

Christoffer D. Hystad

**The title of your master's thesis should be  
written here**

Any undertitle is written here

Master's thesis in Cybernetics and Robotics  
Supervisor: Jan-Tommy Gravdahl  
December 2025

Norwegian University of Science and Technology  
Faculty of Information Technology and Electrical Engineering  
Department of Engineering Cybernetics





## ABSTRACT

Write an abstract/summary of your thesis, and state your main findings here.

A summary should be included in both English and any second language, if this is applicable, regardless if the thesis is written in English or in your preferred language. These should be on separate pages, the English version first.

## PREFACE

Write the preface of your thesis here.

You may include acknowledgements and thanks as part of your preface on this page, or you may add it as a new chapter after the preface.

# CONTENTS

<b>Abstract</b>	i
<b>Preface</b>	ii
<b>Contents</b>	v
<b>List of Figures</b>	v
<b>List of Tables</b>	vi
<b>Abbreviations</b>	vii
<b>1 Introduction</b>	1
1.1 Prelim intro . . . . .	1
1.2 Background & Motivation . . . . .	2
1.3 Problem Definition . . . . .	3
1.4 Overall Method . . . . .	3
1.5 Project Scope & Limitations . . . . .	3
1.6 Report Structure . . . . .	3
<b>2 Literature Review</b>	4
2.1 Orbital Propagation . . . . .	5
2.1.1 Analytical Propagation (Simplified General Perturbations 4 (SGP4))	5

2.1.2	Numerical Orbit Propagation . . . . .	8
2.1.3	Semianalytical and Hybrid Propagators . . . . .	8
2.1.4	Propagator Comparison and Validation . . . . .	8
2.2	Orbital Perturbation Modelling . . . . .	8
2.2.1	Gravity-Field Perturbations . . . . .	8
2.2.2	Atmospheric Drag . . . . .	8
2.2.3	Solar Radiation Pressure . . . . .	8
2.2.4	Third-Body Perturbations . . . . .	8
2.3	Error Growth and Propagation Stability . . . . .	8
2.3.1	Long-Term Orbit Prediction Accuracy . . . . .	8
2.3.2	Sensitivity to Force-Model Fidelity . . . . .	8
2.3.3	Numerical Integration Error . . . . .	8
2.3.4	Regime-Dependent Error Growth . . . . .	8
2.4	Satellite Formation Flying and Relative Motion . . . . .	8
2.4.1	Relative Motion Dynamics . . . . .	8
2.4.2	Sensitivity of Relative Motion to Propagation Error . . . . .	8
2.4.3	Formation-Keeping delta-V Requirements . . . . .	8
2.4.4	Disturbance Effects on Formation Stability . . . . .	8
2.5	Summary of Literature Review . . . . .	8
<b>3</b>	<b>Theory</b>	<b>12</b>
3.1	Numerical Precision . . . . .	12
3.2	Numerical Solvers . . . . .	12
3.3	Vector notation . . . . .	12
3.4	Reference Frames . . . . .	12
3.4.1	Inertial ECI frame . . . . .	12
3.4.2	RTN frame . . . . .	12

3.5	Orbital Mechanics . . . . .	13
3.5.1	Classical Orbital Elements (COEs) . . . . .	13
3.6	Orbital Disturbances . . . . .	13
3.6.1	Earth's Oblateness . . . . .	13
3.6.2	Atmospheric Drag . . . . .	13
3.6.3	Solar Radiation Pressure . . . . .	13
3.7	Two-Line Elements (TLEs) . . . . .	14
3.8	SGP4 Propagator . . . . .	14
<b>4</b>	<b>Methods</b>	<b>15</b>
4.1	Simulation Design . . . . .	15
4.2	Literature Search Methodology . . . . .	15
4.3	Software Development . . . . .	15
4.3.1	Software Architecture . . . . .	15
<b>5</b>	<b>Results</b>	<b>18</b>
5.1	Prelim results . . . . .	18
5.1.1	Simulator output difference plots . . . . .	18
5.1.2	R . . . . .	19
<b>6</b>	<b>Discussion</b>	<b>20</b>
6.1	Future work . . . . .	20
<b>7</b>	<b>Conclusions</b>	<b>21</b>
<b>References</b>		<b>22</b>
<b>Appendices:</b>		<b>24</b>
<b>A - Github repository</b>		<b>25</b>

## LIST OF FIGURES

1.2.1 Launch cost per kilogram to LEO versus first launch date, courtesy of [2] . 2

## LIST OF TABLES

2.5.1 Literature overview for the <i>Orbital propagation</i> theme . . . . .	9
2.5.2 Literature overview for the <i>Orbital perturbation modelling</i> theme . . . . .	10
2.5.3 Literature overview for the <i>Error growth &amp; propagation stability</i> theme . .	10
2.5.4 Literature overview for the <i>Satellite formation flying</i> theme . . . . .	11

---

CHAPTER  
ONE

---

INTRODUCTION

## 1.1 Prelim intro

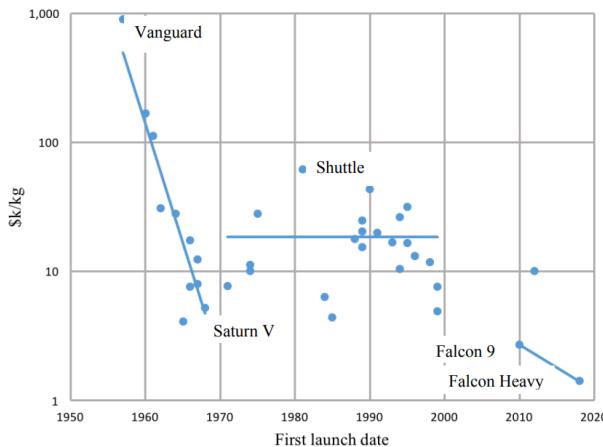
The project: "GNSS-R: Formation flying of small satellites" is a comparative study that investigates the accuracy of orbital propagation methods used within NTNU SmallSatLab's GNSS-R mission studies over longer time horizons. The analysis focuses on translational dynamics and compares two distinct simulation frameworks: the baseline SGP4 propagator implemented through the Skyfield library, and an extended high-fidelity propagator implemented in Basilisk. The latter enables the inclusion of additional perturbation models, such as Earth's gravity gradient, atmospheric drag, solar radiation pressure, and third-body pull, whose impact on long-term orbit prediction is assessed relative to the SGP4 baseline. To guide this investigation, the project addresses three key research questions:

- What is the intrinsic numerical difference between the Skyfield-SGP4 and the Basilisk frameworks when configured identically?
- How does the introduction of extended perturbation models influence the divergence between the frameworks over time?
- How do differences in simulated translational states affect formation-keeping delta-V analyses?

Answering these questions will support the project's goal of guaranteeing the reliability and accuracy of orbital propagation tools used for GNSS-R small-satellite formation studies in future mission design.

## 1.2 Background & Motivation

Global Navigation Satellite Systems (GNSS) have become deeply embedded into our modern infrastructure with multiple constellations, such as GPS, GLONASS, Galileo and BeiDou, ensuring that four satellites are observable from the surface at any given time, thus providing worldwide coverage [1]. The sheer number of operational satellites that broadcast GNSS signals makes it one of the most powerful signals of opportunity for remote sensing, enabling monitoring without the need for onboard transmitters.



**Figure 1.2.1:** Launch cost per kilogram to LEO versus first launch date, courtesy of [2]

At the same time, the space sector has recently undergone rapid changes. Launch costs have fallen significantly due to government outsourcing of launch providers to private companies, and the competitive environment it produced. This is clearly shown in figure 1.2.1, where the per-kilogram launch cost has hit an all-time low with the deployment of SpaceX rockets *Falcon 9* and *Falcon Heavy* [2]. As a result, it has become increasingly feasible for universities, research groups and smaller nations to deploy space-based systems dedicated to leveraging GNSS signals for remote sensing applications.

In parallel with this growth, there has also been an increase in the frequency of reported intentional GNSS interference to disrupt aviation operations. The interference can be divided into separate categories characterized by their difference in method and effect: *Jamming* refers to the relative high-power emission of Radio Frequency (RF) noise to block GNSS signals, while *spoofing* is the transmission of signals that resemble those from real satellites, misleading the GNSS receiver to think it is in a different position [3]. The Norwegian Defense Research Establishment (FFI) states that: "In Norway's northernmost county of Finnmark, close to the Russian border, loss of GNSS signals due to jamming occurs nearly daily. [...] The number of disruptions has increased significantly since Russia's invasion of Ukraine in 2022." [4]. Data from [gpsjam.org](http://gpsjam.org) suggest that similar disruptions have been reported across Eastern Europe, the Middle East, and the Arctic [5]. These events highlight a broader geopolitical trend: GNSS interference is becoming more common. This places increased strategic value on systems that can detect and localize jamming/spoofing events.

GNSS Reflectometry (GNSS-R) offers a promising technique for addressing these

challenges. According to Jin and Komjathy [6], Hall and Cordey [7] were the first to propose that GNSS L-band reflections could be exploited in a bistatic radar configuration to retrieve ocean surface properties, laying the foundation for modern GNSS-R techniques. Recent theory suggest that the same bistatic configuration can be used for surveillance of the GNSS spectrum, where multiple GNSS-R satellites in formation can detect and localize sources of jamming and spoofing. This emerging use-case is one of the central research goals of NTNU SmallSat Lab's GNSS-R project, which aim to develop methods for this process [8]. The project currently investigates this concept through its GNSS-R satellite mission study, exploring formations geometries, orbit configurations and system designs that would enable robust detection of GNSS interference sources and ships.

The specialization project supports the GNSS-R mission study by examining and improving the accuracy of the orbital simulations used

I am

### 1.3 Problem Definition

### 1.4 Overall Method

### 1.5 Project Scope & Limitations

### 1.6 Report Structure

---

## CHAPTER

## TWO

---

## LITERATURE REVIEW

Although astrodynamics is rooted in centuries of celestial mechanics, it only recently emerged as a distinct scientific discipline. As outlined by Szebehely's historical account of the field [9], the development of astrodynamics may be divided into four broad periods: the pre-Newtonian era, the classical or Newtonian period, the contributors from the nineteenth century and finally the rapid expansion that took place in the twentieth century. Although its conceptual roots stretch back to ancient astronomers such as Aristotle and Hipparchus, and its mathematical foundations were laid by Newton's *Principia* [10], Szebehely emphasizes that the practical astrodynamics discipline concerned with applying celestial and ballistic mechanics to artificial bodies, was only formed in the early 1930s. Following the second world war, the field saw rapid growth due to Soviet launch of the *Sputnik* satellite in 1957, and the ensuing space race. With the introduction of more powerful digital computers in the 1960s, Brouwer, Herrick and others led research teams that laid the foundation for analytical and numerical orbit propagation.

Building on this historical foundation, the field has continued to evolve through substantial research effort aimed at increasing predictive accuracy. This trend is indicated by the sharp increase in published work since the early 1990s proposing improved numerical integration methods, orbital propagation schemes and disturbance models. This chapter therefore provides an overview of the theoretical foundations and prior research relevant to this project that have been published after 1960. To provide structure while keeping the scope focused, the review is divided into four themes, each covering aspects of astrodynamics whose understanding is essential for addressing the research questions presented in Section 1.1. The first theme covers analytical and numerical orbital propagation methods and studies assessing their intrinsic accuracy. The second theme examines orbital perturbation models, specifically gravity-field perturbations, atmospheric drag and Solar Radiation Pressure (SRP). These two themes together provide the methodological background for understanding the differences between the Skyfield SGP4 baseline and the relatively high-fidelity Basilisk implementations.

The third theme envelops the error growth and propagation stability of propaga-

tors with a focus on long-term orbit prediction accuracy, sensitivity to perturbation-model fidelity and numerical integration error. Finally, the forth theme covers satellite formation-keeping, and reviewing how absolute propagation error translate into inaccuracies in relative-state predictions and, consequently, into differences in delta-V estimates for formation maintenance. The last two themes is directly related to the project's investigation of how simulator differences evolve over time, and will provide the theory necessary to understand discrepancies between propagations.

The purpose of this literature review is to map the existing knowledge that supports the analysis carried out in this project. The aim is not to identify research gaps or propose new theoretical developments, but rather provide the theoretical background needed to understand the propagation methods, orbital perturbations and mechanisms causing prediction error over longer time-horizons, which are covered in depth in chapter 3. This mapping of established models and methods also serves to contextualize their relative accuracies, forming the foundation for the simulation-based comparison presented in later chapters.

## 2.1 Orbital Propagation

### 2.1.1 Analytical Propagation (SGP4)

Analytical orbit propagation methods rely on closed-form or series-based approximations of the perturbing forces acting on an artificial satellite. Among these, the SGP4 algorithm is the most widely used and has become the de facto standard for propagating publicly distributed Two-Line Element (TLE) sets. As described in detail by Vallado [11], SGP4 operates on mean orbital elements rather than osculating elements, and applies a sequence of analytical corrections to account for dominant perturbations. These corrections include secular and periodic contributions from the Earth's oblateness, simplified treatments of atmospheric drag, and, in some cases, resonance effects. Because the dynamics are expressed through closed-form perturbation series, SGP4 achieves exceptional computational speed and remains efficient even when used to propagate thousands of objects, which has contributed significantly to its long-standing operational use.

Despite these advantages, the analytical nature of SGP4 introduces several well-known limitations. Since the algorithm uses simplified perturbation models and a low-fidelity representation of atmospheric drag, its accuracy deteriorates over longer time horizons, especially in low Earth orbit where drag and solar radiation pressure can drive significant deviations from the mean-element trajectory. Liu *et al.* [12] demonstrate this behaviour empirically, showing that the difference between SGP4 predictions and high-fidelity numerical ephemerides grows noticeably with propagation time. Their work further illustrates how machine-learning techniques can be used to correct these long-term deviations, highlighting the underlying structural limitations of the analytical model. Similarly, Conkey and Zielinski [13] compare classical SGP4 with the extended-precision SGP4-XP variant, showing that improved analytical treatments of atmospheric drag, resonance terms and solar radiation pressure can offer measurable accuracy gains, particularly outside low Earth orbit.

More broadly, studies comparing with high-precision numerical propagators provide valuable insight into the intrinsic accuracy of analytical propagation. Acciarini *et al.* [14] analyse the divergence between SGP4 and a high-fidelity numerical integrator that models detailed perturbations, demonstrating that the differences become increasingly significant over multi-day time spans. Their findings reinforce a key conclusion from the broader literature: while SGP4 is sufficiently accurate for short-term prediction and catalog maintenance, its underlying modelling assumptions impose fundamental limits on long-term orbit fidelity.

In this project, the `Skyfield` Python library is used to provide a baseline trajectory representation, and its implementation relies directly on SGP4 for propagating TLE-based initial conditions. As a result, the SGP4 model serves as the analytical benchmark against which the extended high-fidelity `Basilisk` simulations are compared in later chapters.



### 2.1.2 Numerical Orbit Propagation

### 2.1.3 Semianalytical and Hybrid Propagators

### 2.1.4 Propagator Comparison and Validation

## 2.2 Orbital Perturbation Modelling

### 2.2.1 Gravity-Field Perturbations

### 2.2.2 Atmospheric Drag

### 2.2.3 Solar Radiation Pressure

### 2.2.4 Third-Body Perturbations

## 2.3 Error Growth and Propagation Stability

### 2.3.1 Long-Term Orbit Prediction Accuracy

### 2.3.2 Sensitivity to Force-Model Fidelity

### 2.3.3 Numerical Integration Error

### 2.3.4 Regime-Dependent Error Growth

## 2.4 Satellite Formation Flying and Relative Motion

### 2.4.1 Relative Motion Dynamics

### 2.4.2 Sensitivity of Relative Motion to Propagation Error

### 2.4.3 Formation-Keeping delta-V Requirements

### 2.4.4 Disturbance Effects on Formation Stability

## 2.5 Summary of Literature Review

**Table 2.5.1:** Literature overview for the *Orbital propagation* theme

Field	Literature
Analytical propagation (SGP4)	Vallado (2021) Liu <i>et al.</i> (2021) Conkey & Zielinski (2022) Acciarini <i>et al.</i> (2024)
Numerical orbit propagation	Brouwer & Clemence (1961) Bate, Mueller & White (1971) Battin (1999) Montenbruck & Gill (2000) Vallado (2025)
Semianalytical / hybrid propagators	Kaula (1966) Conkey & Zielinski (2022) Wang <i>et al.</i> (2024)
Propagator comparison & validation	Roscoe <i>et al.</i> (2014) Liu <i>et al.</i> (2021) Conkey & Zielinski (2022) Acciarini <i>et al.</i> (2024) Vallado (2025)

**Table 2.5.2:** Literature overview for the *Orbital perturbation modelling* theme

Field	Literature
Gravity-field perturbations	Kaula (1966) Battin (1999) Montenbruck & Gill (2000) Wakker (2015) Vallado (2021) Wang <i>et al.</i> (2024)
Atmospheric drag	Montenbruck & Gill (2000) Roscoe <i>et al.</i> (2014) Vallado (2021)
Solar radiation pressure (SRP)	Wertz (1978) Montenbruck & Gill (2000) Roscoe <i>et al.</i> (2014) Vallado (2021)
Third-body perturbation	Wertz (1978) Montenbruck & Gill (2000) Roscoe <i>et al.</i> (2014) Vallado (2021)

**Table 2.5.3:** Literature overview for the *Error growth & propagation stability* theme

Field	Literature
Long-term orbit prediction accuracy	Liu <i>et al.</i> (2021) Conkey & Zielinski (2022) Acciarini <i>et al.</i> (2024) Vallado (2025)
Sensitivity to force-model fidelity	Roscoe <i>et al.</i> (2014) Vallado (2021) Conkey & Zielinski (2022) Acciarini <i>et al.</i> (2024) Wang <i>et al.</i> (2024)
Numerical integration error	Bate, Mueller & White (1971) Battin (1999) Montenbruck & Gill (2000) Vallado (2021)
Regime-dependent error growth	Roscoe <i>et al.</i> (2014) Conkey & Zielinski (2022) Vallado (2025)

**Table 2.5.4:** Literature overview for the *Satellite formation flying* theme

<b>Field</b>	<b>Literature</b>
Relative motion dynamics	Montenbruck & Gill (2000) Schaub & Junkins (2003) Kristiansen <i>et al.</i> (2007)
Sensitivity of relative motion to propagation error	Kristiansen <i>et al.</i> (2007) Roscoe <i>et al.</i> (2014) Vallado (2021)
Formation-keeping $\Delta V$ requirements	Kristiansen <i>et al.</i> (2007) Roscoe <i>et al.</i> (2014)
Disturbance effects on formation stability	Kristiansen <i>et al.</i> (2007) Roscoe <i>et al.</i> (2014)

---

CHAPTER  
THREE

---

THEORY

### 3.1 Numerical Precision

### 3.2 Numerical Solvers

### 3.3 Vector notation

### 3.4 Reference Frames

#### 3.4.1 Inertial ECI frame

$\mathcal{F}^i$

#### 3.4.2 RTN frame

x-axis (R) outwards from the position vector (radial direction), z-axis (N) pointing towards the cross product between the position and velocity vectors (along-track direction), y-axis (T) completing the right-handed system (cross-track direction). Denoted by  $\mathcal{F}^r$

$$\hat{R}_r^i = \frac{r_{c/e}^i}{\|r_{c/e}^i\|}, \quad \hat{N}_r^i = \frac{r_{c/e}^i \times v_{c/e}^i}{\|r_{c/e}^i \times v_{c/e}^i\|}, \quad \hat{T}_r^i = \hat{N}_r^i \times \hat{R}_r^i \quad (3.1)$$

Here,  $r_{c/e}^i$  and  $v_{c/e}^i$  are the position and velocity vectors of the chief/leader relative to the Earth's center expressed in the inertial frame. The transformation between ECI and RTN

frames is given by:

$$R_r^i = [\hat{R}_r^i \quad \hat{T}_r^i \quad \hat{N}_r^i], \quad R_i^r = (R_r^i)^T = \begin{bmatrix} (\hat{R}_r^i)^T \\ (\hat{T}_r^i)^T \\ (\hat{N}_r^i)^T \end{bmatrix} \quad (3.2)$$

## 3.5 Orbital Mechanics

### 3.5.1 Classical Orbital Elements (COEs)

## 3.6 Orbital Disturbances

Orbital disturbances in descending order of significance.

### 3.6.1 Earth's Oblateness

### 3.6.2 Atmospheric Drag

### 3.6.3 Solar Radiation Pressure

Radiation is modeled by using solar flux at one astronomical unit and scaling by distance from the sun relative to 1 AU. The solar flux at one AU is taken as:

$$SF_{AU} = 1372.5398 \left[ \frac{W}{m^2} \right] \quad (3.3)$$

The cannonball model assumes the spacecraft is a simple sphere. The radiation pressure at 1AU,  $p_{SR}$ , can be taken as the solar flux divided by the speed of light.

$$p_{SR} = \frac{SF_{AU}}{c} \left[ \frac{N}{m^2} \right] \quad (3.4)$$

Then, a “scaling factor” can be determined. This “scaling factor” is equivalent to the magnitude of the solar radiation force divided by the distance between the spacecraft and the sun:

$$\frac{|\mathbf{F}_{\text{radiation}}|}{|\mathbf{r}_{\text{sun}}|} = -c_R p_{SR} A_d \frac{AU^2}{|\mathbf{r}_{\text{sun}}|^3} \left[ \frac{N}{m} \right] \quad (3.5)$$

$\mathbf{r}_{\text{sun}}$  is the vector from the spacecraft to the sun in the spacecraft body frame and  $cR$  is the reflectivity. This factor is then multiplied by the position vector from the spacecraft to the sun to get the force on the spacecraft due to solar radiation pressure.

$$\mathbf{F}_{\text{radiation}} = \frac{|\mathbf{F}_{\text{radiation}}|}{|\mathbf{r}_{\text{sun}}|} \mathbf{r}_{\text{sun}} [N] \quad (3.6)$$

### 3.7 Two-Line Elements (TLEs)

### 3.8 SGP4 Propagator

---

CHAPTER  
FOUR

---

METHODS

## 4.1 Simulation Design

## 4.2 Literature Search Methodology

## 4.3 Software Development

### 4.3.1 Software Architecture

This section is just for me to organize my thoughts while I work on developing the software architecture.

#### 4.3.1.1 Main Pipeline

Sequential steps executed by the main program.

1. Initialize the simulation environment with global satellite instances and simulation parameters from config.
2. Call a function to run the SGP4 propagator from "GNSS\_R\_System\_Simulator".
  - (a) Its initial conditions are the TLEs and the initial epoch, fetched from the satellite instances (from config).
  - (b) This function call also extracts the states ( $\vec{r}$  and  $\vec{v}$ ) at the initial epoch in the SGP4 output frame (TEME, I believe).

- (c) The function call then transforms the states to the Basilisk inertial frame ( $\text{TEME}(?) \rightarrow \text{ECI}(?)$ ) and adds it to the satellite instances.
  - (d) Log the output trajectories in an external file or standardized data structure for later use.
3. Call a function to run the dynamical Basilisk simulation.
- (a) Its initial conditions are the states from the SGP4 output at the initial epoch, fetched from the satellite instances.
  - (b) Log the output trajectories in an external file or standardized data structure for later use.

#### 4.3.1.2 Config

This subsection describes all the fields included in the config file

- SIMULATION (This will contain all the high-level simulation parameters that are global for all simulation frameworks)
  - `startTime` (Real-world date and time of the simulation start. Is this different from "initial epoch"?)
  - `simulationDuration` (How long period into the future the simulation will simulate)
- SATELLITE (This will contain all the satellite-specific parameters, and there will be one such section for each satellite in the simulation)
  - `satelliteName` (A string identifier for the satellite)
  - `TLELine1` (The first line of the TLE data for the satellite)
  - `TLELine2` (The second line of the TLE data for the satellite)
  - `initialEpoch` (The epoch time corresponding to the TLE data, in a standardized format. Should always be the same as `startTime`???)
  - physical attributes like size, mass, drag coefficient, inertia, etc. (TBD!)
- PLOTTING (This will contain all the parameters related to how the simulation results should be visualized)
  - `savePath` (Path to save the generated plots)
  - Other plotting parameters (TBD!)
- SKYFIELD (This will contain all the parameters specific to the Skyfield simulation)
  - `timeStep` (The time step for the Skyfield simulation)
  - Skyfield-specific parameters (TBD!)
- BASILISK (This will contain all the parameters specific to the Basilisk simulation framework)

- `timeStep` (The time step for the Basilisk simulation)
- `solver` (The numerical solver to be used in the Basilisk simulation)
- `gravityModel` (The gravity model to be used in the Basilisk simulation)
- `radiationPressureModel` (The radiation pressure model to be used in the Basilisk simulation)
- Other Basilisk-specific parameters (TBD!)

The below list shows additional capabilities in Basilisk that have yet to be implemented, but could serve to increase the fidelity of the simulation.

- Track a point relative to the satellite's COM

---

CHAPTER  
FIVE

---

RESULTS

## 5.1 Prelim results

The current simulator implementation has been used to generate preliminary results for a two-satellite scenario consisting of one chief and one follower spacecraft initialized on different orbital planes in LEO. Skyfield’s SGP4 model serves as the baseline, while the Basilisk simulation includes gravity-gradient effects, solar radiation pressure, and lunar third-body pull. Atmospheric drag has not yet been implemented, and both propagators are run over the same integration time for direct comparison.

Figure 1 and 2 show the ECI position and velocity differences between Basilisk and SGP4 for the chief and follower spacecraft, respectively. Figure 3 presents the difference in relative motion between the two spacecrafts expressed in the RTN frame. In all cases, the divergence begins small but increases steadily over time. This divergence can be attributed to a combination of intrinsic numerical simulator difference and additional physical perturbations included only in the Basilisk simulation. The large discrepancy visible in the relative-motion plot indicates that a formation-keeping delta-V analysis performed with the two simulation frameworks would yield significantly different results after 24 hours. Further testing and analysis are required to determine how much of this divergence is caused by numerical effects versus physical modelling differences.

Below is a list over all types of plots that I deem valuable for the specialization project. Each plot-type shall be justified by a short description of its usefulness.

### 5.1.1 Simulator output difference plots

Show the absolute value logarithmic difference between the simulators. This will make it easier to see the small differences for times smaller than 6 hours, and the big differences

after 24 hours. Do this for both the absolute pos/vel difference plot, and the relative formation pos/vel plots.

### 5.1.2 R

---

CHAPTER  
**SIX**

---

DISCUSSION

Discuss your results here.

## 6.1 Future work

Include a section about what should or could be done in future research, or explain any recommended next steps based on the results you got. This should be the last section in the discussion.

How do zxc here's to all the idk This situation is starting to become pretty dire

---

CHAPTER  
**SEVEN**

---

CONCLUSIONS

Give a concise summary of your research and finding here, and include a short summary of any future work as well.

## REFERENCES

- [1] NASA Earthdata. *GNSS - Global Navigation Satellite System*. Accessed: 2025-11-23. 2024. URL: <https://www.earthdata.nasa.gov/data/space-geodesy-techniques/gnss>.
- [2] Harry W. Jones. “The Recent Large Reduction in Space Launch Cost”. In: *48th International Conference on Environmental Systems (ICES)*. ICES-2018-81. Albuquerque, New Mexico, July 2018, pp. 1–10. URL: <https://ntrs.nasa.gov/citations/20200001093>.
- [3] *Hva er jamming, spoofing og meaconing?* FFI Factsheet. Accessed: 2025-11-21. Norwegian Defence Research Establishment (FFI), 2024.
- [4] Norwegian Defence Research Establishment (FFI). *Jamming - a technological cat-and-mouse game*. <https://www.ffi.no/en/news/jamming--a-technological-cat-and-mouse-game>. Accessed: 2025-02-20. 2024.
- [5] GPSJam Project. *GPSJam - Real-time GNSS Interference Map*. <https://gpsjam.org>. Accessed: 2025-11-21. 2025.
- [6] Shuanggen Jin and Attila Komjathy. “GNSS Reflectometry and Remote Sensing: New Objectives and Results”. In: *Advances in Space Research* 46.1 (2010), pp. 111–117. DOI: [10.1016/j.asr.2010.01.014](https://doi.org/10.1016/j.asr.2010.01.014). URL: [https://www.sciencedirect.com/science/article/pii/S0273117710000499?casa\\_token=0z4kQjAlYwsAAAAA:STxQm3Wyqd2XvnpbveQhRXNvqU0fPIugKy2wwwYIkM65IhS2z184e07fH4z2QJhcTMW1f3IT](https://www.sciencedirect.com/science/article/pii/S0273117710000499?casa_token=0z4kQjAlYwsAAAAA:STxQm3Wyqd2XvnpbveQhRXNvqU0fPIugKy2wwwYIkM65IhS2z184e07fH4z2QJhcTMW1f3IT).
- [7] C. Hall and R. Cordey. “Multistatic Scatterometry”. In: *Proceedings of the IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*. IEEE. Edinburgh, Scotland, 1988, pp. 561–562.
- [8] NTNU SmallSat Lab. *CubeSat Concept Using GNSS-R*. <https://www.ntnu.edu/smallsat/gnss-r>. Accessed: 2025-11-21. 2025.
- [9] Victor Szebehely. “The History and Background of Astrodynamics”. In: *Acta Astronautica* 20 (1989), pp. 79–81.
- [10] Isaac Newton. *Philosophiae Naturalis Principia Mathematica*. Londini: Jussu Societatis Regiae ac Typis Josephi Streeter, 1687.
- [11] David A. Vallado. *Fundamentals of Astrodynamics and Applications*. Springer, 2001. URL: <https://books.google.no/books?id=PJL1WzMbkjkC>.

- [12] Zac Yung-Chun Liu et al. “Improved Orbital Propagator Integrated with SGP4 and Machine Learning”. In: *AIAA/USU Small Satellite Conference*. 2021. URL: <https://digitalcommons.usu.edu/smallsat/2021/all2021/193>.
- [13] Dave Conkey and Mitchell Zielinski. “Assessing Performance Characteristics of the SGP4-XP Propagation Algorithm”. In: *Advanced Maui Optical and Space Surveillance Technologies Conference (AMOS)*. 2022. URL: <https://amostech.com/TechnicalPapers/2022/Poster/Conkey.pdf>.
- [14] Giacomo Acciarini, Atilim Günes Baydin, and Dario Izzo. “Closing the Gap Between SGP4 and High-Precision Propagation via Differentiable Programming”. In: *Acta Astronautica* (2024). URL: <https://www.sciencedirect.com/science/article/pii/S0094576524006374>.

## APPENDICES

## A - GITHUB REPOSITORY

All code and latex-files used in this document are included in the Github repository linked below. Further explanations are given in the readme-file.

### **Specialization project report LaTeX repository link**

- <https://github.com/chystad/Specialization-Project-Report-LaTeX->

### **Orbit Simulator repository link**

- [https://github.com/chystad/Orbit\\_Simulator](https://github.com/chystad/Orbit_Simulator)