging to accommodate all this technology in small satellites or even nano-satellites with an edge length of around 10 cm. Another design goal of small satellites designated for formation flying is to have the same ballistic coefficient for all spacecraft, which also requires the need for fuel balancing among the fleet to match the satellite masses. Otherwise, the satellites are differently influenced by atmospheric drag and may drift apart.

An important benefit of using formations instead of single large satellites in Earth observation applications is the increased temporal and spatial resolution, which distributed spacecraft can provide. This is advantageous for research fields which require in-situ measurements (e.g. magnetic field measurements) or different viewing angles at a common target at the same time (e.g. 3D images of the Earth). Beyond this, single-satellite failures or launch fails of a part of the satellites of the formation do not result in total mission loss and thus represent a form of redundancy. For many applications, the concept of graceful degradation can be used. For example, a formation resembling a Synthetic Aperture Radar (SAR) or another Earth observation or telecommunication application suffers from resolution decrease or bandwidth decrease when single satellites cease to work, but is still operable. These smaller single satellites can even be replaced at much lower cost, compared to a usually larger satellite in single-satellite missions. If demanded, a formation could also be extended in orbit with additional satellites, which increases total resolution or bandwidth depending on the application. When using many identical satellites, they may be produced in an assembly line. This could lower expenses compared to multi-million dollar singleton spacecraft and also speed up the Assembly, Integration and Test (AIT) process.

1.2 Motivation

SFF has been discussed for many years in theory, but only within the last 10 years first efforts have been undertaken to perform SFF missions. Further, only one CubeSat formation flying mission has been performed so far. However, ongoing miniaturization and the needs of new scientific and commercial applications are currently pushing the trend towards smaller satellites and towards formations with an increasing number of satellites. This process is going to continue in the future, where we can expect formations of dozens of networked and closed-loop controlled small satellites. Additionally, many possible applications e.g. in the field of Earth observation demand high position control accuracy. These developments emphasize the need for closed-loop distributed control solutions, especially for autonomous formations with a larger number of satellites within the formation. Further, since future applications will move into the direction of using many, but highly miniaturized satellites, there is a special need for solutions that are focused on or at least applicable to small satellites.

1.3 Problem Statement

The aim of this thesis is to develop a distributed Guidance, Navigation and Control (GNC) system for SFF in Low Earth Orbit (LEO). In distinction to other distributed space systems a satellite formation is understood in this work as a set of two or more spacecraft in orbit that measure and control their relative state (position and/or orientation) autonomously (without human or ground-based interaction) (see Chap. 2.2 for a detailed definition).

The distributed GNC system to be developed within this thesis should provide the theoretical basis for a range of SFF missions that require relative control of multiple spacecraft in a decentralized way. To enable this, distributed control techniques are to be applied to space-based relative navigation. Different control methods will be analyzed with respect to their applicability in this field. Among others, the well-established H_∞ robust control framework which allows for dealing with uncertainties within the spacecraft and the dynamics model will be studied. Also MPC is considered, because it shows several advantages like optimality and long-term planning capability. Further, suitable distributed control approaches will be analyzed and then combined with the selected regular control method(s).

Developed concepts and algorithms should be evaluated and validated. Thus extensive software simulations with different realistic scenarios are anticipated. Typical sensor and actuator systems on-board small satellites with their specific characteristics should be considered to guarantee the applicability of the developed methods. ZfT's NetSat mission is intended to be the primary source for the scenario definitions and acts as example application (see Chap. 2.3.2 for detailed description). The

mission consists of four nano-satellites equipped with Continuous Low-Thrust (CLT) propulsion systems that are going to form a formation. Furthermore, the implementation within NetSat should be enabled and prepared, which may provide the opportunity of in-orbit validation in the future.

1.4 Summary of Contributions

Within this thesis, three different formation control methods for continuous low-thrust satellite formation flying in LEO have been designed, implemented and evaluated. The results have been compared with each other with respect to realistic 3D formation flying scenarios of small satellites, especially CubeSats.
The three controllers are, first, a MPC implementation with specific adaptions towards the application and its requirements and constraints, second, a distributed robust controller which has been created within this thesis combining consensus methods for control of distributed systems and H_∞ robust control methodology, and last, a controller that uses plant inversion for control, and combines it with a reference governor to steer the controller towards the target on an optimal trajectory taking various constraints into account.
However, the focus of this thesis is not purely on control theory or flight dynamics, but control is considered as part of the system engineering process of typical small satellite missions. Thus, common sensor and actuator systems have been analyzed to derive their characteristics and their impacts on formation control. This has been used as basis for the design of the three control approaches, which have been adapted or all-new developed within this thesis. A comparison has been performed with respect to the given task of continuous low-thrust small satellite formation flying in three dimensions in LEO, using a specific CubeSat formation flying mission, namely NetSat, and formation flying scenarios derived from it as example.

This work enables from a control theory point of view formation control of small satellites and especially CubeSats with continuous low-thrust propulsion systems in three dimensions in LEO. Thus, the scientific contribution of this work may be summarized as the creation of novel and specific control approaches for this class of applications, which is still lacking methods withstanding the application in real-world space missions, as well as the scientific evaluation and comparison of the created methods.

1.5 Thesis Outline

This thesis is structured as follows: After this introductory chapter, a literature review follows in the State of the Art Chapter 2. It covers, among others, satellite formation flying (Chap. 2.2), nanosatellites and the NetSat mission (Chap. 2.3), furtheron required essentials (Chap. 2.4, 2.5 and 2.6), and fundamentals in control theory (Chap. 2.7, 2.8 and 2.9). Next, satellite formation control is addressed in

Chap. 3 covering *MPC-Based Formation Control* (Chap. 3.2) followed by *Distributed Robust Consensus-Based Control* (Chap. 3.3) and *Plant Inversion-Based Lyapunov Control Combined with a Reference Governor* (Chap. 3.4). Chap. 4 presents typical sensor and actuator systems for SFF (Chap. 4.2 and 4.3), followed by the composition of an overall sensor and actuator system for SFF (Chap. 4.4) and its impacts on navigation and control for SFF and especially for the NetSat formation flying mission (Chap. 4.5). Chap. 5 provides an evaluation of the developed control methods. First, the simulation framework is described (Chap. 5.2), second the scenario definitions are given (Chap. 5.3), and third, the results are presented and discussed (Chap. 5.4). The last section shows the preparation for and outlook to in-orbit testing within NetSat (Chap. 5.5). Chap. 6 sums up the performed work and gives a conclusion.

Chapter 2

State of the Art

2.1 Overview

This chapter summarizes the relevant state of the art in the field of distributed Guidance, Navigation and Control (GNC) for SFF missions and is the basis of the work presented later.

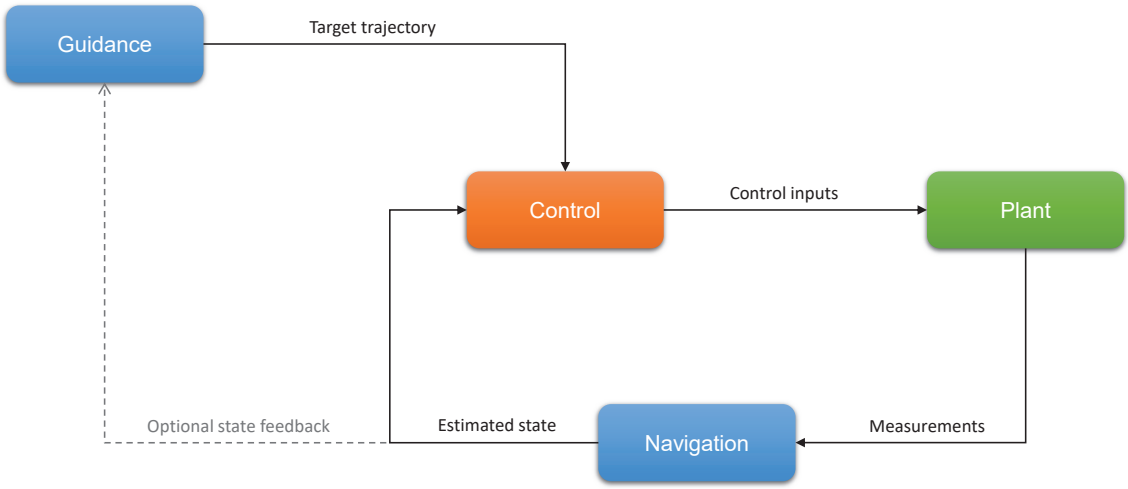


Figure 2.1: Block diagram of a generic GNC structure.

Guidance, Navigation and Control GNC is a branch of engineering that deals with the design of systems to control the movement of vehicles. Guidance refers to the definition of the desired path of travel (the "trajectory") from the vehicle's current location to a designated target (state), as well as desired changes in velocity, rotation and acceleration for following that path. Navigation refers to the determination, at a given time, of the vehicle's location and velocity (the "state vector") as well as its attitude (defining its current state). Control refers to the manipulation of the forces, by way of steering controls, thrusters, etc., needed to track guidance commands while maintaining vehicle stability (control) (Wikipedia Contributors, 2020a). A general

GNC system can be illustrated as shown in Fig. 2.1. The following exact definition has been accepted by several agencies including NASA and ESA:

> "**Navigation** is the process used to find the present and imminent future position, orbit and orientation of a spacecraft using a series of measurements.
>
> **Guidance** is the process of defining a path to move a spacecraft from one point to another or from one orientation to another.
>
> **Control** is the process to maintain a spacecraft within the prescribed path and attitude." (CCSDS 500.0-G-3, 2010, p. 2-4)

The following sections present general definitions of SFF, nano-satellite navigation and control characteristics, coordinate frames, models for relative motion, typical satellite formation topologies and control theory fundamentals up to distributed and formation control.

2.2 Satellite Formation Flying

Arbitrary distributed satellite systems can be divided into Rendezvous and Docking (RvD), SFF, constellations and swarms. Fig. 2.2 shows these types distinguished with respect to control accuracy and inter-satellite separation. Even though all of these areas are of interest, this work focuses on SFF. SFF is at the edge of what is doable with nano-satellites and with satellites in general in terms of control accuracy and in terms of autonomy and complexity of the overall system. High requirements are imposed due to the number of involved satellites and the high relative position control demands for acquiring and maintaining a formation.

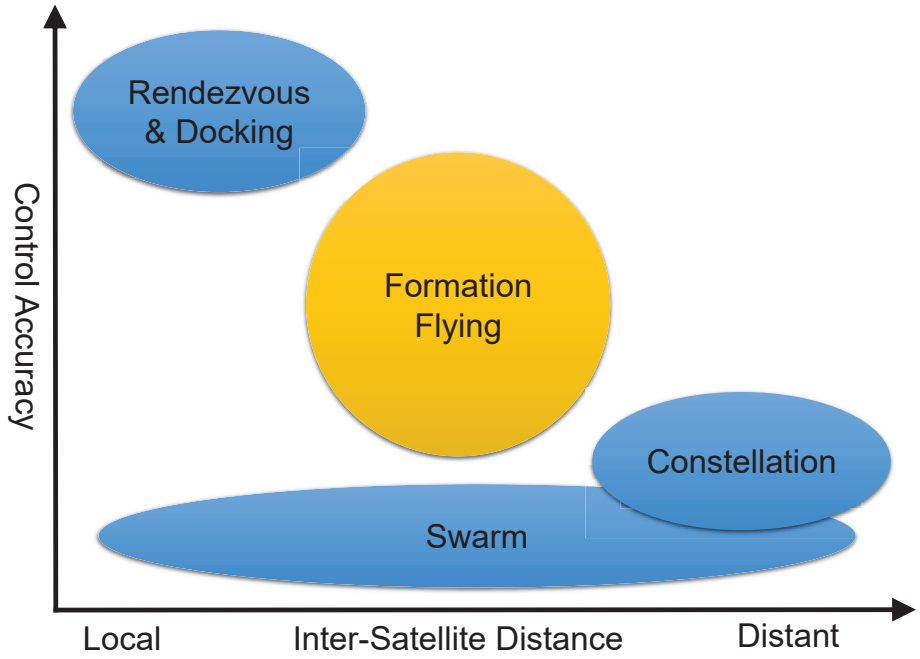


Figure 2.2: Distributed systems in space can be categorized with respect to their inter-satellite separation and their requirements on control accuracy. (Figure based on Gill (2011)).

2.2.1 Definition

Before addressing GNC concepts for Satellite Formation Flying, we should first discuss the question, what exactly can be considered as SFF. Further, a clear distinction of formations towards other distributed space systems like constellations etc. is required. There are different definitions, some more open, some more restrictive. Thus it is not always clear, if a mission forms a formation or rather a constellation or swarm. This depends on the definition. Here we will discuss two definitions and justify the definition chosen for this thesis.

Formation Flying Definition by Kyle T. Alfriend

As proposed by NASA's Goddard Space Flight Center (GSFC) and stated by Kyle T. Alfriend, formation flying can be defined as

> "The tracking or maintenance of a desired relative separation, orientation or position between or among spacecraft." (Alfriend et al., 2010, p. 1)

This definition does not explicitly require control. Also missions only performing relative measurements of position or attitude are included here. Further no common objective, which is the understanding of a formation from an application point of view is mentioned here. Thus this definition is quite broadly defined.

Formation Flying Definition by ESA

During the 2nd International Formation Flying Symposium, 2004 in Washington DC, USA this definition has been agreed upon by official representatives of the following agencies: French National Centre for Space Studies (CNES), ESA, Jet Propulsion Laboratory (JPL), NASA's GSFC, Office National d'Etudes et de Recherches AÃ©rospatiales (ONERA), Canadian Space Agency (CSA), Japan Aerospace Exploration Agency (JAXA), British National Space Centre (BNSC) (now United Kingdom Space Agency (UKSA)), Air Force Research Laboratory (AFRL).

- "The mission consists of 2 or more spacecraft.
- The spacecraft states are directly coupled such that changing the state of one spacecraft affects the state of all other spacecraft. This includes possible state elements from optical delay lines etc.
- The relative position and velocity between the spacecraft are controlled and possibly at certain parts also the relative attitudes.
- The spacecraft are moving on quasi coplanar orbits or perhaps Lagrange points.
- The spacecraft are in close proximity, which means typically below a few km separation where the relative motion is in a linear domain. (though some proposals have rather large distances)
- A plane is defined for the inter spacecraft positions with an arbitrary orientation in space and with respect to a possible local orbital frame. Spacecraft do not all have to be in that plane in their nominal position.
- The GNC requirements are typically high to very high." (Ankersen, 2004)

This definition is more comprehensive, precise and narrow compared to the definition by Alfriend. Control is explicitly required (according to the second and third bullet point). However a common objective for a formation is not required.

Definition Used in this Thesis

Based on the definition of the ESA (Ankersen, 2004), a more concise definition focusing on the crucial aspects of this definition and including a common objective is formulated:

> A set of two or more spacecraft in space that measure and control their relative state (position and/or orientation) autonomously (without human or ground-based interaction) to perform a common goal.

This definition is, because of its brevity, easier to understand and use, while still covering all important aspects, especially including the common objective within a formation. Thus this formulation is proposed and preferred within this thesis.

Distinction From Multi-Satellite Systems

To distinguish SFF from other multi-satellite systems in space, few additional definitions taken from (Alfriend et al., 2010, p. 1) are presented here:

> **Distributed space system:** "An end-to-end system including two or more space vehicles and a cooperative infrastructure for science measurement, data acquisition, processing, analysis and distribution."

> **Constellation:** "A constellation is a collection of space vehicles that constitutes the space element of a distributed space system."

> **Virtual platform:** "A virtual platform is a spatially distributed network of individual vehicles collaborating as a single functional unit, and exhibiting a common system-wide capability to accomplish a shared objective."

Taking these definitions into account, formations are part of distributed space systems and also a subset of constellations.

2.2.2 Overview of Formation Flying Missions

Only few formation flying missions (in the sense of the definition of this thesis) have been performed until present. More are upcoming or planned. A short, though not comprehensive overview is given here as an introduction. A comprehensive overview on distributed space missions with an emphasis on formations is given e.g. by D'Errico (2013).

Performed Formation Flying Missions

- **TerraSAR-X / TanDEM-X:** The mission consists of two satellites sharing the same orbit with a defined relative distance of 200 m. The formation has been established in 2010 and performs radar measurements to create an altitude map of the Earth (D'Errico, 2013, p. 387ff.).

- **Prisma:** The Prisma mission consists of the two satellites Mango and Tango. The Swedish mission was launched in 2010 and is a technology demonstration for relative navigation systems like GPS and radio-based ranging as well as for SFF of two satellites (Alfriend et al., 2010, p. 9, 271f.), (D'Errico, 2013, p. 599ff.).

- **ELISA:** The french ELectronic Intelligence by SAtellite (ELISA) mission aims at demonstrating the ability to locate radar equipment from space and to collect information on the technical characteristics of the observed radar equipment. It consists of four microsatellites which were launched in late 2011 into a LEO. The satellites are placed on two orbital planes such that they form two quasi-isosceles triangles with a base length of 100 km ±15 km. This is achieved by biweekly correction maneuvers (Delmas et al., 2018).

- **GRACE:** The joined mission of NASA and DLR aimed at measuring Earth's gravitational field with two satellites in an orbit of 500 km altitude and an along-track separation of 200 km. For inter-satellite ranging an microwave system was used. However, the relative distance or position was not actively controlled. The satellites were launched in 2012 and decommissioned in 2017 (D'Errico, 2013, p. 547ff.). The successor mission GRACE Follow-On (GRACE-FO) also featuring a laser ranging interferometer was launched in 2018 (Kornfeld et al., 2019).

- **CanX-4 & 5:** The Canadian mission from University of Toronto's Space Flight Laboratory consists of two identical CubeSats of 8U size. It was launched in 2014 and demonstrated relative navigation and for the first time formation flying of two CubeSats in space (Bonin et al., 2015).

- **MMS:** The four-satellite mission aims at measuring Earth's magnetosphere and demonstrated a tetrahedron formation for the first time in orbit. It reached a minimum relative distance of about 10 km. The satellites were launched in an Highly Elliptical Orbit (HEO) by NASA in 2015. The tetrahedron formation has to be maintained within a defined region during each orbit, so regular formation maintenance maneuvers have to be performed (Alfriend et al., 2010, p. 10), (Williams et al., 2016).

- **HawkEye 360 Pathfinder:** The mission that inherited from CanX-4 & 5 consists of three satellites with a 20 cm × 20 cm × 44 cm form factor with a mass of 13.4 kg. It was launched in late 2018 and successfully acquired and maintained

an ATF formation with approximately 125 km inter-satellite distance (Sarda et al., 2018).

Upcoming Formation Flying Missions

- **NetSat**: The NetSat mission is going to consist of four CubeSats with a size of 3 U. It aims at demonstrating 3D formation flying with CubeSats in a circular LEO. The mission of ZfT has been launched in late 2020 and is currently in LEOP. More details are presented in Chap. 2.3.2.
- **PROBA-3**: The ESA mission PROBA-3 is going to consist of two satellites forming a close-range formation on elliptical orbits. Its primary mission is solar coronagraphy which is enabled by precisely aligning the two satellites on a line with the sun such that one satellite occults the disc of the sun. It has been delayed and now is scheduled for a launch in 2022 (ESA, 2020). Initially, it was planned as preparation for the ESA missions XEUS and DARWIN (Alfriend et al., 2010, p. 9f.).
- **CloudCT**: The CloudCT mission is a cooperation the Technion Israel Institute of Technology, the Weizmann Institute of Science and ZfT and aims at performing tomography of clouds by simultaneous observations of a formation of 10 CubeSats. It is scheduled for launch in 2022 (Schilling et al., 2019).

2.3 Pico- and Nanosatellite Navigation and Control

The ongoing miniaturization in electronics, especially the development of smaller sensors, actuators and processors, higher computational capabilities of embedded systems and decreasing costs for satellite launches led to a rise of small satellites. Among them, especially pico- and nanosatellites evolved in recent years and strongly increased in number. Picosatellites are defined by their weight being in the range of 0.1 - 1 kg, whereas nanosatellites range from 1 - 10 kg. Tab. 2.1 gives a general classification of satellites by their mass. Nanosatellites are the smallest satellites that are currently utilized in commercial applications.

Table 2.1: Classification of satellites with respect to their mass (Gill, 2011).

Class	Mass	Example
Large satellite	> 1000 kg	TanDEM-X (Astrium)
Medium-sized satellite	500 - 1000 kg	Iridium (Iridium LLC)
Minisatellite	100 - 500 kg	GRACE (Astrium)
Microsatellite	10 - 100 kg	BIRD (DLR)
Nanosatellite	1 - 10 kg	Delfi-C3 (TU Delft)
Picosatellite	0.1 - 1 kg	BEESAT (TU Berlin)
Femtosatellite	< 0.1 kg	RyF3 Ex (Ryerson)

Most of the pico- and nanosatellites follow the CubeSat standard (cf. (California Polytechnic State University, 2020)) which defines common dimensions and mechanical interfaces. CubeSats share a common size of multiples of 10 cm $\times$ 10 cm $\times$ 10 cm, which a so called unit (1U), and are common in sizes of 1U, 3U, 6U, 16U, though other sizes are also possible. The standardization allows for faster and cheaper integration into launchers and easier reuse of subsystems.

2.3.1 Navigation and Control Capabilities and Limitations

Due to their limited size and weight, pico- and nanosatellites typically show limitations with respect to their control and navigation capabilities. They appear in many different ways:

- **Electric power:** Since less than 60 cm^2 per satellite for picosatellites and few hundred cm^2 for nanosatellites are available for solar cells, the total available electric power is in the order of 1 W for picosatellites and 10 W for nanosatellites.
- **Thrusting capability:** The limited size and weight allows only for small thrusters (typically electrical or cold gas) and limited amount of propellant. Thus the available thrust typically is in the order of µN and the total ΔV is in the order of few tens of $\mathrm{m\,s^{-1}}$ for picosatellites and few hundred of $\mathrm{m\,s^{-1}}$ for nanosatellites.

- **Navigation sensors:** Limited space and power prohibited the use of active navigation sensors in the past. However, small GNSS receivers as well as cameras became available recently. Thus position determination in the order of meters can be reached. However, the sensor accuracy is limited in comparison to larger satellites (see also Chap. 4.2).
- **Communication capability:** Due to power limitations and also limited antenna size, the communication range for Inter-Satellite Link (ISL) is limited as well as the transmission bandwidth.
- **Computing capability:** Available power and space also limit the applicable computing units. Only micro-controllers and recently low power Field Programmable Gate Arrays (FPGAs) are feasible.

In the following sections, examples for typical requirements derived from the NetSat mission are presented.

2.3.2 NetSat - A Four Nanosatellite Formation Flying Mission

From UWE to NetSat

The German research institute Zentrum für Telematik e.V. (ZfT) together with University of Würzburg are successfully engaging into small satellite development and operation since 2005 (Schilling et al., 2015). The University of Würzburg Experimental (UWE) satellite line was kicked off with the UWE-1 satellite, which has been the first German CubeSat in space (Barza et al., 2006). UWE-1 demonstrated Internet-based communication in space and tested novel solar cells in LEO conditions (Schmidt et al., 2007). UWE-2 followed in 2009 with the aim to demonstrate and verify new attitude determination and control methods as well as orbit determination using a GPS receiver in orbit (Schmidt et al., 2008).
With UWE-3 a complete redesign of the UWE satellite bus was performed aiming at high redundancy and robustness within a picosatellite. In addition, it was the first satellite within the UWE line that demonstrated active attitude control in orbit (Busch and Schilling, 2012, Busch et al., 2014, Schilling et al., 2015). After its launch in 2013 UWE-3 remained operational in orbit for more than 6 years demonstrating its robust design. Fig. 2.3 shows a UWE-3 engineering model.
The newest satellite within the UWE family is UWE-4, which was launched in December 2018. It is equipped with a low-thrust electrical propulsion system (Nano Field-Emission Electric Propulsions (NanoFEEPs), (Bock et al., 2014)) and aims at demonstrating orbit control (Bangert et al., 2016). In 2020 UWE-4 demonstrated the first successful orbit control of a 1U CubeSat using electric propulsion (Kramer et al., 2020).
After UWE-4, NetSat follows as the next CubeSat mission. NetSat is a small satellite formation flying mission performed at ZfT. The formation flying aspects as well as the development of the satellites was funded by the European Research Council (ERC) as Advanced Grant, whereas the launch of the satellites was funded by the

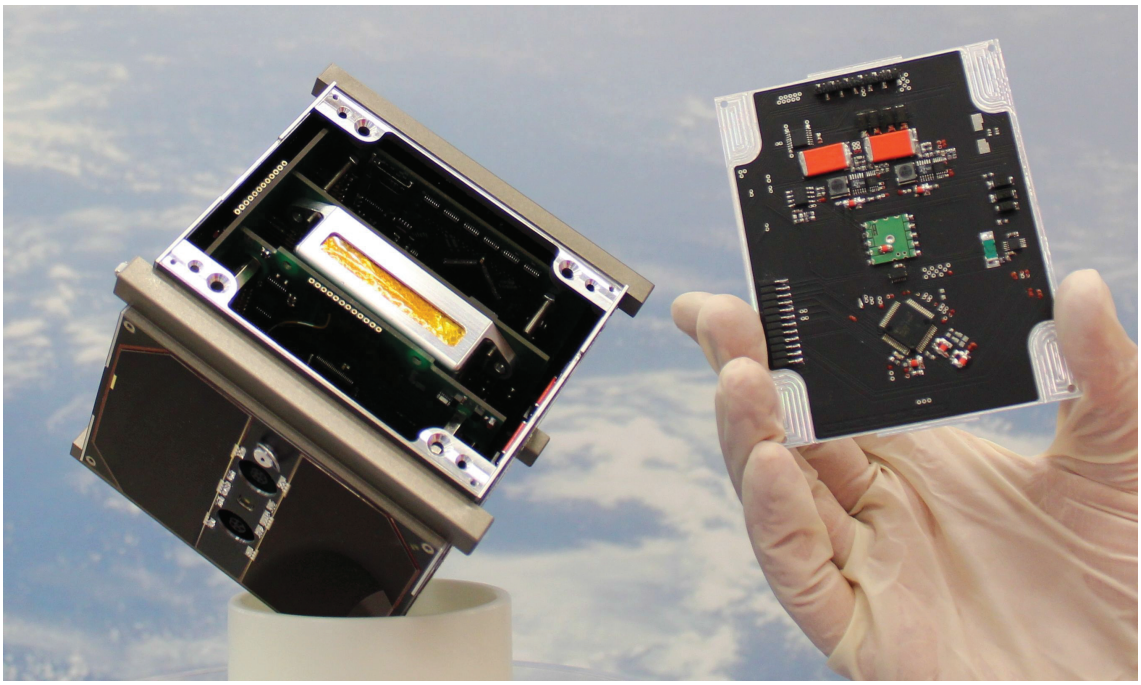


Figure 2.3: UWE-3 CubeSat engineering model with one removed side panel. Redundant electric components and circuits are clearly visible. (Image courtesy: Busch and Schilling (2013))

Bavarian State Ministry of Economic Affairs, Regional Development and Energy. The technology demonstration mission aims at enabling 3D formations (e.g. cartwheel or tetrahedral formations, cf. Chap. 2.6) with CubeSats (Haber et al., 2020, Schilling et al., 2015). The four identical nanosatellites are 3U CubeSats with a size of about 30 cm × 10 cm × 10 cm and a mass of about 3.9 kg each. NetSat was launched on September 28, 2020 into a nearly circular LEO at 560 km altitude and as of now is in LEOP.

Mission Objectives

The main objective of the NetSat mission is to develop control approaches for SFF using low-thrust electric propulsion and to demonstrate with the help of the developed control approaches autonomous formation flying with nanosatellites in three dimensions (cf. Fig. 2.4). This requires relative orbit and position determination as well as adequate control techniques. The project should demonstrate the ability to closed-loop control a networked formation within the constraints imposed by nanosatellites. These constraints are, among others, limited volume and mass, a predefined mechanical interface, available budget and the limited availability of miniaturized hardware components. Further, required networked control strategies and algorithms are developed, implemented and tested. Besides, NetSat will be the first formation that uses continuous low-thrust (using electric propulsion). Formation

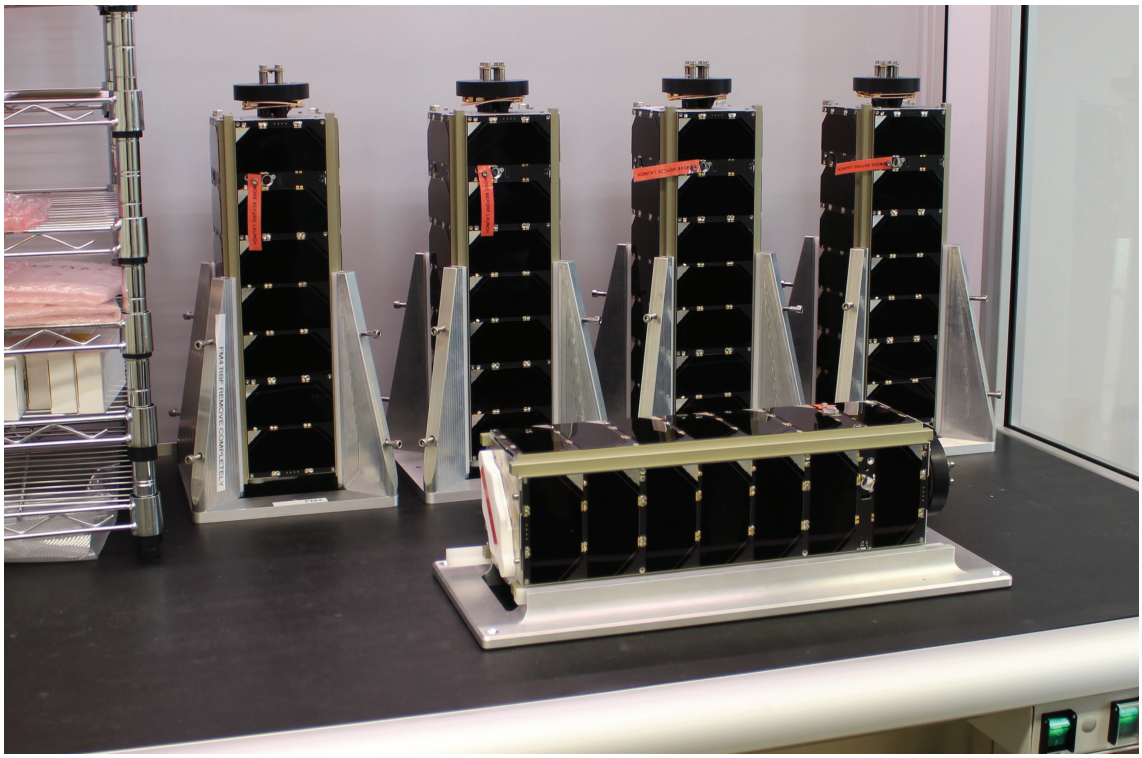


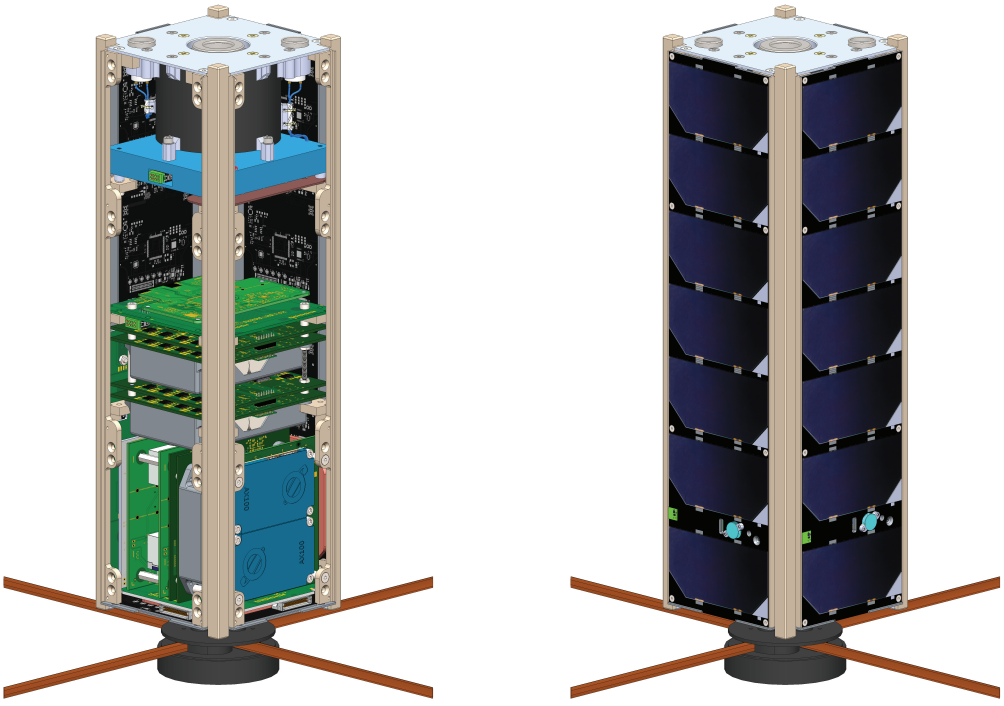
Figure 2.4: The four NetSat satellites (flight models, back) together with the engineering model (front) on the day of delivery to the launch provider. (Image courtesy of the author.)

maintenance and reconfiguration maneuvers with different formation types starting from along-track formation while increasing complexity up to 3D formation types are planned. Besides the basic string of pearl formation also (projected) circular formations, different 3D cartwheel formations and tetrahedral formations are of interest. In the meantime the relative distance should decrease from about 100 km down to about 5 km (maximum distance between two satellites within one orbit). A detailed mission plan with all planned formations, their duration, relative distances and order is developed within this work and presented in Chap. 5.5.1. The scenarios also act as primary reference for this work. The detailed scenario definitions used in this thesis are presented in Chap. 5.3.

Though no primary scientific or commercial payload is planned, contributions to the fields of distributed formation control, relative navigation, on-board autonomy, ISL and miniaturized Attitude and Orbit Control Systems (AOCSs) are expected. Besides, different application scenarios have been evaluated and will be presented shortly at the end of this chapter.

Satellite Characteristics

The four identical NetSat satellites are equipped with magnetorquers, reaction wheels and thrusters to enable for full attitude and orbit control capability. NetSat follows the CubeSat standard (California Polytechnic State University, 2020) with an approximated size of about 34 cm × 10 cm × 10 cm and a mass of about 3.9 kg each. Each of the four 3U CubeSats features an electric propulsion thruster that allows for precise position control. Thus, the satellites have full 6 Degrees of Freedom (DoF) navigation capability. As electric propulsion system an IFM NanoThruster by ENPULSION GmbH (see Chap. 4.3.2 for a detailed description) has been selected. Fig. 2.5 shows the Computer-Aided Design (CAD) model of a NetSat satellite with the thrust system integrated in the upper part. For attitude determination, sun sensors, magnetic field sensors and Inertial Measurement Units (IMUs) are integrated on NetSat. Position determination is handled by four GPS receivers per satellite. As backup solution for position determination, retro-reflectors are placed on each satellite, but also Two Line Elementss (TLEs) can be provided from ground. Inter-satellite link is based on omnidirectional Ultra High Frequency (UHF) communication. Further details on the selected sensor and actuator systems together with the depending navigation and control requirements are presented in Chap. 4.



(i) NetSat CAD with front panels removed (ii) NetSat CAD fully assembled

Figure 2.5: CAD drawing of the design of the NetSat CubeSats. Thrusters can be seen in the upper part of the open satellite.

NetSat was launched on September 28, 2020 on a Soyuz rocket together with 18 other satellites into a nearly circular LEO at 560 km altitude. As typical for most of the CubeSat missions, NetSat was a piggyback payload to the main payload of the rocket, namely three Gonetz-M satellites. All four NetSat satellites have been released on the same orbit. Though, slight deviations in the orbit parameters are a result of the inaccuracies of the deployment mechanism and orbit perturbations. The orbit parameters of all four NetSat satellites are given in Tab. 2.2. These orbits are used as an absolute reference for the simulations within this work.

Table 2.2: NetSat Keplerian elements at 11.11.2020.

Parameter	NetSat-1	NetSat-2	NetSat-3	NetSat-4
Semi-major axis	6946.0 km	6946.0 km	6945.8 km	6945.7 km
Eccentricity	0.002 205	0.002 223	0.002 223	0.002 229
Inclination	97.660°	97.661°	97.660°	97.661°
Argument of perigee	89.005°	89.145°	88.103°	87.653°
Right ascension of ascending node	−110.64°	−110.240°	−110.241°	−110.238°
True anomaly	−89.005°	−89.145°	−88.103°	−87.653°

Possible Applications

Although NetSat being a technology demonstrator mission, there are applications to which it or a similar mission could be applied.

Earth's magnetic field measurements benefit from satellite formations (esp. three-dimensional), since simultaneous measurements at different, precisely measured positions in LEO allow for determination of geomagnetic gradients, so-called gradiometry (cf. Fig. 2.6). Thus, conclusions about Earth's lithospheric field can be drawn. A mission concept based on a CubeSat formation is presented by Nogueira et al. (2015). In other regions of Earth's magnetic field, a 3D satellite formation can be used to study the micro-physics of magnetic reconnection as it is performed by the MMS (Alfriend et al., 2010, p. 10), (Sharma and Curtis, 2005, Williams et al., 2016). In both cases it is important to maintain a 3D formation with a certain accuracy and to measure the relative positions with very high accuracy.

Another application is photogrammetric ash cloud monitoring (cf. Fig. 2.7). On average, at least one large volcano eruption occurs per year where volcanic ash reaches the stratosphere. This can threaten the safety of air traffic even at large distances from the erupting volcano. Satellite formations can be beneficial by providing high spatial and temporal resolution. The satellites in a formation (e.g. a group of four) can observe the same area from slightly different angles, thus being able to compute a 3D model of the observed ash clouds, and therewith provide information on the structure and dispersion of volcanic ash. This requires of course the ability to measure and control the (absolute and relative) position of the satellites and the (absolute and

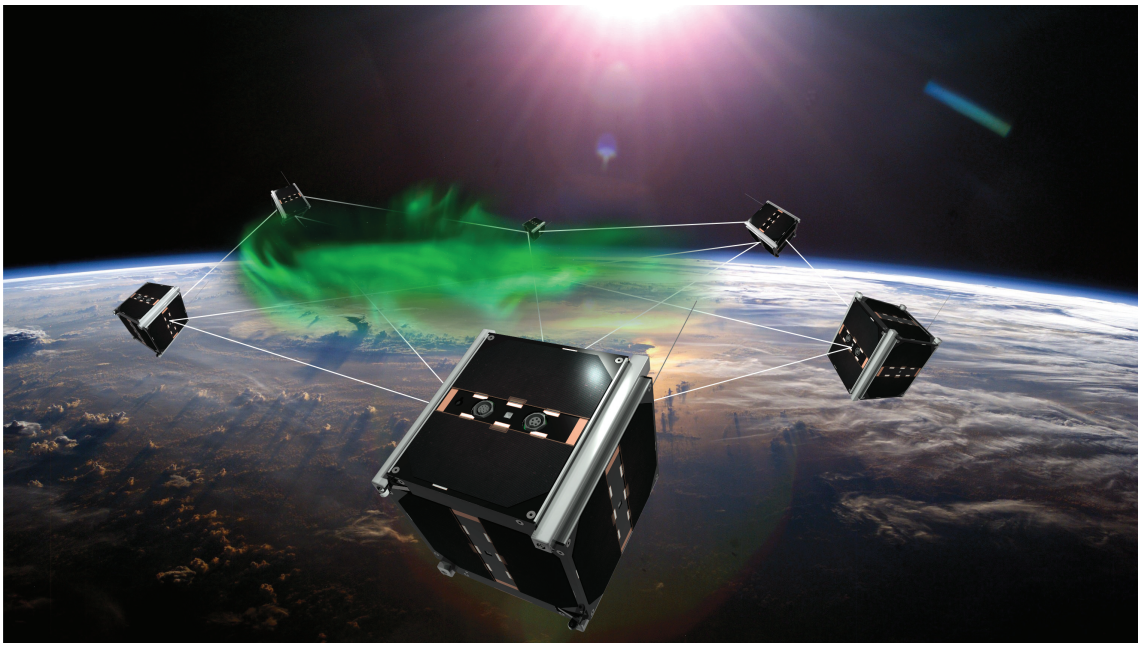


Figure 2.6: Concept scheme of distributed magnetic field measurements using a CubeSat formation.

relative) orientation with very high accuracy. The use of more than one formation adds temporal resolution by being able to observe the scene at different points in time. By using small satellites or CubeSats the number of satellites in a formation or the number of formations can be increased (and thus the temporal resolution) by keeping costs low (Nogueira et al., 2016). This is planned to be implemented in the Telematics Earth Observation Mission (TOM), a mission which is a successor mission to NetSat performed at ZfT (Schilling et al., 2018) and scheduled for launch in late 2021.

These are only two exemplary applications among others in which small satellite formations are especially beneficial. However, small satellite formations are expected to be applied to a wider range of applications in the future.

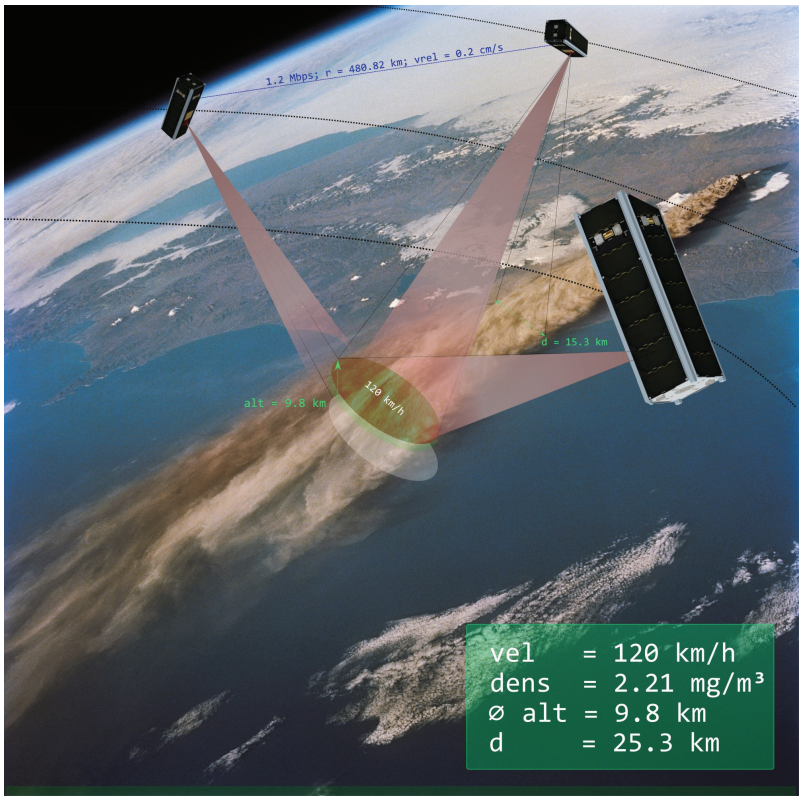


Figure 2.7: Volcanic ash cloud monitoring as planned within the satellite mission TOM. Multiple CubeSats fly in a formation to observe volcanic ash clouds from different angles.

2.4 Coordinate Frames

Coordinate frames define the mathematical space in which SFF and the state of satellites is described. Such coordinate frames can be distinguished by their inherent setup into two groups, Cartesian and orbital frames. This section will introduce these frames and the reasoning which one to use in a particular case. Furthermore, each group is divided into global frames, which describe the absolute state of a satellite with respect to Earth, and local frames, which describe the relative states between multiple satellites of a formation.

In this work the time derivative of a vector will be denoted by ($\dot{}$) and the second time derivative by ($\ddot{}$). Vectors will be denoted by bold lower case letters ($\boldsymbol{v}$) and matrices by bold upper case letters ($\boldsymbol{M}$). A six-field column vector consisting of concatenated position ($\boldsymbol{r}$) and velocity ($\boldsymbol{v}$) vectors of a certain satellite will be called $\boldsymbol{pv}$ vector:

$$\boldsymbol{pv} = \begin{pmatrix} \boldsymbol{r} \\ \boldsymbol{v} \end{pmatrix} \tag{2.1}$$

2.4.1 Cartesian Frames

Both, an inertial and a local coordinate system are necessary to describe the relative motion of two or more satellites. The state vector in the inertial system, which is fixed to the Earth, describes the absolute motion of a satellite with respect to Earth center. The local frame is fixed to the formation and thus able to describe the relative distances and velocities between the satellites. With respect to an initial frame, the local frame itself is moving.

2.4.1.1 Inertial Frame

The inertial frame used throughout this work is the Earth-Centered Inertial (ECI) coordinate system. It will be denoted with the symbol $\mathscr{I}$. This geocentric frame is defined through the equatorial plane of the Earth with $\hat{\boldsymbol{X}}$ pointing to the vernal equinox, $\hat{\boldsymbol{Z}}$ being normal to the fundamental plane and pointing to the north pole and $\hat{\boldsymbol{Y}}$ completing a positive triad. The state vector of a satellite in this frame is defined by

$$\boldsymbol{x} = \begin{pmatrix} x & y & z & v_x & v_y & v_z \end{pmatrix}^T = \begin{pmatrix} x & y & z & \dot{x} & \dot{y} & \dot{z} \end{pmatrix}^T \tag{2.2}$$

with position $\boldsymbol{r} = (x \;\; y \;\; z)^T$, velocity $\boldsymbol{v} = (\dot{x} \;\; \dot{y} \;\; \dot{z})^T$ and distance from the center of the Earth $r = ||\mathbf{r}|| = \sqrt{x^2 + y^2 + z^2}$. The angular momentum of the satellite, which is normal to the position and the velocity vector, is defined by (Chobotov, 2002):

$$\boldsymbol{h} = \boldsymbol{r} \times \dot{\boldsymbol{r}} \tag{2.3}$$

The reference for the ECI frame will be the J2000 (or EME2000) definition. It corresponds, in accordance with the definition above, to Earth's mean equatorial plane at 12:00 Terrestrial Time (TT) on 1$^{\text{st}}$ January 2000 (Montenbruck and Gill, 2012).

2.4.1.2 Local Frame

The local frame $\mathscr{L}$ of satellite used within this thesis is the so-called Local-Vertical, Local-Horizontal (LVLH) coordinate frame. It has its origin in the satellite's center of mass. It is Cartesian, rectangular and dextral rotating. The fundamental plane of the frame is defined through the orbital plane of the satellite. There are different definitions of the directions of the axes. The precise definition of the local frame $\mathscr{L}$ that is used in this work is the LVLH coordinate system as defined by ESA and the International Space Station (ISS) program (Space Station Control Board, 2008). Following (Fehse, 2003, p. 31f.), the $\hat{\boldsymbol{e}}_{\boldsymbol{z}}$ is directed from the spacecraft radially outward to the center of mass of Earth, $\hat{\boldsymbol{e}}_{\boldsymbol{y}}$ is normal to the fundamental plane, negative in the direction of the instantaneous angular momentum vector, and the $\hat{\boldsymbol{e}}_{\boldsymbol{x}}$ completes the right-handed coordinate frame and is in the direction of the orbital velocity vector (see Fig. 2.8). The rotating frame moves with the satellite along its

orbit. Sometimes it is also referred to as Radial-Tangential-Normal (RTN) frame.

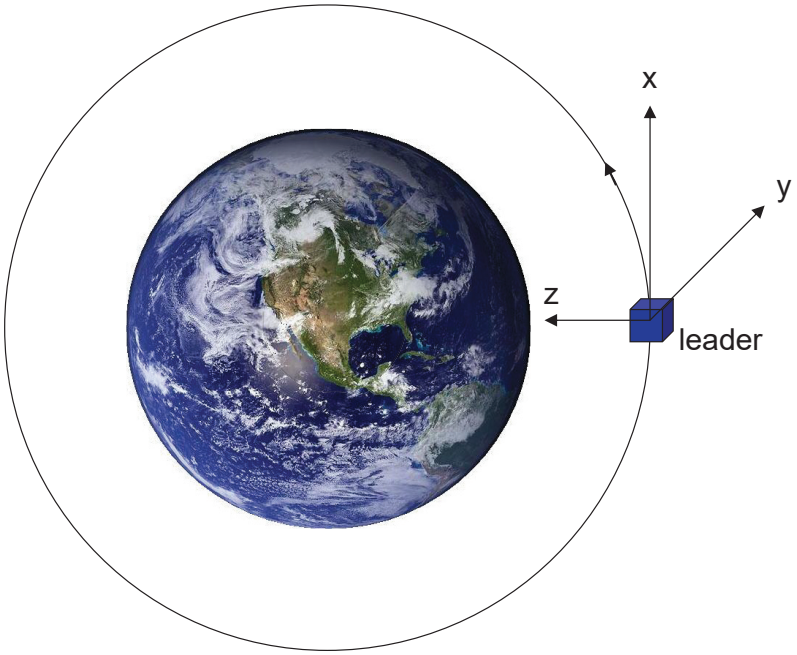


Figure 2.8: The LVLH coordinate frame is centered in a (leader) spacecraft (definition based on (Fehse, 2003, p. 32)).

When a follower (also called deputy or chaser) satellite is viewed in the leader satellite's $\mathscr{L}$ frame, the frame is called Euler-Hill (EH) frame $\mathscr{H}$ (or Hill frame for brevity). Although the individual $\mathscr{L}$ and the $\mathscr{H}$ frame of a follower are very similar in terms of orientation for close formations, this distinction proves to be pretty useful for later analysis. While the $\mathscr{L}$ frame of a follower depicts the true transformation with respect to the $\mathscr{I}$ frame, the satellite's state in the $\mathscr{H}$ frame will always have small errors induced through using a reference at a slightly different position (e.g. the position offset between leader and follower). However, all satellites of a formation share the same $\mathscr{H}$ frame, which is essential for relative motion models. By definition of the EH frame, $\mathscr{L} \hat{=} \mathscr{H}$ holds for the (virtual) leader and it will always be located at $\boldsymbol{r}^{\mathscr{H}} = (0\ \ 0\ \ 0)^T$. When mapping between ECI, orbital elements (cf. Chap. 2.4.2) and the EH frame, the EH frame only has a satisfactory level of accuracy for low-eccentric leader orbits ($e << 1$) and close formations (relative distances between the satellites smaller than $\sim 100\,\text{km}$). (A detailed analysis of the accuracy of the EH frame and the Hill-Clohessy-Wiltshire (HCW) equations as described in Chap. 2.5.3.1) is given in Chap. 2.5.3.3.) Thus, whenever mapping into the EH frame is applied, the just mentioned constraints are assumed to be fulfilled (Alfriend et al., 2010).

An alternative definition of the LVLH coordinate frame by Alfriend et al. (2010) is presented in A.1.

Virtual Leader

Instead of defining the $\mathscr{H}$ in a way that the coordinate frame's origin is on a satellite of the formation, also a virtual leader can be defined as origin and the positions of all satellites in the formation can be given relative to this virtual center.

Transformation Between Earth-Centered Inertial and Euler-Hill Frame

Let $\boldsymbol{r_0}$ be the position vector of the leader satellite and $\boldsymbol{v_0}$ its velocity. The origin of the $\mathscr{H}$ frame viewed in the $\mathscr{I}$ frame is therefore at $\boldsymbol{r_0}$. The rotation vector between the two frames will be denoted by $\boldsymbol{\Omega}$ (with $\dot{\boldsymbol{\Omega}}$ being the angular velocity of $\mathscr{H}$ relative to $\mathscr{I}$). The $\boldsymbol{pv}$ vector of another satellite (called follower spacecraft) consists of $\boldsymbol{r_1}$ and $\boldsymbol{v_1}$. The 3×3 direction cosine matrix $\boldsymbol{C}$ between the $\mathscr{H}$ and the $\mathscr{I}$ frame can be computed as follows (Nicholas, 2013)

$$\hat{\boldsymbol{e}}_{\boldsymbol{x}}^{\mathscr{L}} = \frac{\boldsymbol{r}_{\boldsymbol{0}}^{\mathscr{I}}}{||\mathbf{r_0}||^{\mathscr{I}}} \tag{2.4a}$$

$$\hat{\boldsymbol{e}}_{\boldsymbol{z}}^{\mathscr{L}} = \frac{\boldsymbol{r}_{\boldsymbol{0}}^{\mathscr{I}} \times \boldsymbol{v}_{\boldsymbol{0}}^{\mathscr{I}}}{||\mathbf{r_0} \times \mathbf{v_0}||^{\mathscr{I}}} \tag{2.4b}$$

$$\hat{\boldsymbol{e}}_{\boldsymbol{y}}^{\mathscr{L}} = \hat{\boldsymbol{e}}_{\boldsymbol{z}}^{\mathscr{L}} \times \hat{\boldsymbol{e}}_{\boldsymbol{x}}^{\mathscr{L}} \tag{2.4c}$$

$$\boldsymbol{C} = \left[\begin{array}{ccc} \hat{\boldsymbol{e}}_{\boldsymbol{x}}^{\mathscr{L}} & \hat{\boldsymbol{e}}_{\boldsymbol{z}}^{\mathscr{L}} & \hat{\boldsymbol{e}}_{\boldsymbol{y}}^{\mathscr{L}} \end{array} \right]^T \tag{2.5}$$

Note that the direction cosine matrix $\boldsymbol{C}$ can be used to translate ECI coordinates to any satellite's LVLH frame by using its corresponding $\boldsymbol{pv}$-vector. The position and velocity difference between the leader and the follower satellite is

$$\boldsymbol{pv_d} = \begin{pmatrix} \boldsymbol{r_d} \\ \boldsymbol{v_d} \end{pmatrix} = \begin{pmatrix} \boldsymbol{r_1} - \boldsymbol{r_0} \\ \boldsymbol{v_1} - \boldsymbol{v_0} \end{pmatrix} \tag{2.6}$$

For the follower's $\boldsymbol{pv}$ vector in the $\mathscr{H}$ frame follows with the theorem of rotating reference frames

$$\boldsymbol{p} = \begin{pmatrix} x \\ y \\ z \end{pmatrix} = \boldsymbol{r}_{\boldsymbol{d}}^{\mathscr{H}} = \boldsymbol{C}\boldsymbol{r}_{\boldsymbol{d}}^{\mathscr{I}} \tag{2.7a}$$

$$\boldsymbol{v} = \dot{\boldsymbol{p}} = \begin{pmatrix} v_x \\ v_y \\ v_z \end{pmatrix} = \boldsymbol{v}_{\boldsymbol{d}}^{\mathscr{H}} = \boldsymbol{C}\boldsymbol{v}_{\boldsymbol{d}}^{\mathscr{I}} + \boldsymbol{r}_{\boldsymbol{d}}^{\mathscr{H}} \times \dot{\boldsymbol{\Omega}} \tag{2.7b}$$

The state vector, which only describes the local motion of a follower satellite in the $\mathscr{H}$ frame is therefore

$$\boldsymbol{x} = \begin{pmatrix} \boldsymbol{p} \\ \boldsymbol{v} \end{pmatrix} = \begin{pmatrix} x & y & z & \dot{x} & \dot{y} & \dot{z} \end{pmatrix}^T \tag{2.8}$$

However, only the above state vector in combination with the leader's absolute state fully describes the absolute orbit of the follower. The back transformation is given by the rotation to the $\mathscr{I}$ frame

$$\boldsymbol{r}_d^{\mathscr{I}} = \boldsymbol{C}^{-1}\boldsymbol{p} \tag{2.9a}$$

$$\boldsymbol{v}_d^{\mathscr{I}} = \boldsymbol{C}^{-1}\left(\boldsymbol{v} - \boldsymbol{p} \times \dot{\boldsymbol{\Omega}}\right) \tag{2.9b}$$

followed by a translation of the vectors by the leader's state

$$\boldsymbol{r_1} = \boldsymbol{r_0} + \boldsymbol{r}_d \tag{2.10a}$$

$$\boldsymbol{v_1} = \boldsymbol{v_0} + \boldsymbol{v}_d \tag{2.10b}$$

With this back transformation, the Keplerian orbit of the follower can be calculated from the leader's absolute state in $\mathscr{I}$ and the follower's relative state in $\mathscr{H}$.

Curvilinear Transformation

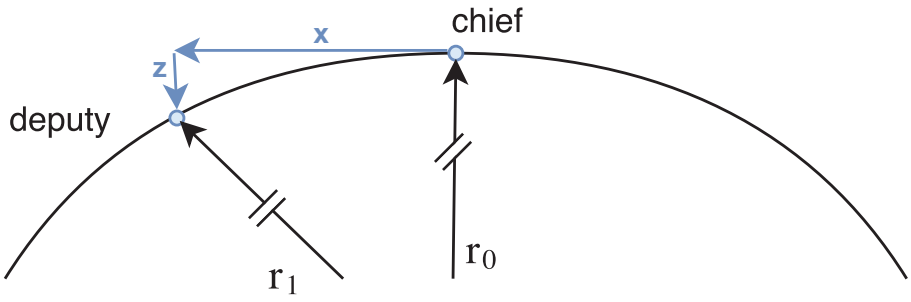


Figure 2.9: ECI-to-Hill transformation error induced through orbit curvature (highly exaggerated).

Vallado and Alfano (2011) suggested a curvilinear transformation between ECI and EH frame to compensate the transformation error induced through the curvature of the orbits. Two satellites at different positions in the same circular orbit have an offset in $\hat{\boldsymbol{e}}_z$ in the $\mathscr{H}$ frame. Figure 2.9 shows this Cartesian transformation error highly exaggerated. But since the two satellites share the same distance from the center of Earth, the radial component of the EH frame should be zero. To overcome this error, the transformation is done in a coordinate system which is based on the equidistant cylindrical system (Weisstein, 2016). A topocentric-horizon system $\mathscr{T}$

(or South-East-Normal (topocentric-horizon) (SEZ)) is generated for the leader. The offset to the follower is measured in angle offsets from the center of Earth. Let the $\mathscr{R}$ frame be defined through the rotation of the $\mathscr{I}$ frame by $\boldsymbol{C}$ (without translation). This means, $\mathscr{T}$ is a cylindrical representation of $\mathscr{R}$. It follows for the two rotation angles ϕ and λ (from leader to follower) in the $\mathscr{T}$ frame

$$\boldsymbol{r}_1^{\mathscr{R}} = \boldsymbol{C}\boldsymbol{r}_1^{\mathscr{I}} \tag{2.11}$$

$$\phi^{\mathscr{T}} = \arcsin\left(\frac{r_z^{\mathscr{R}}}{||\mathbf{r_1}||}\right) \tag{2.12}$$

$$\lambda^{\mathscr{T}} = \operatorname{atan2}\left(r_y^{\mathscr{R}}, r_x^{\mathscr{R}}\right) \tag{2.13}$$

$$\dot{\lambda}_0^{\mathscr{T}} = \frac{\hat{\boldsymbol{e}}_{\boldsymbol{y}}\boldsymbol{C}\boldsymbol{v}_0^{\mathscr{I}}}{||\mathbf{r_0}||^{\mathscr{I}}} \tag{2.14}$$

The position of the follower in the EH frame $\mathscr{H}$ finally computes to

$$x = r_x^{\mathscr{H}} = ||\mathbf{r_1}|| - ||\mathbf{r_0}|| \tag{2.15a}$$

$$y = r_y^{\mathscr{H}} = ||\mathbf{r_0}||\,\lambda \tag{2.15b}$$

$$z = r_z^{\mathscr{H}} = ||\mathbf{r_0}||\,\phi \tag{2.15c}$$

To find the velocity, the angular rates between $\mathscr{I}$ and $\mathscr{T}$ have to be known. Let $\boldsymbol{P}$ be the transformation matrix $\mathscr{R} \to \mathscr{T}$. With

$$\boldsymbol{v}_1^{\mathscr{T}} = \boldsymbol{P}\boldsymbol{C}\boldsymbol{v}_1^{\mathscr{I}} \tag{2.16}$$

holds for the follower's derivatives of the rotation angles in $\mathscr{T}$

$$\dot{\boldsymbol{\phi}} = -\frac{\hat{\boldsymbol{e}}_{\boldsymbol{x}}\boldsymbol{v}_1^{\mathscr{T}}}{||\mathbf{r_1}||} \tag{2.17}$$

$$\dot{\boldsymbol{\lambda}} = \frac{\hat{\boldsymbol{e}}_{\boldsymbol{y}}\boldsymbol{v}_1^{\mathscr{T}}}{||\mathbf{r_1}||\cos\phi} \tag{2.18}$$

Finally, the curvature-corrected velocity in the EH frame $\mathscr{H}$ is

$$v_x = \hat{\boldsymbol{e}}_{\boldsymbol{z}}\boldsymbol{v}_1^{\mathscr{T}} - \hat{\boldsymbol{e}}_{\boldsymbol{x}}\boldsymbol{C}\boldsymbol{v}_0^{\mathscr{I}} \tag{2.19a}$$

$$v_y = ||\mathbf{r_0}||\left(\dot{\boldsymbol{\lambda}} - \dot{\lambda}_0^{\mathscr{T}}\right) \tag{2.19b}$$

$$v_z = ||\mathbf{r_0}||\,\dot{\boldsymbol{\phi}} \tag{2.19c}$$

Although this correction is not significant for close formations, it has a non-negligible influence for larger distances between the satellites, especially in the phase of initial formation acquisition. Since the computational effort is comparable to the transformation in Cartesian coordinates, the curvilinear approach will be used for propagator and controller implementation when converting between the $\mathscr{I}$ and the $\mathscr{H}$ frame. Under strict consideration, using the above transformation routine does result in another non-Cartesian frame, and not in $\mathscr{H}$. However, the non-Cartesian part is wrapped in the transformation and the resulting frame may be used fully commensurable compared to $\mathscr{H}$, while at the same time having a higher accuracy in the $\hat{\boldsymbol{e}}_{\boldsymbol{x}}$, $\hat{\boldsymbol{e}}_{\boldsymbol{y}}$ and $\hat{\boldsymbol{e}}_{\boldsymbol{z}}$ directions.

2.4.2 Orbital Frames

The relative motion model in the Cartesian EH frame lacks the insight of orbital mechanics from the state vector. This insight, however, can be used beneficially in controller design through thrusting action at geometrically advantageous locations in orbit (Schaub et al., 2000b). Therefore, absolute and relative orbital elements are needed to represent the states of the satellites in an orbital frame.

2.4.2.1 Classical Orbital Element

The Classical Orbital Elements (COEs), also known as the Kepler elements, are a set of six parameters which describes the absolute motion of an (artificial) satellite around a central body. Let the state vector in classical orbital elements be

$$\boldsymbol{a_C} = \begin{pmatrix} a & e & i & \omega & \Omega & M \end{pmatrix}^T \tag{2.20}$$

with semi-major axis a [m], eccentricity e (dimensionless), inclination i [rad], argument of perigee ω [rad], right ascension of ascending node Ω [rad] and mean anomaly M [rad] (cf. Fig. 2.10). The transformation between a Cartesian state vector in the ECI frame and COE can be found in Chobotov (2002). For the unperturbed motion, all components of the state vector except the mean anomaly remain constant. Depending on the application, it can be useful to transform the independent variable to the eccentric anomaly E or the true anomaly f. While the mean anomaly increases linearly with time (with 2π-periodicity), the true anomaly is related to the real position of the satellite. The transformation between M and f uses the eccentric anomaly E with the relation

$$M = E - e \sin E \tag{2.21}$$

This demands an iterative solution when converting from M to E, while the reverse transformation can be solved directly. The following formula describes the relation between E and the true anomaly f

$$\cos f = \frac{\cos E - e}{1 - e \cos E} \tag{2.22}$$

The relation between f and M and vice versa is obtained by concatenation of Eq. 2.21 and 2.22. Further characteristics of a Keplerian orbit, which are used later, include the semi-latus rectum p, an eccentricity dependent factor η, the mean motion n and the Keplerian period of the orbit T_O

$$p = a\left(1 - e^2\right) \tag{2.23}$$

$$\eta = \sqrt{1 - e^2} \tag{2.24}$$

$$n = \sqrt{\frac{\mu}{a^3}} \tag{2.25}$$

$$T_O = 2\pi\sqrt{\frac{a^3}{\mu}} = 2\pi n^{-1} \tag{2.26}$$

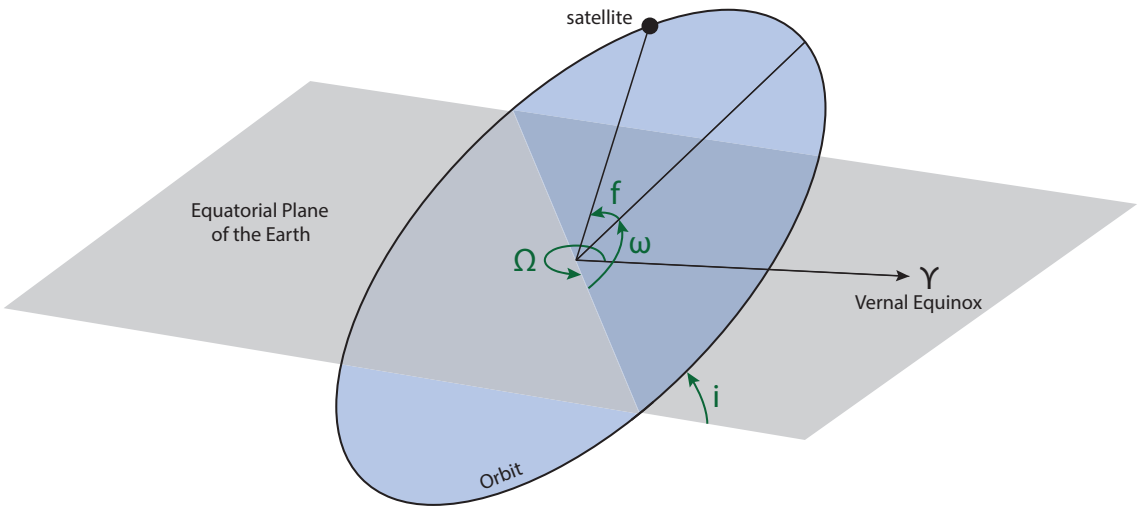


Figure 2.10: Geometry and definition of characteristic parameters of a Keplerian orbit (COE).

1st Order Mapping Between Classical Orbital Elements and Euler-Hill Frame

Further, a first order mapping of the COEs has been performed such that the HCW equations are dependent on the COEs and not on the position and velocity values of the follower in the EH frame. The equations of motion are given by D'Amico (2010) to

$$x \approx -\frac{3}{2}\Delta a\varphi t + a\Delta\varphi + a\Delta\Omega\cos i \quad + 2a\Delta e\sin f \tag{2.27a}$$

$$y \approx \qquad + a\cos(f+\omega)\Delta\Omega\sin i - a\sin(f+\omega)\Delta i \tag{2.27b}$$

$$z \approx -\Delta a \qquad + a\Delta e\cos f \tag{2.27c}$$

In the above equations the operator $\Delta(.)$ denotes real differences in the orbital elements, whereas ∂ denotes relative positions of the follower spacecraft in EH frame. In Eq. 2.27 φ denotes the mean argument of latitude and is defined as $\varphi = \omega + M$. Therefore, φ describes the spacecraft's position on its orbits with respect to its RAAN (right ascension of the ascending node) on the equatorial plane. Note, that φ increases with time. In these equations we observe oscillating parts (sines and cosines), constant parts (Δa, $a\Delta\varphi + a\Delta\Omega\cos i$) and a time dependent part ($-\frac{3}{2}\Delta a\varphi$). For bounded and centered motion these constant and time dependent parts have to become 0. That is

$$\Delta a = 0 \tag{2.28}$$

$$\Delta\varphi + \Delta\Omega\cos i = 0 = \partial\lambda \tag{2.29}$$

The expression $\partial\lambda$ in Eq. 2.29 is called the relative mean longitude and has to be set to 0 for centered motion.

2.4.2.2 Non-Singular Orbital Elements

The COEs have singularities for circular orbits and for $i = \{0°, 90°\}$. Since the nominal NetSat orbit is almost circular, numerical instabilities may occur when using COE. Furthermore, the orbit may become singular when propagating it under the influence of perturbations. Therefore, the first singularity ($e = 0$) is further investigated. For circular orbits the argument of perigee ω is not defined uniquely. More precisely, ω becomes an arbitrary angle measured from the ascending node (the intersection of the orbit with the equatorial plane of Earth). However, the sum of the argument of perigee and the anomaly stays unique and is called the argument of latitude θ

$$\theta = \omega + f \tag{2.30}$$

When only describing the position of a satellite in a circular orbit, the argument of latitude will be used. For more complex calculations (like transforming to/from Cartesian state vectors), a set of six non-singular elements is needed. In this work two different element sets will be used, depending on the application. The first set is called Non-Singular Orbital Element (NOE) with the state vector

$$\boldsymbol{a_N} = \begin{pmatrix} a & \theta & i & q_1 & q_2 & \Omega \end{pmatrix}^T \tag{2.31}$$

where $q_1 = e\cos(\omega)$ and $q_2 = e\sin(\omega)$. The NOE set is the simplest transformation to overcome the COE-singularity at $e = 0$, since the ambiguous argument of perigee

ω has been eliminated. Another set used in this work will be the Equinoctial Orbital Element (EOE) set, based on Walker et al. (1985)

$$\boldsymbol{a_E} = \begin{pmatrix} a & e_x & e_y & i_x & i_y & l_v \end{pmatrix}^T \tag{2.32}$$

where $(e_x \;\; e_y)^T$ is called equinoctial eccentricity vector, $(i_x \;\; i_y)^T$ equinoctial inclination vector and l_v true longitude. They do not have any singularities (including cases stated above) and have a simple transformation from COE. If the initial COE are non-singular, the transformation is described by

$$e_x = e\cos(\omega + \Omega) \tag{2.33a}$$

$$e_y = e\sin(\omega + \Omega) \tag{2.33b}$$

$$i_x = \tan(i/2)\cos\Omega \tag{2.33c}$$

$$i_y = \tan(i/2)\sin\Omega \tag{2.33d}$$

$$l_v = f + \omega + \Omega \tag{2.33e}$$

Sometimes closely related transformations (e.g. using $\tan(i)$ for the inclination vector) are utilized, but they lead to singularities at polar orbits. The set obtained from Eq. 2.33 is guaranteed to be non-singular and has the additional advantage of highly reducing the number of secular terms in its Jacobian matrix, compared to COE and NOE (Broucke and Cefola, 1972). This makes it ideal to use in numerically intensive calculations like orbit propagation. However, the NOE set allows a clearer geometrical insight into the orbit and is thus used for geometric calculations.

2.4.2.3 Differential Orbital Elements

There are two ways to describe the local, or relative, motion of two or more satellites in an orbital frame – Differential Orbital Elements (DOEs) and Relative Orbit Elements (ROEs) (see Chap. 2.4.2.4). ROEs comprise an own frame. DOEs are just the element-wise difference between two satellites' orbital elements (e.g. the absolute difference between their states). For any pair of sets of commensurable orbital elements (COE, NOE, EOE) for the leader ($\boldsymbol{a_0}$) and the follower ($\boldsymbol{a_1}$), they are defined through

$$\partial\boldsymbol{a} = \boldsymbol{a_1} - \boldsymbol{a_0} \tag{2.34}$$

Eq. 2.34 has to be used circumspectly. It is very susceptible for numerical errors, especially in the last field of the vector (the differential anomaly). It is a crucial point to obey this inherent virtue or to find ways to bypass this drawback.

2.4.2.4 Relative Orbit Elements

The ROEs comprise the second local state vector in orbital means. They involve more mathematics than the differential orbital elements, but allow geometrical formation design. From two sets of mean COEs (semi-major axis a, eccentricity e, inclination i, argument of the ascending node Ω, argument of the periapsis ω and the mean anomaly M), the ROEs $\partial\boldsymbol{\alpha}$ are defined as follows (D'Amico, 2010, D'Amico et al., 2009b):

$$\partial\boldsymbol{\alpha} = \begin{pmatrix} \partial a \\ \partial\lambda \\ \partial e_x \\ \partial e_y \\ \partial i_x \\ \partial i_y \end{pmatrix} = \begin{pmatrix} \partial a \\ \partial\lambda \\ \partial \boldsymbol{e}\, cos\,\phi \\ \partial \boldsymbol{e}\, sin\,\phi \\ \partial \boldsymbol{i}\, cos\,\theta \\ \partial \boldsymbol{i}\, sin\,\theta \end{pmatrix} = \begin{pmatrix} (a-a_c)/a_c \\ \varphi - \varphi_c + (\Omega - \Omega_c)\, cos\, i_c \\ e\, cos\,\omega - e_c\, cos\,\omega \\ e\, sin\,\omega - e_c\, sin\,\omega \\ i - i_c \\ (\Omega - \Omega_c)\, sin\, i_c \end{pmatrix} \tag{2.35}$$

with $\varphi = \omega + M$ again being the mean argument of latitude. The subscript $(.)_c$ denotes the leader spacecraft, whereas the COEs of the follower spacecraft are given without subscript. Throughout this thesis $\partial(.)$ denotes a ROE. In order to have a dimensionless state vector, the relative semi-major axis has been divided by a_c. This reference uses the relative mean longitude $\partial\lambda$ instead of the relative mean argument $\partial\varphi = \varphi - \varphi_c$, which is also a common definition used in various occasions. Using $\partial\lambda$ is geometrically more intuitive for formation design, but more difficult to handle when transforming the state vector to the EH frame than $\partial\varphi$ (Maessen and Gill, 2010). Both definitions are used within this thesis at different positions to allow for better understanding in the according context. The relation between the two variables is

$$\partial\lambda = \partial\varphi + (\Omega_1 - \Omega_0)\cos i_0 \tag{2.36}$$

Neglecting any orbit perturbations only φ is time dependent, but not $\partial\lambda$. It follows that the ROE state vector $\partial\boldsymbol{\alpha}$ is constant for a specific formation.

1st Order Mapping Between Relative Orbit Elements and Euler-Hill Frame

A first-order mapping of the ROEs and the EH frame $\mathscr{H}$ providing the follower's position with respect to the leader satellite as a function of $\partial\boldsymbol{\alpha}$ is given by D'Amico (2010), D'Amico et al. (2006)

$$x/a_c = \partial\lambda - \frac{3}{2}\partial a\delta\varphi + 2\partial e \sin(\varphi - \phi) \tag{2.37a}$$

$$y/a_c = -\partial i \sin(\varphi - \theta) \tag{2.37b}$$

$$z/a_c = \partial a + \partial e \cos(\varphi - \phi) \tag{2.37c}$$

Eq. 2.37 assumes a small eccentricity of the leader orbit and small relative distances, e.g. close formations. For stable formations the orbital periods of the satellites have

to be equal. This implies that the center of the in-plane and out-of-plane curves lies actually on the $\hat{\boldsymbol{e}}_{\boldsymbol{x}}$-, $\hat{\boldsymbol{e}}_{\boldsymbol{y}}$ plane. The $\partial\lambda$ quantifies the along-track separation between the two spacecraft. Furthermore, centered motion of the follower satellites around the leader or the virtual center is achieved by equal mean longitudes, which imposes the following constraint on reasonable formations

$$\partial a = 0 \tag{2.38a}$$

$$\partial\lambda = 0 \quad \text{which is identical to} \tag{2.38b}$$

$$\partial\varphi = -\delta\Omega \cos i_0 \tag{2.38c}$$

For this reason, two of the six relative orbital elements are always fixed for centered force-free relative motion. The remaining four elements are composed of the eccentricity vector $\partial\boldsymbol{e}$ and the inclination vector $\partial\boldsymbol{i}$. The formation can be designed by defining the relative eccentricity and inclination vector in polar coordinates

$$\partial\boldsymbol{e} = \begin{pmatrix} \partial e_x \\ \partial e_y \end{pmatrix} = \partial e \begin{pmatrix} \cos\phi \\ \sin\phi \end{pmatrix} \tag{2.39a}$$

$$\partial\boldsymbol{i} = \begin{pmatrix} \partial i_x \\ \partial i_y \end{pmatrix} = \partial i \begin{pmatrix} \cos\theta \\ \sin\theta \end{pmatrix} \tag{2.39b}$$

with ∂e and ∂i being the magnitudes and ϕ and θ the phases. ϕ relates to the relative perigee and θ to the relative ascending node, which renders this polar representation the most useful for geometric formation analysis.

Geometrical Interpretation of the Relative Orbit Elements

The advantage of using $\partial\boldsymbol{\alpha}$ instead of $\boldsymbol{x}$ is given in the possibility of geometrically understanding the formation with only $\partial\boldsymbol{\alpha}$ being provided.

Fig.2.11 shows the relative motion of the follower around the leader for one orbital period in the leader's EH frame, divided into in-plane and out-of-plane motion. The in-plane motion is the projection of the relative motion into the $\hat{\boldsymbol{e}}_{\boldsymbol{x}}$-$\hat{\boldsymbol{e}}_{\boldsymbol{y}}$ plane of $\mathscr{H}$ (which is by definition the orbital plane of the leader) and the out-of-plane motion is the projection into the $\hat{\boldsymbol{e}}_{\boldsymbol{x}}$-$\hat{\boldsymbol{e}}_{\boldsymbol{z}}$ plane. This analysis is done by using the first-order mapping of $\partial\boldsymbol{\alpha}$ into the $\mathscr{H}$ frame (Eq. 2.37) in combination with the constraints of centered force-free formations (Eq. 2.38c), which gives

$$x = 2a_c\partial e \sin(\varphi - \phi) \tag{2.40a}$$

$$y = -a_c\partial i \sin(\varphi - \theta) \tag{2.40b}$$

$$z = -a_c\partial e \cos(\varphi - \phi) \tag{2.40c}$$

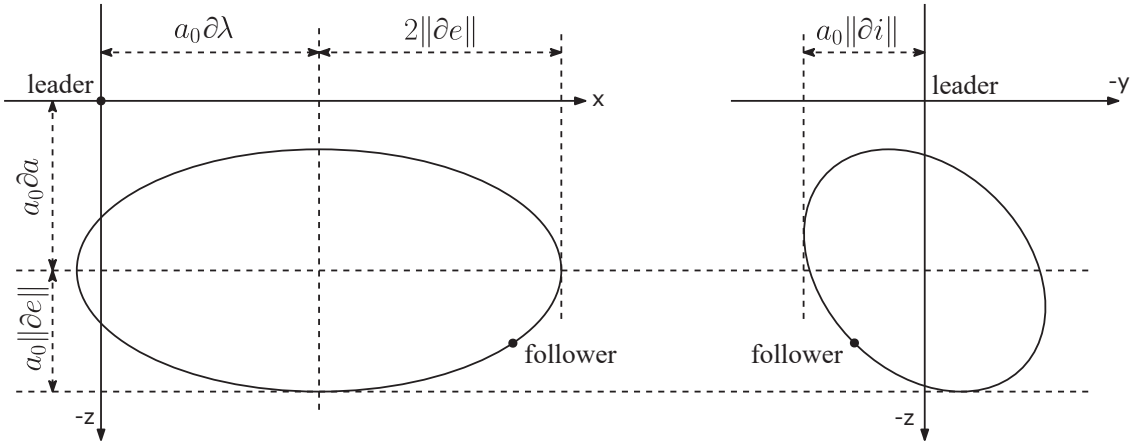


Figure 2.11: Geometric interpretation of ROEs for (nearly) circular orbits. The projection of the relative motion into the $\hat{\boldsymbol{e}}_{\boldsymbol{x}}$-$\hat{\boldsymbol{e}}_{\boldsymbol{z}}$ plane of the EH frame (in-plane motion) is shown on the left. The projection into the $\hat{\boldsymbol{e}}_{\boldsymbol{y}}$-$\hat{\boldsymbol{e}}_{\boldsymbol{z}}$ plane (out-of-plane motion) is shown on the right.

The resulting ellipses for in-plane and out-of-plane motion can be described with the elements of $\partial\boldsymbol{\alpha}$ and the semi-major axis a_c of the leader. The in-plane motion is always a 2-by-1 ellipse. The semi-major axis is defined by $2a_c\partial e$ and the semi-minor axis by $a_c\partial e$, as can be inferred from Eq. 2.40c and 2.40a. Furthermore, the position on the ellipse is only dependent on the sine and cosine of the difference $\varphi - \phi$. Apart from that, the out-of-plane motion is described by an oscillating sine-cosine motion with different phase shifts (Eq. 2.40c and 2.40b), namely $\varphi - \phi$ and $\varphi - \theta$, leading to a tilted ellipse. The maximum extents in radial ($\hat{\boldsymbol{e}}_{\boldsymbol{z}}$) and normal ($\hat{\boldsymbol{e}}_{\boldsymbol{y}}$) direction are given by $a_c\partial e$ and $a_c\partial i$, respectively. Offsets in along-track and radial direction are given by $a_c\partial\lambda$ and $a_c\partial a$ respectively (D'Amico and Montenbruck, 2006). For elliptic orbits this motion is superimposed by an additional mode of twice the frequency and amplitude proportional to the eccentricity of the center spacecraft e_c (Chernick and D'Amico, 2016). Assuming the special case of $\partial e = \partial i$ and $\phi - \theta = 0$, the out-of-plane motion is a circle. For a phase shift difference of $\phi - \theta = 90°$, the out-of-plane motion degrades to a straight line going through the leader's position twice per orbit.

While the relative eccentricity ∂e and the relative inclination ∂i define the shape of the ellipses, the relative perigee ϕ and the relative ascending node θ define the positions on the ellipses. The mean argument of latitude φ (the only time dependent parameter in the discussed equations) is a measure of the satellite's position on its orbit in the ECI reference frame. If $\varphi = \phi$, the follower satellite is at its maximum extent underneath the leader satellite, assuming circular reference orbits. For that reason, the time at which the follower is at a specific position on its bounded motion around the center is controlled by defining ϕ. Similar, when $\varphi = \theta + \pi/2$, the follower has its maximum relative distance in positive normal direction.

2.5 Relative Motion

This section describes the relative motion of two or more spacecraft in orbits around Earth. In general, the relative motion can be computed both, numerically and analytically. Tab. 2.3 compares these two approaches with their characteristics. Since computational effort is a key limitation in small satellites, this section focuses on analytical solutions.

Table 2.3: Methods for computation of relative motion of spacecraft in Earth's orbit (based on Gill (2011)).

Method	Numerical Computation	Analytical Computation
Type	• Numerical propagation of absolute spacecraft trajectories	• Analytical model of relative motion
Typical application	• On-ground orbit planning	• Mission planning • On-board implementation
Benefits	• Accuracy • Generality	• Physical insight • Lower computation effort than numerical methods
Examples	• Runge-Kutta • Differential equation multi-step method	• Hill-Clohessy-Wiltshire equations (Clohessy and Wiltshire, 1960) as a solution to Hill's equations for circular orbits (Hill, 1878) • Yamanaka-Ankersen solution (Yamanaka and Ankersen, 2002) to Tschauner-Hempel equations for elliptical orbits (Hempel and Tschauner, 1964, Tschauner and Hempel, 1964)

First, differential equations in Cartesian coordinates are derived. This includes the central attraction force described by Newton's law of universal gravitation and perturbing forces, which are exerted to the satellites. This is followed by an analysis in

orbital frames, where the motion is split into mean and osculating terms, partitioned into absolute mean and relative mean motion. In the second part the differential equations are used as inputs for (linearized) relative motion models and to find suitable State-Transition Matrices (STMs). Errors of the most common STMs are analyzed.

2.5.1 Equations of Motion in Cartesian Frames

Neglecting any perturbing forces and assuming a spherical Earth, the motion of a satellite in $\mathscr{I}$ frame is fully described with the restricted two-body problem (Keplerian equation of motion)

$$\ddot{\boldsymbol{r}} + \frac{\mu \boldsymbol{r}}{r^3} = 0 \tag{2.41}$$

With the relative position

$$\boldsymbol{r_d} = \boldsymbol{r_1} - \boldsymbol{r_0}\,, \tag{2.42}$$

the relative equation of motion becomes

$$\ddot{\boldsymbol{r}}_d = -\frac{\mu\left(\boldsymbol{r_0} + \boldsymbol{r_d}\right)}{||\mathbf{r_0} + \mathbf{r_d}||^3} + \frac{\mu}{r_0^3}\boldsymbol{r_0} \tag{2.43}$$

With the help of Eq. 2.7, Alfriend et al. (2010) transform Eq. 2.43 into the following component-wise differential equation, which describes the relative motion in the EH frame $\mathscr{H}$

$$\ddot{x} - 2\dot{\theta}_0\dot{z} - \ddot{\theta}_0 z - \dot{\theta}_0^2 y = -\frac{\mu x}{\left[\left(r_0 - z\right)^2 + y^2 + x^2\right]^{3/2}} \tag{2.44a}$$

$$\ddot{y} = -\frac{\mu y}{\left[\left(r_0 - z\right)^2 + y^2 + x^2\right]^{3/2}} \tag{2.44b}$$

$$-\ddot{z} - 2\dot{\theta}_0\dot{x} - \ddot{\theta}_0 x + \dot{\theta}_0^2 z = -\frac{\mu\left(r_0 - z\right)}{\left[\left(r_0 - z\right)^2 + y^2 + x^2\right]^{3/2}} + \frac{\mu}{r_0^2} \tag{2.44c}$$

In the above-given equation, θ_0 is the $\hat{\boldsymbol{e}}_{\boldsymbol{z}}$ component of the frame rotation. The magnitude of θ_0 is equal to the magnitude of $\boldsymbol{\Omega}$, since through definition of the frames holds $\boldsymbol{\Omega}^{\mathscr{H}} = (0\ \ 0\ \ \theta_0)^T$. This system of non-linear differential equations has a single equilibrium at $\boldsymbol{p} = (0\ \ 0\ \ 0)^T$ and infinitely many equilibria for circular leader orbits. For $e_0 = 0$ every position on the leader's orbit is a stable point and the two spacecraft are in a co-orbital motion. Although these equations can be simplified by changing the independent variable from time t to true anomaly f (Szebehely and Giacaglia, 1964), there is no analytical solution at hand. However, Eq. 2.44 is the origin of the linearized models presented later on.

2.5.1.1 Perturbation Forces

All satellite orbits are perturbed by the Earth's non-spherical gravity field, solar radiation pressure, atmospheric drag and the forces exerted by other celestial bodies and thus deviate from a sole Keplerian shape. Most perturbing forces $\boldsymbol{f}_i$ for Earth orbiting satellites may be divided into short-periodic and long-periodic perturbations, where the period used for distinction is in the order of magnitude of the orbital period of the satellite (for LEO e.g. approx. 90 min)

$$\boldsymbol{f}_i(t) = \boldsymbol{f}_{i,sp}(t) + \boldsymbol{f}_{i,lp}(t) \tag{2.45}$$

The total force seen by a satellite at any time instant is therefore

$$\boldsymbol{f}_{total}(t) = \boldsymbol{f}_N(t) + \sum_i \left(\boldsymbol{f}_{i,sp}(t) + \boldsymbol{f}_{i,lp}(t)\right) + \boldsymbol{f}_c(t) \tag{2.46}$$

where $\boldsymbol{f}_N$ is the idealized central-body attraction and $\boldsymbol{f}_c$ is the total applied control force. $\boldsymbol{f}_N$ is accurately known (Eq. 2.41), as well as the control input $\boldsymbol{f}_c$. The evolution of the relative state of two satellites is driven by the differential force between the two satellites, namely

$$\Delta\boldsymbol{f}(t) = \boldsymbol{f}_{1,total}(t) - \boldsymbol{f}_{0,total}(t) \tag{2.47}$$

For the analysis of the relative motion, only the differential perturbing acceleration due to the Earth's oblateness is of higher interest. While the differential acceleration caused by the oblateness of the Earth (J_2-term) is of order $10^{-6} - 10^{-5}\ \mathrm{m\,s^{-2}}$ (for formations of up to 100 km separation), all other differential perturbing forces do not exceed a magnitude of two decimal powers less (D'Amico, 2010). This assumption is valid for satellites in LEO that share the same or similar ballistic coefficients as the NetSat formation. Furthermore, the NetSat satellites use continuous low-thrust propulsion, which means the total propellant mass compared to the whole satellite mass is negligible ($< 1\%$). Thus, fuel-balancing between the satellites may be ignored, which facilitates controller design. Considering this, to accurately model the relative motion to a decent, but practical level, the J_2-term of the Earth may be taken into account, while all other perturbing forces are neglected.
This reduces the unknowns in Eq. 2.47 to $\Delta\boldsymbol{f}_{J2,sp}$ and $\Delta\boldsymbol{f}_{J2,lp}$. Analysis of the gravity field influence shows that short-periodic oscillations induced by J_2 are very small for close formations ($\Delta\boldsymbol{f}_{J2,sp} \approx 0$), while the secular variations on a larger time scale have a more significant influence on the relative motion. Hence, for a first approximation, $\Delta\boldsymbol{f}_{J2,sp}$ may be neglected. The two-body problem (see Eq. 2.41) with a non-spherical gravity model transforms to

$$\ddot{\boldsymbol{r}} = -\nabla_{\boldsymbol{r}}\mathscr{V} + \boldsymbol{u}\,, \tag{2.48}$$

where $\mathscr{V}$ is the sum of the Keplerian attraction and the J-terms of the Earth and $\nabla_{\boldsymbol{r}}$ is the Nabla operator at current position $\boldsymbol{r}$. Further perturbation and control forces are indicated by $\boldsymbol{u}$. Kaplan (1976) developed an explicit formulation for $\mathscr{V}$,

when only the second dynamic form factor of the Earth (J_2) is taken into account

$$\mathscr{V} = -\frac{\mu}{r}\left\{1 - \frac{J_2 R_E^2}{2r^2}\left[\frac{3}{r^2}\left(\boldsymbol{r}\hat{\boldsymbol{K}}\right)^2 - 1\right]\right\} \tag{2.49}$$

Here, $\hat{\boldsymbol{K}}$ is the unit vector along the $\mathscr{I}$ frame's polar axis. An explicit formulation of the gravitational acceleration in x, y and z of the $\mathscr{I}$ frame is derived by Bate et al. (1971), but there is no straightforward direct transformation into the $\mathscr{H}$ frame (like Eq. 2.44 for the unperturbed model) with gravitational perturbations. Alfriend et al. (2010) formulated the relative motion states in terms of the angular momentum $\boldsymbol{h_0}$ (see Eq. 2.3) of the leader satellite in the EH frame

$$x = \frac{\delta \boldsymbol{r}^{\boldsymbol{T}}\left(\boldsymbol{h_0} \times \boldsymbol{r_0}\right)}{||\mathbf{h_0} \times \mathbf{r_0}||} \tag{2.50a}$$

$$y = -\frac{\delta \boldsymbol{r}^{\boldsymbol{T}} \boldsymbol{h_0}}{h_0} \tag{2.50b}$$

$$z = -\frac{\delta \boldsymbol{r}^{\boldsymbol{T}} \boldsymbol{r_0}}{r_0} \tag{2.50c}$$

The relative velocities can be derived by differentiating Eq. 2.50. To accurately model a follower's $\boldsymbol{pv}$ vector in the EH frame with the above equation, the angular momentum vector of the leader (which is subject to J_2 perturbations) has to be known for all times. This induces an impracticable constraint for controller design, which again demands the development of relative propagation equations linearized around the leader satellite's $\boldsymbol{pv}$ vector.

2.5.2 Equations Motion in Orbital Frames

The preceding section analyzed the influence of perturbations in Cartesian frames. This can also be done in orbital frames, where motion description can then be split into mean and osculating terms. For the unperturbed case, considering only the Newtonian attraction $\boldsymbol{f}_N$, all orbital elements – except those including anomaly – stay constant over time. The mean anomaly is related linearly to time, while the other anomaly types are connected via transformation functions (Eq. 2.21 and 2.22). When perturbing or control forces are considered, all orbital elements change over time. Let $\boldsymbol{a}$ be the COE state vector of a certain satellite (without loss of generality also applicable for other orbital frames). The time-dependent, or osculating, orbit of the satellite may now be described with

$$\boldsymbol{a_{osc}}(t) = \boldsymbol{A} + \boldsymbol{a_n} + \boldsymbol{\Delta a}(t) \tag{2.51}$$

where $\boldsymbol{a_n}$ is the nominal (unperturbed) orbital element state at a certain time instant t and $\boldsymbol{\Delta a}(t)$ the perturbation-induced change. $\boldsymbol{A}$ accommodates the mean Keplerian

motion, which influences the anomaly

$$\boldsymbol{A} = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & n\,t \end{pmatrix}^T \tag{2.52}$$

where n is the mean motion. The last part of Eq. 2.51 can be split into short-periodic, long-periodic and drift effects

$$\boldsymbol{\Delta a}\,(t) = \boldsymbol{\Delta a_{sp}}\,(t) + \boldsymbol{\Delta a_{lp}}\,(t) + \boldsymbol{\Delta a_{d}}\,(t) \tag{2.53}$$

In this partition, all oscillating terms must be part of the short- and long-periodic parts and linear terms must be in the drift part. From that, one can define the relationship between an osculating and a mean orbit $\bar{\boldsymbol{a}}$

$$\boldsymbol{a_{osc}}\,(t) = \bar{\boldsymbol{a}}\,(t) + \boldsymbol{\Delta a_{osc}}\,(t) \tag{2.54}$$

where

$$\bar{\boldsymbol{a}}\,(t) = \boldsymbol{A} + \boldsymbol{a_n} + \boldsymbol{\Delta a_{d}}\,(t) \tag{2.55a}$$

$$\boldsymbol{\Delta a_{osc}}\,(t) = \boldsymbol{\Delta a_{sp}}\,(t) + \boldsymbol{\Delta a_{lp}}\,(t) \tag{2.55b}$$

The nominal orbit $\boldsymbol{a_n}$ coincides with the mean orbit at $t = 0$, so $\bar{\boldsymbol{a}}_{\boldsymbol{0}} = \boldsymbol{a_n}$. Due to the fact that the drift part contains only linear terms, Eq. 2.55a can be written as

$$\bar{\boldsymbol{a}}\,(t) = \boldsymbol{A} + \bar{\boldsymbol{a}}_{\boldsymbol{0}} + \frac{d\bar{\boldsymbol{a}}}{dt}\Delta t = \boldsymbol{A} + \bar{\boldsymbol{a}}_{\boldsymbol{0}} + \begin{pmatrix} {}^{d\bar{a}}/_{dt} \\ {}^{d\bar{e}}/_{dt} \\ {}^{d\bar{i}}/_{dt} \\ {}^{d\bar{\omega}}/_{dt} \\ {}^{d\bar{\Omega}}/_{dt} \\ {}^{d\bar{M}_0}/_{dt} \end{pmatrix} \Delta t \tag{2.56}$$

With this correlation between a mean orbit $\bar{\boldsymbol{a}}$, which is subject to a constant drift, and an osculating orbit $\boldsymbol{a_{osc}}$, the effects of perturbation forces on the orbital elements may be investigated. Using the development above, one can infer that the osculating orbit describes the real position at a given time instant (which is directly related to the current $\boldsymbol{pv}$ vector) and the mean orbit describes an averaged position over multiple orbital periods. The transformation between osculating and mean orbit can be done numerically or with analytic methods. The computationally intensive numeric approach is done in orbit propagation tools, like Orbit Extrapolation Kit (Orekit) (Maisonobe and Parraud, 2016). Any combination of perturbing forces can be used as input for the averaging process, whereby all oscillating terms can be eliminated from the mean orbit without the need for explicit equations of the perturbation influence.

The drawback of this approach is the large computational effort for each single transformation, whereby the need for analytic models arises for on-board propagators

and controllers. Since only the J_2-perturbation is of greater interest for formation flying (see Chap. 2.5.1.1), only this force will be investigated in detail. This being by far the largest perturbing force for LEO satellites, many investigations have been undertaken, for example by Walter (1967) and Weisman et al. (2014). Most are based on Brouwer's transformation, which provides a transformation function solely based on the orbital elements and independent of time (Brouwer, 1959). Brouwer normalized the J_2 disturbance force (see Eq. 2.49) with $\epsilon = J_2 R_E^2$ and found an explicit expression for 2.55b with a change of dependent variables from time t to current mean orbit $\bar{\boldsymbol{a}}$

$$\boldsymbol{\Delta a_{osc}}\left(\bar{\boldsymbol{a}}\right) = -\epsilon\left(\boldsymbol{\Delta a_{sp,1}}\left(\bar{\boldsymbol{a}}\right) + \boldsymbol{\Delta a_{sp,2}}\left(\bar{\boldsymbol{a}}\right) + \boldsymbol{\Delta a_{lp}}\left(\bar{\boldsymbol{a}}\right)\right) \tag{2.57}$$

The short-periodic influence is split into two parts to accommodate all present perturbation frequencies. The explicit formulation of the right-hand-side of 2.57 for the NOE set can be found in App. B. Brouwer found them by using a generating function, where the orbit is represented in normalized Delaunay variables. The Delaunay variables are obtained by describing the satellite motion in Hamiltonian mechanics. The Hamiltonian $\mathfrak{H}$ is a function of time t, generalized position coordinates $\boldsymbol{q}$ and conjugate momenta $\boldsymbol{p}$

$$\mathfrak{H}\left(\boldsymbol{q}, \boldsymbol{p}, t\right) = \boldsymbol{p}^T \dot{\boldsymbol{q}} - \mathfrak{L}\left(\boldsymbol{q}, \dot{\boldsymbol{q}}, t\right) \tag{2.58}$$

The conjugate momenta $\boldsymbol{p}$ are derived from the generalized velocity coordinates $\dot{\boldsymbol{q}}$ with

$$\boldsymbol{p} = \frac{\delta \mathfrak{L}}{\delta \dot{\boldsymbol{q}}} \tag{2.59}$$

which are defined through Lagrangian mechanics. The Lagrangian of a dynamic system is described through its kinetic energy $\mathfrak{K}$ and potential energy $\mathfrak{U}$

$$\mathfrak{L}\left(\boldsymbol{q}, \dot{\boldsymbol{q}}, t\right) = \mathfrak{K} - \mathfrak{U} \tag{2.60}$$

The corresponding relationship between general coordinates and classical orbital elements are the Delaunay elements

$$l = M \tag{2.61a}$$

$$g = \omega \tag{2.61b}$$

$$h = \Omega \tag{2.61c}$$

$$L = \sqrt{\mu a} \tag{2.61d}$$

$$G = \sqrt{\mu a\left(1 - e^2\right)} \tag{2.61e}$$

$$H = G \cos i \tag{2.61f}$$

The first three elements (lower case letters) of the Delaunay elements comprise the generalised Hamiltonian coordinates and the latter three (upper case letters) the conjugate momenta

$$\boldsymbol{q} = (l, g, h)^T \tag{2.62a}$$

$$\boldsymbol{p} = (L, G, H)^T \tag{2.62b}$$

Since $\mathfrak{H}$ spans a cotangent space, the time derivatives of the Delaunay elements are

$$\dot{l} = \frac{\delta \mathfrak{H}}{\delta L}, \;\; \dot{g} = \frac{\delta \mathfrak{H}}{\delta G}, \;\; \dot{h} = \frac{\delta \mathfrak{H}}{\delta H} \tag{2.63a}$$

$$\dot{L} = \frac{\delta \mathfrak{H}}{\delta l}, \;\; \dot{G} = \frac{\delta \mathfrak{H}}{\delta g}, \;\; \dot{H} = \frac{\delta \mathfrak{H}}{\delta h} \tag{2.63b}$$

In this representation, the orbit is split into a nominal and a perturbed part

$$\mathfrak{H} = \mathfrak{H}_{\mathfrak{o}} + \epsilon \mathfrak{H}_{\mathfrak{1}} \tag{2.64}$$

Since G is the orbital momentum and H its polar component, the unperturbed part $\mathfrak{H}_{\mathfrak{o}}$ (Keplerian motion) is only a function of L

$$\mathfrak{H}_{\mathfrak{o}} = -\frac{\mu^2}{2L^2} \tag{2.65}$$

Thus, all conjugate momenta and g and h are constant in $\mathfrak{H}_{\mathfrak{o}}$. Brouwer's generating function for the mean-to-osculation transformation is an explicit formulation of $\mathfrak{H}_{\mathfrak{1}}$ in terms of the Delaunay elements, which can be transformed to obtain an explicit formulation for Eq. 2.57. For further information about Hamiltonian orbit representation, the reader is referred to Hori (1966) and Chang and Marsden (2003). This energy-based orbit representation also proves useful to mitigate perturbation influences (see Chap. 2.6.9).

Since the norm of $\boldsymbol{\Delta a_{osc}}(\bar{\boldsymbol{a}})$ is very small compared to $\bar{\boldsymbol{a}}(t)$, the inverse transformation can be expressed as

$$\bar{\boldsymbol{a}}(\boldsymbol{a_{osc}}) = \boldsymbol{a_{osc}} - \boldsymbol{\Delta a_{osc}}(\boldsymbol{a_{osc}}) \tag{2.66}$$

To describe the evolution of the mean and osculating orbit of a satellite with time, additionally the change of the mean orbit $d\bar{a}/dt$ is needed. Schaub et al. (2000b) state that the semi-major axis, the eccentricity and the inclination do not suffer from J_2-drift, hence $d\bar{a}/dt = d\bar{e}/dt = d\bar{i}/dt = 0$. The other mean orbital elements are subject to the following quasi-constant drift rates

$$\frac{d\bar{\omega}}{dt} = \frac{3}{4} J_2 \left(\frac{R_E}{\bar{p}}\right)^2 \bar{n} \left(5\cos^2 \bar{i} - 1\right) \tag{2.67a}$$

$$\frac{d\bar{\Omega}}{dt} = -\frac{3}{2} J_2 \left(\frac{R_E}{\bar{p}}\right)^2 \bar{n} \cos \bar{i} \tag{2.67b}$$

$$\frac{d\bar{M}_0}{dt} = \frac{3}{4} J_2 \left(\frac{R_E}{\bar{p}}\right)^2 \bar{n} \sqrt{1-\bar{e}^2} \left(3\cos^2 \bar{i} - 1\right) \tag{2.67c}$$

Inserting Eq. 2.57 and 2.67 into Eq. 2.54 gives an explicit formulation for the transformation between osculating and mean orbital elements and vice versa. An explicit formulation for Eq. 2.67 with EOE can be found in Gim and Alfriend (2005).

2.5.2.1 Mean Relative Motion

The derivations from the preceding section can also be applied to DOEs. Eq. 2.34 may be used with osculating or mean orbital elements in any orbital frame. With the differential orbital elements and the reference orbit in osculating elements, a transformation to mean differential orbital elements would be possible with Eq. 2.66. For this, the differential orbit has to be transformed to an absolute orbit with Eq. 2.34, then transformed (along with the reference orbit) to a mean orbit and then transformed back to differential elements. This can be simplified by using a direct mean-to-osculating transformation in differential elements (Gim and Alfriend, 2003). The relation is approximated with a Jacobian-like transformation matrix $\boldsymbol{D}$

$$\delta \boldsymbol{a_{osc}} = \boldsymbol{D}\ \delta \bar{\boldsymbol{a}} \tag{2.68}$$

where $\boldsymbol{D}$ is a 6×6 matrix with the partial derivatives of the osculating elements with respect to the mean elements. Using Eq. 2.57, the following representation of $\boldsymbol{D}$ with dependency solely on the current mean reference orbit $\bar{\boldsymbol{a}}(t)$ can be derived

$$\boldsymbol{D}\left(\bar{\boldsymbol{a}}\right) = \frac{\delta \boldsymbol{a_{osc}}}{\delta \bar{\boldsymbol{a}}} = \boldsymbol{I} - \left(J_2 R_E^2\right)\left(\boldsymbol{D_{sp,1}}\left(\bar{\boldsymbol{a}}\right) + \boldsymbol{D_{sp,2}}\left(\bar{\boldsymbol{a}}\right) + \boldsymbol{D_{lp}}\left(\bar{\boldsymbol{a}}\right)\right) \tag{2.69}$$

The elements $\boldsymbol{D_{sp,1}}$, $\boldsymbol{D_{sp,2}}$ and $\boldsymbol{D_{lp}}$ depend on the reference orbit $\bar{\boldsymbol{a}}$ and make use of the transformation vectors $\Delta a_{sp,1}$, $\Delta a_{sp,2}$ and Δa_{lp} of the Brouwer transformation (Eq. 2.57). For orbits with small eccentricities, the terms of $\boldsymbol{D}$ which include multiplications of e with J_2 almost vanish. Alfriend et al. (2010) derived a simplified version

$$\boldsymbol{D_s} \approx \boldsymbol{D} \quad \forall\ e \ll 1 \tag{2.70}$$

where these terms have been dropped. The elements of this simplified version $\boldsymbol{D_s}$ for differential NOE can be found in App. C. For the vice versa conversion from osculating to mean differential elements, $\boldsymbol{D}^{-1}$ has to be found. Since $\boldsymbol{D}$ is almost an identity matrix due to the small perturbation coefficient, this can be omitted and the following relation is valid to a high level of accuracy

$$\delta \bar{\boldsymbol{a}} = -\boldsymbol{D}\left(\boldsymbol{a_{osc}}\right) \delta \boldsymbol{a_{osc}} \tag{2.71}$$

2.5.3 Relative Orbit Propagation in Euler-Hill Frame

In the previous section, differential equations of relative motion and mean motion descriptions have been developed. This section will focus on using these equations as inputs for (linearized) relative motion models to define corresponding STMs. With these STMs, the evolution of the relative state can be predicted. These matrices are also the foundation for controller plant matrices investigated in Chap. 3.2. The first relative orbit propagation model to be addressed is a simpler solution, where orbit eccentricity and J_2 perturbations are neglected. The following models will include either of the two, or both, and grow in complexity. Hence they show a higher congruency with the real relative motion behavior.

2.5.3.1 Hill-Clohessy-Wiltshire Equations

The first STM to be investigated, namely the HCW equations (from Hill, Clohessy and Wiltshire), is one of the first motion descriptions for relative orbital motion (Clohessy and Wiltshire, 1960) ever developed. It is a set of Ordinary Differential Equations (ODEs) obtained by linearizing Eq. 2.44 around the position of the leader satellite, e.g. the origin of the EH frame $\mathscr{H}$. The eccentricity of the leader's orbit is assumed to be (nearly) zero. A Taylor series expansion around the leader including terms up to the first-order provides the following equations (Montenbruck and Gill, 2012)

$$\ddot{x} - 2\,n\,\dot{z} = d_x + u_x \tag{2.72a}$$

$$\ddot{y} + n^2\,y = d_y + u_y \tag{2.72b}$$

$$\ddot{z} + 2\,n\,\dot{x} - 3\,n^2\,z = d_z + u_z \tag{2.72c}$$

with $n = \sqrt{\mu/a^3}$, μ is the Earth's standard gravitational parameter and a the semi-major axis of the orbit on which the coordinate frame's origin is placed. $\boldsymbol{d} = [d_x,\, d_y,\, d_z]^T$ is the vector of environmental perturbations/disturbances and $\boldsymbol{u} = [u_x,\, u_y,\, u_z]^T$ the vector of control accelerations (Fehse, 2003, p. 40). One can directly see that the in-plane motion ($\hat{\boldsymbol{e}}_{\boldsymbol{x}}$-$\hat{\boldsymbol{e}}_{\boldsymbol{z}}$ plane) is coupled, but decoupled from the out-of-plane motion ($\hat{\boldsymbol{e}}_{\boldsymbol{y}}$ direction). The Hill's equations have been solved analytically by Clohessy and Wiltshire (1960) presenting the following solution

$$\begin{aligned} x(t) = &+ [x_0 + \frac{2\,\dot{z}_0}{n}] + [\frac{4\,\dot{x}_0}{n} - 6\,z_0]\,\sin(n\,t) - \frac{2\,\dot{z}_0}{n}\,\cos(n\,t) \\ &+ [6\,n\,z_0 - 3\,\dot{x}_0]\,t + \gamma_x\,[\frac{4}{n^2}\,(1 - \cos(n\,t)) - \frac{3}{2}\,t^2] \end{aligned} \tag{2.73a}$$

$$y(t) = + \frac{\dot{y}_0}{n}\,\sin(n\,t) + y_0\,\cos(n\,t) + \frac{\gamma_y}{n^2}\,[1 - \cos(n\,t)] \tag{2.73b}$$

$$z(t) = +\left[4\ z_0 - \frac{2\ \dot{x}_0}{n}\right] + \frac{\dot{z}_0}{n}\ \sin(n\ t) + \left[\frac{2\ \dot{x}_0}{n} - 3\ z_0\right]\ \cos(n\ t)$$

$$+ \frac{\gamma_z}{n^2}\ [1 - \cos(n\ t)] \tag{2.73c}$$

with the total external accelerations $\boldsymbol{\gamma} = [\gamma_x,\, \gamma_y,\, \gamma_z]^T = \boldsymbol{d} + \boldsymbol{u}$ (Fehse, 2003, p. 40f.). This set of equations is commonly called Hill-Clohessy-Wiltshire equations. An alternative definition of the HCW equations in the LVLH defined by Alfriend et al. (2010) is presented in App. A.2.

To obtain a model describing the force-free motion, the perturbing and control accelerations ($\boldsymbol{d}$ and $\boldsymbol{u}$) are set to zero, leading to the homogeneous form of the second-order ODE Eq. 2.72, which can be written in matrix form as

$$\dot{\boldsymbol{x}}\,(t) = \boldsymbol{A}\ \boldsymbol{x}\,(t) \tag{2.74}$$

with the state vector defined as

$$\boldsymbol{x}\,(t) = \begin{pmatrix} x & y & z & \dot{x} & \dot{y} & \dot{z} \end{pmatrix}^T \tag{2.75}$$

The overall non-homogeneous form of the ODE (including external control forces) or so called state-space representation can be written as

$$\dot{\boldsymbol{x}} = \boldsymbol{A}\ \boldsymbol{x} + \boldsymbol{B}\ \boldsymbol{u} \tag{2.76}$$

or in detail

$$\dot{\boldsymbol{x}} = \begin{bmatrix} \dot{x}(t) \\ \dot{y}(t) \\ \dot{z}(t) \\ \ddot{x}(t) \\ \ddot{y}(t) \\ \ddot{z}(t) \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 2n \\ 0 & -n^2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 3n^2 & -2n & 0 & 0 \end{bmatrix} \cdot \begin{bmatrix} x(t) \\ y(t) \\ z(t) \\ \dot{x}(t) \\ \dot{y}(t) \\ \dot{z}(t) \end{bmatrix}$$

$$+ \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} u_x(t) \\ u_y(t) \\ u_z(t) \end{bmatrix} \tag{2.77}$$

with total external accelerations or control vector $\boldsymbol{u}(t) = [u_x,\, u_y,\, u_z]^T = \frac{\boldsymbol{F}}{m_c}$ and $\boldsymbol{F}$ being the thrust force and m_c the satellite mass. $\boldsymbol{B}$ is called influence matrix and defines the the effect of the thrust on the satellite motion (Fehse, 2003, p. 40f.).

To propagate the satellite motion over a longer duration like an orbit propagator does, the STM $\boldsymbol{\Phi}\,(t, 0)$, which maps the state at time $t_0 = 0$ to the state at time t, is needed

$$\boldsymbol{x}\,(t) = \boldsymbol{\Phi}\,(t, 0)\,\boldsymbol{x}\,(0) \tag{2.78}$$

The STM is the matrix form of the extended HCW equations including both position and velocity. Since Eq. 2.74 is a Linear Time-Invariant (LTI) system, $\mathbf{\Phi}$ can be obtained with the matrix exponential of $\boldsymbol{A}$

$$\mathbf{\Phi}\left(t,0\right)=e^{\boldsymbol{A}t}=\begin{bmatrix}1&0&6(s-nt)&4sn^{-1}-3t&0&2n^{-1}(c-1)\\0&-c&0&0&-sn^{-1}&0\\0&0&-4+3c&-2n^{-1}(1-c)&0&-s/n\\0&0&6n(c-1)&4c-3&0&-2s\\0&ns&0&0&-c&0\\0&0&-3ns&-2s&0&-c\end{bmatrix} \tag{2.79}$$

with $c=\cos(nt)$ and $s=\sin(nt)$. This STM is the basis for many developments of more accurate relative motion models. The HCW solution is only accurate for (nearly) circular leader orbits and very small formations (up to tens of kilometers, cf. Chap. 2.5.3.3 for a detailed evaluation), since the neglected differential perturbations are proportional to the distances between the spacecraft (or the distance between spacecraft and virtual leader, if chosen). An investigation of Eq. 2.79 shows that the in-track position x (first row) is subject to drift through the term

$$\alpha=6ntz(0)-3t\dot{x}(0) \tag{2.80}$$

Only formations where this term vanishes are stable, e.g. α has to be zero in order to establish a force-free formation. This leads to the initial condition of stable formations

$$\dot{x}(0)=2nz(0) \tag{2.81}$$

2.5.3.2 State-Transition Matrices for Eccentric Leader Orbits

The most widespread method of obeying the eccentricity in the relative motion models is to change the independent variable in Eq. 2.44 from time t to true anomaly f. This is achieved by inserting Eq. 2.22 and 2.21 into Eq. 2.44 and setting $M=2\pi\ t\ T_O^{-1}$. The whole transformation is stated by Alfriend et al. (2010) and uses normalized position components $\bar{x}_i=x_i r^{-1}$. With a one-dimensional potential function in $\mathscr{H}$ frame

$$\mathfrak{U}=-\frac{1}{\left[\left(1+\bar{x}\right)^2+\bar{y}^2+\bar{z}^2\right]^{0.5}}+1-\bar{x} \tag{2.82}$$

and a pseudo-potential function

$$\mathfrak{W}=\frac{1}{1+e_0\cos f_0}\left[0.5\left(\bar{x}^2+\bar{y}^2-e_0\bar{z}^2\cos f_0\right)-\mathfrak{U}\right] \tag{2.83}$$

the final transformation of 2.44 to anomaly-dependent form using Eq. 2.83 is

$$\bar{x}''-2\bar{z}'=\frac{\delta\mathfrak{W}}{\delta\bar{x}} \tag{2.84a}$$

$$-\bar{y}'' = \frac{\delta \mathfrak{W}}{\delta \bar{y}} \tag{2.84b}$$

$$\bar{z}'' + 2\bar{x}' = \frac{\delta \mathfrak{W}}{\delta \bar{z}} \tag{2.84c}$$

The notation $(\cdot)'$ denotes the derivative with respect to true anomaly f. It is obtained by the chain rule

$$\frac{d(\cdot)}{dt} = (\cdot)' \dot{f} \tag{2.85a}$$

$$\frac{d^2(\cdot)}{dt^2} = (\cdot)'' \dot{f}^2 + \dot{f}\dot{f}'(\cdot)' \tag{2.85b}$$

Hempel and Tschauner (1964) developed a linearized form of Eq. 2.84, which is referred to as Tschauner-Hempel (TH) equations in literature. Similar to the HCW equations, a first order Taylor expansion around the leader orbit leads to the following differential equations

$$\bar{x}'' = 2\bar{z} \tag{2.86a}$$

$$\bar{y}'' = -\bar{y}' \tag{2.86b}$$

$$\bar{z}'' = \frac{3}{k}\bar{z} - 2\bar{x} \tag{2.86c}$$

where $k = 1 + e\cos(f)$. As in the HCW equations, the decoupling of in-plane and out-of-plane motion also holds for the eccentric model. A closed-form solution to Eq. 2.86 was obtained by Lawden (1963), resulting in a STM for relative motion which incorporates the eccentricity of the leader orbit and thereby giving a major improvement compared to the HCW solution. This STM was extended by Carter (1998), who eliminated a singularity of Lawden's STM for $f = i\pi\,,\ i \in \mathbb{N}$. Carter's STM is therefore a particular solution to Eq. 2.86 and uses a partly-normalized state vector $\boldsymbol{x} = (\bar{x}\ \ \bar{x}'\ \ y\ \ \bar{y}'\ \ z\ \ \bar{z}')^T$. The closed-form solution is given by

$$\boldsymbol{\Phi}_{\boldsymbol{C}}(f) = \begin{bmatrix} \bar{x}_1 & \bar{x}_2 & \bar{x}_3 & 0 & 0 & 0 \\ \bar{x}_1' & \bar{x}_2' & \bar{x}_3' & 0 & 0 & 0 \\ -2S_1 & -2S_2 & -S_3 & -1 & 0 & 0 \\ -2\bar{x}_1 & -2\bar{x}_2 & -(2\bar{x}_3+1) & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \cos f & \sin f \\ 0 & 0 & 0 & 0 & -\sin f & \cos f \end{bmatrix} \tag{2.87}$$

where

$$K_2 = \left(1-e^2\right)^{-5/2}\left(\frac{1}{2}E - \frac{1}{4}\sin(2E) - \frac{e}{3}\sin^3 E\right) \tag{2.88a}$$

$$\bar{x_1} = k \sin f \tag{2.88b}$$

$$\bar{x_2} = 2e\bar{x_1}K_2 - \frac{\cos f}{k} \tag{2.88c}$$

$$\bar{x_3} = -2k \sin f K_2 - \cos^2 \frac{f}{k} - \cos^2 f \tag{2.88d}$$

$$S_1 = -\cos f \,(1 + 0.5e \cos f) \tag{2.88e}$$

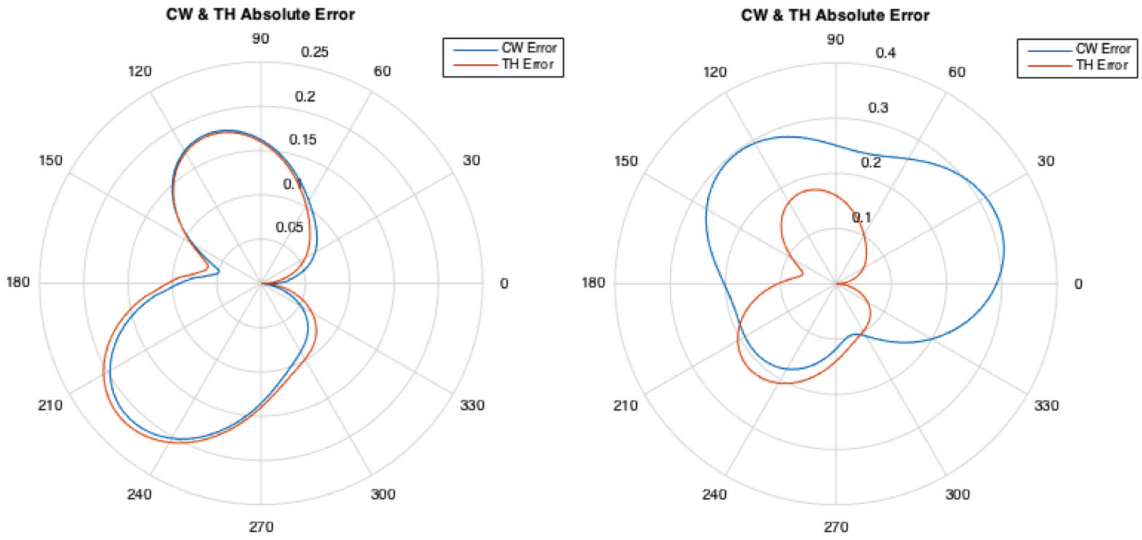
$$S_2 = 3ek^2K_2 - \frac{\sin f}{k} \tag{2.88f}$$

$$S_3 = -6k^2K_2 - \frac{2+k}{2k} \sin (2f) \tag{2.88g}$$

Another notable STM in this field is the STM by Yamanaka and Ankersen. The Yamanaka-Ankersen (YA) STM eliminates Lawden's singularity for circular orbits and is very compact (Yamanaka and Ankersen, 2002). Starting from the Lawden STM, the state vector can also be represented in arc lengths on a spherical triangle in a projection of the relative motion onto the celestial sphere, which improves accuracy, as conducted by Jiang et al. (2007). In contrast to the preceding STMs, there are also analytical solutions to the relative motion in elliptic orbits. They incorporate complex operations like solving Kepler's equation or finding derivatives of the leader's orbital elements in each time step. For more information on this class of STMs, the reader is referred to Broucke (2003) and Lee et al. (2007).

2.5.3.3 Error Determination of the Hill-Clohessy-Wiltshire and Tschauner-Hempel Dynamic Models

This section compares the HCW dynamic model and the TH model with the actual two body motion to determine the error of the linearized models. Therefore, all three models are propagated for one orbit (bounded motion provided) and the absolute error is calculated. This is performed for a wide range of relative spacecraft distances and eccentricities. Fig. 2.12 i) shows the Root Mean Square (RMS) error for one orbit period for a small leader orbit eccentricity. One can see that both dynamic models do not differ a lot. That is due to the fact that for almost zero leader orbit eccentricity both HCW and TH are identical. For an increased leader orbit eccentricity this changes as is shown in Fig. 2.12 ii). Here, the leader orbit eccentricity was increased by a factor of 20 and as HCW assumes (nearly) zero eccentricity, both models error drift apart. One can see that the error does not change for the TH model.
Also from Fig. 2.12 one can infer that the error of a dynamic model does not remain constant over one orbit period. However, for the following simulations comparing different relative distances and eccentricities the maximum error during each orbit period is always considered.

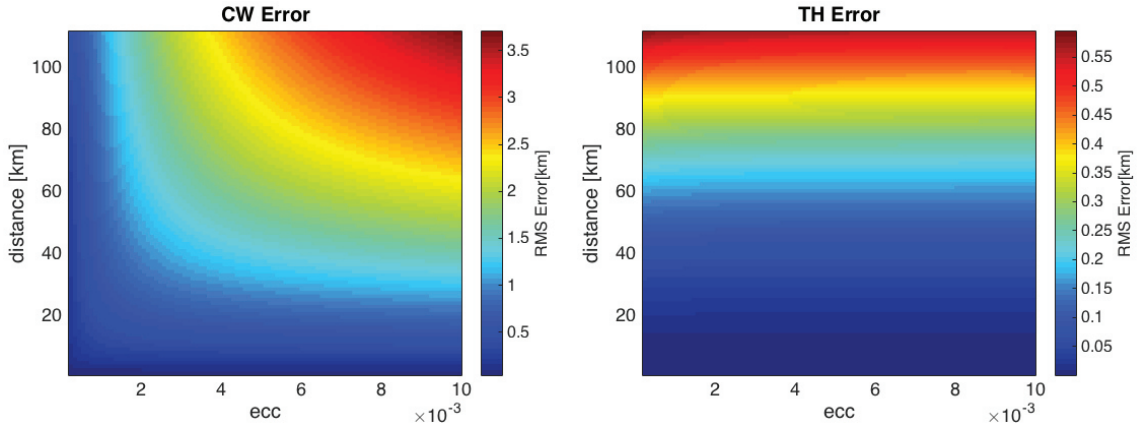


(i) $e_c = 0.00001$, $\delta r_{max} = 67\,\text{km}$ (ii) $e_c = 0.0002$, $\delta r_{max} = 67\,\text{km}$

Figure 2.12: RMS Error of HCW and TH model for two different leader orbit eccentricities as a function of mean anomaly M in [km].

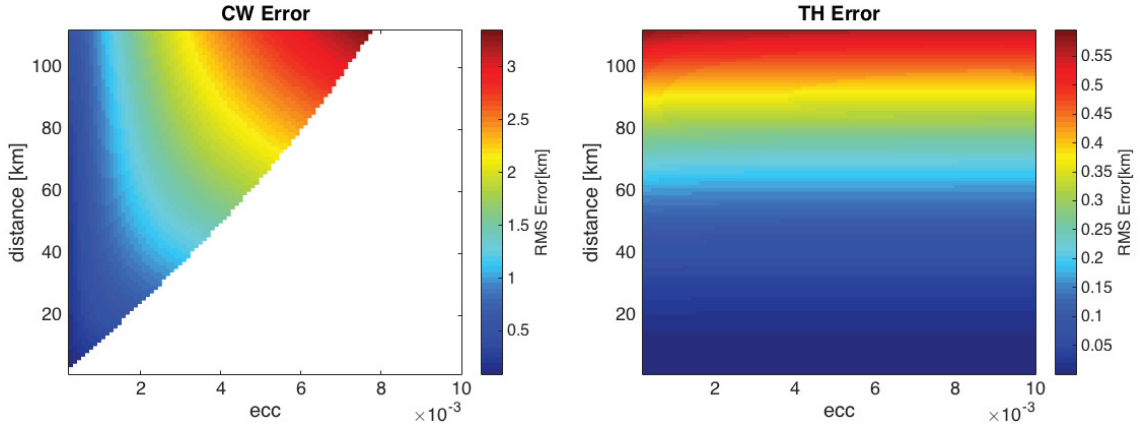
The simulations performed cover relative distances of $0 \leq \delta r_{\text{max}} \leq 110\,\text{km}$ and leader orbit eccentricities of $0 \leq e_c \leq 0.01$. The results are presented in Fig. 2.13. One can see that the HCW error increases with both leader orbit eccentricity and maximum relative distance between the spacecraft, which is reasonable as those are the simplifications made in the HCW model. The TH model on the other hand does not impose the assumption of a circular reference orbit and therefore the error of the TH model does not increase with increasing leader orbit eccentricity. Further, it can be seen that for zero leader orbit eccentricity both the HCW and the TH model show the same orbit error.

In order to find limits for the application of these models in space missions, one has to take the sensors and actuators of the spacecraft into account. As example, we assume that the sensors and actuators on board a spacecraft restrict the maximum error of the used model to 3 % of the actual distance between the spacecraft. Fig. 2.14 shows the simulated errors presented in Fig. 2.13, but with such a threshold of 3 % of the relative distance. Fig. 2.14 i) implies that the HCW model could for a given leader orbit eccentricity only be used above a minimum relative spacecraft distance. That is due to a systematic error introduced by the leader orbit eccentricity. If the relative spacecraft distance is increased, the relative error decreases, because at first the additional error introduced by the distance does not increase as fast as the relative distance. Therefore, at a certain point the relative error is smaller than 3 %. This changes quickly and an upper bound can be found for the maximum relative distance of the spacecraft (see Fig. 2.15).



(i) HCW error (ii) TH error

Figure 2.13: HCW and TH RMS model errors as a function of leader orbit eccentricity and maximum relative spacecraft distance.



(i) HCW error (ii) TH error

Figure 2.14: HCW and TH RMS model errors as a function of leader orbit eccentricity and maximum relative spacecraft distance with a threshold of 3 % of the relative distance.

As only the maximum error had been taken into account in these considerations, we have to look more carefully into detail for a given configuration and therefore this overview cannot be used as granted for any given configuration. Furthermore, there are different ways of achieving a relative distance, e.g. by just changing the inclination and keeping the eccentricity constant for both leader and follower spacecraft. By doing so, no additional eccentricity error would be introduced by increasing the relative distance between both spacecraft. Here, both δi and δe have been increased equally. That is why also an additional eccentricity error is introduced for increasing relative distance.
This computations show that the HCW model can be applied for most spacecraft formations up to hundred kilometers relative distance assuming small eccentricities.

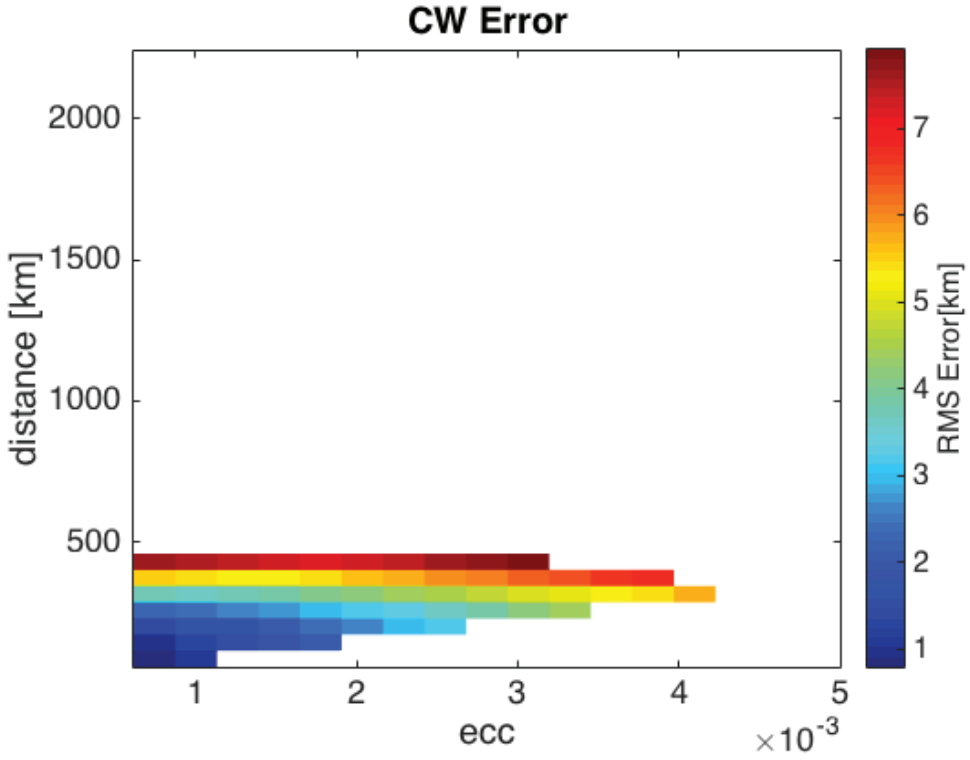


Figure 2.15: HCW RMS model error as a function of leader orbit eccentricity and maximum relative spacecraft distance with an allowed maximum relative model error of 2 %.

In that case, the HCW model is applicable and the computationally more complex TH model does not prove superior compared to it. However, with increasing eccentricity the TH model is to be favored.

2.5.3.4 State-Transition Matrices Incorporating a Non-Spherical Attraction Model

Many of the eccentricity-obeying STMs from the previous sections were developed under the motivation to find relative motion STMs for terminal rendezvous maneuvers. With this focus, the J_2-effect may obviously be neglected, since there is no differential perturbation force due to the non-spherical Earth when the spacecraft are (closely) at the same position. For formations in the size of up to 100 km relative distance, like the largest planned NetSat or TOM configurations, this simplification leads to position errors in the propagation process. Notable STMs which incorporate the J_2-effect into the solution of the HCW equation have been developed by Schweighart and Sedwick (2001) (the so called Schweighart-Sedwick STM), Morgan et al. (2012) and Djojodihardjo (2014). Since they are extensions to the HCW equations, they also neglect the eccentricity. Another approach is to use the TH equations as starting point, like in the so called Yamada STM. Yamada et al. (2012) provided an extended solution to the TH equations which solely uses osculating orbital elements and thus

allows an analytical treatment of the STM. This is achieved by splitting the state into a nominal and an osculating part, which also allows to split the STM into the nominal TH-like STM and a perturbed STM. The current osculating orbit of the leader is needed as input to the Yamada STM. This constrains its applicability, because this information may not be precisely available. A solution, where this information is not needed, is the geometric method used in the Gim-Alfriend (GA) STM (Gim and Alfriend, 2003). The method uses mean orbital elements for the leader's reference orbit and was later extended to the use of EOE (Gim and Alfriend, 2005). The state vector in orbital frame uses the NOE set introduced in Chap. 2.4.2.2. The Cartesian state vector used in the STM is, in contrast to Chap. 2.4.1, ordered as follows

$$\boldsymbol{x}\left(t\right) = \begin{pmatrix} x & \dot{x} & y & \dot{y} & z & \dot{z} \end{pmatrix}^T \tag{2.89}$$

The connection between the DOE and the current state in the Cartesian frame is described by

$$\boldsymbol{x}\left(t\right) = \boldsymbol{\Sigma}\left(t\right)\delta\boldsymbol{a}\left(t\right) = \left[\boldsymbol{A}\left(t\right) + \alpha\ \boldsymbol{B}\left(t\right)\right]\delta\boldsymbol{a}\left(t\right) \tag{2.90}$$

where $\boldsymbol{A}$ contains all terms of the unperturbed case and $\boldsymbol{B}$ all terms affected by J_2, with the leading perturbation coefficient $\alpha = 3J_2R_E^2$. The summation matrix $\boldsymbol{\Sigma}$ is obtained through a method developed by Alfriend et al. (2000), where the position and the velocity of the follower are developed in a Taylor series expansion around the leader's state and split into nominal parts and osculating parts affected by J_2. For the inverse transformation from DOE to $\boldsymbol{x}$, the inverse $\boldsymbol{\Sigma}^{-1}$ is needed. The DOE set at t is acquired with a STM for the differential elements

$$\delta\boldsymbol{a}\left(t\right) = \boldsymbol{\phi}\left(t, t_0\right)\delta\boldsymbol{a}\left(t_0\right) \tag{2.91}$$

The matrix $\boldsymbol{\phi}$ is difficult to obtain, when a high level of accuracy is to be achieved. All elements are non-linearly affected by the J_2 perturbation and lead to non-negligible linearization errors. Therefore, the DOEs are converted into mean element space, allowing to use a mean DOE STM

$$\delta\bar{\boldsymbol{a}}\left(t\right) = \bar{\boldsymbol{\phi}}\left(t, t_0\right)\delta\bar{\boldsymbol{a}}\left(t_0\right) \tag{2.92}$$

In the differential mean element set, oscillations have been filtered out (see Chap. 2.5.2.1), allowing an easier derivation of the STM. For that, the differential drift rates for θ, q_1, q_2 and Ω are calculated, starting from Eq. 2.67. Semi-major axis and inclination are not affected by this drift, leading to a slim STM for the differential mean elements. $\bar{\boldsymbol{\phi}}\left(t, t_0\right)$ is a function of the mean leader orbit and time, which are known quantities – the mean orbit of the leader can be directly propagated with Eq. 2.67. Since $\boldsymbol{\Sigma}$ is a function of the (not known) osculating leader orbit, it has to be adapted to use the mean leader orbit as input. As stated before, $\boldsymbol{A}$ is independent of the J_2 perturbation and therefore not affected by the distinction between osculating and mean orbit. Only the $\boldsymbol{B}$-matrix has to be adjusted to mean element input, leading to

$$\bar{\boldsymbol{\Sigma}}\left(t\right) = \boldsymbol{A}\left(t\right) + \alpha\ \bar{\boldsymbol{B}}\left(t\right) \tag{2.93}$$

$\bar{\boldsymbol{\Sigma}}$ is now a function of mean absolute elements, but still transforms osculating differential elements into Cartesian space. The state $\delta\bar{\boldsymbol{a}}(t)$ must be converted into differential osculating space, using the partial derivatives matrix of Eq. 2.68. The transition for differential mean element state at $t = t_0$ to current state in the Cartesian frame then becomes

$$\boldsymbol{x}(t) = \left[\boldsymbol{A}(t) + \alpha\bar{\boldsymbol{B}}(t)\right] \boldsymbol{D}(t)\, \bar{\boldsymbol{\phi}}(t, t_0)\, \delta\bar{\boldsymbol{a}}(t_0) \tag{2.94}$$

The initial state $\delta\bar{\boldsymbol{a}}(t_0)$ is obtained using the inverse matrices evaluated at $t = t_0$ from the initial Cartesian state

$$\delta\bar{\boldsymbol{a}}(t_0) = \boldsymbol{D}^{-1}(t_0) \left[\boldsymbol{A}(t_0) + \alpha\bar{\boldsymbol{B}}_{\boldsymbol{0}}(t)\right]^{-1} \boldsymbol{x}(t_0) \tag{2.95}$$

Combination of Eq. 2.94 and 2.95 leads to the complete GA STM to transition the state vector in EH frame from t_0 to t

$$\boldsymbol{\Phi}_{\boldsymbol{GA}}(t, t_0) = \left[\boldsymbol{A}(t) + \alpha\bar{\boldsymbol{B}}(t)\right] \boldsymbol{D}(t)\, \bar{\boldsymbol{\phi}}(t, t_0)\, \boldsymbol{D}^{-1}(t_0) \left[\boldsymbol{A}(t_0) + \alpha\bar{\boldsymbol{B}}_{\boldsymbol{0}}(t)\right]^{-1} \tag{2.96}$$

2.5.4 Relative Orbit Propagation in Classical Orbital Elements

When an orbital state vector is chosen, the thrust influence can be described with Gauss's Variational Equations (GVE). In the following form, they describe the derivative of the COEs with respect to time as a function of the current control acceleration $\boldsymbol{u}$ (Battin, 1999, p. 489)

$$\dot{\boldsymbol{a}}_C(t) = \boldsymbol{A}\boldsymbol{x}(t) + \boldsymbol{B}(t)\, \boldsymbol{u}(t) \tag{2.97}$$

written in complete form with matrix elements

$$\frac{d}{dt}\begin{pmatrix} a \\ e \\ i \\ \omega \\ \Omega \\ M \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ n \end{pmatrix} + \begin{bmatrix} \frac{2a^2 p}{rh} & 0 & -\frac{2a^2 e \sin f}{h} \\ \frac{(p+r)\cos f + re}{h} & 0 & -\frac{p \sin f}{h} \\ 0 & -\frac{r\cos\theta}{h} & 0 \\ \frac{(p+r)\sin f}{he} & +\frac{r \sin\theta \cos i}{h \sin i} & +\frac{p\cos f}{he} \\ 0 & -\frac{r\sin\theta}{h\sin i} & 0 \\ -\frac{b(p+r)\sin f}{ahe} & 0 & -\frac{b(p\cos f - 2re)}{ahe} \end{bmatrix} \begin{pmatrix} u_x \\ u_y \\ u_z \end{pmatrix} \tag{2.98}$$

Eq. 2.98 is assuming the perturbation-free case, since only the mean anomaly is subject to change by the mean motion. When thrust influence has to be modeled in perturbed environments, the state-space equations have to be adapted. This does not alter the control influence matrix $\boldsymbol{B}$, thus it comprises a sufficient thrust impact description for different kinds of state-space equations.

2.5.5 Relative Orbit Propagation in Relative Orbit Elements

Another possible way of propagating a relative motion is using ROE (cf. Chap. 2.4.2.4). The full state-space model in ROE consists of three control inputs and seven outputs. The input vector $\bar{\boldsymbol{u}} = (u_x, u_y, u_z)^{\mathrm{T}}$ comprises the control accelerations into radial, tangential and normal direction of the EH orbital frame. The output is formed by the ROE vector $\partial\bar{\boldsymbol{\alpha}}$ augmented with the time derivative of the relative semi-major axis $\partial\dot{a}$ in order to account for differential drag. The state-space model is given by

$$\begin{bmatrix} \partial\dot{\bar{\boldsymbol{\alpha}}} \\ \partial\ddot{a} \end{bmatrix} = \bar{\boldsymbol{A}}_c \begin{bmatrix} \partial\bar{\boldsymbol{\alpha}} \\ \partial\dot{a} \end{bmatrix} + \begin{bmatrix} \bar{\boldsymbol{B}}_c \\ \boldsymbol{0}^{1x3} \end{bmatrix} \bar{\boldsymbol{u}} \tag{2.99}$$

The plant matrix $\bar{\boldsymbol{A}}_c = \bar{\boldsymbol{A}}_{J_2} + \bar{\boldsymbol{A}}_{\mathrm{drag}} + \bar{\boldsymbol{A}}_{\mathrm{K}}$ takes the differential Earth oblateness effect $\bar{\boldsymbol{A}}_{J_2}$, differential drag $\bar{\boldsymbol{A}}_{\mathrm{drag}}$ and Keplerian motion $\bar{\boldsymbol{A}}_{\mathrm{K}}$ into account (Koenig et al., 2016). The complete plant matrix for the augmented state is given by

$$\bar{\boldsymbol{A}}_c = \kappa \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 1/\kappa \\ -\frac{7}{2}EP - \frac{3}{2}\frac{n_c}{\kappa} & 0 & e_x GFP & e_y GFP & -FS & 0 & 0 \\ \frac{7}{2}e_y Q & 0 & -(4e_x e_y G + C)Q & -(1 + 4e_y^2 G - D)Q & 5e_y S & 0 & \frac{D - e_x}{\kappa} \\ -\frac{7}{2}e_x Q & 0 & (1 + 4e_x^2 G - D)Q & (4e_x e_y G - C)Q & -5e_x S & 0 & \frac{C - e_y}{\kappa} \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \frac{7}{2}S & 0 & -4e_x GS & -4e_y GS & 2T & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \tag{2.100}$$

where n_c is the mean motion. The parameters of the plant matrix are given by Koenig et al. (2016) as

$$\gamma = \frac{3}{4} J_2 R_e^2 \sqrt{\mu}, \qquad \eta_c = \sqrt{1 - e_c^2}, \qquad \kappa = \frac{\gamma}{a_c^{7/2} \eta_c^4}, \qquad e_x = e_c \cos\omega_c, \qquad e_y = e_c \sin\omega_c,$$

$$C = \sin\omega_c, \qquad D = \cos\omega_c, \qquad E = 1 + \eta_c, \qquad F = 4 + 3\eta_c, \qquad G = \frac{1}{\eta_c^2},$$

$$P = 3\cos(i_c)^2 - 1, \qquad Q = 5\cos(i_c)^2 - 1, \qquad S = \sin(2i_c), \qquad T = \sin(i_c)^2$$

To account for differential drag, the density-model-free formulation by Koenig et al. (2016) is implemented into $\bar{\boldsymbol{A}}_c$. It is essential to properly initialize $\partial\dot{a}$ by in-situ measurements of $\partial\dot{a}_{\mathrm{drag}}$ by tracking the difference over a few orbits without any

applied maneuvers during that time. The control input matrix $\bar{\boldsymbol{B}}_c \in \mathbb{R}^{6x3}$ is given by Chernick and D'Amico (2016) as

$$\bar{\boldsymbol{B}}_c = \frac{1}{a_c n_c} \begin{bmatrix} \frac{2}{\eta_c} e_c \sin f_c & \frac{2}{\eta_c}(1 + e_c \cos f_c) & 0 \\ -\frac{2\eta_c^2}{1+e_c \cos f_c} & 0 & 0 \\ \eta_c \sin(\omega_c + f_c) & \eta_c \frac{(2+e_c \cos f_c)\cos(\omega_c+f_c)+e_x}{1+e_c \cos f_c} & \frac{\eta_c e_y}{\tan i_c} \frac{\sin(\omega_c+f_c)}{1+e_c \cos f_c} \\ -\eta_c \cos(\omega_c + f_c) & \eta_c \frac{(2+e_c \cos f_c)\sin(\omega_c+f_c)+e_y}{1+e_c \cos f_c} & \frac{-\eta_c e_x}{\tan i_c} \frac{\sin(\omega_c+f_c)}{1+e_c \cos f_c} \\ 0 & 0 & \eta_c \frac{\cos(\omega_c+f_c)}{1+e_c \cos f_c} \\ 0 & 0 & \eta_c \frac{\sin(\omega_c+f_c)}{1+e_c \cos f_c} \end{bmatrix} \tag{2.101}$$

where f_c is the true anomaly.

2.5.5.1 Reduced Model

From the control input matrix $\bar{\boldsymbol{B}}_c$ in Eq. 2.101 one can see that changes of the relative eccentricity vector $\partial \boldsymbol{e}$ and the relative semi-major axis ∂a are most efficiently achieved by applying tangential thrust only. This suggests controlling the system given by Eq. 2.99 without radial thrust. Furthermore, the ΔV lower bound for impulsive maneuvering is yielded using tangential thrust only for in-plane reconfigurations (Chernick and D'Amico, 2016, Gaias and D'Amico, 2015). On the other hand, $\partial\lambda$ can only be controlled directly by radial thrust and therefore waiving radial thrust would mean the loss of full controllability. In this situation, it is convenient to use a reduced model and to control $\partial\dot{\lambda}$ by leveraging Keplerian dynamics. Such a reduced model does not contain the mean along-track separation as given by the relative mean longitude $\partial\lambda$ so that that the reduced ROE vector is defined as $\partial\boldsymbol{\alpha} = (\partial a, \partial e_x, \partial e_y, \partial i_x, \partial i_y)^{\mathrm{T}}$. Furthermore, the control input for the reduced model is defined by the accelerations in along-track and normal direction of the EH frame $\boldsymbol{u} = (u_x, u_y)^{\mathrm{T}}$. The reduced model is given by

$$\begin{bmatrix} \partial\dot{\boldsymbol{\alpha}} \\ \partial\ddot{a} \end{bmatrix} = \boldsymbol{A}_c \begin{bmatrix} \partial\boldsymbol{\alpha} \\ \partial\dot{a} \end{bmatrix} + \begin{bmatrix} \boldsymbol{B}_c \\ \boldsymbol{0}^{1x2} \end{bmatrix} \boldsymbol{u} \tag{2.102}$$

The reduced plant matrix is given by

$$\boldsymbol{A}_c = \kappa \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 1/\kappa \\ \frac{7}{2}e_y Q & -(4e_x e_y G + C)Q & -(1 + 4e_y^2 G - D)Q & 5e_y S & 0 & \frac{D-e_x}{\kappa} \\ -\frac{7}{2}e_x Q & (1 + 4e_x^2 G - D)Q & (4e_x e_y G - C)Q & -5e_x S & 0 & \frac{C-e_y}{\kappa} \\ 0 & 0 & 0 & 0 & 0 & 0 \\ \frac{7}{2}S & -4e_x GS & -4e_y GS & 2T & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \tag{2.103}$$

with the same parameters as for Eq. 2.103. Note that Keplerian dynamics have no influence on $\partial\boldsymbol{\alpha}$. The reduced control input matrix thus is

$$\boldsymbol{B}_c = \frac{1}{a_c n_c} \begin{bmatrix} \frac{2}{\eta_c}(1 + e_c \cos f_c) & 0 \\ \eta_c \frac{(2+e_c \cos f_c)\cos(\omega_c+f_c)+e_x}{1+e\cos f_c} & \frac{\eta_c e_y}{\tan i_c}\frac{\sin(\omega_c+f_c)}{1+e_c\cos f_c} \\ \eta_c \frac{(2+e_c \cos f_c)\sin(\omega_c+f_c)+e_y}{1+e_c\cos f_c} & \frac{-\eta_c e_x}{\tan i_c}\frac{\sin(\omega_c+f_c)}{1+e_c\cos f_c} \\ 0 & \eta_c \frac{\cos(\omega_c+f_c)}{1+e_c\cos f_c} \\ 0 & \eta_c \frac{\sin(\omega_c+f_c)}{1+e_c\cos f_c} \end{bmatrix} \tag{2.104}$$

De Florio et al. (2014) implement the above-mentioned leveraging of Keplerian dynamics by augmenting the state with $\partial\dot{\lambda}$ and a proper in-orbit placement of the impulsive along-track maneuvers. A different method is to change the applied reference of the relative semi-major axis that will then be followed by a stabilizing feedback controller to maintain full controllability of the complete ROE set $\partial\bar{\boldsymbol{\alpha}}$. This approach is presented in Chap. 3.4.

2.6 Satellite Formation Topologies

A satellite formation consists of one reference (or leader) satellite and multiple follower satellites. When all satellites need to be treated equally, a virtual leader satellite can be introduced. Then this virtual leader is located at the geometric center of the formation. We deliberately use the term topology here instead of topography, because the subsequent description is about the relative arrangement of satellites with respect to each other. In this description we do not require any specific reference or coordinate system and exact relative distances are not of importance either. All formations that are presented here are so-called force-free formations. This means that (without taking perturbations into account) they remain stable over time without the need of applying control forces to maintain them.
There are various formation topologies, some of which are described in this section. They can be applied to different applications: An Along Track Formation (ATF) or a Projected Circular Orbit (PCO) can be used as SAR, as optical interferometry or for on-orbit servicing of other spacecraft, as examined in the CanX-4 & 5 mission (Bonin et al., 2015) and planned e.g. for the CloudCT mission (Schilling et al., 2019). Furthermore, a Circular Orbit (CO) or PCO can conduct gravitational and magnetic filed analysis and ground-moving target indication. Another well-known formation type is the Cartwheel Formation (CWF), whose application can be the generation of a Digital Elevation Model (DEM) (e.g. as planned in the TOM mission (Schilling et al., 2017)) or gravity field determination.
A formation is characterized by its above-mentioned type and its size. The size parameter is also called baseline and represents the average distance between the outermost satellites (Tetrahedron Formation (THF), ATF) or the diameter of the relative ellipses (PCO, PCO and CWF). Focus will be directed to ATF, CO, PCO, CWF and Tetrahedron Formation (THF) in that order. They describe one-, two- and three-dimensional configurations, allowing to investigate different levels of relative orbit complexity. Without loss of generality, all formation types will be described for four satellites. To design multi-satellite configurations for a different satellite quantity, the respective spacing parameters can easily be adapted.
In general, the in-plane motion in the EH frame $\mathscr{H}$ is decoupled from the out-of-plane motion. The force-free in-plane motion comprises a 2 by 1 ellipse (cf. Fig. 2.16) while the force-free out-of-plane motion forms a pendulum (cf. Fig. 2.17). Most of the well-known and in this chapter described formation topologies are a combination of these motions.

2.6.1 Initial Acquisition Phase

After deployment of the satellites from the launcher, a slight difference in semi-major axes can occur. In order to ensure a stable formation with bounded motion, i.e. all orbits having the same period, this mismatch has to be canceled out. Thus, the satellites need to perform an orbital maneuver in order to acquire the simplest configuration, namely ATF (see Chap. 2.6.2). This can be divided into first acquiring the same semi-major axes to counteract the initial drift as fast as possible, followed

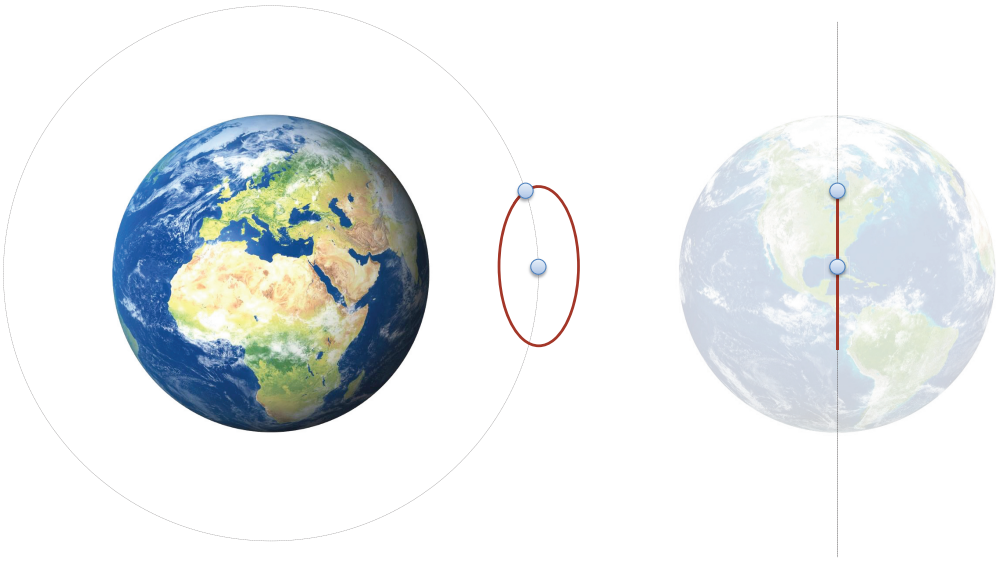


Figure 2.16: In-plane motion of an arbitrary formation

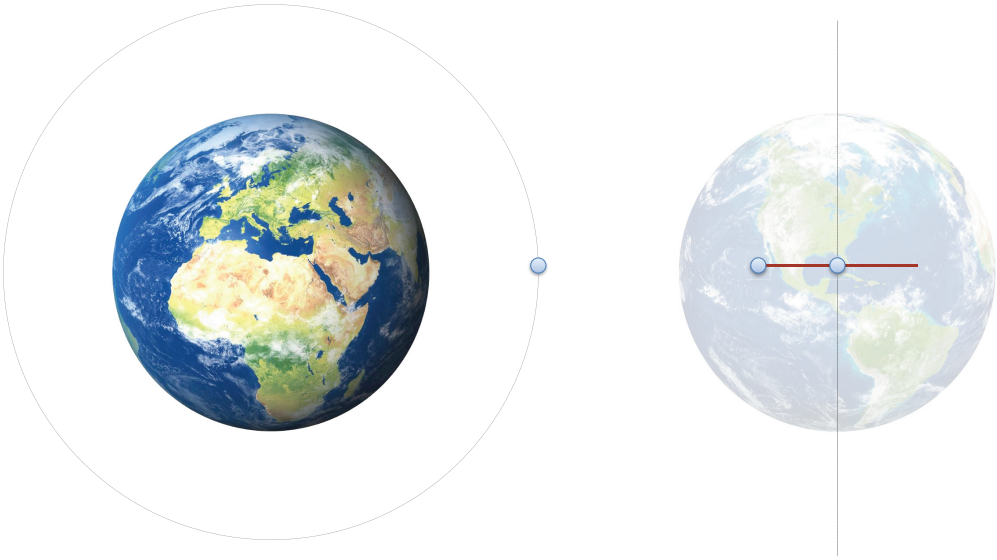


Figure 2.17: Out-of-plane motion of an arbitrary formation

by the acquisition of an ATF. According to *Innovative Solutions In Space BV*, a well-known launch provider for nanosatellites, the maximal drift among two satellites after deployment does not exceed $0.1\,\mathrm{m\,s^{-1}}$, while the major part of it is in along-track direction. Therefore, with a very rough estimation of the initial formation behavior (the initial orbit is assumed to be the nominal NetSat orbit with $a \approx 7000\,\mathrm{km}$ and $e \ll 1$), all six ROEs are zero except the difference in mean argument of latitude $\partial\varphi$, which can be derived from Eq. 2.37. Assuming a non-controllable initial phase of four weeks (which has been the case in the preceding UWE-4 mission) which is blocked

for LEOP and commissioning, the maximum separation in along-track direction ($\partial\phi$ or $\hat{\boldsymbol{e}}_x$ in the EH frame $\mathscr{H}$) at the beginning of the controllable phase is assumed to be around 360 km with the remaining satellites spaced equally in between. This coarse estimation roughly reflects a 360 km ATF configuration, which appears to be the natural starting point of a formation flying mission.

2.6.2 Along-Track Formation

The ATF is the simplest configuration of a formation and is often called String of Pearls. In this formation, two or more satellites are placed into the same orbit and separated by a specific distance along the orbit (cf. Fig. 2.18).

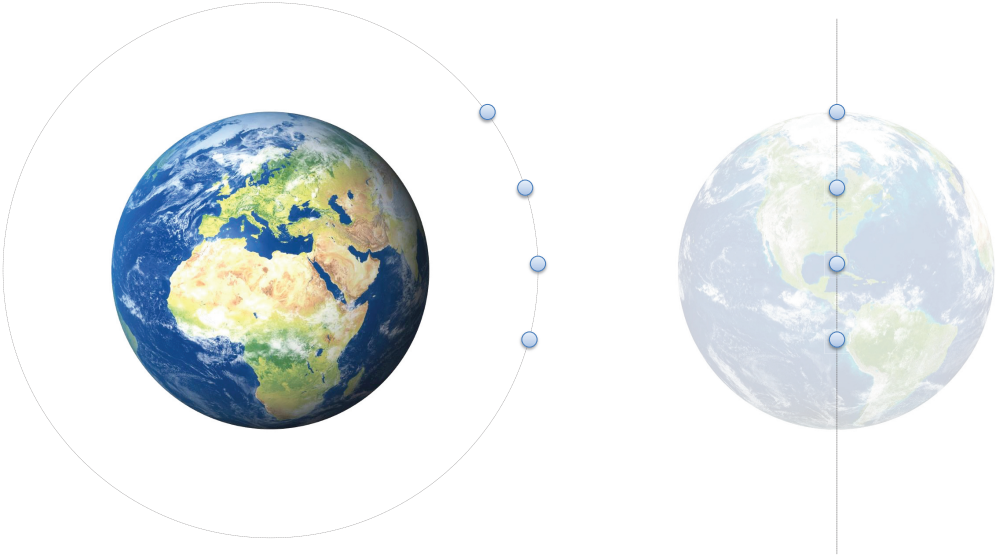


Figure 2.18: Along Track Formation

Since the satellites are separated along the same orbital track, only the argument of latitude $a\partial\varphi \neq 0$, or equivalently the relative mean longitude $\partial\lambda \neq 0$. For formation design, it is useful to multiply the ROE vector by a, so all entries of the state vector share the dimension $[m]$. This gives a direct insight into inter-satellite distances in EH frame as presented in Fig. 2.19 and simplifies formation design. Following the definition of Eq. 2.35, the ATF is described by

$$a\,\partial\boldsymbol{\alpha}_k = a \begin{pmatrix} \partial a \\ \partial\varphi \\ \partial e_x \\ \partial e_y \\ \partial i_x \\ \partial i_y \end{pmatrix} = a \begin{pmatrix} 0 \\ \partial\varphi_{k-1} + \delta r_{\max}/n \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix} \tag{2.105}$$

where index k denotes the follower satellites from 1 to n. The ROE state vector for the (arbitrary) leader satellite is the isotropic vector, resulting in $\partial\varphi = \partial\lambda = 0$. δr_{max} is the maximum distance between the satellites.

Since all satellites share the same orbit, ideally there is no relative movement between the satellites. Viewed in an idealized EH frame, the $\boldsymbol{p}$ vector is constant and the $\boldsymbol{v}$ is zero. As $\partial\lambda/\partial\varphi$ solely impacts the $\hat{\boldsymbol{e}}_{\boldsymbol{x}}$ coordinate, the $\hat{\boldsymbol{e}}_{\boldsymbol{y}}$ and $\hat{\boldsymbol{e}}_{\boldsymbol{z}}$ components of $\boldsymbol{p}$ are always zero in a perfect ATF configuration. In a real EH frame, however, there will always be some movement in $\hat{\boldsymbol{e}}_{\boldsymbol{y}}$ and $\hat{\boldsymbol{e}}_{\boldsymbol{z}}$ direction. This is due to the linearization error of the $\mathscr{H}$ frame definition (Chap. 2.4.1.2).

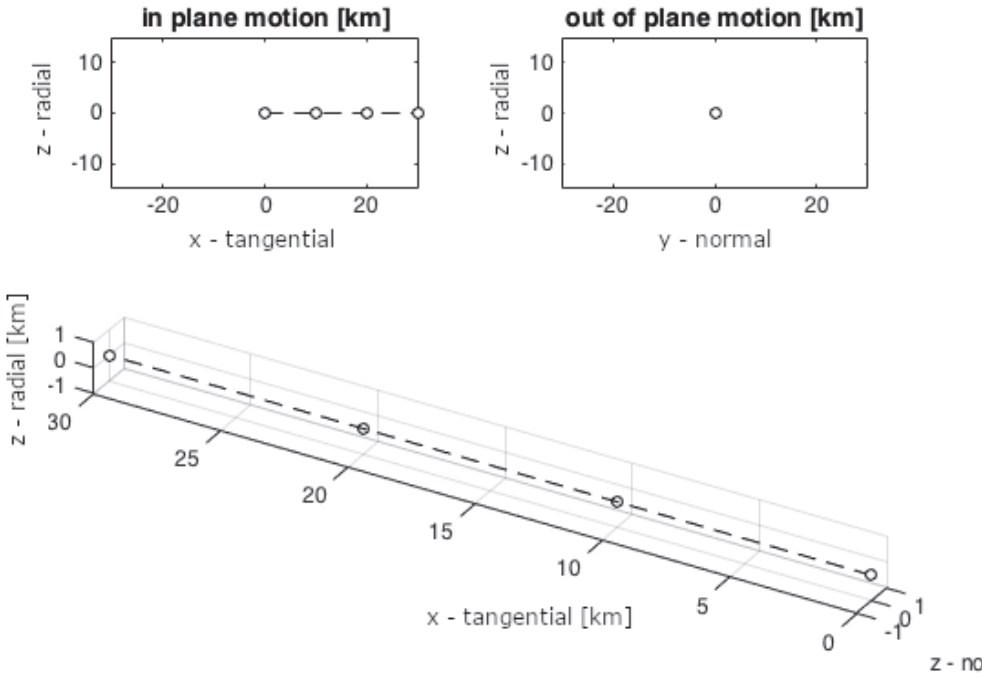
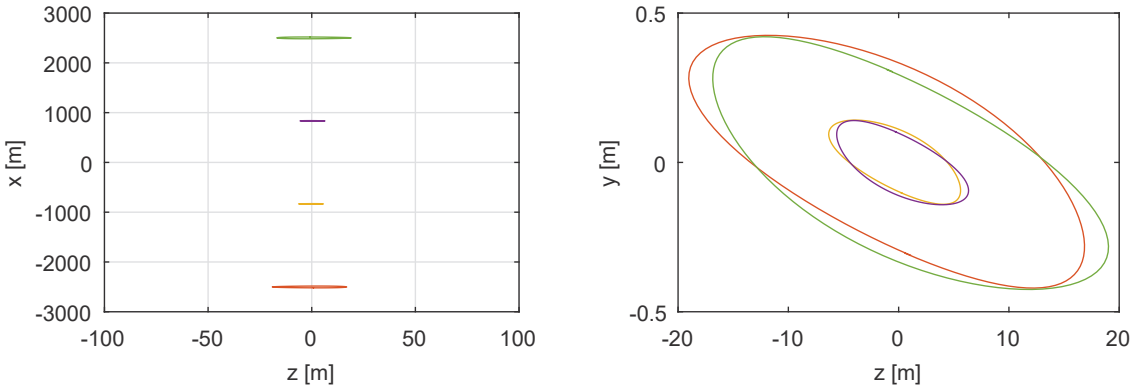


Figure 2.19: Along Track Formation

Fig. 2.20 shows the relative trajectories of the satellites in a 5 km ATF configuration over one orbital period. The formation center was chosen, in contrast to the definition above, to be in the middle of the two outermost satellites. This reduces the linearization error by a factor of two. Fig. 2.20i shows the relative in-plane motion, where the 5 km baseline can be seen in $\hat{\boldsymbol{e}}_{\boldsymbol{x}}$ direction. The linearization error caused by the curvature of the orbit occurs mainly radially, while a cross-track movement is almost non-existent (Fig. 2.20ii). The corresponding state vectors in ROE frame for the described formation are given in Tab. 2.4. Since (ideally) there is no relative movement as well as a separation in only one direction, ATF is a 1D formation.



(i) In-plane motion (ii) Out-of-plane motion

Figure 2.20: Relative satellite motion of a 5 km ATF configuration.

Table 2.4: State vectors for a 5 km ATF configuration in ROEs. All values are given in $[m]$.

Satellite	1	2	3	4
∂a	0	0	0	0
$\partial\varphi$	-2500	-833	833	2500
∂e_x	0	0	0	0
∂e_y	0	0	0	0
∂i_x	0	0	0	0
∂i_y	0	0	0	0

2.6.3 Circular Orbit Formation

The next formation type to investigate is the CO formation, which is formed by a circular relative orbit of a follower satellite (cf. Fig. 2.21). Such a circular reference orbit is defined by the relative velocity being constant at every time instance. In order to yield velocities we take the derivative of Eq. 2.37 and normalize by the velocity of the leader spacecraft

$$v_x/v = 2\delta e\cos(\varphi - \phi) \tag{2.106a}$$

$$v_y/v = -\delta i\cos(\varphi - \theta) \tag{2.106b}$$

$$v_z/v = -\delta e\sin(\varphi - \phi) \tag{2.106c}$$

The absolute velocity of the follower is given by

$$v_d = \sqrt{v^2(\delta v_x^2 + \delta v_y^2 + \delta v_z^2)} \tag{2.107}$$

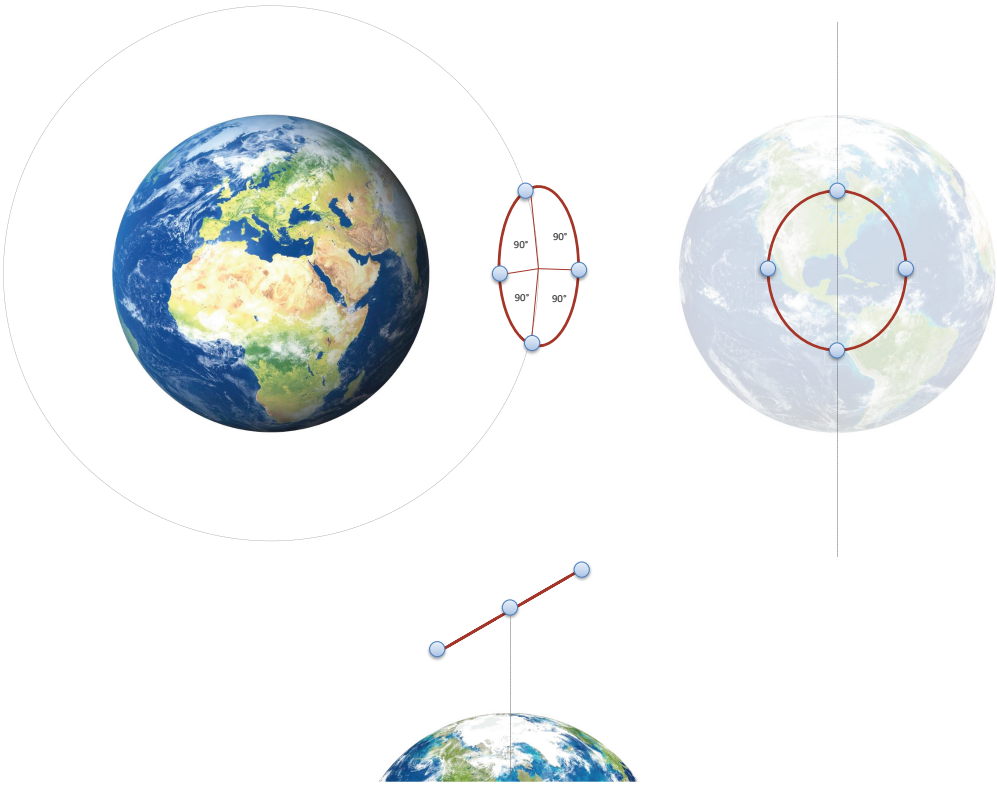


Figure 2.21: Circular Orbit formation

In order to show that v_d is constant it is sufficient to analyze only the radicand R

$$R = \delta v_x^2 + \delta v_y^2 + \delta v_z^2 \tag{2.108}$$

$$= 4\partial e^2 \cos^2(\varphi - \phi) + \delta i^2 \cos^2(\varphi - \theta) + \delta e^2 \sin^2(\varphi - \phi) \tag{2.109}$$

For this expression to stay constant one has to find a set of δe, δi, ϕ and θ for which the derivative of the above becomes zero. The derivative is given by

$$\frac{\delta R}{\delta \varphi} = -8\delta e^2 \cos(\varphi - \phi)\sin(\varphi - \phi) - 2\delta i^2 \cos(\varphi - \theta)\sin(\varphi - \theta)$$

$$+ 2\delta e^2 \sin(\varphi - \phi)\cos(\varphi - \phi) \tag{2.110}$$

This expression can only be solved by rendering the sines and cosines to zero. This can be achieved by setting $\phi - \theta = 0$ or $\phi - \theta = \pm 90°$. As the solution for $\phi - \theta = 0$ will result in negative values for either δe or δi, only the solution for $\phi - \theta = \pm 90°$ is valid and Eq. 2.110 becomes

$$\frac{\delta R}{\delta \varphi} = -8\delta e^2 \cos(\varphi)\sin(\varphi) + 2\delta i^2 \cos(\varphi)\sin(\varphi)$$

$$+ 2\delta e^2 \cos(\varphi)\sin(\varphi) \tag{2.111}$$

After dividing by $\cos(\varphi)\sin(\varphi)$ and setting to zero we find the constraint

$$0 = -8\delta e^2 + 2\delta i^2 + 2\delta e^2 \tag{2.112}$$

$$2\delta i^2 = 6\delta e^2 \tag{2.113}$$

$$\delta i = \sqrt{3}\,\delta e \tag{2.114}$$

Thus, two solutions are found that generate a CO in the co-rotating frame of the leader's orbit with $\phi - \theta = \pm 90°$ and $\delta i = \sqrt{3}\,\delta e$. In the EH frame the motion of a follower in such a CO can be expressed as dynamic state vector with the free design parameter x_0

$$\boldsymbol{x} = \begin{pmatrix} \text{arbitrary} & 0 & 0 & 0 & x_0\sqrt{\frac{3}{4}}\omega & x_0\frac{1}{2}\omega \end{pmatrix}^T \tag{2.115}$$

An example of a CO formation in EH frame is depicted in Fig. 2.22.

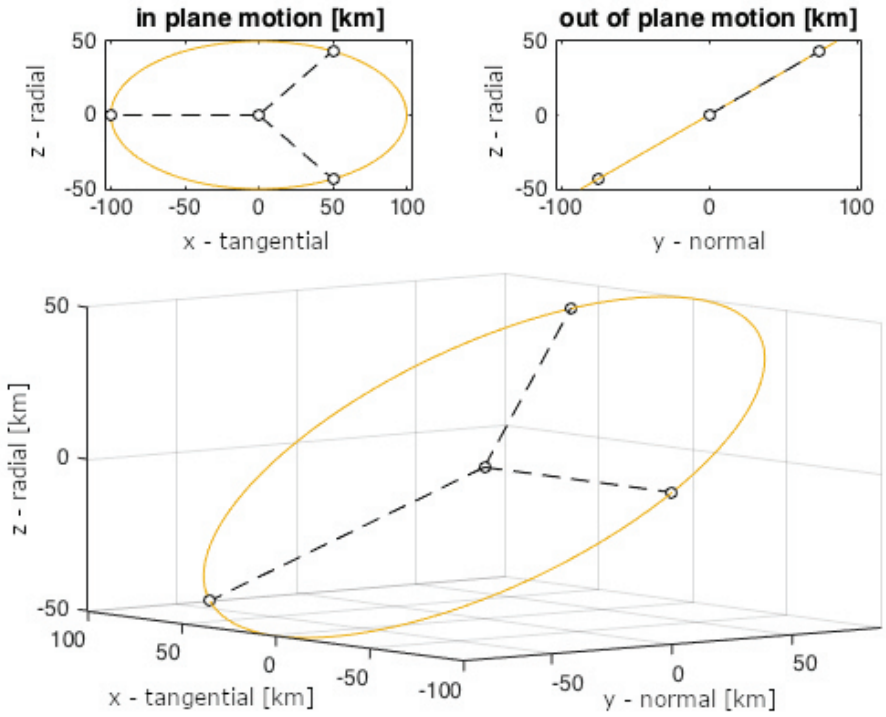


Figure 2.22: Circular Orbit formation

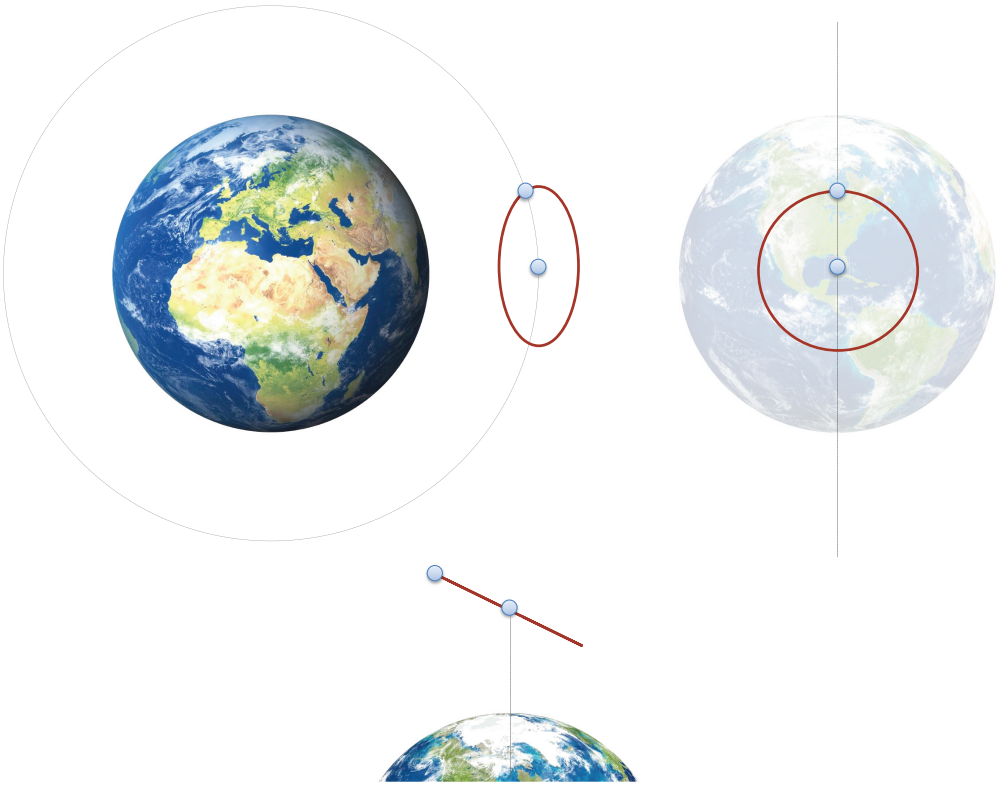


Figure 2.23: Projected Circular Orbit

2.6.4 Projected Circular Orbit Formation

A Projected Circular Orbit (PCO) formation is formed by a circular motion of follower satellite(s) around the (virtual) leader, as seen from the Earth (cf. Fig. 2.23). The follower orbits define a specific ellipse in the EH frame $\mathscr{H}$ which transforms into a circle when projected into the $\hat{\boldsymbol{e}}_{\boldsymbol{x}}$-$\hat{\boldsymbol{e}}_{\boldsymbol{y}}$ plane. The ellipse, on which all followers are located, is tilted by δi. Thus, a straight line is visible in $\hat{\boldsymbol{e}}_{\boldsymbol{y}}$-$\hat{\boldsymbol{e}}_{\boldsymbol{z}}$ plane, as can be seen in Fig. 2.24ii. Fig. 2.24i shows the 2-by-1 in-plane ellipse with four equally spaced followers. The dotted lines depict the instantaneous position vectors of the satellites. The follower orbits of a PCO are defined by the relative velocities in $\hat{\boldsymbol{e}}_{\boldsymbol{y}}$-$\hat{\boldsymbol{e}}_{\boldsymbol{z}}$ plane, whose magnitudes are constant at every time instant. The two mathematical requirements to describe a PCO formation can be inferred from Eq. 2.40 by deploying Eq. 2.39. We consider the relevant velocities in $\hat{\boldsymbol{e}}_{\boldsymbol{x}}$-$\hat{\boldsymbol{e}}_{\boldsymbol{y}}$ direction

$$\delta v_x / v = 2\delta e \cos(\varphi - \phi) \quad , \tag{2.116a}$$

$$\delta v_y / v = -\delta i \cos(\varphi - \theta) \quad . \tag{2.116b}$$

The absolute projected velocity of the follower is given by

$$v_d = \sqrt{v^2(\delta v_x^2 + \delta v_y^2)} \tag{2.117}$$

In order to show v_d is constant it is sufficient to only analyze the radicand R

$$R = \delta v_x^2 + \delta v_y^2 \tag{2.118}$$

$$= 4\delta e^2 \cos^2(\varphi - \phi) + \delta i^2 \cos^2(\varphi - \theta) \tag{2.119}$$

For this expression to stay constant one has to find a set of δe, δi, ϕ and θ for which the derivative of the above becomes zero. The derivative is given by

$$\frac{\delta R}{\delta \varphi} = -8\delta e^2 \cos(\varphi - \phi)\sin(\varphi - \phi)$$

$$- 2\delta i^2 \cos(\varphi - \theta)\sin(\varphi - \theta) \tag{2.120}$$

This expression can only be solved by eliminating the sines and cosines. This can be achieved by setting $\phi - \theta = 0$ or $\phi - \theta = \pm 90°$. As the solution for $\phi - \theta = 0$ will result in negative values for either δe or δi, only the solution for $\phi - \theta = \pm 90°$ is valid and Eq. 2.120 becomes

$$\frac{\delta R}{\delta \varphi} = -8\delta e^2 \cos(\varphi)\sin(\varphi) + 2\delta i^2 \cos(\varphi)\sin(\varphi). \tag{2.121}$$

After dividing by $\cos(u)\sin(u)$ and setting to zero we find the constraint

$$0 = -8\delta e^2 + 2\delta i^2 \tag{2.122}$$

$$2\delta i^2 = 8\delta e^2 \tag{2.123}$$

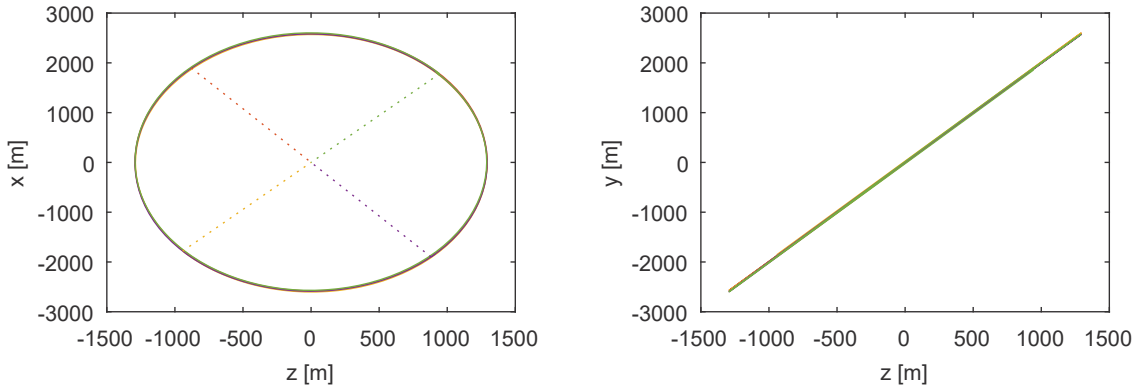
$$\delta i = 2\,\delta e \tag{2.124}$$

Thus, two solutions are found to generate a PCO in the EH frame of the leader's orbit with $\phi - \theta = \pm 90°$ and $\delta i = 2\,\delta e$.

The angle offsets between the four follower satellites on the common circumferential ellipse is $\Delta\phi = {}^1/_2\pi$. The general ROE definition for a PCO formation is therefore

$$a_0 \partial \boldsymbol{a}_k = a \begin{pmatrix} \partial a_k \\ \partial \varphi_k \\ \partial (e_x)_k \\ \partial (e_y)_k \\ \partial (i_x)_k \\ \partial (i_y)_k \end{pmatrix} = \begin{pmatrix} 0 \\ -\dfrac{\partial (i_y)_k}{\tan i_0} \\ a_0 \tau \cos(\gamma_{k-1} + 0.5\pi) \\ a_0 \tau \sin(\gamma_{k-1} + 0.5\pi) \\ 2a_0 \tau \cos(\gamma_{k-1} + \pi) \\ 2a_0 \tau \sin(\gamma_{k-1} + \pi) \end{pmatrix} \tag{2.125}$$

where index $k \in \{1, 2, 3\}$ denotes the follower satellites, τ is the free variable to define the size of the PCO formation and γ distributes the satellites equally, while



(i) In-plane motion (ii) Out-of-plane motion

Figure 2.24: Relative satellite motion of a 5 km PCO configuration.

γ_0 is the second arbitrary design parameter. The relative argument of latitude $\partial\varphi_k$ is defined according to Eq. 2.38c to place the formation center (virtual leader) in the very center of the ellipse. The product $a_0\tau$ directly corresponds to the formation baseline.

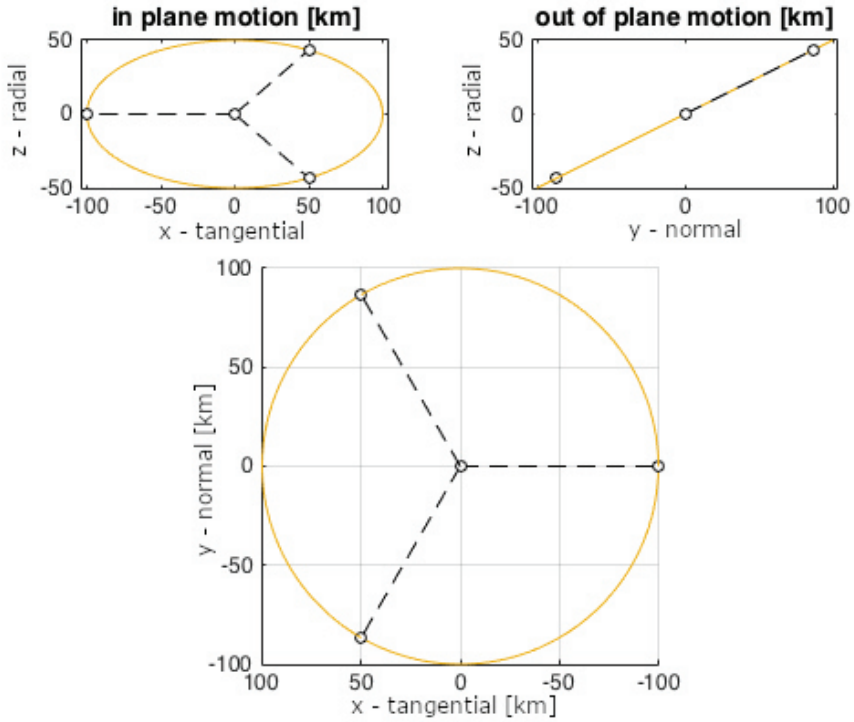


Figure 2.25: Projected Circular Orbit formation

A PCO formation can also be established with three satellites on an ellipse and a leader in the focus as depicted in Fig. 2.25. The state vector of this PCO variant is of the same nature as Eq. 2.125. Only the spacing angle has to be adapted

to $\Delta\phi = {}^{2}/_{3}\pi$ and the leader is located at $r = (0\ \ 0\ \ 0)^T$. In a PCO configuration, all satellites share a common plane of movement, making PCO a two-dimensional formation. The exact state vector for the formation depicted in Fig. 2.24 is stated in Tab. 2.5.

Table 2.5: State vectors for a 5 km PCO configuration in ROEs. All values are given in $[m]$.

Satellite	1	2	3	4
∂a	0	0	0	0
$\partial\varphi$	349	0	−349	0
∂e_x	1291	0	−1291	0
∂e_y	0	1291	0	−1291
∂i_x	0	−2582	0	2582
∂i_y	2582	0	−2582	0

2.6.5 Cartwheel Formation – Moving Plane

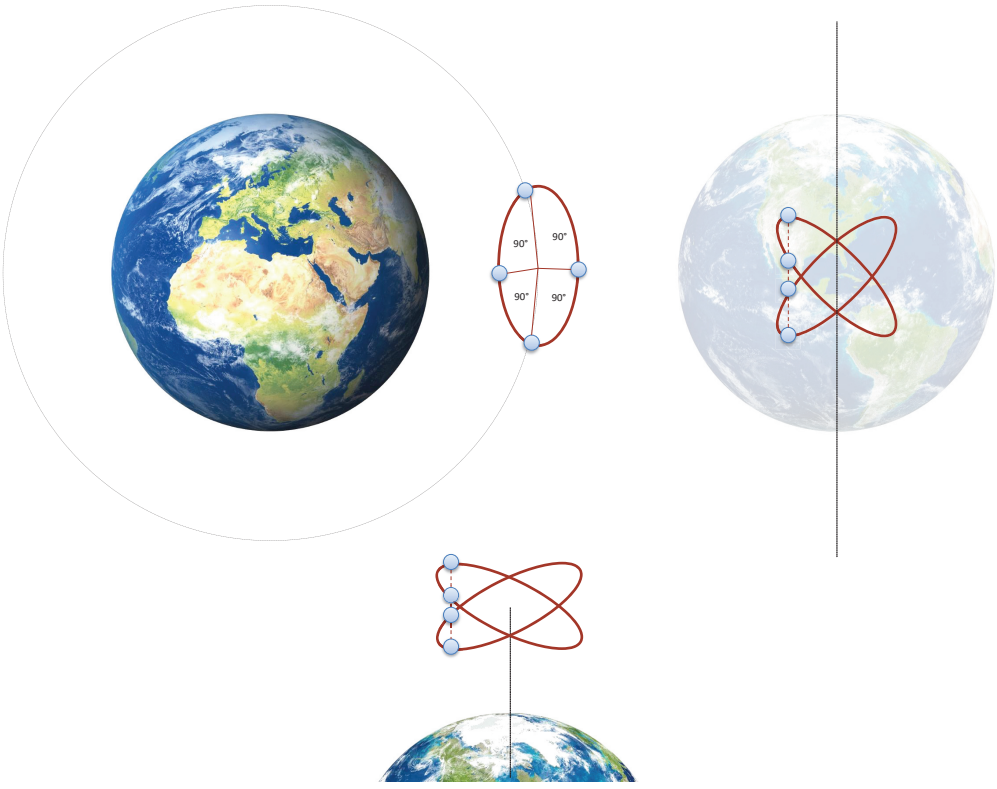


Figure 2.26: Cartwheel Formation – Moving Plane

The Cartwheel Formation (CWF) – Moving Plane describes a 3D formation, in which the follower satellites orbit around a (virtual) leader on differently inclined ellipses

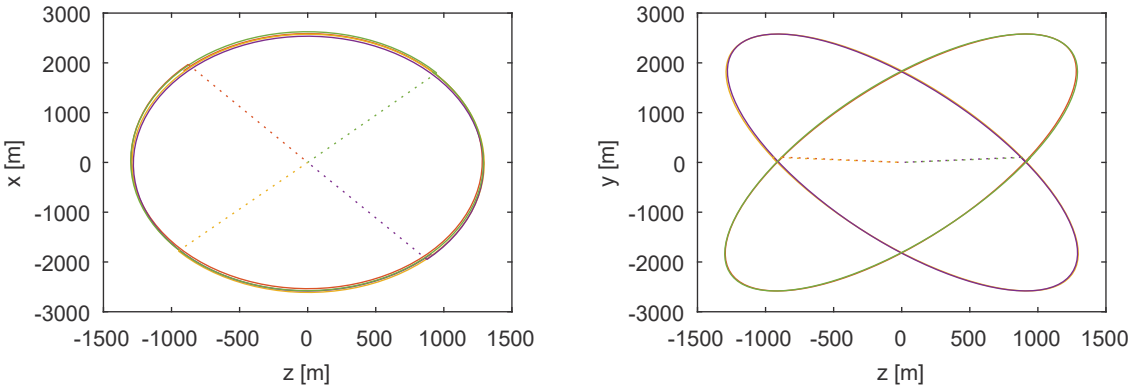
as depicted in Fig. 2.26. The instantaneous positions of the deputies define a plane, which is always parallel to the orbit plane. By this, the 3D formation degenerates to a constantly moving 2D structure, when only the followers are considered. To achieve this behavior, all followers share the very same absolute inclination i, resulting the the constraint $\varphi - \theta = const$. The arguments of perigee θ and the mean arguments of latitude φ are equally spaced throughout the orbit. The along-track and cross-track baselines are coupled and remain in an envelope, thus $\theta = {}^1\!/_4\pi$. This implies the constraint $\partial e = \partial i$ for CWF-MP formations (Peterson et al., 2008). The general CWF-MP description in ROEs is defined as

$$a_0 \delta \boldsymbol{a}_k = a \begin{pmatrix} \partial a_k \\ \partial \varphi_k \\ \partial \left(e_x\right)_k \\ \partial \left(e_y\right)_k \\ \partial \left(i_x\right)_k \\ \partial \left(i_y\right)_k \end{pmatrix} = \begin{pmatrix} 0 \\ -\dfrac{\partial \left(i_y\right)_k}{\tan i_0} \\ a_0 \tau \cos\left(\gamma_{k-1} + 0.5\pi\right) \\ a_0 \tau \sin\left(\gamma_{k-1} + 0.5\pi\right) \\ a_0 \tau \sqrt{2}^{-1} \\ a_0 \tau \sqrt{2}^{-1} \end{pmatrix} \tag{2.126}$$

where again the product $a\tau$ defines the baseline, γ_0 is an arbitrary design parameter and $\partial\varphi$ was chosen to ensure centered motion. Eq. 2.126 matches the PCO definition 2.125 in the relative eccentricity vector. The CWF-MP configuration can be understood as a PCO formation with tilted ellipses. The in-plane motion (Fig. 2.27i) is exactly the same as in a PCO configuration. This is a direct consequence from the common eccentricity vector. Beyond this, the out-of-plane motion (Fig. 2.27ii) is also elliptical for CWF-MP. With four follower satellites, the out-of-plane ellipses are equal by twos. This arises from a 180° shift per two satellites. The above-mentioned planes, which are defined by the instantaneous positions, are visualized as the dotted position lines in Fig. 2.27i and 2.27ii. The perfect alignment of this plane with the out-of-track plane can also be seen in Fig. 2.26. The explicit state vector for this configuration is given in Tab. 2.6. As with the latter formation, CWF-MP can also be built with three circling satellites and a leader in the center. This is depicted in Fig. 2.28.

Table 2.6: State vectors for a 5 km Cartwheel Formation – Moving Plane configuration in ROEs. All values are given in $[m]$.

Satellite	1	2	3	4
∂a	0	0	0	0
$\partial \varphi$	247	247	247	247
∂e_x	1291	0	−1291	0
∂e_y	0	1291	0	−1291
∂i_x	1826	1826	1826	1826
∂i_y	1826	1826	1826	1826



(i) In-plane motion (ii) Out-of-plane motion

Figure 2.27: Relative satellite motion of a 5 km Cartwheel Formation – Moving Plane configuration

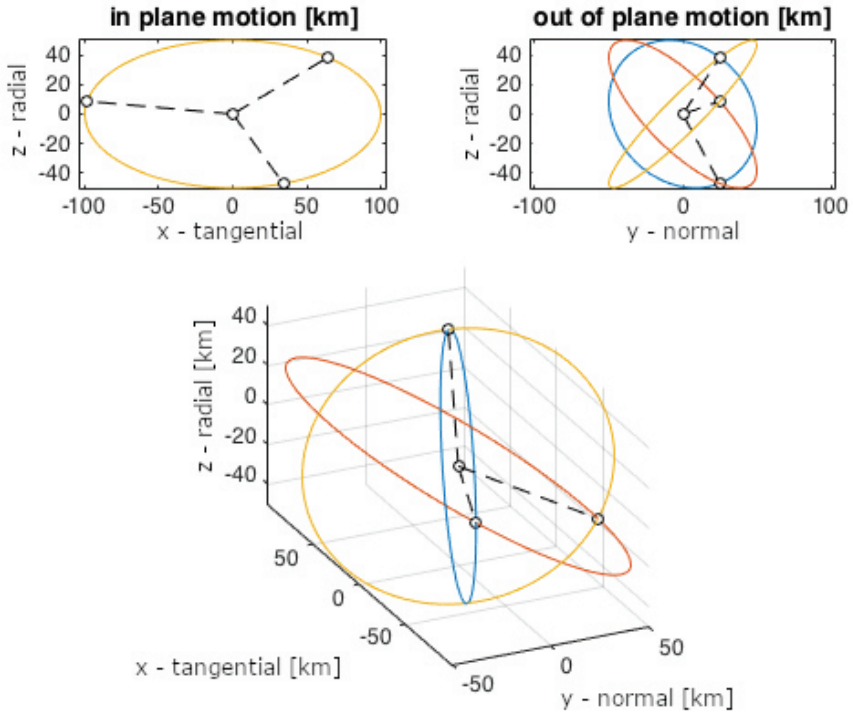


Figure 2.28: Cartwheel Formation – Moving Plane configuration with three circling satellites and one satellite in the center

2.6.6 Cartwheel Formation – 3:1

The Cartwheel Formation – 3:1 is a slight modification to the CWF-MP. It is defined by three satellites having a periodic motion in one plane, namely the $\hat{\boldsymbol{e}}_{\boldsymbol{x}}$-$\hat{\boldsymbol{e}}_{\boldsymbol{z}}$ plane, and one satellite having a periodic motion along the $\hat{\boldsymbol{e}}_{\boldsymbol{y}}$ axis (see Fig. 2.29). The initial conditions for three satellites in one plane and a single satellite performing

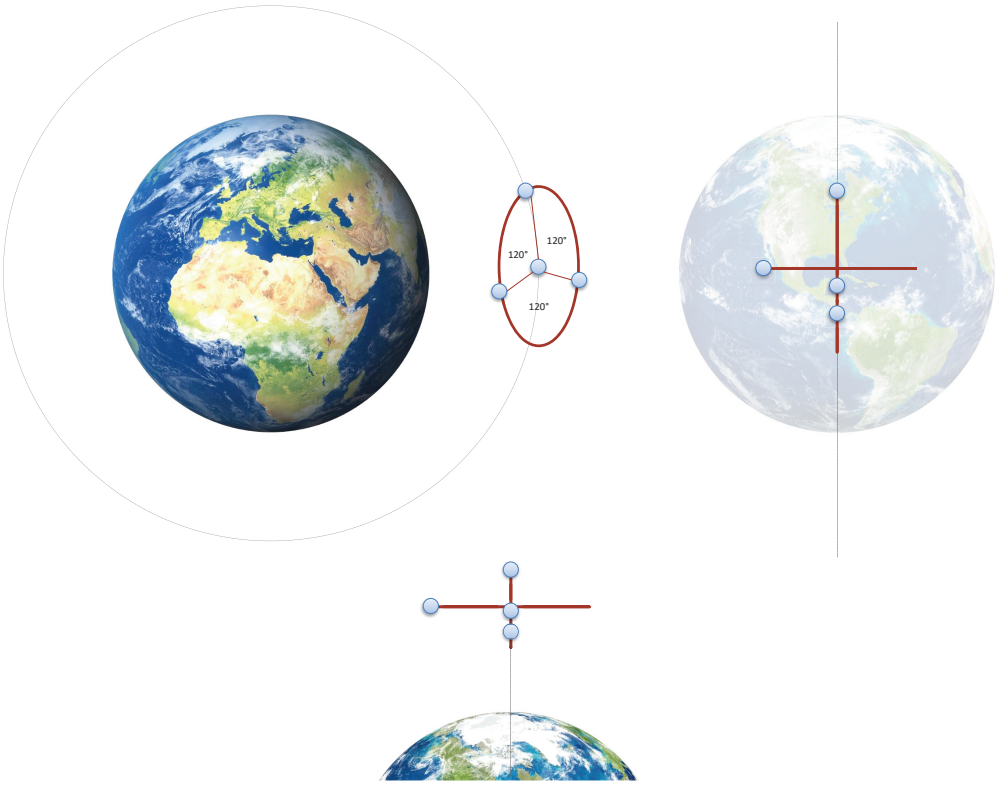


Figure 2.29: Cartwheel Formation – 3:1

the out-of-plane motion in the EH frame is given in Tab. 2.7. The initial states of the second and third satellite of the in plane group can easily be calculated with a force-free time propagation equal to a multiple of 1/3 and 2/3 of the orbital period.

Table 2.7: State vectors for a Cartwheel Formation – 3:1 configuration in the EH frame. Values are given in $[m]$ and $[m/s]$ respectively. A, B and C are arbitrary design parameters.

State	First in-plane satellite	Out-of-plane satellite
x_0	A	0
y_0	0	B
z_0	0	0
$\dot{x}_0$	0	0
$\dot{y}_0$	0	C
$\dot{z}_0$	0	0

Exemplary state vectors in ROE frame for the described formation is given in Tab. 2.8.

Table 2.8: State vectors for a 5 km Cartwheel Formation – 3:1 configuration in ROEs. All values are given in $[m]$.

Satellite	1	2	3	4
∂a	0	0	0	0
$\partial \varphi$	247	247	247	0
∂e_x	1291	−646	−1118	0
∂e_y	0	1118	646	0
∂i_x	0	0	0	2500
∂i_y	0	0	0	0

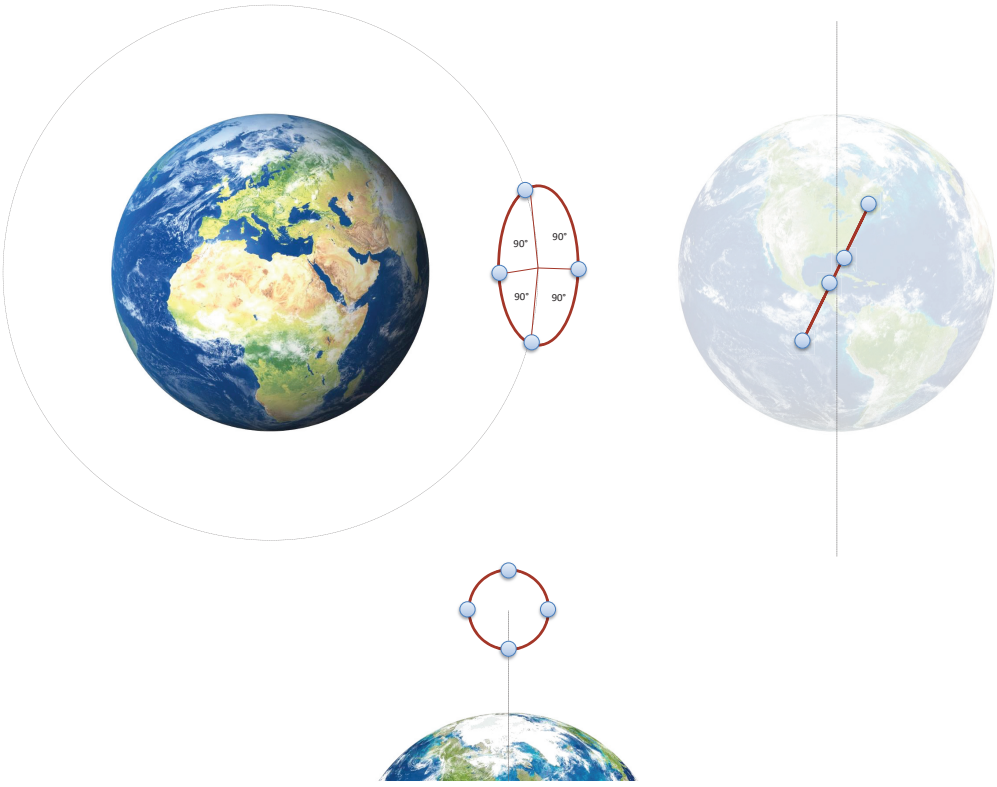


Figure 2.30: Cartwheel Formation – Helix

2.6.7 Cartwheel Formation – Helix

The Cartwheel Formation – Helix is a 2D formation in which all satellites are rotating around a common reference orbit (see Fig. 2.30). As seen from a point on the reference orbit (in front of or behind the formation), the satellites form a helix, which justifies the name (cf. Fig. 2.31).

The satellites are situated in the same plane, which is inclined with respect to the $\hat{\boldsymbol{e}}_{\boldsymbol{x}}$-$\hat{\boldsymbol{e}}_{\boldsymbol{y}}$ plane. The corresponding state vectors in ROE frame for the described formation is given in Tab. 2.9.

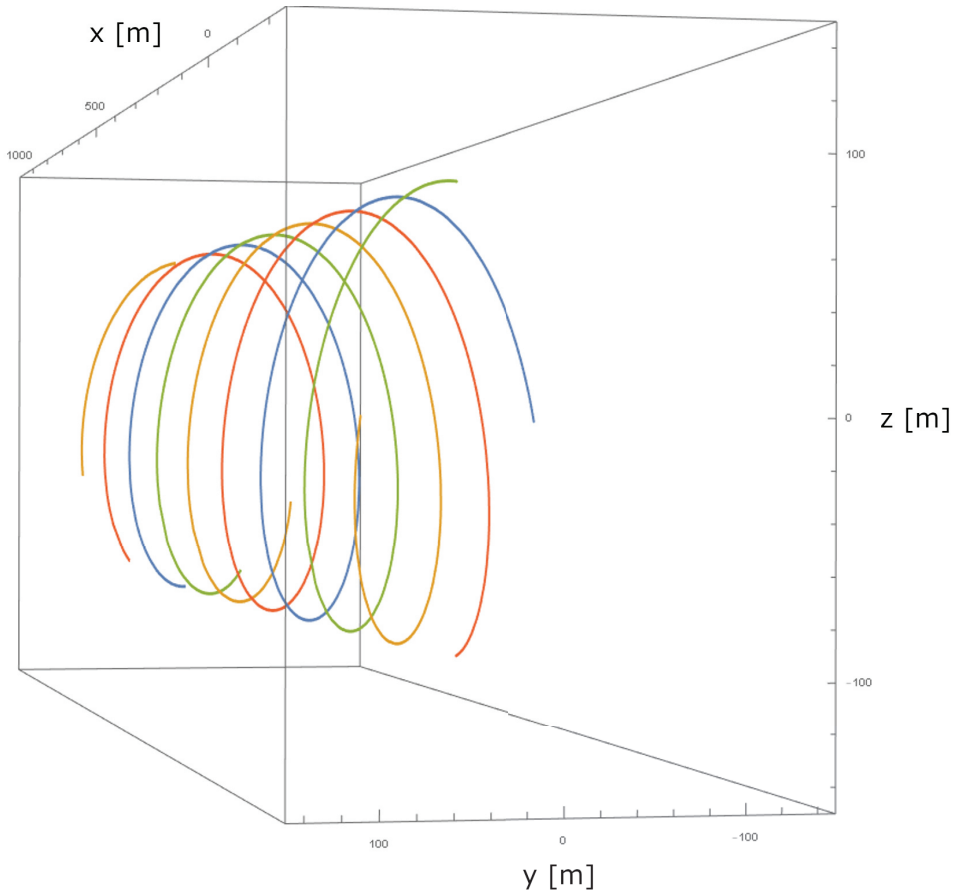


Figure 2.31: Cartwheel Formation – Helix as seen from a point in front of the satellites

Table 2.9: State vectors for a 5 km Cartwheel Formation – Helix configuration in ROEs. All values are given in $[m]$.

Satellite	1	2	3	4
∂a	0	0	0	0
$\partial \varphi$	247	247	247	247
∂e_x	1291	0	−1291	0
∂e_y	0	1291	0	−1291
∂i_x	1826	1826	1826	1826
∂i_y	1826	1826	1826	1826

2.6.8 Tetrahedron Formation

The THF is defined by two stationary and two dynamic satellites in a way that, through their periodic motion, a dynamic tetrahedron is spanned. With a specific selection of the satellites' relative orbits, the tetrahedron maintains a constant volume over time (cf. Fig. 2.32). Two satellites are sharing the same orbit as in an ATF. The other two satellites are situated on an ellipse with a phase difference of 120°.

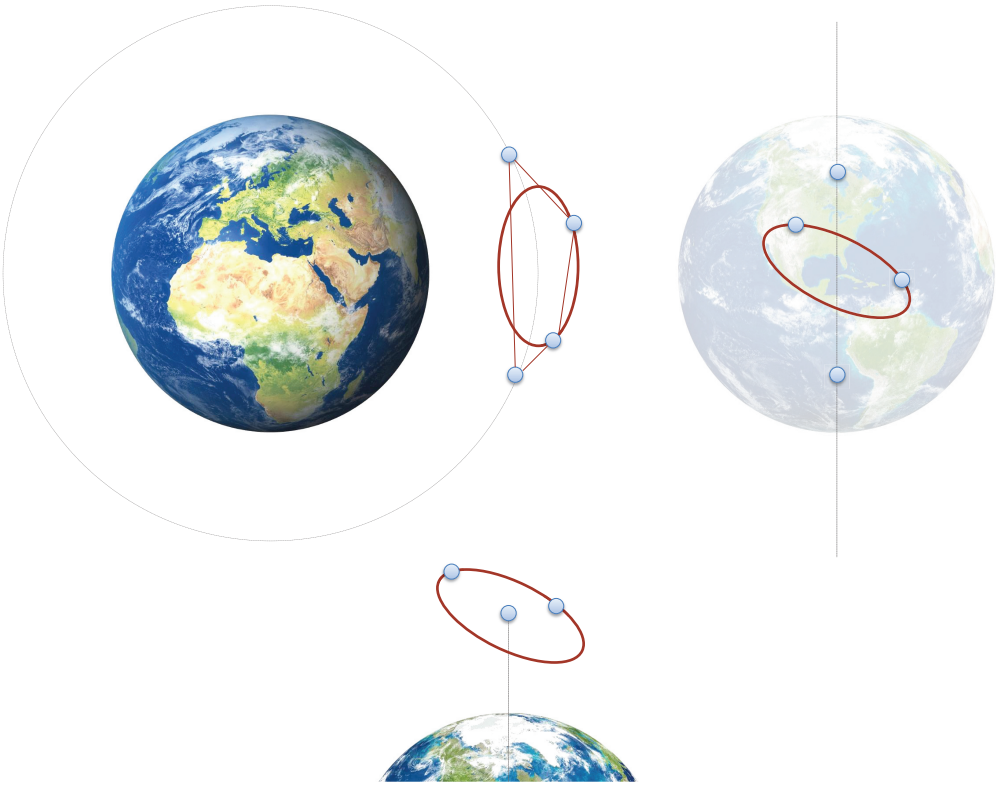


Figure 2.32: Tetrahedron Formation

This formation has been defined by Koptev et al. (2017) and is depicted in Fig. 2.33. Initial conditions are given below in Fig. 2.10.

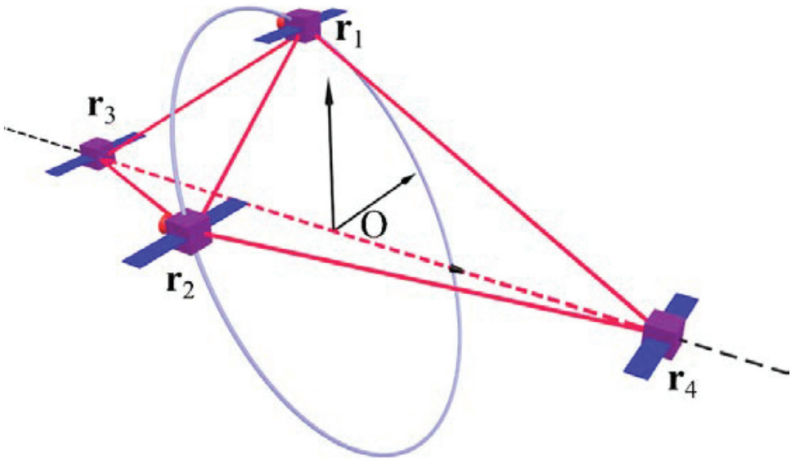


Figure 2.33: Tetrahedron Formation (Image courtesy: Koptev et al. (2017))

The formation design parameter A is defined as $A = \frac{3\sqrt{3}}{40}a$, with a being the semi-major axis of the elliptical motion of the first two satellites.

Table 2.10: State vectors for a Tetrahedron Formation configuration in the EH frame. Values are given in $[m]$ and $[m/s]$. A and B are arbitrary design parameters.

State	First satellite	Second satellite	Stationary satellites
x_0	$-\frac{2}{3}A$	$2A$	B
y_0	$\sqrt{3}A$	$\sqrt{3}A$	0
z_0	$\frac{2\sqrt{2}}{3}A$	0	0
$\dot{x}_0$	$\frac{4\sqrt{2}}{3}\omega A$	0	0
$\dot{y}_0$	0	$\frac{2\sqrt{2}}{\sqrt{3}}\omega A$	0
$\dot{z}_0$	$\frac{1}{3}\omega A$	$-\omega A$	0

2.6.9 J2 Perturbation Mitigation

The ROE state vectors for the described formations were designed taking the constraints from Eq. 2.38c into account. When perturbations are not neglected, these constraints have to be adapted. In this analysis, only the J_2-induced effect will be considered, as the remaining perturbation forces hardly influence formation behavior (cf. Chap. 2.5.1.1). Satellites in a formation will drift apart, if any of their three angle rates do not match. The angle rates are the angular velocities in in-track, cross-track and radial direction. A handy description of the angle rates is inherent in the Delaunay elements, i.e. in the Hamiltonian orbit description in Eq. 2.58 and 2.61. In this representation, the motion is divided into unperturbed and perturbed parts, as Eq. 2.64 denotes (where the perturbation coefficient is now $\epsilon = -J_2$). The unperturbed part $\mathfrak{H}_0$ is explicitly given by Eq. 2.65 and the perturbed part $\mathfrak{H}_1$ is given by the perturbation function from Brouwer's mean-to-osculating transformation (Brouwer, 1959)

$$\mathfrak{H}_1 = \frac{\mu^4 R_E^2}{4L^6}\left(\frac{a}{r}\right)^3\left[\left(3\frac{H^2}{G^2}-1\right)+3\left(1-\frac{H^2}{G^2}\right)\cos\theta\right] \tag{2.127}$$

Following the deviation in Alfriend et al. (2010), the Hamiltonian $\mathfrak{H}$ is now transformed to a normalized representation. For that purpose, distance is normalized by Earth's radius and time is normalized by the orbital period. Thus, R_E and μ are no longer part of $\mathfrak{H}$ and the derived conversions can be simplified. The normalized Hamiltonian $\bar{\mathfrak{H}}$ is then inserted into Eq. 2.63 to find the angular rates $\dot{l}$, $\dot{g}$ and $\dot{h}$. A back-transformation of the explicit formulation of the angular rates to orbital elements gives

$$\dot{l} = \frac{1}{a^{3/2}} + \epsilon\left(\frac{3}{4a^{7/2}\eta^3}\right)\left(1-3\cos^2 i\right) \tag{2.128a}$$

$$\dot{g} = \epsilon\left(\frac{3}{4a^{7/2}\eta^4}\right)\left(1-5\cos^2 i\right) \tag{2.128b}$$

$$\dot{h} = \epsilon \left(\frac{3}{2a^{7/2}\eta^4} \right) \cos i \tag{2.128c}$$

Two satellites will drift apart if and only if their rates are different

$$\delta\boldsymbol{\tau} = \boldsymbol{\tau_1} - \boldsymbol{\tau_0} \neq 0, \qquad \boldsymbol{\tau} = \left\{ \dot{l}, \dot{g}, \dot{h} \right\} \tag{2.129}$$

Ta achieve a stable formation, the satellites must not drift apart, i.e. Eq. 2.129 has to evaluate to zero for all three angular rates. In the following paragraphs, the consequences of this constraint on formation design are examined. The differential angular rates are obtained by expanding Eq. 2.128 in a Taylor series around the leader satellite. From this expansion it can be seen that the differential rates only depend on conjugate momenta, but not on the initial position or angle differences. The initial differential conjugate momenta are a function of δa, $\delta\eta$ and δi. This gives three equations with three unknowns, explicitly

$$\delta a = \frac{1}{2} J_2 a_0 \left(\frac{R_E}{a_0} \right)^2 \frac{3\eta_0 + 4}{\eta_0^5} \left(\left(1 - 3\cos^2 i_0 \right) \delta\eta - \eta_0 \sin\left(2i_0\right) \delta i \right) \tag{2.130a}$$

$$\delta\eta = \frac{5\eta_0 \sin\left(2i_0\right)}{4\left(1 - 5\cos^2 i_0\right)} \delta i \tag{2.130b}$$

$$\delta\eta = -\frac{\eta_0 \tan i_0}{4} \delta i \tag{2.130c}$$

Only if all three sub-equations are fulfilled, the formation will not suffer from any drift due to the J_2-factor of the Earth. This well-determined system imposes three constraints on initial formation design, restricting the shape of the formation. Or, in other words, restricting the available ROE state vectors derived earlier in this chapter. The first one constrains δa as a function of $\delta\eta$ and δi, the other two are coupled and both constrain $\delta\eta$ and δi. A mismatch in δa causes a drift in the first Delaunay angle $\dot{l}$ as can be inferred from Eq. 2.61a, since the mean rotation rate is a function of the semi-major axis (Eq. 2.25). Considering that all formations are restricted to a zero-offset in semi-major axis if perturbations are neglected, this constraint can easily be integrated in formation state vectors. This does not restrict available types or sizes of formations. Only the first element of the desired ROE vector has to be replaced with the result from Eq. 2.130a.

The other two constraints are coupled in $\delta\eta$ and δi and only their trivial solution

$$\delta\eta = \delta i = 0 \tag{2.131}$$

satisfies Eq. 2.129. Fulfilling Eq. 2.131 would only allow String-Of-Pearls formations, since the relative eccentricity and inclination vectors would have to be zero. Even regarding a single constraint of the two coupled ones to a high extent limits available formation shapes, since it determines the eccentricity vector after an inclination vector is chosen (and vice versa). Consequently, these constraints will not be regarded in

formation state vector design. Counteracting these differential rates will be part of the control effort and increase total ΔV consumption during formation maintenance phases. This effort, anyhow, is small compared to the total J_2-mitigation effort consisting of all three constraints (Eq. 2.130).

2.7 Control Theory Fundamentals

This section presents the relevant control theory fundamentals that are the basis of the later on presented control algorithms. Though, a basic knowledge in control theory is required.

2.7.1 Fundamentals

General State-Space Representation of a Single Linear Time-Invariant System

A linear system, being both time-invariant (LTI) or time-varying (Linear Time-Varying (LTV)) can be described as a state-space model. Usually it is derived from differential equations describing the dynamics of the physical system, but it can also be composed directly or transformed from a transfer function in frequency domain. Considering only the time-invariant case (LTI), such systems can be represented in state-space in the form

$$\dot{\boldsymbol{x}}(t) = \boldsymbol{A}\ \boldsymbol{x}(t) + \boldsymbol{B}\ \boldsymbol{u}(t) \tag{2.132}$$

$$\boldsymbol{y}(t) = \boldsymbol{C}\ \boldsymbol{x}(t) + \boldsymbol{D}\ \boldsymbol{u}(t) \tag{2.133}$$

with the state vector $\boldsymbol{x}(t)$, the output vector $\boldsymbol{y}(t)$ and the input or control vector $\boldsymbol{u}(t)$. $\boldsymbol{A}$ is the state or system matrix, $\boldsymbol{B}$ the input matrix, $\boldsymbol{C}$ the output matrix and $\boldsymbol{D}$ the feed-through or feed-forward matrix. In LTI the matrices $\boldsymbol{A}$, $\boldsymbol{B}$, $\boldsymbol{C}$, $\boldsymbol{D}$, which define the state-space system, are not time-dependent. For further fundamentals about LTI control systems the reader is referred to Lunze (2014b, p. 72).

Block Diagram Representation of State-Space Systems

A general state space system can be represented as a block diagram. Such a generic block diagram is shown in Fig. 2.34.

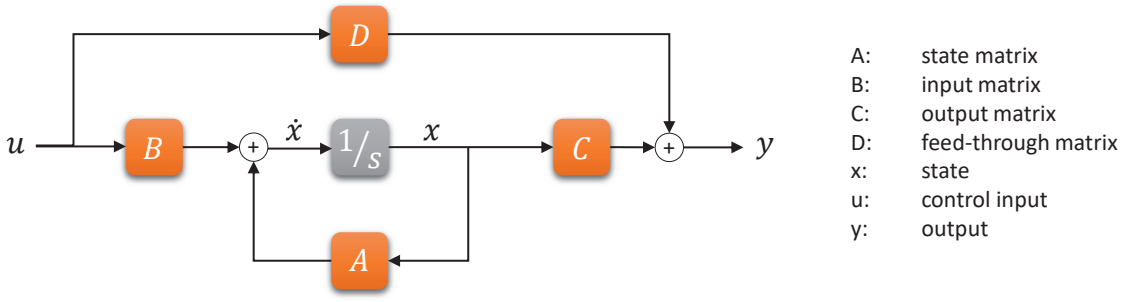


Figure 2.34: Block diagram representation of a general state space system. (Lunze, 2014b, p. 74)

Transformation Between State-Space Model and Transfer Function

Any state-space model can be transformed into a transfer function $\boldsymbol{G}(s)$ (Skogestad and Postlethwaite, 2001, p. 116), (Lunze, 2014a, p. 20)

$$\boldsymbol{G}(s) = \boldsymbol{C}^T \left(s\boldsymbol{I} - \boldsymbol{A}\right)^{-1} \boldsymbol{B} + \boldsymbol{D} \tag{2.134}$$

The inverse transformation from transfer function $\boldsymbol{G}(s)$ to a state-space model $(\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{C}, \boldsymbol{D})$ cannot be formulated in a general formula, but is dependent on the order of the system and the number of inputs and outputs. Different procedures can be applied to compute this transformation. Depending on the procedure, either the Observability Canonical Form, Controllability Canonical Form or the Diagonalized Form of a state-space model can be obtained. (Skogestad and Postlethwaite, 2001, pp. 119)

2.7.2 Linear Quadratic Regulator

This section introduces the Linear Quadratic Regulator (LQR). The derivation is based on Lunze (2014a). LQR is an optimal feedback-based control scheme, which optimizes a gain matrix $\boldsymbol{K}$ in a linear state-feedback control law

$$\boldsymbol{u} = -\boldsymbol{K}\boldsymbol{x} \tag{2.135}$$

with state deviation $\boldsymbol{x} = \boldsymbol{x_e} - \boldsymbol{x_r}$ being the difference between the current state estimate and the reference state (desired state). Eq. 2.135 is subject to a state-space definition like the one stated in Chap. 2.7.1. The control input (i.e. thrust) is optimized with the use of a quadratic cost function (hence the name of the controller)

$$J_1 = \int_0^{t_1} \left(\boldsymbol{x}^T(t)\, \boldsymbol{Q}\boldsymbol{x}(t) + \boldsymbol{u}^T(t)\, \boldsymbol{R}\boldsymbol{u}(t)\right) dt \tag{2.136}$$

where the symmetric positive definite gain matrices $\boldsymbol{Q}$ and $\boldsymbol{R}$ weight the time-dependent state deviation and control inputs, respectively. When this cost function is minimized, the input is optimized over the planning horizon t_1. By varying the ratio between $\boldsymbol{Q}$ and $\boldsymbol{R}$, one can direct the attention to either one of the vectors $\boldsymbol{x}$ or $\boldsymbol{u}$. This means, the chosen matrix ratio represents the trade-off between total ΔV consumption to achieve a target orbit and total time until this orbit is reached. With the definition above, J_1 is dependent on the target time t_1. To get a LTI, the planning horizon for the cost function Eq. 2.136 is shifted to infinity

$$J = \int_0^{\infty} \left(\boldsymbol{x}^T(t)\, \boldsymbol{Q}\boldsymbol{x}(t) + \boldsymbol{u}^T(t)\, \boldsymbol{R}\boldsymbol{u}(t)\right) dt \tag{2.137}$$

To get a finite value for the integral, $\boldsymbol{x} \to \boldsymbol{0}$ for $t \to \infty$ must hold, i.e. the solution must be feasible under the given constraints. The optimization problem for control

input and execution time can now be stated as

$$\min_{\boldsymbol{u}(t)} = J\left(\boldsymbol{x_0}, \boldsymbol{u}\left(t\right)\right) \tag{2.138}$$

This dynamic control law can be transformed to a static control law with the LTI feedback Eq. 2.135

$$\min_{\boldsymbol{K}} = J\left(\boldsymbol{x_0}, -\boldsymbol{K}\boldsymbol{x}\left(t\right)\right) \tag{2.139}$$

The cost function J is now minimized over the control gain matrix and not the thrust input. It is static, because the control gain matrix is, in contrast to the thrust input, constant over time. To find the optimal control gain $\boldsymbol{K}$, Eq. 2.139 has to be solved. For this, the time-dependent state deviation $\boldsymbol{x}\left(t\right)$ is expressed in terms of the initial state and its state transition matrix, as defined in Eq. 2.73 and 2.79. The time-dependent thrust input is expressed in its state contribution with Eq. 2.132 and 2.135, leading to the new cost function

$$J = \int_0^\infty \left(\boldsymbol{x}_0^T e^{\boldsymbol{A}^T t}\boldsymbol{Q}e^{\boldsymbol{A}t}\boldsymbol{x}_0 + \boldsymbol{x}^T\left(t\right)\boldsymbol{K}^T\boldsymbol{R}\boldsymbol{K}\boldsymbol{x}\left(t\right)\right)dt \tag{2.140}$$

This can be split and $\boldsymbol{x}_0$ can be extracted from the integral to give the form

$$J = \boldsymbol{x}_0^T\boldsymbol{P}\boldsymbol{x}_0 + \int_0^\infty \boldsymbol{x}^T\left(t\right)\boldsymbol{K}^T\boldsymbol{R}\boldsymbol{K}\boldsymbol{x}\left(t\right)dt \tag{2.141}$$

where

$$\boldsymbol{P} = \int_0^\infty e^{\boldsymbol{A}^T t}\boldsymbol{Q}e^{\boldsymbol{A}t}dt \tag{2.142}$$

With the calculus of variations, it is shown that $\boldsymbol{P}$ can be implicitly expressed as

$$\boldsymbol{A}^T\boldsymbol{P} + \boldsymbol{P}\boldsymbol{A} = -\boldsymbol{Q} \tag{2.143}$$

The expression is a Lyapunov equation, which is an important tool for stability analysis (see Chap. 2.7.3). There exists a positive definite solution $\boldsymbol{P}$, if $\boldsymbol{Q}$ is positive definite (as is by definition of Eq. 2.136) and $\boldsymbol{A}$ is asymptotically stable. Minding the feedback contribution Eq. 2.135 of $\boldsymbol{u}$ in the state space equation gives

$$\bar{\boldsymbol{A}} = \boldsymbol{A} - \boldsymbol{B}\boldsymbol{K} \tag{2.144}$$

With this transformation, the cost function can be expressed as

$$J = \int_0^\infty \boldsymbol{x}^T\left(t\right)\bar{\boldsymbol{Q}}\boldsymbol{x}\left(t\right)dt \tag{2.145}$$

with

$$\bar{\boldsymbol{Q}} = \boldsymbol{Q} + \boldsymbol{K}^T\boldsymbol{R}\boldsymbol{K} \tag{2.146}$$

If this is included to the definition of $\boldsymbol{P}$, one can write the minimization problem as

$$\min_{\boldsymbol{K}} J = \min_{\boldsymbol{K}} \boldsymbol{x}_0^T\boldsymbol{P}\boldsymbol{x}_0 \tag{2.147}$$

where $\boldsymbol{P}$ is defined through the extended state equations

$$\bar{\boldsymbol{A}}^T\boldsymbol{P} + \boldsymbol{P}\bar{\boldsymbol{A}} = -\bar{\boldsymbol{Q}} \tag{2.148}$$

The solution to Eq. 2.147 is found by differentiating J with respect to $\boldsymbol{K}$ and evaluates to

$$\boldsymbol{K} = \boldsymbol{R}^{-1}\boldsymbol{B}^T\boldsymbol{P} \tag{2.149}$$

or, back-transformed to input space, the optimal thrust profile is given by

$$\boldsymbol{u}(t) = -\boldsymbol{R}^{-1}\boldsymbol{B}^T\boldsymbol{P}\boldsymbol{x}(t) \tag{2.150}$$

The inverse of $\boldsymbol{R}$ exists, since it is positive definite. The optimization problem is now broken down to finding an explicit expression for $\boldsymbol{P}$. This is found by inserting Eq. 2.150 into Eq. 2.148, which is known as the Algebraic Riccati Equation (ARE)

$$\boldsymbol{A}^T\boldsymbol{P} + \boldsymbol{P}\boldsymbol{A} - \boldsymbol{P}\boldsymbol{B}\boldsymbol{R}^{-1}\boldsymbol{B}^T\boldsymbol{P} + \boldsymbol{Q} = 0 \tag{2.151}$$

The ARE may be solved numerically or by matrix factorization techniques provided by e.g. MATLAB.

Another way of computing the resulting feedback controller / feedback gain matrix is to use quadratic programming (Barik et al., 2012).

2.7.3 Lyapunov Stability

Control Lyapunov Functions (CLFs) are Lyapunov functions for controllable systems. With a valid CLF, a feedback matrix $\boldsymbol{K}$ can iteratively be found. This derivation is loosely based on Schaub et al. (2000b) and Lunze (2014a). As with LQR control, model fidelity but also complexity is adjusted with the selection of the state-space matrix $\boldsymbol{A}$ and the control influence matrix $\boldsymbol{B}$. CLF-based controller design in the scope of formation flying is an emerging field. Some related publications are presented in Chap. 2.9.2.

In general, a state-dependent Lyapunov function proves the stability of an ODE. The general state space model Eq. 2.132 is a set of ODEs and may be controlled with a linear feedback gain. The target of CLF-based controller design is to find the feedback matrix $\boldsymbol{K}$. The difficult step in creating a CLF-based control law is to find a CLF which proves stability and minimizes the thrust input. This can sometimes only be done with brute-force or trial-and-error techniques. The start of this process is a state-space dependent function which has to be proven to be stable in the sense of Lyapunov. Therefore, in the first place it is called a Lyapunov candidate function V and constrained to

$$V(\boldsymbol{x}) : \mathbb{R}^n \to \mathbb{R} \qquad V(\boldsymbol{0}) = 0,\, V(\boldsymbol{x}) > 0 \tag{2.152}$$

The candidate function maps from the state-space to a scalar and has to be positive definite except at the origin, where it has to evaluate to zero. An equilibrium point $\boldsymbol{x_e}$ of a system is stable in the sense of Lyapunov, if for any neighborhood ϵ there exists a neighborhood δ around $\boldsymbol{x_e}$, so that $\|\boldsymbol{x_0}\| < \delta$ leads to $\|\boldsymbol{x}(t)\| < \epsilon$. Or, in other words, it is stable, if any state in the local neighborhood of an equilibrium stays in the vicinity of it for all times. The more restrictive asymptomatic stability demands $\lim_{t\to\infty} \|\boldsymbol{x}(t)\| = 0$. With this definition, a system is asymptotically stable, if any initial state in the local neighborhood of an equilibrium point reaches this point at some time. Transferred to control Lyapunov functions, this is the same as to demand that V must be monotonically decreasing to reach the equilibrium point. Thus, a candidate function proves stability if and only if

$$\dot{V}(\boldsymbol{x}(t), \boldsymbol{u}(t)) < 0 \qquad \forall t \tag{2.153}$$

The time-derivative of V is in fact a function of the state and the thrust input, since it influences the state derivative. Adapted to relative position control, this is illustrated with the linear feedback control law

$$\boldsymbol{u} = -\boldsymbol{B}^{-1}(\boldsymbol{A}\boldsymbol{x_e} + \boldsymbol{P}\boldsymbol{x}) \tag{2.154}$$

with a positive definite error influence matrix $\boldsymbol{P}$, the error input sharing the same dimension as the state and $\boldsymbol{B}$ being non-singular. Let the candidate Lyapunov function be

$$V(\boldsymbol{x}) = 0.5\boldsymbol{x}^T\boldsymbol{x} \tag{2.155}$$

The reference state is assumed to be constant in this case. Taking the derivative and inserting first the state space equation, then Eq. 2.154 gives

$$\begin{aligned}
\dot{V}(\boldsymbol{x}(t), \boldsymbol{u}(t)) =& \boldsymbol{x}^T\dot{\boldsymbol{x}} \\
=& \boldsymbol{x}^T(\dot{\boldsymbol{x_e}} - \dot{\boldsymbol{x_r}}) \\
=& \boldsymbol{x}^T(\boldsymbol{A}\boldsymbol{x_e} + \boldsymbol{B}\boldsymbol{u} - \boldsymbol{0}) \\
=& \boldsymbol{x}^T\left\{\boldsymbol{A}\boldsymbol{x_e} - \boldsymbol{B}\boldsymbol{B}^{-1}[\boldsymbol{A}\boldsymbol{x_e} + \boldsymbol{P}(\boldsymbol{x_e} - \boldsymbol{x_r})]\right\} \\
=& -\boldsymbol{x}^T\boldsymbol{P}(\boldsymbol{x_e} - \boldsymbol{x_r}) \\
=& -\boldsymbol{x}^T\boldsymbol{P}\boldsymbol{x}
\end{aligned} \tag{2.156}$$

This solution is strictly negative, since the feedback gain matrix $\boldsymbol{P}$ is positive definite. Thus, the proposed function Eq. 2.155 proves stability of the control law Eq. 2.154. This process of proving Lyapunov stability increases in complexity if a variable reference is chosen.

2.7.4 Model Predictive Control

Model Predictive Control (MPC) or Receding Horizon Control is an advanced control method that has been developed since the 1980s. It relies on dynamic models and allows the current timeslot to be optimized, while keeping future timeslots in account. This is achieved by optimizing a finite time-horizon, but only implementing the current timeslot. It has the ability to anticipate future events and can take control actions accordingly. It allows for large time delays and high-order dynamics. It predicts the change in the dependent variables of the modeled system that will be caused by changes in the independent variables. It is based on iterative, finite-horizon optimization of a plant model. MPC uses an internal dynamic model of the process, a history of past control moves and an optimization cost function J over the receding prediction horizon, to calculate the optimum control moves (Wikipedia Contributors, 2020b).

The optimal control input is found according to various system constraints and a cost function J defined out of different weight functions. The main characteristic lies in the receding nature of the controller which involves iterative re-planning and re-optimization of the control input to achieve the lowest cost throughout the moving horizon window. Following the MPC method the control objectives are fulfilled by the minimization of a cost function. A quadratic cost function can e.g. be chosen as (MathWorks, 2020)

$$J(I_k) = \sum_{j=1}^{n_u} \sum_{i=0}^{p-1} [\frac{w_{i,j}^u}{s_j^u} (x_j(k+i|k) - x_{j,target}(k+i|k))]^2 \tag{2.157}$$

where n_u is the number of manipulated variables, p the planning horizon, k the current control interval, x_j and $x_{j,target}$ the associated state value according to the state space model and the desired target state value for the j_{th} state and $w_{i,j}$ and s_j the associated variable weights and scaling for that state. If terminal weights are desired, Eq. 2.157 becomes

$$J(I_k) = \sum_{j=1}^{n_u} [\frac{w_{i,p-1}^u}{s_j^u} (x_j(k+p-1|k) - x_{j,target}(k+p-1|k))]^2 \tag{2.158}$$

Any model mismatches, manipulated input and measured output disturbances that cause accumulated target deviation over time are thus taken into account in such iterative manner. Concepts and theory underlying MPC can be traced back as early as the 1960s mostly in the work of Kalman and the LQR mathematical formulation. As soon as the needs of industry for optimal control with system constraints, planning and regular updating were met within feasible computational workloads the use of MPC started to spread beginning from the control of chemical and petroleum slow dynamic systems. In the last decades there has been a wide integration of MPC in industrial applications thanks to the leaps in computational performance of embedded systems being able to solve demanding quadratic programming problems on-line for increasingly fast and complex systems (Wang, 2009).

MPC has also already been applied to space applications e.g. spacecraft rendezvous (Hartley et al., 2012). Concerning the ability to handle constraints, to precisely adjust performance to time-varying mission criteria and to tune any parameters easily after deployment, MPC is distinctively fitting to SFF. In particular for the 3D formation acquisition task in LEO, which includes periodic motion with frequencies in the order of hours and limited perturbations, the computational resources and power consumption of using a variant of MPC are low enough to be selected as primary control strategy for on-board implementation in small satellites.

2.7.4.1 Quadratic Optimization Problem

Solving the optimal control problem to derive a trajectory that satisfies the final state and thrust constraints with a minimum cost involves simplifying the cost function to a quadratic form and turning the problem into a LQR problem. The LQR problem then has to be solved iteratively within tight sample rates highlighting the need for an efficient and low throughput Quadratic Programming (QP) algorithm that forms the main computation load of LQR.
The solution to a MPC control iteration comes down to solving a QP problem. There are several methods used for solving that problem and although minimizing a convex quadratic function is something that has been researched since 1955, it has been a major subject for ongoing research in resource efficient algorithms that can be implemented in the current low cost microelectronic solutions, achieving energy and resource efficient control for remote applications. The current most prominent methods for solving the QP of an MPC controller are: Active Set methods (best for small and medium sized problems), Interior Point methods (can deal with large scale problems), Gradient Projection methods, Conjugate Gradient methods, Fast Dual Gradient Projection methods, Forward-Backwards Newton methods and Augmented Lagrangian methods.

2.7.4.2 Explicit MPC

What has been first introduced in relevant literature as Dynamic Programming with constraints and later named Parametric Programming or Multi-parametric Programming is commonly called Explicit MPC. It is the use of a QP algorithm to solve all possible discrete states $\boldsymbol{x}(k)$ for a given frame k off-line.
It can deal with every possible discrete control system problem in an iterative way forming a $N+1$ dimensional vector map of associated states and input controls that point to their derived next state in the minimum cost trajectory that satisfies the cost function J and the given constraints. (The remaining cost of that trajectory all the way to the end can also be included as redundant information). An extra dimension denotes the discrete steps of control and for example in a three dimensional motion of a spacecraft through time it can denote time increments and their associated optimal control input for each possible state.
The main issue that occurs implementing Explicit MPC is the required memory necessary for storing all this information for a $N+1$ dimensional control problem.

For a spacecraft for example the required memory to control its motion in EH frame within a range of 10 km divided in discrete steps of 1 m distance in each dimension and a maximum control timeframe of 1000 min with a 1 min control interval is resulting in $10^4 * 10^4 * 10^4 * 10^3 = 10^{15}$ memory slots (e.g 4 B each leads to approximately 4 TB of required memory)[1]. Since memory resources are limited in satellite missions, the explicit implementation of MPC is not considered.

2.7.4.3 Implementation in a Digital Discretized System

The outcome of a single MPC computation is a trajectory plan for the whole planning horizon. If this interval is extended, the far-time optimality increases at the cost of computational load. A quite short interval can be used in an on-board real-time controller. Also, MPC can be used as a closed-loop control system when only the first command of the plan is implemented and a new plan is generated in every time step. This section describes the practical implementation in a digital discretized system and is based on Tillerson et al. (2002) and Breger and How (2005). It also uses the EH frame state-space definition 2.76. Without loss of generality, this definition can be replaced with a LTV model like the ones derived in Chap. 2.5.3.2 and 2.5.3.4.

A key parameter of the MPC is the sample period T_s, i.e. the interval for which thrust commands are generated. These commands can be implemented in the thrust system with a Zero-Order Hold (ZOH) discretization method (see Chap. 2.9.4), limiting the maximum sample period. Discretization of the state-space Eq. 2.76 is conducted with approximate integration, assuming a very small state and thrust change over one sampling period. The goal of the following equation reshaping process is to find a form that is suitable for numerical solving in a computational system, starting with the state space equation of the system. The discretized form of Eq. 2.76 becomes

$$\boldsymbol{x}\left(k+1\right) \approx e^{\boldsymbol{A}_k T_s} \boldsymbol{x}\left(k\right) + \int_0^{T_s} e^{\boldsymbol{A}_k \tau} d\tau \boldsymbol{B}\left(k\right) \boldsymbol{u}\left(k\right) \tag{2.159}$$

In the case of the control being constant over the sampling time, this equation is not only an approximation though. With the definition of the discrete state transition matrix $\boldsymbol{\Phi}$ (see Chap. 2.5.3.1) and the discrete control influence matrix $\boldsymbol{G}_k$ it can be expressed as

$$\boldsymbol{x}\left(k+1\right) = \boldsymbol{\Phi}\boldsymbol{x}\left(k\right) + \boldsymbol{G}_k \boldsymbol{u}\left(k\right) \tag{2.160}$$

with $\boldsymbol{x} \in \mathbb{R}^n$ and $\boldsymbol{u} \in \mathbb{R}^m$ and

$$\boldsymbol{\Phi} = e^{\boldsymbol{A}_k T_s} \tag{2.161a}$$

$$\boldsymbol{G}_k = \int_0^{T_s} e^{\boldsymbol{A}_k \tau} d\tau \boldsymbol{B}\left(k\right) \tag{2.161b}$$

[1] There are few algorithms and methods that can reduce this memory issue with certain trade-offs like state Increment Dynamic Programming and Lagrange Multipliers Polynomial Approximation.

To let the controller predict all states in the time frame of the planning horizon, $\boldsymbol{x}(k)$ must be calculable from $\boldsymbol{x}(0)$ for any k. This is done with discrete convolution of the state and input matrices

$$\boldsymbol{x}\left(k\right) = \boldsymbol{\Phi}^{(k,k)}\boldsymbol{x}\left(0\right) + \sum_{i=0}^{k-1} \boldsymbol{\Phi}^{(k-i-1,k)}\boldsymbol{G}_i\boldsymbol{u}\left(i\right) \tag{2.162}$$

where the convolution is defined with

$$\boldsymbol{\Phi}^{(j,k)} = \begin{cases} \boldsymbol{\Phi}_{k-1}\cdots\boldsymbol{\Phi}_{k-j+1}\boldsymbol{\Phi}_{k-j} & 2 \leq j \leq k \\ \boldsymbol{\Phi}_{k-1} & j = 1 \\ \boldsymbol{I} & j = 0 \end{cases} \tag{2.163}$$

Using this definition, the state $\boldsymbol{x}(0)$ is multiplied k times with the STM $\boldsymbol{\Phi}(0, T_s)$, propagating the initial state until $t = kT_s$. The control influence is propagated depending on the position in the planning horizon. While the first control influence at $t = 0$ is propagated $k - 1$ times, the last control input at the end of the planning horizon is not at all propagated with the STM. This procedure results in the exact contribution of each thrusting step to the final state. Eq. 2.162 can be transformed into the following matrix form

$$\boldsymbol{x}\left(k\right) = \boldsymbol{A}\left(k\right)\boldsymbol{U}_k + \boldsymbol{b}\left(k\right) \tag{2.164}$$

where $\boldsymbol{A} \in \mathbb{R}^{n\times(k+1)m}$ and $\boldsymbol{b} \in \mathbb{R}^{n\times 1}$. They are given by

$$\boldsymbol{A}\left(k\right) = \begin{bmatrix} \boldsymbol{\Phi}^{(k-1,k)}\boldsymbol{G}_0 & \boldsymbol{\Phi}^{(k-2,k)}\boldsymbol{G}_1 & \cdots & \boldsymbol{\Phi}^{(0,k)}\boldsymbol{G}_{k-1} & \boldsymbol{0} \end{bmatrix} \tag{2.165a}$$

$$\boldsymbol{b}\left(k\right) = \boldsymbol{\Phi}^{(k,k)}\boldsymbol{x}\left(0\right) \tag{2.165b}$$

The control vector $\boldsymbol{U}_k \in \mathbb{R}^{(k+1)m\times 1}$ is a concatenation of all thrust input values for each time step

$$\boldsymbol{U}_k = \begin{pmatrix} \boldsymbol{u}\left(0\right)^T & \boldsymbol{u}\left(1\right)^T & \cdots & \boldsymbol{u}\left(k-1\right)^T & \boldsymbol{u}\left(k\right)^T \end{pmatrix}^T \tag{2.166}$$

This dynamic model serves as the basis for the numerical optimization program, which minimizes the thrust input to achieve an optimal control. The optimization problem can be stated with the equality constraint to reach the desired target state as

$$\min_{U_k} \sum_{j=0}^{k} c_j \|\boldsymbol{u}\left(j\right)\| \quad \text{subject to } \boldsymbol{x}(k) = \boldsymbol{x_r}(k) \tag{2.167}$$

where the variables c_j weight the thrust input depending on the time instant in the control horizon. c_j can be an all-ones vector for example, the vector can impose less weight on early thrust actions to motivate a faster target approach.

When an error box around the target state is used, the equality constraint becomes an inequality constraint of the form

$$\| x_i(k) - x_{i,r}(k) \| \leq \epsilon_i \qquad \epsilon_i \geq 0 \; \forall i \in \{1, n\} \tag{2.168}$$

with error box dimensions ϵ_i for all state vector components. By enforcing such a constraint at every (other) time step, a formation maintenance controller can be designed which does not counteract every minor deviation. The controller just cares about holding the satellite in the defined error box around the reference state.

2.7.4.4 Definition of Constraints

Additional constraints can be added to the optimization problem Eq. 2.167 to cope with satellite-inherent properties. When the maximum thrust limit is defined with

$$u^{\min}(i) \leq u(i) \leq u^{\max}(i) \tag{2.169}$$

the inequality constraining thrust levels can be written as

$$\begin{bmatrix} \boldsymbol{I} \\ -\boldsymbol{I} \end{bmatrix} \boldsymbol{U}_k \leq \begin{pmatrix} \boldsymbol{U}_k^{\max} \\ \boldsymbol{U}_k^{\min} \end{pmatrix} \tag{2.170}$$

where $\boldsymbol{U}_k^{\max}$ and $\boldsymbol{U}_k^{\min}$ are appropriately dimensioned vectors of $u^{\max}(i)$ and $u^{\min}(i)$, respectively. This enforces the priorly stated scalar equation at all time steps. In the same manner, actuator rate limits can be enforced with

$$r^{\min}(i) \leq u(i+1) - u(i) \leq r^{\max}(i) \tag{2.171}$$

where r is the maximum rate change of a thruster during one time step T_s. Tillerson et al. (2002) furthermore provide a method to re-formulate the problem Eq. 2.167 as a minimization problem. In the current form, it is impossible to minimize the input vector $\boldsymbol{U}_k$, since the individual thrust values may have negative and positive values. To overcome this, they introduce slack variables for positive and negative contribution and double the size of the input vector to

$$\hat{\boldsymbol{U}}_k = \begin{pmatrix} \boldsymbol{U}_k^{+} \\ \boldsymbol{U}_k^{-} \end{pmatrix} \tag{2.172}$$

where

$$\boldsymbol{U}_k = \boldsymbol{U}_k^{+} - \boldsymbol{U}_k^{-} \tag{2.173}$$

with the conditions

$$\boldsymbol{U}_k^{+} \geq 0, \;\; \boldsymbol{U}_k^{-} \geq 0 \tag{2.174}$$

The positive and negative parts of the control input are simply concatenated, which means that the new extended input vector is positive definite. The numerically

solvable linear minimization problem is finally formulated as

$$J = \min_{\hat{\boldsymbol{U}}_k} \boldsymbol{C}^T \hat{\boldsymbol{U}}_k \tag{2.175}$$

using an extended weighting matrix $\boldsymbol{C}^T$ and being subject to the relative orbit constraint

$$\begin{bmatrix} +\boldsymbol{A}(k) & -\boldsymbol{A}(k) \\ -\boldsymbol{A}(k) & +\boldsymbol{A}(k) \end{bmatrix} \hat{\boldsymbol{U}}_k \leq \begin{pmatrix} +\boldsymbol{x_r}(k) - \boldsymbol{b}(k) + \boldsymbol{\epsilon} \\ -\boldsymbol{x_r}(k) + \boldsymbol{b}(k) + \boldsymbol{\epsilon} \end{pmatrix} \tag{2.176}$$

and the extended maximum thrust constraint

$$\begin{bmatrix} +\boldsymbol{I} & -\boldsymbol{I} \\ -\boldsymbol{I} & +\boldsymbol{I} \end{bmatrix} \hat{\boldsymbol{U}}_k \leq \begin{pmatrix} \boldsymbol{U}_k^{\max} \\ \boldsymbol{U}_k^{\min} \end{pmatrix} \tag{2.177}$$

In the same manner, the maximum rate constraint can be extended. Any other state space constraint (for example enforcing Eq. 2.81 in EH frame) can be accommodated as long as it follows the given form to minimize a matrix inequality subject to the thrust input vector.

2.7.5 Robust Control

Robust control is a branch of control theory that explicitly deals with uncertainty. A controller designed for a particular set of parameters is said to be robust, if it would also work well under a different set of assumptions. Rather than adapting to variations of measurements, the controller is designed to work assuming that certain variables will be unknown, but bounded. H_∞ loop-shaping is an typical example. Fig. 2.35 presents a block diagram of a general control configuration in a robust setup.

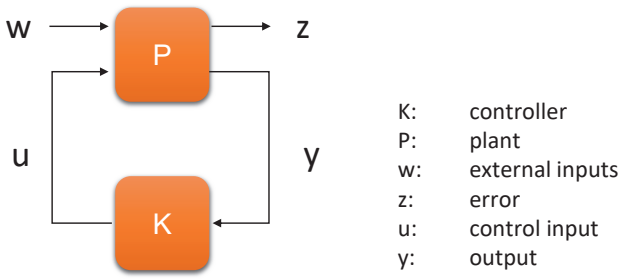


Figure 2.35: General control configuration (Zhou et al., 1996, p. 442)

A generalized plant including external inputs $\boldsymbol{w}$, errors $\boldsymbol{z}$, control inputs $\boldsymbol{u}$ and outputs $\boldsymbol{y}$ can be represented as a general state-space model with the state $\boldsymbol{x}$ and its derivation $\dot{\boldsymbol{x}}$ as internal variables

$$\begin{aligned} \dot{\boldsymbol{x}} &= \boldsymbol{A}\ \boldsymbol{x} + \boldsymbol{B_1}\ \boldsymbol{w} + \boldsymbol{B}_2\ \boldsymbol{u} \\ \boldsymbol{z} &= \boldsymbol{C}_1\ \boldsymbol{x} + \boldsymbol{D}_{11}\ \boldsymbol{w} + \boldsymbol{D}_{12}\ \boldsymbol{u} \\ \boldsymbol{y} &= \boldsymbol{C}_2\ \boldsymbol{x} + \boldsymbol{D}_{21}\ \boldsymbol{w} + \boldsymbol{D}_{22}\ \boldsymbol{u} \end{aligned} \tag{2.178}$$

External inputs $\boldsymbol{w}$ can contain various inputs from a reference signal $\boldsymbol{r}$ to disturbances $\boldsymbol{d}$ to noise $\boldsymbol{n}$. The state-space model can be – as every control system expressed in a state-space representation – transformed into a Transfer Function (TF) (cf. Chap. 2.7.1) (Lunze, 2014a, p. 20), (Zhou et al., 1996, p. 442) using

$$\boldsymbol{P}(s) = \boldsymbol{C}(s\boldsymbol{I} - \boldsymbol{A})^{-1}\boldsymbol{B} + \boldsymbol{D} \tag{2.179}$$

as

$$\boldsymbol{P}(s) = \begin{array}{c|cc} & \boldsymbol{w} & \boldsymbol{u} \\ \hline \boldsymbol{z} & \boldsymbol{P}_{11}(s) & \boldsymbol{P}_{12}(s) \\ \boldsymbol{y} & \boldsymbol{P}_{21}(s) & \boldsymbol{P}_{22}(s) \end{array} \tag{2.180}$$

With the help of Eq. 2.179, we introduce the notation

$$\boldsymbol{P}(s) = \begin{array}{c|c|cc} & \boldsymbol{x} & \boldsymbol{w} & \boldsymbol{u} \\ \hline \dot{\boldsymbol{x}} & \boldsymbol{A} & \boldsymbol{B}_1 & \boldsymbol{B}_2 \\ \hline \boldsymbol{z} & \boldsymbol{C}_1 & \boldsymbol{D}_{11} & \boldsymbol{D}_{12} \\ \boldsymbol{y} & \boldsymbol{C}_2 & \boldsymbol{D}_{21} & \boldsymbol{D}_{22} \end{array} \tag{2.181}$$

which implies Eq. 2.179 and translates the state-space representation to a TF (Zhou et al., 1996, p. 442).

2.7.5.1 H_∞ Control Approach

The H_∞ control approach considers the H_∞ norm of the Closed-Loop Transfer Function (CLTF)

$$\boldsymbol{T}_{zw} = \boldsymbol{F}_L(\boldsymbol{P}, \boldsymbol{K}) \tag{2.182}$$

with F_L representing the lower Linear Fractional Transformation (LFT) (Doyle et al., 1991). It is defined for a given complex matrix $\boldsymbol{M}$ and a given rational matrix $\boldsymbol{\Delta}_L$ with

$$\boldsymbol{M} = \begin{pmatrix} \boldsymbol{M}_{11} & \boldsymbol{M}_{12} \\ \boldsymbol{M}_{21} & \boldsymbol{M}_{22} \end{pmatrix} \tag{2.183}$$

as

$$\boldsymbol{F}_L(\boldsymbol{M}, \boldsymbol{\Delta}_L) := \boldsymbol{M}_{11} + \boldsymbol{M}_{12}\boldsymbol{\Delta}_L\ (\boldsymbol{I} - \boldsymbol{M}_{22}\boldsymbol{\Delta}_L)^{-1}\ \boldsymbol{M}_{21} \tag{2.184}$$

The H_∞ control problem can be defined in two ways (Zhou et al., 1996, p. 443):

a) **Optimal control problem**: Finding a stabilizing controller that minimizes

$$\|\boldsymbol{T}_{zw}\|_\infty \tag{2.185}$$

b) **Suboptimal control problem**: Finding a stabilizing controller that satisfies

$$\|\boldsymbol{T}_{zw}\|_\infty < \gamma, \gamma > 0 \tag{2.186}$$

The H_∞ control problem is solvable, if and only if, the following assumptions apply (Skogestad and Postlethwaite, 2001, p. 370):

A1. $(\boldsymbol{A}, \boldsymbol{B}_2)$ is stabilizable and $(\boldsymbol{C}_2, \boldsymbol{A})$ is detectable,

A2. $\boldsymbol{D}_{12}$ and $\boldsymbol{D}_{21}$ have full rank,

A3. $\begin{bmatrix} \boldsymbol{A} - j\omega\boldsymbol{I} & \boldsymbol{B}_2 \\ \boldsymbol{C}_1 & \boldsymbol{D}_{12} \end{bmatrix}$ has full column rank for all ω,

A4. $\begin{bmatrix} \boldsymbol{A} - j\omega\boldsymbol{I} & \boldsymbol{B}_1 \\ \boldsymbol{C}_2 & \boldsymbol{D}_{21} \end{bmatrix}$ has full rank rank for all ω.

2.8 Distributed Control

This section presents the fundamentals of the distributed control methods used in this thesis.

2.8.1 Graph Theory Fundamentals

Interconnection topologies between satellites or agents in a distributed control system can most adequately be described using graph theory. Thus the relevant terms and definitions used later on will be introduced.

Graph

A graph (or undirected graph) is an ordered pair $G = (V,\ E)$, where V is a set of vertices (or nodes) and E is a set of edges (see Fig. 2.36). Each edge itself is an unordered pair of vertices. $e_{ij} = \{v_i,\ v_j\}$ e.g. describes the edge between the vertices v_i and v_j. For an undirected graph $e_{ij} = e_{ji}$ (Tian, 2012).

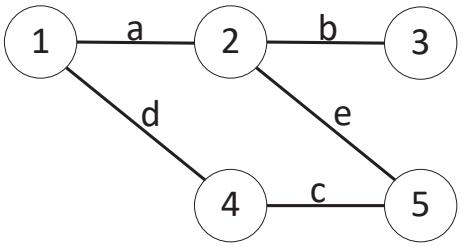


Figure 2.36: An undirected graph with its vertices are labeled with numbers and its edges with letters.

Directed Graph

A directed graph (or digraph) is a graph in which the edges have orientations (see Fig. 2.37). Thus an edge is an ordered pair of two vertices. $e_{ij} = (v_i,\ v_j)$ therefore describes the edge from vertex v_i to vertex v_j. In general, for directed graphs $e_{ij} \neq e_{ji}$ (Tian, 2012).

Path

A path in a directed graph is an ordered sequence of vertices such that any consecutive pair within the sequence is an edge of the directed graph. In other words, a path is a walk through a graph from a vertex via vertices to a vertex along edges. The length of the path is defined by the number of consecutive edges in the path. Fig. 2.38 visualizes a path within a graph (Tian, 2012).

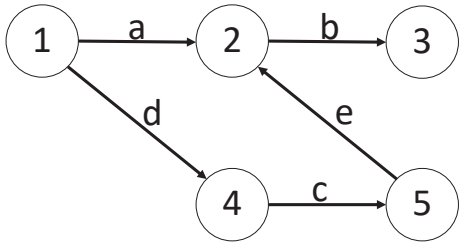


Figure 2.37: Directed graph with its vertices are labeled with numbers and its edges with letters. Edges show arrows to visualize their direction. E.g. a is the edge from vertex 1 to vertex 2.

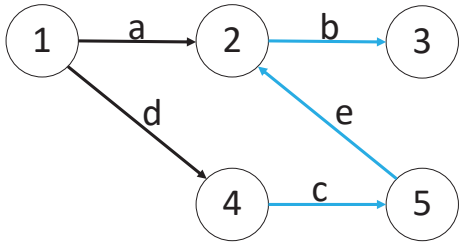


Figure 2.38: Directed graph with a path from vertex 4 to vertex 3 highlighted.

Reachability

A vertex v_i within a directed graph is said to be reachable from another vertex v_j, if there exists a path from v_j to v_j. A vertex is globally reachable, if there exists a path from every other vertex. A graph is said to be connected (strongly connected in case of directed graphs), if every vertex is globally reachable. Cf. Fig. 2.38 as an example where vertex 3 is a globally reachable node, since it is reachable from every other vertex (Tian, 2012).

Neighbor

A vertex v_i is said to be a neighbor of vertex v_j, if there exists an edge $e_{ji} = (v_j,\ v_i)$ from vertex v_j to vertex v_i. For undirected graphs, if v_i is a neighbor of v_j then also v_j is a neighbor of v_i. E.g. in Fig. 2.38 Node 2 is a neighbor of Node 1, but not the other way round (Tian, 2012).

Adjacency Matrix

Neighborhood relations within a graph can be expressed in an algebraic way as a matrix, the so called adjacency matrix $\boldsymbol{A}_j = [a_{ij}] \in \mathbb{R}^{n \times n}$ with

$$a_{ij} = \begin{cases} 1 & \text{if } \exists\ e_{ij} = (v_i,\ v_j) \in E \\ 0 & \text{otherwise} \end{cases} \tag{2.187}$$

A graph is undirected, if $a_{ij} = a_{ji}$ (Tian, 2012).

Degree Matrix

The degree matrix $\boldsymbol{D}_g = [d_{ij}] \in \mathbb{R}^{n \times n}$ is a diagonal matrix stating the number of edges connected to each node. It is defined as

$$d_{ij} = \begin{cases} deg(v_i) & \text{if } i = j \\ 0 & \text{otherwise} \end{cases} \tag{2.188}$$

where the degree $deg(v_i) = \sum_j a_{ij}$ counts the number of outgoing edges of a vertex v_i (Tian, 2012).

Laplacian Matrix

The Laplacian matrix $L \in \mathbb{R}^{n \times n}$ is a matrix representation of the graph. It combines adjacency and degree matrix and is defined as

$$\boldsymbol{L} = \boldsymbol{D}_g - \boldsymbol{A}_j \tag{2.189}$$

To get a better understanding of its physical meaning we can consider the case that only relative measurement between the agents along the connected edges is available. In this case the relative measurement of agent i can be written using $\boldsymbol{L} = [l_{ij}]$ as

$$y_i(t) = \sum_j l_{ij}\ x_j(t) \tag{2.190}$$

or in matrix form

$$\boldsymbol{y}(t) = \boldsymbol{L}\ \boldsymbol{x}(t) \tag{2.191}$$

Thus it can be understood as relative-measurement matrix (Tian, 2012).

2.8.2 Distributed Control Systems

A distributed control system is a control system wherein individual controllers are distributed throughout a larger system. Thus, it differs from non-distributed systems, that consist of a single controller at a central location. In a distributed control system, a set of controllers is connected via a communication network (Çela et al., 2014, Lunze, 2019). According to Tian (2012) common features of distributed control systems are distributed interconnections, local control rules, scalability and of course cooperation among the agents. Since in a distributed control system there is no central controller, the cooperation among the agents is crucial for the functionality of the system. Another characteristic is the spatially distributed interconnection of the agents. Each agent/local controller thus does not only take information about its own state/output into account, but also considers the information of some other agents. Since they can be distributed among large distances delays or package loss may also play a role. Distributed control systems feature a local control rule. Because there is no centralized supervisor/controller, each agent makes its own decisions based on its own sensor inputs and the information provided by its neighboring

agents. An advantage of a local control rule is the high fault-tolerance capability, since failing of a single or few agents will not stop the complete system from working. Another reason why a local control rule is beneficial, is scalability. Since a local control rule uses only local information from the agent itself or from neighboring agents, it usually does not depend on the size of the whole system and thus allows for an arbitrary number of agents in the system. Scalability enables the adaption of the control system towards new applications without changing the underlying system. Further distributed control systems can be differentiated into homogeneous and heterogeneous systems depending on the types and characteristics of the agents, if they are equal or differ (Tian, 2012).

Connection to Graph Theory

In a distributed control system, the network topology or interconnection scheme can be modeled using directed graphs. Each vertex represents a subsystem or agent. If there is information flow (i.e. a communication link) from one agent to another, then there exists an edge with the same direction in the graph. If all communication links are bidirectional, the topology can be described by an undirected graph.

General State-Space Representation of a Distributed Linear Time-Invariant System

A distributed LTI control system $\boldsymbol{\Sigma}$ consisting of N subsystems Σ_i can be described as

$$\boldsymbol{\Sigma} = (\Sigma_1, \Sigma_2, ..., \Sigma_N)^T \tag{2.192}$$

and a subsystem Σ_i is defined as

$$\dot{\boldsymbol{x}}_{\boldsymbol{i}} = \boldsymbol{A}_{ii}\ \boldsymbol{x}_i + \sum_{i \neq j}^{N} \boldsymbol{A}_{ij}\ \boldsymbol{x}_j + \boldsymbol{B}_{ii}\ \boldsymbol{u}_i + \sum_{i \neq j}^{N} \boldsymbol{B}_{ij}\ \boldsymbol{u}_j \tag{2.193}$$

$$\boldsymbol{y}_i = \boldsymbol{C}_{ii}\ \boldsymbol{x}_i + \sum_{i \neq j}^{N} \boldsymbol{C}_{ij}\ \boldsymbol{x}_j + \boldsymbol{D}_{ii}\ \boldsymbol{u}_i + \sum_{i \neq j}^{N} \boldsymbol{D}_{ij}\ \boldsymbol{u}_j \tag{2.194}$$

with the time dependencies of $\dot{\boldsymbol{x}}_i(t)$, $\boldsymbol{y}_i(t)$, $\boldsymbol{x}_i(t)$, $\boldsymbol{x}_j(t)$, $\boldsymbol{u}_i(t)$, $\boldsymbol{u}_j(t)$ being omitted to improve readability. The terms $\boldsymbol{A}_{ii}$, $\boldsymbol{B}_{ii}$, $\boldsymbol{C}_{ii}$, $\boldsymbol{D}_{ii}$ describe the behavior of the subsystem Σ_i itself, whereas the mixed terms $\boldsymbol{A}_{ij}$, $\boldsymbol{B}_{ij}$, $\boldsymbol{C}_{ij}$, $\boldsymbol{D}_{ij}$ describe the influence of other subsystems Σ_j on the subsystem Σ_i. Thus, the coupling terms can be named as

$\sum_{i \neq j}^{N} \boldsymbol{A}_{ij}\ \boldsymbol{x}_j$: state coupling

$\sum_{i \neq j}^{N} \boldsymbol{B}_{ij}\ \boldsymbol{u}_j$: input coupling

$\sum_{i \neq j}^{N} \boldsymbol{C}_{ij}\ \boldsymbol{x}_j$: output coupling

$\sum_{i \neq j}^{N} \boldsymbol{D}_{ij}\ \boldsymbol{u}_j$: feed-through coupling

Communication Delays

If communication delays between the subsystems/agents are of interest, they can be modeled by replacing t by $t - \tau_{ij}$ in in the state-space representation of a distributed system (Eq. 2.193 and 2.194), where τ_{ij} is the communication delay from agent j to agent i (Tian, 2012). This leads to the following general state-space representation including delays

$$\dot{\boldsymbol{x}}_i(t) = \boldsymbol{A}_{ii}\ \boldsymbol{x}_i(t) + \sum_{i \neq j}^{N} \boldsymbol{A}_{ij}\ \boldsymbol{x}_j(t - \tau_{ij}) + \boldsymbol{B}_{ii}\ \boldsymbol{u}_i(t) + \sum_{i \neq j}^{N} \boldsymbol{B}_{ij}\ \boldsymbol{u}_j(t - \tau_{ij}) \tag{2.195}$$

$$\boldsymbol{y}_i(t) = \boldsymbol{C}_{ii}\ \boldsymbol{x}_i(t) + \sum_{i \neq j}^{N} \boldsymbol{C}_{ij}\ \boldsymbol{x}_j(t - \tau_{ij}) + \boldsymbol{D}_{ii}\ \boldsymbol{u}_i(t) + \sum_{i \neq j}^{N} \boldsymbol{D}_{ij}\ \boldsymbol{u}_j(t - \tau_{ij}) \tag{2.196}$$

Additionally, in some cases it makes sense to introduce a self-delay τ_{ii} to match the communication delays. Thus the local subsystem is also delayed, leading to

$$\dot{\boldsymbol{x}}_i(t) = \boldsymbol{A}_{ii}\ \boldsymbol{x}_i(t - \tau_{ii}) + \sum ... + \boldsymbol{B}_{ii}\ \boldsymbol{u}_i(t - \tau_{ii}) + \sum ... \tag{2.197}$$

$$\boldsymbol{y}_i(t) = \boldsymbol{C}_{ii}\ \boldsymbol{x}_i(t - \tau_{ii}) + \sum ... + \boldsymbol{D}_{ii}\ \boldsymbol{u}_i(t - \tau_{ii}) + \sum ... \tag{2.198}$$

Decentralized and Distributed Control

A controller can be used to compute the control input for the distributed system Σ described in Eq.2.193 and 2.194, e.g. $\boldsymbol{u}_i(t) = -\boldsymbol{K}\ \boldsymbol{x}_i(t)$ with a gain matrix $\boldsymbol{K}$. If the controller takes only the local state into account meaning

$$\boldsymbol{u}_i(t) = f(\boldsymbol{x}_i(t)) \tag{2.199}$$

it uses a decentralized control approach. If it also takes states of other subsystems into account meaning

$$\boldsymbol{u}_i(t) = f(\boldsymbol{x}_i(t), \boldsymbol{x}_j(t), ...) \tag{2.200}$$

it uses a distributed control approach.

2.8.2.1 Distributed Consensus Approach

There are different types of distributed control systems. A promising way of cooperation in a distributed control system is finding a common agreement or consensus among the agents (Lunze, 2019, Tian, 2012). The distributed consensus approach is a method to compute the control input $\boldsymbol{u}_i(t)$ of distributed systems Σ_i in which only the controllers, but not the plants are coupled. Fig. 2.39 shows an exemplary block diagram for two coupled controllers with these characteristics. With respect to the general representation in Eq. 2.193 and 2.194 the matrices $\boldsymbol{A}_{ij}$, $\boldsymbol{B}_{ij}$, $\boldsymbol{C}_{ij}$ and $\boldsymbol{D}_{ij}$ are equal $0\ \forall\ i \neq j$.

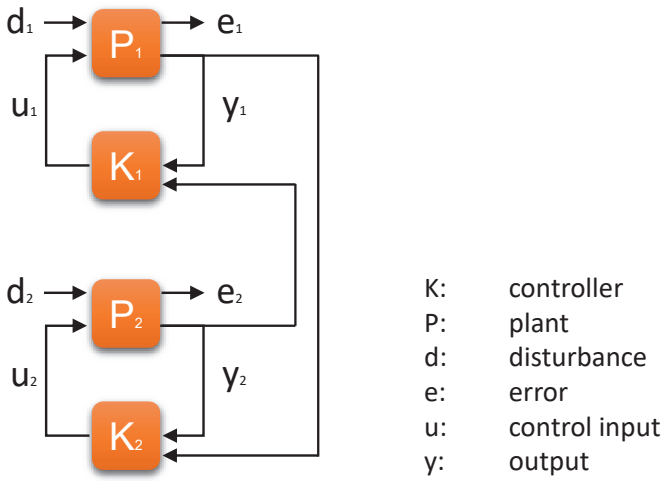


Figure 2.39: Two plants $\boldsymbol{P}_1$ and $\boldsymbol{P}_2$ with their controllers $\boldsymbol{K}_1$ and $\boldsymbol{K}_2$. The plants are not coupled, but the controllers are. $\boldsymbol{y}_1$ influences $\boldsymbol{K}_2$ and $\boldsymbol{y}_2$ influences $\boldsymbol{K}_1$.

In this case, the control input can then be defined by a consensus protocol (for a regulator problem) by (Ren and Beard, 2008, p. 26)

$$\boldsymbol{u}_i(t) = \boldsymbol{K} \sum_j a_{ij} \left(\boldsymbol{y}_j(t) - \boldsymbol{y}_i(t)\right) \tag{2.201}$$

with a_{ij} are the elements of the adjacency matrix $\boldsymbol{A}_j$ and $\boldsymbol{K}$ is the gain matrix for a single subsystem/agent. With the state-space definition of a single system $(\boldsymbol{A},\ \boldsymbol{B},\ \boldsymbol{C},\ 0)$ and by linking together the states of the subsystems $\boldsymbol{x} = (x_1, x_2, ..., x_N)^T$ to form the state of the overall system the closed-loop system with consensus protocol can be written in matrix form as (Tian, 2012, eq. 6.59)

$$\dot{\boldsymbol{x}}(t) = \left(\boldsymbol{I}_N \otimes \boldsymbol{A} - \boldsymbol{L} \otimes \boldsymbol{BKC}\right) \boldsymbol{x}(t) \tag{2.202}$$

where $\otimes$ denotes the Kronecker product, an operator on two matrices defined as

$$\boldsymbol{A} \otimes \boldsymbol{B} = \begin{pmatrix} a_{11}\,\boldsymbol{B} & \dots & a_{1m}\,\boldsymbol{B} \\ \vdots & & \vdots \\ a_{n1}\,\boldsymbol{B} & \dots & a_{nm}\,\boldsymbol{B} \end{pmatrix} \tag{2.203}$$

with the properties

$$(\boldsymbol{A} \otimes \boldsymbol{B}) \cdot (\boldsymbol{C} \otimes \boldsymbol{D}) = (\boldsymbol{A} \cdot \boldsymbol{C}) \otimes (\boldsymbol{B} \cdot \boldsymbol{D}) \tag{2.204}$$

$$\boldsymbol{A} \otimes (\boldsymbol{B} + \boldsymbol{C}) = \boldsymbol{A} \otimes \boldsymbol{B} + \boldsymbol{A} \otimes \boldsymbol{C} \tag{2.205}$$

$$\boldsymbol{A} \otimes \boldsymbol{B} \neq \boldsymbol{B} \otimes \boldsymbol{A} \quad \text{(in general)} \tag{2.206}$$

$$\boldsymbol{A} \otimes (\boldsymbol{B} \otimes \boldsymbol{C}) = (\boldsymbol{A} \otimes \boldsymbol{B}) \otimes \boldsymbol{C} \tag{2.207}$$

$$(\alpha_A \boldsymbol{A} \otimes \alpha_B \boldsymbol{B}) = \alpha_A \ \alpha_B \ (\boldsymbol{A} \otimes \boldsymbol{B}) \tag{2.208}$$

This control approach enforces the satellites to reduce the relative distances / vectors along all interconnections.

Reference Tracking

The consensus protocol for regulation problems presented in Eq. 2.201 can be adapted for reference tracking problems

$$\boldsymbol{u}_i(t) = \boldsymbol{K} \sum_j a_{ij} \ [\boldsymbol{r}_{ji} - (\boldsymbol{y}_j(t) - \boldsymbol{y}_i(t))] \tag{2.209}$$

with $\boldsymbol{r}_{ji}$ being the reference vector from satellite i to satellite j. Obviously, $\boldsymbol{r}_{ij} = -\boldsymbol{r}_{ji}$ and $\boldsymbol{r}_{ii} = 0$. Fig. 2.40 shows an exemplary block diagram of three systems with controllers coupled via the consensus approach.

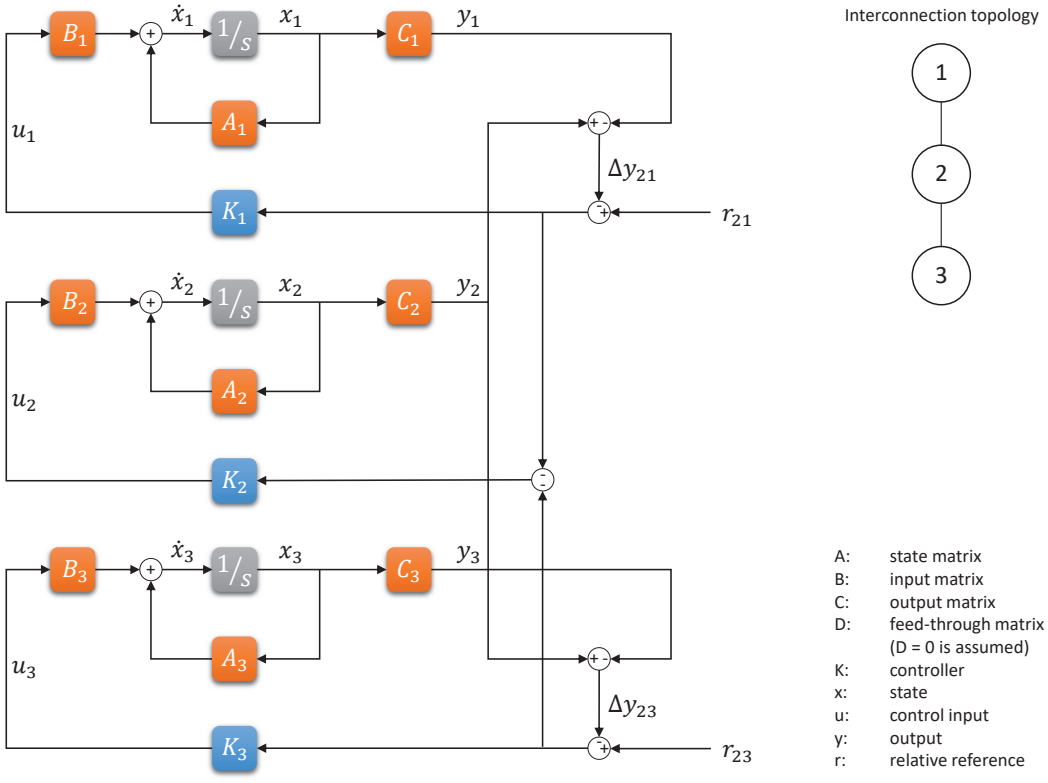


Figure 2.40: Exemplary block diagram of three distributed systems with their controllers coupled via the consensus approach. The interconnection topology is shown as a graph in the upper right corner.

Additionally, Fig. 2.41 presents a schematic explaining the relative references $\boldsymbol{r}_{ij}$, which act as inputs to distributed controllers as stated in Eq. 2.209.

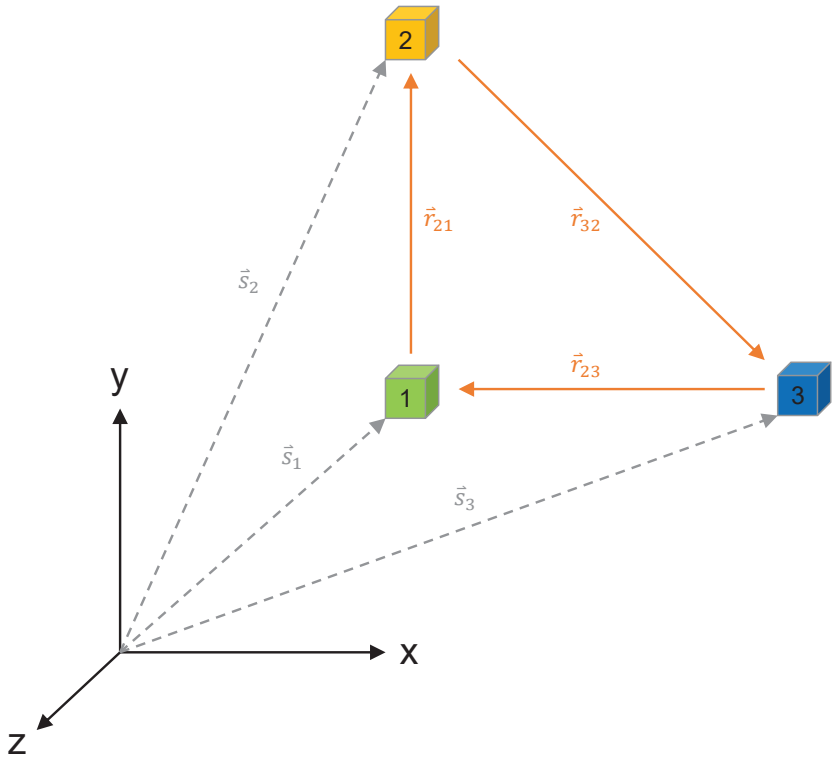


Figure 2.41: Schematic drawing of three agents in an arbitrary coordinate system. Their states are labeled $\boldsymbol{s}_1$, $\boldsymbol{s}_2$ and $\boldsymbol{s}_3$. Relative references are labeled according to Eq. 2.209 with $\boldsymbol{r}_{ji}$ being the reference vector from agent i to agent j. For readability, the inverse references $\boldsymbol{r}_{ij} = -\boldsymbol{r}_{ji}$ are omitted.

To reformulate Eq. 2.209 in matrix form the references $\boldsymbol{r}_{ij}$ can be organized in a matrix. Exemplary it is given below for a system $\boldsymbol{\Sigma}$ consisting of four subsystems / agents

$$\boldsymbol{R} = \begin{pmatrix} 0 & \boldsymbol{r}_{12} & \boldsymbol{r}_{13} & \boldsymbol{r}_{14} \\ -\boldsymbol{r}_{12} & 0 & \boldsymbol{r}_{23} & \boldsymbol{r}_{24} \\ -\boldsymbol{r}_{13} & -\boldsymbol{r}_{23} & 0 & \boldsymbol{r}_{34} \\ -\boldsymbol{r}_{14} & -\boldsymbol{r}_{24} & -\boldsymbol{r}_{34} & 0 \end{pmatrix} \tag{2.210}$$

$\boldsymbol{R}$ is skew-symmetric and has the same zero-values as the adjacency matrix $\boldsymbol{A}_j$. Using $\boldsymbol{R}$, the closed-loop system of this extended tracking problem can also be written in matrix form

$$\dot{\boldsymbol{x}}(t) =$$

$$(\boldsymbol{I}_N \otimes \boldsymbol{A})\, \boldsymbol{x}(t) - \left\{(\boldsymbol{L} \otimes \boldsymbol{BKC}) \odot \left[\left(\mathbf{1}_{6\times 1} \otimes \boldsymbol{x}^T(t)\right) + (\boldsymbol{R} \otimes \mathbf{1}_{6\times 1})\right]\right\} \cdot \mathbf{1}_{N\cdot 6\times 1} \tag{2.211}$$

or

$$\dot{\boldsymbol{x}}(t) =$$

$$(\boldsymbol{I}_N \otimes \boldsymbol{A} - \boldsymbol{L} \otimes \boldsymbol{BKC})\, \boldsymbol{x}(t) + [(L \otimes \boldsymbol{BKC}) \odot (\boldsymbol{R} \otimes \mathbf{1}_{6\times 1})] \cdot \mathbf{1}_{N\cdot 6\times 1} \tag{2.212}$$

where $\odot$ denotes the Hadamard product (or element-wise product) of two matrices, which is defined for two matrices $\boldsymbol{A}$ and $\boldsymbol{B}$ with equal size $(n \times m)$ by

$$\boldsymbol{A} \odot \boldsymbol{B} = (a_{ij} \cdot b_{ij}) = \begin{pmatrix} a_{11} \cdot b_{11} & \cdots & a_{1m} \cdot b_{1m} \\ \vdots & \ddots & \vdots \\ a_{n1} \cdot b_{n1} & \cdots & a_{nm} \cdot b_{nm} \end{pmatrix} \tag{2.213}$$

The aim of formation control is that all agents follow a predefined or given trajectory while maintaining a particular spacial formation. Thus, the consensus approach is especially suitable for formation control.

Fixing Coordinate Origin to an Agent

In some cases, e.g. for easier measuring of the states of the involved agents and easier definition of the coordinate frame origin, it makes sense to fix the coordinate frame origin to one agent's position. This means that the state of this agent, called leader, is always zero. The leader then is passive and no control input should be applied to him to control the internal structure of the formation. To enable this, the interconnection graph has to be directed and the adjacency matrix has to be asymmetric. Connections to the leader are possible, but connections from the leader are forbidden. This leads to the following condition for the elements of an adjacency matrix $\boldsymbol{A}_j$ and a fixed, passive leader l.

$$a_{ij} = 0 \iff i = l \tag{2.214}$$

2.9 Formation Control

Formation control is defined as the methods which allow for acquisition, maintenance or alteration of the coupled relative state vectors of the satellites in a formation. In contrast to absolute control of satellites, whose point of reference is usually the center of the Earth, relative control uses a local point of reference. This point depends on the chosen control law and thereby on the utilized local coordinate frame. It can be, among others, the origin of the EH frame $\mathscr{H}$ or the absolute COE set of the leader (e.g. while using DOEs or ROEs). Various research on general relative control has been conducted so far, but mostly with a focus impulsive control and maneuvers among two satellites, namely RvD. This work focuses on CLT control of CubeSat formations, for example the NetSat mission. This section gives an overview and a classification of different relative control schemes.

2.9.1 General Description of a Formation Flying Controller

An arbitrary control loop of a single satellite within a formation can be described as shown in Fig. 2.42. As input to the controller and as definition of the target state, either a formation (for maintenance) or a trajectory (for acquisition or transition) can act as input.

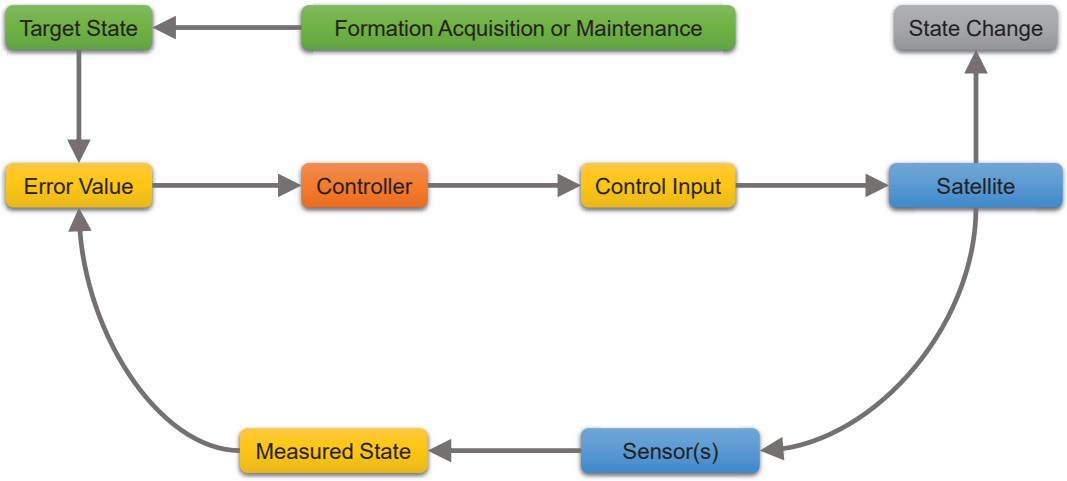


Figure 2.42: General description of a control loop of a single satellite within a formation.

2.9.2 Historical Development of Continuous Low-Thrust Control

The slowly emerging field of CLT relative control started with optimization techniques. Early CLT relative control was conducted by posing an optimization problem,

preferably linearly. Armellin et al. (2004) developed such a controller with linear optimization for unperturbed dynamics of formations smaller than 1 km. This is the basis for the Model Predictive Control (MPC) described in Chap. 2.7.4. Another notable work on CLT is a long-term asteroid rendezvous maneuver described by Schattel et al. (2016). Despite focusing on the rendezvous, it gives insight to thrust modeling techniques. The impeding mismatch to the pursued controller design is the absence of a guiding central gravity field in the vicinity of an asteroid in deep space. Gao and Li (2010) present a CLT control approach which combines global optimization with Control Lyapunov Function (CLF) (see Chap. 2.7.3). It is an off-line control technique and on account of this rather a trajectory planner than a controller. A further approach in the field of CLT trajectory planning is the in-plane Geostationary Transfer Orbit (GTO) planner by Konstantinov et al. (2016). The CLT relative controller by Edlerman and Kronhaus (2016) concentrates on long-term constraint fulfilling in terms of attitude maneuvers and power budgeting. It is, however, only suitable to control one element of the ROE state vector, namely the relative semi-major axis ∂a. An on-line CLT relative controller for elliptic reference orbits is described by Sherrill (2013). It belongs to the class of Lyapunov controllers and uses Floquet's theory to exploit the periodicity of the linear differential equations representing the relative dynamics. Again, this work focuses on rendezvous and very close formation configurations. Another type of CLT trajectory planners is comprised of non-linear multi-objective optimizations. Varga and Perez (2016) uses a Multi-Objective Evolutionary Algorithm (MOEA) to solve many-revolution low-thrust maneuvers obeying J_2 and eclipse phases. Its underlying problem description is based on the Q-law by Petropoulos (2005). The Q-law is a complex Lyapunov controller, whose CLFs are proximity quotients to the target. They are obtained through analytical models of the maximum rate of change of the differential orbital elements. To better understand the influence of J_2 on controller performance, the reader is referred to Djojodihardjo (2014). The publication extensively examines the J_2-effect, but is only valid for near-equatorial reference orbits. A conclusion about the importance of J_2 modeling in controller design is drawn in Breger and How (2007). They state that the ΔV consumption of a J_2-including controller used in a realistic environment is comparable to the consumption of a non-J_2-including controller used in a non-perturbed environment. This means most of the ΔV needed to mitigate the non-spherical relative effects of the Earth can be avoided by taking these effects into account during controller design.

A detailed overview of different control strategies for SFF is given in Chap. 2.9.5.

2.9.3 Classification

There are many criteria to classify control strategies. One previously mentioned is the distinction by thrust model constraints. As position control is conducted with electric propulsion, it is necessary to have a continuous thrust model. The electric propulsion system of the NetSat satellites provides a combined thrust in the order of

magnitude of 1×10^{-4} N. Using impulsive thrust models and control strategies would lead to significant errors with the ultra-low force provided by electrical propulsion. Even a very small orbit correction of $0.1\,\mathrm{m\,s^{-1}}$ takes approximately half an hour. In this time, the anomaly changes about 110° for the given mission specifications, which is clearly too large. It is important to note that a continuous thrust model is meant in the sense of long-duration thrusting action. This also includes time-discrete models where thrust is assumed to be constant over a short period of time.

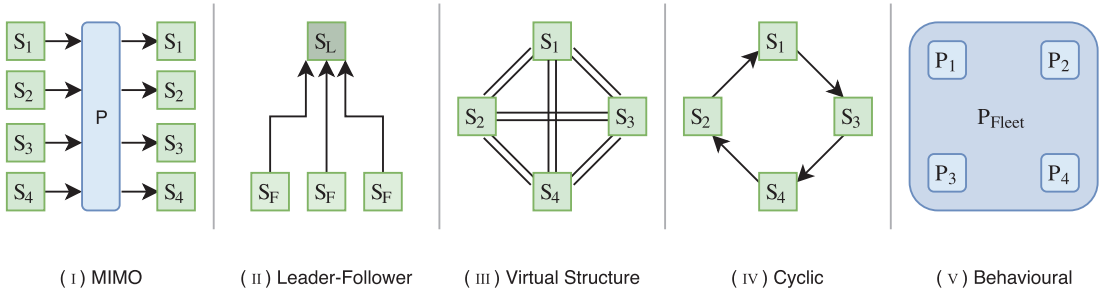


Figure 2.43: Formation control architectures as defined by Scharf et al. (2004). P: controller plant, S: satellite.

Scharf et al. (2004) provide a classification of the formation structure into five different architectures. The structures of all types defined in their controller survey is shown in Fig. 2.43. Without loss of generality, all types are shown with a formation of four satellites. They are distinguished by the inter-satellite interconnection schemes (depicted in green) or by controller plant structure (depicted in blue). Multiple-Input Multiple-Output (MIMO) controllers (cf. Fig. 2.43i) combine all satellites of a formation to a single plant. The state vector of this formation-level controller plant is the concatenation of the state vectors of all satellites. A leader/follower controller (Fig. 2.43ii) is a hierarchical structure with one leader and multiple follower(s). This is the classical approach of relative satellite dynamics, which emerged from RvD maneuvers. The followers do not have knowledge about the relative states among themselves. They only control their state in reference to the leader. The leader can be a randomly defined leader satellite or a virtual formation center without a spacecraft at the origin. In a controller based on a virtual structure (Fig. 2.43iii), the formation is treated as a rigid body. The satellites are particles in this body, which are all connected by dynamic links. Each satellite has its own plant which includes rigid body motion and dynamics like contraction and expansion descriptions of the links. The links directly correspond to the relative distances between the spacecraft. A virtual satellite structure can always be broken down into a leader-follower or a cyclic controller, depending on the implementation. Like the virtual structure, also the cyclic method (Fig. 2.43iv) is a non-hierarchical controller. This type can be understood as a cyclic connection of leader-follower pairs, where each leader is the follower of another spacecraft. This means that each satellite has exactly one leader and one follower. Cyclic controllers can be modeled with dependency digraphs or potential fields. Both cases render stability proofs of feedback controllers very

complicatedly (or even impossibly), since the feedback function grows in complexity with the number of spacecraft. The last controller type is the so-called behavioral plant architecture (Fig. 2.43v). It consists of a hierarchical plant structure. Each satellite has (multiple) single-state controllers at the lower control level to pursue a desired satellite state. The higher level is the fleet or formation plant, which pursues high-level goals concerning all spacecraft. The name of this type comes from the goals of the different plants, which are contrary and sometimes even mutually exclusive. Thus, the satellites may show competing behaviors and a strategy has to be found which combines these to achieve the highest-level goal. Depending on the specific implementation, this control type can be broken down into a structured combination of MIMO, leader-follower and cyclic controllers.

Due to the two structural breakdowns, the whole classification results in three independent structures, namely MIMO, leader-follower and cyclic. Scharf et al. (2004) conclude that MIMO is preferable in terms of ΔV, because of its guaranteed global optimality as a direct consequence of the plant synthesis. Drawbacks are a possible high impact of single-satellite failures and the high amount of needed input information and inter-satellite communication. To synthesize the plant, each satellite must know the relative state to every other spacecraft of the formation. In contrast, a leader-follower architecture needs less information input. Each follower is only connected to one leader, reducing the necessary inter-satellite communication. The reduced inter-satellite communication requirement also increases error robustness. An error in a follower satellite does not affect any other satellite of the formation. Even if a leader-follower configuration provides locally optimal solutions, a disadvantage arises when considering formation-level optimality – the combination of multiple locally optimal solutions does not automatically result in a globally optimal solution. A cyclic architecture provides a mixture of the advantages and drawbacks of the two aforementioned strategies. They may perform better in terms of ΔV compared to leader-follower controllers and furthermore can provide a more balanced distribution of ΔV consumption. However, communication effort is increased due to the higher amount of inter-satellite connections. As mentioned above, a major downside of this structure is its lack of sophisticated stability-proving techniques. This is a very important step of controller design, since global as well as local stability of the controller has to be guaranteed.

Another scheme is to distinguish into open-loop and closed-loop controllers. Controllers without feedback may be further divided by their execution period. If this period extends over several orbits, the on-line controller may be viewed in the same way as an off-line trajectory planner. In other words, a trajectory planner can be understood as a long-term controller without feedback. Other schemes are based on (non-)linearity or time invariance. Instances of those can belong to any combination of the previously mentioned classifications, as long as they are not mutually exclusive. Furthermore, the goal to be achieved may change the intrinsic behavior of a controller. The type of the goal can be distinguished by the reference trajec-

tory. In formation maintenance phases, this reference is constant. During formation acquisition/reconfiguration maneuvers, which include the initial acquisition phase, the reference trajectory is time-varying until the desired formation is obtained. This may be modeled with a single target trajectory or a continuously altering reference, obtained by a so-called reference governor.

2.9.4 Discretization

In CLT models, as said by the name, the thrust may change continuously. Thus, the thrust input has to be discretized to an appropriate level of accuracy. Normally, this is done with a ZOH approach. Fig. 2.44 shows this discretization procedure at the example function $f(t) = \sin(t)$ with a discretization step size of 1. ZOH means that the level of the very time instant of a discretization step is maintained by the discretized function until the next discretization step is reached. In the example figure, the step size is clearly too high to represent the original function at a decent level. In controller design, the step size has to be adapted to an extent that is small enough so that only minor changes in thrust input occur during one time step. This is also applicable to continuous thrust inputs (from non-linear models), which should be accommodated in the state space equation of a relative orbit propagator.

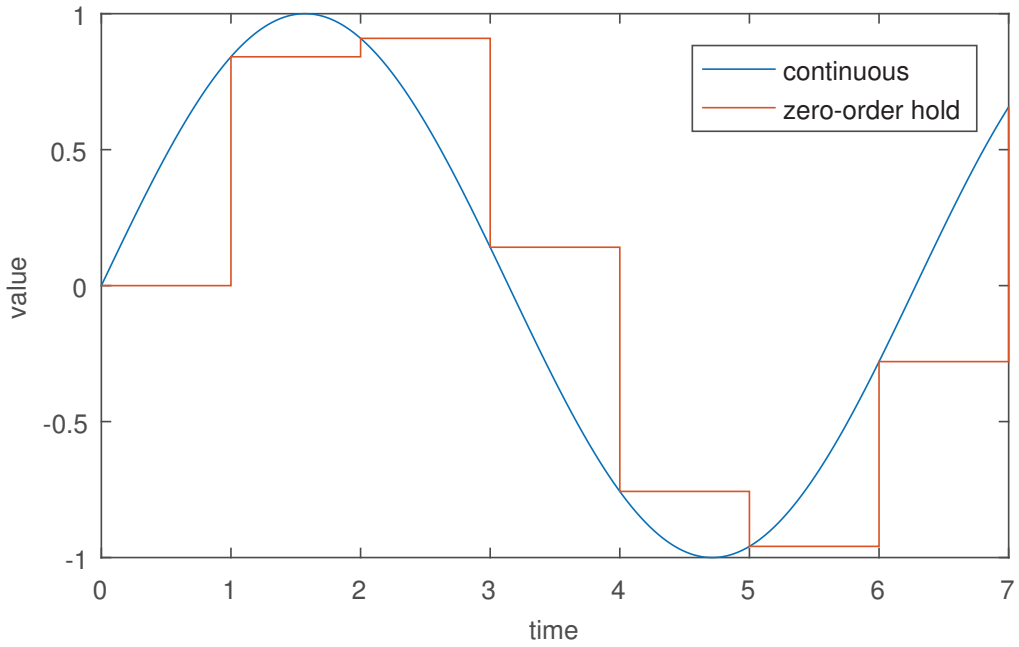


Figure 2.44: Continuous function $f = \sin(t)$ and its discretized counterpart with Zero-Order Hold assumption with $\Delta t = 1$.

2.9.5 Overview of Different Control Strategies for Satellite Formation Flying

Extending Chap. 2.9.2, this section provides an overview of different control strategies for SFF discussed in literature.

Optimal Bounded Low-Thrust Rendezvous with Fixed Terminal Approach Direction

Guelman and Aleshin (2001) provide a closed form solution of unbounded low-thrust propulsion for a given time interval and both initial and final conditions. The non-linear bounded case is numerically solved by employing MATLAB functions. They divide rendezvous into two stages:

1. Transfer to an intermediate point lying on the desired approach vector. This intermediate point is usually in close proximity to the target.

2. Fixed terminal approach along a given approach vector for rendezvous.

For the purpose of SFF only the first stage is of interest, since no rendezvous is foreseen. Bevilacqua (see next section) has used Guelman's approach and applied this method on the Sedwick-Schweighart STM (cf. Chap. 2.5.3.4) instead of the HCW equations to also include the J_2 orbit perturbations.

Multiple Spacecraft Rendezvous Maneuvers by Differential Drag and Low Thrust Engines

Bevilacqua et al. (2010) employ differential drag for in-plane reconfiguration by using anti-along-track acceleration. In order to do so, they use steerable drag-plates. Those are not of interest for CLT SFF. But as the anti-along-track acceleration does not allow for out-of-plane maneuvers they propose a second maneuvering stage where they suggest low-thrust propulsion as Guelman did (see Chap. 2.9.5). Further they refine Guelman's solution by applying it on the Sedwick-Schweighart STM which includes periodic J_2 orbit perturbation (for full orbit periods only).

Optimal Impulsive Closed-Form Control for Spacecraft Formation Flying and Rendezvous

Riggi and D'Amico (2016) propose to use three impulses for in-plane reconfiguration and one impulse for out-of-plane reconfiguration, both under the influence of J_2 for nearly circular orbits. Further, a solution is found for non-circular orbits by neglecting J_2.
Closed-form solutions in the ROEs are provided. The open question remains, how to incorporate low-thrust solutions as found e.g. by Guelman (see Chap. 2.9.5) into this approach. Disregarding of the loss of optimality, possibly thrust periods around the optimal impulsive thrust locations can be proposed.

Analytical Mechanics of Space Systems

Schaub and Junkins (2003) developed a controller based on the Lyapunov theory using mean orbital elements. Using mean orbital elements in the feedback loop allows to control each orbit element independently of the others. Instead, if feeding back position and velocity, one always controls all orbit elements at a time. It is known that orbit elements should be changed at defined locations in order to optimize fuel-consumption.

This controller includes J_2 perturbation and assumes that the error between the desired and actual state is small. For the purpose of satellite formation maintenance this assumption also holds true. Feeding back six constraints for a 3-dimensional control vector is an overdetermined problem. Therefore, Schaub and Junkins (2003) use a Least Square Method to solve for the thrust vector. Thus, the controller no longer ensures stability. Furthermore, this controller does not yet include an upper limit for thrust.

Adaptive Nonlinear Control of Multiple Spacecraft Formation Flying

The approach of de Queiroz et al. (2000) considers the full nonlinear dynamics describing the relative positioning of multiple satellites for control design. Using Lyapunov-based control design and stability analysis techniques, they develop a nonlinear adaptive control law that guarantees global asymptotic convergence of the spacecraft' relative position to any sufficiently smooth desired trajectory, despite the presence of unknown, constant, or slow-varying spacecraft masses, disturbance forces and gravity forces. In this paper spacecraft actuators are considered to be capable of providing continuous time control efforts, as opposed to being of impulsive type. The controller adapts the system's constant parameter vector θ

$$\boldsymbol{\theta} = \begin{bmatrix} m_f & m_f MG & \dfrac{m_f}{m_l} & F_{d,x} & F_{d,y} & F_{d,z} \end{bmatrix} \tag{2.215}$$

where the subscript f denotes the follower spacecraft and the subscript l the leader spacecraft. The vector $\boldsymbol{F}_d$ is the total constant disturbance force vector. This constant disturbance force vector is the major limitation of this controller, because disturbance forces in LEO are mostly not constant and significantly varying in direction. Apart from this $\boldsymbol{\theta}$ is unknown at the initialization and will be found by the controller. The feedback quantity is the filtered tracking error $\boldsymbol{r}(t) = \dot{\boldsymbol{e}}(t) + \Lambda \boldsymbol{e}(t)$ with the position tracking error $\boldsymbol{e}$ and the control gain matrix $\Lambda \in \mathbb{R}^{3\times3}$, which is constant, diagonal and positive-definite. The control law finally is

$$\boldsymbol{u}_f = \boldsymbol{W}\hat{\boldsymbol{\theta}} + \boldsymbol{K}\boldsymbol{r} \tag{2.216}$$

Here, $\boldsymbol{W}$ is the regression matrix of known quantities describing the spacecraft's motion, and $\boldsymbol{K} \in \mathbb{R}^{3\times3}$ is a constant, diagonal, positive-definite control gain matrix.

The parameter gets updated by the following relationship

$$\dot{\hat{\theta}} = \boldsymbol{\Sigma} \boldsymbol{W}^T \boldsymbol{r} \tag{2.217}$$

where $\boldsymbol{\Sigma} \in \mathbb{R}^{6\times6}$ is a constant, diagonal, positive-definite adaption gain matrix. The paper provides an example and also states the exact values of all gain matrices included.
In the provided example the simulation ran with an initial $\boldsymbol{\theta} = 0.5 \cdot \boldsymbol{\theta}_{real}$. The controller required about 30 h to find the real parameter vector. In the aforementioned realistic case of non-constant disturbance forces, it is not clear how fast $\boldsymbol{\theta}$ can converge.

Linear-Quadratic Stationkeeping for the STS Orbiter

Redding et al. (1989) developed a regulator for the Space Shuttle (Space Transportation System (STS)) as continuous closed-loop control for maneuvering during stationkeeping with other spacecraft to help the on-board crew and to optimize the process. A threshold for minimum required thrust levels is introduced, beneath which the actuators won't fire. Further, also an upper limit is introduced to represent the maximum available thrust. For fuel optimality usually two maneuvers are performed, but for high precision stationkeeping shorter duty cycles have to run on the On-Board Computer (OBC). This controller design introduces a minimal time that has to pass between any two firings. The authors justify this by the fact that fuel optimal solutions have two maneuvers. But for our purposes of CLT maneuvering only (infinitesimal) short time periods should be between any maneuvers. Further, smaller sample times are advantageous in extreme disturbance environments and/or when tight formations are required. The quadratic cost function is defined as

$$J = \sum_{i+1}^{\infty} \left[\boldsymbol{x}_i^{\mathrm{T}} \boldsymbol{Q} \boldsymbol{x}_i + \boldsymbol{\Delta v}_i^{\mathrm{T}} \boldsymbol{R} \boldsymbol{\Delta v}_i \right] \tag{2.218}$$

and the constant gain controller has the form

$$\Delta \boldsymbol{v}_i = \boldsymbol{K}(\boldsymbol{x}_c - \boldsymbol{x}_i) \tag{2.219}$$

which is computed to minimize J. The cost matrices $\boldsymbol{Q}$ and $\boldsymbol{R}$ are found by applying Bryon's rule of thumb. Therefore, both cost matrices are diagonal, with elements q_{ii} and r_{ii} equal to the inverse square of the nominal deviation of the i^{th} output $\boldsymbol{x}_{nom}$ and $\boldsymbol{\Delta v}_{nom}$.

Sliding Mode Control for Low-Thrust Earth-Orbiting Spacecraft Formation Maneuvering

> "Sliding mode control has been recognized as one of the key approaches for complex nonlinear dynamic systems operating under system uncertainties and disturbances." (Liu et al., 2006)

Liu et al. (2006) developed a sliding mode control using nonlinear dynamics. Two tasks are distributed between a leader and a follower satellite. The leader is controlled independently of the follower and maneuvers itself into a desired absolute orbit. The follower satellite on the other hand holds relative position and velocity information available. Its controller keeps the follower in the desired relative geometry with respect to the leader satellite. The interesting task for our purposes is the second one, where the follower maintains its desired relative geometry. This approach is not fuel optimal, but guarantees to be globally stable. Using nonlinear dynamics improves fuel consumption. But bounded low-thrust is not taken into account. This sliding mode controller calculates the required thrust with no restrictions according to the thruster capabilities.

Control of Satellite Formations

> "A quadratic cost function is used to estimate the fuel required for analytical simplicity as well as its applicability for power-limited low-thrust propulsion." (Vadali et al., 2001)

The controller presented by Vadali et al. (2001) is based on the HCW equations which are modified to include the effect of J_2. They formulate the following extended HCW equations (given in an alternative definition of the LVLH coordinate frame, cf. App. A.1 and A.2)

$$\ddot{x} - 2(\bar{n}_0 + \varepsilon)\dot{y} - 3(\bar{n}_0 + \varepsilon)^2 x = u_x \tag{2.220a}$$

$$\ddot{y} + 2(\bar{n}_0 + \varepsilon)\dot{x} = u_y \tag{2.220b}$$

$$\ddot{z} + \bar{n}_0^2 z = u_z + 2A\bar{n}_0 a_0 \cos\alpha \sin\theta_0 \tag{2.220c}$$

where ε is the difference between the in-plane and out-of-plane angular rates. Further,

$$A = \frac{3}{2} J_2 n_0 \left(\frac{R_e}{a_0}\right)^2 \sin\left(i_0\right)^2 c_2 \tag{2.221}$$

$$\bar{n}_0 = \dot{\omega}_0 + \dot{M}_0 \tag{2.222}$$

The cost function is

$$J = \frac{\bar{n}_0}{4\pi^2} \int_0^{2\pi/\bar{n}_0} \int_0^{2\pi} (u_x^2 + u_y^2 + u_z^2) d\alpha(0) dt \tag{2.223}$$

This controller ensures all spacecraft to experience the same ΔV consumption, while J_2 perturbation is counteracted. An average of 12.3% less fuel consumption than that required by the “brute force” approach is claimed.

Long-Term Formation Keeping of Satellite Constellation Using Linear-Quadratic Controller

Ulybyshev (1998) verified the applicability of LQR control in the scope of formation maintenance and long-term tangential maneuvers. In the CanX-4 & 5 mission, LQR control was used for formation keeping as well as formation reconfiguration (Eyer, 2009). They showed that it is favorable to select different gain matrix ratios during the two maneuver types. Both sources did not use CLT but rather a maximum thrust level in the order of magnitude of 1 N.

J_2-Modified GVE-Based MPC for Formation Flying Spacecraft

Non-low-thrust examination of MPC in relative formation control was conducted by Breger and How (2005) with promising results. Later on, they extended their model to use equinoctial orbital elements to remove some singularities (Breger and How, 2007). A terminal-approach (rendezvous) maneuver analysis is given by Hartley et al. (2012). Both results encourage the use of MPC in formation control, no matter if using a Cartesian or an orbital state vector.

Conclusion

Several publications on CLT satellite formation control can be found. But yet, there is no solution capable of providing fuel-optimal reconfigurations, which works in a distributed manner or takes various constraints (like very low thrust or limited inter-satellite communication) into account. Some aspects are covered by some publications, though. Few controllers use models including J_2 disturbances to encounter for perturbations. Rarely differential drag, third body or solar radiation pressure (and shadowing) are considered. The investigated approaches do not provide detailed information about stability, no stability analysis is shown. In summary, many aspects are covered in different publications, though no suitable solution covering all requirements of low-thrust CLT SFF control is present.

Chapter 3

Satellite Formation Control

3.1 Overview

In this thesis, three different control approaches have been developed. They were designed with a special focus on CLT SFF and corresponding missions like NetSat. The three approaches use different control concepts, namely *MPC-Based Formation Control*, *Distributed Robust Consensus-Based Control* and *Plant Inversion-Based Lyapunov Control Combined with a Reference Governor*. All of them are described in the following sections. An evaluation of the control approaches follows in Chap. 5.

3.2 MPC-Based Formation Control

This section gives a detailed view on the specific MPC implementation developed in this thesis. It is based on the MPC definition in Chap. 2.7.4 and starts with describing the overall mathematical formulation, followed by the definition of the constraints and presenting the complete MPC program flow logic.

3.2.0.1 General Implementation

The developed MPC based on Chap. 2.7.4 is formed by the scalar cost function in Eq. 2.175 with additional imperative constraints. The internal state vector of the controller is the mean DOE element set

$$\delta\bar{\boldsymbol{a}} = \begin{pmatrix} \delta a & \delta e & \delta i & \delta\omega & \delta\Omega & \delta M \end{pmatrix} \tag{3.1}$$

This set is used to match the output of the GVE Eq. 2.98, which is used for the control inputs. The complete controller description is given by

$$J = \min_{\hat{\boldsymbol{U}}_k} \boldsymbol{C}^T \hat{\boldsymbol{U}}_k \tag{3.2}$$

subject to the constraints

$$\boldsymbol{A}_{eq}\hat{\boldsymbol{U}}_k = \boldsymbol{b}_{eq} \tag{3.3a}$$

$$\boldsymbol{A}_{ieq}\hat{\boldsymbol{U}}_k \leq \boldsymbol{b}_{ieq} \tag{3.3b}$$

where $\hat{\boldsymbol{U}}_k$ is the concatenated control input over the planning horizon Eq. 2.172, $\boldsymbol{C}$ is the weighting matrix, $\boldsymbol{b}_{eq}$ and $\boldsymbol{b}_{ieq}$ the final state error the controller has to overcome and $\boldsymbol{A}_{eq}$and $\boldsymbol{A}_{ieq}$ matrices that define the constraints the MPC computation is subject to (cf. Chap. 2.7.4.3).

3.2.0.2 Definition of Constraints

Opposing to the deviation in Chap. 2.7.4, the final relative orbit is enforced with the equality constraint Eq. 3.3a, whose reasons are stated lateron in this section. All other restrictions are encapsulated in the inequality constraint Eq. 3.3b. In one execution step, the MPC not only computes the control input for the current time step, but for all subsequent steps until the end of the planning horizon t_f. This also includes controller-internal orbit propagation over quite a large time to compute the matrix elements of the equality constraint. When using Hill dynamics in this time domain, the deviations grow too large for an effective control plan. Hence, the MPC uses more exact dynamics in the equations of motion. However, better dynamics can be used in any controller and thus are not a characteristics of a good controller. The right hand side of the equality constraint Eq. 3.3a is the final state error $\boldsymbol{b}_{eq}$, which the controller has to overcome. The computation of $\boldsymbol{b}_{eq}$ is independent of

the control input (i.e. the uncontrolled dynamics) and can be done beforehand the actual MPC computation. For $\boldsymbol{b}_{eq}$, even the eccentric dynamics derived in Chap. 2.5.3.2 are too inaccurate. The ultimate solution is to use a modified GA STM as propagator developed by Gim and Alfriend (2003) and extended by Reinthal (2016, pp. 73). With this propagator, the spacecraft position is propagated until the end of the planning horizon. The result is then converted to mean DOEs and subtracted from the desired mean DOE vector at the final time t_f

$$\boldsymbol{b}_{eq} = \delta\bar{\boldsymbol{a}}_{r,f} - \boldsymbol{\Phi}_{MGA}\left(\delta\bar{\boldsymbol{a}}_0, t_f\right) \tag{3.4}$$

where $\boldsymbol{\Phi}_{MGA}\left(\boldsymbol{x}, t\right)$ gives the state vector $\boldsymbol{x}$ at any time t with the method described. The left hand side of the equality constraint $\boldsymbol{A}_{eq}$ equals Eq. 2.165a and is repeated here for clarification.

$$\boldsymbol{A}_{eq} = \begin{bmatrix} \boldsymbol{\Phi}^{(k-1,k)}\boldsymbol{G}_0 & \boldsymbol{\Phi}^{(k-2,k)}\boldsymbol{G}_1 & \cdots & \boldsymbol{\Phi}^{(0,k)}\boldsymbol{G}_{k-1} & \boldsymbol{0} \end{bmatrix} \tag{3.5}$$

It consists of the concatenated control input matrices convoluted with the discrete state transitions for the respectively appropriate time frames. The control impact $\boldsymbol{G}_k$, defined in Eq. 2.161b is propagated for one time step T_s with the differential orbit STM

$$\boldsymbol{A}_k = \boldsymbol{\Phi}_{DOE}\left(0, dt\right) = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ -\frac{3ndt}{2a} & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \tag{3.6}$$

If the differential semi-major axis vanishes, the DOE STM degrades to an identity matrix. If not, δa has an influence on the relative argument of latitude (lower left matrix element). This STM neglects all perturbing forces on the state vector. It turned out to be sufficient to use these basic dynamics for $\boldsymbol{A}_{eq}$. For the investigated control input levels in the order of magnitude of $1 \times 10^{-5}\,\mathrm{m\,s^{-2}}$, the influence on the absolute state is so small that the state transition matrix Eq. 3.6 is accurate enough. For the matrix $\boldsymbol{G}_k$, the very same $\boldsymbol{A}_k$-matrix was used, giving

$$\boldsymbol{G}_k = \int_0^{dt} e^{\boldsymbol{A}_k \tau} d\tau \boldsymbol{B}\left(k\right) \approx \begin{bmatrix} T_s & 0 & 0 & 0 & 0 & 0 \\ 0 & T_s & 0 & 0 & 0 & 0 \\ 0 & 0 & T_s & 0 & 0 & 0 \\ 0 & 0 & 0 & T_s & 0 & 0 \\ 0 & 0 & 0 & 0 & T_s & 0 \\ -\frac{3nT_s}{4a}dt & 0 & 0 & 0 & 0 & T_s \end{bmatrix} \boldsymbol{B}\left(k\right) \tag{3.7}$$

with a ZOH assumption on $\boldsymbol{A}_k$, a total propagation time $dt = t_f - t$ and the GVEs in Eq. 2.98 for the control influence $\boldsymbol{B}(k)$. The GVEs are dependent on the instantaneous osculating orbit. The orbit is obtained by propagating the mean orbital elements using the mentioned modified GA STM. The result is then converted using

a Brouwer transformation (Eq. 2.57) to an osculating element set.

The yet not mentioned inequality constraint Eq. 3.3b consists of the maximum thrust constraint Eq. 2.177 and the rate limit Eq. 2.171.[1] The matrix form over the planning horizon for the rate limits is

$$\begin{bmatrix} \boldsymbol{V} \\ -\boldsymbol{V} \end{bmatrix} \boldsymbol{U}_k \leq \begin{pmatrix} \boldsymbol{R}_k^{\max} \\ \boldsymbol{R}_k^{\min} \end{pmatrix} \tag{3.8}$$

with

$$\boldsymbol{V} = \begin{bmatrix} -1 & 0 & 0 & 1 & 0 & \cdots & 0 \\ 0 & -1 & 0 & 0 & 1 & & \vdots \\ \vdots & & \ddots & & & \ddots & \\ 0 & \cdots & & 1 & 0 & 0 & -1 \end{bmatrix} \tag{3.9}$$

$\boldsymbol{R}_k$ is defined analogue to $\boldsymbol{U}_k$ in Eq. 2.170. Both constraints can be combined to one single inequality

$$\boldsymbol{L}\boldsymbol{U} \leq \boldsymbol{d} \tag{3.10}$$

with

$$\boldsymbol{L} = \begin{bmatrix} \boldsymbol{I} \\ -\boldsymbol{I} \\ \boldsymbol{V} \\ -\boldsymbol{V} \end{bmatrix} \tag{3.11}$$

and

$$\boldsymbol{d} = \begin{pmatrix} \boldsymbol{U}_k^{max} \\ \boldsymbol{U}_k^{min} \\ \boldsymbol{R}_k^{max} \\ \boldsymbol{R}_k^{min} \end{pmatrix} \tag{3.12}$$

The MPC inequality Eq. 3.3b uses the extended input vector Eq. 2.172 to account for positive and negative input while still posing a minimization problem. This format is achieved by extending the previously stated matrices appropriately

$$\boldsymbol{A}_{ieq} = \begin{bmatrix} \boldsymbol{L} & -\boldsymbol{L} \\ -\boldsymbol{I} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{I} \end{bmatrix} \tag{3.13a}$$

$$\boldsymbol{b}_{ieq} = \begin{pmatrix} \boldsymbol{d} \\ \boldsymbol{0} \\ \boldsymbol{0} \end{pmatrix} \tag{3.13b}$$

With this, the minimization problem formulation Eq. 3.2 is complete and can be solved by linear programming techniques.

[1] A specific thrust profile or rate limit that a real-world propulsion system requires or enforces, can be implemented as time-variant function set.

3.2.0.3 Complete MPC Program Flow Logic

The whole MPC process is depicted in Fig. 3.1. The control loop starts at the green input variables and terminates in the blue thrust profile if a solution is found by the MPC_f-block, which solves the constrained minimization problem Eq. 3.2. If not, the input is adjusted over and over until the process terminates. The diagram contains two special steps for reconfiguration maneuvers, which are explained later. The actual model predictive logic is contained in the yellow block MPC_f. This is in fact the process of finding a solution to Eq. 3.2 under the given constraints and is conducted with MATLAB's linear programming function *linprog*.

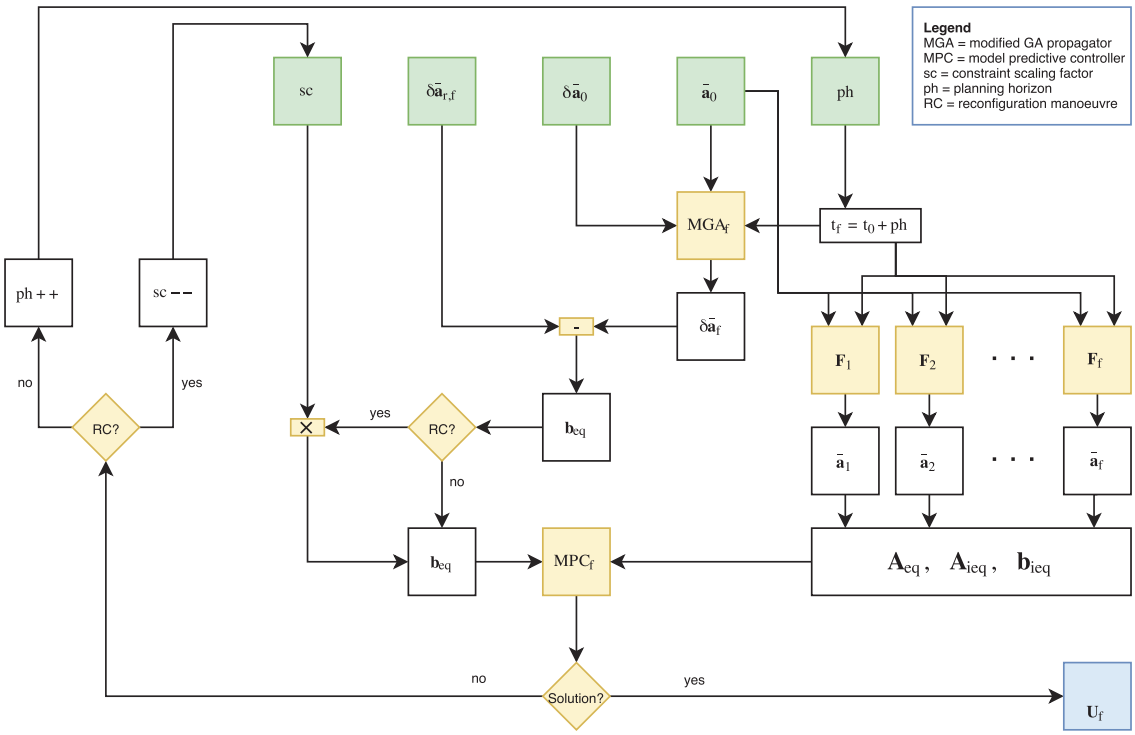


Figure 3.1: Diagram showing the complete MPC logic. Input is depicted in green, output in blue.

Since there are no error tolerances or a boundary box included for the final state, the problem is not always feasible. In contrast to the deviation in Chap. 2.7.4 and the relative orbit MPC analysis by Tillerson et al. (2002), this design step was done on purpose to reduce the total ΔV consumption. When the terminal orbit constraint is not feasible and an adjustable error margin is used, the thrusters will operate at their maximum rates over a large time, which is not desirable. The influence of the control acceleration rises and falls with the sines and cosines of the true anomaly and the argument of latitude (see Eq. 2.98). In the times in between with low control throughput, ΔV is wasted. In contrast to the cited source, the developed controller varies the input parameters when it hits an unsolvable minimization problem. In formation maintenance mode, the planning horizon is increased, giving the

spacecraft more time to counteract any state errors. This should only occur after uncontrolled phases in a given configuration (like in a period foreseen for inter-satellite communication). If the problem would not be feasible after a controlled phase (i.e. almost no state error in the beginning), it means that relative perturbation forces are larger than the maximal control output. In this case, also an extended planning horizon does not lead to feasible solutions, since the perturbation acceleration increases analogously. For formation acquisition/reconfiguration maneuvers, a scaling factor for the terminal orbit constraint is used. In one planning period, it is only compensated for a fraction of the difference between desired orbit and current orbit. This is necessary for almost any maneuver conducted with CLT, since it takes a very long time for noticeable orbit changes. When no valid thrust profile is found by the MPC, this scaling factor is lowered, reducing the orbit difference to overcome during one planning interval.

3.3 Distributed Robust Consensus-Based Control

In this section, a combination of distributed control using the consensus approach introduced in Chap. 2.8 and robust H_∞-control inducted in Chap. 2.7.5 is presented. It mainly comprises contents (namely the mathematical derivation of the generalized plant description P and Fig. 3.2) developed by the author and published in Scharnagl et al. (2019). The main objectives of control are reference tracking for acquiring a given relative formation topology, together with disturbance rejection (e.g. errors of actuator systems) under the presence of (sensor) noise. In addition, the required control input should be optimized. Thus a mixed sensitivity approach is chosen, which is presented in the following section.

3.3.1 Mixed Sensitivity Closed-Loop System with Distributed Controller Interconnections

Based on the work presented in Chp. 2.8.2.1 and especially in Fig. 2.40 on the distributed consensus approach, we first compile a block diagram of the closed-loop system of a single agent i with mixed sensitivity robustness and an arbitrary number n of interconnections to other agents' controllers. This is presented in Fig. 3.2.

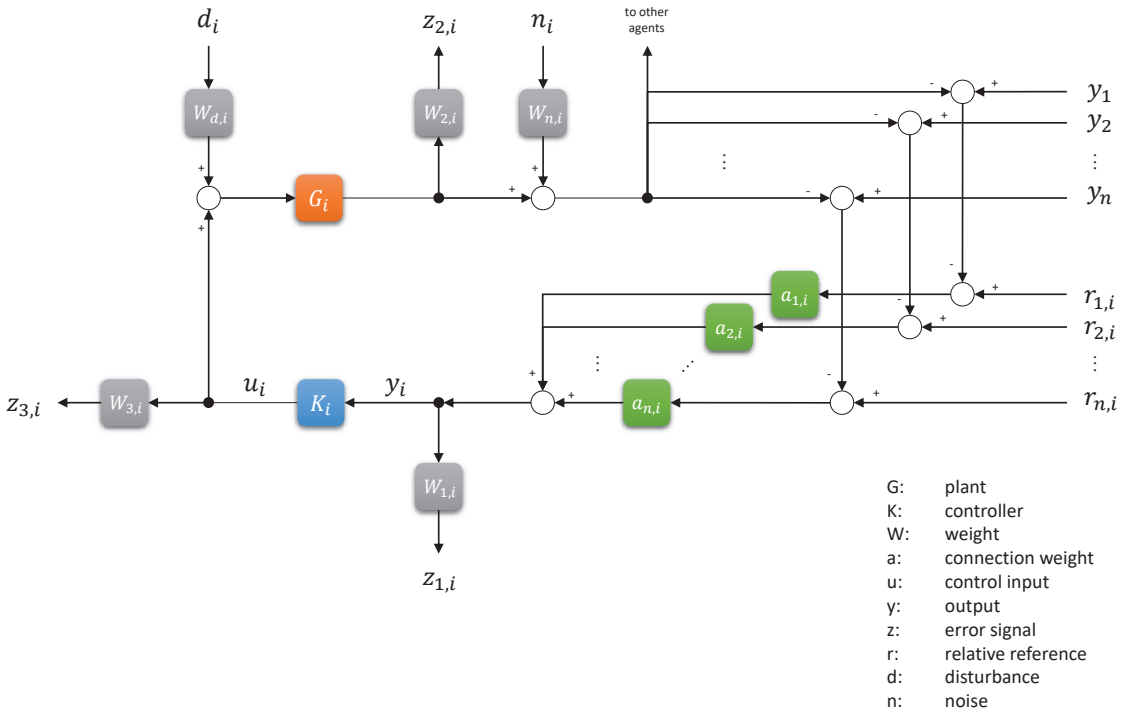


Figure 3.2: Mixed sensitivity closed-loop system with distributed controller interconnections.

3.3.2 Obtaining Generalized Plant for Single Agent

From the block diagram in Fig. 3.2 the complete generalized plant for mixed sensitivity with distributed consensus based reference inputs for agent i with interconnections to N other agents can be derived

$$P_i = \begin{array}{c|c} & \begin{array}{ccccccc} d & n & y_1 & y_2 \ldots y_N & r_{1i} & r_{2i} \ldots r_{Ni} & u \end{array} \\ \hline \begin{array}{c} z_1 \\ z_2 \\ z_3 \\ y \end{array} & \left(\begin{array}{ccccccc} W_1\, A_i\, G_i\, W_d & W_1\, A_i\, W_n & -W_1\, a_{1i} & \ldots & W_1\, a_{1i} & \ldots & W_1\, A_i\, G_i \\ W_2\, G_i\, W_d & 0 & 0 & 0 & 0 & 0 & W_2\, G_i \\ 0 & 0 & 0 & 0 & 0 & 0 & W_3 \\ \hline A_i\, G_i\, W_d & A_i\, W_n & -a_{1i} & \ldots & a_{1i} & \ldots & A_i\, G_i \end{array}\right) \end{array} \tag{3.14}$$

with y being the output of agent i, d the disturbance and n the noise. a_{ji} is the weight on the connection from agent j to agent i. In general case $a_{ji} \neq a_{ij}$, though in practical applications communication between agents is often bidirectional, which leads to $a_{ji} = a_{ij}$. Further, $A_i = \sum_j a_{ji}$ and r_{ji} is the reference for the relative state from agent j to agent i. Further $G_d = G_i$ is assumed here.

Computation of the Closed-Loop Transfer Function

The CLTF T_{zw} of an agent i can be computed by combining the generalized plant description given in Eq. 3.14 with a controller K_i using a lower LFT. By dividing the overall matrix into four block matrices M_{ij} Eq. 3.14 can be reformulated to

$$P_i = \begin{array}{c|c} & \begin{array}{cc} d/n/y/r & u \end{array} \\ \hline \begin{array}{c} z \\ y \end{array} & \left(\begin{array}{cc} M_{11} & M_{12} \\ \hline M_{21} & M_{22} \end{array}\right) \end{array} \tag{3.15}$$

The CLTF T_{zw} can be computed by combining the above matrix and the controller K_i in a lower LFT

$$T_{zw} = F_L(P_i, K_i) = M_{11} + M_{12}K_i\,(I - M_{22}K_i)^{-1}\,M_{21} \tag{3.16}$$

By introducing the sensitivity $S_i^* = (I - M_{22}K_i)^{-1} = (I - A_iG_iK_i)^{-1}$ the CLTF can be expressed as

$$T_{zw} = \begin{matrix} \\ z_1 \\ z_2 \\ z_3 \end{matrix} \begin{matrix} \begin{matrix} d & & n & & y_1 & y_2 \cdots y_N & r_{1i} & r_{2i} \cdots r_{Ni} \end{matrix} \\ \begin{pmatrix} W_1\ A_i\ G_i\ W_d + W_1\ A_i\ G_i\ K_i\ S_i\ A_i\ G_i\ W_d & W_1\ A_i\ W_n + W_1\ A_i\ G_i\ K_i\ S_i\ A_i\ W_n & -W_1\ a_{1i} - W_1\ A_i\ G_i\ K_i\ S_i\ a_{1i} & & W_1\ a_{1i} + W_1\ A_i\ G_i\ K_i\ S_i\ a_{1i} & \\ W_2\ G_i\ W_d + W_2\ G_i\ K_i\ S_i\ A_i\ G_i\ W_d & W_2\ G_i\ K_i\ S_i\ A_i\ W_n & -W_2\ G_i\ K_i\ S_i\ a_{1i} & \cdots & W_2\ G_i\ K_i\ S_i\ a_{1i} & \cdots \\ W_3\ K_i\ S_i\ A_i\ G_i\ W_d & W_3\ K_i\ S_i\ A_i\ W_n & -W_3\ K_i\ S_i\ a_{1i} & & W_3\ K_i\ S_i\ a_{1i} & \end{pmatrix} \end{matrix} \tag{3.17}$$

This equation can be simplified using the identities $1 + A_i\ G_i\ K_i\ S_i = S_i$ and $G_i\ K_i\ S_i = T_i$ (derived from the definition of the sensitivity function $S = (I + G\ K)^{-1}$ and the complementary sensitivity function $T = (I + G\ K)^{-1}\ G\ K$, cf. Skogestad and Postlethwaite (2001, pp. 22)):

$$T_{zw} = \begin{matrix} \\ z_1 \\ z_2 \\ z_3 \end{matrix} \begin{matrix} \begin{matrix} d & & n & & y_1 & y_2 \cdots y_N & r_{1i} & r_{2i} \cdots r_{Ni} \end{matrix} \\ \begin{pmatrix} W_1\ S_i\ A_i\ G_i\ W_d & W_1\ S_i\ A_i\ W_n & -W_1\ S_i\ a_{1i}\ a_{1i} & & W_1\ S_i\ a_{1i} & \\ W_2\ (1 + G_i\ K_i\ S_i\ A_i)\ G_i\ W_d & W_2\ T_i\ A_i\ W_n & -W_2\ T_i\ a_{1i} & \cdots & W_2\ T_i\ a_{1i} & \cdots \\ W_3\ K_i\ S_i\ A_i\ G_i\ W_d & W_3\ K_i\ S_i\ A_i\ W_n & -W_3\ K_i\ S_i\ a_{1i} & & W_3\ K_i\ S_i\ a_{1i} & \end{pmatrix} \end{matrix} \tag{3.18}$$

By further assuming $A_i = \alpha \cdot I$ (which is equivalent to the fact that a_{ji} weights connections between agents and not channels individually), we can use the identity $A_i \cdot X = X \cdot A_i$ and simplify the element $(2,1)$ of T_{zw}, which is the TF $T_{z_2,d}$. Further it is to note that, if $A_i = -I$, which means that there is no gain on the connections between the agents, the CLTF simplifies to

$$T_{zw} = \begin{matrix} \\ z_1 \\ z_2 \\ z_3 \end{matrix} \begin{matrix} \begin{matrix} d & & n & & y_1 & y_2 \cdots y_N & r_{1i} & r_{2i} \cdots r_{Ni} \end{matrix} \\ \begin{pmatrix} -W_1\ S_i\ G_i\ W_d & -W_1\ S_i\ W_n & -W_1\ S_i\ a_{1i}\ a_{1i} & & W_1\ S_i\ a_{1i} & \\ W_2\ S_i\ G_i\ W_d & -W_2\ T_i\ W_n & -W_2\ T_i\ a_{1i} & \cdots & W_2\ T_i\ a_{1i} & \cdots \\ -W_3\ K_i\ S_i\ G_i\ W_d & -W_3\ K_i\ S_i\ W_n & -W_3\ K_i\ S_i\ a_{1i} & & W_3\ K_i\ S_i\ a_{1i} & \end{pmatrix} \end{matrix} \tag{3.19}$$

which is identical to the classical mixed sensitivity CLTF of a single system (except of the additional reference inputs y_i and $r_{ji}, j \neq i$ and renaming a_{1i} to W_r).

3.3.3 Obtaining the Generalized Plant for the Overall System

Starting from the plant description of the distributed systems or agents in Eq. 3.14, we now compose the general plant of the overall system P. Thus we define $A = diag(A_1, A_2, ..., A_N)$ with $A_i = \sum_j a_{ji}$. We further assume that the individual agents are equal.

Thus

$$G = G_i \qquad W_1 = W_{1,i} \qquad W_2 = W_{2,i}$$
$$W_3 = W_{3,i} \qquad W_d = W_{d,i} \qquad W_N = W_{N,i} \;\; \forall\, i \tag{3.20}$$

The general plant description can then be composed of the P_i of the individual agents under the given assumptions to

$$P = \begin{array}{c|cccccccccccc|ccc}
 & d_1 & d_2 & \dots & n_1 & n_2 & \dots & r_{11} & r_{21} & r_{31} & \dots & r_{12} & \dots & u_1 & u_2 & \dots \\
z_{1|1} & W_1\ A_1\ G\ W_d & -W_1\ a_{21}\ G\ W_d & & W_1\ A_1\ W_n & 0 & & 0 & W_1\ a_{21} & W_1\ a_{31} & & 0 & & W_1\ A_1\ G & -W_1\ a_{21}\ G & \\
z_{1|2} & -W_1\ a_{12}\ G\ W_d & W_1\ A_2\ G\ W_d & & 0 & W_1\ A_2\ W_n & & 0 & 0 & 0 & & W_1\ a_{12} & & -W_1\ a_{12}\ G & W_1\ A_2\ G & \\
\vdots & & & & & & & & & & & & & & & \\
z_{1|n} & -W_1\ a_{1n}\ G\ W_d & -W_1\ a_{n1}\ G\ W_d & & 0 & 0 & & 0 & 0 & 0 & & 0 & & -W_1\ a_{1n}\ G & -W_1\ a_{2n}\ G & \\
z_{2|1} & W_2\ G\ W_d & 0 & & 0 & 0 & & 0 & 0 & 0 & & 0 & & W_2\ G & 0 & \\
z_{2|2} & 0 & W_2\ G\ W_d & & 0 & 0 & & 0 & 0 & 0 & & 0 & & 0 & W_2\ G & \\
\vdots & & & & & & & & & & & & & & & \\
z_{2|n} & 0 & 0 & & 0 & 0 & & 0 & 0 & 0 & & 0 & & 0 & 0 & \\
z_{3|1} & 0 & 0 & & 0 & 0 & & 0 & 0 & 0 & & 0 & & W_3 & 0 & \\
z_{3|2} & 0 & 0 & & 0 & 0 & & 0 & 0 & 0 & & 0 & & 0 & W_3 & \\
\vdots & & & & & & & & & & & & & & & \\
z_{3|n} & 0 & 0 & & 0 & 0 & & 0 & 0 & 0 & & 0 & & 0 & 0 & \\
y_1 & A_1\ G\ W_d & -a_{21}\ G\ W_d & & A_1\ W_n & 0 & & 0 & a_{21} & a_{31} & & 0 & & A_1\ G & -a_{21}\ G & \\
y_2 & -a_{12}\ G\ W_d & A_2\ G\ W_d & & 0 & A_2\ W_n & & 0 & 0 & 0 & & a_{12} & & -a_{12}\ G & A_2\ G & \\
\vdots & & & & & & & & & & & & & & & \\
y_n & -a_{1n}\ G\ W_d & -a_{n1}\ G\ W_d & & 0 & 0 & & 0 & 0 & 0 & & 0 & & -a_{1n}\ G & -a_{2n}\ G &
\end{array} \tag{3.21}$$

To reduce the size and increase the readability and usability of this matrix, we can combine different parts of it as matrices. Thus we use the Kronecker product $\otimes$ with its properties defined in Eq. 2.203 to 2.208 and the Hadamard product $\odot$ as defined in Eq. 2.213. Further, let $A_i = \sum_j a_{ji}$ (this can be understood as the effect of all weights $a_j i, j = 1...N$ on the output y_i of a system i before entering the controller K_i) and

$$A_d = \begin{pmatrix} A_1 & \dots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \dots & A_N \end{pmatrix} \tag{3.22}$$

$$A_a = \begin{pmatrix} 0 & a_{2,1} & \dots & a_{N-1,1} & a_{N,1} \\ a_{1,2} & 0 & & & a_{N,2} \\ \vdots & & \ddots & & \vdots \\ a_{1,N-1} & & & 0 & a_{N,N-1} \\ a_{1,N} & a_{2,N} & \dots & a_{N-1,N} & 0 \end{pmatrix} \tag{3.23}$$

and

$$L = A_d - A_a \tag{3.24}$$

For better readability we define the unit matrices

$$I_N = \begin{pmatrix} 1 & & & & \\ & 1 & & & \\ & & \ddots & & \\ & & & 1 & \\ & & & & 1 \end{pmatrix}_{N \times N} \tag{3.25}$$

and

$$I_{a \times b} = \begin{pmatrix} 1 & \dots & 1 \\ \vdots & \ddots & \vdots \\ 1 & \dots & 1 \end{pmatrix} \tag{3.26}$$

Now we can rewrite Eq. 3.21 with inputs and outputs in vector form as

$$P = \begin{array}{c|ccc|c} & \vec{d} & \vec{n} & \vec{r} & \vec{u} \\ \vec{z_1} & (1_N \otimes W_1) \cdot L \cdot (1_N \otimes (G\ W_d)) & (1_N \otimes W_n) \cdot A_d \cdot (1_N \otimes W_n) & (1_{1\times N} \otimes ((1_N \otimes W_1) \cdot A_d)) \odot (1_N \otimes 1_{S\times SN}) & (1_N \otimes W_1) \cdot L \cdot (1_N \otimes G) \\ \vec{z_2} & 1_N \otimes (W_2\ G\ W_d) & 0 & 0 & 1_N \otimes (W_2\ G) \\ \vec{z_3} & 0 & 0 & 0 & 1_N \otimes W_3 \\ \hline \vec{y} & L \cdot (1_N \otimes (G\ W_d)) & A_d \cdot (1_N \otimes W_n) & (1_{1\times N} \otimes A_d) \odot (1_N \otimes 1_{S\times SN}) & L \cdot (1_N \otimes G) \end{array} \tag{3.27}$$

with N being the number of agents or subsystems and S the number of states per subsystem (or alternatively $S \cdot N$ the total number of states of the overall system). Now we have a general plant description of a distributed multi-agent system using consensus control that can be applied to H_∞ synthesis.

It has to be noted that the matrix giving the general plant description of the overall system in Eq. 3.27 grows with the number of agents. Thus using this equation is not suitable for large scale systems with hundreds or thousands of agents, since it will not be possible to find numerical solutions for extremely large matrices. However, Eq. 3.14 which describes the generalized plant of a single agent only scales with the number of interconnected agents and thus can be used also in large scale systems as long as the number of interconnected neighbors remains small.

3.3.4 Application to Satellite Formation Flying

If we now consider spacecraft as agents, we can define the states of the spacecraft as their position and velocity in EH coordinate frame in a dynamically decoupled way[2] in analogy to Eq. 2.8:

$$\boldsymbol{x}_i = (x_i,\ \dot{x}_i,\ y_i,\ \dot{y}_i,\ z_i,\ \dot{z}_i)^T \tag{3.28}$$

So the state of each satellite contains only local information of the satellite itself. Since the states of the satellites are not coupled and the satellites are not physically interconnected[3], there is no influence of the state or the control inputs of other satellites on the satellite itself. Thus $A_{ij} = B_{ij} = C_{ij} = D_{ij} = 0 \ \forall \ i \neq j$ and we can reduce the general state-space model in Eq. 2.193 and 2.194 to

$$\begin{aligned} \dot{x}_i(t) &= A_i\ x_i(t) + B_i\ u_i(t) \\ y_i(t) &= C_i\ x_i(t) + D_i\ u_i(t) \end{aligned} \tag{3.29}$$

Further, we can assume in this example that the matrices A_i and B_i are identical for all satellites, since they all share the same dynamic model and that $D = 0$ (assuming there is no feed-through of control input). This leads to:

$$\begin{aligned} \dot{x}_i(t) &= A\ x_i(t) + B\ u_i(t) \\ y_i(t) &= C_i\ x_i(t) \end{aligned} \tag{3.30}$$

The distributed control problem now consists of finding appropriate control inputs $u_i(t) = f(\dot{x}_i(t), \dot{x}_j(t))$.

An implementation for a specific SFF scenario with weights and performed H_∞ synthesis is presented in Chap. 5.

[2] The definition of decoupled states feels natural, though it is not unique. One could also define the state in a coupled manner e.g. by taking the relative vectors between the satellites instead of the absolute position and velocity in the EH frame. The state of Satellite 1 would then consist of the relative position and velocity vectors to all other connected satellites. However, this state definition would not be minimal any more.

[3] The assumption that the satellites are not physically interconnected and thus $A_{ij} = B_{ij} = C_{ij} = D_{ij} = 0 \ \forall \ i \neq j$ is only valid as long as some weaker physical effects are neglected. Examples are microgravity, the momentum transfer on a satellite caused by the propellant of other satellites or electric or magnetic repulsion or attraction. Such effects would lead to $A_{ij} \neq 0$ or $B_{ij} \neq 0$ and are partly nonlinear.

3.4 Plant Inversion-Based Lyapunov Control Combined with a Reference Governor

This section presents a control approach in ROE that combines plant-inversion as underlying control scheme with proven Lyapunov stability and a sophisticated Reference Governor (RG). It is especially designed for CLT e.g. using electric propulsion in SFF. It is based on the joined work by Steindorf, D'Amico and Scharnagl published in Steindorf et al. (2017). It comprises parts of this publication, namely the mathematical derivation of the reduced plant matrix A, control input matrix B and feedback gain matrix P as well as Fig. 3.4 and 3.5. The two innovative features of the algorithm are the introduction of a RG that drives the state vector towards the desired value without over-passing particular constraints, and the Lyapunov based controller which increases the thrust when approaching fuel optimal locations. In addition to the joined work of Steindorf et al. (2017), an extension from two satellites to N satellites and a suiting collision avoidance strategy is developed and presented in Chap. 3.4.3.

The controller is based on states given in ROE and the reduced model presented in Chap. 2.5.5, where the control input does not have a component along the radial direction $u = (0, u_y, u_z)^T$. This is more efficient in terms of fuel and allows to control all the ROE vector elements. However, $\partial\lambda$ needs to be controlled indirectly, through a change in ∂a, as explained later. For better understanding the reduced ROE state vector, the plant matrix and the control input matrix are repeated below.

The reduced ROE vector is defined as $\partial\boldsymbol{\alpha} = (\partial a, \partial e_x, \partial e_y, \partial i_x, \partial e_y)^T$ and the overall state-space model is

$$\begin{bmatrix} \partial\dot{\boldsymbol{\alpha}} \\ \partial\ddot{a} \end{bmatrix} = \boldsymbol{A} \begin{bmatrix} \partial\boldsymbol{\alpha} \\ \partial\dot{a} \end{bmatrix} + \begin{bmatrix} \boldsymbol{B} \\ 0^{1x2} \end{bmatrix} \boldsymbol{u} \tag{3.31}$$

The reduced plant matrix is given by

$$\boldsymbol{A} = \kappa \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & \frac{1}{\kappa} \\ \frac{7}{2}e_yQ & -(4e_xe_yG + C)Q & -(1 + 4e_y^2G - D) & 5e_yS & 0 & \frac{D - e_x}{\kappa} \\ -\frac{7}{2}e_xQ & (1 + 4e_x^2G - D)Q & (4e_xe_yG - C)Q & -5e_xS & 0 & \frac{C - e_y}{\kappa} \\ 0 & 0 & 0 & 0 & 0 & 0 \\ \frac{7}{2}S & -4e_xGS & -4e_yGS & 2T & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \tag{3.32}$$

The reduced control input matrix is

$$\boldsymbol{B} = \frac{1}{a\eta_c} \begin{bmatrix} \frac{2}{\eta_c}(1+ecosf) & 0 \\ \eta_c \frac{(2+ecosf)cos(w+f)+e_x}{1+ecosf} & \frac{ne_y}{tan\, i} \frac{sin(w+f)}{1+ecosf} \\ \eta \frac{(2+ecosf)sin(w+f)+e_y}{1+ecosf} & -\frac{ne_x}{tan\, i} \frac{sin(w+f)}{1+ecosf} \\ 0 & \eta_c \frac{cos(w+f)}{1+ecosf} \\ 0 & \eta_c \frac{sin(w+f)}{1+ecosf} \\ 0 & 0 \end{bmatrix} \tag{3.33}$$

3.4.1 Control Strategy

When performing a maneuver, the aim of the controller is to lead the system to a new target state. Let us define the current state vector as $\partial\boldsymbol{\alpha}$ and the target/reference state vector as $\partial\boldsymbol{\alpha}_r$. A classic controller would just drive the system towards the target reference without considering eventual constraints. There are various constraints a controller needs to deal with. For instance, thrust constraints arise from limited fuel availability and possible thruster saturation. Time constraints are dictated by the type of mission and require the satellites to reach a certain configuration in a limited number of orbits. In the context of collision avoidance, single or multiple terms of the state vector, i. e. eccentricity vector, are desirable to be contained between a certain range, giving rise to wall constraints. The controller proposed in this work is designed to drive the system towards the target state taking in consideration all the named constraints. In order to achieve this, the control strategy is executed by two main elements: the RG and the Lyapunov Controller (LC). The basic idea is that, instead of moving directly to the target state, the system is driven towards an intermediate state, called applied reference $\partial\boldsymbol{\alpha}_a$. Every time a control loop is executed, a new $\partial\boldsymbol{\alpha}_a$ is calculated by the RG. The new value of $\partial\boldsymbol{\alpha}_a$ is given to the LC, and the control law aims to stabilize the current state to the $\partial\boldsymbol{\alpha}_a$. The LC does not know which is the final target reference. It just knows the temporary value of an intermediate target state, the applied reference.
To better visualize this concept one can consider Fig. 3.3 where a 2D representation of the problem is given. The problem can be associated with a person $\partial\boldsymbol{\alpha}$ trying to reach a specific point $\partial\boldsymbol{\alpha}_r$ on a plane. Between him and the point there are several walls (constraints) that do not allow him to execute a straight line trajectory. The person needs someone able to guide him to the target point without hitting the walls. The applied reference $\partial\boldsymbol{\alpha}_a$ can be associated with a guide for $\partial\boldsymbol{\alpha}$ and is provided by the RG. At each time interval a new applied reference is calculated by the RG and represents the next step that the current state should reach or be driven to by the controller. When $\partial\boldsymbol{\alpha}$ reaches $\partial\boldsymbol{\alpha}_a$, a new value of $\partial\boldsymbol{\alpha}_a$ is provided. As a result, $\partial\boldsymbol{\alpha}$ does not know where the target point is, it just follows its guide $\partial\boldsymbol{\alpha}_a$. At the end of the path, when $\partial\boldsymbol{\alpha}_a$ has reached $\partial\boldsymbol{\alpha}_r$, $\partial\boldsymbol{\alpha}$ reaches $\partial\boldsymbol{\alpha}_r$ as well, completing the maneuver.

The following sections give a mathematical and detailed description of this concept.

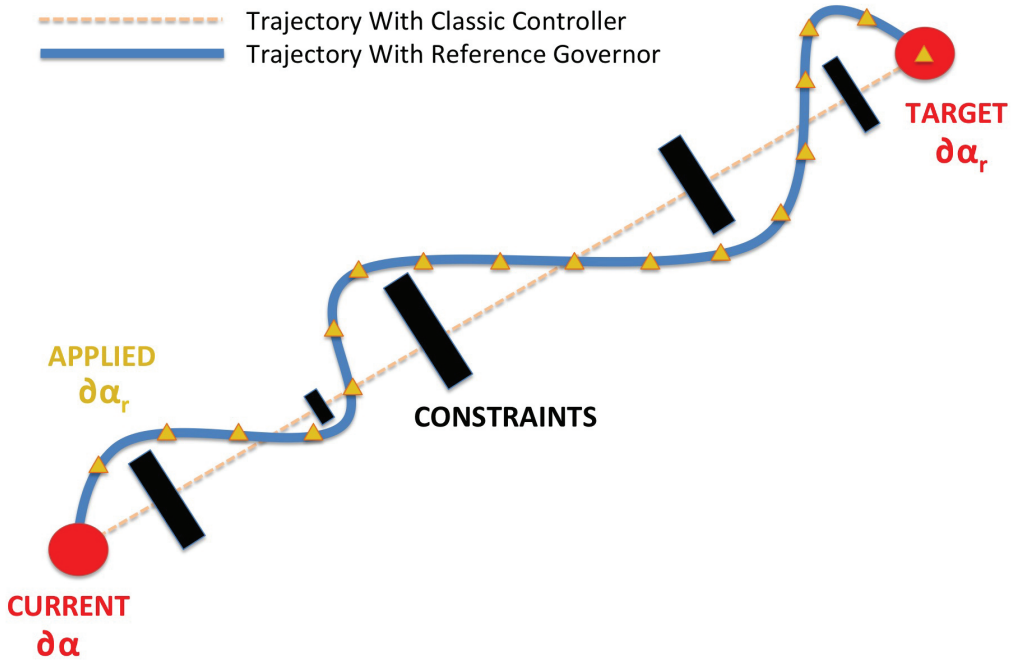


Figure 3.3: Differences between classical controller and reference governor

3.4.1.1 Lyapunov Controller

The feedback controller has to stabilize the system at an applied reference $\partial\boldsymbol{\alpha}_a$. A plant inversion-based control law, which ensures the relative spacecraft state to asymptotically tend to the applied reference, is given by

$$\boldsymbol{u} = -\boldsymbol{B}^{\star}\left[\boldsymbol{A}\partial\boldsymbol{\alpha} + \boldsymbol{P}\Delta\partial\boldsymbol{\alpha}\right] \tag{3.34}$$

with $\Delta\partial\boldsymbol{\alpha} = \partial\boldsymbol{\alpha} - \partial\boldsymbol{\alpha}_a$ and $(.)^{\star}$ denoting the pseudo inverse of a matrix. Let the system described by Eq. 2.102 be subject to the control law given in Eq. 3.34 with $\boldsymbol{P}$ being positive definite. A Lyapunov function candidate can e.g. be chosen to

$$V = \frac{1}{2}\Delta\partial\boldsymbol{\alpha}^{\mathrm{T}}\Delta\partial\boldsymbol{\alpha} \tag{3.35}$$

If the derivative of Eq. 3.35 is negative definite, one can proof that Eq. 3.35 is a Lyapunov function. The derivative is derived by

$$\dot{V} = \Delta\partial\boldsymbol{\alpha}^{\mathrm{T}}\Delta\partial\dot{\boldsymbol{\alpha}} = \Delta\partial\boldsymbol{\alpha}^{\mathrm{T}}\left(\partial\dot{\boldsymbol{\alpha}} - \partial\dot{\boldsymbol{\alpha}}_a\right) = \Delta\partial\boldsymbol{\alpha}\left(\boldsymbol{A}\partial\boldsymbol{\alpha} + \boldsymbol{B}\left(-\boldsymbol{B}^{\star}\left[\boldsymbol{A}\partial\boldsymbol{\alpha} + \boldsymbol{P}\Delta\partial\boldsymbol{\alpha}\right]\right)\right)$$

$$= -\Delta\partial\boldsymbol{\alpha}^{\mathrm{T}}\boldsymbol{P}\Delta\partial\boldsymbol{\alpha} \tag{3.36}$$

which is negative definite as long as $\boldsymbol{P}$ is positive definite. In order to improve fuel efficiency, the Lyapunov feedback gain matrix $\boldsymbol{P}$ is designed such that it determines ΔV efficient locations to apply control inputs in the following way

$$\boldsymbol{P} = \frac{1}{k}\begin{pmatrix} \cos(J)^N & 0 & 0 & 0 & 0 \\ 0 & \cos(J)^N & 0 & 0 & 0 \\ 0 & 0 & \cos(J)^N & 0 & 0 \\ 0 & 0 & 0 & \cos(H)^N & 0 \\ 0 & 0 & 0 & 0 & \cos(H)^N \end{pmatrix} \tag{3.37}$$

where $k \in \mathbb{R}^+$ is an arbitrary large scaling scalar. The matrix $\boldsymbol{P}$ regulates the intensity of the thrust vector as a function of the instantaneous value of the absolute mean argument of latitude, $\varphi = \omega + M$, of the satellite where the control law is executed. When the value of φ is close to an optimal value $\bar{\varphi}$ the intensity of the control output maximizes thanks to the cosine terms in the P matrix. The argument of the cosines is $J = \varphi - \bar{\varphi}_{ip}$ for the in-plane and $H = \varphi - \bar{\varphi}_{oop}$ for the out-of-plane case. Fig. 3.4 visualizes this concept. The exponent $N \in \mathbb{N}|N \bmod 2 = 0 \wedge N \geq 2$ defines to which extent the control inputs are centered around the fuel optimal locations to control the in-plane (∂a, $\partial\boldsymbol{e}$) motion at $\bar{\varphi}_{ip}$ and the out-of-plane ($\partial\boldsymbol{i}$) motion at $\bar{\varphi}_{oop}$. The optimal locations for near-circular orbits to apply thrust are given by Chernick and D'Amico (2016)

$$\bar{\varphi}_{ip} = \operatorname{atan2}\left(\frac{\Delta\partial e_y}{\Delta\partial e_x}\right) \tag{3.38}$$

$$\bar{\varphi}_{oop} = \operatorname{atan2}\left(\frac{\Delta\partial i_y}{\Delta\partial i_x}\right) \tag{3.39}$$

in the case of a dominant variation of the eccentricity vector ($\Delta\partial e \gg \Delta\partial a$), which is valid for bounded motions. A more complex solution for optimal maneuver locations in eccentric reference orbits is given by Chernick and D'Amico (2016). For each specific point in time there is an optimal location $\bar{\varphi}_{ip}$ and $\bar{\varphi}_{oop}$. Those optimal locations are constantly updated from the current error between the current and the applied value of the relative inclination and eccentricity vector. Fuel balancing between the spacecraft pair is achieved by keeping the spacecraft attitudes constant, with their thrusters aligned in positive flight and normal orbit direction. If the control law given in Eq. 3.34 yields a positive control input it is executed by the follower spacecraft, whereas a negative control input is executed by the leader spacecraft.

Of course, this approach is only valid for two satellites. An extension to SFF of N satellites is discussed in Chap. 3.4.3.

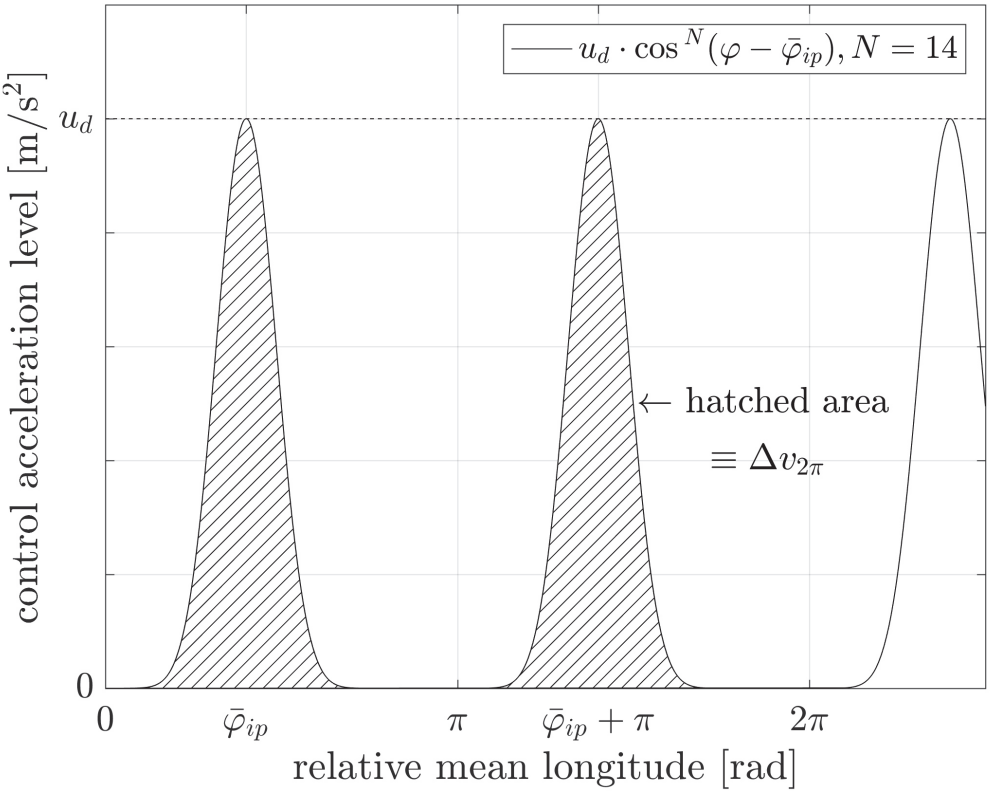


Figure 3.4: Control inputs of the form $u_d \cdot \cos^N(\varphi - \bar{\varphi}_{ip})$ as shaped by the feedback gain matrix $\boldsymbol{P}$. Here shown for $N = 14$ and a random fuel optimal relative mean latitude $\bar{\varphi}_{ip}$. The hatched area represents the ΔV that is applied during one complete orbit period.

3.4.1.2 Controlling the Relative Mean Longitude

The relative mean longitude $\partial\lambda$ can only be controlled with thrust into the radial direction of the EH frame. Waiving radial thrust would mean the loss of full controllability. In this case, it is suitable to use a reduced model and control $\partial\dot{\lambda}$ by leveraging Keplerian dynamics. In fact, if there is no radial thrust applied, $\partial\lambda$ can be indirectly controlled by leveraging natural (Keplerian) orbit dynamics as suggested by

$$\partial\dot{\lambda} = -\frac{3}{2} n \, \partial a \tag{3.40}$$

The relative semi-major axis ∂a can be changed by applying tangential thrust (see Eq. 3.33). When a difference in relative semi-major axis is established, it will cause a drift in the Along-Track direction. Hence, the increase or decrease of $\partial\lambda$ is a function

of time and not a matter of thrust any longer. When the relative mean longitude tracking error $\Delta\partial\lambda$ approaches zero, the velocity of $\partial\lambda$, $\partial\dot{\lambda}_a$, should also be zero. This is obtained by using the function in Fig. 3.5 which outputs the desired value of $\partial\dot{\lambda}_a$ when a given value of tracking error $\Delta\partial\lambda$ rises.

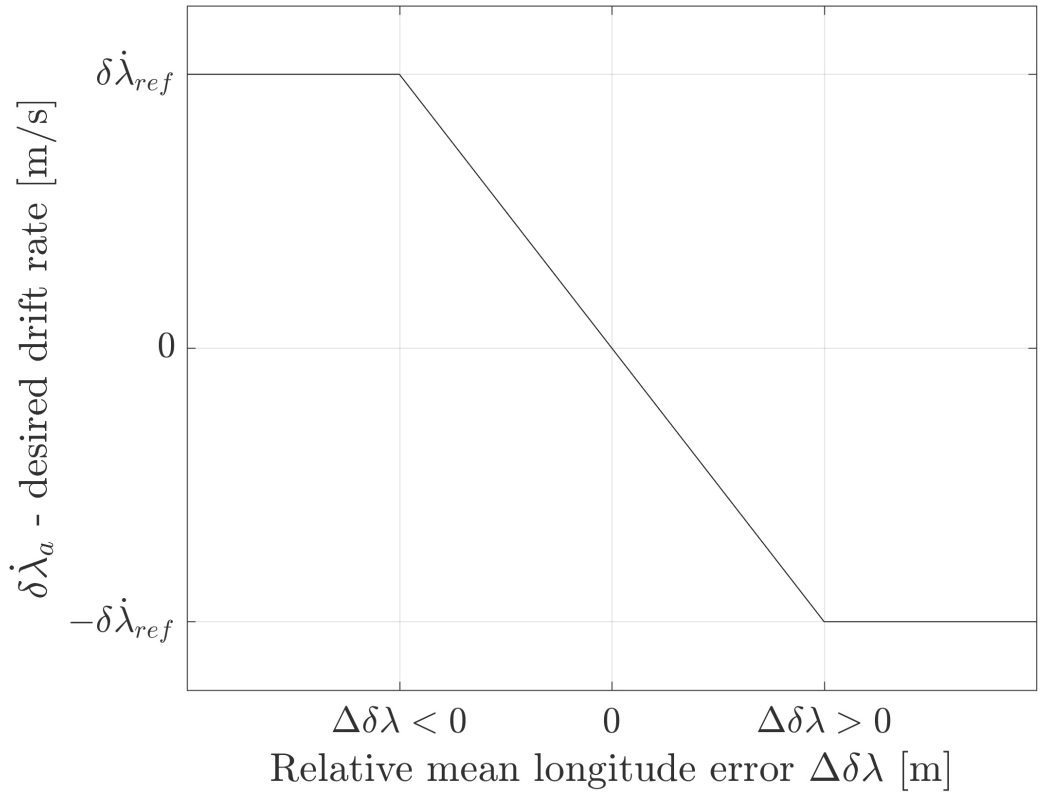


Figure 3.5: Desired drift rate of the relative mean longitude $\partial\lambda$.

The function in Fig. 3.5 is centered at the desired reference $\partial\lambda_a$ and it is given by

$$\partial\dot{\lambda}_a = \begin{cases} -min\left\{\left|\frac{\Delta\partial\lambda}{\tau}\right|, \partial\dot{\lambda}_{ref}\right\}, & if\ \Delta\partial\lambda \geq 0 \\ min\left\{\left|\frac{\Delta\partial\lambda}{\tau}\right|, \partial\dot{\lambda}_{ref}\right\}, & if\ \Delta\partial\lambda \leq 0 \end{cases} \tag{3.41}$$

where $\tau \in \mathbb{R}^+$ is an arbitrary large scaling factor, which sets the desired drift rate for a given error. In Eq. 3.41, the first case refers to a positive error ($\Delta\partial\lambda \geq 0$) where a negative drift is required to minimize it. When a negative error ($\Delta\partial\lambda \leq 0$) is present, a positive drift rate reduces it. The drift rate is limited by an upper bound $\partial\dot{\lambda}_{ref}$ and a lower bound $-\partial\dot{\lambda}_{ref}$. The drift rate $\partial\dot{\lambda}_{ref}$ for upper bound is given by

$$\partial\dot{\lambda}_{ref} = \frac{3}{2} n \left|\partial a_{ref}\right| \tag{3.42}$$

In order to derive ∂a_{ref}, it is necessary to calculate the effect of tangential thrust on ∂a. The effect of a continuous tangential burn on the relative semi-major axis is

given by the control input matrix $\boldsymbol{B}_c$ in Eq. 2.101 by

$$\Delta\partial a_{tan} = \frac{2}{an\eta}(1 + ecos(f))\Delta v_{tan} \tag{3.43}$$

where the velocity change Δv_{tan} is given by

$$\Delta v_{tan} = \frac{u_d.T}{4} \prod_{q=N,N-2,\ldots}^{4} \frac{q-1}{1} \tag{3.44}$$

with the desired acceleration u_d, orbit period T and a scaling exponent N as specified in the next section during design of the feedback gain matrix. The desired relative semi-major axis ∂a_{ref} is set to half of $\Delta\partial a_{tan}$

$$|\partial a_{ref}| = \frac{|\Delta\partial a_{tan}|}{2} \tag{3.45}$$

Once $\partial\dot{\lambda}_a$ is determined by Eq. 3.41, the desired semi-major axis has to be calculated applying Eq. 3.40.

$$\partial a_a = -\frac{2}{3}\frac{\partial\dot{\lambda}_a}{n_c} \tag{3.46}$$

When the desired relative semi-major axis ∂a_a is applied to the controller as tracking input, the mean longitude $\partial\lambda$ is indirectly controlled as well.

3.4.1.3 Tracking the Relative Semi-Major Axis

The in-plane motion of the relative geometry is defined by the relative eccentricity vector $\partial\boldsymbol{e}$ and the relative semi-major axis ∂a. The control input matrix, given by the GVE, shows that $\partial\boldsymbol{e}$ and ∂a are coupled such that control efforts to stabilize either $\partial\boldsymbol{e}$ or ∂a could possibly destabilize the other one. The required direction of a tangential burn for controlling a given error of the relative eccentricity vector changes by 180° every half an orbit. The required tangential burn for a given relative semi-major axis error does not change its direction. This conflict of interest is illustrated in Fig. 3.6.

Taking into account the before-mentioned facts, it is not possible to directly control $\partial\lambda$. Consequently, in order to change $\partial\lambda$ to a desired value, it is necessary to infer a $\partial\dot{\lambda}$ through a change in ∂a. However this ∂a cannot be too far from the initial value because, after $\partial\lambda$ has reached the desired value, ∂a has to be brought back to its initial value. Assuming the fact that desired changes in the relative semi-major axis are much smaller than in the relative eccentricity vector, which is true for bounded formations, this problem can be solved by defining a limit for the tracking error of the relative semi-major axis to be

$$|\Delta\partial a^{\max}| = \frac{|\Delta\partial a_{\tan}|}{2} \tag{3.47}$$

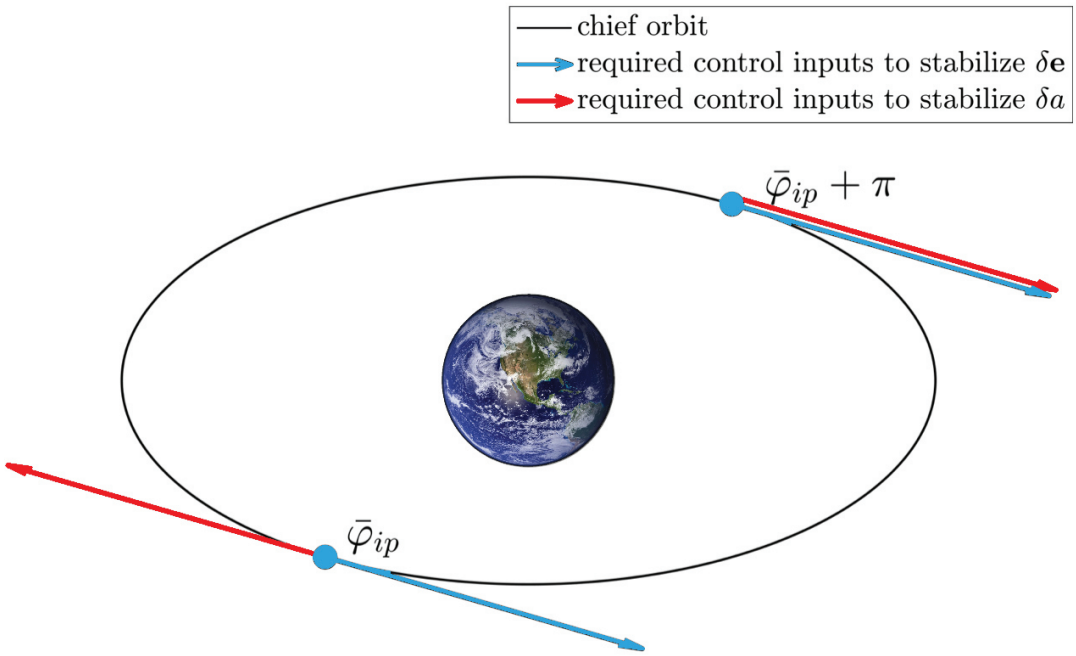


Figure 3.6: Absolute leader orbit with fuel optimal locations to apply tangential thrust for in-plane reconfigurations, $\varphi = \bar{\varphi}_{ip}$ and $\varphi = \bar{\varphi}_{ip} + \pi$ (based on Steindorf et al. (2017)).

which is half the change of the relative semi-major axis due to one tangential burn. Therefore, the error of ∂a will always be $\Delta\partial a = \pm\frac{\Delta\partial a_{\tan}}{2}$ at the beginning of a tangential burn and $\Delta\partial a = \mp\frac{|\Delta\partial a_{\tan}|}{2}$ at the end of a tangential burn. Thus, an oscillation of ∂a around the desired reference ∂a_a is yielded. The maximum tracking error $|\Delta\partial a^{\max}|$ can be enforced by defining the feedback gain matrix $\boldsymbol{P}$ given in Eq. 3.37 such that the error feedback is set to zero, if the maximum error $|\Delta\partial a^{\max}|$ (or following Eq. 3.46 $|\partial\dot{\lambda}_a^{\max}|$) is violated. Thus $\boldsymbol{P}$ is reformulated as

$$\boldsymbol{P}_{|\Delta\partial a|\geq|\Delta\partial a^{\max}|} = \frac{1}{k}\begin{pmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \cos(H)^N & 0 \\ 0 & 0 & 0 & 0 & \cos(H)^N \end{pmatrix} . \tag{3.48}$$

3.4.2 Reference Governor

This section describes the guidance of the applied reference $\partial\boldsymbol{\alpha}_a$, at which the Lyapunov controller stabilizes the relative state, in order to satisfy the constraints

imposed on the reconfiguration and to satisfy the complete set of target conditions. A Reference Governor (RG) is developed that will guide the applied reference $\partial\boldsymbol{\alpha}_a$ through the ROE state-space while ensuring the satisfaction of the constraint set. The RG uses an attractive potential field that is directed to the desired reference $\partial\boldsymbol{\alpha}_r$ and the constraints are realized by repulsive potential functions. Together, both the repulsive and the attractive fields compose a potential field map ϕ with gradient $\boldsymbol{\rho}$ at the current applied reference $\partial\boldsymbol{\alpha}_a$. Thus, the ROE state is guided along this gradient to the desired reference without violating any constraints. Following the potential field gradient $\boldsymbol{\rho}$ ensures that the steady state error is minimum without violating the constraint set. The rate of change $|\partial\dot{\boldsymbol{\alpha}}_a|$ at which the applied reference is guided, is controlled by formulating a Lyapunov threshold Γ for the Lyapunov function given by Eq. 3.35.

The RG receives the 5D state vectors of the reduced model $\partial\boldsymbol{\alpha}$ and the reference $\partial\boldsymbol{\alpha_r}$ as input, and computes the applied reference $\partial\boldsymbol{\alpha}_a$ in each control loop execution. The value of $\partial\boldsymbol{\alpha}_a$ results from integration of Eq. 3.49 (Nicotra et al., 2016).

$$\partial\dot{\boldsymbol{\alpha}}_a = \xi \left[\Gamma - V\right] \cdot \boldsymbol{\rho} \tag{3.49}$$

where $\xi \in \mathbb{R}^+$ is an arbitrary large scalar. Eq. 3.49 is the central equation of the RG.

Eq. 3.49 calculates the vector rate of change of $\partial\boldsymbol{\alpha}_a$. The norm of this vector is given by the scalar $\xi(\Gamma - V)$ while the direction is given by the vector $\boldsymbol{\rho}$. The Lyapunov threshold Γ determines the maximum allowed tracking error since the Lyapunov function $V = \frac{1}{2}\Delta\partial\boldsymbol{\alpha}^{\mathrm{T}}\Delta\partial\boldsymbol{\alpha}$ is a function of the tracking error itself. The Lyapunov function has an upper limit defined by Γ. Each of the constraints imposed on the system, such as maximum thrust and minimum maneuver time, impose a constraint through a specific value Γ_i. The most restrictive Lyapunov threshold

$$\Gamma = \min\left\{\Gamma_i\right\} \tag{3.50}$$

is to be enforced by the RG (Nicotra et al., 2015). Fig. 3.7 visualizes this behavior.

Note that the derived Lyapunov thresholds Γ_i are suited for the specific Lyapunov function in use and may change, as soon as a different Lyapunov function is used. All Γ_i are derived in the following sections. If the Lyapunov value V is smaller than its threshold Γ, all constraints are satisfied. Thus all constraints are translated into a single constraint by

$$V \leq \Gamma \tag{3.51}$$

If the Lyapunov value V gets close to the Lyapunov threshold, the change of the applied reference tends to zero, because of $\lim_{V\to\Gamma}(\Gamma - V) = 0$, giving the system time to converge to the applied reference. When the tracking error $\Delta\partial\boldsymbol{\alpha}$ is large, the RG slows down the change of $\partial\boldsymbol{\alpha}_a$, giving the state time to reach the applied position. When $\partial\boldsymbol{\alpha}$ has almost reached the applied state, the RG changes the applied state rapidly, and so on until the final target position is reached. If the ROE state

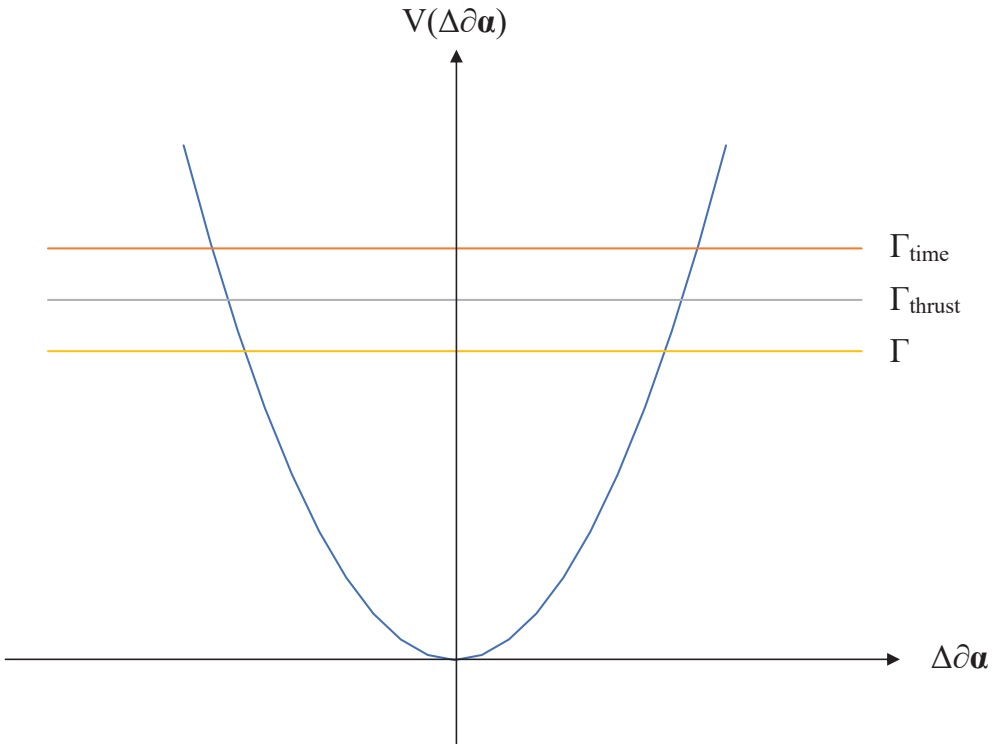


Figure 3.7: Lyapunov function V (blue) together with different Lyapunov thresholds Γ_i and the overall Lyapunov threshold Γ

is exactly at the applied reference $V = 0$ the change of the applied reference is maximized such that $\partial\boldsymbol{\alpha}_a$ reaches $\partial\boldsymbol{\alpha}_\mathrm{r}$ as quickly as possible. The gradient of ϕ is given by Nicotra et al. (2015)

$$\boldsymbol{\rho} = \boldsymbol{\nabla}\phi \tag{3.52}$$

The Nabla operator $\boldsymbol{\nabla}(.)$ is defined, in such way that it takes the partial derivatives with respect to the applied reference $\partial\boldsymbol{\alpha}_a$, as

$$\boldsymbol{\nabla} = \left(\frac{\partial}{\partial\partial a}, \frac{\partial}{\partial\partial e_x}, \frac{\partial}{\partial\partial e_y}, \frac{\partial}{\partial\partial i_x}, \frac{\partial}{\partial\partial i_y}\right)_a$$

The gradient is a conservative function that tends to zero at $\partial\boldsymbol{\alpha}_a = \partial\boldsymbol{\alpha}_r$ and goes to infinity at the constraint boundaries. Since the sum of conservative functions is also conservative. The term $\boldsymbol{\rho}$ gives the direction of the change of $\partial\boldsymbol{\alpha}_a$. By taking into account separate terms for each defined constraint, $\boldsymbol{\rho}$ can be constructed (Nicotra et al., 2015). After all potential field gradients are found, the vector field $\boldsymbol{\rho}$ is given by

$$\boldsymbol{\rho} = -\boldsymbol{\nabla}\bar{\phi} - \sum_{i=1}^{M} \boldsymbol{\nabla}\phi_i \tag{3.53}$$

where $\bar{\phi}$ is the global attractive potential field and ϕ_i are the repulsive potential fields of all constraints. M is the number of defined constraints.

The choice of which direction to give to the applied reference is dictated by the potential field gradient. The applied reference is guided along the negative values of $\boldsymbol{\rho}$. As it appears in Eq. 3.53, the overall potential field $\boldsymbol{\rho}$ is the gradient of the sum of the background term $\bar{\phi}$, pushing the applied reference towards the target, and several local potentials representing the constraints. The global potential gradient is defined as a positive vector. With the minus, it becomes negative (attraction). Local potentials are designed in such a way that, when the applied state gets close to a constraint, the local potential gradient relative to that specific constraints increases its value, becomes greater than the $\bar{\phi}$ field and makes $\boldsymbol{\rho}$ positive. As a consequence, when $\partial\boldsymbol{\alpha}$ enters the area of influence of a specific constraint, the overall field $\boldsymbol{\rho}$ is positive. Since $\partial\boldsymbol{\alpha}$ naturally moves towards negative gradients, it will be pushed away from the constraint, leaving the constraint influence area.

3.4.2.1 Potential Field Map

The global attractive potential function is given by (Nicotra et al., 2015)

$$\bar{\phi} = \begin{cases} \|\partial\boldsymbol{\alpha}_a - \partial\boldsymbol{\alpha}_r\|, & \text{if } \|\partial\boldsymbol{\alpha}_a - \partial\boldsymbol{\alpha}_r\| \geq \eta \\ \frac{1}{2}\frac{\|\partial\boldsymbol{\alpha}_a - \partial\boldsymbol{\alpha}_r\|^2}{\eta} + \frac{1}{2}\eta, & \text{otherwise} \end{cases} \tag{3.54}$$

where η is an arbitrary parameter defined below with $\eta \geq 1$ and $\eta \in \mathbb{R}$. The gradient of the global potential field is given by

$$\boldsymbol{\nabla}\bar{\phi} = \frac{\partial\boldsymbol{\alpha}_a - \partial\boldsymbol{\alpha}_r}{\max\{\|\partial\boldsymbol{\alpha}_a - \partial\boldsymbol{\alpha}_r\|, \eta\}} \tag{3.55}$$

which is unitary if $\|\partial\boldsymbol{\alpha}_a - \partial\boldsymbol{\alpha}_r\| \geq \eta$ and tends to zero for $\|\partial\boldsymbol{\alpha}_a - \partial\boldsymbol{\alpha}_r\| < \eta$. The parameter η is defined such that the gradient tends to zero for very small tracking errors with respect to the desired reference $\partial\boldsymbol{\alpha}_r$. Otherwise the applied reference $\partial\boldsymbol{\alpha}_a$ would not converge to the desired reference. From this condition, the behavior of the applied reference, when it almost reaches the target, is asymptotic in the sense that $\partial\boldsymbol{\alpha}_a$ does approach $\partial\boldsymbol{\alpha}_r$ with a lower velocity to avoid oscillations around the target state.

3.4.2.2 Constraints

Local potentials enter Eq. 3.53 in the summation term. The gradients are calculated separately and combined in the summation term. Local potentials are defined to limit the value of components of the state vector and assure collision avoidance. Each of the states can be limited individually, imposing a wall constraint on the single element, or coupled with another element with the definition of a circular no-entry zone. Both those methods are applied only to ∂e_x, ∂e_y, ∂i_x,∂i_y. ∂a and $\partial\lambda$

are treated separately because of the passive control of $\partial\lambda$ through $\partial\dot{a}$ introduced previously.

Time Constraint

Mission requirements usually impose time constraints on reconfigurations. The proposed control strategy allows to calculate the required thrust level in order to achieve a desired reconfiguration with a given optimal ΔV change. The feedback gain matrix in Eq. 3.37 defines the shape of the applied control inputs (see Fig. 3.4). The control inputs are characterized by a cosine shape centered at the fuel-optimal location with the even and positive exponent N. The ΔV change $\Delta v_{2\pi}$ for one orbit period of such continuous control accelerations is given by

$$\Delta v_{2\pi} = u_d \int_{\varphi=0}^{\varphi=2\pi} \cos(\varphi)^N \mathrm{d}\varphi \tag{3.56}$$

with u_d being the desired control acceleration and φ the mean argument of latitude. This integral can be solved using the reduction formula, which is given by

$$\int \cos(\varphi)^N \mathrm{d}\varphi = \frac{\sin(\varphi)\cos(\varphi)^{N-1}}{N} + \frac{N-1}{N}\int \cos(\varphi)^{N-2}\mathrm{d}\varphi \tag{3.57}$$

Note that N is defined to be an even number. Therefore, the term $\frac{1}{N}\sin(\varphi)\cos(\varphi)^{N-1}$ with an odd exponent for the cosine can always be expressed in terms of $\sin(d\cdot\varphi)$ with d being an even number. An example for $N=4$ is

$$\begin{aligned}
\frac{1}{4}\sin(\varphi)\cos(\varphi)^3 &= \frac{1}{4}\left[\sin(\varphi)\cos(\varphi)\cdot\cos(\varphi)^2\right] \\
&= \frac{1}{4}\left[\frac{1}{2}\sin(2\varphi)\cdot\left(\frac{1}{2}+\frac{1}{2}\cos(2\varphi)\right)\right] \\
&= \frac{1}{16}\sin(2\varphi) + \frac{1}{16}\sin(2\varphi)\cos(2\varphi) \\
&= \frac{1}{16}\sin(2\varphi) + \frac{1}{32}\sin(4\varphi)
\end{aligned} \tag{3.58}$$

For the integration from 0 to 2π Eq. 3.58 becomes zero. Since $\sin(\varphi)\cos(\varphi)^{N-1}$ has only odd exponents as in the previous example, the first summand of Eq. 3.57 is always zero and can be neglected. Thus, Eq. 3.57 can be rewritten as

$$\int_0^{2\pi} \cos(\varphi)^N \mathrm{d}\varphi = \frac{N-1}{N}\int_0^{2\pi} \cos(\varphi)^{N-2}\mathrm{d}\varphi \tag{3.59}$$

One can easily see that this reduction formula can be applied sequentially until the exponent $N-2$ is reduced to $N-2=2$, which is given for $N=4$. This product

sequence is given by

$$\Delta v_{2\pi} = u_d \prod_{q=N,N-2,\cdots}^{4} \left[\frac{q-1}{q}\right] \cdot \int_0^{2\pi} \cos(\varphi)^2 \mathrm{d}\varphi \qquad \text{which can be rewritten as}$$

$$\Delta v_{2\pi} = u_d \prod_{q=N,N-2,\cdots}^{4} \left[\frac{q-1}{q}\right] \cdot \int_0^{2\pi} \frac{1}{2} + \frac{1}{2}\cos(2\varphi)\mathrm{d}\varphi \qquad \text{substitute } x = 2\varphi \text{ and } \frac{\mathrm{d}x}{\mathrm{d}\varphi} = 2$$

$$\Delta v_{2\pi} = u_d \prod_{q=N,N-2,\cdots}^{4} \left[\frac{q-1}{q}\right] \cdot \left(\frac{1}{2}\varphi\Big|_0^{2\pi} + \frac{1}{2}\int_0^{2\pi} \cos(x)\frac{\mathrm{d}x}{2}\right) \qquad \text{adjust integration boundaries}$$

$$\Delta v_{2\pi} = u_d \prod_{q=N,N-2,\cdots}^{4} \left[\frac{q-1}{q}\right] \cdot \left(\pi + \frac{1}{4}\sin(x)\Big|_0^{4\pi}\right) \qquad \text{resubstitute } x = 2\varphi$$

$$\Delta v_{2\pi} = u_d \prod_{q=N,N-2,\cdots}^{4} \left[\frac{q-1}{q}\right] \cdot \left(\pi + \frac{1}{4}\sin(2\varphi)\Big|_0^{2\pi}\right) \qquad \text{simplify}$$

$$\Delta v_{2\pi} = u_d \pi \cdot \prod_{q=N,N-2,\cdots}^{4} \frac{q-1}{q} \quad \left[\left(\frac{\mathrm{m}}{\mathrm{s}^2} \cdot \mathrm{rad}\right) \Big/ \mathrm{orbit}\right] \qquad \text{multiply with } \frac{T}{2\pi}$$

$$\Delta v_{2\pi} = \frac{1}{2} u_d T \cdot \prod_{q=N,N-2,\cdots}^{4} \frac{q-1}{q} \quad \left[\left(\frac{\mathrm{m}}{\mathrm{s}}\right) \Big/ \mathrm{orbit}\right] \tag{3.60}$$

where T is the leader's orbit period. The required optimal total in-plane $\Delta v_{ip}^{\mathrm{opt}}$ and out-of-plane Δv_{oop}^{opt} velocity changes are given by

$$\Delta v_{ip}^{\mathrm{opt}} = a_c n_c \cdot \frac{\|\Delta\partial \boldsymbol{e}\|}{2\eta_c} \quad \text{see Chernick and D'Amico (2016)} \tag{3.61}$$

$$\Delta v_{oop}^{\mathrm{opt}} = a_c n_c \frac{1-e_c}{\eta_c} \|\Delta\partial \boldsymbol{i}\| \tag{3.62}$$

Eq. 3.61 requires that $\frac{\Delta\partial a}{2(1+e_c)}$ is smaller than $\frac{\|\Delta\partial \boldsymbol{e}\|}{2\eta^2}$ to be valid. For reconfigurations from one bounded formation to another bounded formation this restriction is always satisfied. The applied velocity increment $\Delta v_{2\pi}$ has to be equal to the velocity change that is required during one orbit period $\frac{\Delta v^{\mathrm{opt}}}{\#\ \mathrm{orbits}}$

$$\frac{\Delta v^{\mathrm{opt}}}{\#\ \mathrm{orbits}} = \frac{1}{2} u_d T \cdot \left(\prod_{q=N,N-2,\cdots}^{4} \frac{q-1}{q}\right) \tag{3.63}$$

with the only unknown being u_d. Solving for the desired thrust levels into the orbit normal $u_{d,oop}$ and the orbit tangential direction $u_{d,ip}$ yields

$$u_{d,ip} = \frac{2\Delta v_{ip}^{\text{opt}}}{T \cdot \#\text{orbits}} \prod_{q=N,N-2,\cdots}^{4} \frac{q}{q-1} \quad \left[\frac{\text{m}}{\text{s}^2}\right] \tag{3.64}$$

$$u_{d,oop} = \frac{2\Delta v_{oop}^{\text{opt}}}{T \cdot \#\text{orbits}} \prod_{q=N,N-2,\cdots}^{4} \frac{q}{q-1} \quad \left[\frac{\text{m}}{\text{s}^2}\right] \tag{3.65}$$

Of course, it has to be ensured that the thrusters also need to be capable of producing the desired control acceleration u_d. Furthermore, since the along-track separation is only indirectly controlled by leveraging Keplerian dynamics the maximum error in the along-track direction $|\Delta\partial\lambda|$ has to be smaller than

$$|\Delta\partial\lambda| \leq \frac{3}{2} n |\partial a_a| \cdot \#\text{orbits} \cdot T \tag{3.66}$$

Thrust Constraint

For low-thrust propulsion systems it is important that the thrusters are not continuously operated in saturation. Therefore a thrust constraint is formulated that ensures that a user defined thrust level is not exceeded by the control law. Note that the time constraint and the thrust constraint are mutually exclusive. The thrust constraint is given by

$$\|\boldsymbol{F}\| \leq F_{\text{desired}} \tag{3.67}$$

where F_{desired} has to be within the thruster levels, but larger than the relative dynamic perturbations such that a reconfiguration or formation maintenance is still possible.

Wall Constraint

Next, wall constraints shall be discussed. To be able to express them in one specific term, a vector $\boldsymbol{c}$ is defined. A specific vector $\boldsymbol{c}_i$ defines the ROE vector element i to be constrained by setting all elements to zero, except of one element which corresponds to the constrained element which is set to 1. An example for constraining ∂i_x is given by

$$\boldsymbol{c}_{\boldsymbol{i_x}} = \begin{pmatrix} 0 & 0 & 0 & 1 & 0 \end{pmatrix}^{\text{T}} \tag{3.68}$$

such that $\boldsymbol{c}_{\boldsymbol{i_x}}^{\text{T}} \partial\boldsymbol{\alpha} = \partial i_x$. To keep the description generic, $\boldsymbol{c_i}$ is the extraction vector for the i-th element of $\partial\boldsymbol{\alpha}$. The wall constraint defines an upper/lower limit for a specific element of $\partial\boldsymbol{\alpha}$.
Figure 3.8 represents the case where one $\partial\boldsymbol{\alpha}$ component has a lower bound. Let C_i be the current distance of the element state to its limit d_i

$$C_i = \left| \boldsymbol{c}_i^T \partial\boldsymbol{\alpha}_a - d_i \right| \tag{3.69}$$

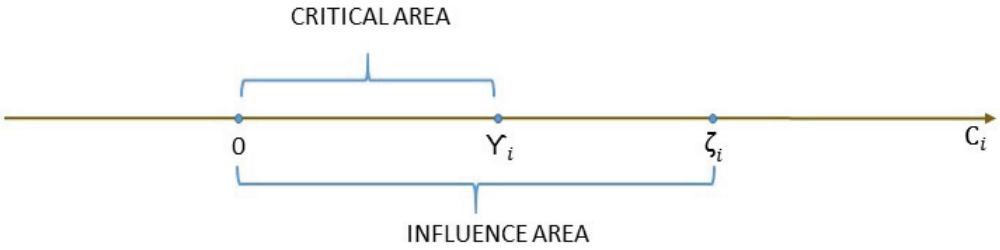


Figure 3.8: Graphical representation of the wall constraint

Let Υ_i bound the critical zone where the constraint potential is very high and ζ_i represent the distance below which the applied state is in the constraint influence area. When C_i is greater than the influence area limit ζ_i, the wall local potential is zero. Conversely, when $C_i < \zeta_i$, the local gradient is calculated by Eq. 3.70. The first case of Eq. 3.70 refers to the upper boundary case, the second case to the lower boundary for the selected element.

$$\boldsymbol{\nabla}\phi_i = \begin{cases} \frac{\Upsilon i^2(\zeta_i^2 - C_i^2)}{(\zeta_i^2 - \Upsilon_i^2)C_i^2}\boldsymbol{c_i} : \boldsymbol{c_i}\partial\alpha \leqslant d_i \\ \frac{-\Upsilon_i^2(\zeta_i^2 - C_i^2)}{(\zeta_i^2 - \Upsilon_i^2)C_i^2}\boldsymbol{c_i} : \boldsymbol{c_i}\partial\alpha \geqslant d_i \end{cases} \tag{3.70}$$

Circular No-Entry Zone Constraint

Applied vector elements can be constrained using the circular no-entry zone as well. Here, two elements instead of one are extracted and a condition is imposed on both of them at the same time. This becomes meaningful when dealing with the inclination and eccentricity vector. Taking the relative eccentricity vector as an example, the circular constraint allows the vector to not assume coordinates inside a specific circle on the ∂e_x, ∂e_y plane, as shown in Fig. 3.9.

A circular boundary of radius $r_i \in \mathbb{R}$, centered at $\boldsymbol{x}_i \in \mathbb{R}^2$ for any combination of two ROE can be formulated using

$$\|\boldsymbol{G}_i^{\mathrm{T}}\partial\boldsymbol{\alpha} - \boldsymbol{x}_i\| \geq r_i \tag{3.71}$$

with $\boldsymbol{G}_i \in \mathbb{N}^{5x2}$ and $r_i > 0$. The parameter $\boldsymbol{G}_i$ defines the ROE vector elements to be constrained, with all elements being zero but two elements which correspond to the constrained elements being 1. For example, in order to constrain the relative eccentricity vector $\partial\boldsymbol{e}$ one has to define

$$\boldsymbol{G} = \begin{pmatrix} 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \end{pmatrix}^{\mathrm{T}} \tag{3.72}$$

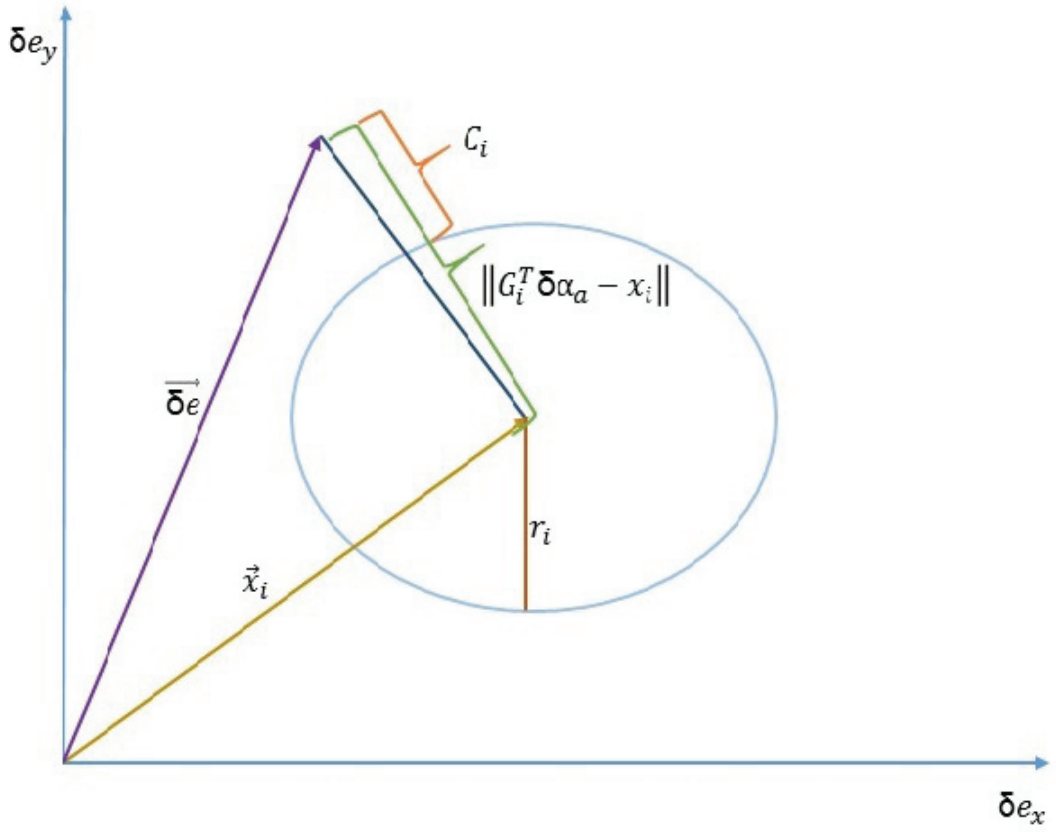


Figure 3.9: Graphical representation of the circular no-entry zone constraint

such that $\boldsymbol{G}^{\mathrm{T}}\partial\boldsymbol{\alpha} = \partial\boldsymbol{e}$. Let C_i again be the current distance of the element state to its limit r_i

$$C_i = \left\|\boldsymbol{G}_i^T\partial\boldsymbol{\alpha}_a - \boldsymbol{x}_i\right\| - r_i \tag{3.73}$$

The local potential field gradient relative to the circular no-entry zone is given by

$$\boldsymbol{\nabla}\phi_i = \frac{-\tau_i^2(\zeta_i^2 - C_i^2)}{(\zeta_i^2 - \tau_i^2)C_i^2}\frac{\boldsymbol{G_i}(\boldsymbol{G}_i^T\partial\boldsymbol{\alpha} - \boldsymbol{x}_i)}{\|\boldsymbol{G}_i\partial\boldsymbol{\alpha} - \boldsymbol{x}_i)\|} \tag{3.74}$$

Passive Collision Avoidance

A further local potential field gradient can be added in Eq. 3.53 to guarantee collision avoidance. The main idea of the presented concept is to increase the Along-Track separation between leader and follower, when the distance between the two satellites[4] projected on the $\hat{\boldsymbol{e}}_y$-$\hat{\boldsymbol{e}}_z$ plane gets smaller than a certain value. When the $\hat{\boldsymbol{e}}_y$-$\hat{\boldsymbol{e}}_z$ safety separation is re-established, the Along-Track separation is again reduced to the desired value. The principle is the same as the other potentials. A distance term C_{yz} is defined (Eq. 3.75). If this distance is less than a safety value ζ_{yz} and the

[4]An extension to N satellites is presented in Chap. 3.4.3.

along track separation $\partial\lambda$ is also smaller than a previously defined value, the local potential gets active (Eq. 3.76).

$$C_{yz} = min\left\{\|\partial \boldsymbol{e}\| - \varepsilon, \|\partial \boldsymbol{i}\| - \varepsilon, \|\partial \boldsymbol{e}.\partial \boldsymbol{i}\| - \varepsilon\sqrt{\partial e^2 + \partial i^2 - \varepsilon^2}\right\} \tag{3.75}$$

$$\phi_{yz} = \begin{cases} \frac{-\tau_{yz}^2(\zeta_{yz}^2 - C_{yz}^2)}{(\zeta_{yz}^2 - \tau_{yz}^2)C_{yz}^2} & \text{if } C_{yz} \leqslant \zeta_{yz} \text{ and } |\partial\lambda| \leq 2 \cdot \|\partial \boldsymbol{e} + \varepsilon\| \\ 0 & \text{otherwise} \end{cases} \tag{3.76}$$

$$\boldsymbol{\nabla}\phi_{yz} = \phi_{yz} \cdot (-\boldsymbol{\nabla}\bar{\phi}) \tag{3.77}$$

From Eq. 3.77, when the minimum distance between the satellites in the $\hat{\boldsymbol{e}}_{\boldsymbol{y}}$-$\hat{\boldsymbol{e}}_{\boldsymbol{z}}$ plane coincides with the minimum allowed distance ε, the collision avoidance local potential gradient has a unitary value. This means that in this specific case $\boldsymbol{\rho} = 0$ and, according to Eq. 3.49, $\partial\boldsymbol{\alpha}$ does not change anymore. In this condition, while collision avoidance is being re-established, the along-track separation is increased to avoid actual collision, according to

$$\partial\lambda_r = \partial\lambda_{safe} \tag{3.78}$$

$$\partial\lambda_{safe} = \begin{cases} 2 \cdot \|\partial \boldsymbol{e}\| + \varepsilon, & \text{if } C_{yz} \leqslant \zeta_{yz} \text{ and } |\partial\lambda| \leq 2 \cdot \|\partial \boldsymbol{e} + \varepsilon\| \text{ and } \partial\lambda > 0 \\ -2 \cdot \|\partial \boldsymbol{e}\| - \varepsilon, & \text{if } C_{yz} \leqslant \zeta_{yz} \text{ and } |\partial\lambda| \leq 2 \cdot \|\partial \boldsymbol{e} + \varepsilon\| \text{ and } \partial\lambda \geqslant 0 \end{cases} \tag{3.79}$$

The RG decides, if the state is in the collision avoidance influence region based on the value of C_{yz} and its relation to the ζ_{yz} constant. The expression of C_{yz} calculated through Eq. 3.75 comes from

$$\|\partial \boldsymbol{e}\| \geqslant \varepsilon \tag{3.80}$$

$$\|\partial \boldsymbol{i}\| \geqslant \varepsilon \tag{3.81}$$

$$\|\partial \boldsymbol{e} \cdot \partial \boldsymbol{i}\| \geqslant \varepsilon\sqrt{\partial e^2 + \partial i^2 - \varepsilon^2} \tag{3.82}$$

Those equations impose three conditions on the magnitudes of the relative inclination and eccentricity vector and on their scalar product. Those three conditions are the expression of a single more complex equation given by

$$\partial r_{yz}^{min} = \frac{\sqrt{2}\,\|\partial \boldsymbol{e} \cdot \partial \boldsymbol{i}\|}{(\partial e^2 + \partial i^2 + \|\partial \boldsymbol{e} + \partial \boldsymbol{i}\| \cdot \|\partial \boldsymbol{e} - \partial \boldsymbol{i}\|)^{1/2}} \tag{3.83}$$

∂r_{yz}^{min} is the minimum distance between the satellites when the relative motion of the follower around the leader is bounded (D'Amico, 2010). During one orbit period, the distance between the two changes. Eq. 3.83 calculates the value of the distance

at the closest point. Instead of using this constraint, it is possible to put conditions directly on ∂e, etc. by rewriting the three conditions in Eq. 3.80, 3.81, 3.82 in the following way

$$\|\partial \boldsymbol{e}\| - \varepsilon \geqslant 0 \tag{3.84}$$

$$\|\partial \boldsymbol{i}\| - \varepsilon \geqslant 0 \tag{3.85}$$

$$\|\partial \boldsymbol{e} \cdot \partial \boldsymbol{i}\| - \varepsilon\sqrt{\partial e^2 + \partial i^2 - \varepsilon^2} \geqslant 0 \tag{3.86}$$

The difference between $\|\partial \boldsymbol{e}\|$ and ε, $\|\partial \boldsymbol{i}\|$ and ε, $\|\partial \boldsymbol{e} \cdot \partial \boldsymbol{i}\|$ and ε should be greater or equal than 0 at the same time. However, if the smallest of the three terms is greater than 0 all three will be as well. This is where Eq. 3.75 comes from.

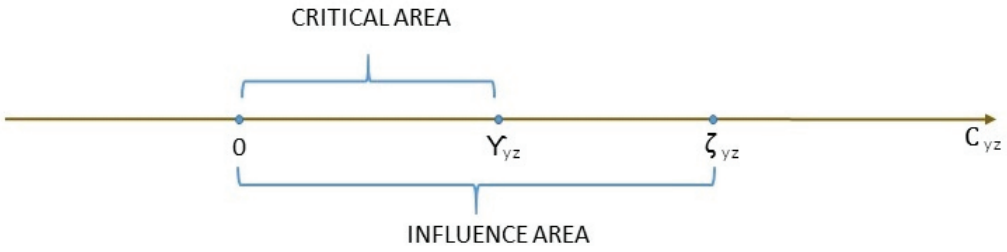


Figure 3.10: Graphical representation of the collision avoidance constraint

Fig 3.10 presents what was already presented for the wall constraints for the case of collision avoidance. $C_{yz} = 0$ means the state is on the boundary of the collision zone. $C_{yz} = \Upsilon_{yz}$ is the limit that defines the critical area, while $C_{yz} = \zeta_{yz}$ delimits the collision avoidance influence area.

It should be reminded that the collision avoidance technique described here refers to the case in which only two satellites are considered and one of theses satellites is placed in the center of the EH frame. As a consequence, the collision avoidance is later on re-formulated for N satellites and presented in Chap. 3.4.3.

3.4.2.3 Lyapunov Thresholds

The individual Lyapunov thresholds Γ_i are calculated independently for each constraint and then, the minimum among them is taken and used in Eq. 3.49. In this section, the different individual thresholds are derived based on the derivations of the constraints in the previous sections.

Γ_{time} is calculated starting from the control input Eq. 3.34. The component in the tracking error direction, normalized by the tracking error module, is

$$\hat{u}_c = \|\boldsymbol{B}^* \boldsymbol{P} \hat{\boldsymbol{\rho}}_e\| \tag{3.87}$$

where $\hat{\rho}_e = \frac{\Delta\partial\boldsymbol{\alpha}}{\|\Delta\partial\boldsymbol{\alpha}\|}$ is the current tracking error direction. The value of the desired control acceleration can be calculated taking the square root of the sum of the squares of $u_{d,ip}$ and $u_{d,oop}$ as given in Eq. 3.64 and 3.65.

$$u_d = \sqrt{u_{d,ip}^2 + u_{d,oop}^2} \tag{3.88}$$

u_d represents the needed control input when the maneuver has to be executed in a given number of orbits, i.e. when there is a time constraint. The ratio between u_d by $\hat{u}_c$ is the tracking error in $\hat{\rho}_e$ direction required to obtain the controller input u_d.

$$\|\Delta\partial\boldsymbol{\alpha}_{req}\| = \frac{u_d}{\hat{u}_c} \tag{3.89}$$

According to Eq. 3.35 there is a Lyapunov value corresponding to $\Delta\partial\boldsymbol{\alpha}_{req}$. This Lyapunov value is defined as the Lyapunov constraint Γ_{time}.

$$\Gamma_{time} = \frac{1}{2}\|\Delta\partial\boldsymbol{\alpha}_{req}\|^2 = \frac{1}{2}\left(\frac{u_d}{\|\boldsymbol{B}^*\boldsymbol{P}\hat{\boldsymbol{\rho}}_e\|}\right)^2 \tag{3.90}$$

In Eq. 3.90 the pseudo-inverse of $\boldsymbol{B}$, $\boldsymbol{B}^*$, and the feedback gain matrix $\boldsymbol{P}$ are computed at the ΔV-optimal mean arguments of latitude for in-plane reconfigurations $\bar{\varphi}_{ip}$ and for out-of-plane reconfigurations $\bar{\varphi}_{oop}$ (Steindorf et al., 2017). Thus, $\boldsymbol{P}$ will just be the identity matrix.

The same reasoning is applied to derive the thrust constraint for the Lyapunov threshold Γ_{thrust}, substituting u_d with the ratio of the maximum desired thrust F_{desired} and the spacecraft mass m_{sc}.

$$\Gamma_{thrust} = \frac{1}{2}\left(\frac{F_{desired}/m_{sc}}{\|\boldsymbol{B}^*\boldsymbol{P}\hat{\boldsymbol{\rho}}_e\|}\right)^2 \tag{3.91}$$

It is important to note that the time and thrust constraints cannot be applied simultaneously. The Lyapunov threshold for the wall constraints can be found by using the margin of the constraint

$$C_i = |\boldsymbol{c}_i^{\mathrm{T}}\partial\boldsymbol{\alpha}_a - d_i| \tag{3.92}$$

which is defined in Eq. 3.69. Rewriting the margin C_i in Lyapunov form of Eq. 3.35 yields

$$\Gamma_i = \frac{1}{2}C_i^2 = \frac{1}{2}(\boldsymbol{c}_i^{\mathrm{T}}\partial\boldsymbol{\alpha}_a - d_i)^2 \tag{3.93}$$

Similarly, the margin of the circular no-entry-zone is given by

$$C_i = \|\boldsymbol{G}_i^{\mathrm{T}}\partial\boldsymbol{\alpha}_a - \boldsymbol{x}_i\| - r_i \tag{3.94}$$

where $\boldsymbol{G}_i$, $\boldsymbol{x}_i$ and r_i are are defined in Eq. 3.71. In analogy to Eq. 3.93, the Lyapunov threshold for the circular no-entry-zone constraint 3.71 can be formulated by

$$\Gamma_i = \frac{1}{2}C_i^2 = \frac{1}{2}\left(\|\boldsymbol{G}_i^{\mathrm{T}}\partial\boldsymbol{\alpha}_a - \boldsymbol{x}_i\| - r_i\right)^2 \tag{3.95}$$

The Lyapunov threshold for the passive collision avoidance constraint is determined by first calculating the minimum cross-track separation given by D'Amico (2010) for near-circular orbits with $\partial a = 0$

$$\partial r_{yz}^{\min} = \frac{\sqrt{2}|\partial\boldsymbol{e}\cdot\partial\boldsymbol{i}|}{\left(\partial e^2 + \partial i^2 + \|\partial\boldsymbol{e}+\partial\boldsymbol{i}\|\cdot\|\partial\boldsymbol{e}-\partial\boldsymbol{i}\|\right)^{1/2}} \tag{3.96}$$

The maximum allowable tracking error $\Delta\partial\boldsymbol{\alpha}_{\max}$ can be determined by performing a first order Taylor expansion of Eq. 3.96 at the actual relative state $\partial\boldsymbol{\alpha}$ and is given by

$$\Delta\partial\boldsymbol{\alpha}_{\max} = \Delta\partial\boldsymbol{\alpha}\left(1 - \frac{\partial r_{yz}^{\min} - \varepsilon}{\left[0 \quad \frac{\partial\partial r_{yz}^{\min}}{\partial\partial e_x} \quad \frac{\partial\partial r_{yz}^{\min}}{\partial\partial e_y} \quad \frac{\partial\partial r_{yz}^{\min}}{\partial\partial i_x} \quad \frac{\partial\partial r_{yz}^{\min}}{\partial\partial i_y}\right]_{\partial\boldsymbol{\alpha}} \cdot \Delta\partial\boldsymbol{\alpha}_a}\right) \tag{3.97}$$

In 3.97 the state vector is considered to be made just of the ∂e and ∂i vector components because they are the only one influencing the $\hat{\boldsymbol{e}}_{\boldsymbol{y}}$-$\hat{\boldsymbol{e}}_{\boldsymbol{z}}$ plane inter-satellite distance. The denominator in Eq. 3.97 describes the unit change of $\partial r_{yz}^{\min}$ for an increase of the tracking error, while the numerator is the margin of the constraint (see the defintion of ε in Eq. 3.77). Therefore, the fraction describes how much the tracking error is allowed to become larger. It is noteworthy that an increase of the tracking error not always results in a smaller $\partial r_{yz}^{\min}$. The fraction nominator indicates how much the minimum distance ∂r_{yz}^{min} is allowed to decrease before reaching the allowed minimum value ε. The denominator quantifies the decrease of ∂r_{yz}^{min} when the tracking error increases in the current direction, supposed constant. Since this rate of change is negative (from the previous assumption), the quantity inside the brackets has the form of $1+\Delta$, where Δ represents how much the tracking error can still increase before reaching $\partial r_{yz}^{min} = \varepsilon$. The entire expression calculates the maximum allowed tracking error which can be used in Eq. 3.35 to calculate the Lyapunov threshold relative to collision avoidance.

$$\Gamma_{yz} = \begin{cases} \frac{1}{2}\Delta\partial\boldsymbol{\alpha}_{max}^T, & \text{for } \left[0 \ \frac{\partial r_{yz}^{min}}{\partial e_x} \ \frac{\partial r_{yz}^{min}}{\partial e_y} \ \frac{\partial r_{yz}^{min}}{\partial i_x} \ \frac{\partial r_{yz}^{min}}{\partial i_y}\right]\cdot\Delta\partial\boldsymbol{\alpha}_a < 0 \\ \infty, & \text{elsewhere} \end{cases} \tag{3.98}$$

As previously mentioned, for this reasoning to be valid, an increase in tracking error should lead to a decrease of ∂r_{yz}^{min}. If the opposite happens the Lyapunov threshold relative to collision avoidance is not effective anymore and it should be eliminated. A way to implement this, is to set it equal to infinity and it will be ignored during the calculation of the overall Lyapunov threshold. The overall Lyapunov threshold

is then derived by computing the minimum of all individual Lyapunov thresholds.

$$\Gamma = min\left\{\Gamma_{time},\, \Gamma_{thrust},\, \Gamma_i,\, \Gamma_{rn}\right\} \tag{3.99}$$

3.4.3 Extension to N Satellites and Collision Avoidance

A controller for two satellites, leader and follower, flying in formation has been introduced so far in this chapter. The control action is jointly executed by both spacecraft as a function of the sign of the resulting control input vector. If the control law in Eq. 3.34 yields a positive control input it is executed by the follower, while, if the $\boldsymbol{u}$ vector has a negative sign, it is executed by the leader. It is clear that the main aim of this controller is maintaining a given relative configuration, neglecting the absolute orbital elements of both satellites and supposing that the EH frame is always centered on the leader spacecraft.

However, SFF mission design usually follows another approach. A virtual leader performs a given reference orbit and, in the general case, the center of the EH frame does not coincide with a physical satellite. Once established, the formation is constituted by more than one follower. The N satellites belonging to the formation have their own controller based on their own ROE vector, describing the relative motion with respect to the virtual leader. Each satellite ignores the presence of the other satellites if no communication between them is established. For collision avoidance purposes the spacecraft should in fact communicate and know the ROE vectors of all the other satellites of the formation.

That the key difference between the approach discussed previously and the extension to N satellites formation lies in the impossibility of performing a control action on the leader. In Chap. 2.5.5 the state propagation matrix $\boldsymbol{A}$ and control input matrix $\boldsymbol{B}$ have been introduced. The $\boldsymbol{A}$ and $\boldsymbol{B}$ values depend on the absolute orbital elements of the leader or follower when the control action is executed respectively on the leader or follower. In the extension to N satellites the control action is always executed on the follower. For this reason $\boldsymbol{A}$ and $\boldsymbol{B}$ always refer to the follower satellite where the controller is applied.

Another key difference between the single leader/single follower approach and the N satellites application is the collision avoidance strategy. In the previous sections a passive collision avoidance approach has been developed defining a maximum distance between the follower and the center of the EH frame. Applying the same concept to a formation of N satellites would not be effective since the satellites would not be able to come close to the origin of the relative frame, but they could collide when they are far from it. For this reason, a new collision avoidance strategy has been developed, however with the need of a functioning inter-satellite communication. In the following sections two different strategies are presented.

3.4.3.1 Collision Avoidance

The idea at the base of the first collision avoidance strategy is the introduction of terms in the potential field of Eq. 3.53 that make the overall potential field positive, when two satellites get too close to each other: the closer the satellites are, the higher the overall potential becomes. In case of N satellites, $N-1$ potential terms need to be introduced. Fig. 3.11 shows the specific case of $N=4$, when the controlled satellite is Satellite 1, which sees three more spacecraft. In this situation to the existing potential field, three more terms need to be added, one for each collision harm. In Fig. 3.11, since there are three satellites that Satellite 1 can possibly collide with, there will be three more terms. At the same time, Satellite 2 needs three more terms as well, to avoid collision with Satellites 1, 3 and 4 and so on.

Figure 3.11: Schematic of communication process between the controlled Satellite 1 and the formation

Eq. 3.100 is the updated potential field gradient expression, where the $N-1$ collision avoidance terms appear as $\nabla\phi\alpha_j$.

$$\boldsymbol{\rho} = -\nabla\bar{\phi} - \nabla\phi_{ix} - \nabla\phi_{ey} - \nabla\phi_i - \nabla\phi_{\alpha_j} \qquad j = 1 : N-1 \tag{3.100}$$

Let the i-th satellite of the formation be the one where the control action is executed. Let j-th satellite be an in-view satellite for which the collision risk needs to be excluded. Let $\Delta s\partial\boldsymbol{\alpha}_j$ be the norm of the vector difference between the i-th and the j-th satellites ROEs (Eq. 3.101 and 3.102).

$$\partial\boldsymbol{\alpha}_i = \begin{pmatrix} \partial a_i \\ \partial\lambda_i \\ \partial e_{x,i} \\ \partial e_{y,i} \\ \partial i_{x,i} \\ \partial i_{y,i} \end{pmatrix} \tag{3.101}$$

$$\partial \boldsymbol{\alpha}_j = \begin{pmatrix} \partial a_j \\ \partial \lambda_j \\ \partial e_{x,j} \\ \partial e_{y,j} \\ \partial i_{x,j} \\ \partial i_{y,j} \end{pmatrix} \tag{3.102}$$

$$\Delta s \partial \boldsymbol{\alpha}_j = \| \partial \boldsymbol{\alpha}_i - \partial \boldsymbol{\alpha}_j \| \, a \tag{3.103}$$

The semi-major axis of the leader a is added to make the upcoming definition of a threshold for collision avoidance clearer. For any type of formation or any relative configuration, collision surely happens when two spacecraft have the same ROE vector. In other words, when the sum of the squared difference between each i-th and j-th ROE approaches zero, the i-th and j-th satellites are colliding. This leads to the definition of an influence area, inside which there is collision harm, defined by a given value of $\Delta s \partial \boldsymbol{\alpha}_j$. Letting this user-input threshold value be C_s, the potential $\phi_{\boldsymbol{\alpha}_j}$ is defined by

$$\phi_{\boldsymbol{\alpha}_j} = \begin{Bmatrix} \frac{a}{\Delta s \partial \boldsymbol{\alpha}_j} & if \ \Delta s \partial \boldsymbol{\alpha}_j \leq C_s \\ 0 & if \ \Delta s \partial \boldsymbol{\alpha}_j > C_s \end{Bmatrix} \tag{3.104}$$

where a is again the leader semi-major axis that makes the potential non-dimensional. Fig. 3.12 shows that this simple approach leads to the desired behavior. In fact, when the inter-satellite euclidean distance decreases the potential increases with the inverse of the distance, acting as a force pushing the satellites far from each other. When the distance between the j-th and i-th satellites is greater than the threshold C_s the collision avoidance potential term does not have any influence on the overall potential.
Inside the influence area delimited by C_s, $\nabla \phi_{\boldsymbol{\alpha}_j}$ should be able to change the overall potential gradient from negative (attractive) to positive (repulsive). Since the only negative potential gradient is the global potential field gradient, when $\Delta s \partial \boldsymbol{\alpha}_j < C_s$, $\nabla \phi_{\alpha_j}$ direction should be opposite to $\nabla \bar{\phi}$ with a magnitude of $\phi_{\boldsymbol{\alpha}_j}$. This is expressed by the following Eq. 3.105.

$$\nabla \phi_{\boldsymbol{\alpha}_j} = \phi_{\boldsymbol{\alpha}_j} \cdot \left(-\nabla \bar{\phi} \right) \tag{3.105}$$

Calculation of $\nabla \phi_{\alpha_j}$ is done for each of the $N-1$ satellites ($j \neq i$) and included in $\boldsymbol{\rho}$ calculation, as seen in Eq. 3.100.

Alternative Approach

The alternative approach performs collision harm detection in the same way discussed before, i. e. checking if $\Delta s \partial \boldsymbol{\alpha}_j$ is less or equal than C_s. After checking whether the satellites are inside the collision influence area, the idea is to define the potential directly as a function of the single ROE differences between the i-th and j-th satellite, without using the global potential. As an advantage, when the satellites are close to

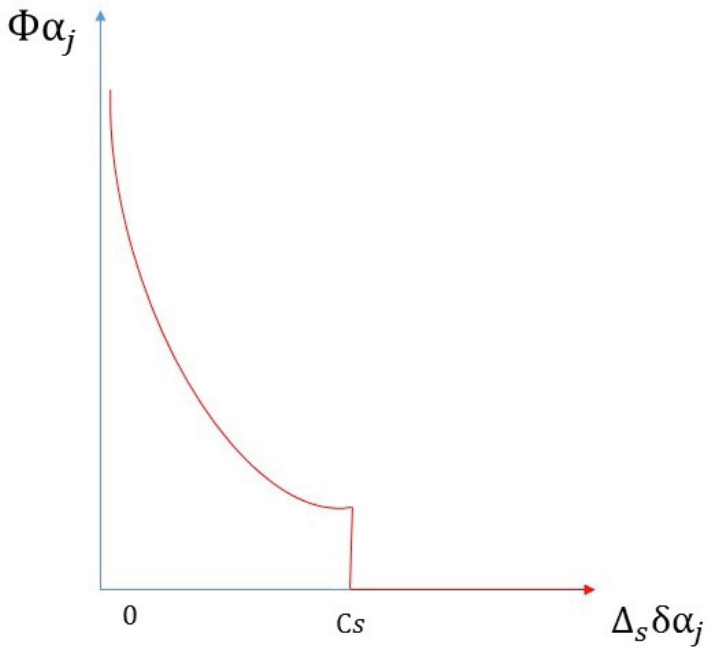


Figure 3.12: The potential function for collision avoidance as a function of the distance between any two satellites

each other, the gradient acts to increase only the specific relative elements that have close values. In this way, the satellites feel a repulsive force in the same direction in which they are colliding. The expression of $\nabla\phi_{\boldsymbol{\alpha}_j}$ is shown in Eq. 3.106.

$$\nabla\phi_{\boldsymbol{\alpha}_j} = -\begin{pmatrix} 0 \\ \frac{1}{\Delta\partial\lambda} \\ \frac{1}{\Delta\partial e_x} \\ \frac{1}{\Delta\partial e_y} \\ \frac{1}{\Delta\partial i_x} \\ \frac{1}{\Delta\partial i_y} \end{pmatrix} \tag{3.106}$$

Comparison

To compare and evaluate the two collision avoidance strategies an ATF maneuver has been selected. Four satellites are placed on the same orbit at distinct positions of 200 m, 100 m, 0 m and −100 m distance to the origin of the EH frame. Satellite 1 has to change its position from 200 m to −200 m which would cause him to collide with the other satellites. Initial and target conditions are given in Tab. 3.1.
Fig. 3.13 shows the trajectories of the four satellites while applying the main collision avoidance strategy. It can be seen that only the satellites, whose trajectories coincide, are moving. Satellite 1 increases the spiral size in order to avoid collision with the

Table 3.1: Initial and target states of the ATF scenario to evaluate the collision avoidance methods in ROEs in $[m]$.

	Initial conditions				Target conditions			
Satellite	1	2	3	4	1	2	3	4
∂a	0	0	0	0	0	0	0	0
∂e_x	0	0	0	0	0	0	0	0
∂e_y	0	0	0	0	0	0	0	0
∂i_x	0	0	0	0	0	0	0	0
∂i_y	0	0	0	0	0	0	0	0
∂u	200	100	0	-100	-200	100	0	-100

other satellites. When it finally arrives at its target location $y = -200\,m$, it does not manage to stop quickly. It needs to reduce the orbit size slowly, traveling for 100 m further, coming back and moving forward again. This is due to the use of a very low thrust propulsion system, which requires time for significant orbit changes.

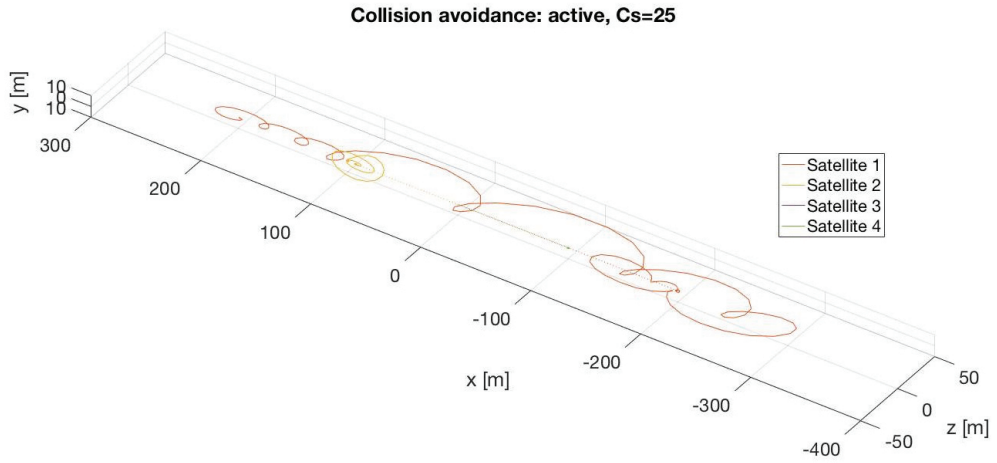


Figure 3.13: Main collision avoidance method performing an ATF scenario with $C_s = 25\,m$.

Fig. 3.14 shows the trajectories of the four satellites while applying the alternative collision avoidance strategy. Here, the motion of Satellite 2 becomes more convoluted and Satellite 3 moves as well. Using this method, a greater ΔV consumption is needed except for Satellite 4 which does not perform any motion. This can be seen from Tab. 3.2 which presents the ΔV consumption per satellite and in total for both methods. The analysis and comparison between the two approaches reveal that both methods accomplish collision avoidance. Although the alternative approach requires a ΔV consumption for the satellites that are not performing the maneuver as well. As Method 1 is more fuel-efficient, it has been selected as more suitable for the controller in replacement of the original single follower/single leader collision avoidance strategy.

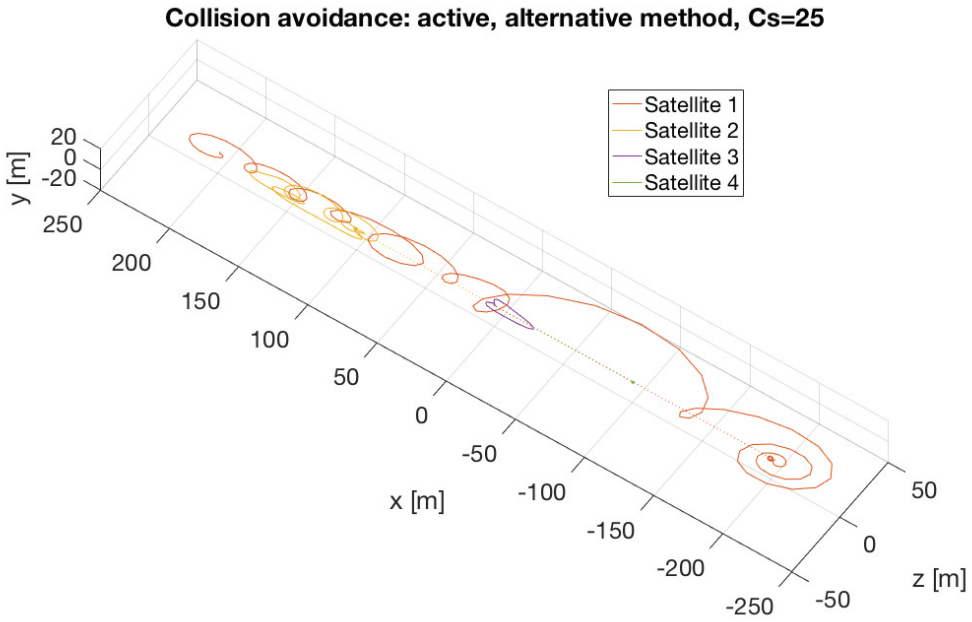


Figure 3.14: Alternative collision avoidance method performing an ATF scenario with $C_s = 25\,m$.

Table 3.2: ΔV consumption during ATF scenario while performing different collision avoidance methods ($C_s = 25\ m$) in $[m/s]$.

Satellite	ΔV Method 1	ΔV Method 1
1	0.0838	0.0945
2	0.0320	0.0628
3	0	0.0274
4	0	0
Sum	0.1158	0.1847

Chapter 4

Sensor and Actuator Systems

4.1 Overview

This chapter presents sensor and actuator systems that are essential for SFF, ranging from navigation sensors, attitude sensors and actuators to propulsion systems, to analyze, what is required to compose a functioning and comprehensive overall GNC system for a SFF satellite. For each type, characteristics and limitations are depicted as well as a typical example is given. In addition, typical filtering or control methods are explained as well as common verification approaches. The set of sensors and actuators may vary significantly based on the individual mission goal and duration, autonomy requirements, the satellites' size, the selected main orbit, disturbances present in this orbit, chosen formation topologies and maneuvers, pointing/positioning accuracy requirements etc. Thus, the examples are based on the NetSat mission, since it shows many common characteristics and requirements of typical SFF missions.

4.1.1 Accuracy Requirements

Different SFF missions and applications demand various requirements in terms of measurement and control accuracy. The scaling laws defined by Gill et al. (2010) present a guideline for mission planning of such missions. They provide an initial guess on required accuracies for SFF by contrasting different GNC parameters with the relative distance / along-track separation. They are presented in Tab. 4.1. Although they can be a helpful initial guess, the parameters for each individual mission depend on many more factors and have to be decided on a case-by-case basis.

4.2 Sensor Systems for Satellite Formation Flying

Sensor systems for SFF span from attitude sensors like sun sensors and magnetometers to navigation sensors like GNSS. Even the ISL has to be considered as a sensor system, because it (indirectly) provides relative position and velocity measurements. There are typical performance requirements that apply to all kinds of sensors like the accuracy i.e. the quality of the orientation measurement with respect to a reference

Table 4.1: Gill's scaling laws for mission analysis of GNC parameters in SFF (Gill et al., 2010).

Parameter	Variable/Relation	Sample Scenario
Along-track separation	d_1	10 km
Control window size	$d_{cw} = d_1 \cdot 10^{-1}$	1 km
Control accuracy requirement	$\sigma_c = d_{cw} \cdot 10^{-1}$	100 m
Navigation accuracy requirement	$\sigma_n = \sigma_c \cdot 10^{-1}$	10 m

direction for an attitude sensor or the quality of position measurement with respect to a reference point for a navigation sensor. The range in which a sensor can provide reliable values is another typical performance requirement. For attitude sensors it may be the variation width of the angular motion where the accuracy requirements can still be met or for navigation sensors the variation of velocity measurements within which the accuracy requirements still hold.

4.2.1 Attitude Sensors

The selection of attitude sensors depends on the mission requirements like accuracy (see Chap. 4.1.1), available mass, size, electrical power and thus is highly mission-dependent. On the other hand, the selection of attitude sensors affects the performance of the mission and its schedule. For the NetSat mission the selected set of attitude sensors (detailed in the next sections) allows for attitude determination with an accuracy of up to 1°, but limits this to two axes during eclipse time due to the unavailability of sun sensors in this phase. Thus thrust maneuvers have to be planned in sun time. In the following sections, typical attitude sensors are detailed, however the sensor selection is limited to those required for typical formation flying missions like NetSat. Further sensors providing e.g. higher accuracy as required by some scientific missions are not covered here.

4.2.1.1 Inertial Measurement Units

An Inertial Measurement Unit (IMU) is a device that measures angular velocities and accelerations that act on it. This is enabled by including gyroscopes and accelerometers on all axes. Nowadays especially on small satellites sensors based on Micro-Electro-Mechanical Systems (MEMS) are commonly used, due to their small size and low power consumption. Typical sensors for space applications show accuracies of $0.01\,^\circ\,\mathrm{s}^{-1}$ or better for the gyroscopes and 10 mg or better for the accelerometers (cf. e.g. Bosch Sensortec GmbH (2012)). However, IMUs typically suffer from sensor drift which leads to an accumulation of their errors over time. Thus, depending on the chosen IMU sensor orientation or position values computed from its sensor data are not reliable any more after seconds to minutes and another sensor (e.g. sun sensor, GNSS) is required for recalibration. Because of that, IMUs

are commonly used in combination with other sensors to complement orientation or position measurements especially when other sensors are temporary not available.

In the NetSat mission MEMS-based IMUs are used from the manufacturer Bosch Sensortec GmbH, namely the BMX055 Digital 9-axis sensor as depicted in Fig. 4.1. They are placed inside the satellites on the Attitude Determination and Control System (ADCS) Printed Circuit Board (PCB) and the outer panels for redundancy reasons. They provide an accuracy of $0.004\,^{\circ}\,\mathrm{s}^{-1}$ for the gyroscopes and 0.98 mg for the accelerometers. Within the ADCS of NetSat sensor values are filtered with the help of a Kalman filter and used for attitude propagation, which is a common approach for space missions.

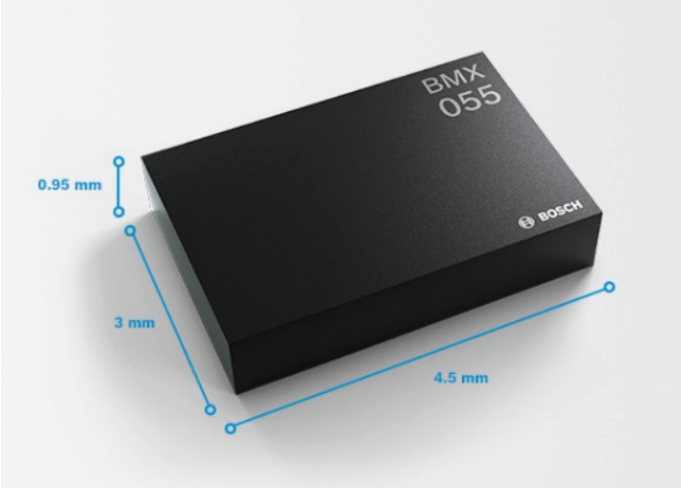


Figure 4.1: Inertial Measurement Unit BMX055 from Bosch Sensortec GmbH used e.g. in the NetSat mission. (Image courtesy: (Bosch Sensortec GmbH, 2012))

IMUs can be tested by providing an external controlled rotation rate and an external controlled acceleration. This can be provided e.g. by dynamic test facilities (so-called turn tables). Such a facility is situated in Würzburg, Germany at the S4 - Smart Small Satellite Systems GmbH and is depicted in Fig. 4.2. This facility is able to provide highly accurate rotation rates with an accuracy of $0.0001\,^{\circ}\,\mathrm{s}^{-1}$ (S4 - Smart Small Satellite Systems GmbH, 2019). Thus the angular velocities and accelerations provided by an IMU can be verified. Further, such facilities are used to calibrate IMUs before launch.

IMUs are a very common and useful type of attitude sensor and are part of almost every satellite due to their small size and low power consumption and their ability to bridge gaps in measurements of absolute sensors like sun sensors during eclipse or GNSS during outages. However, the usage of IMUs alone is not reasonable because their errors accumulate significantly over time.

Figure 4.2: Dynamic test facilities (turn tables) at S4 - Smart Small Satellite Systems GmbH, Würzburg providing high accuracy pointing and rotation to test attitude sensors. (Image courtesy: S4 - Smart Small Satellite Systems GmbH)

4.2.1.2 Magnetometer

The next attitude sensor type of interest are magnetometers, which measure magnetic fields or more precisely magnetic flux. Common sensors provide measurements already in three dimensions, so a magnetic flux vector $\vec{B}$ is provided. Typical sensors provide accuracies of 1 µT or better (cf. e.g. NewSpace Systems (2020) or Bosch Sensortec GmbH (2012)).

In space applications the surrounding magnetic field of the Earth is of interest. With the knowledge of the (rough) position of the satellite and a model of Earth's magnetic field, the satellite's orientation can be computed. However, the orientation can only be determined in two axes. The orientation along the axis parallel to the Earth's magnetic field lines cannot be computed, since it cannot be derived from the magnetic field / flux vector (only the direction of the vector can be measured, not the rotation around this vector). In addition, the real measured Earth's magnetic field may be different to the model stored on-board the satellite due model uncertainties or time-varying disturbances. Further, because Earth's magnetic field changes quickly around the poles, the accuracy of the orientation determination around the poles decreases. Here additional sensors like IMUs are especially helpful. Magnetometers

can also be disturbed by electric or electronic components on-board the satellite itself, since the generated currents create magnetic fields. Thus a superposition of Earth's magnetic field and the satellite-dependent disturbance fields is measured. Thus it is highly recommended to separate magnetometers from the strong currents on-board the satellites. This may be achieved by placing the sensors on the outer panels. Some satellites with extremely high accuracy requirements also implement booms where the magnetometers are placed on (e.g. the ESA SWARM mission).
Magnetometers are part of almost every mission because of their low costs, robustness, small size and low power consumption. E.g. in NetSat the combined sensor BMX055 Digital 9-axis sensor (Bosch Sensortec GmbH, 2012) is placed on all panels and on the ADCS PCB for redundancy reasons and provides three-axis magnetic field / flux measurements with an accuracy of 0.3 µT with a size of 3 mm × 4.5 mm × 0.95 mm. The magnetic flux measurements are fed into a Kalman filter with which the satellite's orientation can be determined (in two axes) with an accuracy below 5°.
Magnetometers require a precisely controlled external magnetic field to be tested and calibrated. Usually a set of three Helmholtz-coils (on in each spatial direction) is used for this purpose. Thus, the Earth's magnetic field on ground (measured with another already calibrated magnetometer) can be compensated and a precomputed magnetic field vector as it would be expected on a certain position in orbit can be applied. A typical test facility is the three-axis Magnetic Coil Facility at ESA's European Space Research and Technology Centre (ESTEC) shown in Fig. 4.3. However, any calibration is preferably repeated in orbit.

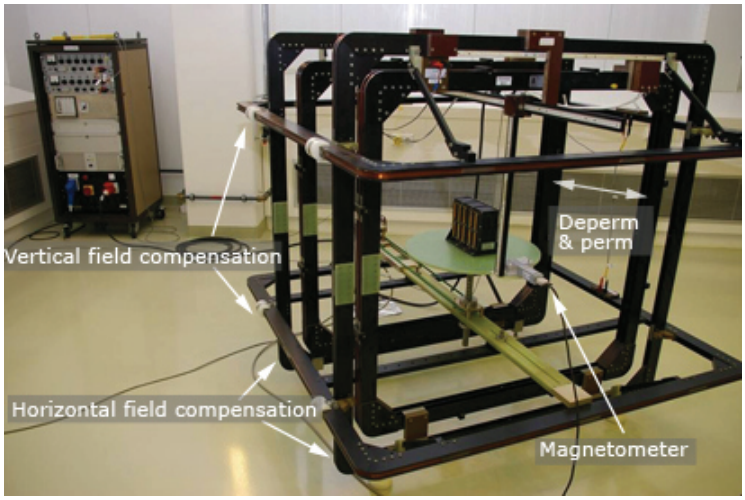


Figure 4.3: Three-axis Magnetic Coil Facility at ESA's ESTEC in Noordwijk, the Netherlands consisting of three perpendicularly placed Helmholtz coils. (Image courtesy: ESA)

Magnetometers are among the most common attitude sensors due to their small size, low power demands and low costs. They can provide reliable, though not highly accurate attitude determination in two axes. Usually they are combined with other

sensors which can bridge gaps during outages (e.g. over the poles) and sensors which can provide the orientation along the third axis (e.g. sun sensors, star senors).

4.2.1.3 Sun Sensors

Sun sensors are a type of sensor that determines the sun angle relative to the sensor, in detail determines the vector from the sensor towards the sun. There are different types of sensors ranging from quadrant photo diodes to camera systems, depending on accuracy requirements. Sun sensors are placed on almost every Earth orbiting satellite and commonly multiple sensors are installed per satellite to be able to track the sun continuously. Typical sun sensors for small satellites provide accuracies in the order of 1° or better (cf. e.g. Solar MEMS Technologies (2016)). However, sun sensors cannot work during eclipse phases where the sun is not visible. In LEO orbits this phase is approximately 30 min per orbit period of approximately 90 min. In addition, sun sensors may not be able to determine a sun vector, if the satellite is rotating too fast. So depending on the selected sun sensor and the tumbling rate of the satellite, first a detumbling (e.g. using magnetometers and magnetorquers) has to be performed. (This also applies to other vision-based attitude sensors like Earth or star sensor.)

In NetSat a specific miniature camera with adapted software and optics is used as sun sensor (cf. Fig. 4.4). The sensor developed by OptaSensor GmbH features a resolution of 250 pixel × 250 pixel within a volume of 1.0 mm × 1.0 mm × 1.7 mm and a nominal power consumption of less than 4.2 mW. In NetSat a specific lens with a field of view of 90° is used so that the sun can be tracked continuously, if one sensor is placed on each side of a CubeSat. Sun estimation and sun vector computation is performed with a self-developed algorithm (Chmiela, 2018).

To test and calibrate a sun sensor both a sun simulator generating sun-like light with the correct spectrum and comparable intensity as the sun and dynamic test facilities (turn tables) as depicted in Fig. 4.2 are required. With such a facility the sensor (or the complete satellite including the sensor) can be oriented in a well-defined way with respect to the simulated sun, so that the sun sensor can be calibrated and its performance can be validated. Fig. 4.5 shows such a calibration setup whereas Fig. 4.6 shows a calibration matrix with the measured accuracies for the different pixels of the camera / sun sensor. If the sensor is calibrated properly, an accuracy of approximately 0.1° can be reached (Chmiela, 2018).

Sun sensors are another common and useful attitude sensor. Within the last years miniaturization led to smaller and smaller sensors so that they can be easily integrated into any satellite, especially CubeSats. Sun sensors can provide three axis attitude information with accuracies of at least 1°. Their main drawback is their unavailability during eclipse times. Within these time frames other attitude sensors like magnetometers and IMUs are required to bridge the gap or a lower accuracy has to be accepted.

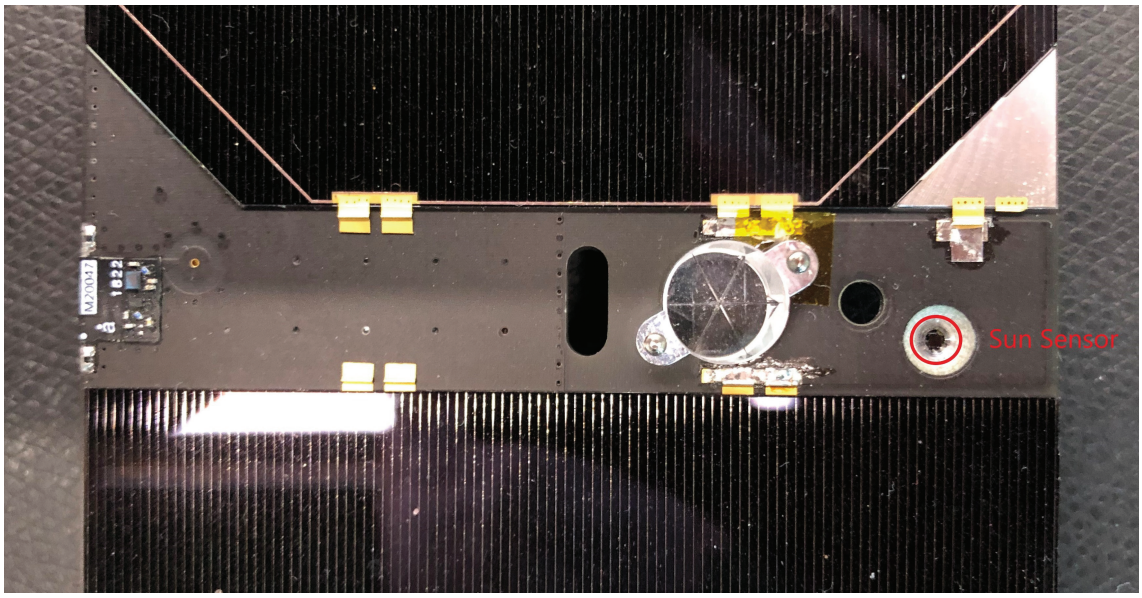


Figure 4.4: NetSat sensor bar located at the lower end of each 3U panel. It holds a GNSS antenna, a retro-reflector as well as the adapted miniature camera used as sun sensor (highlighted in red) with a size of 1.0 mm × 1.0 mm × 1.7 mm. (Image courtesy of the author.)

4.2.2 Navigation Sensors

Navigation sensors are required to determine the relative and absolute positions and velocities of satellites within a formation. (As a minimum, the absolute position of one leader satellite together with the relative positions of the other satellites of the formation with respect to this leader satellite are required.) SFF imposes high demands on navigation sensor systems. Although high accuracy is desirable, also other characteristics like size, weight, complexity or cost need to be taken into account especially for small satellite missions. Hence, to select suitable sensors for SFF missions, a Design Option Tree (DOT) is presented in Fig. 4.7. The most inexpensive and uncomplicated approach is the use of TLEs, but it is also the most inaccurate one. Thus it is not recommended and not suitable for most of the formation flying applications. The accuracy achievable with this approach is about 30 - 100 m in radial and cross-track direction and better than 500 m in along-track direction (Gill, 2011, Kirschner et al., 2001). In addition, this accuracy decays over time and TLEs usually are updated only on a daily base as a maximum. Absolute navigation sensors like GPS or other GNSS sensors provide higher accuracies up to few meters within a size that is suitable for small satellites. More accuracy can be achieved by relative navigation sensors like cameras, which provide, depending on the range sub-centimeter accuracy, but occupy – depending on the optics – significantly more space. An intermediate solution is the use of radio-frequency based ranging, which typically provides less accuracy than optical systems, but on the other hand can be

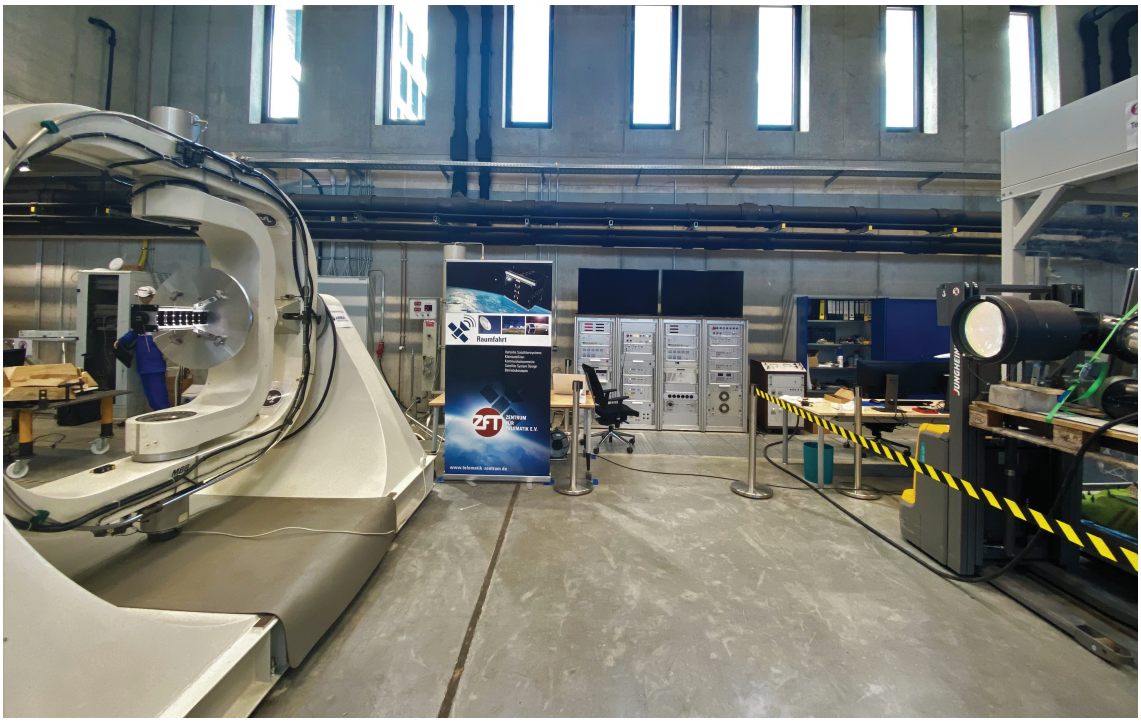


Figure 4.5: Calibration and verification setup for a sun sensor using a dynamic test facility (turn tables) and a sun simulator. The 3U CubeSat with the sun sensor is placed on the turn table on the left hand side, which can perform highly precise orientations, and is illuminated by the sun simulator on the right hand side. (Image courtesy: S4 - Smart Small Satellite Systems GmbH)

used for larger relative distances and requires less pointing accuracy of the sensor itself towards the targeted other satellite. In addition, this sensor may be combined with the ISL communication system and thus saving volume, mass and power within the satellite.
Among these presented different sensor options only few are of special interest to SFF by providing sufficiently high accuracy over a significantly large range of inter-satellite distance. These are mainly GNSS sensors as well as radio-based ranging systems. Thus, these are detailed in the following sections.

4.2.2.1 GNSS Sensors

Global Navigation Satellite System (GNSS) sensors are the most common sensor type among relative navigation sensors. Especially on small satellites where other larger navigation sensors are infeasible, GNSS sensors are often the only option. They use reference signals by a GNSS satellite constellation like the American GPS, the Russian GLONASS, the European Galileo or the Chinese BeiDou network to determine its position and velocity. Several satellites of the used GNSS constellation

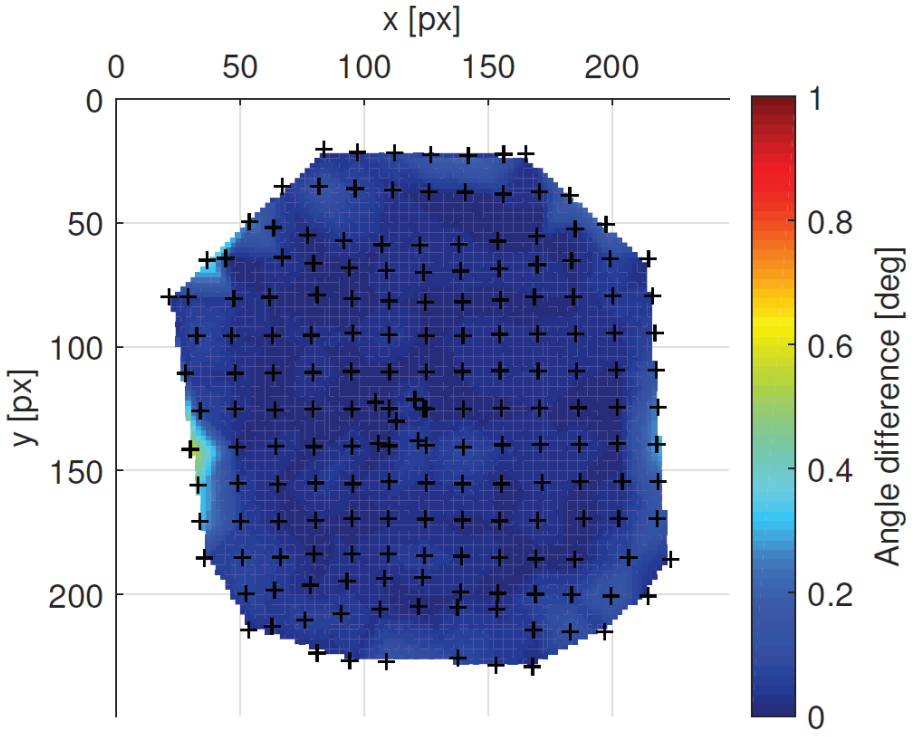


Figure 4.6: Results of sun sensor / camera calibration showing the calibrated and verified accuracies for the different pixels of the sensor (Chmiela, 2018).

have to be within the field of view of the sensor. (At least four to get an unambiguous position in three dimensions.) This is feasible in LEOs, as long as no shading e.g. by the satellite itself or misalignment of the GNSS antenna with respect to the GNSS constellation occurs, since the GNSS constellations are located above LEO satellites in Medium Earth Orbits (MEOs). Small satellites and CubeSats with higher accuracy requirements usually implement a GNSS receiver with an attached antenna, however for redundancy reasons also multiple receivers/antennas may be incorporated. The sensors typically provide a state vector (position and velocity) measurement at least every second with an accuracy of 40 m for position and 1 m s^{-1} for velocity or better. In addition, a time signal based on the GNSS reference signal is provided with typical accuracies better than 100 ns (cf. e.g. NovAtel Inc. (2020)). Besides the mentioned shading and antenna misalignment which can lead to unsteady measurements, if not enough satellites of the GNSS constellation are in the field of view of the GNSS antenna, one major disadvantage is the required start-up time of the sensor. Depending on the duration of inactivity the start-up phase may last from 1 min to 12 min, because a list of orbit parameters of the GNSS constellation satellites, which is part of the GNSS signal itself, has to be received.

In NetSat four redundant GNSS receiver/antenna pairs are incorporated, one set on each of the four 3U side panels. So no pointing of the antennas towards the GNSS constellation is necessary and at least two receivers can receive multiple GNSS signals

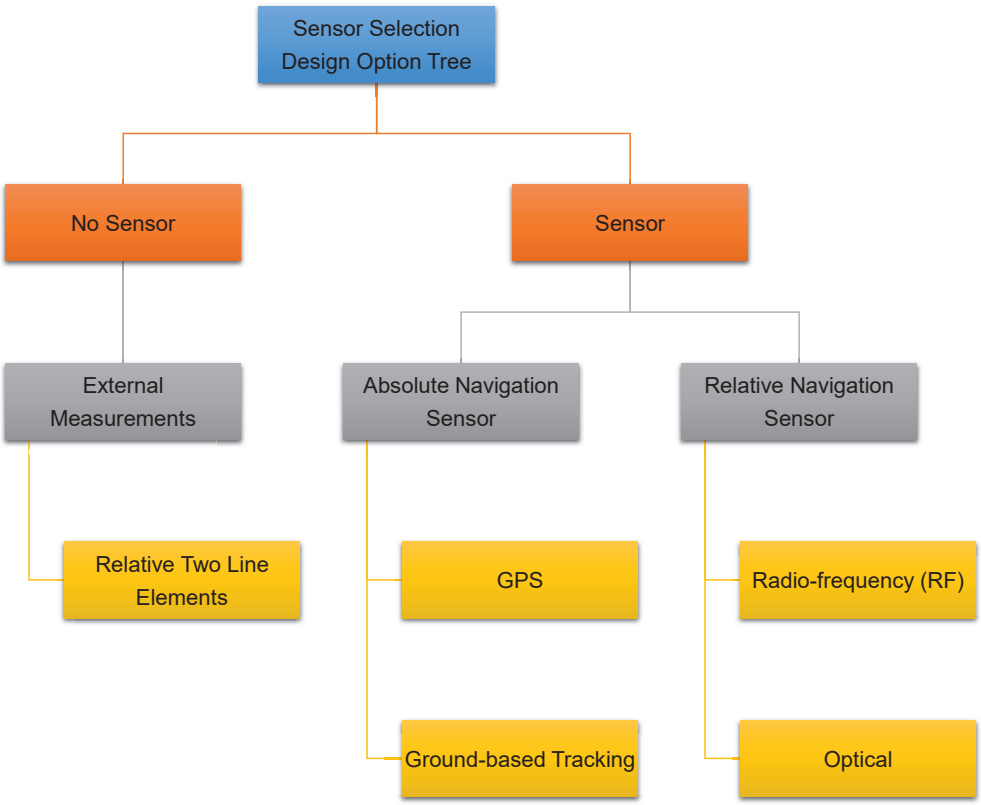


Figure 4.7: Design Option Tree (DOT) for formation flying sensor selection. Yellow boxes indicate sensor hardware, whereas gray boxes indicate sensor types (Gill, 2011).

continuously. Thus, redundancy is implemented. As GNSS sensor a GNSS200 by Hyperion Technologies B.V. is chosen (cf. Fig. 4.8). It has been selected because of its small size of 20 mm × 15 mm × 3 mm and low power consumption of less than 150 mW at peak. This allows for the described highly redundant setup. The sensor provides an accuracy of better than 8 m in postion (Hyperion Technologies B.V., 2019).

Position determination based on a GNSS sensor can be improved by filtering. In NetSat e.g. a Kalman filter is implemented which improves the accuracy of the determination to below 5 m in position and below 0.1 m s^{-1} in velocity (Kunzi, 2018). GNSS receivers can be tested in multiple ways. A sensor can be placed just under open sky and reception as well as accuracy can be verified. However, typical environmental conditions as well as velocities as encountered in LEO are not in place (e.g. typical satellite velocity in LEO is about 7.5 km s^{-1}. There are specific GNSS simulators (e.g. the GNSS hardware simulator at DLR Institute of Communications and Navigation in Oberpfaffenhofen, Germany as depicted in Fig. 4.9) which can generate GNSS signals as if the receiver would be in a specific predefined orbit. Thus

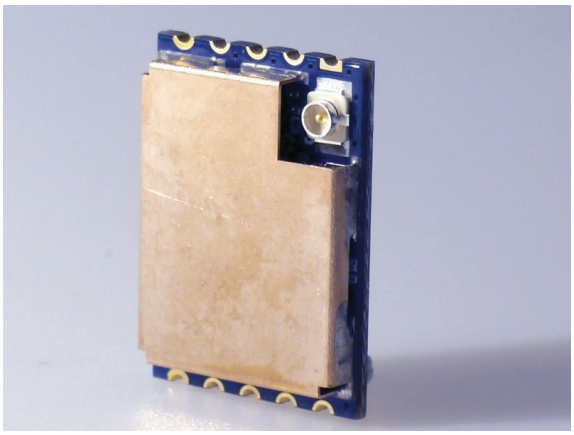


Figure 4.8: GNSS receiver for small satellites by Hyperion Technologies B.V., Delft, Netherlands providing accurate position and velocity measurements in space within a size of 20 mm × 15 mm × 3 mm. (Image courtesy: Hyperion Technologies B.V.)

realistic signals can be simulated and GNSS receivers can be evaluated properly. However, as such a simulator is directly connected to the GNSS receiver, realistic testing of the antenna is not possible. The described open sky test is more suitable for antenna evaluation. There are also specific anechoic chambers to evaluate various types of antennas.

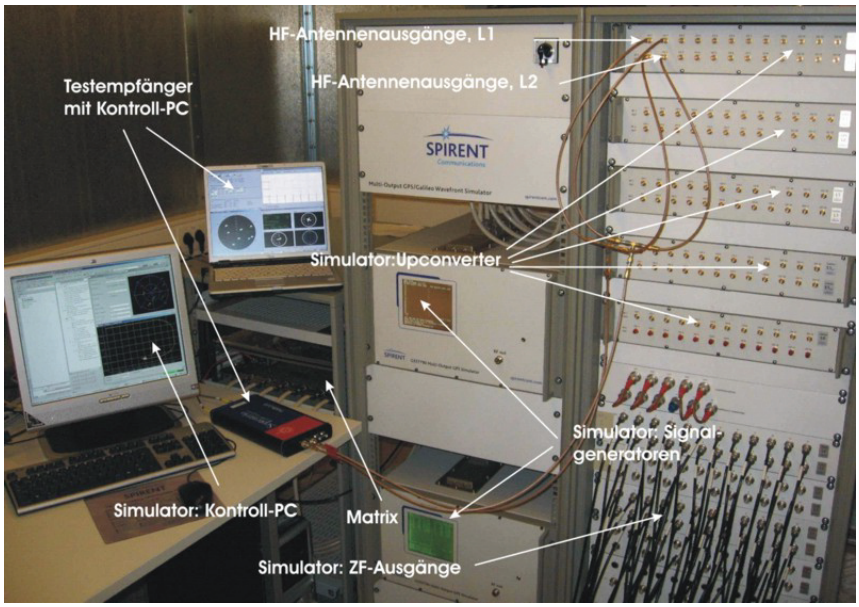


Figure 4.9: GNSS hardware simulator at DLR Institute of Communications and Navigation in Oberpfaffenhofen, Germany. (Image courtesy: DLR)

GNSS sensors are one of the most common navigation sensors in space and especially in small satellites mostly the only sensor implemented. They provide high accuracy position and velocity measurements with high frequency and can be nearly continuously active. They are relatively small and show very low power requirements. Thus, for most of the small satellite applications with position/velocity requirements (including SFF) GNSS sensors are the most suitable choice. To gain best results, the use of a filter (e.g. Kalman filter) is highly recommended. If accuracies below 1 m are required, different sensors have to be considered. Also extensions to GNSS like Differential GPS can be considered, if accuracy requirements concern only the relative position/velocity between satellites.

4.2.2.2 Radio-Based Ranging

In contrast to GNSS-based navigation, radio-based ranging (or inter-satellite ranging) is a method to determine relative distances between satellites. Radio signals are transmitted from one satellite to another. The time of flight of the radio signal is measured and with it the relative distance between sender and receiver can be computed. No absolute position with respect to Earth and also no three-dimensional state vector can be obtained, though. On the other hand, radio-based ranging can provide relative distances with very high accuracies in the order of 10 cm (cf. e.g. Tang et al. (2018)). The accuracy correlates with the time of flight measurement, so highly accurate clocks/oscillators on-board the satellites are required. In addition, active transmitters and receivers have to be implemented on the involved satellites. One specific approach is the so-called Two-Way Ranging where both receivers and transmitters are integrated on each satellite and the ranging is performed in a way that the transmitted signal from one satellite is received by a second satellite and directly sent back to the first satellite. So two measurements are performed in parallel and errors induced by offsets between the clocks on the two satellites (which is one main error source) are compensated. Other main error sources are temperature, solar influence and Doppler shift among the satellites. The last one can be reduced by partial correlation, whereas influence of temperature and sun can be reduced by avoiding orientations where the antennas are facing sun or earth (Alawieh et al., 2016).
Radio-based ranging has been used in few space missions so far, e.g. the Chinese BeiDou constellation where it is included to improve the accuracy of the GNSS measurement by gaining more accurate knowledge of the satellites' positions (combined with ground-based ranging as absolute reference). The satellites of the latest generation are equipped with Ka-band phased array antennas capable of beam forming. Radio-based ranging has been evaluated showing accuracies of approximately 10 cm (Tang et al., 2018).
For CubeSats radio-based ranging has been studied within the GAMALINK project[1] (Alawieh et al., 2016). However to the knowledge of the author, radio-based ranging has not been used on CubeSats so far, nor are market-ready products available.

[1] `http://gamalink.tekever.com/`

Radio-based ranging is a new method that can theoretically provide high precision relative distance measurements among satellites. Up to now, this technology is not ready for small satellites, especially CubeSats, but shows potential for application in the near future as miniaturization advances.

4.2.3 Inter-Satellite Link

The Inter-Satellite Link (ISL) is included as sensor system for SFF here as well, because it indirectly provides navigation information that is required for SFF. An ISL is any direct communication link between satellites. Since in most SFF missions, especially in small satellite missions, the satellites are equipped with GNSS sensors for position/velocity determination, ISL is required to exchange such position/velocity information, so that the satellites are able to compute relative state vectors. These are required for e.g. forming the EH frame or computing ROEs and to perform formation control. Because of that, ISL is a crucial technology for SFF as long as there are no relative state measurements available. ISLs are commonly based on radio communication e.g. in UHF or S-band, but also optical links are possible. No matter which medium is used, ISLs are always limited in range (e.g. in the order of 100 km for UHF) and thus limit the possible formation baseline. However, thrusting capabilities to maintain such formation baselines are commonly more limiting. In addition, ISLs need to guarantee a regular position/velocity state exchange with at least the frequency that is demanded by the mission and formation control requirements (e.g. once per orbit for looser formations up to once per second for close range high precision formations). Of course, both range of the ISL as well as communication frequency are limited by the available power on-board the satellite.
In NetSat an UHF transceiver with omnidirectional antenna (cf. Fig. 4.10) is implemented in each satellite and used as ISL. As transceiver a NanoCom AX100 from GOMSPACE A/S, Aalborg, Denmark is used (GomSpace A/S, 2019) wheras the omnidirectional redundant crossed dipole antenna is an in-house development. In first in-orbit experiments in NetSat the ISL proved its functionality and exceeded all expectations by being capable of bridging distances of up to 1000 km, whereas only distances of up to 100 km were foreseen.
ISLs are no sensor systems for SFF in the classical sense, but are a critical technology for navigation in SFF, because relative position/velocity states can typically only be generated with the help of ISLs.

4.3 Actuator Systems for Satellite Formation Flying

Actuator systems for SFF span from attitude actuators like reaction wheels and magnetorquers to propulsion systems like impulsive or CLT thrusters. There are typical performance requirements that apply to all kinds of actuators like the accuracy i.e. the quality of the achieved attitude or position compared to the commanded state. Another typical performance requirement is the range in which the actuator can perform it's actions (and in which the accuracy requirements are met). In

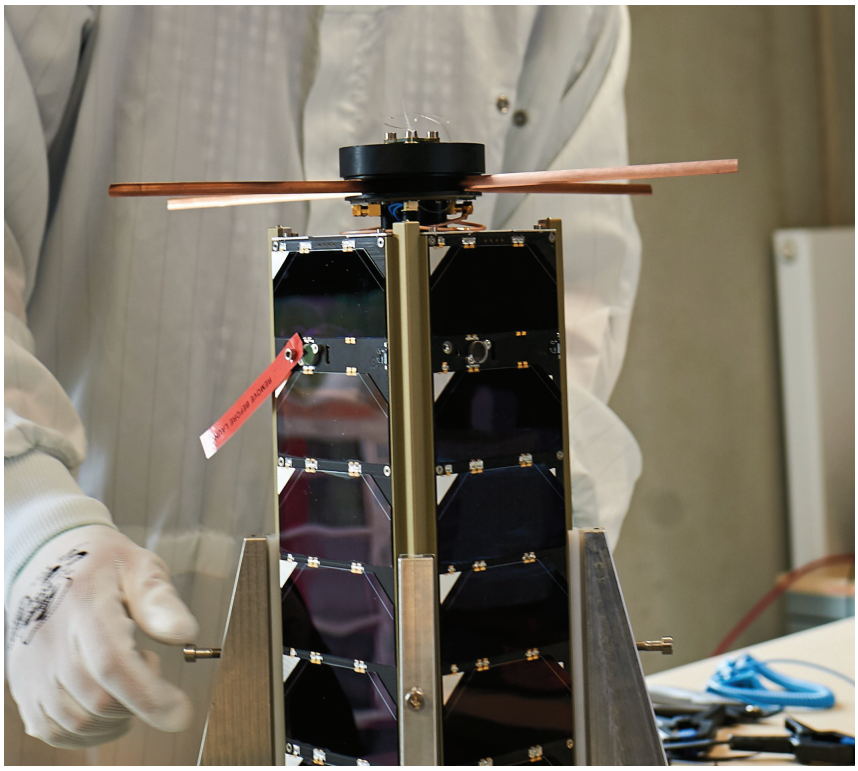


Figure 4.10: A NetSat satellite with deployed redundant crossed dipole antennas in UHF band. (Image courtesy: Zentrum für Telematik e.V.)

addition, there are few performance requirements that apply to attitude actuators or propulsion systems only and which are mentioned there.

4.3.1 Attitude Actuators

The selection of appropriate attitude actuators depends on the mission requirements like accuracy (see Chap. 4.1.1), available mass, size, electrical power as well as the mission goal and thus is highly mission-dependent. On the other hand, the selection of attitude actuators affects the performance of the mission and its schedule. For the NetSat mission e.g. the selected set of attitude actuators (detailed in the next sections) allows for attitude control with an accuracy of up to 1°, but limits this to two axes during eclipse time due to the unavailability of sun sensors in this phase. This is critical, since SFF in 3D is envisaged within the mission. Thus thrust maneuvers have to be planned in sun time. Further, since there is only one thruster on-board of most small satellites, the attitude actuators within the Attitude Control System (ACS) are required to change the satellite's direction during thrust maneuvers. E.g. for an arbitrary change of attitude between two different thrust vectors on a NetSat satellite up to 135 s are needed (Savran, 2020, p. 43).
In the following sections, typical attitude sensors are introduced in detail. Thrusters can also be used for attitude control, though this is only feasible for larger satellites

which are able to hold multiple thrusters in multiple directions and thus are not included here.

4.3.1.1 Magnetorquer

A magnetorquer is an electromagnet that is activated within Earth's magnetic field. Thus, a torque is created to rotate the electromagnet with its electric field lines antiparallel to Earth's magnetic field lines (similar to a magnetic compass). The torque generated by the magnetic moment $\vec{\mu}$ of an electromagnet is defined as

$$\vec{T}_{mag} = \vec{\mu} \times \vec{B}_{Earth} \tag{4.1}$$

where $\vec{B}_{Earth}$ is the external magnetic field of Earth at that point. Since $\vec{a} \times \vec{b} = 0 \iff \vec{a} \parallel \vec{b}$, only torques perpendicular to $\vec{B}_{Earth}$ can be generated. Thus, detumbling and spin stabilization, but no three-axis stabilized control is possible. The rotation along Earth's magnetic field lines remains arbitrary. In three-axis stabilized satellites, e.g. as required for SFF missions, magnetorquers are commonly combined with other attitude actuators like reaction wheels or momentum exchange devices in general. However, they are no single solution either, because their stored momentum needs to be unloaded (so-called desaturation) from time to time. This can be handled by magnetorquers. In satellites, also small satellites and CubeSats, commonly a set of three magnetorquers are implemented (perpendicular to each other). There are two typical designs of magnetorquers, either as air coil or as rod with magnetic core. Rods can provide higher torques than air coils, but on the other hand create a residual dipole moment which may affect the satellite's attitude stability, when no attitude control is active.

For CubeSats e.g. magnetorquer of both types are available providing magnetic moments of $0.1\,\mathrm{A\,m^2}$ or higher at nominal power consumptions of 1 W or lower with masses below 100 g. Rods typically show residual dipole moments of $1\,\mathrm{mA\,m^2}$ (cf. e.g. CubeSpace (2016) and ISIS - Innovative Solutions In Space B.V. (2013)). In NetSat a set of in-house developed magnetic air coils is used with a magnetic moment of $0.06\,\mathrm{A\,m^2}$ at a nominal power consumption of 300 mW and a mass of 25 g (cf. Fig. 4.11). The magnetic coils are placed on the inner side of each 3U side panel and at the inner side of the z-panel (below the UHF antenna). So they are redundant along two axes.

To verify magnetorquers and magnetorquer-based attitude control an external controlled magnetic field needs to be provided, preferably in three dimensions. This can be realized with the help of a set of Helmholtz coils as described in Chap. 4.2.1.2 like e.g. the ESA three-axis Magnetic Coil Facility shown in Fig. 4.3. In addition, an air bearing platform is necessary to allow the tested satellite/system to perform rotations. Sophisticated testbeds combine Helmholtz coils with an air bearing table such that magnetic attitude control can be tested on ground. One example is the ADCS Test Facility at the Hawaii Space Flight Laboratory, Honolulu, U.S.A. shown in Fig. 4.12.

Figure 4.11: Inner side of the NetSat 3U side panel holding the magnetorquer. (Image courtesy of the author.)

Magnetorquers are a common type of attitude actuators and part of many satellite missions, especially also small satellites. They can be used for detumbling and stabilization (around two axes). For three axis attitude control magnetorquers are commonly combined with other acutators like reaction wheels.

4.3.1.2 Reaction Wheels

Reaction wheels are typical momentum exchange devices and most common among small satellites, because other types are mostly too large and heavy for small satellites. Reaction wheels are electric motors placed on-board a satellite with a fixed spin axis. If the motor is activated, a torque is generated and the rotor in the motor starts turning. Because of conservation of angular momentum, the inverse torque that acts on the rotor acts on the satellite body and thus the satellite starts spinning in the opposite direction of the rotor. By placing three reaction wheels perpendicular to each other within a satellite, the satellite can be controlled or stabilized in three axes. Besides, since reaction wheels are commonly controllable with high accuracy, they are suitable for fine-pointing as well as slew maneuvers. On the other hand, reaction wheels can cause – due to imbalances in the electrical motor – noise and vibrations within the satellite. In addition, as every momentum exchange device, reaction wheels need another attitude actuator (e.g. magnetorquer) to unload (or desaturate) their stored momentum from time to time. Typical reaction wheels

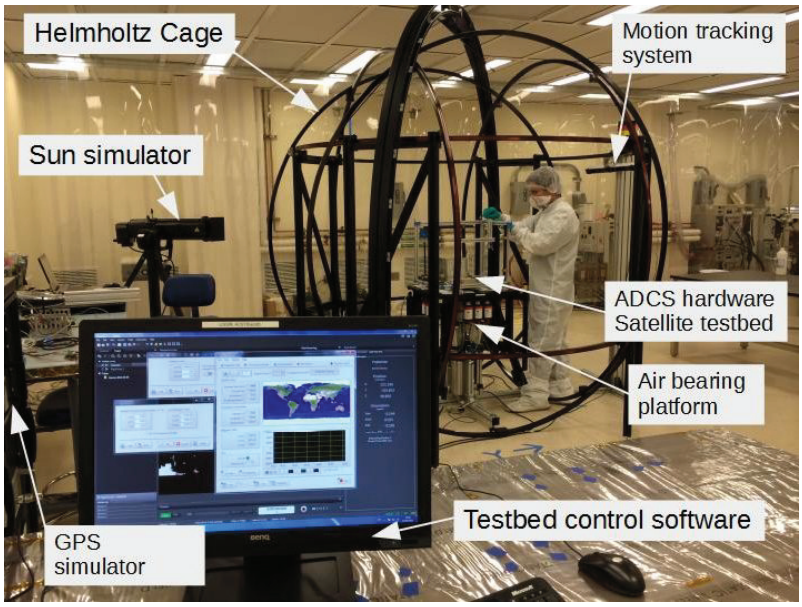


Figure 4.12: ADCS Test Facility at the Hawaii Space Flight Laboratory, Honolulu, U.S.A consisting of three perpendicularly arranged Helmholtz coils and an air bearing where the satellite/system under evaluation is placed on. (Image courtesy: Hawaii Space Flight Laboratory)

(especially for CubeSats) are capable for rotation speeds of 5000 min^{-1} or higher and are available in various sizes and torques, e.g. in the order of 2 mN m at 200 mW power consumption and 100 g mass (cf. e.g. CubeSpace (2020)).

In the NetSat mission six cyber reaction wheels, a joined development of WITTENSTEIN cyber motor GmbH and Zentrum für Telematik e.V., are integrated in each satellite, two of them in each spatial direction for redundancy reasons. Fig. 4.13 shows the assembly and placement of the reaction wheels on a PCB in NetSat. The reaction wheels can be driven with up to 20 000 min^{-1} rotational speed and provide a torque of 0.1 mN m at 300 mW power consumption within a volume of 20 mm × 20 mm × 20 mm and 30 g mass (WITTENSTEIN cyber motor GmbH, 2020). For desaturation magnetorquers are used within NetSat.

To verify attitude control with reaction wheels an air bearing setup as described in Chap. 4.3.1.1 (see Fig. 4.12 for an example) and which is also required for magnetorquer evaluation can be used. Thus, attitude control based on reaction wheels can be performed and the satellite can rotate (nearly) freely on the air bearing. If combined with Helmholtz coils, the combination of reaction wheels, magnetorquers, magnetometers and IMUs can also be tested.

Reaction wheels are highly precise actuators and are especially suitable for CubeSats due to their availability in small sizes and with low power consumption. They are integrated in missions, where precise three axis attitude control is necessary, e.g. for Earth observation or SFF. As every momentum exchange device, reaction

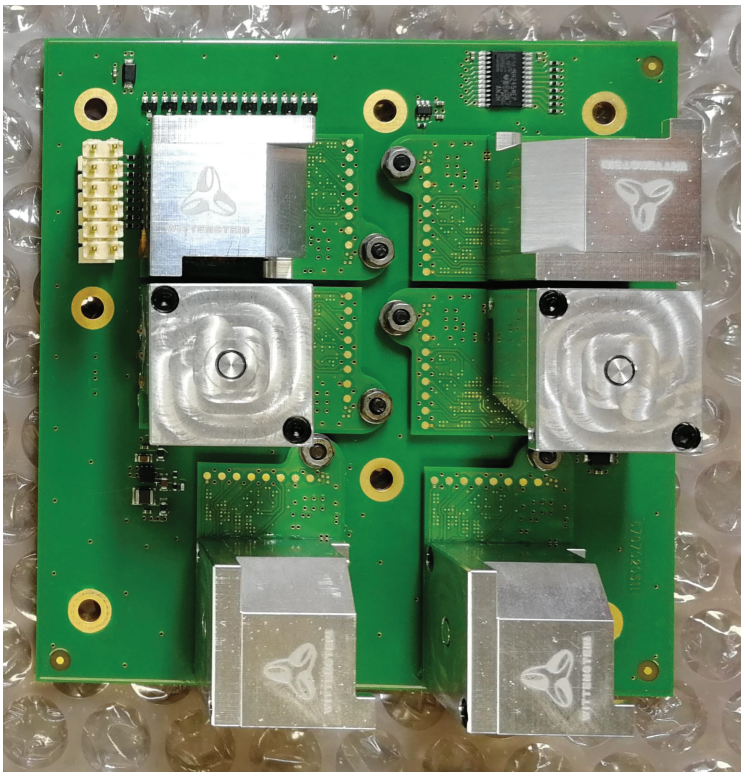


Figure 4.13: Set of six cyber reaction wheels from WITTENSTEIN cyber motor GmbH used within the NetSat mission. (Image courtesy: Zentrum für Telematik e.V.)

wheels need other attitude actuators (like magnetometers) for momentum unloading (desaturation). They are commonly implemented together with them.

4.3.2 Propulsion Systems

Propulsion systems are – besides attitude actuators – the main actuator system for satellites. Especially for SFF missions like NetSat they are essential, since they allow for changing the satellite's position in space. There are different types of propulsion systems. For small satellites the availability is limited, still various models of different types are on the market (cf. Yost and Weston (2020, pp. 47ff.)). Most of the thrusters available for small satellites can be categorized into two types, though. Either they fall under the category of impulsive thrust systems or Continuous Low-Thrust systems. Both show advantages and disadvantages which will be detailed in the according sections. The choice mainly depends on the mission's requirements. One of the most critical requirements is the intended mission life time, which typically ranges from one to three years after LEOP for small satellites. Some commercial missions though require up to five years. The life time of a SFF mission depends on the required propellant within the propulsion systems of the satellites or more precisely

on the total ΔV the propulsion systems can provide. From the mission life time, the planned formation topologies (cf. Chap. 2.6) and formation baselines arises the required total ΔV for the mission. Another requirement is the position accuracy that is demanded while maintaining a target formation. This may be in the order of few hundred meters, but in specific applications it may go down to few meters or even centimeters. Based on these requirements the most suitable propulsion system can be selected.

4.3.2.1 Impulsive Thrust Systems

Impulsive thrust systems are defined as propulsion systems that can provide high thrusts in short time and thus can perform impulsive orbit or formation maneuvers. Among those, most common are chemical thrusters, which generate thrust by emitting a gas or a mixture of gases. In small satellite missions there is typically one thruster integrated per satellite, if orbit or formation maneuvering capability is required. Impulsive thrusters are commonly easy to operate i.e. usually do not require a heat-up phase or other long-lasting preparation and show low power demands. However, they are commonly less precise in the thrust that they can provide (i.e. the minimal thrust bit that can be generated is typically significantly higher than for CLT systems). In addition, the total ΔV that can be provided by an impulsive thrust system is typically lower than for CLT systems. On the other hand, the maximum thrust that can be generated is about three orders of magnitude higher. So in contrast to CLT systems impulsive thrust systems are stronger and easier to operate, but on the other hand less precise and less prolonged. Typical impulsive thrusters for small satellites and especially CubeSats can provide a maximum thrust in the order of 100 mN to 1000 mN with a power consumption in the order of 10 W (Yost and Weston, 2020, p. 59ff.).
In NetSat no impulsive thrust system is used, however for the TOM mission the CubeDrive chemical thruster of Dawn Aerospace Ltd., Delft, Netherlands (depicted in Fig. 4.14) is designated. It uses Nitrous oxide (N2O) and Propene (C3H6) as propellants and can provide up to 1000 mN thrust (at 30 °C) and a total ΔV of about 100 m s^{-1} for a typical 3U CubeSat. The minimum impulse bit, which can be generated, is 35 mN s, whereas typical power consumption during firing is 12.5 W (Dawn Aerospace Ltd., 2020).
Impulsive thrust systems show several advantages regarding control. Since there is no significant preparation or heat-up time necessary, control algorithms do not need to take long-term planning into account (e.g. as MPC would be capable of) and thus less complex methods may be used. In addition, they are capable of performing comparably high thrusts, so no strict constraints have to be implemented within the controller which further eases its implementation. Considering the control approaches presented in Chap. 3, e.g. *MPC-Based Formation Control* is covering the needs of a CLT propulsion system, but may not be required for an impulsive system (though still showing benefits). On the other hand, *Distributed Robust Consensus-Based Control* is not capable of maintaining hard constrains or providing long-term planning, but may be perfectly suitable for impulsive thrust systems.

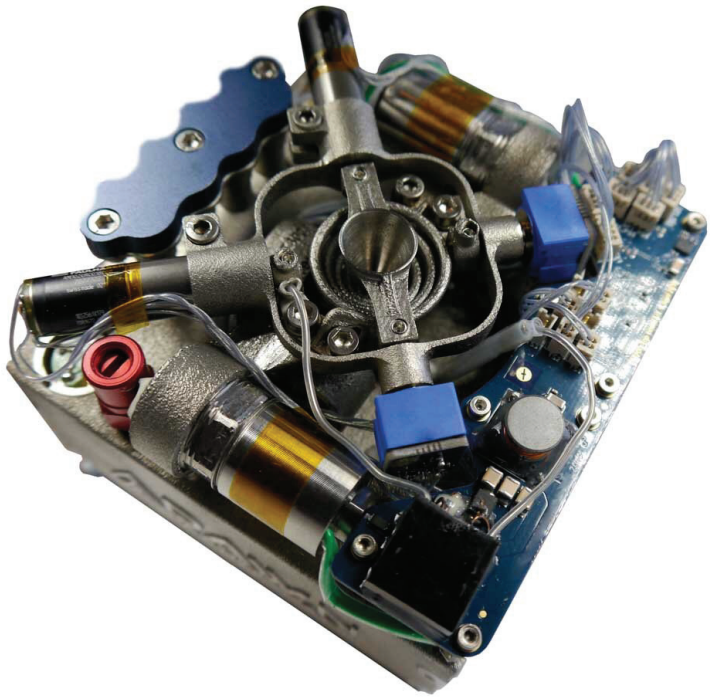


Figure 4.14: CubeDrive chemical propulsion system to be used in TOM. (Image courtesy: Dawn Aerospace Ltd.)

Testing of propulsion systems for satellites on ground is complicated in most cases, because most of the thrusters can only be operated properly in a vacuum environment. In addition, a thrust gauge is required to measure the generated thrust. The combination of both setups is very sophisticated and only few test facilities exist, e.g. at manufacturers. One facility is located at DLR Institute of Aerodynamics and Flow Technology in Göttingen, Germany (cf. Fig. 4.15). However testing procedures are complicated and the usage is costly, so testing at satellite integrators is mainly limited to the electrical interface of the thruster or its characteristics in vacuum without verification of the thrust characteristics.

In summary, impulsive thrust systems are one of the most common propulsion systems for small satellites, especially CubeSats, and show several advantages like ease of use and full maneuverability with high thrusts. In addition, impulsive systems allow for less complex control algorithms and thus simplify the development process. On the other hand, the available total ΔV is limited in comparison to CLT systems and thus mission with high requirements on life time or formation baselines may not be feasible.

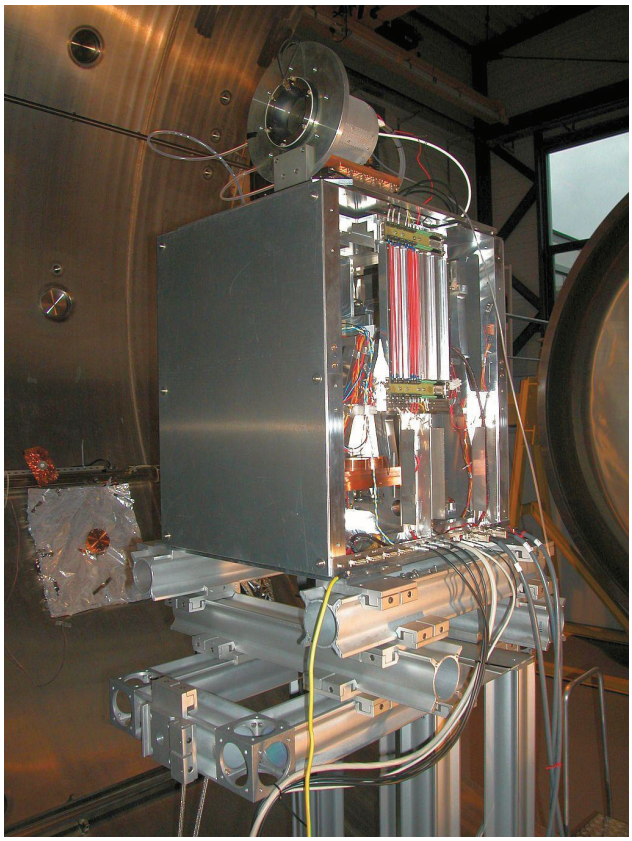


Figure 4.15: Thrust measurement device with mounted thruster at DLR Institute of Aerodynamics and Flow Technology. (Image courtesy: DLR)

4.3.2.2 CLT Systems

Continuous Low-Thrust (CLT) systems are characterized by their ability to generate thrust with a low amplitude over long durations (e.g. minutes or hours instead of seconds). However, this feature may be judged as a disadvantage, since most CLT systems are only capable of providing low thrust (i.e. in the order of mN) in contrast to other propulsion systems which are able to generate thrust with two to three orders of magnitude higher (e.g. chemical propulsion systems). The most common CLT systems in small satellites and CubeSats are electric propulsion systems like Field-Emission Electric Propulsions (FEEPs), which use an electric field to accelerate charged ions that are extracted from a heated liquidized metal like Gallium or Indium. Typically one thruster is placed per satellite, however as shown in the UWE-4 mission also multiple miniaturized thrusters (NanoFEEPs by Morpheus Space GmbH, Dresden, Germany) can be integrated (Kramer et al., 2020). Thus, thrust pointing or even attitude control is feasible. However, the available propellant mass in such a setup may be reduced and thus the total ΔV is limited. Electric propulsion systems usually require a heat-up phase to liquidize the propellant (e.g. in the order of 10 min to 30 min), so that maneuvers have to

be planned in advance. This has to be taken into account by the controller or an overlaying scheduler has to be implemented to cover the required planning. On the other hand, electric propulsion systems are capable of high specific impulses and thus can provide high amounts of total ΔV, e.g. up to $1000\,\mathrm{m\,s^{-1}}$ for a typical 3U CubeSat depending on the chosen thrust. (The ΔV efficiency is indirectly proportional to the thrust magnitude.) In addition, since very low thrust is generated (typically in the order of 50 µN to 500 µN), CLT systems can operate very precise maneuvers like high precision station-keeping in a formation. However, for larger orbit or formation changes a significant amount of time is required (e.g. in the order of weeks, cf. Chap. 5 where multiple formation flying scenarios are simulated). In addition, as thrust is generated by accelerating charged particles with the help of a high-voltage electric field, electric propulsion features significant power demands in the order of 10 W to 100 W during firing for typical CubeSat systems (Yost and Weston, 2020, p. 64ff.).

In NetSat a FEEP thruster, namely the IFM NANO Thruster by ENPULSION GmbH, Wiener Neustadt, Austria as depicted in Fig. 4.16 is implemented. It uses Indium as propellant and can provide up to 350 µN of thrust. If a thrust of 150 µN on average is used instead, a total ΔV of about $400\,\mathrm{m\,s^{-1}}$ can be reached for a typical 3U CubeSat. To operate the thruster, up to 40 W of power are required (ENPULSION GmbH, 2018).

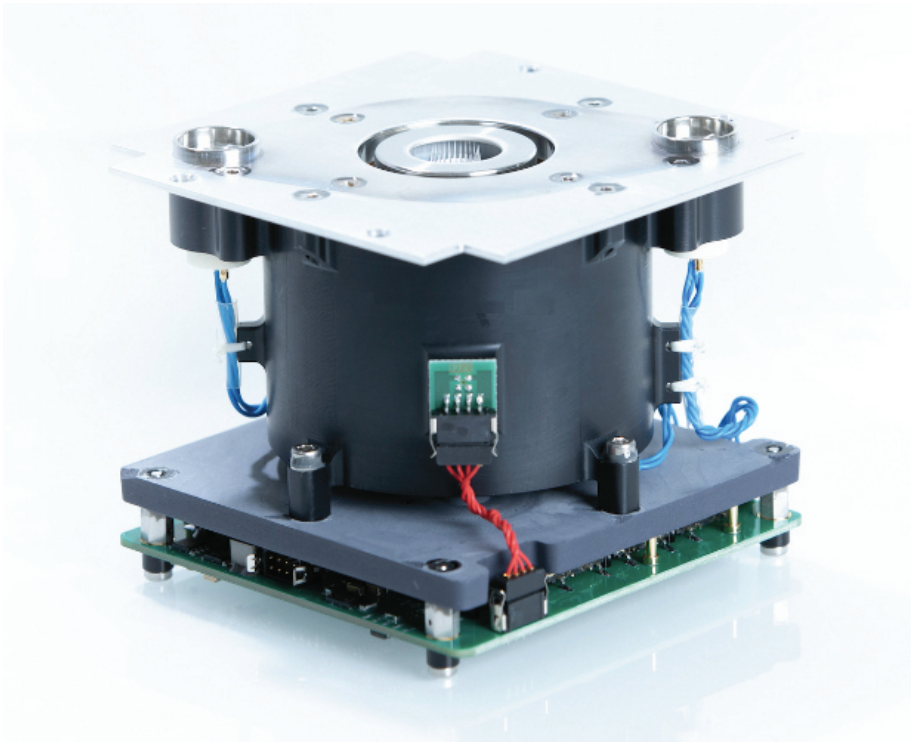


Figure 4.16: IFM NanoThruster electric propulsion system to be used in NetSat. (Image courtesy: ENPULSION GmbH)

CLT systems for small satellites and CubeSats commonly require significant preparation or heat-up time before being able to fire as well as significant power to be available (i.e. batteries to be fully charged), so long-term planning is necessary to start the preparation in advance to the planned operation. This has to be taken into account in the controller (as possible within e.g. a MPC) or a superordinated planner has to be implemented. Further, since only very low thrusts are possible, the controller has to be able to enforce hard constraints on the maximum thrust. The low thrusts lead also to long maneuver times (e.g. for typical formation flying scenarios in the order of several days or weeks as described in Chap. 5). Thus, optimal control methods are of particular advantage, because the limited thrust can be applied in an optimal way and further delays are avoided. On the other hand, CLT systems are capable of converging with very high accuracy to the target state, because of their ability to provide very small thrust bits and thus are especially suitable for high-precision control. CLT systems impose a lot of requirements on control and thus require sophisticated control approaches like the *MPC-Based Formation Control* presented in Chap. 3.2. Simpler control approaches may not be able to cover all requirements.
Testing CLT systems has the same requirements and implications as testing other thrust systems as described in Chap. 4.3.2.1. In addition, typical electric propulsion systems show even higher demands towards the vacuum environment, typically requiring a density of 1×10^{-5} mbar or lower. Such test facilities that allow electric thrusters to be fired or even their force to be measured, are rare and their usage is expensive. So testing at satellite integrators' facilities is limited to the electrical interface of the thruster.
In contrast to impulsive thrusters, CLT systems like electric propulsion systems are using fuel highly efficiently, thus providing very high amounts of total ΔV with a limited propellant mass. On the other hand, they are only capable of providing low magnitudes of thrust which may allow for high precision maneuvering, but on the other hand causes larger orbit/formation changes to last long durations. In addition, their complexity in operation (i.e. heat-up phase) and their high demands on available power require long-term planning and cause controllers to be more complex. Their main advantages are for missions which require long life times or large formation baselines, where a very high amount of ΔV is essential.

4.4 Forming an Overall System

To form a functioning and comprehensive overall system for SFF, several sensor and actuators are required. For SFF applications in general, 3-axis attitude determination is necessary, which requires appropriate sensor systems. Magnetometers alone are not sufficient, since they only provide attitude determination in two axes (perpendicular to the Earth's magnetic field lines). IMUs provide 3-axis attitude measurements in principle, but they do not suffice alone, since their accuracy decreases significantly over time (in the order of minutes). Sun sensors can provide 3-axis attitude, but are not functioning during eclipse. Usually a combination of

these sensors is selected to provide 3-axis attitude determination in small satellite missions with a good trade-off of costs, mass, volume, power consumption and accuracy. If neither financial nor volume/mass/power budgets are limited, star sensors are a superior solution, since they can provide attitude information continuously and with high accuracy (significantly better than 1° depending on chosen model). In addition, attitude control in three axes is required. Magnetorquer can only provide attitude control in two axes (perpendicular to the Earth's magnetic field lines). The rotation around the magnetic field lines remains uncontrolled. Reaction wheels can be added to complement this to achieve 3-axis attitude control. On the other hand, reaction wheels alone cannot serve to provide 3-axis attitude control, since they need another attitude actuator (e.g. magnetorquer or thruster for larger satellites) to desaturate them. Usually a combination of magnetorquer and reaction wheels is optimal for small satellites. Depending on the position accuracy requirements and planned formation baseline, different navigation sensors can be chosen. For most small satellite applications GNSS sensors are a suitable choice because of their small size, high accuracy and the fact that they can be used for almost arbitrary formation baselines. If formation flying in 3D is desired (as it is e.g. in the NetSat mission), active propulsion systems are mandatory. Both impulsive/chemical as well as CLT/electric thrusters show advantages and are fitting most of the small satellite formation flying missions. The particular mission requirements decide which system to prefer.

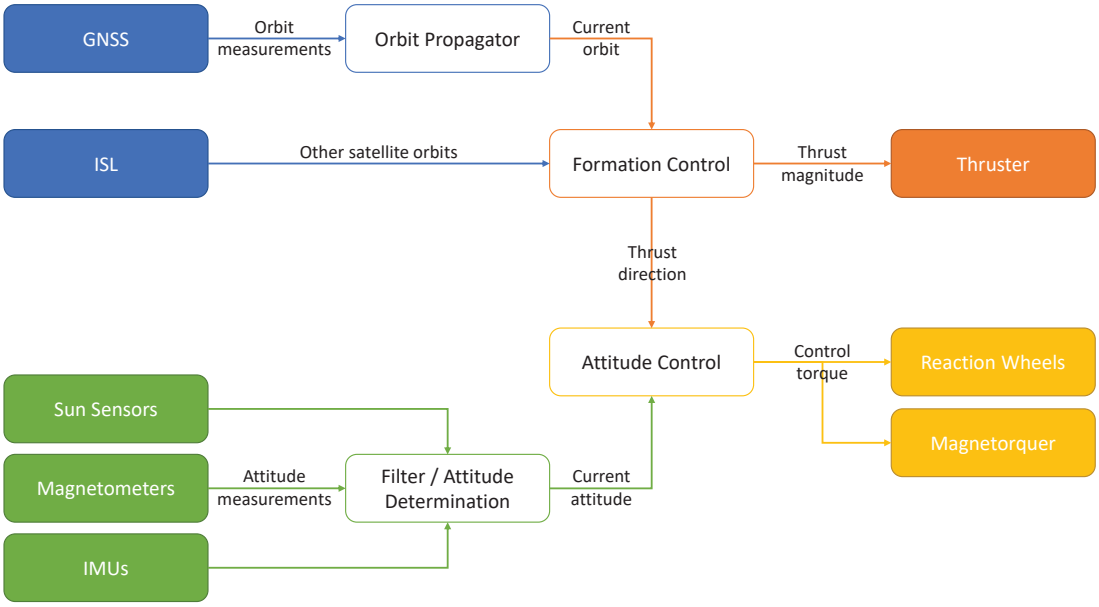


Figure 4.17: Interconnection scheme of sensors, actuators and software modules onboard a SFF mission as designed for the NetSat mission.

Thus, a typical overall system for small satellite formation flying in 3D may consist of magnetometers, IMUs, sun sensors and GNSS on the sensing side and magnetorquers,

reaction wheels and a propulsion system on the acting side. This is the chosen setup for the NetSat satellites. Fig. 4.17 presents a way how the individual sensors, actuators and internal software modules on-board a formation flying mission like NetSat interact.

4.5 Impacts on Navigation and Control for Satellite Formation Flying

In this chapter, a set of sensors and actuators has been defined that is required for a functioning and comprehensive overall system for SFF mission. The presented examples based on the NetSat mission (and typical sensor and actuator systems in general) show characteristics and limitations, which impact on the navigation and control capabilities of a SFF mission. Thus, they should act as a basis for any realistic simulation of satellite formation control. Since NetSat is chosen as primary application scenario within this work, requirements are defined on the NetSat mission and its sensor and actuator systems. The following devices are implemented:

- The satellites are equipped with an IFM NANO Thruster that is intended to be driven at maximum 350 µN. To achieve a good trade-off between thrust, energy and available ΔV a thrust level of 150 µN is preferred. With a given satellite mass of 3.9 kg, a maximum acceleration of $3.85 \times 10^{-5}\,\mathrm{m\,s^{-2}}$ is achieved.
- With the above given maximum thrust, the total ΔV per satellite is in the order of $400\,\mathrm{m\,s^{-1}}$. This is a trade-off between maximum thrust and total available ΔV.
- Since there is only one thruster on-board each NetSat satellite, a minimum time to change the thrust direction is required. For an arbitrary change of attitude between two different thrust vectors up to 135 s are needed (Savran, 2020, p. 43). This rotation can be performed by the ACS.
- The state exchange of the satellites within the formation depends on the ISL. Thus, the state exchange is limited to – in the worst case – once per orbit, even if regular exchanges once per minute are foreseen.
- Attitude determination can be performed with an accuracy of up to 1° and is limited to two axes during eclipse time due to the unavailability of sun sensors in this phase. Thus, thrust maneuvers have to be planned in sun time.
- Relative distance (understood as minimum distance between two satellites within a whole orbit time) should not be below 1 km for collision safety and should not exceed 100 km to be within the range of the ISL.

Chapter 5

Evaluation

5.1 Overview

In this chapter, the developed control methods are evaluated. First, the Orekit-based Formation Simulation Framework (FSF) is introduced, which has been developed at ZfT and also partly within this work. It is used later on to perform the simulations to evaluate the controllers. Then, the underlying scenarios based on the NetSat mission as example are presented. Next, the software simulations that have been performed using these scenarios are described and interpreted with respect to controller performance of the developed control methods. Last, the preparation steps that have been performed for in-orbit testing within the NetSat mission are outlined.

5.2 Simulation Framework

To simulate the satellite dynamics and control in general as well as to evaluate the developed controllers in particular, a comprehensive software framework, namely the Formation Simulation Framework (FSF) has been developed. It features (perturbed) orbit propagation, formation state vector design, formation-level trajectory planning, propagator and controller evaluation, performance benchmarking and an extensible debug interface. The framework, which is written in Java, features a fully configurable satellite model and extends the well-known open-source framework Orbit Extrapolation Kit (Orekit)[1] for propagating the dynamic states of the satellites. Orekit is a software simulation framework for satellite orbit propagation. It consists of different dynamic models and orbit propagators including a high precision orbit propagator. Further, it contains an elaborate satellite model as well as environment models. Thus Orekit was chosen as base for a comprehensive satellite formation simulation framework, which is required to evaluate the developed controller. While Orekit is a suitable base for the developed FSF, it had to be adapted for formation simulation, since it is designed only for single satellite simulations. This affects the function principle of the orbit propagation within the framework. (Some efforts have been made towards SFF in recent updates, but are still limited in functionality.) In addition, the environment model has to be configured and a more extensive satellite model has to be included to encounter for CubeSats and according characteristics like CLT electric propulsion as to be used in the considered applications like e.g. the NetSat mission. The first section in this chapter introduces the main workflow of the simulation software, whereas the following sections go into further detail about the satellite model and the orbit propagator.

5.2.1 Overview

The FSF software uses configuration files in CSV-format (Comma-Separated Values (CSV)). From these configuration files, dynamic databases are created at program start-up, one for the satellite model and one for the orbital dynamics simulation. The satellite model database stores initial information about all active subsystems and is also used to exchange up-to-date data between the subsystems during orbit propagation. When a whole formation is propagated, each satellite has its own database. This means, every spacecraft is treated individually and inter-satellite communication and data exchange may be modeled at different levels of complexity, independently of the current satellite configuration. After initialization, the orbits of all satellites are instantiated simultaneously with Orekit (OREKIT, 2016).

Fig. 5.1 depicts the whole execution process of the developed FSF software. During the propagation process, the different satellite subsystems execute their procedures at their respective frequencies, e.g. the power system can be simulated with lower frequency than the ACS. This part is illuminated in detail in Chap. 5.2.3. Every

[1] `https://www.orekit.org/`

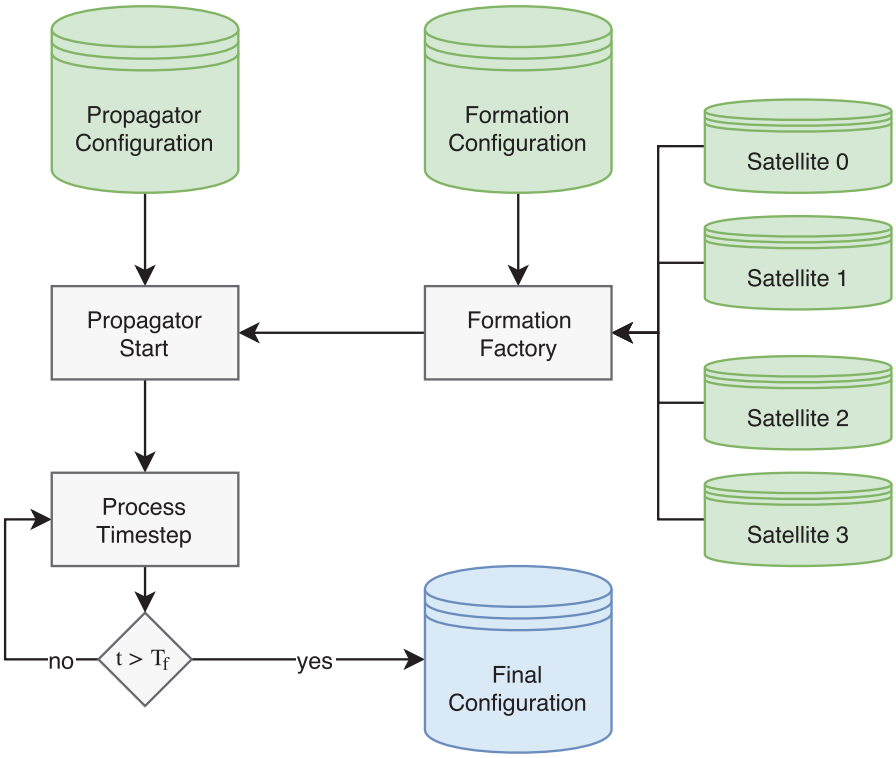


Figure 5.1: Data flow and execution process of the FSF software framework. Input data (form the configuration databases) is depicted in green, output in blue.

piece of information in the satellite databases and the current relative and absolute orbits may be viewed in live-updated MATLAB plots. The data exchange with MATLAB is realized over a network connection (based on User Datagram Protocol (UDP)), giving the possibility to plot the results on a remote machine. The software includes multiple on-board propagators and controllers of different levels of complexity. Beyond this, it provides implementations and conversion procedures for all frames mentioned in Chap. 2.4, perturbation models and a formation-level trajectory planner for all kinds of maneuvers. It features convenient formation design not only in ROEs (as used in this chapter later on), but also in various absolute and relative coordinate frames.

5.2.2 Satellite Model

The satellite model was developed with a focus on reconfigurability, while, at the same time, the computation speed of the subsystem procedures was considered to achieve a reasonable orbit propagation time. Furthermore, it was developed to provide the NetSat team with a model which can be used at all phases of mission design, independently of the focus of this work.

5.2.2.1 Subsystem Structure

The satellite model consists of the bare satellite structure and the subsystems. The sole purpose of the structure is to update the database with the current spacecraft state (like on-board time, orbit, mass, ***pv*** vector) after each propagation step and to call the procedures (so called *handle-step()* methods) of all active and waiting subsystems. A subsystem is meant to be waiting when its last procedure call is longer ago than its execution interval (measured in on-board time). A subsystem in the sense of this software model does not have to be a classical satellite subsystem, but may also be a very simple procedure which has to be executed at certain time intervals (e.g. as simple as printing debug information to the console when the satellite enters eclipse). Subsystems are created via the CSV-configuration by adding a defining structure to the table. This definition features a name, which will also be the Java class name of the subsystem, which is generated from the configuration file. The definition also includes an active flag (to easily switch off some functionality, if needed), the execution interval, a verbose flag (indicating if the subsystem is allowed to print to the console) and a unique execution position. Thus, if multiple subsystems are in waiting mode at the same time instant, the order of execution may be controlled. For example, the Attitude and Orbit Determination System (AODS) has to be executed before the formation controller, because it relies on up-to-date orbit information. Since all this information is part of the database, it may be altered during execution, e.g. activating a battery loading procedure only during non-eclipse phases or enabling housekeeping data storage only during times without ground station contact. A broad overview of the most important subsystems stated in their order of execution is given in the following list. If multiple alternatives for a specific function are stated, this may be configured through the database.

1. Attitude and Orbit Determination System (AODS)
   It retrieves ("senses") the current orbit and attitude from the propagator. This may be the exact values from the propagator (Orekit) or an extended version with an error model. The error model can be applied on absolute or relative state vectors. This can be used to evaluate on-board propagator sensitivity to input state deviations. In a later implementation, the direct Orekit input can be substituted with attitude and position sensor models.

2. Mission Handler (MH)
   It works as the major planning instance and updates the control target at configured times to different types of formations. Thereby, the whole course of a mission (like the NetSat mission) may be simulated with a single propagation run. The mission handler can react to database events like the termination of a formation change maneuver. Once triggered, it can update the new control target, change the active controller implementation or send a status update to the other satellites of the formation.

3. On-Board Propagator
   It propagates the relative state with reference to the (virtual) leader of the

formation. The implementation features various on-board propagators with increasing level of accuracy.

4. Formation Controller
   It generates control commands for the thrust system to reach a certain relative orbit. The implementation features multiple control methods for different cases like formation maintenance and formation acquisition maneuvers. The control approaches developed in Chap. 3 have been implemented here.

5. Thrust System
   It takes the control input from the controller and applies it to the thrusters, which results in a contribution to the physics of the dynamic model of the propagator. It takes thruster capabilities, power consumption and mass loss into account. It can be set to test mode to allow and forward any input to the physical propagation model (e.g. ignoring physical limitations of the thruster). In normal mode the input is capped at the maximum rate of the thrusters. Also the maximum turning rate of the satellite is obeyed, since thrusters in CubeSats are commonly only directed into one direction. This prevents an instantaneous change of the thrust vector. The thrust model includes accurate mass-loss and power consumption calculations based on the applied acceleration.

6. Inter-Satellite Link (ISL)
   It communicates with the other satellites in the formation and updates the relative state. Depending on the configuration, this can be a perfect real-time link (meaning the satellites know at all times the true relative positions and velocities of each other) or a link which updates information from time to time. The default state-exchange frequency is set to one time per orbit, but can be changed arbitrarily.

7. Power Control and Distribution Unit (PCDU)
   It loads the battery in sunshine phases, computes the estimated power consumption of the on-board computer and the power consumption of the active subsystems. It has to be the last subsystem to be executed to account for all current changes in the power consumption. The subsystem can also raise warnings on low battery levels and deactivate high consumers, which are not critical for short periods of time, like the thrust system.

Further important subsystems like housekeeping or ground station communication are not part of this work, because focus was set to autonomous formation control. This is only marginally impacted by the two systems stated above. If needed, they may easily be integrated in the future due to the expandable nature of the overall model.

5.2.2.2 Data Handling

Internal satellite data is handled and processed by a dynamic database, generated from specially formatted CSV files containing all relevant parameters. This database

is structured into tables (one for each subsystem) and contains all parameters which need to have a configurable value at start-up or which impact multiple subsystems. Thereby this database is the designated way for subsystems to exchange or forward data between each other. This renders a subsystem's implementation totally independent of other parts of the satellite, resulting in a modular model. All parameters are stamped with the time of their last update and may be locked to prevent further value changes. A single parameter can be a simple atomic value like a floating point number, an array of atomic values or a complex datatype like an orbit definition. If needed, further datatypes may be added by defining parse- and copy-methods for the type. The implementation also features event-driven interaction, i.e. the database calls an event handler in a subsystem if a certain parameter is changed. The database is directly connected to the debug interface to visualize real-time data dumps (see Chap. 5.2.4).

5.2.3 Orbit Propagation

The iterative orbit propagator handles all satellites equally (e.g. does not distinguish between leader and follower spacecraft) and propagates their absolute orbits simultaneously for each time step. It is based on the open-source orbital dynamics framework Orekit, which is a pure Java library and only depends on the Hipparchus mathematics library (Hipparchus, 2016).

5.2.3.1 Orbit Extrapolation Kit Interface

Orbit Extrapolation Kit (Orekit) provides packages for date and time handling, frame transformations, ODE integration, attitude handling and perturbation force models. However, it does not support multiple satellites[2] or relative state descriptions. The developed formation propagator is the basis of the dynamics of the FSF which sets up on Orekit, but has also access to the underlying mathematics library. The schematics of this connection are depicted in Fig. 5.2.
Developed extensions include multi-satellite support, configuration via CSV files, thrust force models and the interface to the previously mentioned satellite model. The instantaneous state of each spacecraft is, among others, defined by its osculating $\boldsymbol{pv}$ vector and its mass. The dynamics of the state are represented by ODEs obeying gravitation, perturbation forces and inertial satellite forces like thrust. To propagate the state, these equations (with their initial values comprised by the spacecraft states) are converted to match the format of the mathematical Hipparchus library. Fig. 5.3 depicts the propagation process in detail. The bold gray arrows contain the current environmental settings and all satellite state vectors and follow the main execution loop. After each time step, the mapper brings this accumulated state vector into an ODE-friendly format. Within Hipparchus, the states are integrated using a numerical integrator and then back-transformed to spaceflight dynamics to give insight into the spacecraft states at the next timestep. After the simulation time

[2]This feature has been added in recent updates, but is still too limited for our needs.

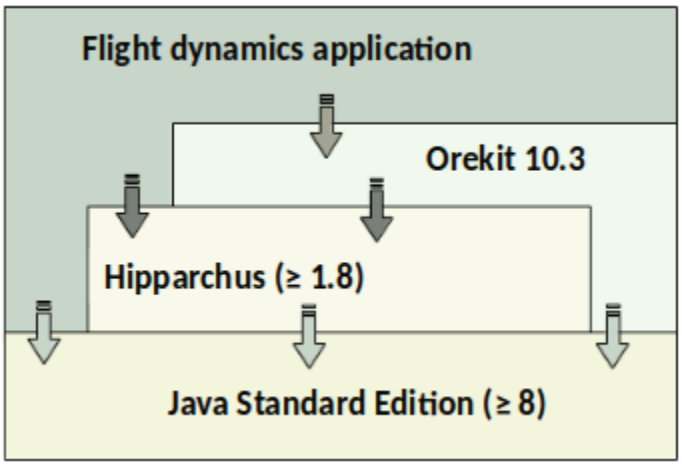


Figure 5.2: FSF-propagator (Flight dynamics application) and Orekit interaction (Image courtesy: `https://www.orekit.org/`).

has been increased, environment events (e.g. eclipse entering) and satellite-internal events are handled. Then, all waiting subsystems are executed in their respective order. Before the next propagation step begins, relevant data is sent to the debug interface.

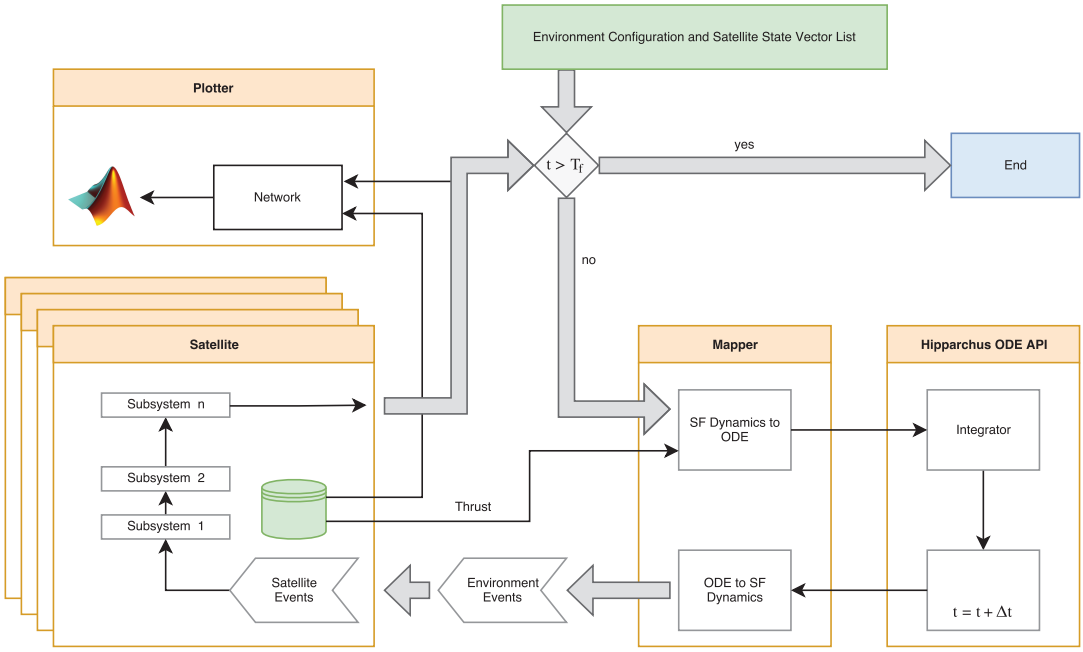


Figure 5.3: Propagation process in detail. The start is depicted in green, the end in blue.

5.2.3.2 Frame and Date Handling

The satellite states are integrated (e.g. propagated) in absolute osculating ***pv*** coordinates. The utilized state vector is therefore equal to Eq. 2.2. For the ECI frame, the J2000 (or EME2000) reference was chosen since this is prevalent in orbital dynamics and already implemented in Orekit (see Chap. 2.4.1). The J2000 frame is defined in reference to Terrestrial Time (TT). Since other timescales are also important for various data input sources or desired output formats, conversion functions from TT to International Atomic Time (Temps Atomique International) (TAI) and Universal Time Coordinated (UTC) are exposed to the framework user. TAI is very important since it is the most accurate and regular timescale available and also used as a reference for the GPS time with a constant offset to GPS reference time of 19 s. TT is also defined via a constant offset to TAI.

$$\mathrm{TT} = \mathrm{TAI} + 32.184\,\mathrm{s} \tag{5.1}$$

UTC has 86400 SI-seconds per day and is synchronized with TAI. To accommodate the slowing of the Earth's rotation, leap seconds are introduced at irregular intervals to keep the difference to TAI below 0.9 s (Montenbruck and Gill, 2012). Whenever a state vector is utilized in the framework, the reference frame and the timescale has to be provided, and both are checked against their counterparts when combining different state vectors. This ensures compatibility and exchangeability of state vectors over all parts of the FSF software. In addition to the absolute Orekit-inherent frame J2000, the software includes conversion methods between absolute and relative ***pv*** vectors (Eq. 2.15 and 2.19), Keplerian to equinoctial and non-singular orbital elements (Eq. 2.33 and 2.31), any set of absolute orbital elements to DOEs (Eq. 2.34) and ROEs (Eq. 2.35). Whenever absolute orbital elements are used, they are internally represented as EOE to overcome singularities at small eccentricities (see Chap. 2.4.2.2). Furthermore, relative Cartesian and orbital frames may be converted with Eq. 2.35. All orbital frames (both absolute and relative) may be provided in osculating or mean elements, depending on the current application or required input/output formats. The conversion is done with Eq. 2.57 and 2.66, respectively for absolute and relative conversions. Explicit formulations in terms of mean orbital elements can be found in App. B and App. C.

5.2.3.3 Integrator Evaluation

Hipparchus offers a vast variety of different ODE integrators, from a simple Euler integrator to a complex sixth-order Runge-Kutta (RK) integrator. The integrators differ in runtime and accuracy. To evaluate the best-suited integrator, multiple implementations with different integration step sizes were tested.
Fig. 5.4 shows the results in terms of accuracy and runtime on a log-log-scale for all tested integrators. For the accuracy (position deviation after four orbits), the result of the most accurate integrator with the smallest step size is used as a reference position (or truth model). The test set-up was one satellite in a LEO orbit,

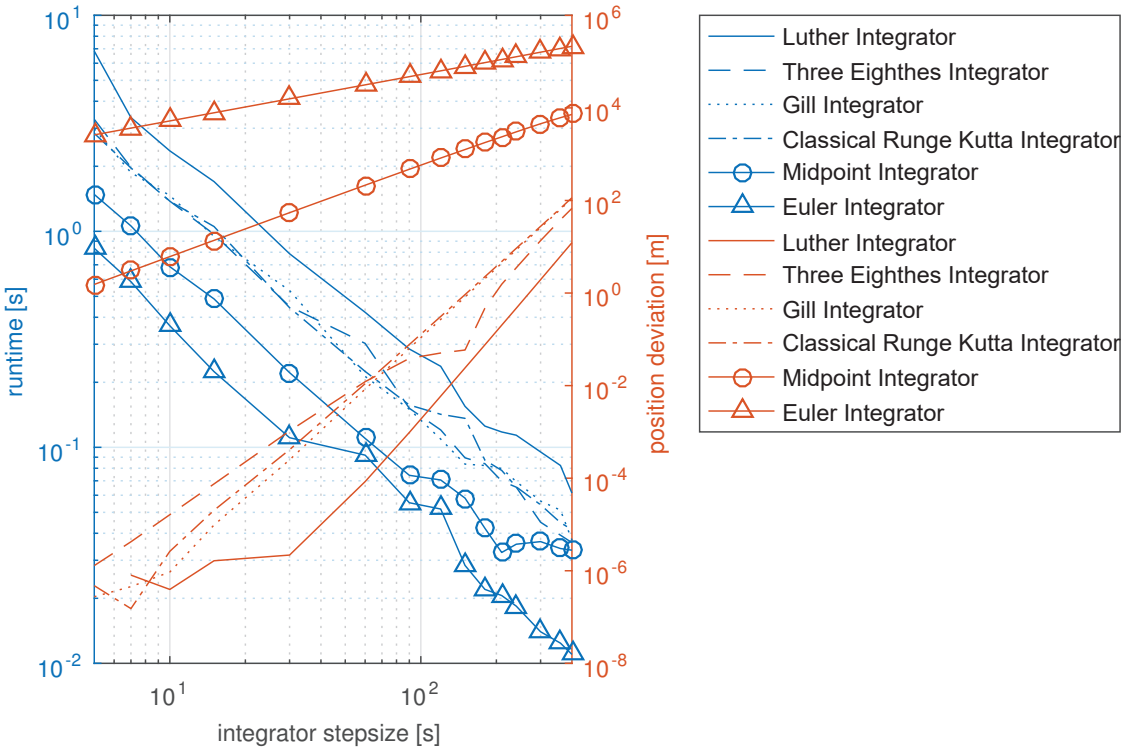


Figure 5.4: Propagator benchmark for different integrators and integration step sizes. Setup: one satellite propagated for four orbital periods on an Intel i5 machine. Computation time (runtime) on the primary ordinate, absolute accuracy (deviation of the final position obtained by most accurate integrator with smallest possible step size) on the secondary ordinate.

propagated for four Keplerian periods on an Intel i5 machine. During propagation, Earth oblateness and atmospheric drag was taken into account, while the satellite exerted no thrust. One can observe a positive exponential relationship between step size and accuracy as well as a negative exponential relationship between step size and runtime. Furthermore, to achieve an accuracy below one meter (which is the order of the desired position accuracy of the relative orbit determination system, see Tab. 4.1), Euler and Midpoint integrator disqualify even for the lowest step sizes. To analyze the suitability, however, the relative accuracy between satellites is of much higher interest than the absolute accuracy. Therefore, the exact same benchmark was conducted again with a formation of four satellites in a 5 km PCO orbit, this time with relative position deviation (with respect to the formation center) on the secondary axis of ordinates (Fig. 5.5).

Each data-point was obtained by five individual runs to achieve higher accuracy in terms of runtime. Error bars for the runtime and the position deviation are neglected in the plot in favor of higher readability. However, the dependency between integrator step size and runtime shows up more clearly compared to Fig. 5.4. As before, the two best integrators in terms of runtime do not have sufficient accuracy. Up to a step size of 60 s, all other integrators result in a deviation below one meter.

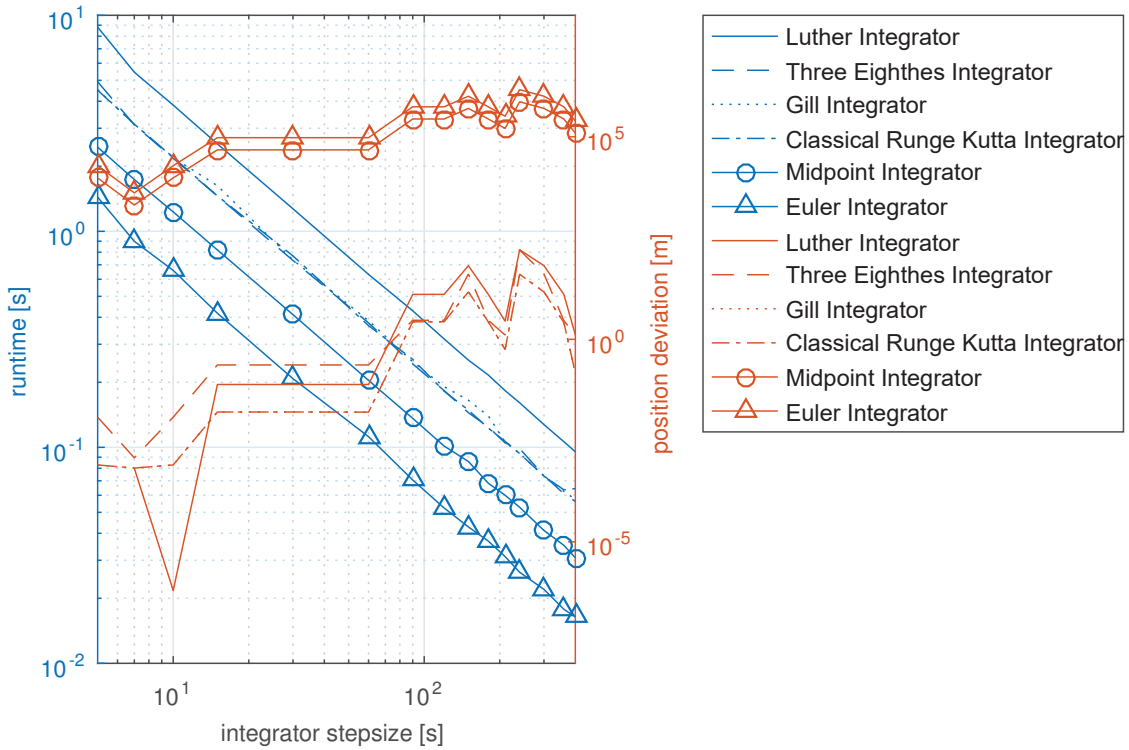


Figure 5.5: Propagator benchmark for different integrators and integration step sizes. Setup: four satellites propagated for four orbital periods on an Intel i5 machine. Computation time (runtime) on the primary ordinate, relative accuracy (accumulated deviation of the final position of the formation centre obtained by most accurate integrator with smallest possible step size) on the secondary ordinate.

A further decrease in step size (down to 13 s) does not lead to higher accuracy, while at the same time computation time rises. The sixth-order RK integrator (Luther integrator) has a 50% higher runtime compared to the three remaining integrators, which are almost identical in runtime (Three-Eight's integrator, Gill integrator, Classical fourth-order RK integrator). Out of these, the Classical RK integrator slightly outperforms the other two in terms of deviations. This leads to the decision to use the fourth-order RK integrator with a step size of 60 s during orbit propagation in the software framework. The step size is also the shortest possible interval of consecutive executions of a subsystem's step handler method. If shorter periods are required for a certain subsystem, the integrator step size may be adapted at the cost of a higher runtime. Thus, the stated 60 s define only the upper bound of the usable step size value.

5.2.3.4 Included Forces

The ODEs to propagate a satellite's state are composed of a superposition of all forces that act on the satellite

$$\boldsymbol{f}_{total} = \sum (\boldsymbol{f}_K + \boldsymbol{f}_p + \boldsymbol{f}_s) \tag{5.2}$$

where $\boldsymbol{f}_K$ is the Keplerian attraction force (Eq. 2.41), $\boldsymbol{f}_p$ are perturbing forces and $\boldsymbol{f}_s$ is the sum of all satellite-internal forces. In the current implementation, $\boldsymbol{f}_s$ only consists of the instantaneous output force of the thrusters. This could be extended to also incorporate reaction wheel forces, if they are not aligned with the main inertia axes of the satellite.

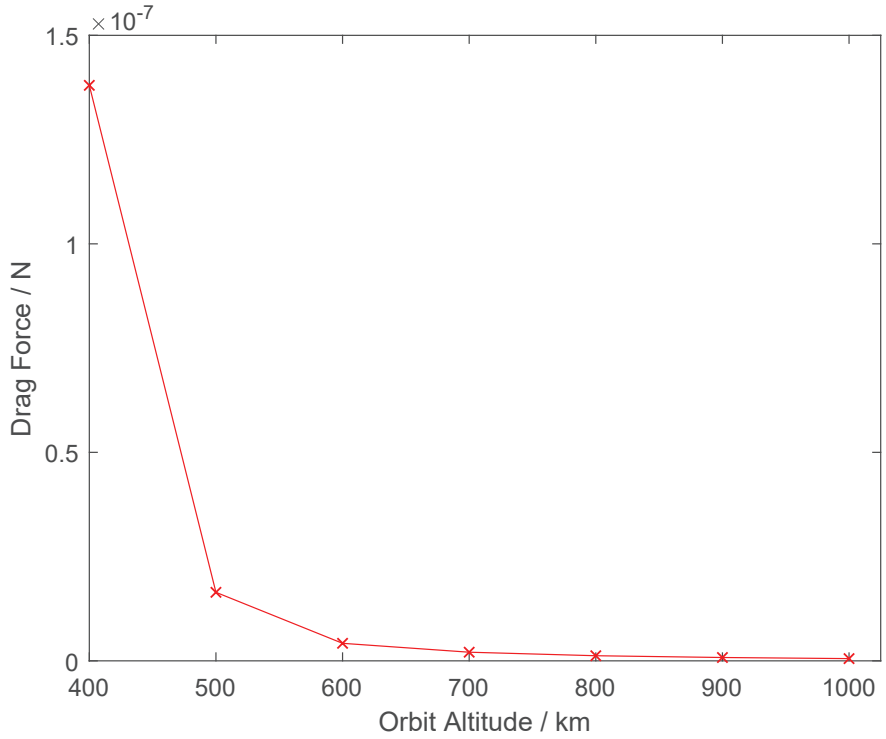


Figure 5.6: Drag force on a CubeSat in LEO computed with the Orekit framework using the JB2000 drag model.

According to Chap. 2.5.1.1, of all perturbing forces only the non-spherical character of the Earth has a major influence on relative formation dynamics in LEOs above 500 km. This is modeled with the Holmes-Featherstone attraction model using the modified forward row method, whose implementation is part of the Orekit framework (Holmes and Featherstone, 2002). The spherical harmonics of the Earth are modeled up to degree and order of ten, which is a reasonable compromise between accuracy and runtime. Higher degrees are not useful for the implemented integrator configuration, since the added correction terms are below the integrator accuracy. To get a higher fidelity, also the very low impact of atmospheric drag was included.

Even though Chap. 2.5.1.1 showed the possibility to neglect this perturbation force, atmospheric drag gains influence in lower altitudes. Fig. 5.6 shows the computed drag force as a function of the orbit altitude using the JB2000 model. By including a drag model, the propagator can also be used to simulate the final phase of the NetSat mission, where a low-altitude graveyard orbit or a possible re-entry is sought. Fig. 5.7 shows simulations of orbit lifetime as a function of the initial orbit altitude and compares the Harris-Priester and the JB2000 model. The Orekit inherent implementation of the modified Harris-Priester atmosphere model was chosen, since it combines high computation speed with relatively high accuracy. It uses solar activity dependent altitude-versus-density tables (Montenbruck and Gill, 2012).

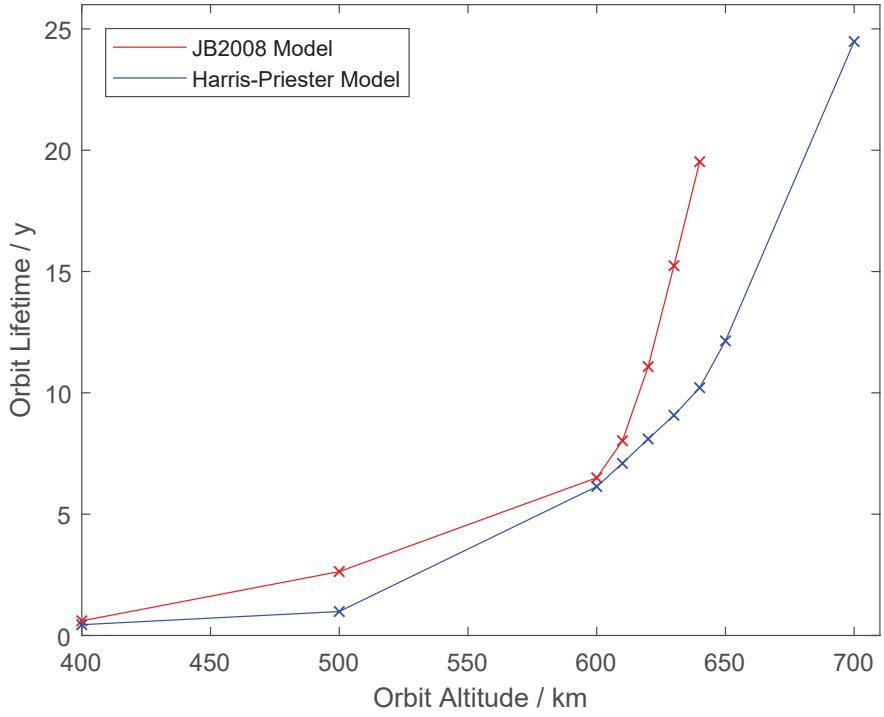
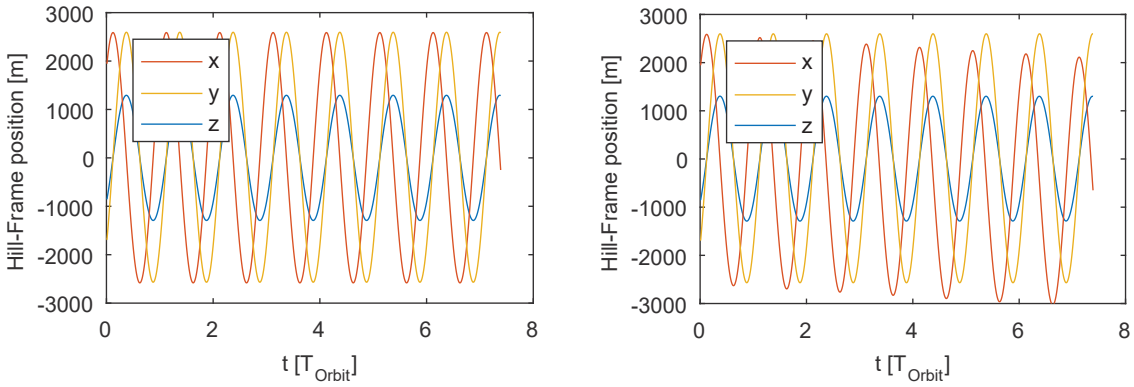


Figure 5.7: Duration of passive deorbiting of a CubeSat computed with the Orekit framework using and comparing the JB2000 and Harris-Priester drag model.

The influence of the non-spherical shape of the Earth on the relative position of two satellites is shown in Fig. 5.8. The two images depict the relative position of a satellite over the time of half a day. The satellite is in a 5 km PCO orbit around a virtual center. The absolute orbit is the nominal NetSat orbit ($T = 95.5\ min$). This leads to slightly more than seven revolutions in twelve hours. Image (I) shows this period in all three components of the EH frame position vector. In image (II), the satellites were propagated in the same manner, except that the J-terms of the Earth were taken into account. The period of the position components stays unchanged, but one can clearly see a vertical shift of the $\hat{\boldsymbol{e}}_{\boldsymbol{y}}$ component over time. After half a

day, the difference in comparison to the unperturbed propagation is as high as 350 m, or 7% of the baseline. This makes it obvious that this effect may not be neglected for the on-board propagator. Fig. 5.9 shows a 3D representation of the perturbed trajectory over time. The J-terms-influence continuously perturbs the orbit and shifts it in the negative $\hat{\boldsymbol{e}}_{\boldsymbol{y}}$ direction, while the other two components stay nearly uninfluenced. This result is in accordance with the theoretical background of Chap. 2.5.1.1. When the differential atmospheric drag is also included in the propagation process, the final state vector hardly differs from the result above. The difference after half a day is below 10 m, around 0.2% of the baseline. The velocity part of the state vector is altered by less than $5 \times 10^{-4}\,\mathrm{m\,s^{-1}}$ over twelve hours. Although this force will not be part of the on-board propagator, this small influence will be included in the absolute Orekit propagation process to gain the highest possible accuracy.



(i) Without perturbing forces

(ii) Non-spherical Earth taken into account

Figure 5.8: Relative position in EH frame of a 5 km PCO formation (one satellite around virtual center) for twelve hours.

5.2.4 Debug Interface

The debug interface is realized as a UDP client, which writes real-time data dumps to the network interface, packed into an extensible protocol. The data dumps are solely represented by floating point arrays for all datatypes (e.g. a one-field array for an atomic double value and a six-field array for an orbit definition, consisting of the six Keplerian parameters). Any combination of parameters from the satellite databases may be marked as debug output by setting a flag in the database. The tri-state-flag may be any of *no debug output*, *debug output grouped by parameter*, or *debug output grouped by satellites*. When the propagator is initialized, it sends a network message with the names of all parameters to be debugged during the propagation process. Then, after each time step, all parameters are sent as an array of floating point numbers in the order of their previously sent names.
Furthermore, any information may be sent over the debug interface when a message type is defined for the desired data (as has been done for the database dump). This

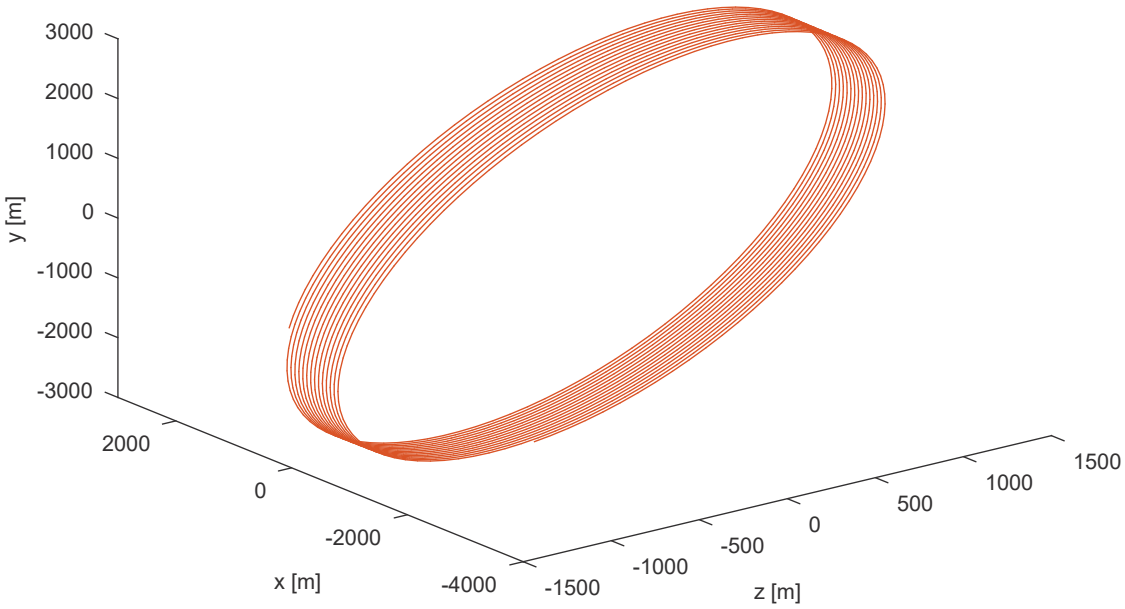


Figure 5.9: 5 km PCO orbit propagated for one day.

also allows to watch simulation metrics which are not part of the satellite model. Since single messages may be larger than the maximum size of a UDP package, a message abstraction layer has been included. Data is sent packed into a message protocol, as defined by Tab. 5.1. Each message is split automatically by the client into multiple network packages, when it is larger than the maximum UDP package size.

Table 5.1: Network message protocol (all parameters except *payload* are integers).

Parameter	Size [byte]	Description
ID	4	unique identifier (auto-incrementing)
TYPE	4	message type
TSIZE	4	total payload size of message
NPKG	2	number of packages in this message
PID	2	identifier of the current package
DSIZE	4	data (payload) size in this package
OFFSET	4	offset of this package's data in message
PAYLOAD	DSIZE	(part of) actual message data

The debug server waits until he receives all packages of a certain message, only then it is processed. The server forwards received messages to a message processor, which displays the content of the messages. Two different processors are implemented. The first one is a simple text-based processor, which displays all received values to the console and can also save the data to CSV-formatted text files on request.

The second one is a graphical processor, which displays all data in MATLAB graphs. When it receives data from the satellite databases, data streams from all satellites are combined and visualized, ordered by parameter or satellite (depending on the flag). Fig. 5.10 denotes the relationship between ordering style and output, which results in different grouping of the atomic values. Depending on the situation or simulation task, any combination of these two may prove to be useful. Plotting routines for other message types have to be added separately. At the moment, there are also routines to visualize absolute and relative orbits in 3D space and thrust profiles.

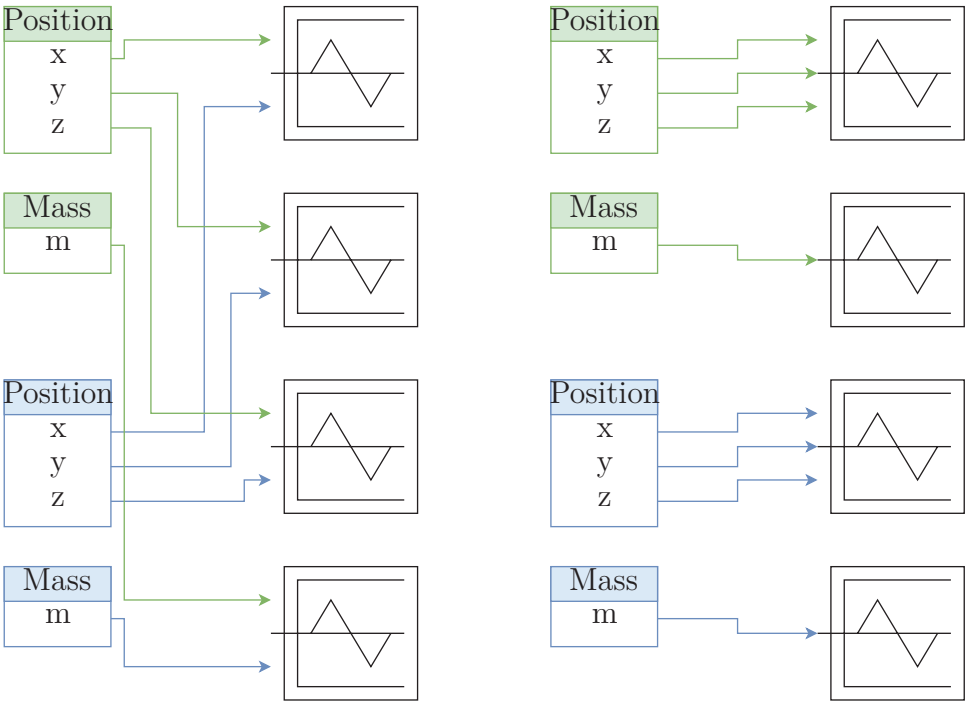


Figure 5.10: Two different methods to group atomic debug values – the left-hand side illustrates grouping by satellites, the right-hand-side grouping by parameter. Each color represents the database of one satellite.

5.3 Scenario Definitions

Within the NetSat mission, which acts as an example mission for this work, a sequence of several formations is planned to be executed. A detailed mission plan listing all designated formations has been created and is presented in Chap. 5.5.1. However, for detailed evaluation of the methods developed, two specific formations are selected, namely the Cartwheel Formation – 3:1 as introduced in Chap. 2.6.6 and the Tetrahedron Formation as presented in Chap. 2.6.8. They have been chosen because of the fact, that they are 3D formations and thus more complex and challenging than the other scenarios in the mission plan, like ATF or PCO. All scenarios and simulations have been based on the same reference orbit. This reference orbit uses the planned and actual orbits of the NetSat satellites presented in Tab. 2.2 as baseline and adapts it slightly for better applicability and understanding. The COEs of the reference orbit are presented in Tab. 5.2.

Table 5.2: Reference orbit in COEs for all simulations (based on NetSat orbits.)

Parameter	Value
Semi-major axis	6950.0 km
Eccentricity	0.01
Inclination	97.660°
Argument of perigee	0°
Right ascension of ascending node	0°
True anomaly	0°

The *Distributed Robust Consensus-Based Control* approach presented in Chap. 3.3 cannot be performed with NetSat-like satellites and scenarios due to the lack of a thrust constraint in its implementation. Thus, for this control approach a scenario based on ESA's Darwin study, namely a spaceborne distributed telescope, is considered and presented later on.

5.3.1 Cartwheel Formation – 3:1

The Cartwheel Formation – 3:1 is a specific form of a Helix formation, which is defined by three satellites having a periodic motion in the orbit plane, namely the $\hat{\boldsymbol{e}}_x$-$\hat{\boldsymbol{e}}_z$ plane, and one satellite having a periodic motion along the $\hat{\boldsymbol{e}}_y$ axis. It is described in detail in Chap. 2.6.6. Both the acquisition phase starting form an ATF and the maintenance phase are considered for controller evaluation. The initial and target states for three satellites in the orbit plane and a single satellite performing the out-of-plane motion are given in ROE in Tab. 5.3.

5.3.2 Tetrahedron Formation

The Tetrahedron Formation is defined by two stationary and two dynamic satellites in a tilted ellipse in way that, through their periodic motion, a dynamic tetrahedron

Table 5.3: CWF – 3:1 initial and target state

	Initial State [m]				Target State [m]			
ROE	Sat 1	Sat 2	Sat 3	Sat 4	Sat 1	Sat 2	Sat 3	Sat 4
∂a	0	0	0	0	0	0	0	0
∂e_x	0	0	0	0	2500	-1250	-1250	0
∂e_y	0	0	0	0	0	2165	-2165	0
∂i_x	0	0	0	0	0	0	0	2500
∂i_y	0	0	0	0	0	0	0	0
∂u	5000	1666	-1666	-5000	0	0	0	0

of constant volume is formed between the four satellites continuously throughout the orbit. It has been defined by Koptev et al. (2017) and is described in Chap. 2.6.8 in detail. Both the acquisition phase starting form an ATF and the maintenance phase are considered for controller evaluation. The initial and target states for the two stationary satellites and the two periodically moving satellites are given in ROE in Tab. 5.4.

Table 5.4: THF initial and target state

	Initial State [m]				Target State [m]			
ROE	Sat 1	Sat 2	Sat 3	Sat 4	Sat 1	Sat 2	Sat 3	Sat 4
∂a	0	0	0	0	0	0	0	0
∂e_x	0	0	0	0	0	1250	0	0
∂e_y	0	0	0	0	0	0	1250	0
∂i_x	0	0	0	0	0	2500	-2500	0
∂i_y	0	0	0	0	0	2500	2500	0
∂u	5000	1666	-1666	-5000	5000	0.99	0.99	-5000

5.3.3 Spaceborne Distributed Telescope

In this scenario, five satellites form a spaceborne distributed telescope. Four satellites are situated in an equilateral square in the $\hat{\boldsymbol{e}}_{\boldsymbol{x}}$-$\hat{\boldsymbol{e}}_{\boldsymbol{y}}$ plane representing the mirrors of a telescope. One satellite is located above the center of the square shifted towards $-\hat{\boldsymbol{e}}_{\boldsymbol{x}}$ direction representing the sensor of the telescope. This scenario is loosely based on ESA's Darwin study that aimed at detecting Earth-like planets orbiting nearby stars by collecting infrared light with several satellites placed at the Lagrange point L2 and combining it in one satellite using interferometry (cf. Figure 5.11). In contrast to Darwin, we place the satellites in a LEO to make use of Hill's equations. The position, where such a spaceborne telescope would be placed, depends mainly on the scientific aim of the observation (which is out of scope of this work). However, a LEO is of higher interest for other SFF missions and thus preferred as example. The formation will be under the influence of orbit dynamics which will lead to a decay

of the formation topology. Thus, continuous control is required for maintaining it.

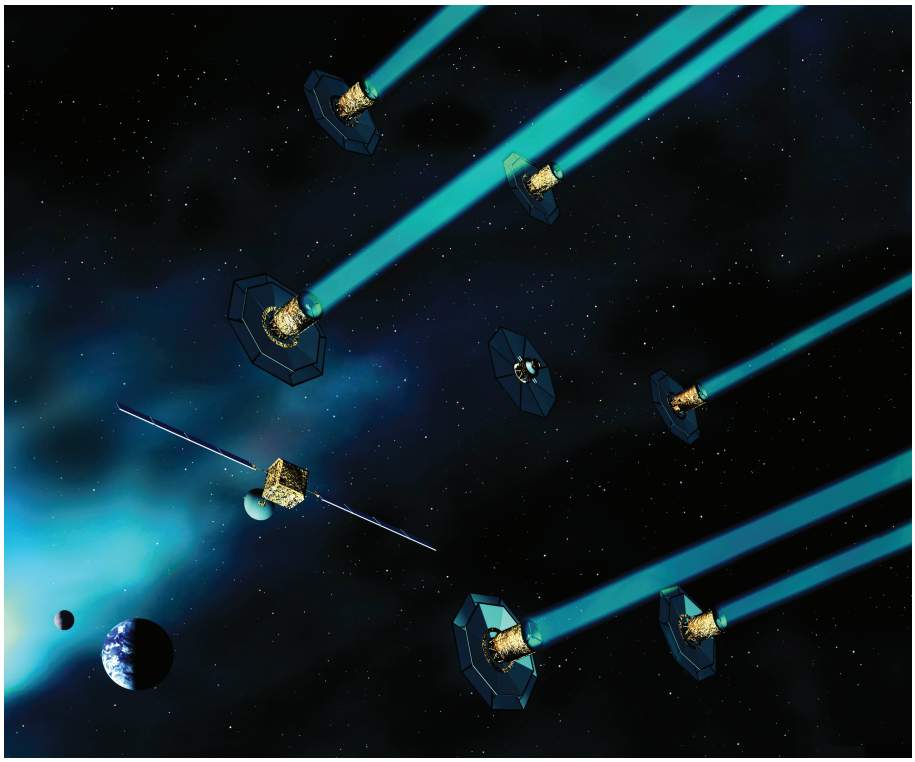


Figure 5.11: Artificial drawing of ESA's Darwin mission study. (Image: ESA/-Darwin, 2002, `http://www.esa.int/spaceinimages/Images/2002/11/Darwin_will_combine_light_from_four_or_five_telescopes_and_send_it_down_to_Earth`)

As target reference a static (in EH frame fixed and not moving) pentahedral formation as required for a spaceborne telescope is used. It shows a $10 \times 10\ m$ base in the $\hat{\boldsymbol{e}}_x$-$\hat{\boldsymbol{e}}_y$ plane and the tip is located at $-10\ m$ along $\hat{\boldsymbol{e}}_z$ axis. This target formation is not force-free, which means that orbit dynamics (cf. Eq. 2.72) are acting on the satellites and degrading the formation topology. Thus, after acquiring this formation, the controllers remain active for maintaining it. The satellites start from initial positions on v-bar $(-20, -10, 0, 10, 20$ m), which are – in contrast to the target formation – force-free. So the initial formation would not degrade due to orbit dynamics and active control is required to move the satellites away from it.

5.3.4 Controller-Specific Parameters

In the following sections, controller-specific parameters that have been used in the performed simulations are presented. In general, all simulations ran in the FSF in Java which includes Orekit that has been introduced in Chap. 5.2. In addition, MathWorks MATLAB Version 2020a and the according toolboxes were deployed in the controller implementation of the *MPC-Based Formation Control* and the

Distributed Robust Consensus-Based Control. For all simulations a simulation time step of 60 s is chosen.

5.3.4.1 MPC-Based Formation Control

The controller has been implemented in MathWorks MATLAB (Version 2018b) together with the MPC and Optimization Toolboxes and connected to FSF and Orekit.

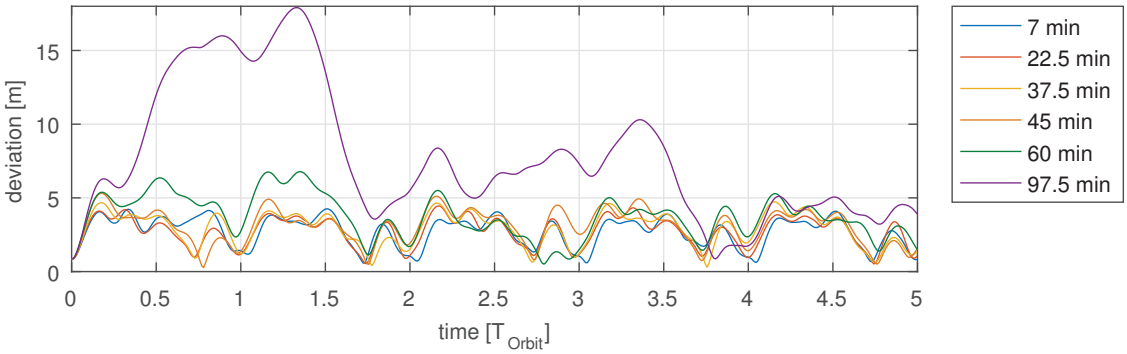


Figure 5.12: Position deviation over simulation time for different planning horizons (5 km CWF).

The two most important design parameters of the MPC algorithm are the planning horizon and the control interval (sometimes also referred to as control horizon). The planning horizon is the amount of time over which MPC optimizes the thrust profile. Since MPC generates globally optimal solutions over the planning horizon, a longer horizon should lower the long-term ΔV consumption. In the implementation described in Chap. 3.2.0.1, the orbit is only enforced at the end of the planning horizon, leading to larger position errors in between, when the horizon is extended as can be seen in Fig. 5.12. It shows the maximal position deviation from the target reference for different planning horizons between 7 min and 97.5 min, where the latter corresponds to one orbital period. Below 60 min, the deviation does not fall significantly for shorter intervals. The five-meter-level between full orbital periods, which is shared by all intervals below 60 min, is a result from the deviation of the relative orbit propagator $\boldsymbol{\Phi}_{MGA}\left(\delta\bar{\boldsymbol{a}}_0, t_f\right)$, which receives a position update in worst case only once per orbit. This is the minimum inter-satellite communication requirement within the NetSat mission (cf. Chap. 4.5) and also representative for other similar CubeSat missions. In other words, there is no need to lower the planning horizon below 60 min in terms of orbit deviation, since it does not lead to a gain in accuracy.
As stated before, the planning horizon also influences the total ΔV consumption, which is depicted in Fig. 5.13. The longer the horizon, the better the solution becomes. Below half an orbital period (around 50 min), the per-orbit ΔV rate rises rapidly. In the range of these short planning horizons, very small high-frequency position oscillations are counteracted, which consumes a lot of ΔV. For longer

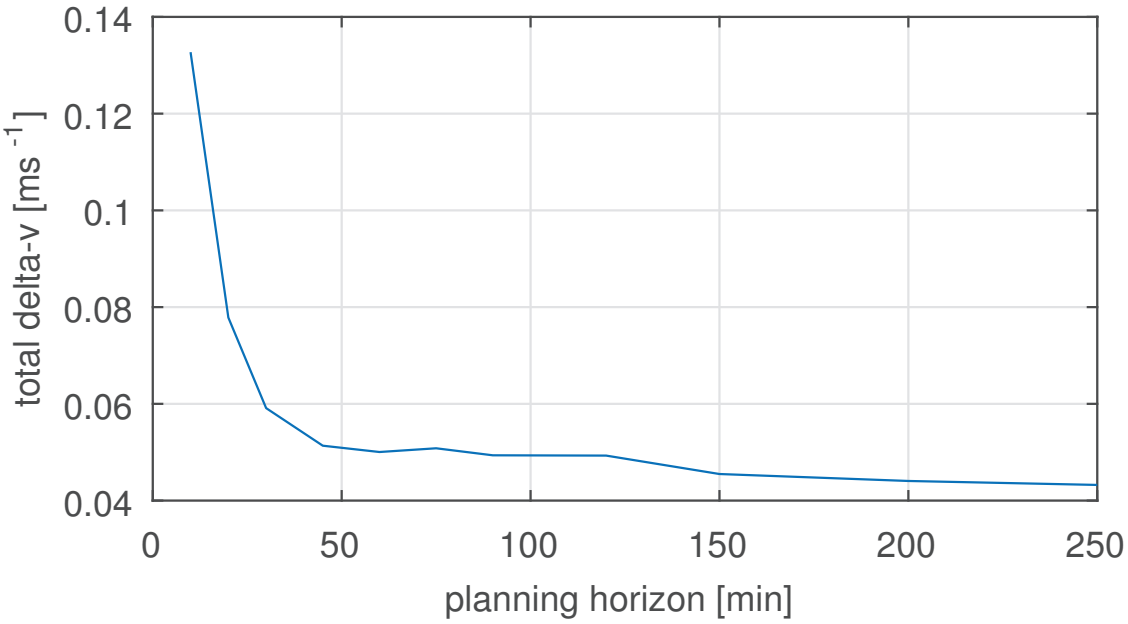


Figure 5.13: Per-orbit ΔV over MPC planning horizon (5 km CWF).

planning horizons, these oscillations are smoothed out and ΔV is saved. The figure also reveals that there is only a very small gain in optimality for much longer planning horizons. Combining the result of the two observations in terms of position deviation and ΔV consumption, a planning interval of around half the orbital period – 50 min – seems to be the best choice. This choice also allows a per-orbit long-term planning between maintenance and reconfiguration maneuvers, since the end of one orbit period always coincides with the end of a planning horizon.

The MPC performance also depends on the interval at which new thrust profiles are generated (control interval, see beginning of Chap. 2.7.4). If this interval is shorter than the planning horizon, the last part of the generated thrust profile is dismissed and replaced with the beginning of the newly generated profile of the next MPC execution. The influence of the control interval on ΔV consumption is shown in Fig. 5.14 for the three planning horizons 30 min, 60 min and 97.5 min. Each line ends at a planning interval which coincides with the planning horizon. If the control interval is larger, there would be uncontrolled times between the end of the planning horizon and a new control interval. Although the data points seem to be arbitrarily distributed at first sight, one can draw a rough conclusion from the figure. All three curves have in common that they reveal the lowest ΔV consumption for a very short control interval (from seconds up to a few minutes) and for a control interval equal to the planning horizon. Since it seems unlikely that the on-board computer can solve the MPC optimization at a high frequency, the planning intervals below one minute are dismissed. This leads to the conclusion that the optimal choice for the control interval is the length of the planning horizon.

For formation acquisition/reconfiguration maneuvers, a scaling factor for the terminal orbit constraint (target position of the satellite) is used. Thus, in one planning period,

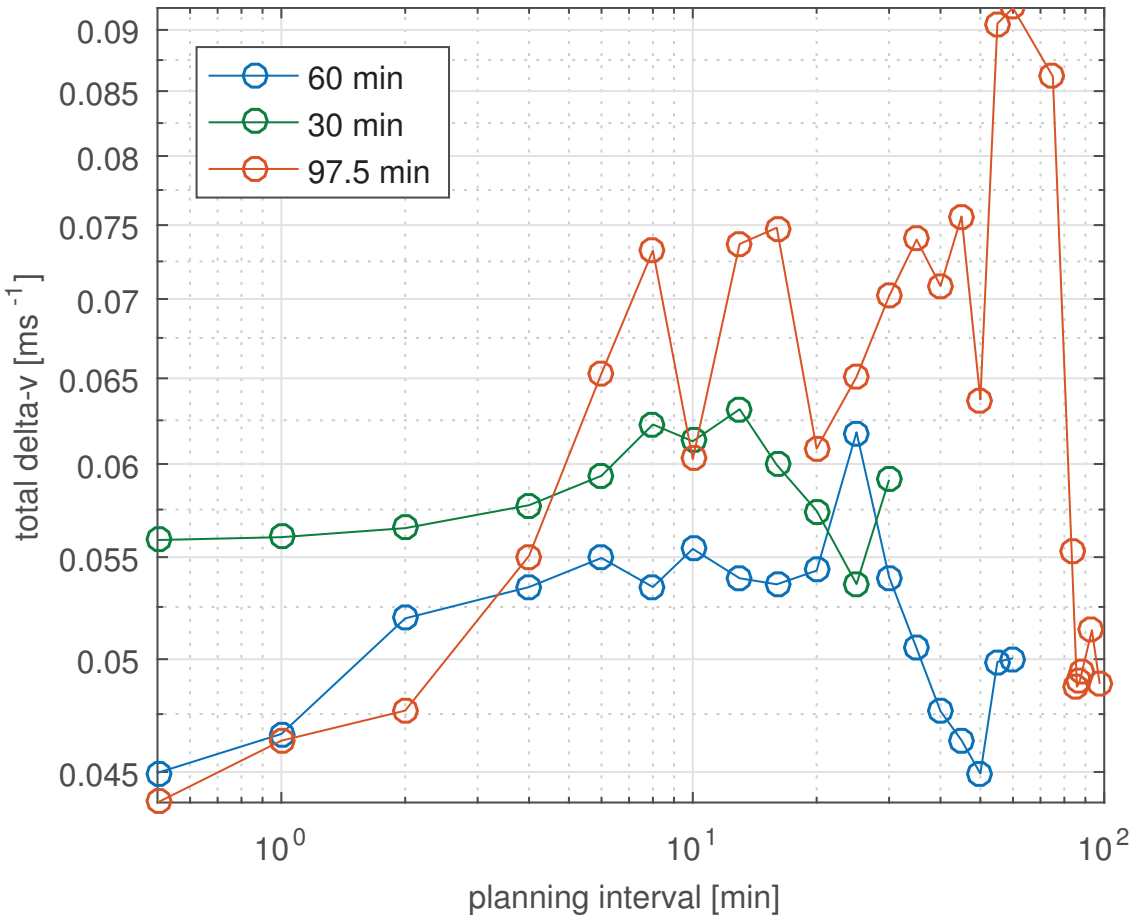


Figure 5.14: Per orbit ΔV over MPC control interval for different planning horizons (5 km CWF).

it is only compensated for a fraction of the difference between the desired orbit and the current orbit. This is necessary for almost any maneuver conducted with CLT, since it takes a very long time for noticeable orbit changes. The scaling factor is dynamically adapted. The MPC starts with a scaling factor of $c = 1$ (no scaling at all) and tries to find a valid thrust profile to bring the satellite to the target state. If no valid thrust profile is found by the MPC, this scaling factor is lowered, reducing the orbit difference to overcome during one planning interval. This step is iterated until a valid thrust profile is found. The scaling factor depends on the thrust system used on-board the satellites, the orbit, the formation type and the formation baseline. Based on these factors a initial scaling factor $c < 1$ (e.g. $c = 0.1$) can be chosen to reduce the computational effort (especially in on-board implementation).

5.3.4.2 Distributed Robust Consensus-Based Control

The controller has been implemented in MathWorks MATLAB (Version 2020a) together with the Robust Control Toolbox. The Hill's equations (Eq. 2.77) are used as the propagation model for the simulations performed. Although, they are limited

in accuracy for specific applications (e.g. large scale formations, eccentric orbits) and do not include orbit perturbations (like J2 effect or atmospheric drag), they are suitable and sufficient for the considered scenario, since it is a very close range scenario (below 50 m) and the main focus is on the distributed control part.

The H_∞ controller synthesis performed depends highly on the setup chosen (number of satellites, interconnection scheme, input and output weights) and thus is presented here together with the required parameters.
To present a realistic robust control example, we further assume disturbances acting on the satellite. We assume a Gaussian distributed force acting on the satellite with a magnitude of 50 μN (standard normal distribution). This force represents thruster uncertainties of a typical electric propulsion system.

Network Topology

Further, we do not assume global knowledge in the sense that each spacecraft has a communication link and state exchange with every other spacecraft, but only with its neighbors. This is especially true for communication links that are limited in power and thus in range (like in CubeSats).

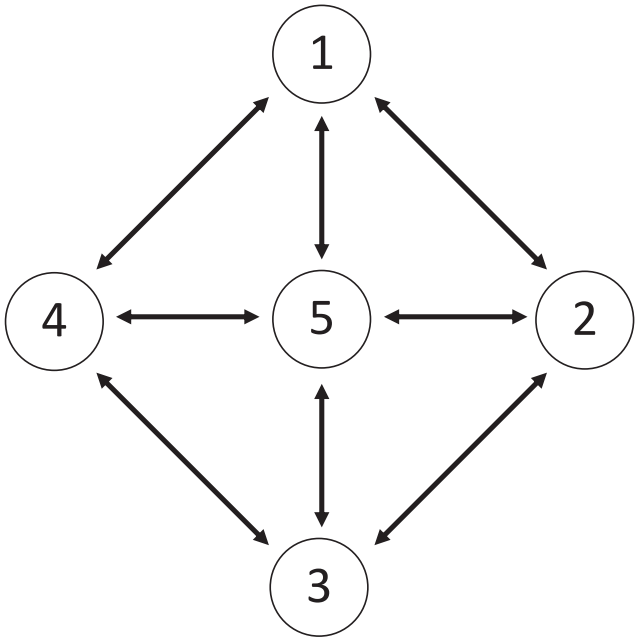


Figure 5.15: Interconnection graph of five satellites in a spaceborne telescope example. Interconnections between satellites are assumed to be bidirectional.

The network topology of this scenario is visualized in Fig. 5.15. Since every vertex of the interconnection graph/agent except of Number (5) is only connected to all but one other vertex/agent, the graph is not fully interconnected or complete. Based on the graph, the degree matrix D, which states the number of interconnections of the

subsystems/agents, and the adjacency matrix A_j, whose non-zero elements represent the interconnections, can be composed. (Definitions are given in Chap. 2.8.)

$$\boldsymbol{D} = \begin{pmatrix} 3 & 0 & 0 & 0 & 0 \\ 0 & 3 & 0 & 0 & 0 \\ 0 & 0 & 3 & 0 & 0 \\ 0 & 0 & 0 & 3 & 0 \\ 0 & 0 & 0 & 0 & 4 \end{pmatrix} \tag{5.3}$$

$$\boldsymbol{A}_j = \begin{pmatrix} 0 & 1 & 0 & 1 & 1 \\ 1 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 1 & 1 \\ 1 & 0 & 1 & 0 & 1 \\ 1 & 1 & 1 & 1 & 0 \end{pmatrix} \tag{5.4}$$

From $\boldsymbol{D}$ and $\boldsymbol{A}_j$ the Laplacian matrix can be computed to

$$\boldsymbol{L} = \boldsymbol{D} - \boldsymbol{A}_j = \begin{pmatrix} 3 & -1 & 0 & -1 & -1 \\ -1 & 3 & -1 & 0 & -1 \\ 0 & -1 & 3 & -1 & -1 \\ -1 & 0 & -1 & 3 & -1 \\ -1 & -1 & -1 & -1 & 4 \end{pmatrix} \tag{5.5}$$

Input and Output Weights

As we can see from the previous section and Eq. 3.30 the control problem can be formulated as a distributed state-space system. Thus, a *Distributed Robust Consensus-Based Control* approach as derived in Chap. 3.3.2 can be applied. Eq. 3.19 presents the CLTF of an individual agent in a distributed control scenario and is applied in the presented scenario[3]. The control system is modeled in MATLAB including the aforementioned disturbance inputs d on all satellites, which represent thruster uncertainties of a typical electric propulsion system for small satellites with a maximum thrust force in the order of few 100 µN. We define the input weight for disturbance as a static scaling gain

$$\boldsymbol{W}_d = 50 \cdot 10^{-6} \cdot \boldsymbol{I}_3 \tag{5.6}$$

[3]Note that this approach does not control the position of the center of the formation. It only controls the relative vectors within the formation. Due to disturbances, the center of the formation might move with time, but the shape of the formation will be kept constant. Further, since relative vectors are defined with respect to a basis of a vector space, namely the axes of the EH coordinate frame, the orientation of the formation is fixed with respect to these axes.

and the input weight for the reference value also as a static gain

$$\boldsymbol{W}_r = \boldsymbol{I}_6 \tag{5.7}$$

with $\boldsymbol{I}_k$ being identity matrices of size $k \times k$. The output weights are defined as first-order weighting function with specified static gain, crossover frequency, and high-frequency gain. Their TFs are defined as

$$\boldsymbol{W}_1 = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix} \cdot \frac{60.2\ s + 0.802}{s + 0.0802} \tag{5.8}$$

$$\boldsymbol{W}_2 = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix} \cdot \frac{10\ s + 13.27}{s + 16.58} \tag{5.9}$$

$$\boldsymbol{W}_3 = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \cdot \frac{s + 0.1}{s + 10} \tag{5.10}$$

Controller Synthesis

The robust control problem is defined either by finding a stabilizing controller that minimizes the H_∞ norm of the CLTF with external inputs (disturbance d, references r) and output error signals (z_1, z_2 and z_3) $\boldsymbol{T}_{zw}$ or (as the suboptimal control problem that can be solved numerically, cf. Chap. 2.7.5.1 and Eq. 2.186) as establishing a stabilizing controller with

$$\|\boldsymbol{T}_{zw}\|_\infty < \gamma, \gamma > 0$$

In addition to that, we impose constraints on the structure of the controller in question in order to limit its complexity. As a reasonable compromise between performance and complexity, we require the controller to be a state-space system with 6 states, 6 inputs and 3 outputs. Thus it's state-space representation has the following form.

$$\boldsymbol{A}_c = \begin{pmatrix} a_{11} & \cdots & a_{16} \\ \vdots & \ddots & \vdots \\ a_{61} & \cdots & a_{66} \end{pmatrix} \qquad \boldsymbol{B}_c = \begin{pmatrix} b_{11} & \cdots & b_{16} \\ \vdots & \ddots & \vdots \\ b_{61} & \cdots & b_{66} \end{pmatrix}$$

$$\boldsymbol{C}_c = \begin{pmatrix} a_{11} & \cdots & a_{16} \\ \vdots & \ddots & \vdots \\ a_{31} & \cdots & a_{36} \end{pmatrix} \qquad \boldsymbol{D}_c = \begin{pmatrix} a_{11} & \cdots & a_{16} \\ \vdots & \ddots & \vdots \\ a_{31} & \cdots & a_{36} \end{pmatrix} \tag{5.11}$$

The H_∞ synthesis was performed in MATLAB. Its *hinfstruct()* function successfully finds a stabilizing controller for all satellites and achieves closed-loop H_∞ norm values of $\gamma = 0.679$ for Satellites 1 to 4 and $\gamma = 0.641$ for Satellite 5 after 305 iterations. For Satellite 1 to 4 the stabilizing controller is computed to

$$\boldsymbol{A}_c = \begin{pmatrix} -0.347 & -1.382 & 0.882 & 0.889 & 0.841 & -0.173 \\ 0.262 & -1.505 & -0.644 & -0.813 & 0.073 & -0.090 \\ -0.710 & -0.258 & -0.731 & -0.673 & -0.678 & 1.062 \\ 0.207 & 0.957 & -0.102 & -1.703 & 0.740 & 0.189 \\ -0.138 & 0.606 & 0.329 & 0.226 & 0.018 & 0.235 \\ 1.481 & -0.038 & -0.962 & -0.225 & -0.459 & -1.700 \end{pmatrix} \tag{5.12}$$

$$\boldsymbol{B}_c = \begin{pmatrix} -0.753 & -0.398 & -0.494 & -0.246 & 0.608 & 0.171 \\ -0.057 & 0.070 & -0.706 & 0.050 & 0.249 & -0.114 \\ 0.228 & -0.271 & -0.709 & -0.571 & 0.154 & -0.258 \\ -0.169 & 0.212 & 0.329 & 0.230 & -0.416 & -0.166 \\ 0.293 & 0.344 & 0.799 & 0.367 & 0.406 & -0.060 \\ -0.063 & -0.290 & -0.336 & -0.143 & -0.745 & 0.392 \end{pmatrix} \tag{5.13}$$

$$\boldsymbol{C}_c = \begin{pmatrix} 0.198 & -0.396 & 0.492 & 0.202 & 0.695 & -0.477 \\ 0.123 & -0.184 & -1.057 & -1.942 & -0.107 & 0.469 \\ 0.265 & 1.011 & -0.315 & -0.669 & -0.112 & -0.075 \end{pmatrix} \tag{5.14}$$

$$\boldsymbol{D}_c = \begin{pmatrix} -0.496 & 0.106 & -0.163 & 0.050 & -0.184 & 0.135 \\ 0.167 & 0.115 & -0.458 & -0.052 & -0.162 & -0.293 \\ 0.113 & 0.027 & 0.229 & -0.003 & -0.509 & -0.013 \end{pmatrix}. \tag{5.15}$$

The computed controllers for Satellites 1 to 4 are identical, because their positions and connections within the network topology (as shown in Fig. 5.15) are symmetric. Satellite 5 differs and thus the H_∞ synthesis leads to a different controller:

$$\boldsymbol{A}_c = \begin{pmatrix} -0.903 & -1.186 & -0.761 & 0.483 & 0.195 & -0.041 \\ -0.981 & -0.334 & 0.910 & 0.396 & -0.539 & -0.130 \\ -0.937 & -0.042 & 0.215 & -0.273 & -1.667 & -0.087 \\ 0.204 & 0.137 & -0.172 & 0.109 & 0.029 & -0.081 \\ 0.856 & 0.426 & -0.123 & -0.115 & 2.840 & 0.302 \\ -1.612 & -0.540 & 1.686 & 0.581 & -2.489 & -0.350 \end{pmatrix} \tag{5.16}$$

$$\boldsymbol{B}_c = \begin{pmatrix} -0.903 & -1.186 & -0.761 & 0.483 & 0.196 & -0.041 \\ -0.981 & -0.334 & 0.910 & 0.396 & -0.539 & -0.130 \\ -0.937 & -0.042 & 0.215 & -0.273 & -1.667 & -0.087 \\ 0.204 & 0.137 & -0.172 & 0.109 & 0.029 & -0.081 \\ 0.858 & 0.426 & -0.123 & -0.115 & 2.840 & 0.302 \\ -1.612 & -0.540 & 1.686 & 0.581 & -2.489 & -0.351 \end{pmatrix} \tag{5.17}$$

$$\boldsymbol{C}_c = \begin{pmatrix} 0.295 & -0.606 & -0.184 & -0.504 & 0.219 & 0.207 \\ -0.419 & 0.540 & -0.191 & -1.692 & 0.404 & -0.241 \\ 0.457 & 1.008 & 0.203 & 0.523 & -0.201 & -0.413 \end{pmatrix} \tag{5.18}$$

$$\boldsymbol{D}_c = \begin{pmatrix} -0.461 & -0.048 & 0.201 & 0.091 & -0.364 & -0.076 \\ -0.055 & -0.175 & -0.597 & 0.040 & -0.259 & -0.067 \\ 0.285 & 0.260 & 0.168 & -0.084 & -0.439 & -0.040 \end{pmatrix}. \tag{5.19}$$

With these controllers, the given example can be simulated with its results presented later on in Chap. 5.4.3.

5.3.4.3 Plant Inversion-Based Lyapunov Control Combined with a Reference Governor

This controller has been implemented completely in the FSF without involving MATLAB. The controller's parameters used in the simulations, which were introduced in Chap. 3.4, were set to the values shown in Tab. 5.5.

Table 5.5: Controller parameters for the *Plant Inversion-Based Lyapunov Control Combined with a Reference Governor* presented in Chap. 3.4.

Parameter	Value	Description
N	4	Defines the shape of the thrust around the optimal thrust locations, used in $\boldsymbol{P}$ matrix
k	200	Control gain, used in $\boldsymbol{P}$ matrix
ξ	$1 \cdot 10^{10}$	Influences the velocity of the applied reference
τ	$1 \cdot 10^{4}$	Inversely proportional to the $\partial\lambda$ control gain
C_s	25	Influence zone for collision avoidance in meters

5.4 Simulation Results

With the controller parameters and the simulation scenarios defined in Chap. 5.3, the three control approaches developed in Chap. 3 can now be simulated. This section is structured in three main subsections covering the three simulation scenarios CWF – 3:1, THF and spaceborne telescope. Within each of those subsections the different controllers are simulated, relevant graphs are presented and discussed and for each scenario a summary of the results is given.

5.4.1 Cartwheel Formation – 3:1

The CWF – 3:1 formation scenario covers two phases, namely the acquisition and maintenance phase. Both are simulated and presented subsequently with a summary afterwards.

5.4.1.1 Acquisition Phase

The initial states from which the satellites start the maneuver of the CWF – 3:1 acquisition phase as well as the target states are defined in Tab. 5.3 and repeated here for better readability.

CWF – 3:1 initial and target state

	Initial State [m]				Target State [m]			
ROE	Sat 1	Sat 2	Sat 3	Sat 4	Sat 1	Sat 2	Sat 3	Sat 4
∂a	0	0	0	0	0	0	0	0
∂e_x	0	0	0	0	2500	-1250	-1250	0
∂e_y	0	0	0	0	0	2165	-2165	0
∂i_x	0	0	0	0	0	0	0	2500
∂i_y	0	0	0	0	0	0	0	0
∂u	5000	1666	-1666	-5000	0	0	0	0

MPC-Based Formation Control

Fig. 5.16 and 5.17 show the motion of the four satellites during this acquisition maneuver in the EH frame. The control is computed using the *MPC-Based Formation Control* (cf. Chap. 3.2) to acquire the CWF – 3:1 formation. The motion from the starting points on $\hat{\boldsymbol{e}}_{\boldsymbol{x}}$ axis towards the ellipse forming the CWF – 3:1 formation is formed by multiple spirals, which can clearly be seen from Fig. 5.17. This is the case, because the thrusters of the satellites are limited and the maneuver takes several days – roughly 15 days for this scenario – to be performed.

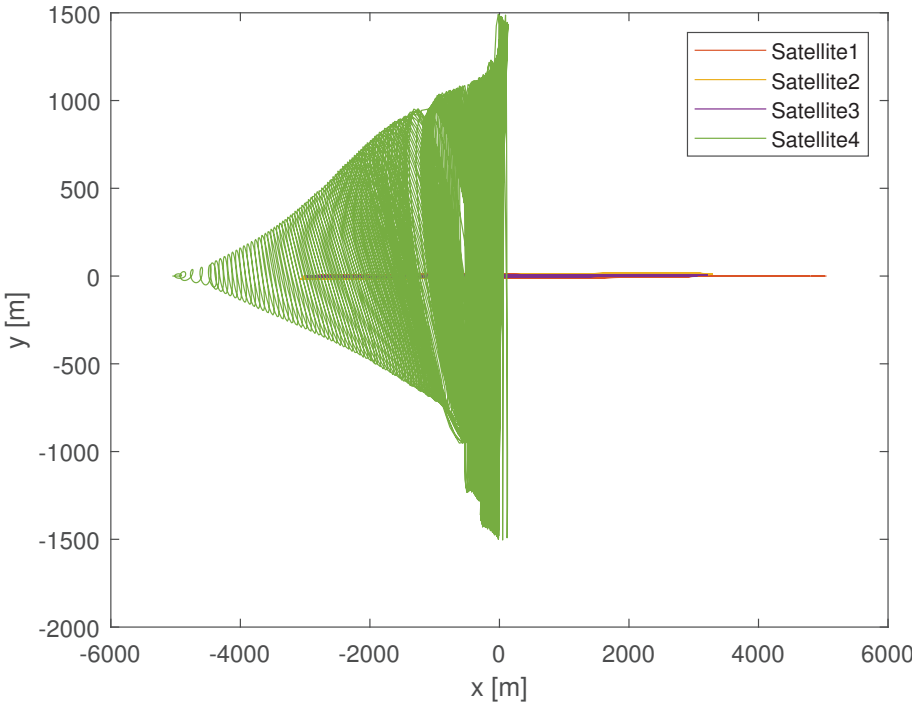


Figure 5.16: $\hat{e}_x$-$\hat{e}_y$ motion of the four satellites in EH frame using *MPC-Based Formation Control* to acquire the CWF – 3:1.

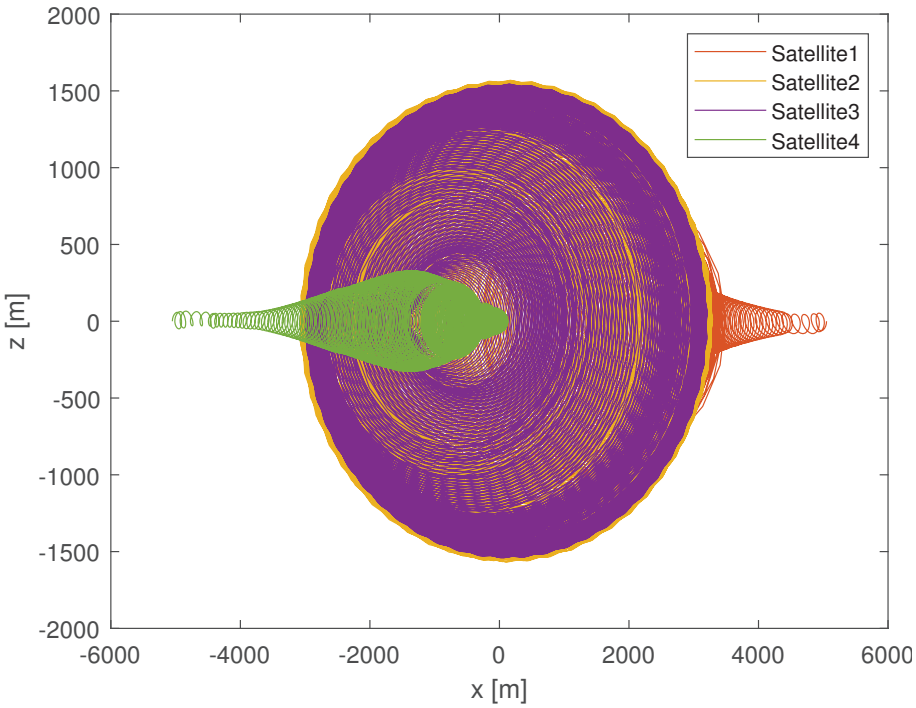


Figure 5.17: $\hat{e}_x$-$\hat{e}_z$ motion of the four satellites in EH frame using *MPC-Based Formation Control* to acquire the CWF – 3:1.

Fig. 5.18 shows the relative error between the starting points and the target points in ROEs over time for the four simulated satellites. The relative errors of all satellites converge to zero, thus the target formation is reached. The time required for reaching the target state varies between the four satellites from about 180 to about 290 orbits, which roughly corresponds to 12 to 20 days. The difference in the convergence time depends on the different initial conditions and the fact that one satellite (Satellite 4) has a different role and thus also a different target state which requires less time to be acquired.

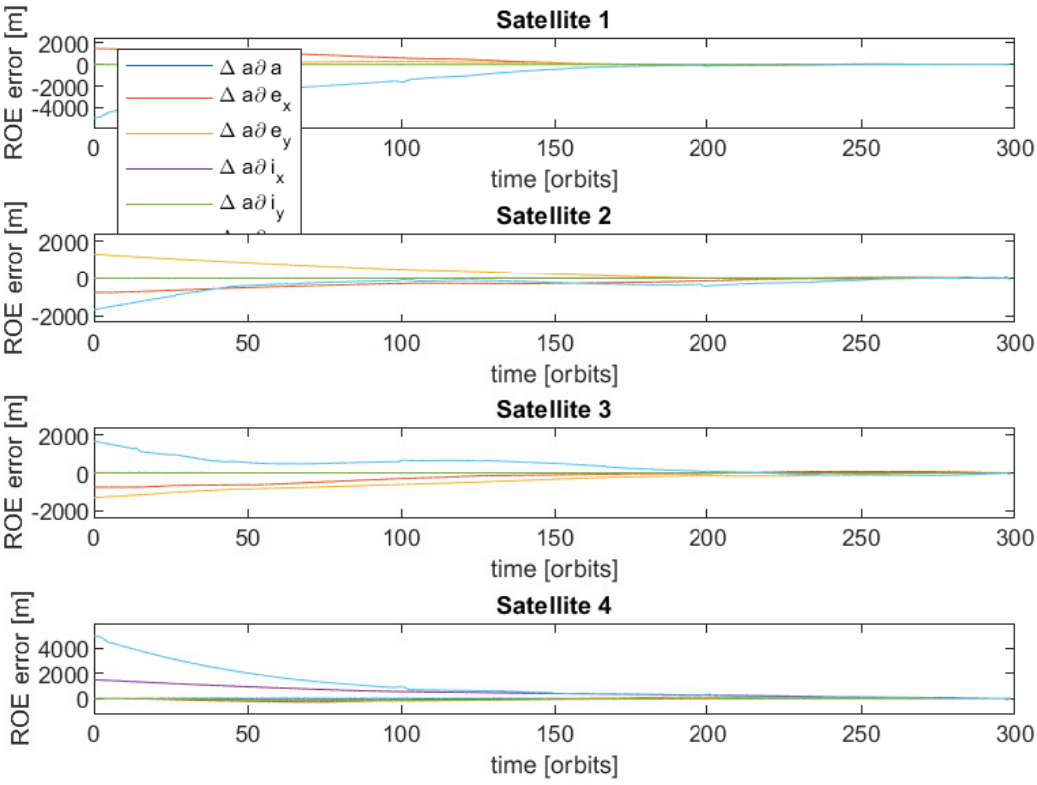


Figure 5.18: Relative error of the four satellites in ROEs over time using *MPC-Based Formation Control* to acquire the CWF – 3:1.

Fig. 5.19 shows the thrust activity and distribution of the four satellites in EH frame over time. One can clearly see that Satellite 4, which performs the out-of-plane motion, has a strong thrust contribution along the $\hat{e}_y$ axis, whereas the other three satellites are mainly active in the $\hat{e}_x$-$\hat{e}_z$ plane. One can further see that the thrust profiles are limited to the maximum thrust that has been specified in the MPC configuration (and which corresponds to realistic values for CubeSats, cf. Chap. 4.5). Further, one can see that the thrust phases (the "spikes" in the figures, which are actually phases of several minutes duration) are distributed at thrust-optimal positions as computed by the MPC.

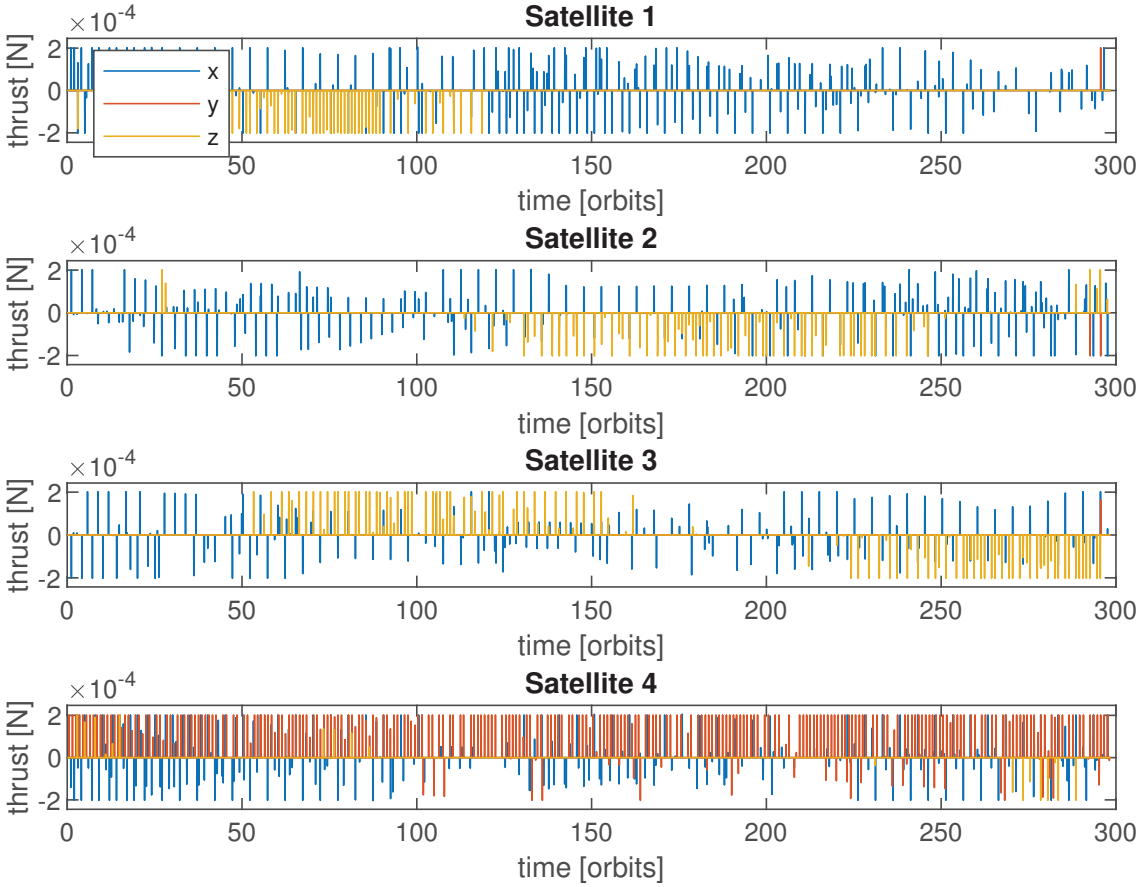


Figure 5.19: Thrust profile of the four satellites in EH frame over time using *MPC-Based Formation Control* to acquire the CWF – 3:1.

Plant Inversion-Based Lyapunov Control Combined with a Reference Governor

Fig. 5.20 and 5.21 show the motion of the four satellites during this acquisition maneuver in the EH frame. The control is computed using the *Plant Inversion-Based Lyapunov Control Combined with a Reference Governor* (cf. Chap. 3.4) to acquire the CWF – 3:1 formation. The motion from the starting points on $\hat{\boldsymbol{e}}_{\boldsymbol{x}}$ axis towards the ellipse forming the CWF – 3:1 formation is formed by multiple spirals, which can clearly be seen from Fig. 5.17. This is the case, because the thrusters of the satellites are limited and the maneuver takes several days – roughly 3 days for this scenario – to be performed. The motion of Satellites 1 to 3 seems very structured and straight forward. Only Satellite 4, which performs the out-of-plane motion shows some detours to reach the target. This may lead to the conclusion that this controller is stronger in in-plane-motion than in out-of-plane motion.

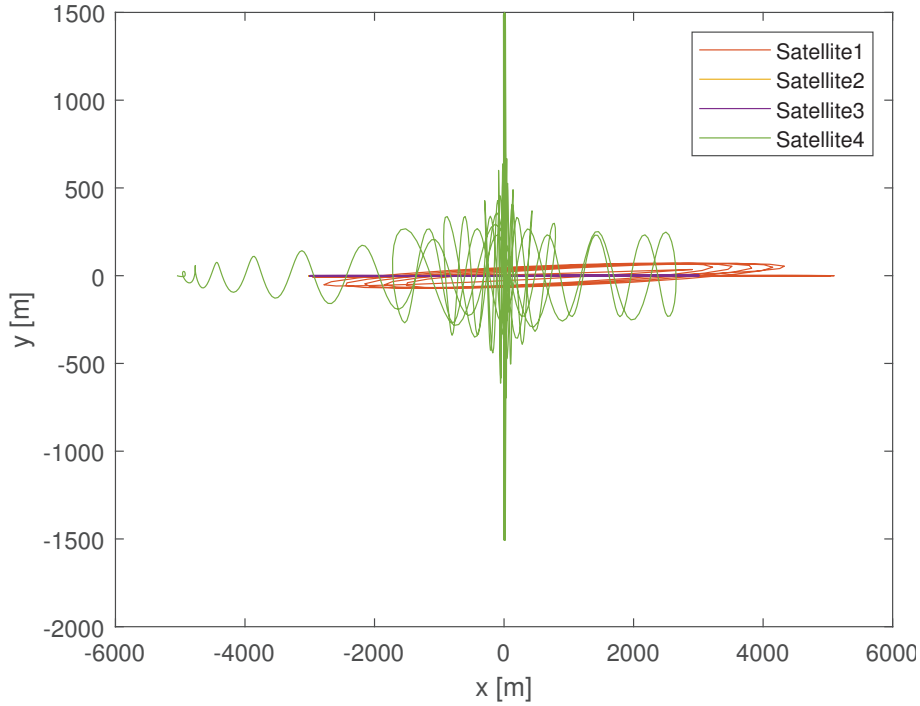


Figure 5.20: $\hat{\boldsymbol{e}}_{\boldsymbol{x}}$-$\hat{\boldsymbol{e}}_{\boldsymbol{y}}$ motion of the four satellites in EH frame using *Plant Inversion-Based Lyapunov Control Combined with a Reference Governor* to acquire the CWF – 3:1.

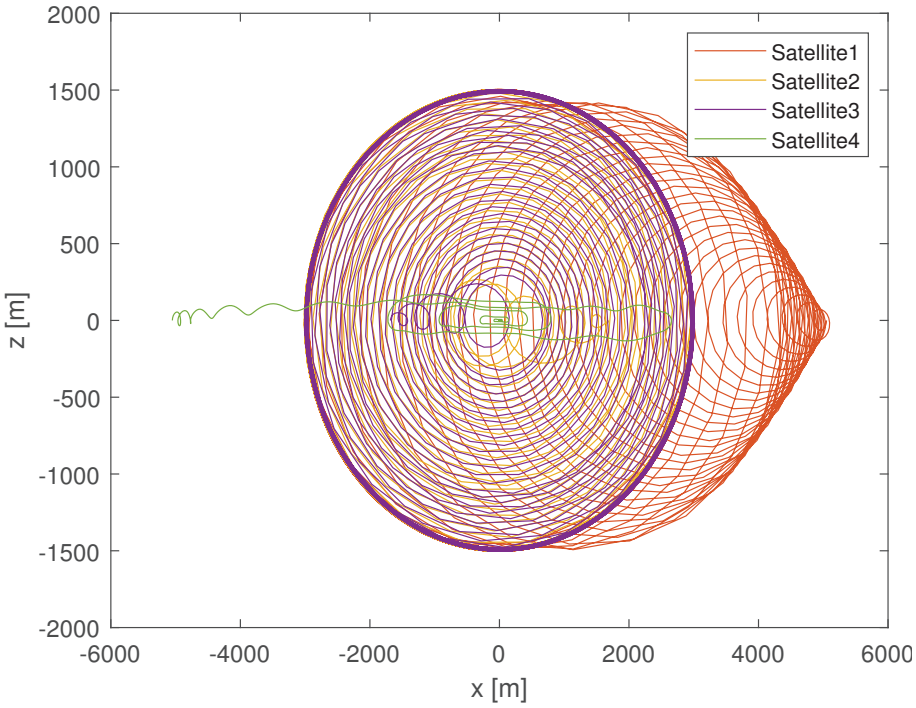


Figure 5.21: $\hat{e}_x$-$\hat{e}_z$ motion of the four satellites in EH frame using *Plant Inversion-Based Lyapunov Control Combined with a Reference Governor* to acquire the CWF – 3:1.

Fig. 5.22 shows the relative error between the starting points and the target points in ROEs over time for the four simulated satellites. The relative errors of all satellites converge to zero, thus the target formation is reached. The time required for reaching the target state varies between the four satellites from about 30 to about 60 orbits, which roughly corresponds to 2 to 4 days. The difference in the convergence time depends on the different initial conditions and the fact that one satellite (Satellite 4) has a different role and thus also a different target state which requires less time to be acquired.

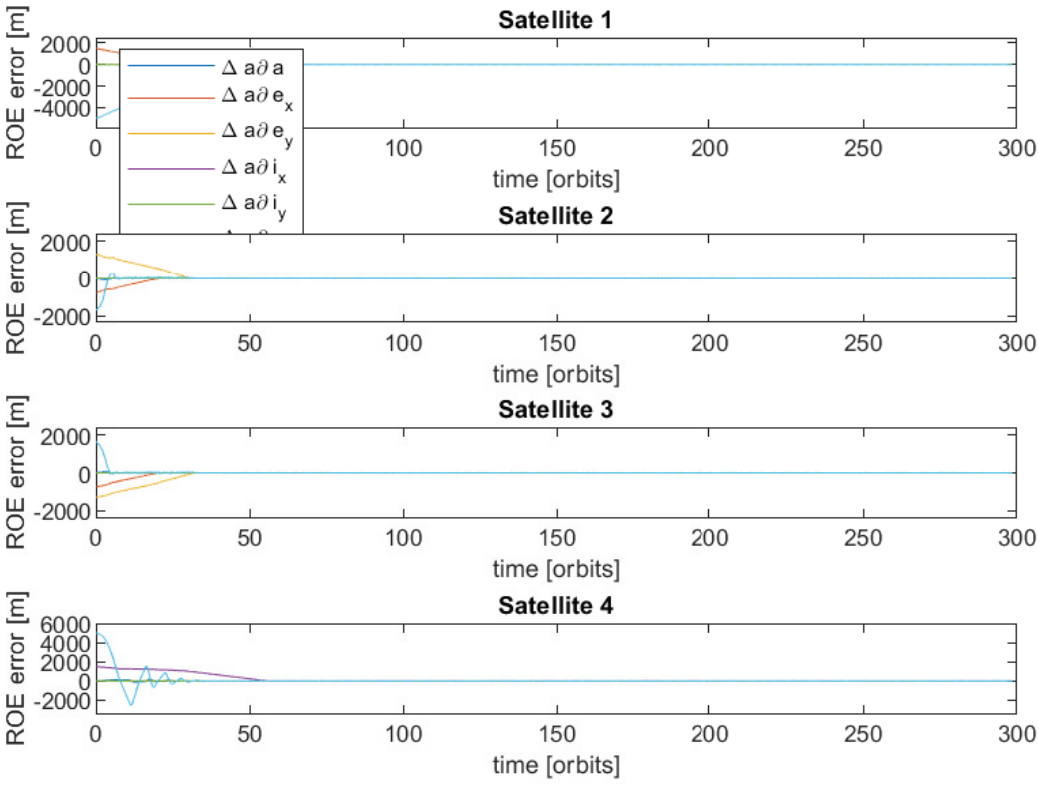


Figure 5.22: Relative error of the four satellites in ROEs over time using *Plant Inversion-Based Lyapunov Control Combined with a Reference Governor* to acquire the CWF – 3:1.

Fig. 5.23 shows the thrust activity and distribution of the four satellites in EH frame over time. Using this control approach, thrust is only applied in the $\hat{\boldsymbol{e}}_x$-$\hat{\boldsymbol{e}}_y$ plane no matter, if an in-plane or out-of-plane motion is aimed. However, this approach requires much more ΔV for out-of-plane motions than if all three axes would be used for controlling. On the other hand, this approach reduces the required attitude control maneuvers to be performed by the satellites. One can clearly distinguish between the first phase, where the target positions are reached and the latter part where only maintenance is performed. Control in $\hat{\boldsymbol{e}}_x$ is mainly active during the first phase, whereas control in $\hat{\boldsymbol{e}}_y$ direction is continuously required to counteract the influence of perturbations. One can further see that the thrust profiles are limited to the maximum thrust that has been specified in the controller configuration (and which corresponds to realistic values for CubeSats, cf. Chap. 4.5).

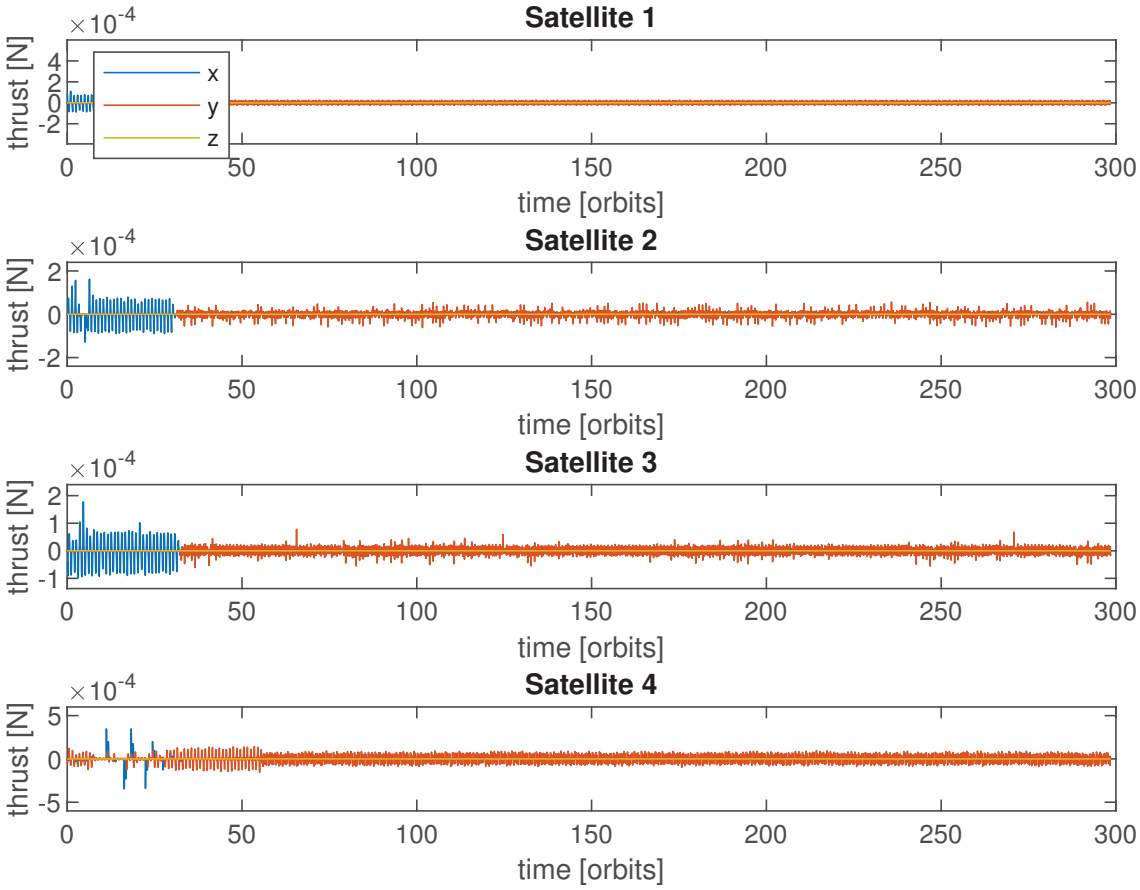


Figure 5.23: Thrust profile of the four satellites in EH frame over time using *Plant Inversion-Based Lyapunov Control Combined with a Reference Governor* to acquire the CWF – 3:1.

Summary

All simulated controllers drive the satellites to the target formation. So all controllers show convergence. To further compare the control approaches, in addition to the presented graphs in the previous sections, the required total ΔV for this Cartwheel Formation – 3:1 acquisition maneuver has been computed. The ΔV values for the four satellites for both control approaches are listed in Tab. 5.6. One can clearly see that the *Plant Inversion-Based Lyapunov Control Combined with a Reference Governor* requires significantly more ΔV for each individual satellite as well as for all satellites together, compared to the *MPC-Based Formation Control*. Especially Satellite 4, which performs the out-of-plane motion requires nearly double the amount of ΔV. This highlights the lower efficiency of the control approach for out-of-plane motions.

Table 5.6: Total ΔV consumption in [m/s] for CWF – 3:1 acquisition maneuver for the different control approaches.

Satellite	MPC	ROE
1	3.27	4.57
2	2.56	3.82
3	2.60	3.77
4	5.91	10.34

In addition to the ΔV consumption, the time until the target state is reached plays a significant role, too. The *MPC-Based Formation Control* requires up to 290 orbits, whereas *Plant Inversion-Based Lyapunov Control Combined with a Reference Governor* only requires up to 60 orbits. This is nearly five times faster. On the other hand, it requires between 1.4 and 1.7 times more ΔV. In addition, it should be mentioned that the MPC-Based controller causes much more computational load than the Lyapunov controller, which is a purely analytical solution.

In summary, both control approaches are suitable for this scenario and can consider the CubeSat-dependent requirements. The choice among them may depend on three factors: the time the target should be reached, the available ΔV and the available computational resources on the satellite.

5.4.1.2 Maintenance Phase

In the maintenance phase of the the CWF – 3:1, the initial and target states for each satellite are identical. They are given in Tab. 5.3.

MPC-Based Formation Control

Fig. 5.24 shows the motion of the four satellites during this maintenance maneuver in the EH frame. The control is computed using the *MPC-Based Formation Control* (cf. Chap. 3.2) to maintain the CWF – 3:1 formation. It is simulated for 14 days or 210 orbits. From Fig. 5.24 one can clearly see that there is only slight deviation from the target state, which means that the controller performs the maintenance task well.

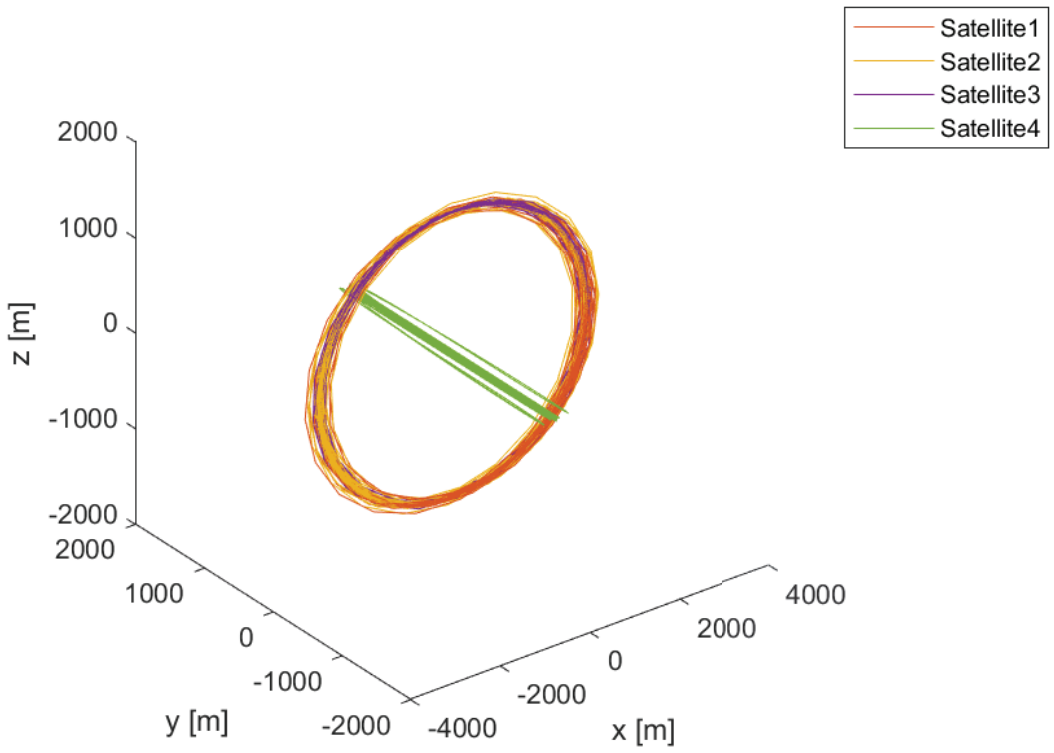


Figure 5.24: Motion of the four satellites in EH frame using *MPC-Based Formation Control* to maintain the CWF – 3:1 in 3D.

Fig. 5.25 shows the relative error during the maintenance maneuver in ROEs over time for the four simulated satellites. The relative errors of all satellites remain close to zero, thus the target formation is maintained. However, minor deviations arise from time to time, especially in the $\partial\lambda$ component (which corresponds to the $\hat{\boldsymbol{e}}_x$ axis). However, these deviations of up to 200 m are comparably small with respect to the formation baseline (cf. Tab. 5.3) and are compensated quickly.

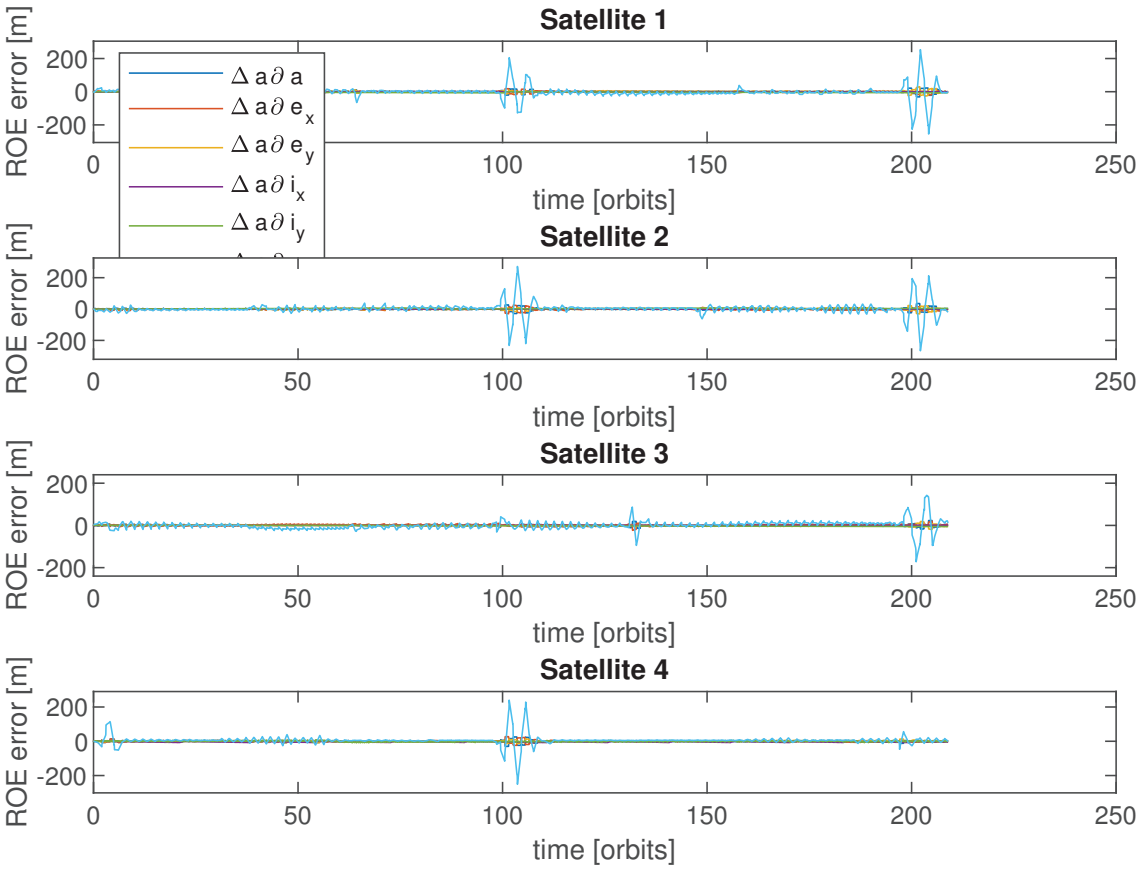


Figure 5.25: Relative error of the four satellites in ROEs over time using *MPC-Based Formation Control* to maintain the CWF – 3:1.

Fig. 5.26 shows the thrust activity and distribution of the four satellites in EH frame over time. One can clearly see that Satellite 4, which performs the out-of-plane motion, has a strong thrust contribution along the $\hat{e}_y$ axis, whereas the other three satellites are mainly active in the $\hat{e}_x$-$\hat{e}_z$ plane. One can further see that the thrust profiles are limited to the maximum thrust that has been specified in the MPC configuration (and which corresponds to realistic values for CubeSats, cf. Chap. 4.5). Furthermore, one can see that the thrust phases (the "spikes" in the figures, which are actually phases of several minutes duration) are distributed at thrust-optimal positions as computed by the MPC.

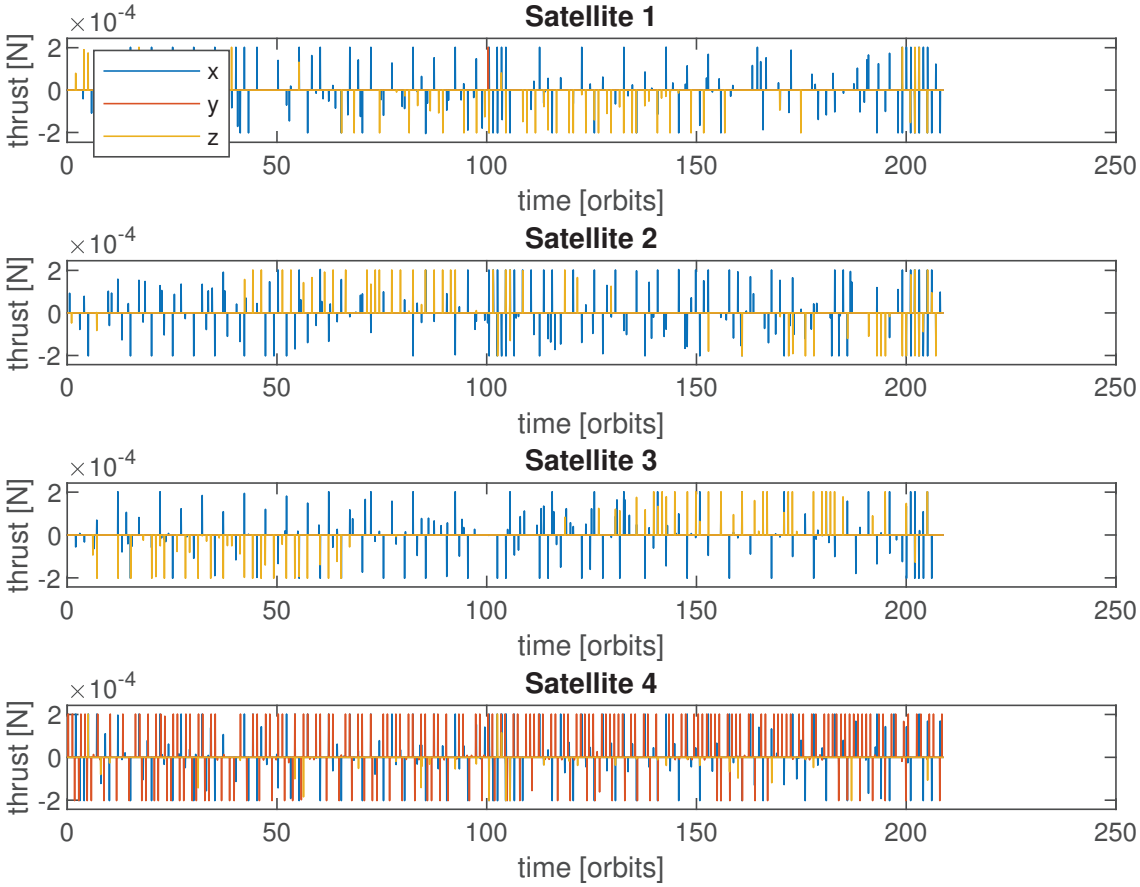


Figure 5.26: Thrust profile of the four satellites in EH frame over time using *MPC-Based Formation Control* to maintain the CWF – 3:1.

Plant Inversion-Based Lyapunov Control Combined with a Reference Governor

Fig. 5.27 shows the motion of the four satellites during this maintenance maneuver in the EH frame. The control is computed using the *Plant Inversion-Based Lyapunov Control Combined with a Reference Governor* (cf. Chap. 3.4) to maintain the CWF – 3:1 formation. It is simulated for 14 days or 210 orbits. From Fig. 5.24 one can clearly see that there is almost no deviation from the target state, which means that the controller performs the maintenance task remarkably well.

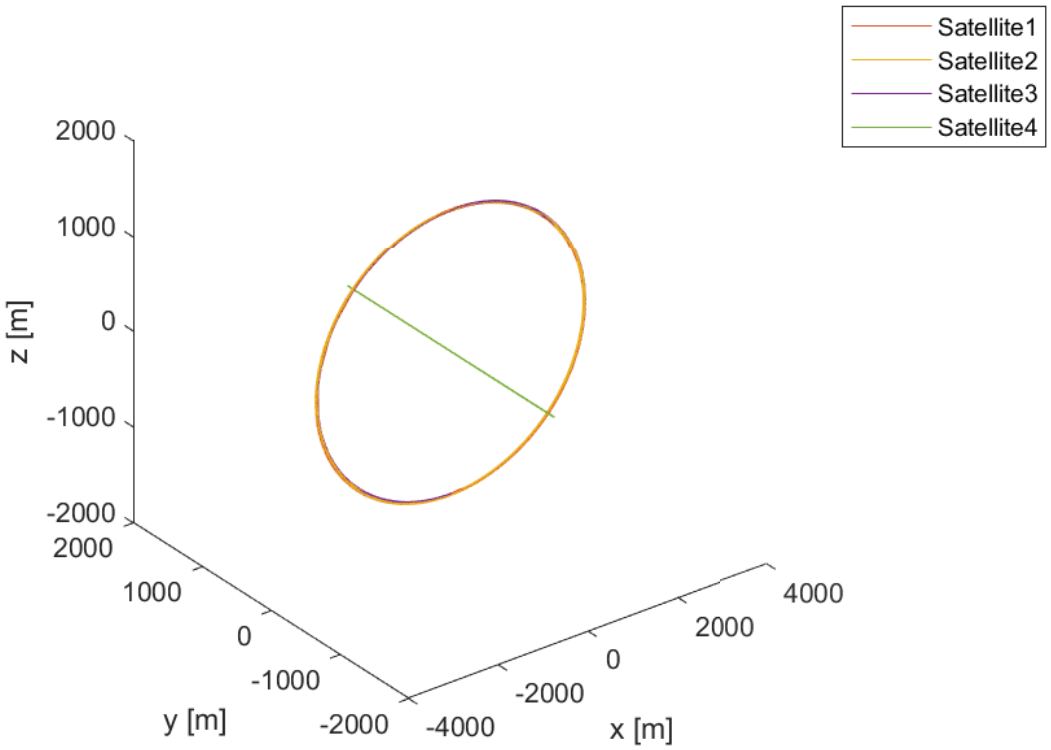


Figure 5.27: Motion of the four satellites in EH frame using *Plant Inversion-Based Lyapunov Control Combined with a Reference Governor* to maintain the CWF – 3:1 in 3D.

Fig. 5.28 shows the relative error during the maintenance maneuver in ROEs over time for the four simulated satellites. The relative errors of all satellites remain close to zero, thus the target formation is maintained. All deviations that arise from orbit perturbations are compensated, such that the error remains below 20 m for all satellites for the whole duration of 14 days.

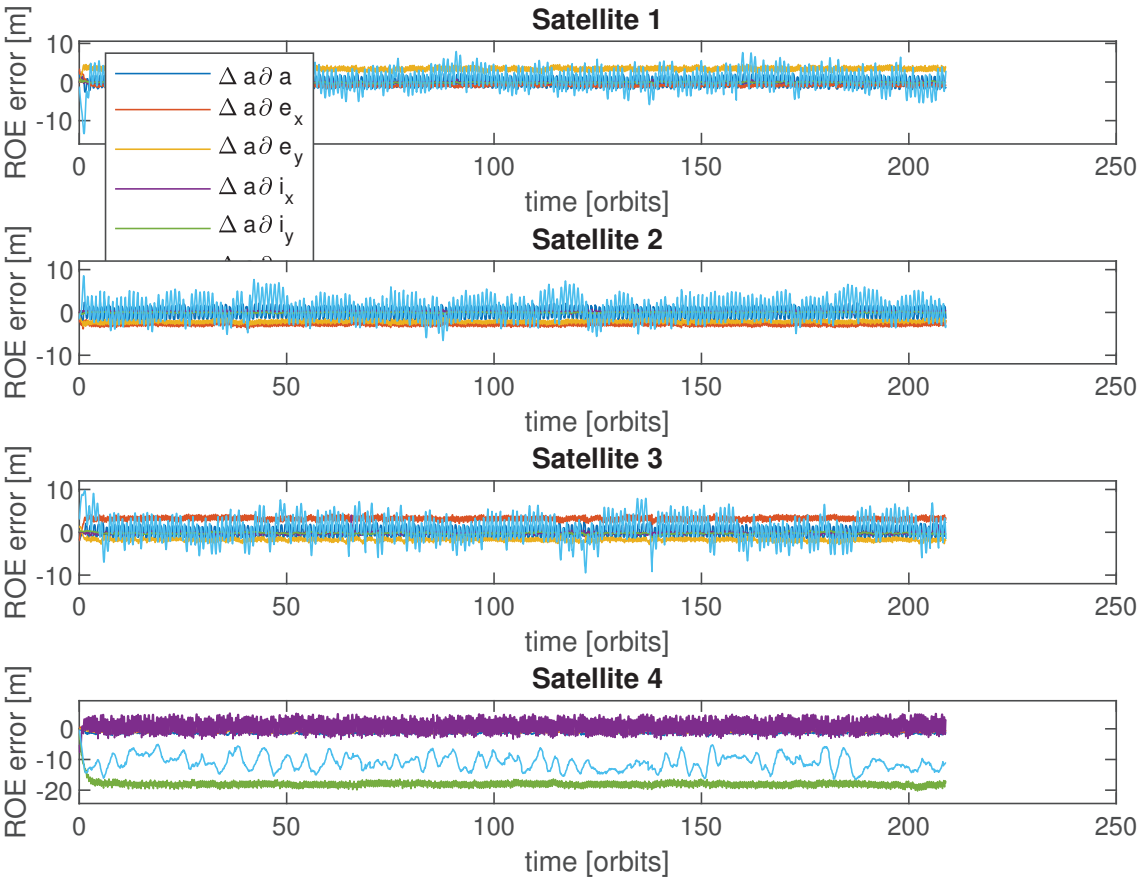


Figure 5.28: Relative error of the four satellites in ROEs over time using *Plant Inversion-Based Lyapunov Control Combined with a Reference Governor* to maintain the CWF – 3:1.

Fig. 5.29 shows the thrust activity and distribution of the four satellites in EH frame over time. Using this control approach, thrust is only applied in the $\hat{\boldsymbol{e}}_x$-$\hat{\boldsymbol{e}}_y$ plane no matter, if an in-plane or out-of-plane motion is aimed. However, this approach requires much more ΔV for out-of-plane motions than if all three axes would be used for controlling. On the other hand, this approach reduces the required attitude control maneuvers to be performed by the satellites. One can clearly see that during this maintenance phase, control is mainly required in $\hat{\boldsymbol{e}}_y$ direction and is almost continuously active to counteract the influence of perturbations. One can further see that the thrust profiles are limited to the maximum thrust that has been specified in the controller configuration (and which corresponds to realistic values for CubeSats, cf. Chap. 4.5).

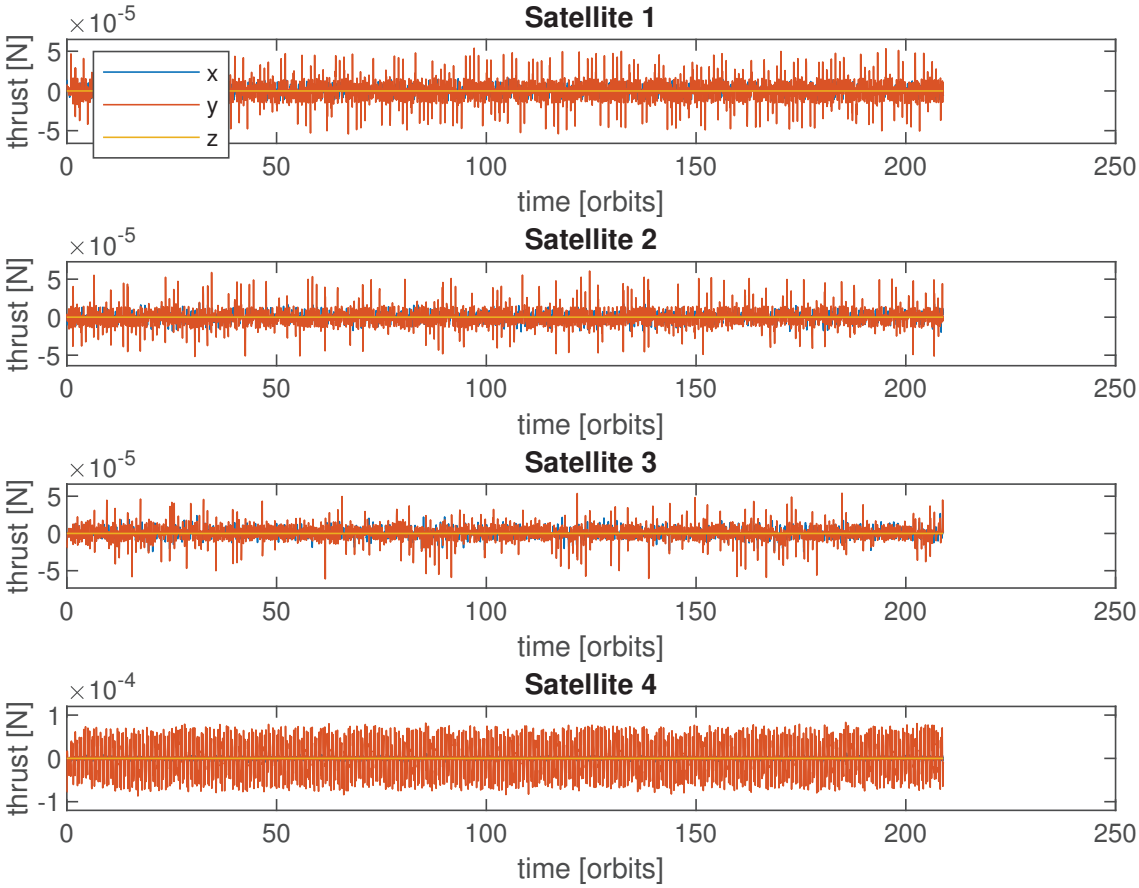


Figure 5.29: Thrust profile of the four satellites in EH frame over time using *Plant Inversion-Based Lyapunov Control Combined with a Reference Governor* to maintain the CWF – 3:1.

Summary

All simulated controllers maintain the satellites at the target formation. To further compare the control approaches, in addition to the presented graphs in the previous sections, the required total ΔV for this Cartwheel Formation – 3:1 maintenance maneuver has been computed. The ΔV values for the four satellites for both control approaches are listed in Tab. 5.7. The ΔV consumption of the compared control approaches differs slightly between the individual satellites, but remains within the same range. The overall ΔV consumption of the *Plant Inversion-Based Lyapunov Control Combined with a Reference Governor* is slightly higher than with the *MPC-Based Formation Control* approach. The differences however are not severe.

Table 5.7: Total ΔV consumption in [m/s] for CWF – 3:1 maintenance maneuver for the different control approaches.

Satellite	MPC	ROE
1	2.07	2.45
2	2.27	1.93
3	2.03	1.81
4	4.55	6.66

In addition to the ΔV consumption, the maximum tolerable relative error, plays a significant role, too. The relative error rises in some cases up to 200 m by using the *MPC-Based Formation Control*, whereas by using the *Plant Inversion-Based Lyapunov Control Combined with a Reference Governor* it only reaches up to 20 m. This is one whole order of magnitude difference. In addition, it should be mentioned that the MPC-Based controller causes much more computational load than the Lyapunov controller, which is a purely analytical solution.

In summary, both control approaches are suitable for this scenario and can fulfill the CubeSat-dependent requirements. The *Plant Inversion-Based Lyapunov Control Combined with a Reference Governor* performs better, because the maximum relative error is significantly lower and the requirements on computational resources on the satellite are much weaker.

5.4.2 Tetrahedron Formation

The THF scenario covers two phases, namely the acquisition and maintenance phase. Both are simulated and presented subsequently with a summary afterwards.

5.4.2.1 Acquisition Phase

The initial states from which the satellites start the maneuver of the THF acquisition phase as well as the target states are defined in Tab. 5.4 and repeated here for better readability.

THF initial and target state

	Initial State [m]				Target State [m]			
ROE	Sat 1	Sat 2	Sat 3	Sat 4	Sat 1	Sat 2	Sat 3	Sat 4
∂a	0	0	0	0	0	0	0	0
∂e_x	0	0	0	0	0	1250	0	0
∂e_y	0	0	0	0	0	0	1250	0
∂i_x	0	0	0	0	0	2500	-2500	0
∂i_y	0	0	0	0	0	2500	2500	0
∂u	5000	1666	-1666	-5000	5000	0.99	0.99	-5000

MPC-Based Formation Control

Fig. 5.30 and 5.31 show the motion of the four satellites during this acquisition maneuver in the EH frame. The control is computed using the *MPC-Based Formation Control* (cf. Chap. 3.2) to acquire the THF. The motion from the starting points on $\hat{\boldsymbol{e}}_{\boldsymbol{x}}$ axis towards the ellipse forming the THF consists of multiple spirals, which can clearly be seen from Fig. 5.31. This is the case, because the thrusters of the satellites are limited and the maneuver takes several days – roughly 25 days for this scenario – to be performed.

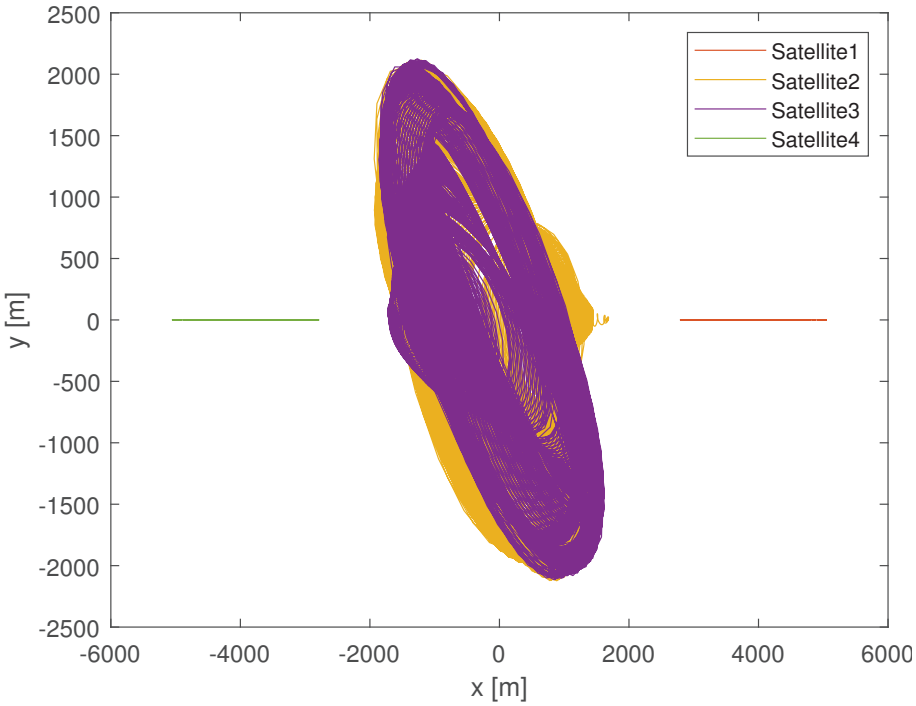


Figure 5.30: $\hat{e}_x$-$\hat{e}_y$ motion of the four satellites in EH frame using *MPC-Based Formation Control* to acquire the THF.

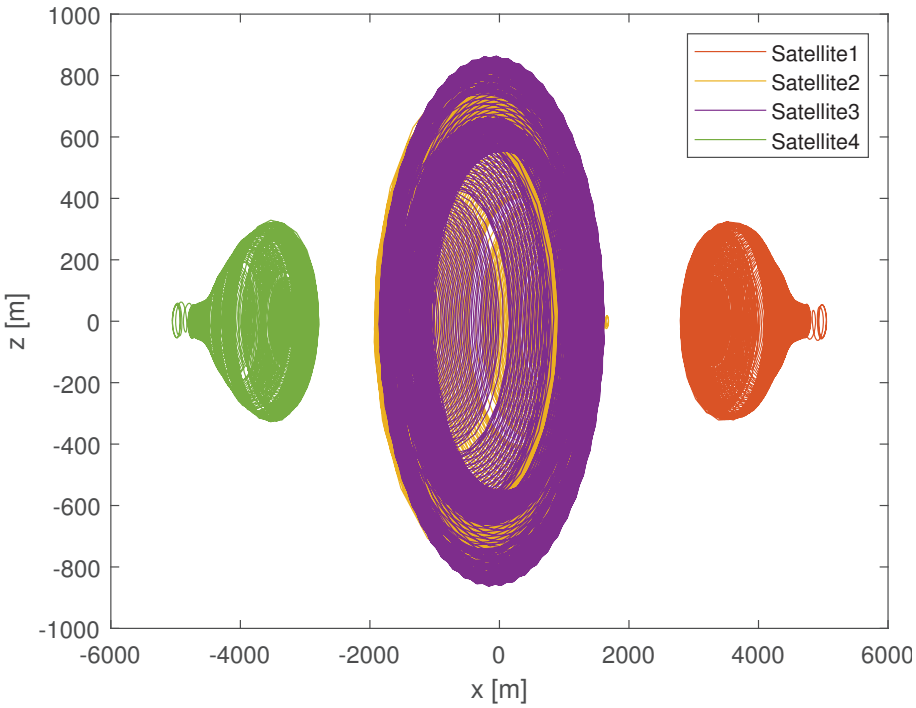


Figure 5.31: $\hat{e}_x$-$\hat{e}_z$ motion of the four satellites in EH frame using *MPC-Based Formation Control* to acquire the THF.

Fig. 5.32 shows the relative error between starting points and target points in ROEs over time for the four simulated satellites. The relative errors of all satellites converge to zero, thus the target formation is reached. The time required for reaching the target state varies between the four satellites from about 300 to about 400 orbits, which roughly corresponds to 18 to 27 days. The difference in the convergence time depends on the different initial conditions and the fact that two satellites (Satellite 2 and 3) have a different role and thus also a different target state which requires more time to be acquired.

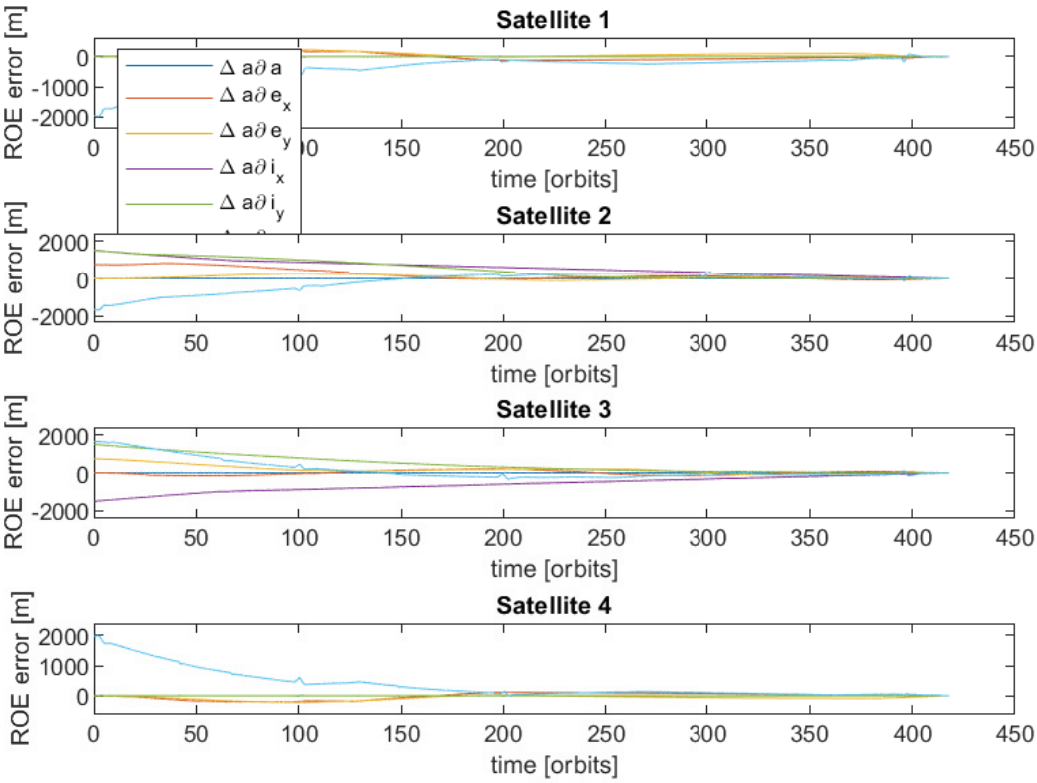


Figure 5.32: Relative error of the four satellites in ROEs over time using *MPC-Based Formation Control* to acquire the THF.

Fig. 5.33 shows the thrust activity and distribution of the four satellites in EH frame over time. One can clearly see that the Satellites 2 and 3, which perform the out-of-plane motion, have a strong thrust contribution along the $\hat{e}_y$ axis, whereas the other two satellites are mainly active in the $\hat{e}_x$-$\hat{e}_z$ plane. In addition, the Satellites 2 and 3 require much more thrust to acquire the desired formation than the Satellites 1 and 4, which is also due to the more complex motion (including out-of-plane motion) that these two satellites have to perform. In addition, a clear symmetry of Satellite 1 and 4 and Satellite 2 and 3 is visible in the thrust profile, which reflects the structure of the THF. One can further see that the thrust profiles are limited to the maximum thrust that has been specified in the MPC configuration (and which corresponds to realistic values for CubeSats, cf. Chap. 4.5). Further, one can see that the thrust phases (the "spikes" in the figures, which are actually phases of several minutes duration) are distributed at thrust-optimal positions as computed by the MPC.

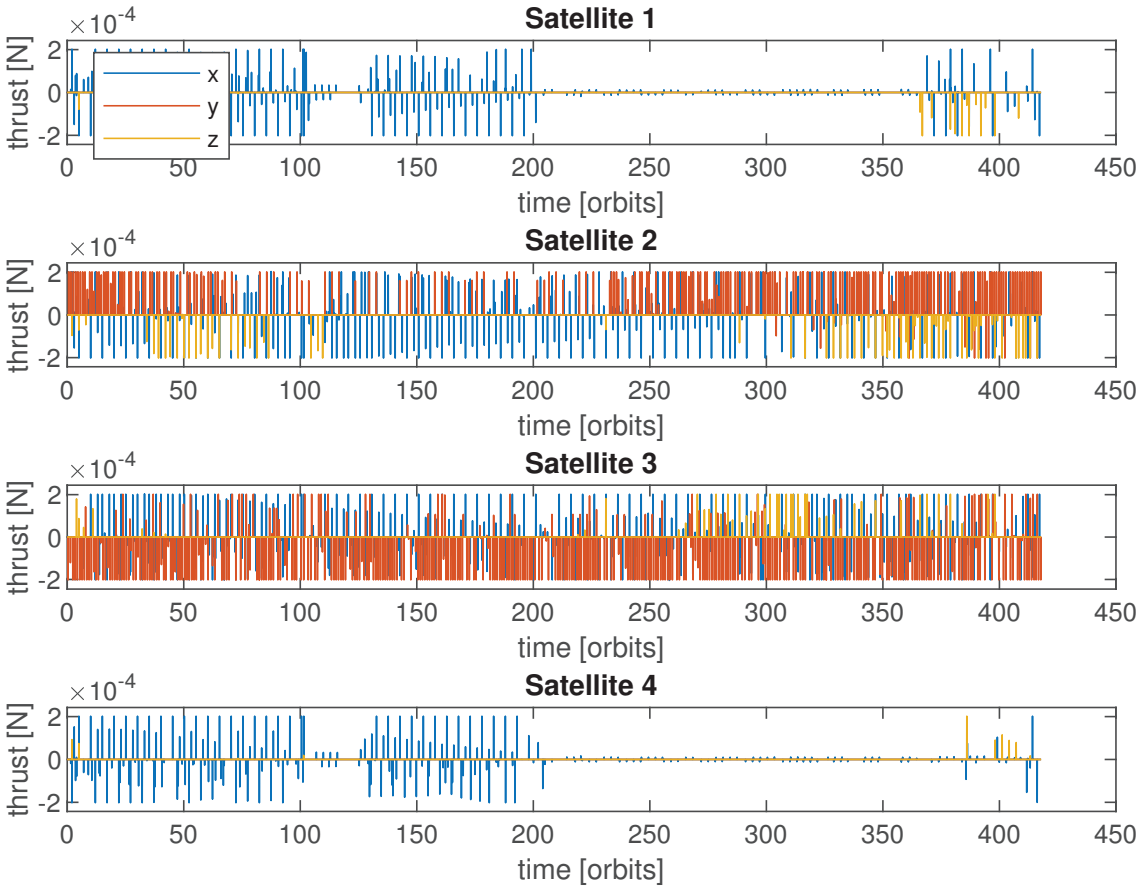


Figure 5.33: Thrust profile of the four satellites in EH frame over time using *MPC-Based Formation Control* to acquire the THF.

Plant Inversion-Based Lyapunov Control Combined with a Reference Governor

Fig. 5.34 and 5.35 show the motion of the four satellites during this acquisition maneuver in the EH frame. The control is computed using the *Plant Inversion-Based Lyapunov Control Combined with a Reference Governor* (cf. Chap. 3.4) to acquire the THF formation. The motion from the starting points on $\hat{\boldsymbol{e}}_{\boldsymbol{x}}$ axis towards the ellipse forming the THF formation is formed by multiple spirals, which can clearly be seen in Fig. 5.35. This is the case, because the thrusters of the satellites are limited and the maneuver takes several days – roughly 3 days for this scenario – to be performed.

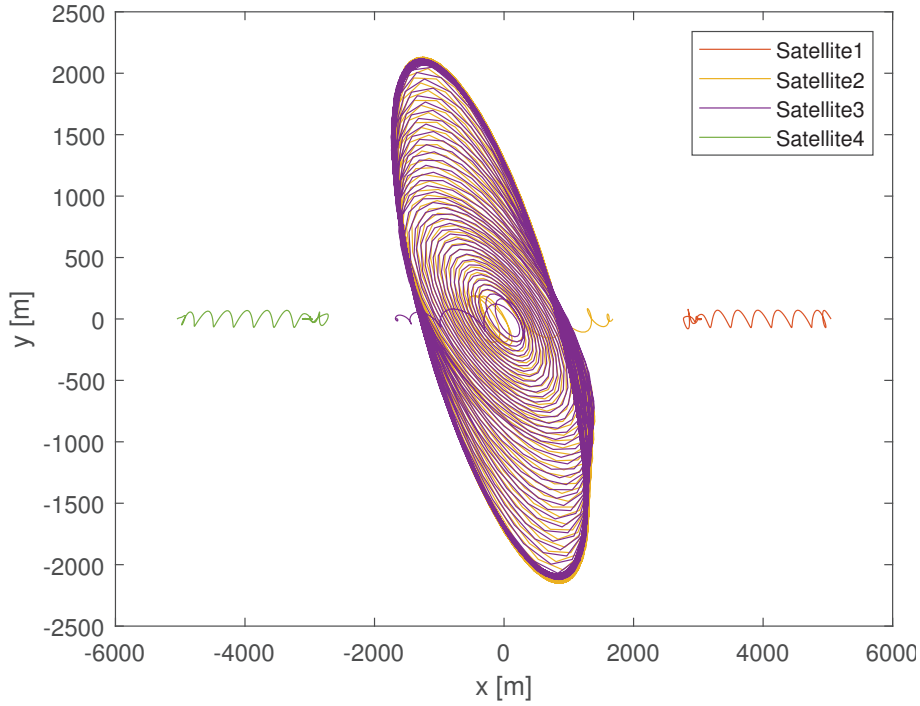


Figure 5.34: $\hat{\boldsymbol{e}}_{\boldsymbol{x}}$-$\hat{\boldsymbol{e}}_{\boldsymbol{y}}$ motion of the four satellites in EH frame using *Plant Inversion-Based Lyapunov Control Combined with a Reference Governor* to acquire the THF.

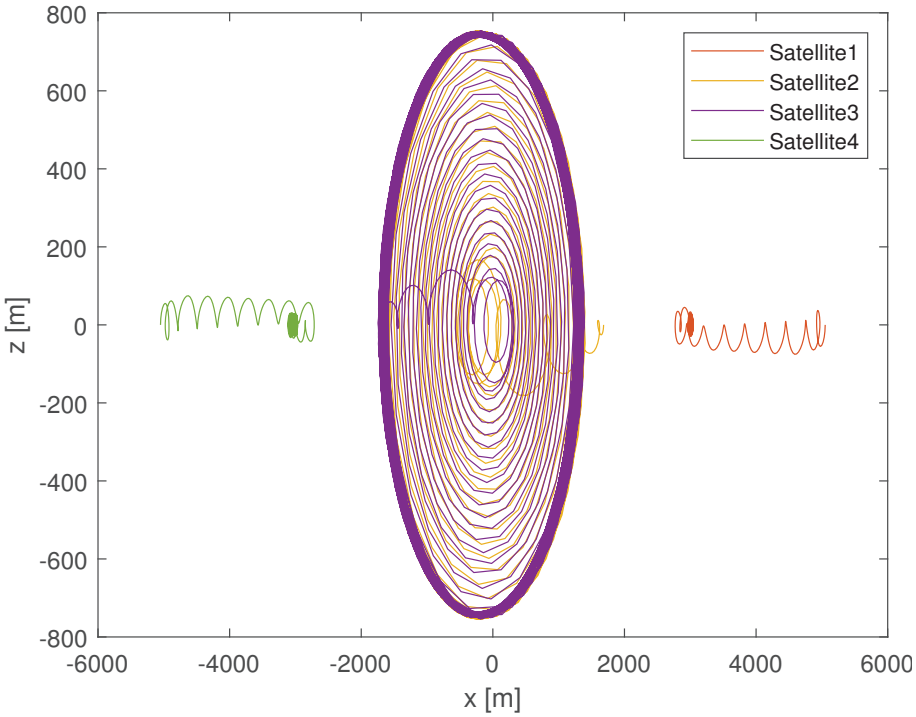


Figure 5.35: $\hat{\boldsymbol{e}}_{\boldsymbol{x}}$-$\hat{\boldsymbol{e}}_{\boldsymbol{z}}$ motion of the four satellites in EH frame using *Plant Inversion-Based Lyapunov Control Combined with a Reference Governor* to acquire the THF.

Fig. 5.36 shows the relative error between the starting points and the target points in ROEs over time for the four simulated satellites. The relative errors of all satellites converge to zero, thus the target formation is reached. The time required for the four satellites to reach the target state varies from about 20 to about 50 orbits, which roughly corresponds to 2 to 3 days. The difference in the convergence time depends on the different initial conditions and the fact that two satellites (Satellite 2 and 3) have a different role and thus also a different target state which requires more time to be acquired.

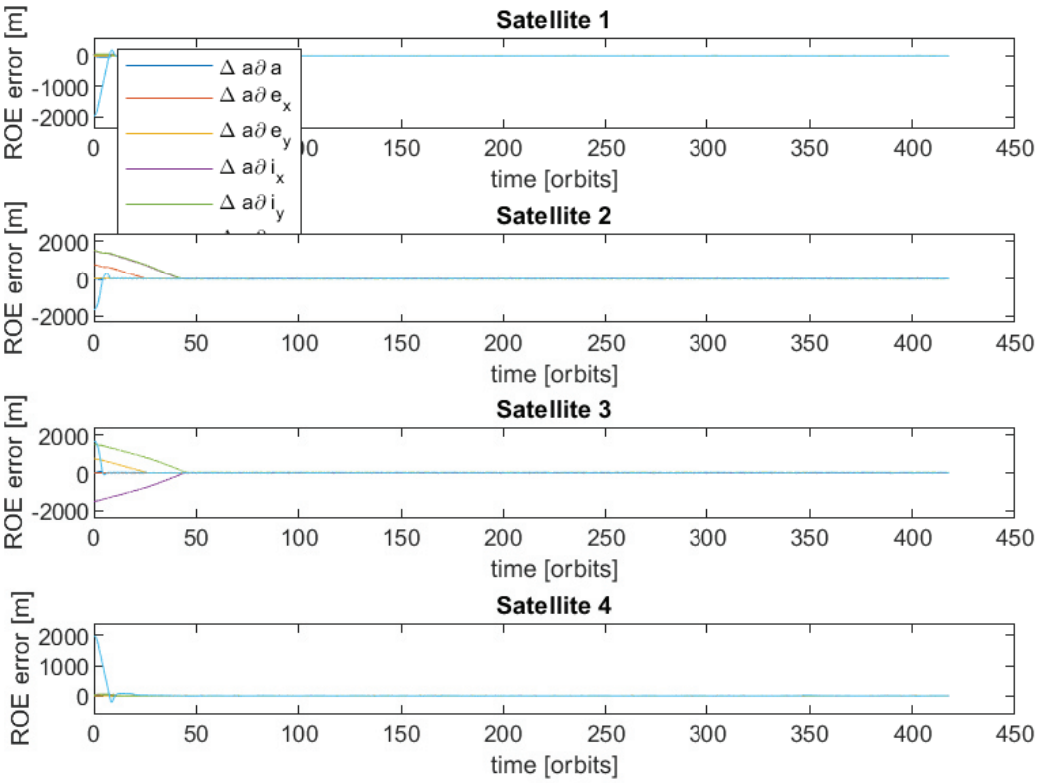


Figure 5.36: Relative error of the four satellites in ROEs over time using *Plant Inversion-Based Lyapunov Control Combined with a Reference Governor* to acquire the THF.

Fig. 5.37 shows the thrust activity and distribution of the four satellites in EH frame over time. Using this control approach, thrust is only applied in the $\hat{\boldsymbol{e}}_x$-$\hat{\boldsymbol{e}}_y$ plane no matter, if an in-plane or out-of-plane motion is aimed at. However, this approach requires much more ΔV for out-of-plane motions than if all three axes were used for controlling. On the other hand, this approach reduces the required attitude control maneuvers to be performed by the satellites. One can clearly see that the Satellites 2 and 3 require much more thrust to acquire the desired formation than the Satellites 1 and 4. After reaching the target state, the required thrust of those two satellites is much higher. Again, the symmetry of the formation with respect to these satellites can be recognized in the thrust profile. One can further see that the thrust profiles are limited to the maximum thrust that has been specified in the controller configuration and that corresponds to realistic values for CubeSats (cf. Chap. 4.5).

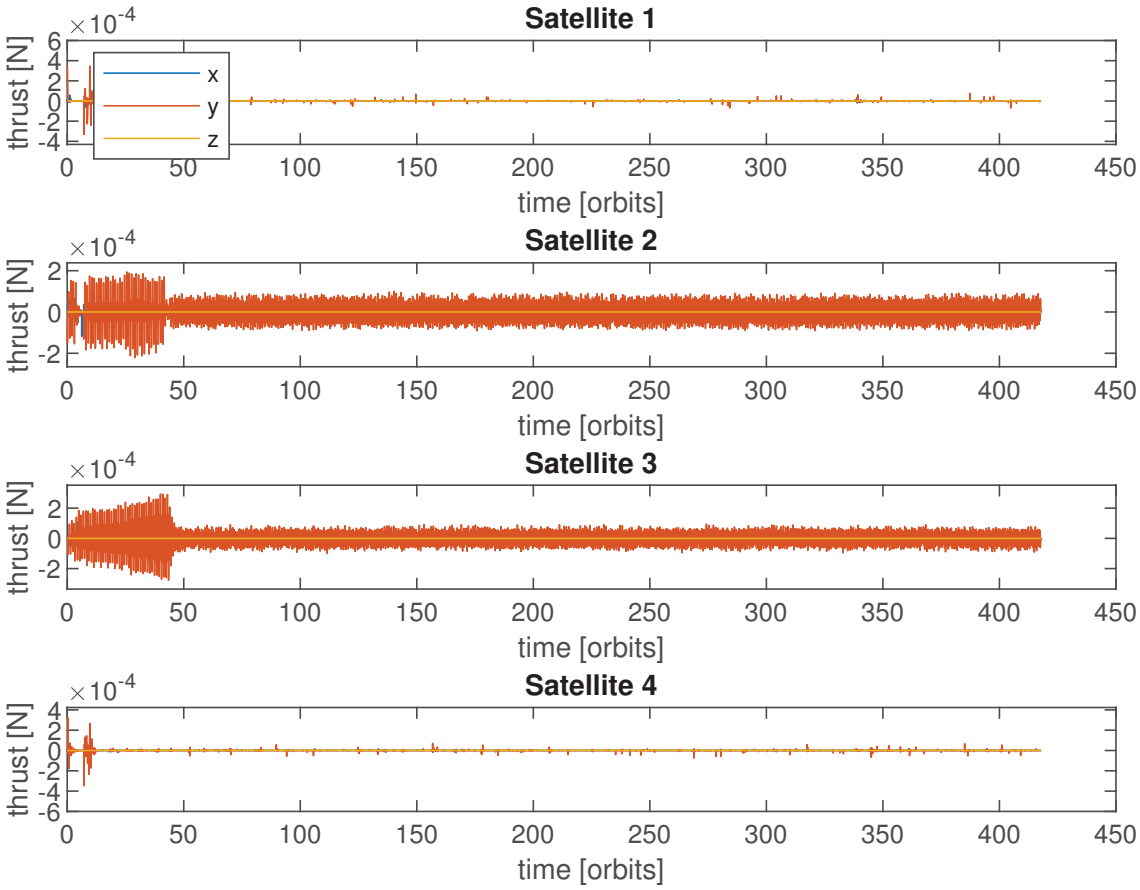


Figure 5.37: Thrust profile of the four satellites in EH frame over time using *Plant Inversion-Based Lyapunov Control Combined with a Reference Governor* to acquire the THF.

Summary

All simulated controllers drive the satellites to the target formation. So all controllers show convergence. To further compare the control approaches, in addition to the presented graphs in the previous sections, the required total ΔV for this Tetrahedron Formation acquisition maneuver has been computed. The ΔV values of the four satellites for both control approaches are listed in Tab. 5.8. One can clearly see that the *Plant Inversion-Based Lyapunov Control Combined with a Reference Governor* requires significantly more ΔV for Satellite 2 and 3, but significantly less for Satellite 1 and 4. For all satellites together, the ΔV consumption is nearly double, compared to the *MPC-Based Formation Control*. This highlights the lower efficiency of the Plant-Inversion-Based control approach for out-of-plane motions.

Table 5.8: Total ΔV consumption in [m/s] for THF acquisition maneuver for the different control approaches.

Satellite	MPC	ROE
1	1.80	0.84
2	7.19	15.36
3	9.15	16.78
4	1.54	0.98

In addition to the ΔV consumption, the time until the target state is reached plays a significant role, too. The *MPC-Based Formation Control* requires up to 400 orbits, whereas *Plant Inversion-Based Lyapunov Control Combined with a Reference Governor* only requires up to 50 orbits. This is nearly eight times faster. On the other hand, it requires about 1.8 times more ΔV in total. In addition, it should be mentioned that the MPC-Based controller causes much more computational load than the Lyapunov controller, which is a purely analytical solution.

In summary, both control approaches are suitable for this scenario and can meet the CubeSat-dependent requirements. The choice among them may depend on three factors: the time the target should be reached, the available ΔV and the available computational resources on the satellite.

5.4.2.2 Maintenance Phase

In the maintenance phase of the the THF, the initial and target states for each satellite are identical. They are given in Tab. 5.4.

MPC-Based Formation Control

Fig. 5.38 shows the motion of the four satellites during this maintenance maneuver in the EH frame. The control is computed using the *MPC-Based Formation Control* (cf. Chap. 3.2) to maintain the THF formation. It is simulated for 14 days or 210 orbits. From Fig. 5.38 one can clearly see that there is only slight deviation from the target state, which means that the controller performs the maintenance task well.

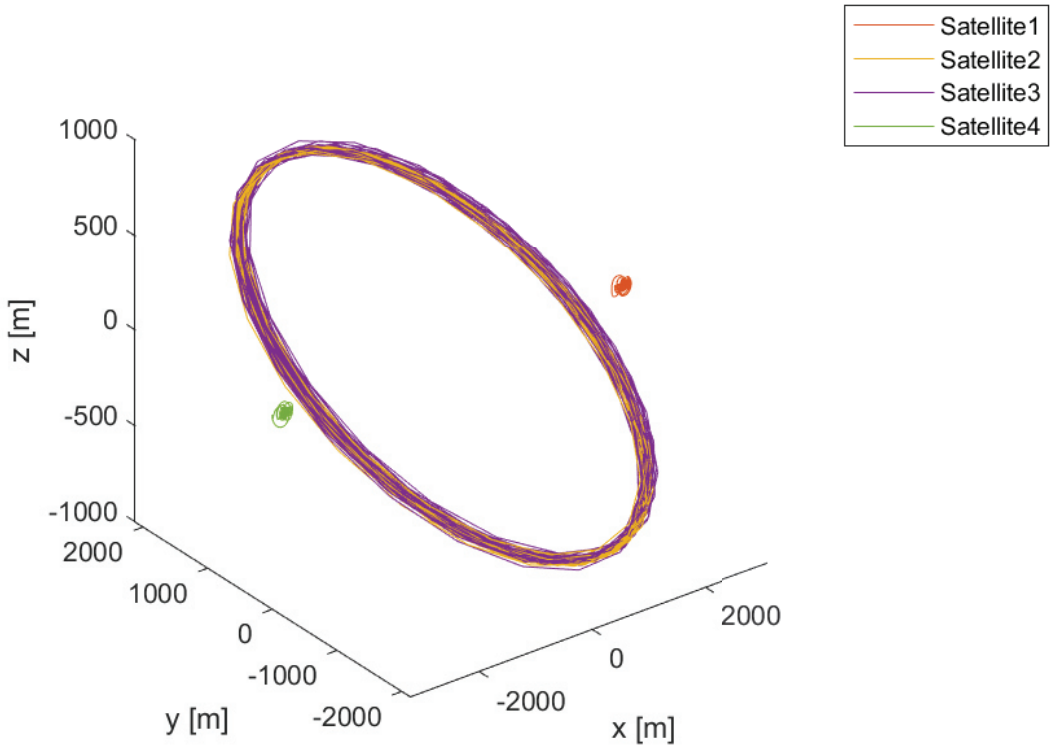


Figure 5.38: Motion of the four satellites in EH frame using *MPC-Based Formation Control* to maintain the THF in 3D.

Fig. 5.39 shows the relative error during the maintenance maneuver in ROEs over time for the four simulated satellites. The relative errors of all satellites remain close to zero, thus the target formation is maintained. However, minor deviations arise from time to time, especially in the $\partial\lambda$ component (which corresponds to the $\hat{\boldsymbol{e}}_x$ axis). However, these deviations of up to 200 m are comparably small with respect to the formation baseline (cf. Tab. 5.3) and can be compensated quickly.

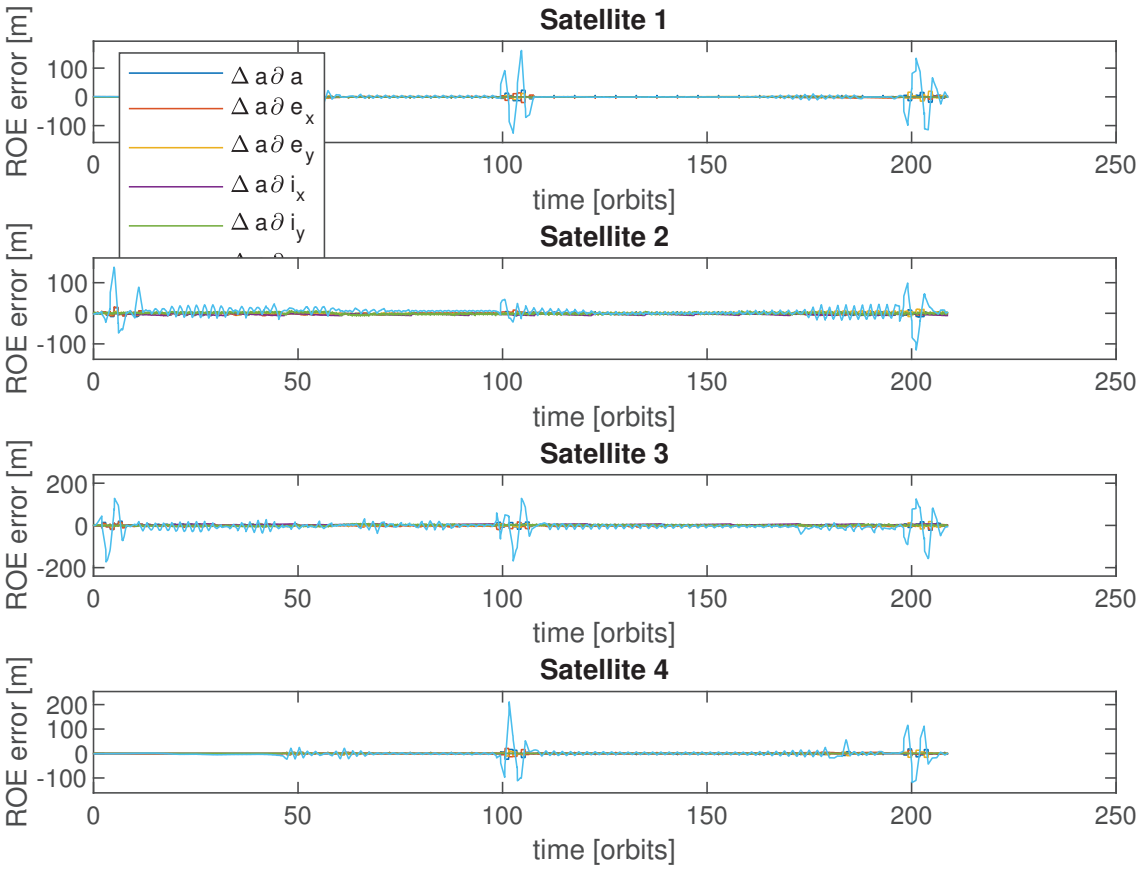


Figure 5.39: Relative error of the four satellites in ROEs over time using *MPC-Based Formation Control* to maintain the THF.

Fig. 5.40 shows the thrust activity and distribution of the four satellites in EH frame over time. One can clearly see that the Satellites 2 and 3, which perform the out-of-plane motion, have a strong thrust contribution along the $\hat{e}_y$ axis, whereas the other three satellites are mainly active in the $\hat{e}_x$-$\hat{e}_z$ plane. These two satellites also require much more thrust to maintain the desired formation than the Satellites 1 and 4. This is due to their out-of-plane part and the greater baseline of their target formation state. Further, the symmetry of the formation with respect to these satellites is also clearly visible in the thrust profile. In addition, one can see that the thrust profiles are limited to the maximum thrust that has been specified in the MPC configuration (and which corresponds to realistic values for CubeSats, cf. Chap. 4.5). Further, one can see that the thrust phases (the "spikes" in the figures, which are actually phases of several minutes duration) are distributed at thrust-optimal positions as computed by the MPC.

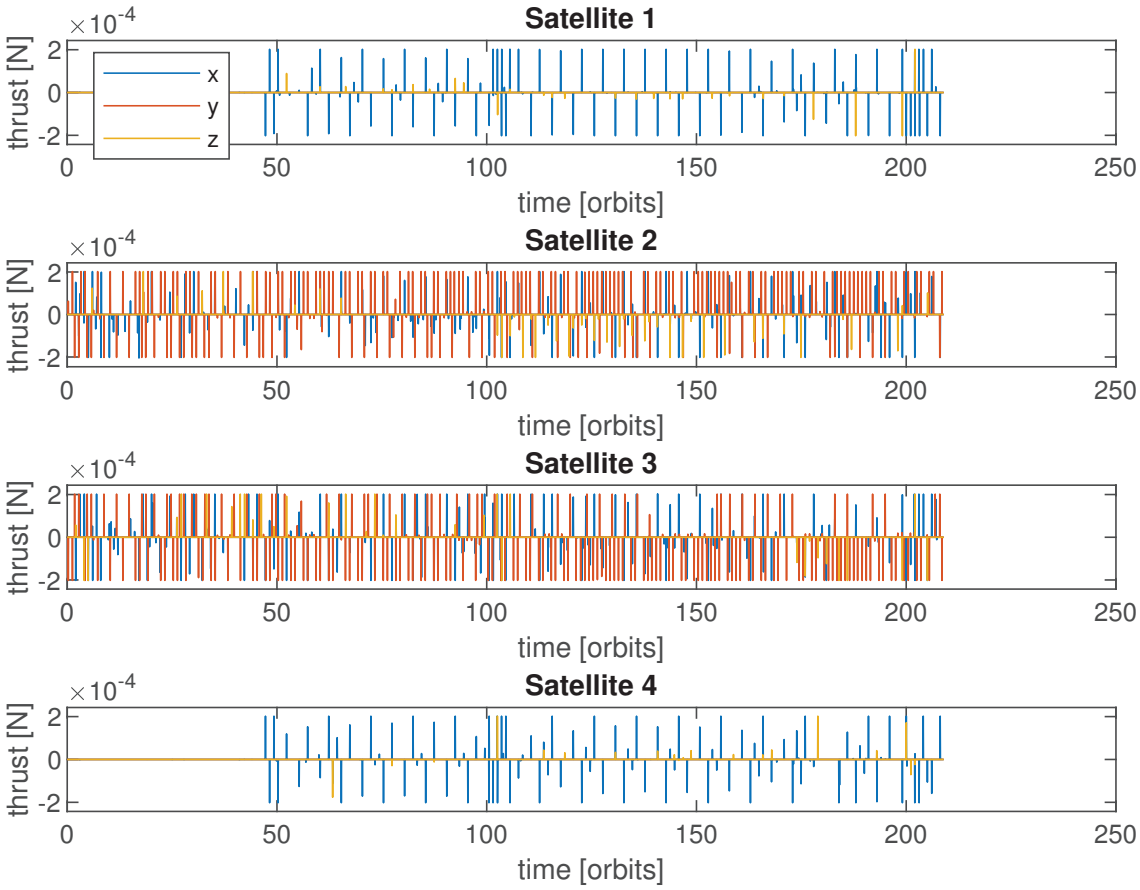


Figure 5.40: Thrust profile of the four satellites in EH frame over time using *MPC-Based Formation Control* to maintain the THF.

Plant Inversion-Based Lyapunov Control Combined with a Reference Governor

Fig. 5.41 shows the motion of the four satellites during this maintenance maneuver in the EH frame. The control is computed using the *Plant Inversion-Based Lyapunov Control Combined with a Reference Governor* (cf. Chap. 3.4) to maintain the THF. It is simulated for 14 days or 210 orbits. From Fig. 5.41 one can clearly see that there is almost no deviation from the target state, which means that the controller performs the maintenance task remarkably well.

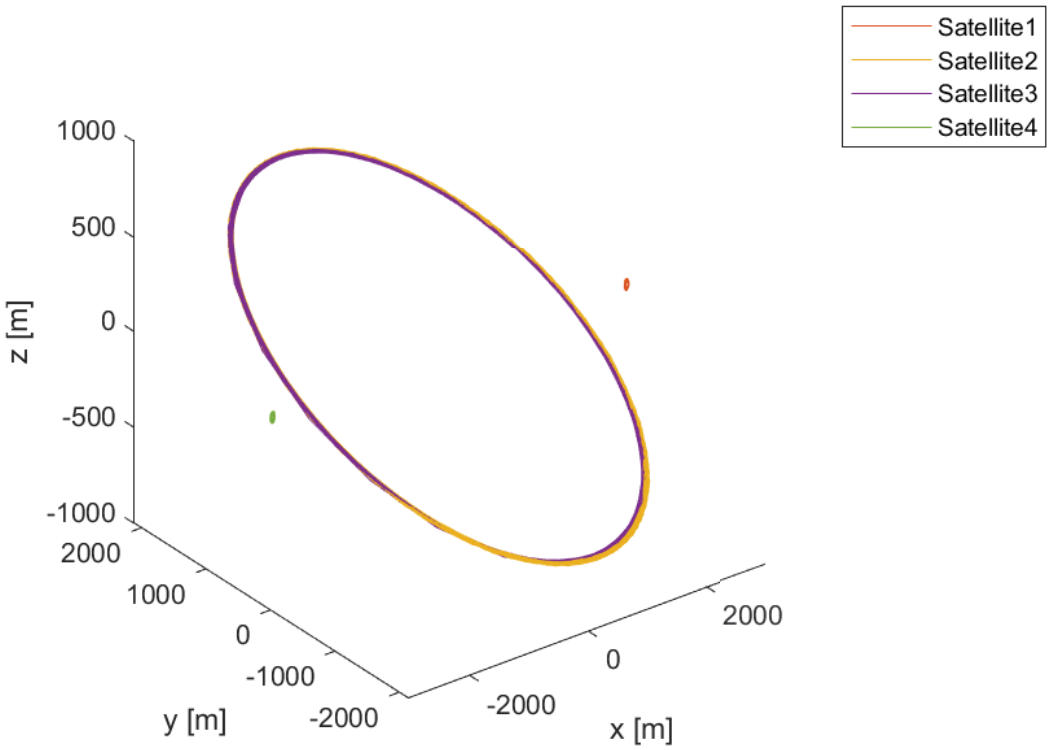


Figure 5.41: Motion of the four satellites in EH frame using *Plant Inversion-Based Lyapunov Control Combined with a Reference Governor* to maintain the THF in 3D.

Fig. 5.42 shows the relative error during the maintenance maneuver in ROEs over time for the four simulated satellites. The relative errors of all satellites remain close to zero, thus the target formation is maintained. All deviations that arise from orbit perturbations are compensated, so that the error remains below 40 m for all satellites for the whole duration of 14 days.

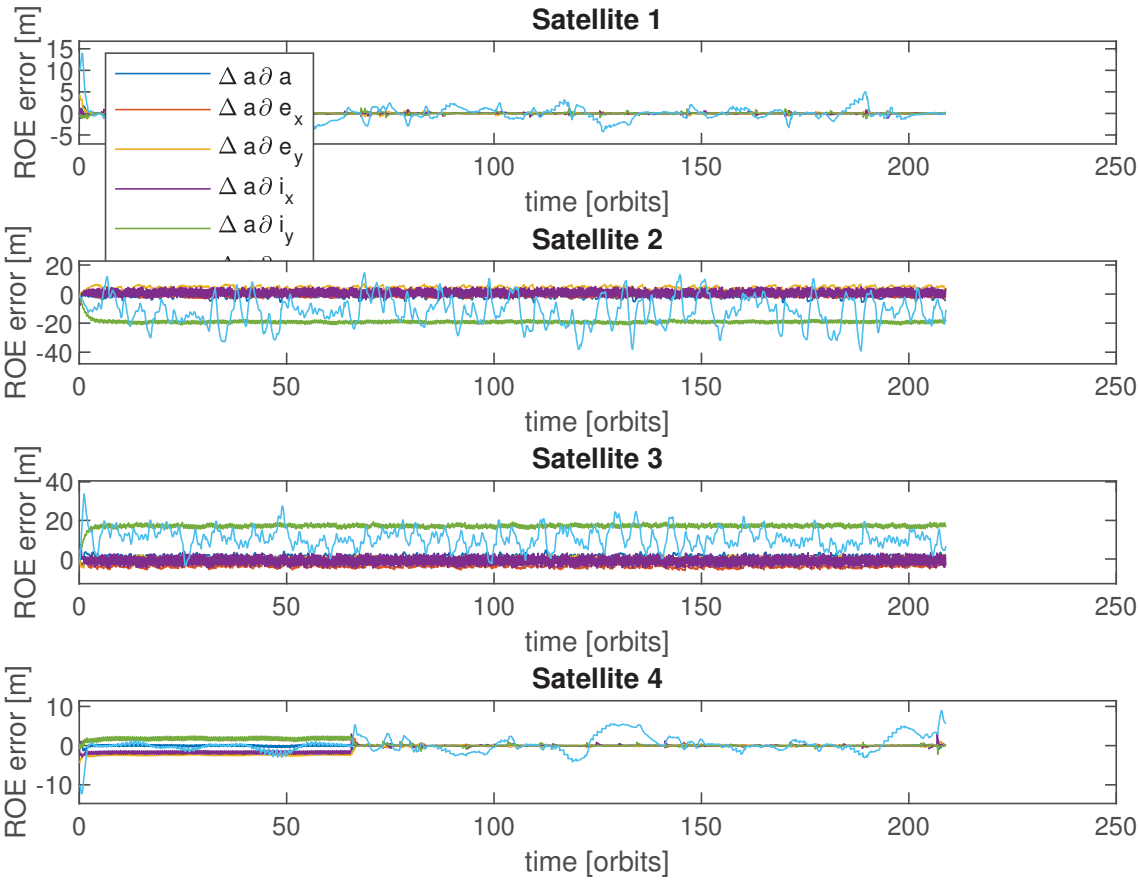


Figure 5.42: Relative error of the four satellites in ROEs over time using *Plant Inversion-Based Lyapunov Control Combined with a Reference Governor* to maintain the THF.

Fig. 5.43 shows the thrust activity and distribution of the four satellites in EH frame over time. Using this control approach, thrust is only applied in the $\hat{\boldsymbol{e}}_x$-$\hat{\boldsymbol{e}}_y$ plane no matter, if an in-plane or out-of-plane motion is aimed at. However, this approach requires much more ΔV for out-of-plane motions than if all three axes were used for controlling. On the other hand, this approach reduces the required attitude control maneuvers to be performed by the satellites. One can clearly see that during this maintenance phase, control is mainly required in $\hat{\boldsymbol{e}}_y$ direction and is almost continuously active to counteract the influence of perturbations. In addition, the Satellites 2 and 3 require much more thrust to maintain the desired formation than the Satellites 1 and 4. This is due to their out-of-plane part and the greater baseline of their target formation state. One can further see that the thrust profiles are limited to the maximum thrust that has been specified in the controller configuration (and which corresponds to realistic values for CubeSats, cf. Chap. 4.5).

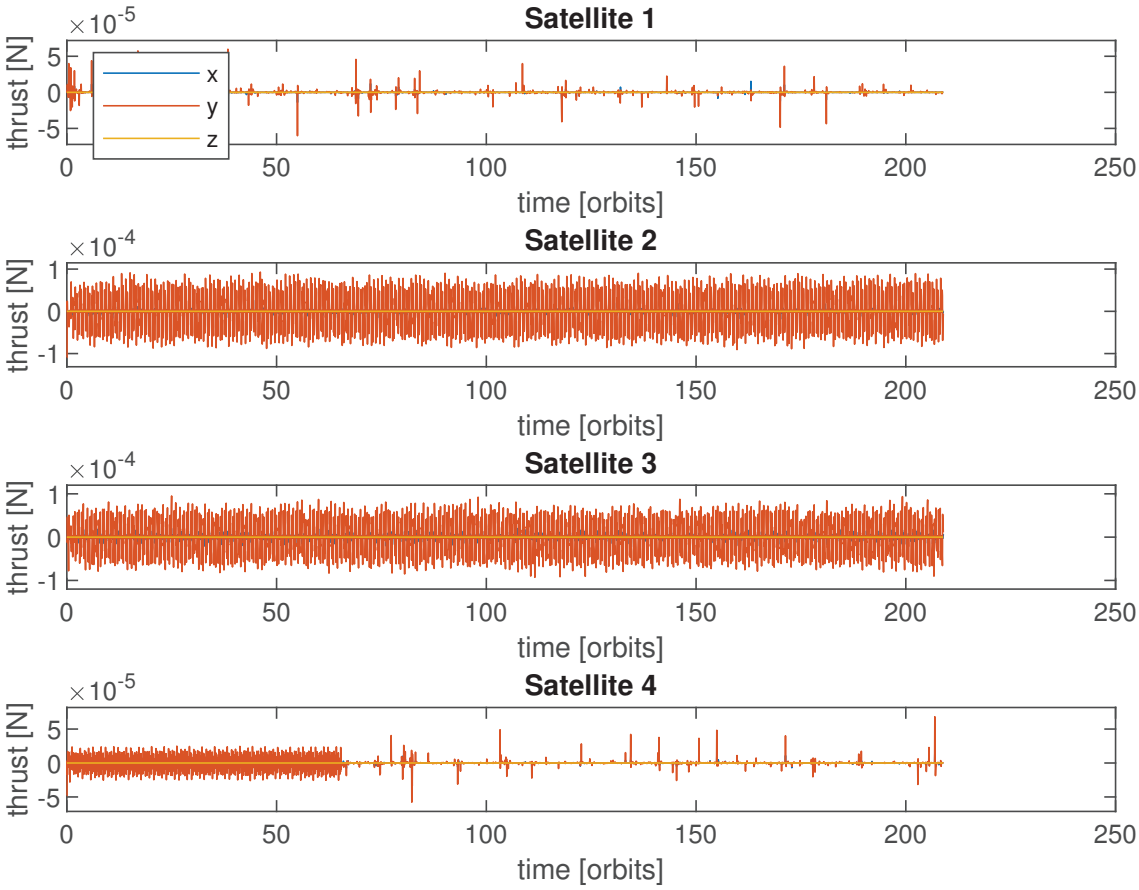


Figure 5.43: Thrust profile of the four satellites in EH frame over time using *Plant Inversion-Based Lyapunov Control Combined with a Reference Governor* to maintain the THF.

Summary

All simulated controllers maintain the satellites at the target formation. To further compare the control approaches, in addition to the presented graphs in the previous sections, the required total ΔV for this Tetrahedron Formation maintenance maneuver has been computed. The ΔV values for the four satellites for both control approaches are listed in Tab. 5.9. One can clearly see that the *Plant Inversion-Based Lyapunov Control Combined with a Reference Governor* requires significantly more ΔV for Satellite 2 and 3, but significantly less for Satellite 1 and 4. For all satellites together, the ΔV consumption is about 30 % higher, compared to the *MPC-Based Formation Control*. This highlights the lower efficiency of the *Plant-Inversion-Based Lyapunov Control* approach for out-of-plane motions.

Table 5.9: Total ΔV consumption in [m/s] for THF maintenance maneuver for the different control approaches.

Satellite	MPC	ROE
1	1.08	0.25
2	4.43	6.89
3	4.70	6.68
4	1.08	0.81

In addition to the ΔV consumption also the maximum tolerable relative error, plays a significant role. The relative error rises in some cases up to 200 m by using the *MPC-Based Formation Control*, whereas by using the *Plant Inversion-Based Lyapunov Control Combined with a Reference Governor* it only reaches up to 40 m. This is only one fifth. In addition, it should be mentioned that the *MPC-Based Formation Controller* causes much more computational load than the Lyapunov controller, which is a purely analytical solution.

In summary, both control approaches are suitable for this scenario and can meet the CubeSat-dependent requirements. The choice among them may depend on three factors: the maximum tolerable relative error, the available ΔV and the available computational resources on the satellite.

5.4.3 Spaceborne Distributed Telescope

Since the *Distributed Robust Consensus-Based Control* (cf. Chap. 3.3) cannot take thrust constrains into account – in contrast to the other two presented control approaches – it doesn't make sense to compare it directly with them. However, its advantage is to counteract disturbances (due to its robust nature) and to maintain a relative formation with high precision. Thus, a different scenario was chosen to demonstrate this control approach. These simulation results are part of the publication of Scharnagl et al. (2019).
This target formation is not force-free, which means that orbit dynamics are acting on the satellites and degrading the formation topology over time. Thus, after acquiring this formation, the controllers remain active not only for counteracting disturbances, but also for maintaining the formation. So we do not distinguish between acquisition and maintenance phase. The satellites start from an ATF with different spacing on the $\hat{\boldsymbol{e}}_x$ axis (v-bar). As target reference a static (in EH frame fixed and not moving) pentahedral formation as required for a spaceborne telescope is used. It shows a 10×10 m base in the $\hat{\boldsymbol{e}}_x$-$\hat{\boldsymbol{e}}_y$ plane and the tip is located at -10 m along $\hat{\boldsymbol{e}}_z$ axis.

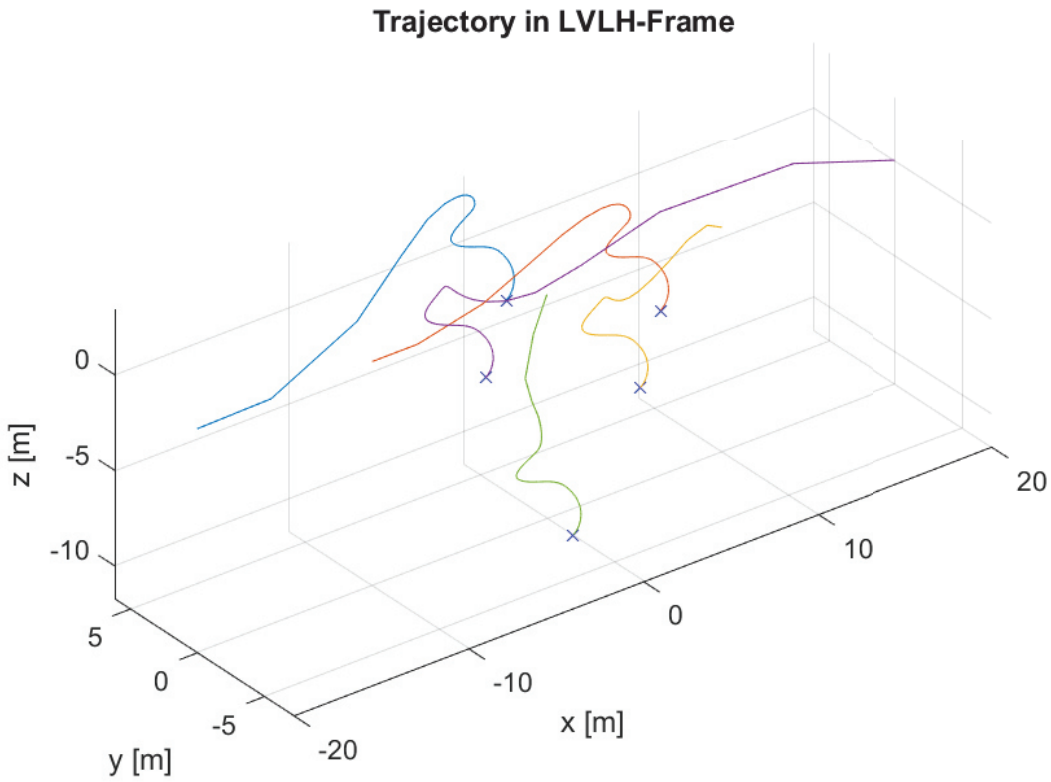


Figure 5.44: 3D plot of the trajectories of the five satellites subject to the presented consensus-based controller starting on v-bar $(-20, -10, 0, 10, 20$ m).

Fig. 5.44 to 5.47 show the trajectories of the satellites from the given starting points to the pentahedral target formation, both as 3D view and the different 2D views.

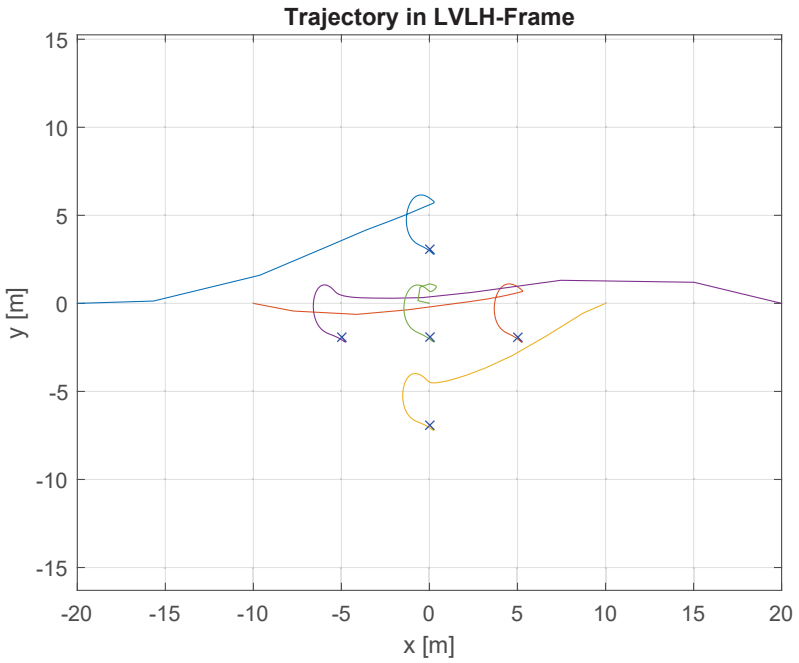


Figure 5.45: $\hat{e}_x$-$\hat{e}_y$ motion of the trajectories of the five satellites subject to the presented consensus-based controller starting on v-bar $(-20, -10, 0, 10, 20$ m).

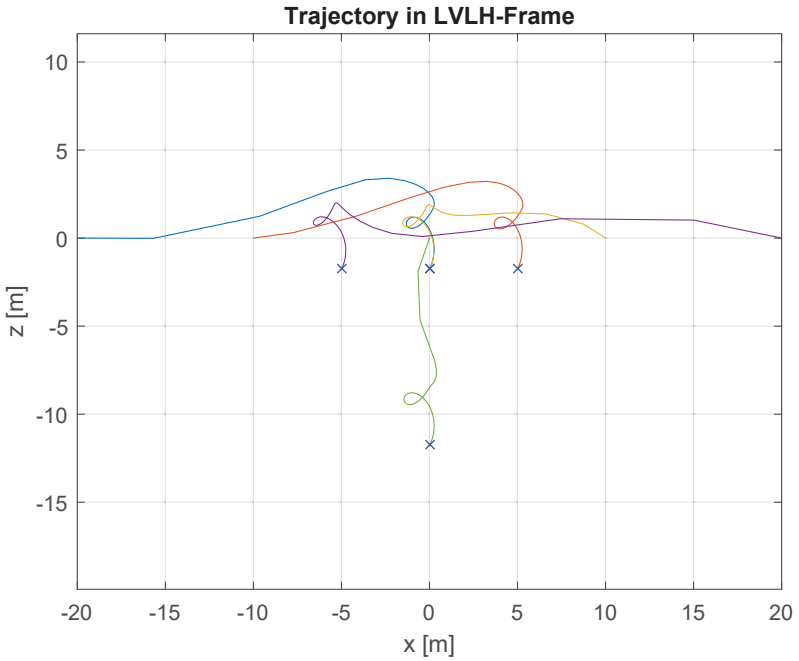


Figure 5.46: $\hat{e}_x$-$\hat{e}_z$ motion of the trajectories of the five satellites subject to the presented consensus-based controller starting on v-bar $(-20, -10, 0, 10, 20$ m).

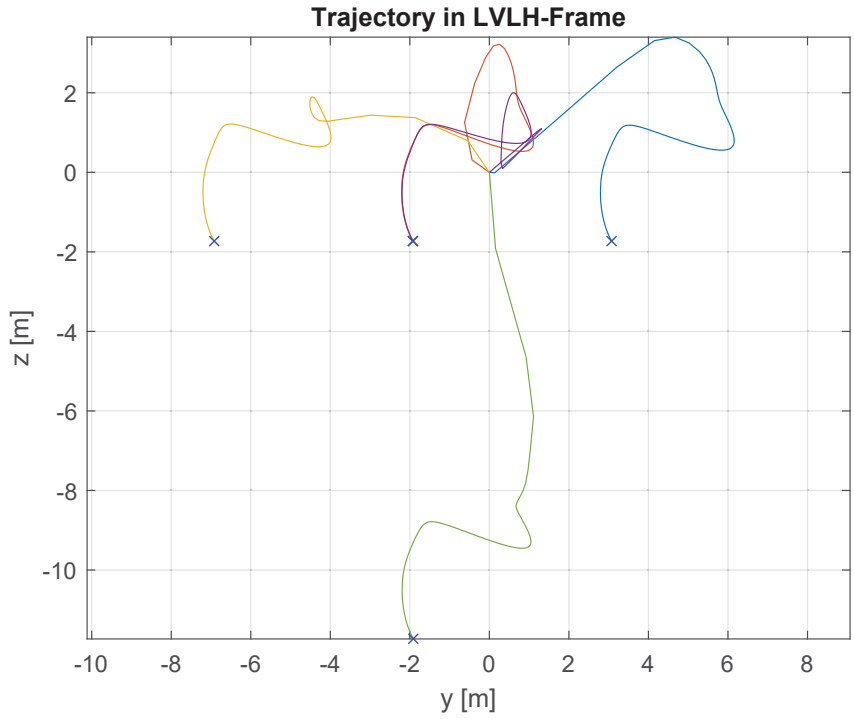


Figure 5.47: $\hat{e}_y$-$\hat{e}_z$ motion of the trajectories of the five satellites subject to the presented consensus-based controller starting on v-bar $(-20, -10, 0, 10, 20$ m).

From Fig. 5.44 to 5.47 one can clearly see the convergence of the satellites' positions from their starting points towards the desired target formation. During this transfer the controller counteracts both, the orbit dynamics given by the Hill's equations (Eq. 2.72) and the induced disturbances (thruster perturbations). There are no motions around the target points visible, which would be caused by any disturbances, which shows that the controller performs quite well with regards to disturbance rejection. Further, one can see that the last part of the the trajectories of all satellites are identical. Thus, the controller first tried to acquire the target formation and after reaching it, the formation drifted while remaining perfectly in shape. This is why the controller tries to acquire/maintain the relative formation, not the absolute positions.

Fig. 5.48 shows the control accelerations and the disturbance acting on Satellite 1 over the simulation time.

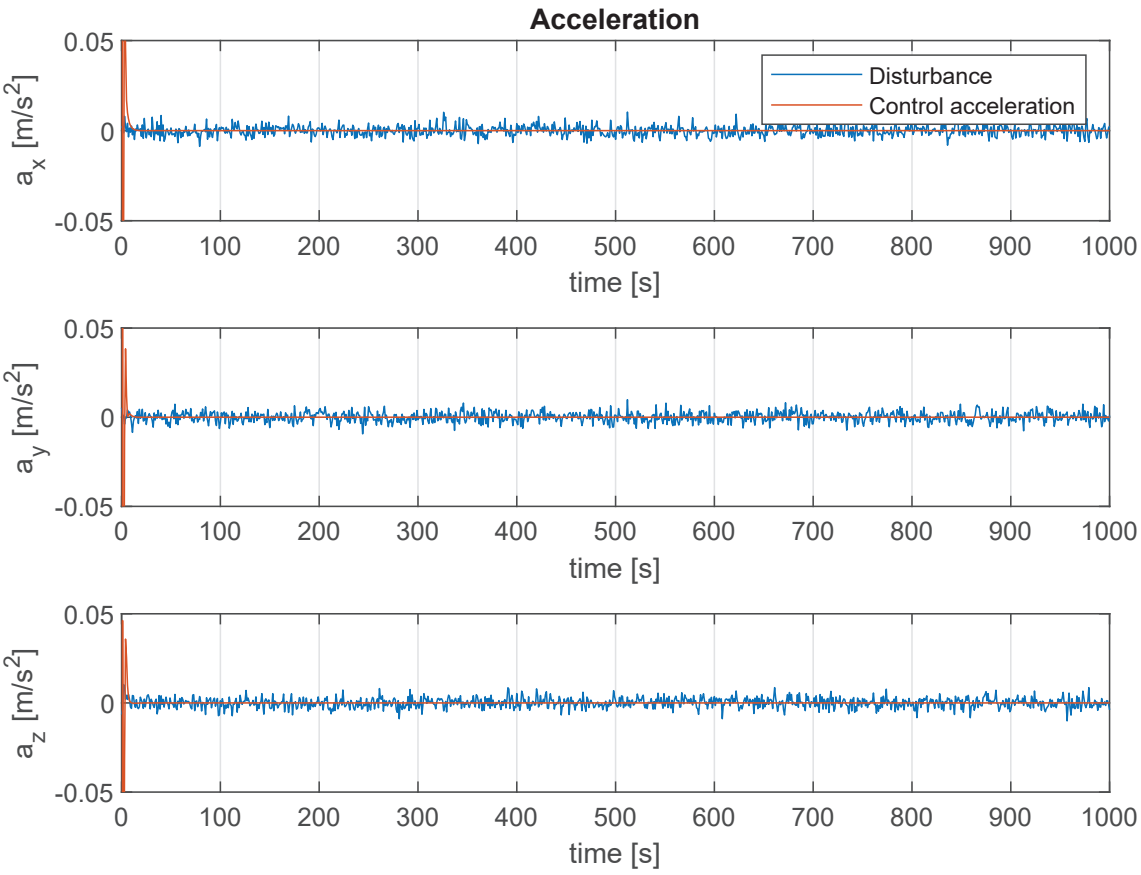


Figure 5.48: Accelerations acting on Satellite 1 during simulation time, namely disturbance and control acceleration.

As one can clearly see, most of the control acceleration is applied during the first seconds of the maneuver, which is mainly applied for changing the satellites' position and converging to the target states. The acquisition phase happens so quickly, since there is no limitation on the maximum allowed thrust. After that, the controllers remain active to force the satellites to stay on their target positions. The controllers are counteracting disturbances and – since the formations are not force-free – also orbit dynamics, which would cause a degradation of the formation within minutes. However, this formation maintenance requires much less control acceleration than the initial acquisition of the formation.

5.5 Preparation for In-Orbit Testing within the NetSat Mission

The developed control approaches should be – if they are applicable – implemented in the NetSat mission. The four NetSat satellites have been launched on September 28, 2020 into a LEO (Tab. 2.2 presents the NetSat orbit parameters) and are currently in commissioning phase. (See Chap. 2.3.2 for details on the NetSat mission.)

The on-board software of NetSat, namely on the AOCS, has been prepared to include different controllers. Thus a generic controller interface has been created and implemented. Fig. 5.49 shows the concept and the interconnections of this controller interface. It receives state measurements (position and velocity) from the navigation system (GNSS, TLEs or retro-reflector tracking), but can also be fed with simulated data from the Orekit-based FSF. The controller interface can be implemented both by orbit controllers as well by closed-loop formation controllers like the ones presented in this work.

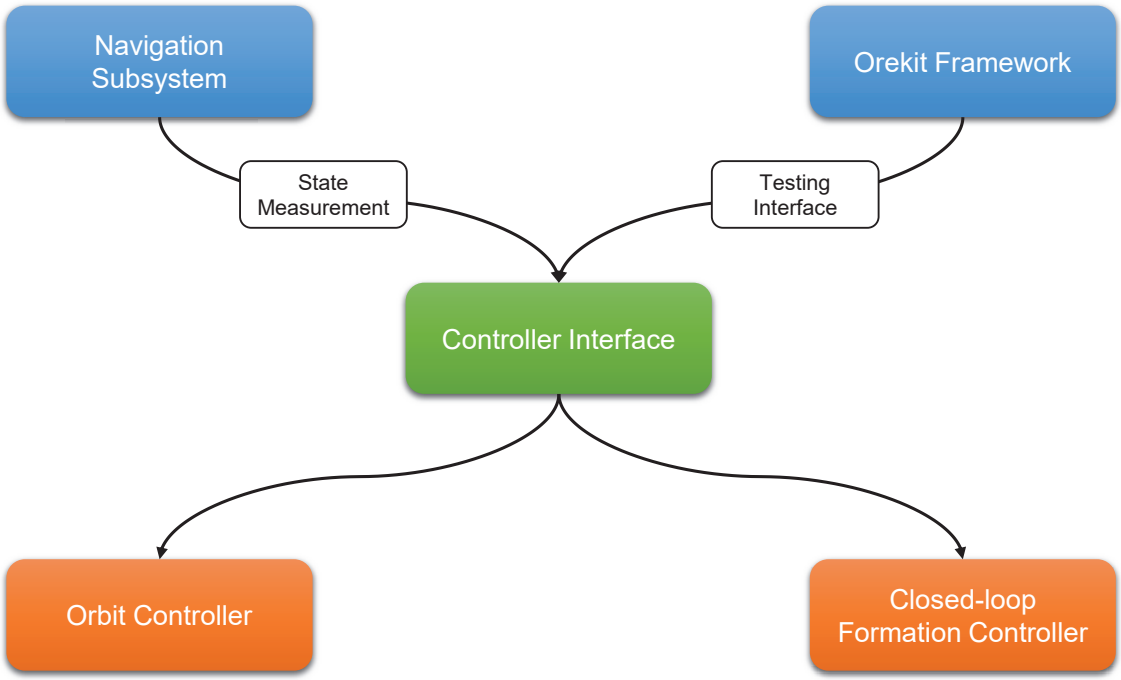


Figure 5.49: The embedded controller interface with its inheritors orbit and formation controller and its connections to the navigation subsystem and the Orekit framework for testing purposes.

A formation controller following the developed controller interface receives several inputs and needs to provide a specific output. As visualized in Fig. 5.50, it obtains the current time (from the navigation subsystem from the GPS receiver), the remaining

propellant mass of the thrust system, the current position and velocity of the satellite as well as the current (propagated) positions and velocities of the other satellites in the formation as received via the ISL and propagated by the navigation subsystem. Thus, the formation controller has to compute a control input / thrust vector and provide it as a 3D vector in the LVLH coordinate frame. It can also provide a sequence of thrust vectors for a longer period in time.

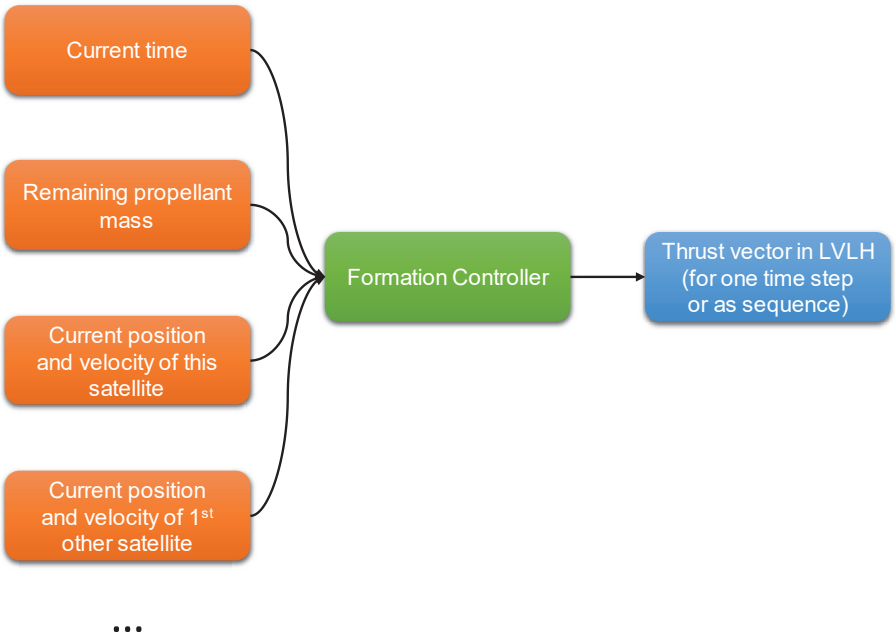


Figure 5.50: Embedded formation controller interface with its inputs and output.

At the current point in time, a *MPC-Based Formation Controller* (cf. Chap. 3.2) has been implemented on NetSat and is waiting for its use in the mission, which would also show its applicability in a real-world scenario. The *Distributed Robust Consensus-Based Control* (cf. Chap. 3.3) is currently not foreseen to be implemented on NetSat, since in its present state, it cannot take constraints like limited thrust into account. However, an extension has already been started and may end in an embedded implementation as well. The *Plant Inversion-Based Lyapunov Control Combined with a Reference Governor* (cf. Chap. 3.4), which is comparably suitable for this type of missions (CLT SFF) is currently under preparation for on-board implementation and may be added via software update, if the embedded implementation and succeeding software tests on ground are successful.

As soon as the commissioning phase of the NetSat mission is finished successfully and all main subsystems like navigation, AOCS, ISL and thrust system are fully operational, the formation flying scenarios can be performed using the implemented *MPC-Based Formation Controller*.

5.5.1 NetSat Mission Plan

Using the developed FSF and the *MPC-Based Formation Control* approach, a complete mission plan for the NetSat mission is proposed. The detailed plan is given in Tab. 5.10. The plan comprises 9 months and is divided into 19 phases.

Table 5.10: Proposed mission plan for the NetSat mission.

Phase	Dimension	Action	Type	Duration [Days]
1	-	Initial uncontrolled drifting	Drift	30
2	-	Initial drift compensation	Drift compensation	10
3	-	Orbit assimilation	Drift compensation	20
4	1	ATF – 10 km	Acquisition	20
5	1	ATF – 10 km	Drift	14
6	1	ATF – 10 km	Maintenance	14
7	2	CWF – 3:1 – in plane – 5 km	Acquisition	10
8	2	CWF – 3:1 – in plane – 5 km	Drift	14
9	2	CWF – 3:1 – in plane – 5 km	Maintenance	14
10	3	CWF – 3:1 – out of plane – 5 km	Acquisition	4
11	3	CWF – 3:1 – out of plane – 5 km	Drift	14
12	3	CWF – 3:1 – out of plane – 5 km	Maintenance	14
13	1	ATF – 10 km	Acquisition	20
14	1	ATF – 10 km	Maintenance	14
15	3	THF – 5 km	Acquisition	14
15	3	THF – 5 km	Acquisition	14
16	3	THF – 5 km	Drift	14
17	3	THF – 5 km	Maintenance	14
18	1	ATF – 10 km	Acquisition	20
19	1	ATF – 10 km	Maintenance	14
		End of mission		**272**

For each formation, the acquisition phase with the time required for it, uncontrolled drift for two weeks and formation maintenance for another two weeks are simulated. The mission plan starts with a phase of uncontrolled drift for 30 days, which represents the commissioning phase of the satellites. Then, initial drift compensation is performed to achieve identical orbit periods (semi-major axis) of all satellites. After that, the orbit assimilation follows, where all COEs except of the true anomaly are driven to be equal. Next, the first formation, namely an ATF is planned, where the satellites will be distributed equally within an interval of 10 km on the same orbit. Afterwards, the first two-dimensional formation, the CWF – 3:1 in-plane part, with a baseline of 5 km is performed. It follows the out-of-plane part of the the CWF – 3:1 formation, completing it. After this first three-dimensional formation, the satellites

go back to the 10 km ATF, which acts as an intermediate hold point for analyzing satellite data and preparing the next formation scenario. The second 3D formation is the THF which follows with a baseline of 5 km. After successfully performing the THF, the satellites again go back to the 10 km ATF and remain there for any further tasks.

With the proposed plan, all objectives in terms of formation configuration of the NetSat mission are met. The plan includes one-, two- and three-dimensional phases, each with enough time to conduct detailed analysis of the controller performance, formation topologies and the satellite systems in general.

Chapter 6

Conclusions

To conclude this thesis, first a summary of the scientific contribution is given and the advantages and drawbacks of the developed control methods are summarized. Special focus is set towards the application within Continuous Low-Thrust Satellite Formation Flying missions, especially the novel and unique CubeSat mission NetSat. Finally, different suggestions how to continue with the outcome of this work are presented as future work.

6.1 Summary

Within this thesis, three different formation control approaches have been designed, implemented and evaluated for the new application of Continuous Low-Thrust Satellite Formation Flying of CubeSats. Thus, its scientific contribution can be understood as the novel development of control approaches for this specific class of application scenarios, which is still lacking methods withstanding the application in real-world space missions, as well as the scientific evaluation and comparison of the developed methods.

ZfT's NetSat mission acted as the primary application scenario. Based on it, different sensor and actuator systems have been analyzed with respect to their characteristics and its impact on navigation and control. Thus, a realistic set of requirements and constraints for the control algorithms within this work could be defined and realistic simulations that guarantee the applicability to real-world missions were enabled. First, a *MPC-Based Formation Control* approach has been developed. Second, a *Distributed Robust Consensus-Based Control* approach has been designed and last, a *Plant Inversion-Based Lyapunov Control Combined with a Reference Governor* has been created. The three control approaches have been evaluated in different scenarios e.g. a Cartwheel Formation – 3:1 and a Tetrahedron Formation, both split into acquisition and maintenance phase.

All three control approaches differ in their characteristics, weaknesses and strengths:

The *MPC-Based Formation Control* guarantees optimality and has the ability to take hard and soft constraints into account, like maximum thrust, control time windows or maximum acceleration rates. Furthermore, in the simulated scenarios it

shows the lowest overall ΔV consumption. On the other hand, the time to reach the target state was comparably long in the simulated scenarios. However, this might be improved by fine-tuning the controller parameters. Apart from that, this control approach requires significant computation effort, because it needs to solve a numeric convex optimization problem at least once per orbit. Thus, it is limited to satellites with significant computing power beyond standard microcontrollers.

The *Distributed Robust Consensus-Based Control* provides robustness with respect to sensor noise and disturbances and is distributed using the consensus approach, which renders it unique among all considered methods. It requires significant computational effort, but only once, when the H_∞ synthesis is performed. The execution of the synthesized controller itself is not computationally complex any more and can be performed on regular microcontrollers available on CubeSats. However, the main drawback of this approach is that the modeling of exact constraints like maximum allowed thrust is not easily and precisely achievable in frequency domain, which is required for H_∞ synthesis.

The *Plant Inversion-Based Lyapunov Control Combined with a Reference Governor* uses the Relative Orbit Elements as coordinate frame, which allows clear understanding of the relative trajectories and time-independent design of force-free formations. The implementation of a Reference Governor allows for long-term optimization and implementation of various constraints. Stability is provided by applying Lyapunov theory. The controller limits itself to perform orbit maneuvers only in two directions which on the one hand increases ΔV consumption, but on the other hand reduces the necessary attitude corrections to perform thrust maneuvers significantly. The controller acts with comparably high precision, especially during maintenance maneuvers and is comparably fast in acquisition maneuvers. On the other hand, the ΔV consumption is comparably high. The controller needs to solve an Algebraic Riccati Equation every time step, which requires some computation power e.g. provided by stronger microcontrollers.

In summary, both the *MPC-Based Formation Control* and the *Plant Inversion-Based Lyapunov Control Combined with a Reference Governor* are suitable for Continuous Low-Thrust Satellite Formation Flying applications like e.g. the novel and unique CubeSat 3D Formation Flying mission NetSat, of course showing different advantages and disadvantages. *Distributed Robust Consensus-Based Control* however lacks the ability of limiting the maximum thrust level and thus is not applicable to satellites using Continuous Low-Thrust.

6.2 Suggestions for Future Work

There are different areas to continue the work performed in this thesis. The most promising topics for further investigation and future work are as follows:

As formation control requires the pointing of the satellites' propulsion systems into specific directions at specific points in time, precise attitude control is of particular importance. The attitude control system has to be able to perform the mission-dependent thrust pointing behavior, which imposes various requirements and constraints on the selected controller. Thus, the development of an attitude controller specifically suiting for CubeSat formation flying missions (like NetSat) is of major importance and may be studied as future work.

One additional future step is the evaluation of the *MPC-Based Formation Control* in a real space mission, especially a Continuous Low-Thrust Satellite Formation Flying scenario. The implementation of an embedded version of it has already been finished. An in-orbit demonstration is planned within the 3D formation flying mission NetSat.

Furthermore, the *Plant Inversion-Based Lyapunov Control Combined with a Reference Governor* may be implemented and evaluated in a real space scenario as well. Its embedded implementation has already been started. It could be deployed e.g. in one of the upcoming Satellite Formation Flying missions of ZfT. Then, a direct comparison of the two control approaches would be possible.

Besides, the *Distributed Robust Consensus-Based Control* lacks the ability to take constraints (e.g. thrust constraint) into account. Thus, it is, in its current state, not feasible for application in Continuous Low-Thrust Satellite Formation Flying, because of the strong limitations of thrust systems for small satellites and especially CubeSats. Adding constraints to this approach would strongly move it towards real-world applications.

In addition, the *MPC-Based Formation Control* requires a significant amount of memory and computing power, which is not easily available on embedded systems in small satellites. Thus, an improvement towards performance e.g. using an explicit MPC formulation or a distributed computation approach among multiple subsystems would be beneficial to reduce computational effort and computation time on the satellites.

Bibliography of the Author

- **Julian Scharnagl**, Florian Kempf, Slavi Dombrovski, and Klaus Schilling. **NetSat - Challenges of a Formation Composed of 4 Nano-Satellites**. In *Proceedings of the 72th International Astronautical Congress*, Dubai, UAE, 2021. IAF.

- Roland Haber, **Julian Scharnagl**, and Klaus Schilling. **NetSat - A Formation of Four 3U-Satellites Using Electric Propulsion**. In *Proceedings of the 71th International Astronautical Congress*. IAF, 2020.

- **Julian Scharnagl**, Florian Kempf, Johannes Dauner, and Klaus Schilling. **Distributed robust consensus-based control with collision avoidance for satellite formation flying**. In *Proceedings of the 70th International Astronautical Congress*, Washington DC, USA, 2019a. IAF.

- **Julian Scharnagl**, Florian Kempf, and Klaus Schilling. **Combining Distributed Consensus with Robust H-Infinity-Control for Satellite Formation Flying**. *Electronics*, 8(3):319, 2019b. ISSN 2079-9292. doi: 10.3390/electronics8030319. URL: `http://www.mdpi.com/2079-9292/8/3/319`.

- Andreas Freimann, **Julian Scharnagl**, Timon Petermann, Holger Döbler, and Klaus Schilling. **CubeSat Energy Modelling for Improved Mission Planning and Operations**. In *Proceedings of the 70th International Astronautical Congress*, Washington DC, USA, 2019. IAF.

- Florian Kempf, **Julian Scharnagl**, and Klaus Schilling. **Networked and distributed cooperative attitude control of fractionated small satellites**. In *Proceedings of the 69th International Astronautical Congress*, Bremen, Germany, 2018. IAF.

- **Julian Scharnagl**, Panayiotis Kremmydas, and Klaus Schilling. **Model Predictive Control for Continuous Low Thrust Satellite Formation Flying**. In *Proceedings of the IFAC Workshop on Networked & Autonomous Air & Space Systems*, Santa Fe, NM, USA, 2018. IFAC.

- Andreas Freimann, Florian Kempf, Oliver Ruf, **Julian Scharnagl**, and Klaus Schilling. **Integrated framework for high fidelity simulation of distributed systems of satellites and ground vehicles**. In *Proceedings of the 68th International Astronautical Congress*, Adelaide, Australia, 2017. IAF.

- Lukas M. Steindorf, Simone D. Amico, **Julian Scharnagl**, Florian Kempf, and Klaus Schilling. **Constrained Low-Thrust Satellite Formation-Flying Using Relative Orbit Elements**. In *Proceedings of the 27th AAS/AIAA Space Flight Mechanics Meeting*, San Antonio, Texas, USA, 2017. AAS.
- **Julian Scharnagl** and Klaus Schilling. **New Hardware-in-the-Loop Testing Concept for Small Satellite Formation Control Based on Mobile Robot Platforms**. In *Proceedings of the 4th IFAC Symposium on Telematics Applications*, Porto Alegre, Brasil, 2016. IFAC.
- Klaus Schilling and **Julian Scharnagl**. **Safe Near Range Navigation Based on 3D Time-of-Flight Cameras**. In *Proceedings of the 13th International Symposium on Artificial Intelligence, Robotics and Automation in Space*, Beijing, China, 2016.
- **Julian Scharnagl**, Alexander Kleinschrodt, Andreas Freimann, Karthik Ravandoor, Philip Bangert, Lakshminarasimhan Srinivasan, and Klaus Schilling. **Nanosatellite Constellations for Large-Scale Railway Transportation Monitoring**. In *Proceedings of the 3rd IAA Conference on University Satellite Missions and CubeSat Workshop*, Rome, Italy, 2015a. IAA.
- Klaus Schilling, Philip Bangert, Stephan Busch, Slavi Dombrovski, Andreas Freimann, Alexander Kramer, Tiago Nogueira, Dmitry Ris, **Julian Scharnagl**, and Tristan Tzschichholz. **NetSat: A Four Pico/Nano-Satellite Mission for Demonstration of Autonomous Formation Flying**. In IAF, editor, *Proceedings of the 66th International Astronautical Congress*, Jerusalem, Israel, 2015.
- **Julian Scharnagl**, Lakshminarasimhan Srinivasan, Karthik Ravandoor, and Klaus Schilling. **Autonomous Collision Avoidance for Rendezvous and Docking in Space Using Photonic Mixer Devices**. In *Proceedings of the Workshop on Advanced Control and Navigation for Autonomous Aerospace Vehicles*, Seville, Spain, 2015b. IFAC.
- Tiago Nogueira, **Julian Scharnagl**, Stavros Kotsiaros, and Klaus Schilling. **NetSat-4G: A four nano-satellite formation for global geomagnetic gradiometry**. In *Proceedings of the 10th IAA Symposium on Small Satellites for Earth Observation*, Berlin, Germany, 2015. IAA.
- Leonardo Regoli, **Julian Scharnagl**, Karthik Ravandoor, and Klaus Schilling. **Pose Estimation and Collision Avoidance for Rendezvous and Docking in Space Using Photonic Mixer Devices**. In *Proceedings of the 64th International Astronautical Congress*, Beijing, China, 2013. IFAC.
- **Julian Scharnagl**, Leonardo Regoli, and Klaus Schilling. **On-line Collision Detection in Space Using Photonic Mixer Devices**. In *Proceedings of the 19th IFAC Symposium on Automatic Control in Aerospace*, pages 143–148, Würzburg, Germany, 2013. IFAC. doi: 10.3182/20130902-5-DE-2040.00033.

Appendices

Appendix A

Alternative Definitions

This appendix presents alternative definitions that have not been used in this thesis for the sake of completeness.

A.1 Alternative Definition of Local-Vertical, Local-Horizontal Coordinate Frame

In addition to the definition of Fehse (2003) used in this thesis, the LVLH can also defined following (Alfriend et al., 2010, p. 16): The $\hat{\boldsymbol{e}}_{\boldsymbol{x}}$ is directed from the spacecraft radially outward, $\hat{\boldsymbol{e}}_{\boldsymbol{z}}$ is normal to the fundamental plane, positive in the direction of the instantaneous angular momentum vector, and the $\hat{\boldsymbol{e}}_{\boldsymbol{y}}$ completes the right-handed coordinate frame and is in the direction of the orbital velocity vector (see Fig. A.1).

A.2 Alternative Definition of Hill-Clohessy-Wiltshire Equations

The Hill's equations defined in the EH frame $\mathscr{H}$ following (Alfriend et al., 2010, p.84 ff.) and presented in the section above are defined as

$$\ddot{x} - 2\,n\,\dot{y} - 3\,n^2\,x = d_x + u_x \tag{A.1}$$

$$\ddot{y} + 2\,n\,\dot{x} \qquad\qquad = d_y + u_y \tag{A.2}$$

$$\ddot{z} \qquad\quad + \quad n^2\,z = d_z + u_z \tag{A.3}$$

with $n = \sqrt{\mu/a^3}$, μ is the Earth's standard gravitational parameter and a the semi-major axis of the orbit of the coordinate frame's origin. $\boldsymbol{d} = [d_x,\, d_y,\, d_z]^T$ is the vector of environmental perturbations/disturbances and $\boldsymbol{u} = [u_x,\, u_y,\, u_z]^T$ the vector of control accelerations. The according form of the analytic solution by Clohessy and Wiltshire (Clohessy and Wiltshire, 1960) is defined as

$$x(t) = +\,[4\;x_0 + \frac{2\;\dot{y}_0}{n}] + \frac{\dot{x}_0}{n}\;\sin(n\;t) - [3\;x_0 + \frac{2\;\dot{y}_0}{n}]\;\cos(n\;t)$$

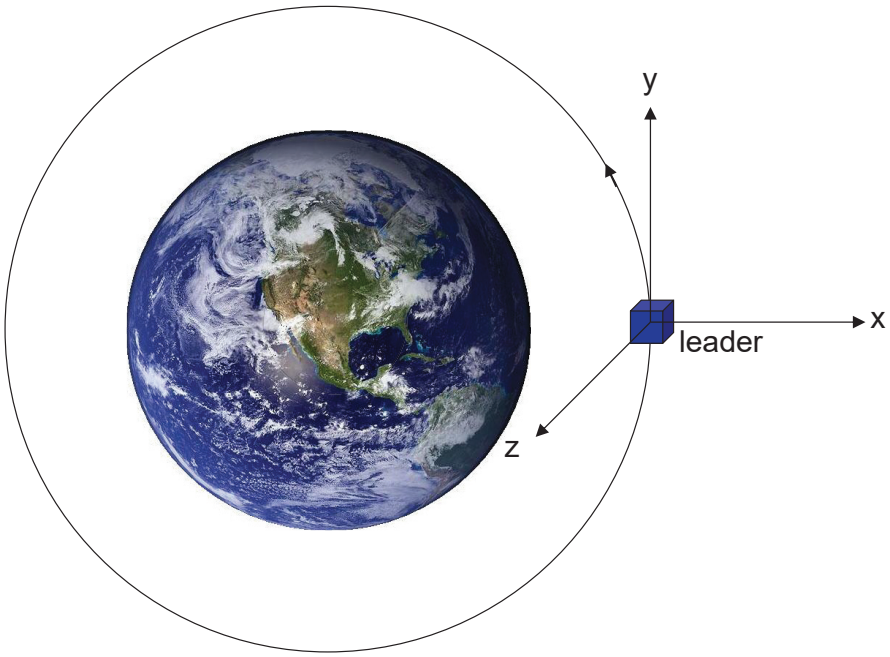


Figure A.1: The LVLH coordinate frame is centered in a (leader) spacecraft (definition based on (Alfriend et al., 2010, p. 16)).

$$+ \frac{\gamma_x}{n^2} \left[1 - \cos(n\ t)\right] \tag{A.4}$$

$$y(t) = + \left[y_0 - \frac{2\ \dot{x}_0}{n}\right] + \left[6\ x_0 + \frac{4\ \dot{y}_0}{n}\right]\ \sin(n\ t) + \frac{2\ \dot{x}_0}{n}\ \cos(n\ t)$$

$$- \left[6\ n\ x_0 + 3\ \dot{y}_0\right]\ t + \gamma_y\ \left[\frac{4}{n^2}\ (1 - \cos(n\ t)) - \frac{3}{2}\ t^2\right] \tag{A.5}$$

$$z(t) = + \frac{\dot{z}_0}{n}\ \sin(n\ t) + z_0\ \cos(n\ t) + \frac{\gamma_z}{n^2}\ \left[1 - \cos(n\ t)\right] \tag{A.6}$$

with the total external accelerations $\boldsymbol{\gamma} = [\gamma_x,\ \gamma_y,\ \gamma_z]^T = \boldsymbol{d} + \boldsymbol{u}$. (Alfriend et al., 2010, p.84 ff.)

Appendix B

Brouwer Transformation

This appendix contains the explicit formulation of the mean-to-osculating orbit transformation stated in Eq. 2.57. The formulation is split into one long-periodic and two short-periodic parts. All input variables have to be mean orbital values. The transformation makes use of the following variables

$$\eta = \sqrt{1 - e^2} \tag{B.1a}$$

$$\Theta = \frac{1}{1 - 5\cos^2 i} \tag{B.1b}$$

$$\epsilon_1 = \sqrt{q_1^2 + q_2^2} \tag{B.1c}$$

$$\epsilon_2 = q_1 \cos\Theta + q_2 \sin\Theta \tag{B.1d}$$

$$\epsilon_3 = q_1 \sin\Theta - q_2 \cos\Theta \tag{B.1e}$$

The elements of $\boldsymbol{\Delta a_{lp}}(\bar{\boldsymbol{a}})$ are

$$a_{lp} = 0 \tag{B.2a}$$

$$\begin{aligned}\lambda_{lp} = {} & \frac{q_1 q_2 \sin^2 i}{8a^2\eta^2(1+\eta)}\left(1 - 10\Theta\cos^2 i\right) \\ & + \frac{q_1 q_2}{16a^2\eta^4}\left(3 - 55\cos^2 i - 280\Theta\cos^4 i - 400\Theta\Theta\cos^6 i\right)\end{aligned} \tag{B.2b}$$

$$\begin{aligned}\theta_{lp} = {} & \lambda_{lp} - \frac{\sin^2 i}{16a^2\eta^4}\left(1 - 10\Theta\cos^2 i\right) \\ & \left[q_1 q_2\left(3 + 2\frac{\eta^2}{1+\eta}\right) + 2\left(q_1\sin\Theta + q_2\cos\Theta\right) + \frac{\epsilon_1\sin(2\theta)}{2}\right]\end{aligned} \tag{B.2c}$$

$$i_{lp} = \frac{\sin(2i)}{32a^2\eta^4}\left(1 - 10\Theta\cos^2 i\right)\left(q_1^2 - q_2^2\right) \tag{B.2d}$$

$$q_{1,lp} = -\frac{q_1 \sin^2 i}{16a^2\eta^2}\left(1 - 10\Theta\cos^2 i\right) - \frac{q_1 q_2^2}{16a^2\eta^4}\left(3 - 55\cos^2 i - 280\Theta\cos^4 i - 400\Theta\Theta\cos^6 i\right) \tag{B.2e}$$

$$q_{2,lp} = \frac{q_2 \sin^2 i}{16a^2\eta^2}\left(1 - 10\Theta\cos^2 i\right) + \frac{q_1^2 q_2}{16a^2\eta^4}\left(3 - 55\cos^2 i - 280\Theta\cos^4 i - 400\Theta\Theta\cos^6 i\right) \tag{B.2f}$$

$$\Omega_{lp} = \frac{q_1 q_2 \cos i}{8a^2\eta^4}\left(11 + 80\Theta\cos^2 i + 200\Theta\Theta\cos^4 i\right) \tag{B.2g}$$

The elements of $\boldsymbol{\Delta a_{sp1}}(\bar{\boldsymbol{a}})$ are

$$a_{sp1} = \frac{1 - 3\cos^2 i}{2a\eta^6}\left(\epsilon_2^3 - \eta^3\right) \tag{B.3a}$$

$$\lambda_{sp1} = \frac{\epsilon_3\left(1 - 3\cos^2 i\right)}{4a^2\eta^4\left(1+\eta\right)}\left[\left(1+\epsilon_2\right)^2 + 1 + \epsilon_2 + \eta^2\right] + \frac{3\left(1 - 5\cos^2 i\right)}{4a^2\eta^4}\left(\theta - \lambda + \epsilon_3\right) \tag{B.3b}$$

$$\theta_{sp1} = \lambda_{sp1} - \frac{\epsilon_3\left(1 - 3\cos^2 i\right)}{4a^2\eta^4\left(1+\eta\right)}\left[\left(1+\epsilon_2\right)^2 + \eta\left(1+\eta\right)\right] \tag{B.3c}$$

$$i_{sp1} = 0 \tag{B.3d}$$

$$q_{1,sp1} = \frac{1 - 3\cos^2 i}{4a^2\eta^4\left(1+\eta\right)}\left[\left(\epsilon_2^2 + \eta^2\right)\left(q_1 + \left(1+\eta\right)\cos\Theta\right)\right. \\ \left.\left(1+\epsilon_2\right)\left[\left(1+\eta\right)\cos\Theta + q_1\left(\eta - \epsilon_2\right)\right]\right] \\ - \frac{3q_2\left(1 - 5\cos^2 i\right)}{4a^2\eta^4}\left(\theta - \lambda + \epsilon_3\right) \tag{B.3e}$$

$$q_{2,sp1} = \frac{1-3\cos^2 i}{4a^2\eta^4(1+\eta)}\left[\left(\epsilon_2^2+\eta^2\right)(q_2+(1+\eta)\sin\Theta)\right.$$
$$\left.(1+\epsilon_2)\left[(1+\eta)\sin\Theta+q_2(\eta-\epsilon_2)\right]\right] \tag{B.3f}$$
$$+\frac{3q_1(1-5\cos^2 i)}{4a^2\eta^4}(\theta-\lambda+\epsilon_3)$$

$$\Omega_{sp1} = \frac{3\cos i}{2a^2\eta^4}(\theta-\lambda+\epsilon_3) \tag{B.3g}$$

The elements of $\boldsymbol{\Delta a_{sp2}}(\bar{\boldsymbol{a}})$ are

$$a_{sp2} = -\frac{3\sin^2 i}{2a\eta^6}\epsilon_2^3\cos(2\theta) \tag{B.4a}$$

$$\lambda_{sp2} = -\frac{3\epsilon_3\sin^2 i\cos(2\theta)}{4a^2\eta^4(1+\eta)}(1+\epsilon_1)(2+\epsilon_2) - \frac{\sin^2 i}{8a^2\eta^2(1+\eta)}$$
$$[3(q_1\sin\Theta+q_2\cos\Theta)+(q_1\sin(3\theta)-q_2\cos(3\theta))]$$
$$-\frac{3-5\cos^2 i}{8a^2\eta^4}[3(q_1\sin\Theta+q_2\cos\Theta)+3\sin(2\theta)$$
$$+(q_1\sin(3\theta)-q_2\cos(3\theta))] \tag{B.4b}$$

$$\theta_{sp2} = \lambda_{sp2} - \frac{\sin^2 i}{32a^2\eta^4(1+\eta)}\left[36q_1q_2 - 4\left(3\eta^2+5\eta-1\right)\right.$$
$$(q_1\sin\Theta+q_2\cos\Theta)+12\epsilon_2q_1q_2-32(1+\eta)\sin(2\theta)$$
$$-\left(\eta^2+12\eta+39\right)(q_1\sin(3\theta)-q_2\cos(3\theta))$$
$$+36q_1q_2\cos(4\theta)-18\left(q_1^2-q_2^2\right)\sin(4\theta)$$
$$\left.-3\left(q_1^2-3q_2^2\right)q_1\sin(5\theta)+3\left(3q_1^2-q_2^2\right)q_2\cos(5\theta)\right] \tag{B.4c}$$

$$i_{sp2} = -\frac{\sin(2i)}{8a^2\eta^4}[3(q_1\cos\Theta-q_2\sin\Theta)+3\cos(2\theta)$$
$$+(q_1\cos(3\theta)+q_2\sin(3\theta))] \tag{B.4d}$$

$$
\begin{aligned}
q_{1,sp2} = & \frac{q_2\left(3-5\cos^2 i\right)}{8a^2\eta^4}\left[3\left(q_1\sin\Theta+q_2\cos\Theta\right)+3\sin\left(2\theta\right)\right.\\
& \left.+\left(q_1\sin\left(3\theta\right)-q_2\cos\left(3\theta\right)\right)\right]\\
& +\frac{\sin^2 i}{8a^2\eta^4}\left[3\left(\eta^2-q_1^2\right)\cos\Theta+3q_1q_2\sin\Theta\right.\\
& \left.-\left(\eta^2+3q_1^2\right)\cos\left(3\theta\right)-3q_1q_2\sin\left(3\theta\right)\right]\\
& -\frac{3\sin^2 i\cos\left(2\theta\right)}{16a^2\eta^4}\left[10q_1+\left(8+3q_1^2+q_2^2\right)\cos\Theta\right.\\
& +2q_1q_2\sin\Theta+6\left(q_1\cos\left(2\theta\right)+q_2\sin\left(2\theta\right)\right)\\
& \left.+\left(q_1^2-q_2^2\right)\cos\left(3\theta\right)+2q_1q_2\sin\left(3\theta\right)\right]
\end{aligned}
\tag{B.4e}
$$

$$
\begin{aligned}
q_{2,sp2} = & -\frac{q_1\left(3-5\cos^2 i\right)}{8a^2\eta^4}\left[3\left(q_1\sin\Theta+q_2\cos\Theta\right)+3\sin\left(2\theta\right)\right.\\
& \left.+\left(q_1\sin\left(3\theta\right)-q_2\cos\left(3\theta\right)\right)\right]\\
& \frac{\sin^2 i}{8a^2\eta^4}\left[3\left(\eta^2-q_2^2\right)\sin\Theta+3q_1q_2\cos\Theta\right.\\
& \left.+\left(\eta^2+3q_2^2\right)\sin\left(3\theta\right)+3q_1q_2\cos\left(3\theta\right)\right]\\
& -\frac{3\sin^2 i\cos\left(2\theta\right)}{16a^2\eta^4}\left[10q_2+\left(8+q_1^2+3q_2^2\right)\sin\Theta\right.\\
& +2q_1q_2\cos\Theta+6\left(q_1\sin\left(2\theta\right)-q_2\cos\left(2\theta\right)\right)\\
& \left.+\left(q_1^2-q_2^2\right)\sin\left(3\theta\right)-2q_1q_2\cos\left(3\theta\right)\right]
\end{aligned}
\tag{B.4f}
$$

$$
\begin{aligned}
\Omega_{sp2} = & -\frac{\cos i}{4a^2\eta^4}\left[3\left(q_1\sin\Theta+q_2\cos\Theta\right)\right.\\
& \left.+3\sin\left(2\theta\right)+\left(q_1\sin\left(3\theta\right)-q_2\cos\left(3\theta\right)\right)\right]
\end{aligned}
\tag{B.4g}
$$

Appendix C

Simplified Differential Mean to Osculation Transformation

This appendix presents the elements of the matrix $\boldsymbol{D_s}$ defined in Eq. 2.70. All matrix elements which are not stated below are zero. The matrix indices start with zero. Additional variables that are used are

$$\tau_1 = -8a^{-2}\sin^2 i\left(1 - 10\Theta\cos^2 i\right) \tag{C.1a}$$

$$\tau_2 = \left(1 - 3\cos^2 i\right) \tag{C.1b}$$

$$\tau_3 = \left(1 - 5\cos^2 i\right) \tag{C.1c}$$

Long-periodic part of $\boldsymbol{D_s}$

$$\begin{aligned} D^{lp}_{1,3} &= \tau_1 \sin\Theta \\ D^{lp}_{1,4} &= \tau_1 \cos\theta \end{aligned} \tag{C.2a}$$

$$D^{lp}_{3,3} = 0.5\tau_1 \tag{C.2b}$$

$$D^{lp}_{4,4} = -0.5\tau_1 \tag{C.2c}$$

Short-periodic (1) part of $\boldsymbol{D_s}$

$$\begin{aligned} D^{sp1}_{0,3} &= 3\tau_2\frac{\cos\Theta}{2a} \\ D^{sp1}_{0,4} &= 3\tau_2\frac{\sin\Theta}{2a} \end{aligned} \tag{C.3a}$$

$$
\begin{aligned}
D_{1,3}^{sp1} &= 9\tau_3 \frac{\sin\Theta}{4a^2} \\
D_{1,4}^{sp1} &= -9\tau_3 \frac{\cos\Theta}{4a^2}
\end{aligned}
\tag{C.3b}
$$

$$
\begin{aligned}
D_{3,0}^{sp1} &= -3\tau_2 \frac{\cos\Theta}{2a^3} \\
D_{3,1}^{sp1} &= -3\tau_2 \frac{\sin\Theta}{4a^2} \\
D_{3,2}^{sp1} &= 9\sin(2i)\,\frac{\cos\Theta}{4a^2} \\
D_{3,3}^{sp1} &= 3\tau_2 \frac{2+\cos(2\Theta)}{8a^2} \\
D_{3,4}^{sp1} &= 3\tau_2 \frac{\sin(2\Theta)}{8a^2}
\end{aligned}
\tag{C.3c}
$$

$$
\begin{aligned}
D_{4,0}^{sp1} &= -3\tau_2 \frac{\sin\Theta}{2a^3} \\
D_{4,1}^{sp1} &= 3\tau_2 \frac{\cos\Theta}{4a^2} \\
D_{4,2}^{sp1} &= 9\sin(2i)\,\frac{\sin\Theta}{4a^2} \\
D_{4,3}^{sp1} &= 3\tau_2 \frac{\sin(2\Theta)}{8a^2} \\
D_{4,4}^{sp1} &= 3\tau_2 \frac{2-\cos(2\Theta)}{8a^2}
\end{aligned}
\tag{C.3d}
$$

$$
\begin{aligned}
D_{5,3}^{sp1} &= 9\cos i \frac{\sin\Theta}{4a^2} \\
D_{5,4}^{sp1} &= -9\cos i \frac{\cos\Theta}{4a^2}
\end{aligned}
\tag{C.3e}
$$

Short-periodic (2) part of $\boldsymbol{D_s}$

$$
\begin{aligned}
D_{0,0}^{sp2} &= 3\sin^2 i \frac{\cos(2\Theta)}{2a^2} \\
D_{0,1}^{sp2} &= 3\sin^2 i \frac{\sin(2\Theta)}{a} \\
D_{0,2}^{sp2} &= -3\sin(2i)\,\frac{\cos(2\Theta)}{2a} \\
D_{0,3}^{sp2} &= -9\sin^2 i \frac{\cos\Theta + \cos(3\Theta)}{4a} \\
D_{0,4}^{sp2} &= 9\sin^2 i \frac{\sin\Theta - \sin(3\Theta)}{4a}
\end{aligned}
\tag{C.4a}
$$

$$
\begin{aligned}
D_{1,0}^{sp2} &= -(6 - 7\sin^2 i)\frac{\sin(2\Theta)}{4a^3} \\
D_{1,1}^{sp2} &= (6 - 7\sin^2 i)\frac{\cos(2\Theta)}{4a^2} \\
D_{1,2}^{sp2} &= -7\sin(2i)\,\frac{\sin(2\Theta)}{8a^2} \\
D_{1,3}^{sp2} &= \frac{24 - 47\sin^2 i}{32a^2}\sin\Theta + \frac{\cos^2 i \sin(3\Theta)}{4a^2} \\
D_{1,4}^{sp2} &= \frac{24 - 47\sin^2 i}{32a^2}\cos\Theta - \frac{\cos^2 i \cos(3\Theta)}{4a^2}
\end{aligned}
\tag{C.4b}
$$

$$
\begin{aligned}
D_{2,0}^{sp2} &= \frac{3\sin(2i)\cos(2\Theta)}{4a^3} \\
D_{2,1}^{sp2} &= \frac{3\sin(2i)\sin(2\Theta)}{4a^2} \\
D_{2,2}^{sp2} &= -\frac{3\cos(2i)\cos(2\Theta)}{4a^2} \\
D_{2,3}^{sp2} &= -\frac{\sin(2i)}{8a^2}(3\cos\Theta + \cos(3\Theta)) \\
D_{2,4}^{sp2} &= \frac{\sin(2i)}{8a^2}(3\sin\Theta - \sin(3\Theta))
\end{aligned}
\tag{C.4c}
$$

$$
\begin{aligned}
D_{3,0}^{sp2} &= \frac{\sin^2 i}{4a^3}(3\cos\Theta + 7\cos(3\Theta)) \\
D_{3,1}^{sp2} &= \frac{3\sin^2 i}{8a^2}(\sin\Theta + 7\sin(3\Theta)) \\
D_{3,2}^{sp2} &= -\frac{\sin(2i)}{8a^2}(3\cos\Theta + 7\cos(3\Theta)) \\
D_{3,3}^{sp2} &= -\frac{3\sin^2 i}{16a^2}\left(3 + 10\cos(2\Theta) + 3\cos(4\Theta)\right) \\
D_{3,4}^{sp2} &= \frac{3(3 - 5\cos^2 i)}{8a^2}\sin(2\Theta) - \frac{9\sin^2 i\sin(4\Theta)}{16a^2}
\end{aligned}
\tag{C.4d}
$$

$$
\begin{aligned}
D_{4,0}^{sp2} &= -\frac{\sin^2 i}{4a^3}(3\sin\Theta - 7\sin(3\Theta)) \\
D_{4,1}^{sp2} &= \frac{3\sin^2 i}{8a^2}(\cos\Theta - 7\cos(3\Theta)) \\
D_{4,2}^{sp2} &= \frac{\sin(2i)}{8a^2}(3\sin\Theta - 7\sin(3\Theta)) \\
D_{4,3}^{sp2} &= -\frac{3(3-5\cos^2 i)\sin(2\Theta)}{8a^2} - \frac{9\sin^2 i\sin(4\Theta)}{16a^2} \\
D_{4,4}^{sp2} &= \frac{3\sin^2 i}{16a^2}(3 - 10\cos(2\Theta) + 3\cos(4\Theta))
\end{aligned}
\tag{C.4e}
$$

$$
\begin{aligned}
D_{5,0}^{sp2} &= \frac{3\cos i\sin(2\Theta)}{2a^3} \\
D_{5,1}^{sp2} &= -\frac{3\cos i\cos(2\Theta)}{2a^2} \\
D_{5,2}^{sp2} &= \frac{3\sin i\sin(2\Theta)}{4a^2} \\
D_{5,3}^{sp2} &= -\frac{\cos i}{4a^2}(3\sin\Theta + \sin(3\Theta)) \\
D_{5,4}^{sp2} &= -\frac{\cos i}{4a^2}(3\cos\Theta - \cos(3\Theta))
\end{aligned}
\tag{C.4f}
$$

List of Tables

List of Figures

References

Mohammad Alawieh, Niels Hadaschik, Norbert Franke, and Christopher Mutschler. **Inter-Satellite Ranging in the Low Earth Orbit**. In *Proceedings of the 10th International Symposium on Communication Systems, Networks and Digital Signal Processing (CSNDSP)*, pages 1–6, Prague, Czech Republic, 2016. IEEE. ISBN 9781509025268. doi: 10.1109/CSNDSP.2016.7573920. URL: `https://ieeexplore.ieee.org/abstract/document/7573920`.

Kyle T. Alfriend, Dong-Woo Gim, and Hanspeter Schaub. **Gravitational Perturbations, Nonlinearity and Circular Orbit Assumption Effects on Formation Flying Control Strategies**. *Guidance and control 2000*, AAS 00-012:139–158, 2000. URL: `http://hanspeterschaub.info/Papers/Brecken2000.pdf`.

Kyle T. Alfriend, Srinivas R. Vadali, Pini Gurfil, Jonathan P. How, and Louis S. Breger. ***Spacecraft Formation Flying: Dynamics, control and navigation***. Butterworth-Heinemann, Oxford, UK, 2010. ISBN 978-0-7506-8533-7. URL: `http://www.sciencedirect.com/science/book/9780750685337http://proquest.safaribooksonline.com/9780750685337`.

Finn Ankersen. **Multiple Spacecraft Classification for Formation Flying Definition**. Technical report, ESA, Washington DC, USA, 2004.

Finn Ankersen. ***Guidance, Navigation, Control and Relative Dynamics for Spacecraft Proximity Maneuvers***. Phdtesis, Aalborg University, 2010. URL: `http://vbn.aau.dk/ws/files/61599334/finnAnkersenThesis.pdf`.

Jean-Sebastien Ardaens, Simone D'Amico, Dieter Ulrich, and Denis Fischer. **Tandem-x autonomous formation flying system**. In *3rd International Symposium on Formation Flying, Missions and Technology*, pages 23–25, 2008. URL: `http://elib.dlr.de/57870/01/Ardaens.pdf`.

Roberto Armellin, Mauro Massari, and Amalia Ercoli Finzi. **Optimal formation flying reconfiguration and station keeping maneuvers using low thrust propulsion**. In *18th International Symposium on Space Flight Dynamics*, volume 548, page 429. Munich, Germany: Oct, 2004. URL: `http://issfd.org/ISSFD_2004/papers/P1024.pdf`.

Philip Bangert, Slavi Dombrovski, Alexander Kramer, and Klaus Schilling. **UWE-4: Advances in the Attitude and Orbit Control of a Pico-Satellite**. In *Proceedings of the 4S Symposium on Small Satellites Systems and Services*, Valetta, Malta, 2016. ISBN 9781424449682. doi: 10.1002/biot.201500297.

S. K. Barik, M. P. Biswal, and D. Chakravarty. **Multiobjective two-stage stochastic programming problems with interval discrete random variables**. *Advances in Operations Research*, 2012:1–21, 2012. doi: 10.1155/2012/279181. URL: https://doi.org/10.1155/2012/279181.

R. Barza, Y. Aoki, and K. Schilling. **CubeSat UWE-1 Technology Tests and In-Orbit Results**. 57th International Astronautical Congress, 2006.

Roger R. Bate, Donald D. Mueller, and Jerry E. White. ***Fundamentals of astrodynamics***. Courier Corporation, 1971. URL: https://books.google.de/books?hl=de&lr=&id=UtJK8cetqGkC&oi=fnd&pg=PR5&dq=Fundamentals+of+astrodynamics&ots=WCdyXfy-m4&sig=I8Crg8XDxvj_r0UepS68p9Ga6zw.

Richard H. Battin. ***An Introduction to the Mathematics and Methods of Astrodynamics***. AIAA Education Series, Reston, VA, USA, revised ed edition, 1999. ISBN 9781563473425. doi: 10.2514/4.861543.

Riccardo Bevilacqua, Jason S. Hall, and Marcello Romano. **Multiple spacecraft rendezvous maneuvers by differential drag and low thrust engines**. *Celestial Mechanics and Dynamical Astronomy*, 106(1):69–88, 2010. ISSN 09232958. doi: 10.1007/s10569-009-9240-3.

D. Bock, M. Bethge, and M. Tajmar. **Highly Miniaturized FEEP Thrusters for CubeSat Applications**. In *Proceedings of the 4th Spacecraft Propulsion Conference*, Cologne, Germany, 2014.

Grant Bonin, Niels Roth, Scott Armitage, Josh Newman, Ben Risi, and Robert E. Zee. **CanX-4 and CanX-5 Precision Formation Flight: Mission Accomplished!** In *Proceedings of the 29th Annual AIAA/USU Conference on Small Satellites*, Logan, UT, USA, 2015. AIAA.

Bosch Sensortec GmbH. **BMX055 - Digital 9-axis sensor**. Technical report, Bosch Sensortec GmbH, Reutlingen, Germany, 2012. URL: https://www.bosch-sensortec.com/media/boschsensortec/downloads/product_flyer/bst-bmx055-fl000.pdf.

Louis Breger and Jonathan P. How. **J2-modified GVE-based MPC for formation flying spacecraft**. In *AIAA Guid. Navig. Contr. Conf. Exhi*, pages 1–12, 2005. URL: http://arc.aiaa.org/doi/pdf/10.2514/6.2005-5833.

Louis Breger and Jonathan P. How. **Gauss's variational equation-based dynamics and control for formation flying spacecraft**. *Journal of guidance, control, and dynamics*, 30(2):437–448, 2007. URL: http://arc.aiaa.org/doi/abs/10.2514/1.22649.

Roger A. Broucke. **Solution of the elliptic rendezvous problem with the time as independent variable**. *Journal of Guidance, Control, and Dynamics*, 26(4):615–621, 2003. URL: http://arc.aiaa.org/doi/abs/10.2514/2.5089.

Roger A. Broucke and Paul J. Cefola. **On the equinoctial orbit elements**. *Celestial Mechanics*, 5(3):303–310, 1972. URL: http://link.springer.com/article/10.1007/BF01228432.

Dirk Brouwer. **Solution of the problem of artificial satellite theory without drag**. *The Astronomical Journal*, 64:378, 1959. URL: http://adsabs.harvard.edu/full/1959AJ.....64..378B7.

Stephan Busch and Klaus Schilling. **UWE-3: A Modular System Design for the Next Generation of Very Small Satellites**. In *Proceedings of the Small Satellites Systems and Services Symposium*, Portorož, Slovenia, 2012. ESA.

Stephan Busch and Klaus Schilling. **Robust and Efficient OBDH Core Module for the Flexible Picosatellite Bus UWE-3**. In *Proceedings of the 19th IFAC Symposium on Automatic Control in Aerospace*, pages 218–223, Würzburg, Germany, 2013. IFAC. ISBN 9783902823465.

Stephan Busch, Philip Bangert, and Klaus Schilling. **Attitude Control Demonstration for Pico-Satellite Formation Flying by UWE-3**. In *Proceedings of the Small Satellites Systems and Services Symposium*, Porto Petro, Spain, 2014. ESA.

California Polytechnic State University. **Cubesat Design Specification Rev. 14**. Technical report, Cal Poly, San Luis Obispo, CA, USA, 2020. URL: https://www.cubesat.org/cds-announcement.

Thomas E. Carter. **State transition matrices for terminal rendezvous studies: brief survey and new example**. *Journal of Guidance, Control, and Dynamics*, 21(1):148–155, 1998. URL: http://arc.aiaa.org/doi/abs/10.2514/2.4211.

CCSDS 500.0-G-3. **Navigation Data - Definitions and Conventions. Green Book**. Technical Report 3, CCSDS, Washington DC, USA, 2010.

Arben Çela, Mongi Ben Gaid, Xu-Guang Li, and Silviu-Iulian Niculescu. ***Optimal Design of Distributed Control and Embedded Systems***. Springer International Publishing Switzerland, Cham, Switzerland, 2014. ISBN 9783319027289.

Dong Eui Chang and Jerrold E. Marsden. **Geometric derivation of the Delaunay variables and geometric phases**. *Celestial Mechanics and Dynamical Astronomy*, 86(2):185–208, 2003. URL: http://link.springer.com/article/10.1023/A:1024174702036.

M. Chernick and S. D'Amico. **New closed-form solutions for optimal impulsive control of spacecraft relative motion**. *AIAA/AAS Astrodynamics Specialist Conference, AIAA SPACE Forum (AIAA 2016-5659)*, 2016. doi: DOI10.2514/6.2016-5659.

Kevin Chmiela. ***Development and testing of a high precision sun sensor for the UWE-4 satellite***. Masterthesis, University of Würzburg, 2018.

Vladimir A. Chobotov. ***Orbital mechanics***. Aiaa, 2002. URL: https://books.google.de/books?hl=de&lr=&id=SuPQmbqyrFAC&oi=fnd&pg=PR5&dq=orbital+mechanics&ots=kYFGAHfxAZ&sig=UX3TtxlrojBiGLc1nguLMIf60L4.

W. H. Clohessy and R. S. Wiltshire. **Terminal guidance system for satellite rendezvous**. *Journal of the Aerospace Sciences*, 27(9):653–658, 1960. URL: http://arc.aiaa.org/doi/abs/10.2514/8.8704.

CubeSpace. **CubeTorquer and CubeCoil Brochure**. Technical report, CubeSpace, Stellenbosch, South Africa, 2016. URL: https://www.cubespace.co.za/products/adcs-components/cubetorquer/.

CubeSpace. **CubeWheel Interface Control Document**. Technical report, CubeSpace, Stellenbosch, South Africa, 2020. URL: https://www.cubespace.co.za/products/adcs-components/cubewheel/.

S. D'Amico and O. Montenbruck. **Proximity operations of formatiln flying spacecraft using an eccentricity/inclination vector separation**. Journal of Guidance, Control and Dynamics, 2006.

Simone D'Amico. ***Autonomous Formation Flying in Low Earth Orbit***. PhD thesis, TU Delft, 2010. URL: http://repository.tudelft.nl/view/ir/uuid:a10e2d63-399d-48e5-884b-402e9a105c70/.

Simone D'Amico, Eberhard Gill, and Oliver Montenbruck. **Relative orbit control design for the PRISMA formation flying mission**. In *AIAA GNC Conference*, 2006. URL: http://arc.aiaa.org/doi/pdf/10.2514/6.2006-6067.

Simone D'Amico, Jean-Sebastien Ardaens, and Oliver Montenbruck. **Navigation of formation flying spacecraft using GPS: the PRISMA technology demonstration**. In *Proceedings of the 22nd International Technical Meeting of the Satellite Division of the Institute of Navigation (ION-GNSS'09)*, pages 2250–2264, 2009a. URL: https://www.researchgate.net/profile/Jean-Sebastien_Ardaens/publication/224993197_Navigation_of_Formation_Flying_Spacecraft_using_GPS_the_PRISMA_Technology_Demonstration/links/0c9605231bdbf96566000000.pdf.

Simone D'Amico, Sergio De Florio, Robin Larsson, and Matti Nylund. **Autonomous formation keeping and reconfiguration for remote sensing spacecraft**. In *21st International Symposium on Space Flight Dynamics*, volume 28, 2009b. URL: http://www.academia.edu/download/46106048/02e7e52b31f7a37174000000.pdf.

Simone D'Amico, J. S. Ardaens, and Robin Larsson. **In-flight demonstration of formation control based on relative orbital elements**. In *4th International Conference on Spacecraft Formation Flying Missions & Technologies*, pages 18–20, 2011. URL: https://www.researchgate.net/profile/Jean-Sebastien_Ardaens/publication/225024879_In-Flight_Demonstration_of_Formation_Control_based_on_Relative_Orbital_Elements/links/0c9605231bdbe7011d000000.pdf.

Dawn Aerospace Ltd. **CubeSat Propulsion Modules**, 2020. URL: https://www.dawnaerospace.com/products/cubedrive. Accessed: 2021-07-11.

S. De Florio, S. D'Amico, and G. Radice. **Virtual formation method for precise autonomous absolute orbit control**. Journal of Guidance, Navigation and Control, 2014.

Marcio S. de Queiroz, Vikram Kapila, and Qiguo Yan. **Adaptive nonlinear control of multiple spacecraft formation flying**. *Journal of Guidance Control and Dynamics*, 23(3):385–390, 2000.

Cedric Delmas, Catherine Hourtolle, and Yoann Prevot. **Formation Flight Handling with 4 Satellites - Electronic Earth Observation as an Application**. *15th International Conference on Space Operations, 2018*, pages 1–12, 2018. doi: 10.2514/6.2018-2325.

Marco D'Errico, editor. ***Distributed Space Missions for Earth System Monitoring***. Springer New York, New York, USA, 2013. ISBN 978-1-4614-4540-1. doi: 10.1007/978-1-4614-4541-8. URL: http://link.springer.com/10.1007/978-1-4614-4541-8.

Harijono Djojodihardjo. **The Influence of J2 on Formation Flying of Micro-Satellites in Low Near Equatorial Orbits**. *Journal of Advances in Information Technology*, 5(1), February 2014. ISSN 1798-2340. doi: 10.4304/jait.5.1.5-14. URL: http://www.jait.us/index.php?m=content&c=index&a=show&catid=142&id=734.

John Doyle, Andy Packard, and Kemin Zhou. **Review of LFTs, LMIs, and mu**. In *Proceedings of the 30th IEEE Conference on Decision and Control*, pages 1227–1232, Brighton, U.K., 1991. IEEE. ISBN VO -. doi: 10.1109/CDC.1991.261572.

W Dunbar and R Murray. **Model predicitive control of coordinated multi-vehicle formations**. *Proceedings of the 41st IEEE Conference on Decision and Control*, 2002. doi: DOI10.1109/CDC.2002.1185108.

Eviatar Edlerman and Igal Kronhaus. **Analysis of Electric Propulsion Capabilities in Establishment and Keeping of Formation Flying Nanosatellites**. *Darmstadt, Germany*, March 2016.

ENPULSION GmbH. **IFM NANO Thruster**. Technical report, ENPULSION GmbH, Wiener Neustadt, Austria, 2018. URL: https://www.enpulsion.com/wp-content/uploads/ENP2018-001.G-ENPULSION-NANO-Product-Overview.pdf.

ESA. **About Proba-3**, 2020. URL: https://www.esa.int/Enabling_Support/Space_Engineering_Technology/Proba_Missions/About_Proba-3. Accessed: 2020-11-01.

C. P. Escoubet, M. Fehringer, and M. Goldstein. **Introduction the Cluster mission**. In *Annales Geophysicae*, volume 19, pages 1197–1200, 2001. URL: https://hal.archives-ouvertes.fr/hal-00316909/.

Jesse Koovik Eyer. ***A dynamics and control algorithm for low earth orbit precision formation flying satellites***. PhD thesis, University of Toronto, 2009. URL: https://tspace.library.utoronto.ca/handle/1807/19186.

Wigbert Fehse. ***Automated Rendezvous and Docking of Spacecraft***. Cambridge Aerospace Series. Cambridge University Press, Cambridge, 2003. ISBN 9780511543388. doi: 10.1017/CBO9780511543388. URL: http://ebooks.cambridge.org/ref/id/CBO9780511543388.

G. Gaias and S. D'Amico. **Impulsive maneuvers for formation reconfiguration using relative orbital elements**. *Journal of Guidance, Control, and Dynamics*, 38(6):1036–1049, 2015. doi: DOI10.2514/1.G000189.

Yang Gao and Xinfeng Li. **Optimization of low-thrust many-revolution transfers and Lyapunov-based guidance**. *Acta Astronautica*, 66(1):117–129, 2010. URL: http://www.sciencedirect.com/science/article/pii/S0094576509003257.

Eberhard Gill. **Control Approaches for Formations of Small Satellites**. In *RTO-EN-SCI-231 - Small Satellite Formations For Distributed Surveillance: System Design and Optimal Control Considerations*, chapter 3. NATO Science and Technology Organization, Würzburg, Germany, rto lectur edition, 2011. URL: https://www.cso.nato.int/pubs/rdp.asp?RDP=RTO-EN-SCI-231.

Eberhard Gill, P. Sundaramoorthy, J. Bouwmeester, B. Zandbergen, and R. Reinhard. **Formation Flying within a Constellation of Nano-Satellites: The QB50 Mission**. In *Proceedings of the 6th International Workshop on Satellite Constellation and Formation Flying*, Taipei, Taiwan, 2010.

Dong-Woo Gim and Kyle T. Alfriend. **State Transition Matrix of Relative Motion for the Perturbed Noncircular Reference Orbit**. *Journal of Guidance, Control, and Dynamics*, 26(6):956–971, November 2003. ISSN 0731-5090, 1533-3884. doi: 10.2514/2.6924. URL: http://arc.aiaa.org/doi/abs/10.2514/2.6924.

Dong-Woo Gim and Kyle T. Alfriend. **Satellite Relative Motion Using Differential Equinoctial Elements**. *Celestial Mechanics and Dynamical Astronomy*, 92(4):295–336, August 2005. ISSN 0923-2958, 1572-9478. doi: 10.1007/s10569-004-1799-0. URL: http://link.springer.com/10.1007/s10569-004-1799-0.

GomSpace A/S. **NanoCom AX100 Datasheet**. Technical report, GomSpace A/S, Aalborg, Denmark, 2019. URL: https://gomspace.com/shop/subsystems/communication-systems/nanocom-ax100.aspx.

John Louis Goodman. **History of space shuttle rendezvous and proximity operations**. *Journal of Spacecraft and Rockets*, 43(5):944–959, 2006. URL: http://arc.aiaa.org/doi/abs/10.2514/1.19653.

Colonel GB Green, P. D. Massatt, and N. W. Rhodus. **The GPS 21 primary satellite constellation**. *Navigation*, 36(1):9–24, 1989. URL: http://onlinelibrary.wiley.com/doi/10.1002/j.2161-4296.1989.tb00978.x/abstract.

Moshe Guelman and M. Aleshin. **Optimal Bounded Low-Thrust Rendezvous with Fixed Terminal-Approach Direction**. *Journal of Guidance, Control, and Dynamics*, 24(2):378–385, 2001. ISSN 0731-5090. doi: 10.2514/2.4722. URL: http://doi.aiaa.org/10.2514/2.4722.

Roland Haber, Julian Scharnagl, and Klaus Schilling. **NetSat - A Formation of Four 3U-Satellites Using Electric Propulsion**. In *Proceedings of the 71th International Astronautical Congress*, Online, 2020. IAF.

Edward N. Hartley, Paul A. Trodden, Arthur G. Richards, and Jan M. Maciejowski. **Model predictive control system design and implementation for spacecraft rendezvous**. *Control Engineering Practice*, 20(7):695–713, 2012. ISSN 09670661. doi: 10.1016/j.conengprac.2012.03.009. URL: http://www.sciencedirect.com/science/article/pii/S0967066112000664.

P. Hempel and J. Tschauner. **Über Beschleunigungsprogramme minimaler Übergangsenergie für das Rendezvous-Manöver**. *Astronautica Acta*, 10: 221–237, 1964.

G.W. Hill. **Researches in the Lunar Theory**. *American Journal of Mathematics*, 1(3):245–260, 1878. doi: 10.2307/2369430. URL: http://www.jstor.org/stable/2369313.

Hipparchus. **Hipparchus 1.0 Java Mathematics Library**, 2016. URL: https://www.hipparchus.org/apidocs/index.html. Accessed: 2016-11-04.

S. A. Holmes and W. E. Featherstone. **A unified approach to the Clenshaw summation and the recursive computation of very high degree and order normalised associated Legendre functions**. *Journal of Geodesy*, 76(5):

279–299, May 2002. ISSN 0949-7714, 1432-1394. doi: 10.1007/s00190-002-0216-2. URL: http://link.springer.com/10.1007/s00190-002-0216-2.

Gen-ichiro Hori. **Theory of general perturbation with unspecified canonical variable**. *Publications of the Astronomical Society of Japan*, 18:287, 1966. URL: http://adsabs.harvard.edu/full/1966PASJ...18..287H.

Hyperion Technologies B.V. **GNSS200 - GNSS Receiver**. Technical report, Hyperion Technologies B.V., Delft, Netherlands, 2019. URL: https://hyperiontechnologies.nl/products/gnss200/.

ISIS - Innovative Solutions In Space B.V. **Magnetorquer Board (iMTQ)**. Technical report, ISIS - Innovative Solutions In Space B.V., Delft, Netherlands, 2013. URL: https://www.isispace.nl/wp-content/uploads/2016/02/iMTQ-Brochure-v1.pdf.

Fanghua Jiang, Junfeng Li, and Hexi Baoyin. **Approximate analysis for relative motion of satellite formation flying in elliptical orbits**. *Celestial Mechanics and Dynamical Astronomy*, 98(1):31–66, April 2007. ISSN 0923-2958, 1572-9478. doi: 10.1007/s10569-007-9067-8. URL: http://link.springer.com/10.1007/s10569-007-9067-8.

Marshall H. Kaplan. **Modern spacecraft dynamics and control**. *New York, John Wiley and Sons, Inc., 1976. 427 p.*, 1, 1976.

Michael Kirschner, O Montenbruck, and S Bettadpur. **Flight Dynamics Aspects of the GRACE Formation Flying**. In *Proceedings of the 2nd International Workshop on Satellite Constellations and Formation Flying*, pages 1–8, Haifa, Israel, 2001.

Michael Kirschner, Franz-Heinrich Massmann, and Michael Steinhoff. **GRACE**. In *Distributed Space Missions for Earth System Monitoring*, pages 547–574. Springer, 2013. URL: http://link.springer.com/chapter/10.1007/978-1-4614-4541-8_19.

Adam W Koenig, Tommaso Guffanti, and Simone D'Amico. **New State Transition Matrices for Relative Motion of Spacecraft Formations in Perturbed Orbits**. In *Proceedings of the 2016 AIAA Space and Astronautics Forum and Exposition*, Long Beach, CA, USA, 2016. AIAA. ISBN 978-1-62410-445-9. doi: 10.2514/6.2016-5635.

M. S. Konstantinov, I. A. Nikolichev, and Min Thein. **Optimization of low thrust Multi-Revolution Orbital Transfers using the Method of Dual Numbers**. In *6th International Conference on Astrodynamics Tools and Techniques*, 2016. URL: https://indico.esa.int/indico/event/111/session/21/contribution/99/material/paper/0.pdf.

M Koptev, Y Mashtakov, M Ovchinnikov, and S Shestakov. **Novel Approach To Construction and Maintenance of Tetrahedral Formation**. In *Proceedings of the 9th International Workshop on Satellite Constellations and Formation Flying*, pages 1–11, Boulder, CO, USA, 2017. IAF.

Richard P Kornfeld, Bradford W Arnold, Michael A Gross, Neil T Dahya, William M Klipstein, Peter F Gath, and Srinivas Bettadpur. **GRACE-FO: the gravity recovery and climate experiment follow-on mission**. *Journal of Spacecraft and Rockets*, 56(3):931–951, 2019.

Alexander Kramer, Philip Bangert, and Klaus Schilling. **UWE-4: First electric propulsion on a 1U cubesat-in-orbit experiments and characterization**. *Aerospace*, 7(7):1–21, 2020. ISSN 22264310. doi: 10.3390/AEROSPACE7070098.

Florian Kunzi. ***Development and Testing of a GPS-based Navigation System for Nanosatellites***. Masterthesis, University of Würzburg, 2018.

Derek F. Lawden. ***Optimal trajectories for space navigation***. Butterworths, 1963.

Daero Lee, J. E. Cochran, and J. H. Jo. **Solutions to the variational equations for relative motion of satellites**. *Journal of Guidance, Control, and Dynamics*, 30(3):669–678, 2007. URL: `http://arc.aiaa.org/doi/pdf/10.2514/1.24373`.

Hui Liu, Junfeng Li, and Baoyin Hexi. **Sliding mode control for low-thrust Earth-orbiting spacecraft formation maneuvering**. *Aerospace Science and Technology*, 10(7):636–643, 2006. ISSN 12709638. doi: 10.1016/j.ast.2006.04.008.

Jan Lunze. ***Regelungstechnik 2: Mehrgrößensysteme, Digitale Regelung***. Springer, Berlin, Germany, 8th edition, 2014a. ISBN 978-3-642-53943-5. doi: 10.1007/978-3-642-53944-2. URL: `http://www.springer.com/978-3-642-53943-5?wt_mc=ThirdParty.SpringerLink.3.EPR653.About_eBook`.

Jan Lunze. ***Regelungstechnik 1: Systemtheoretische Grundlagen, Analyse und Entwurf einschleifiger Regelungen***. Springer, Heidelberg, Germany, 10th edition, 2014b. ISBN 978-3-642-53908-4. doi: 10.1007/978-3-642-53909-1. URL: `http://www.springer.com/978-3-642-53908-4?wt_mc=ThirdParty.SpringerLink.3.EPR653.About_eBook`.

Jan Lunze. ***Networked Control of Multi-Agent Systems***. Bookmundo Direct, 2019. ISBN 9789463867139.

D. C. Maessen and Eberhard Gill. ***Relative orbital element estimation and observability analysis for formation flying satellites using inter-satellite range measurements only***. American Institute of Aeronautics and Astronautics (AIAA), 2010. URL: `http://arc.aiaa.org/doi/pdf/10.2514/6.2010-7885`.

Kris Maine, Carrie Devieux, and Pete Swan. **Overview of IRIDIUM satellite network**. In *WESCON/'95. Conference record.'Microelectronics Communications Technology Producing Quality Products Mobile and Portable Power Emerging Technologies'*, page 483. IEEE, 1995. URL: http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=485428.

L. Maisonobe and P. Parraud. **From low level Toolbox to Orbit Determination: Handling Users Requests in Orekit**. In *6th International Conference on Astrodynamics Tools and Techniques*, 2016. URL: https://indico.esa.int/indico/event/111/session/27/contribution/73/material/paper/0.pdf.

S. Mathavaraj and R. Padhi. ***Satellite Formation Flying - High Precision Guidance using Optimal and Adaptive Control Techniques***. Springer Nature Singapore, Singapore, China, 1 edition, 2021. ISBN 978-981-15-9630-8.

MathWorks. **Optimization Problem**, 2020. URL: https://de.mathworks.com/help/mpc/ug/optimization-problem.html.

Oliver Montenbruck and Eberhard Gill. ***Satellite orbits: models, methods and applications***. Springer Science & Business Media, 2012. URL: https://books.google.de/books?hl=de&lr=&id=tLU-wjQktoMC&oi=fnd&pg=PR5&dq=Satellite+Orbits+%E2%80%93+Models,+Methods,+Applications&ots=GE3KIJ6iMp&sig=67yLGiFC25KbUJ8HAkP7NcH9pIs.

Daniel Morgan, Soon-Jo Chung, Lars Blackmore, Behcet Acikmese, David Bayard, and Fred Y. Hadaegh. **Swarm-keeping strategies for spacecraft under J2 and atmospheric drag perturbations**. *Journal of Guidance, Control, and Dynamics*, 35(5):1492–1506, 2012. URL: http://arc.aiaa.org/doi/pdf/10.2514/1.55705.

NewSpace Systems. **Magnetometer - NMRM-Bn25o485**. Technical report, NewSpace Systems, Somerset West, South Africa, 2020. URL: https://www.newspacesystems.com/wp-content/uploads/2020/10/NewSpace-Magnetometer_2020_10a.pdf.

Austin Kyle Nicholas. ***Attitude and formation control design and system simulation for a three-satellite CubeSat mission***. PhD thesis, Massachusetts Institute of Technology, 2013. URL: http://dspace.mit.edu/handle/1721.1/82488.

Marco M. Nicotra, Mihovil Bartulovic, Emanuele Garone, and Bruno Sinopoli. **A Distributed Explicit Reference Governor for Constrained Control of Multiple UAVs**. In *Proceedings of the 5th IFAC Workshop on Distributed Estimation and Control in Networked Systems*, pages 156–161, Philadelphia, PA, USA, 2015. IFAC. doi: 10.1016/j.ifacol.2015.10.323. URL: http://linkinghub.elsevier.com/retrieve/pii/S2405896315022144.

Marco M. Nicotra, Roberto Naldi, and Emanuele Garone. **A Robust Explicit Reference Governor for Constrained Control of Unmanned Aerial Vehicles**. In *Proceedings of the 2016 American Control Conference*, pages 6284–6289, Boston, MA, USA, 2016. ISBN 9781467386814.

Tiago Nogueira, Julian Scharnagl, Stavros Kotsiaros, and Klaus Schilling. **NetSat-4G: A four nano-satellite formation for global geomagnetic gradiometry**. In *Proceedings of the 10th IAA Symposium on Small Satellites for Earth Observation*, Berlin, Germany, 2015. IAA.

Tiago Nogueira, Slavi Dombrovski, Stephan Busch, Klaus Schilling, Klemen Zakšek, and M. Hort. **Photogrammetric Ash Cloud Observations by Small Satellite Formations**. In *Proceedings of the 3rd IEEE Workshop on Metrology for Aerospace*, pages 3–8, Florence, Italy, 2016. IEEE.

NovAtel Inc. **OEM7700 - Multi-Frequency, GNSS Receiver Delivers Robust Positioning and Simplifies Integration**. Technical report, NovAtel Inc., Calgary, Alberta, Canada, 2020. URL: `https://novatel.com/support/high-precision-gnss-gps-receivers/oem-receiver-boards/oem7-receivers/oem7700`.

OREKIT. **OREKIT 8.0 Spaceflight Dynamics Java Library**, 2016. URL: `https://www.orekit.org/site-orekit-8.0/apidocs/index.html`. `Accessed: 2016-11-04`.

Erica Peterson, Robert Zee, and Georgia Fotopoulos. **Possible Orbit Scenarios for an InSAR Formation Flying Microsatellite Mission**. In *Proceedings of the 2008 Small Satellite Conference*, Logan, UT, USA, 2008. Utah State University. URL: `http://digitalcommons.usu.edu/smallsat/2008/all2008/35/`.

Anastassios E. Petropoulos. **Refinements to the Q-law for low-thrust orbit transfers**. *Advances in the Astronautical Sciences*, 120:963–983, 2005.

J. Pluym and C. Damaren. **Dynamics and control of spacecraft formation flying: Reference orbit selection and feedback control**. *Proceedings of the 13th Canadian Astronautics Conference*, 2006.

Michael E. Polites. **Technology of automated rendezvous and capture in space**. *Journal of Spacecraft and Rockets*, 36(2):280–291, 1999. URL: `http://arc.aiaa.org/doi/pdf/10.2514/2.3443`.

M. Queiroz, V. Kapila, and Q. Yan. **Adaptive nonlinear control of multiple spacecraft formation flying**. *Journal of Guidance, Control, and Dynamics*, 2000.

David C. Redding, Neil J. Adams, and Edward T. Kubiak. **Linear-Quadratic Stationkeeping for the STS Orbiter**. *Journal of Guidance*, 12(2):248–255,

1989. ISSN 0731-5090. doi: 10.2514/3.20398. URL: http://arc.aiaa.org/doi/abs/10.2514/3.20398.

Eric Reinthal. ***Relative orbit propagation and control for satellite formation flying using continuous low-thrust***. Masterthesis, University of Würzburg, 2016.

Wei Ren and Randal W. Beard. ***Distributed Consensus in Multi-vehicle Cooperative Control: Theory and Applications***. Springer London, London, U.K., 2008. ISBN 978-1-84800-014-8 DOI. doi: 10.1007/978-1-84800-015-5.

Lucas Riggi and Simone D'Amico. **Optimal Impulsive Closed-Form Control for Spacecraft Formation Flying and Rendezvous**. In *Proceedings of the 2016 American Control Conference*, Boston, MA, USA, 2016.

A. Robertson, G. Inalhan, and J. How. **Formation control strategies for a separated spacecraft interferometer**. *Proceedings of the American Control Conference*, 1999.

S4 - Smart Small Satellite Systems GmbH. **Dynamic Bench Test Facility**. Technical report, S4 - Smart Small Satellite Systems GmbH, Würzburg, Germany, 2019. URL: https://www.s4-space.com/files/S4_Flyer_TurnTables_Datasheet.pdf.

K. Sarda, N. Roth, R. E. Zee, Dan CaJacob, and Nathan G. Orr. **Making the Invisible Visible: Precision RF-Emitter Geolocation from Space by the HawkEye 360 Pathfinder Mission**. In *Proceedings of the 32nd Annual AIAA/USUConference on Small Satellites*, Logan, UT, USA, 2018. AIAA. ISBN 1416667792.

Niyazi Savran. ***Integration of an Attitude Simulation Framework in a Multi Satellite Formation Flying Simulation Framework***. Masterthesis, University of Würzburg, 2020.

Daniel P. Scharf, Fred Y. Hadaegh, and Scott R. Ploen. **A survey of spacecraft formation flying guidance and control. part ii: control**. In *American Control Conference, 2004. Proceedings of the 2004*, volume 4, pages 2976–2985. IEEE, 2004. URL: http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=1384365.

Julian Scharnagl, Florian Kempf, and Klaus Schilling. **Combining Distributed Consensus with Robust H-Infinity-Control for Satellite Formation Flying**. *Electronics*, 8(3):319, 2019. ISSN 2079-9292. doi: 10.3390/electronics8030319. URL: http://www.mdpi.com/2079-9292/8/3/319.

Anne Schattel, Mitja Echim, Matthias Knauer, and Christof Büskens. **Low Thrust Trajectory Optimization for Autonomous Asteroid Rendezvous**

Missions. In *6th International Conference on Astrodynamics Tools and Techniques*, 2016. URL: https://indico.esa.int/indico/event/111/session/23/contribution/55/material/paper/0.pdf.

H. Schaub, S. R. Vadali, J. L. Junkins, and K. T. Alfriend. **Spacecraft formation flying control using mean orbit elements**. *Journal of the Astronautical Sciences*, 2000a.

Hanspeter Schaub and John L. Junkins. ***Analytical Mechanics of Space Systems***. AIAA Education Series, Reston, VA, USA, 2003. doi: 10.2514/4.861550.

Hanspeter Schaub, Srinivas R. Vadali, John L. Junkins, and Kyle T. Alfriend. **Spacecraft formation flying control using mean orbit elements**. *Journal of the Astronautical Sciences*, 48(1):69–87, 2000b. URL: http://dnc.tamu.edu/drjunkins/yearwise/2000/archival/JAS_V48_N1_2000_vscmg.pdf.

Klaus Schilling, Philip Bangert, Stephan Busch, Slavi Dombrovski, Andreas Freimann, Alexander Kramer, Tiago Nogueira, Dmitry Ris, Julian Scharnagl, and Tristan Tzschichholz. **NetSat: A Four Pico/Nano-Satellite Mission for Demonstration of Autonomous Formation Flying**. In IAF, editor, *Proceedings of the 66th International Astronautical Congress*, Jerusalem, Israel, 2015.

Klaus Schilling, Tristan Tzschichholz, Iurii Motroniuk, Ilham Mammadov, Oliver Ruf, Christopher Schmidt, Nicolas Appel, Alexander Kleinschrodt, Sergio Montenegro, and Andreas Nüchter. **TOM: A Formation for Photogrammetric Earth Observation by Three CubeSats**. In *Proceedings of the 4th IAA Conference on University Satellite Missions and CubeSat Workshop*, Rome, Italy, 2017.

Klaus Schilling, Anna Aumann, Iurii Motroniuk, Ilham Mammadov, Daniel Garbe, Oliver Ruf, Veaceslav Dombrovski, and Maros Hladký. **TOM - A Pico-Satellite Formation For 3D Earth Observation**. In *Proceedings of the 4S Symposium on Small Satellites Systems and Services*, Sorrento, Italy, 2018. ESA.

Klaus Schilling, Yoav Y Schechner, and Ilan Koren. **CloudCT - Computed Tomography of Clouds by a Small Satellite Formation**. In *Proceedings of the 12th IAA Symposium on Small Satellites for Earth Observation*, Berlin, Germany, 2019. IAA.

Marco Schmidt, Rajesh Shankar Priya, and Klaus Schilling. **The PICO-Satellite UWE-1 and IP Based Telecommunication Experiments**. In *Proceedings of the 17th IFAC Symposium on Automatic Control in Aerospace*, Toulouse, France, 2007. IFAC.

Marco Schmidt, Karthik Ravandoor, Oliver Kurz, Stephan Busch, and Klaus Schilling. **Attitude Determination for the Pico-Satellite UWE-2**. In

Proceedings of the 17th World Congress, pages 14036–14041, Seoul, South Korea, 2008. IFAC. ISBN 9781123478.

Samuel A. Schweighart and Raymond J. Sedwick. **Development and analysis of a high fidelity linearized J2 model for satellite formation flying**. *SM Thesis, Massachusetts Institute of Technology, Dept. Aeronautics and Astronautics*, 2001. URL: http://arc.aiaa.org/doi/pdf/10.2514/6.2001-4744.

Prasenjit Sengupta. ***Dynamics and control of satellite relative motion in a central gravitational field***. PhD thesis, Texas A&M University, 2007. URL: http://oaktrust.library.tamu.edu/handle/1969.1/4724.

A. Surjalal Sharma and Steven A. Curtis. **Magnetospheric multiscale mission**. In A. S. Sharma and P. K. Kaw, editors, *Nonequilibrium Phenomena in Plasmas*, chapter 8, pages 179–196. Springer, The Netherlands, 2005.

Ryan Sherrill. ***Dynamics and Control of Satellite Relative Motion in Elliptic Orbits using Lyapunov-Floquet Theory***. PhD thesis, Auburn University, 2013. URL: https://holocron.lib.auburn.edu/handle/10415/3594.

G. Sholomitsky. **Infra-red space interferometer**. In *28th Int. Astro. Fed. Congress, 1977*, 1977. URL: http://ci.nii.ac.jp/naid/10019785729/.

Sigurd Skogestad and Ian Postlethwaite. ***Multivariable Feedback Control: Analysis and Design***. Wiley, Chichester, U.K., 2nd editio edition, 2001. ISBN 978-0470011683. doi: 978-0-470-01167-6. URL: http://www.nt.ntnu.no/users/skoge/book.html.

Dan Smith. **Operations innovations for the 48-satellite globalstar constellation**. In *Proceedings of the 16th AIAA International Communications Satellite Systems Conference*, volume 542. Washington, DC: American Institute of Aeronautics and Astronautics, 1996. URL: http://arc.aiaa.org/doi/pdf/10.2514/6.1996-1051.

Solar MEMS Technologies. **SSOC-D60 - Technical specification, Interfaces and Operation**. Technical report, Solar MEMS Technologies, Seville, Spain, 2016. URL: https://www.solar-mems.com/ssoc/.

Space Station Control Board. **SSP 30219 Revision J - Space Station Reference Coordinate Systems**. Technical report, NASA, ESA, JAXA, CSA, ASI, RSA, Houston, TX, USA, 2008.

Lukas M. Steindorf, Simone D. Amico, Julian Scharnagl, Florian Kempf, and Klaus Schilling. **Constrained Low-Thrust Satellite Formation-Flying Using Relative Orbit Elements**. In *Proceedings of the 27th AAS/AIAA Space Flight Mechanics Meeting*, San Antonio, Texas, USA, 2017. AAS.

Victor Szebehely and Giorgio EO Giacaglia. **On the elliptic restricted problem of three bodies**. *The Astronomical Journal*, 69:230, 1964. URL: http://adsabs.harvard.edu/full/1964AJ.....69..230S.

Chengpan Tang, Xiaogong Hu, Shanshi Zhou, Li Liu, Junyang Pan, Liucheng Chen, Rui Guo, Lingfeng Zhu, Guangming Hu, Xiaojie Li, Feng He, and Zhiqiao Chang. **Initial results of centralized autonomous orbit determination of the new-generation BDS satellites with inter-satellite link measurements**. *Journal of Geodesy*, 92(10):1155–1169, 2018. ISSN 14321394. doi: 10.1007/s00190-018-1113-7. URL: https://doi.org/10.1007/s00190-018-1113-7.

Yu-Ping Tian. ***Frequency-Domain Analysis and Design of Distributed Control Systems***. Wiley, Singapore, China, 1st editio edition, 2012. ISBN 9780470828205.

Michael Tillerson, Gokhan Inalhan, and Jonathan P. How. **Co-ordination and control of distributed spacecraft systems using convex optimization techniques**. *International Journal of robust and nonlinear control*, 12(2-3):207–242, 2002. URL: http://onlinelibrary.wiley.com/doi/10.1002/rnc.683/abstract.

J. Tschauner and P. Hempel. **Optimale Beschleunigungsprogramme für das Rendezvous-Manöver**. *Astronautica Acta*, 10:296–307, 1964.

Yuri Ulybyshev. **Long-Term Formation Keeping of Satellite Constellation Using Linear-Quadratic Controller**. *Journal of Guidance, Control, and Dynamics*, 21(1):109–115, January 1998. ISSN 0731-5090, 1533-3884. doi: 10.2514/2.4204. URL: http://arc.aiaa.org/doi/abs/10.2514/2.4204.

S. R. Vadali, S. S. Vaddi, K. Naik, and Kyle T. Alfriend. **Control of Satellite Formations**. In *Proceedings of the AIAA Guidance, Navigation and Control Conference and Exhibit*, Montreal,Canada, 2001. AIAA. ISBN 9781563479786. doi: 10.2514/6.2001-4028. URL: http://dx.doi.org/10.2514/6.2001-4028.

David A. Vallado and Salvatore Alfano. **Curvilinear Coordinates for Covariance and Relative Motion Operations**. In *2011 AAS/AIAA Astrodynamics Specialist Conference*, Girdwood, AK, USA, 2011. Analytical Graphics Inc.

Gabor I. Varga and Jose M. Sanchez Perez. **Many-Revolution low-thrust Orbit Transfer Computation using Equinoctial Q-Law including J2 and Eclipse Effects**. *AIAA/AAS Paper*, pages 15–590, 2016. URL: https://indico.esa.int/indico/event/111/session/21/contribution/97/material/paper/0.pdf.

John G. Walker. **Satellite constellations**. *Journal of the British Interplanetary Society*, 37:559–572, 1984. URL: http://adsabs.harvard.edu/abs/1984JBIS...37..559W.

M. J. H. Walker, B. Ireland, and Joyce Owens. **A set modified equinoctial orbit elements**. *Celestial mechanics*, 36(4):409–419, 1985. URL: http://link.springer.com/article/10.1007/BF01227493.

H. G. Walter. **Conversion of osculating orbital elements into mean elements**. *The Astronomical Journal*, 72:994, 1967. URL: http://adsabs.harvard.edu/full/1967AJ.....72..994W]].

Liuping Wang. ***Model Predictive Control System Design and Implementation Using MATLAB***. Springer London, London, U.K., 2009. ISBN 9780857296344. doi: 10.1007/978-1-4471-4399-4. URL: http://www.springer.com/engineering/control/book/978-0-85729-634-4.

R. M. Weisman, M. Majji, and K. T. Alfriend. **Analytic characterization of measurement uncertainty and initial orbit determination on orbital element representations**. *Celestial Mechanics and Dynamical Astronomy*, 118(2):165–195, 2014. URL: http://link.springer.com/article/10.1007/s10569-013-9529-0.

Eric W. Weisstein. **Cylindrical Equidistant Projection**, 2016. URL: http://mathworld.wolfram.com/CylindricalEquidistantProjection.html. Accessed: 2016-07-14.

Wikipedia Contributors. **Guidance, Navigation, and Control — Wikipedia, The Free Encyclopedia**, 2020a. URL: https://en.wikipedia.org/w/index.php?title=Guidance,_navigation,_and_control&oldid=961757278. Accessed: 2020-11-01.

Wikipedia Contributors. **Model Predictive Control — Wikipedia, The Free Encyclopedia**, 2020b. URL: https://en.wikipedia.org/w/index.php?title=Model_predictive_control&oldid=988269842. Accessed: 2020-11-01.

Trevor Williams, Neil Ottenstein, Eric Palmer, and Mitra Farahmand. **Initial Satellite Formation Flight Results from the Magnetospheric Multiscale Mission**. In *Proceedings of the 2016 AIAA/AAS Astrodynamics Specialist Conference*, Long Beach, CA, USA, 2016. AIAA. URL: http://link.springer.com/chapter/10.1007/1-4020-3109-2_8.

WITTENSTEIN cyber motor GmbH. **Cyber ® Reaction Wheel 2**. Technical report, WITTENSTEIN cyber motor GmbH, Igersheim, Germany, 2020. URL: https://cyber-motor.wittenstein.de/de-de/reaktionsrad-satellit/.

Katsuhiko Yamada, Masaya Kimura, Takeya Shima, and Shoji Yoshikawa. **New state transition matrix for formation flying in J2-perturbed elliptic orbits**. *Journal of Guidance, Control, and Dynamics*, 35(2):536–547, 2012. URL: http://arc.aiaa.org/doi/pdf/10.2514/1.53926.

Koji Yamanaka and Finn Ankersen. **New State Transition Matrix for Relative Motion on an Arbitrary Elliptical Orbit**. *Journal of Guidance, Control, and Dynamics*, 25(1):60–66, 2002. ISSN 0731-5090. doi: 10.2514/2.4875. URL: `http://arc.aiaa.org/doi/abs/10.2514/2.4875`.

Bruce Yost and Sasha (ed.) Weston. **State-of-the-Art Small Spacecraft Technology**. Technical report, Ames Research Center, Moffett Field, CA, USA, 2020. URL: `https://www.nasa.gov/sites/default/files/atoms/files/soa2020_final3.pdf`.

Kemin Zhou, John C. Doyle, and Keith Glover. ***Robust and Optimal Control***. Prentice Hall, Englewood Cliffs, New Jersey, USA, 1996. ISBN 0134565673. doi: 10.1016/0967-0661(96)83721-X. URL: `http://www.ulb.tu-darmstadt.de/tocs/109091736.pdf`.

The author appreciated the support by the European Research Council (ERC) within the Advanced Grant "NetSat" under the Grant Agreement No. 320377 and the support by the European Space Agency (ESA) within the Networking/Partnering Initiative (NPI) under Contract No. 4000113715/15/NL/MH/ats.

Lehrstuhl Informatik VII
Robotik und Telematik
Am Hubland
D-97074 Wuerzburg

Tel.: +49 (0) 931 - 31 - 86678
Fax: +49 (0) 931 - 31 - 86679

schi@informatik.uni-wuerzburg.de
http://www7.informatik.uni-wuerzburg.de

ISSN: 1868-7474 (online)
ISSN: 1868-7466 (print)
ISBN: 978-3-945459-42-3 (online)

Die Schriftenreihe

wird vom Lehrstuhl für Informatik VII: Robotik und Telematik der Universität Würzburg herausgegeben und präsentiert innovative Forschung aus den Bereichen der Robotik und der Telematik.

Die Kombination fortgeschrittener Informationsverarbeitungsmethoden mit Verfahren der Regelungstechnik eröffnet hier interessante Forschungs- und Anwendungsperspektiven. Es werden dabei folgende interdisziplinäre Aufgabenschwerpunkte bearbeitet:

- Robotik und Mechatronik: Kombination von Informatik, Elektronik, Mechanik, Sensorik, Regelungs- und Steuerungstechnik, um Roboter adaptiv und flexibel ihrer Arbeitsumgebung anzupassen.

- Telematik: Integration von Telekommunikation, Informatik und Steuerungstechnik, um Dienstleistungen an entfernten Standorten zu erbringen.

Anwendungsschwerpunkte sind u.a. mobile Roboter, Tele-Robotik, Raumfahrtsysteme und Medizin-Robotik.

Zitation dieser Publikation

SCHARNAGL, J. (2022). Distributed Guidance, Navigation and Control for Satellite Formation Flying Missions. Schriftenreihe Würzburger Forschungsberichte in Robotik und Telematik, Band 26. Würzburg: Universität Würzburg.
DOI: 10.25972/OPUS-28753

Dissertation an der Universität Würzburg im Rahmen der Graduate School of Science and Technology