

# **Orbit Determination for Low-Altitude Satellites Using Semianalytical Satellite Theory**

**Rodrigo Moreira de Oliveira**

Thesis to obtain the Master of Science Degree in

## **Aerospace Engineering**

Supervisor: Prof. João Pedro Castilho Pereira Santos Gomes

### **Examination Committee**

Chairperson: Prof. Paulo Jorge Coelho Ramalho Oliveira

Supervisor: Prof. João Pedro Castilho Pereira Santos Gomes

Member of the Committee: Dr. Todd Allan Ely

**January 2021**



To my mother and father.



## **Acknowledgments**

I would like to express my gratitude to my thesis supervisor, Professor João Pedro Gomes, for the assistance, guidance and motivation throughout the entire project. Without his key advice and valuable inputs, this work would not have been possible.

I would also like to thank all my friends for having been beside me throughout my life, in both the good and bad moments, and without whom I would have never had the motivation to arrive this far. You have certainly made my academic journey much easier and enjoyable.

To my family, a very special thank you for all the love and care. To my mother, father and brother thank you for constant support, care and dedication in every step of the way that led to this moment.

Last but not least, I would also like to acknowledge Instituto Superior Técnico for providing me, over long hours of work and dedication, the necessary tools that have allowed me to pursue my studies, open my horizons and grow as an individual.



## Resumo

O seguimento de satélites fornecido por sistemas de Determinação de Órbita requer a monitorização frequente e precisa de trajetórias orbitais, que é fundamental para o sucesso de missões espaciais. Esta tese aborda o problema de estimação sequencial em tempo-real de um satélite em Órbita Terrestre Baixa (com aplicações em detecção remota, comunicações ou observação terrestre) seguido por estações terrestres. Apesar do uso de sensores GNSS ter sido favorecido nos últimos anos para este tipo de missões, devido ao seu baixo custo e capacidade comprovada de seguimento, estes sistemas nem sempre são confiáveis. Assim, é necessário prever sistemas auxiliares que permitam salvaguardar a estimação da posição do satélite em caso de falha do sistema GNSS. Esta tese combina a Teoria Semianalítica de Satélites com técnicas de estimação de Filtros de Kalman para desenvolver dois algoritmos sequenciais: o Filtro Semianalítico de Kalman *Extended* e o Filtro Semianalítico de Kalman *Unscented*. Este último é um algoritmo novo proposto neste trabalho, que combina a teoria semi-analítica com um Filtro de Kalman *Unscented*. São discutidas implicações da interação entre estes filtros e a teoria semianalítica de propagação orbital. Estes algoritmos são ainda avaliados em termos de eficiência, precisão e rapidez de convergência por comparação com Filtros de Kalman *Extended* e *Unscented* associados a propagadores orbitais do tipo *Cowell*, algoritmos mais bem consolidados na literatura. Após serem efetuadas experiências em ambiente simulado, conclui-se que os filtros Semianalíticos podem competir com filtros *Cowell*, em termos de precisão e eficiência, na determinação e predição de efemérides de satélites.

**Palavras-chave:** Determinação de Órbita, Filtros de Kalman, Órbita Terrestre Baixa, Propagação Semianalítica, Propagação *Cowell*, Estimação Eficiente





## Abstract

The tracking of spacecraft provided by Orbit Determination systems requires frequent and accurate monitoring of their orbital trajectories, which, in many cases, is fundamental to the success and deliverance of space missions. This thesis focuses on the problem of sequential real-time estimation of a Low Earth Orbit satellite (with applications in remote sensing, communications, Earth observation, etc.) tracked by ground stations. Although the use of GNSS sensors has been favored in recent years for this kind of mission scenario due to their low-cost, availability and proven capability of satellite tracking, these systems might not always be reliable. In view of this, it is necessary to provide auxiliary systems that safeguard Orbit Determination procedures in case of GNSS failure. This thesis combines the Semianalytical Satellite Theory with current Kalman filtering techniques to develop and study two sequential algorithms: the Extended Semianalytical Kalman Filter and the Unscented Semianalytical Kalman Filter. The latter is a novel algorithm proposed in this research, combining semianalytical propagation with Unscented Kalman Filters. The design implications of the interaction between these filters and the orbital propagation/perturbation theory are discussed. These algorithms are then evaluated in terms of efficiency, accuracy and speed of convergence by comparison with Special Perturbation Cowell Extended and Unscented Kalman Filters, which are more established algorithms in the literature. By conducting some simulation test cases, it is found that Semianalytical filters can compete with Cowell filters in accurately and efficiently determining and predicting satellite ephemerides.

**Keywords:** Orbit Determination, Kalman Filters, Low Earth Orbit, Semianalytical Propagation, Cowell Propagation, Efficient Filtering



# Contents

Acknowledgments . . . . .	v
Resumo . . . . .	vii
Abstract . . . . .	ix
List of Tables . . . . .	xiii
List of Figures . . . . .	xv
Nomenclature . . . . .	xvii
Glossary . . . . .	xix
<b>1 Introduction</b>	<b>1</b>
1.1 Motivation . . . . .	1
1.2 Objectives and Deliverables . . . . .	3
1.3 Thesis Outline . . . . .	4
<b>2 Background</b>	<b>5</b>
2.1 Satellite Navigation . . . . .	5
2.2 Reference Frames . . . . .	6
2.2.1 Earth Centered Reference Systems . . . . .	6
2.2.2 Satellite Coordinate System . . . . .	8
2.2.3 Topocentric System . . . . .	8
2.2.4 Transformations . . . . .	9
2.3 Orbital State-Space Models . . . . .	11
2.3.1 State Vector . . . . .	11
2.3.2 Keplerian Orbital Element Set . . . . .	11
2.3.3 Equinoctial Orbital Element Set . . . . .	12
2.4 Orbital Mechanics . . . . .	12
2.4.1 Kepler's Laws and Two-Body Problem . . . . .	12
2.4.2 Perturbed Motion . . . . .	14
2.5 Nonlinear Filtering . . . . .	15
2.5.1 Nonlinear Transformation of Random Variables . . . . .	16
2.5.2 Kalman Filter . . . . .	19
2.6 Signals in the Frequency Domain . . . . .	22

2.6.1	Fourier Series . . . . .	22
2.6.2	Discrete Fourier Transform . . . . .	24
2.6.3	Relation Between Discrete Fourier Transform and Fourier Series . . . . .	24
2.7	Performance Evaluation . . . . .	26
2.7.1	Accuracy . . . . .	26
2.7.2	Computational Cost . . . . .	26
<b>3</b>	<b>Modeling and Implementation</b>	<b>27</b>
3.1	Dynamics Modeling . . . . .	27
3.1.1	Osculating Dynamics . . . . .	28
3.1.2	Mean Dynamics . . . . .	31
3.2	Measurement Modeling . . . . .	40
3.2.1	Measurement Noise . . . . .	41
3.2.2	Station Availability . . . . .	42
3.3	Model Linearization . . . . .	42
3.3.1	Osculating Dynamics Jacobian . . . . .	42
3.3.2	Mean Dynamics Jacobian . . . . .	44
3.3.3	Measurement Model Jacobian . . . . .	44
3.4	Forming Orbit Determination Filters . . . . .	45
3.4.1	Cowell Filters . . . . .	46
3.4.2	Semianalytical Filters . . . . .	46
3.4.3	Filter Initialization . . . . .	51
<b>4</b>	<b>Simulation Results</b>	<b>53</b>
4.1	Test Case 1 - Single Ground Station . . . . .	53
4.1.1	Propagation Results . . . . .	55
4.1.2	Filter Results . . . . .	62
4.2	Test Case 2 - Multiple Ground Stations . . . . .	70
<b>5</b>	<b>Conclusions</b>	<b>73</b>
5.1	Contributions . . . . .	75
5.2	Future Work . . . . .	75
	<b>Bibliography</b>	<b>77</b>
<b>A</b>	<b>Orbit Determination Filters</b>	<b>82</b>
A.1	Differential Correction Batch Least Squares . . . . .	82
A.2	Extended Semianalytical Kalman Filter . . . . .	84
A.3	Filter Flowcharts . . . . .	86
<b>B</b>	<b>Linearization of Dynamical Systems and Discretization of Process Noise</b>	<b>89</b>

<b>C Numerical Implementation of Earth's Gravitational Acceleration</b>	<b>91</b>
C.1 Recursive Computation of the ALFs . . . . .	91
C.2 Recursive Computation of the Potential Partial Derivatives . . . . .	92
<b>D Orbital State-Space Transformations</b>	<b>93</b>
D.1 Conversion Between State Vector and Equinoctial Elements . . . . .	93
D.1.1 From State Vector to Equinoctial Elements . . . . .	94
D.1.2 From Equinoctial Elements to State Vector . . . . .	95
D.2 Partial Derivatives of State Vector With Respect to the Equinoctial Elements . . . . .	96
D.3 Partial Derivatives of Equinoctial Elements With Respect to Velocity . . . . .	97
<b>E Equations of Averaging</b>	<b>99</b>



# List of Tables

2.1	Earth's physical constants. . . . .	15
4.1	Initial epoch and orbital state for Test Case 1. . . . .	53
4.2	Sensor measurement noise standard deviations for Test Case 1. . . . .	54
4.3	GMAT configuration for reference orbit generation. . . . .	54
4.4	Physical properties of the satellite. . . . .	55
4.5	Cowell propagation gravitational field study. . . . .	57
4.6	$\overline{\text{RMSE}}$ of mean element propagation compared with DSST. . . . .	60
4.7	CPU computation time for Semianalytical propagation with different output requested steps. . . . .	61
4.8	Trajectory averaged position and velocity $\overline{\text{RMSE}}$ for the same simulation run illustrated in Figure 4.8. . . . .	64
4.9	Simulation environments for the initial error analysis. . . . .	65
4.10	Measurement models for the observability analysis. . . . .	67
4.11	Location of the ground stations and their associated measurement noise standard deviations for Test Case 2. . . . .	71
4.12	Initial epoch and orbital state for Test Case 2. . . . .	71





# List of Figures

2.1	Representation of the ECEF frame. . . . .	7
2.2	Satellite Coordinate System. . . . .	8
2.3	Geometry for two bodies in inertial frame. . . . .	13
2.4	Geometries of the Keplerian orbit. . . . .	14
2.5	Major perturbing accelerations of Earth satellites. . . . .	15
2.6	Predict-update sequence of Kalman filters. . . . .	20
3.1	Secular and periodic perturbation on orbital elements. . . . .	28
3.2	Earth's gravitational field spherical harmonics. . . . .	30
3.3	Atmospheric Exponential Model parameters as a function of the satellite's altitude $h_{ellp} = \ r\  - R_{\oplus}$ . Source [3]. . . . .	32
3.4	Illustration of the Semianalytical propagator. . . . .	40
3.5	3-D trajectory plot of Semianalytical filtering . . . . .	52
4.1	GMAT reference trajectory for Test Case 1. . . . .	55
4.2	Averaged position and velocity $\overline{\text{RMSE}}$ as a function of the integration step size, for a Keplerian Orbit propagation. . . . .	56
4.3	Full force model study of Cowell propagation. . . . .	58
4.4	Tuning DFT lengths of the Semianalytical propagator. . . . .	59
4.5	Position and velocity RMSE comparison of the developed Semianalytical propagator and DSST. . . . .	61
4.6	CPU computation time comparison between the developed Cowell and Semianalytical propagators for propagation arcs from 1 day to 7 days. . . . .	62
4.7	Study of the filter initialization procedure. . . . .	63
4.8	Time evolution of the position RMSE of the filtered trajectories outputted by EKF, UKF, ESKF and USKF relative to the reference trajectory, for a single MC run. . . . .	64
4.9	Position RMSE considering orbital estimates given directly by the inverse of the measurement function. . . . .	65
4.10	Time evolution of the position RMSE for EKF, UKF, ESKF and USKF, considering different initial error environments. . . . .	66

4.11 Evolution of the condition number of the observability Gramian in one passage versus the measurement arc length. . . . .	68
4.12 Steady-state averaged $\overline{\text{RMSE}}$ after one ground station pass considering three different measurement models. . . . .	69
4.13 Results of Test Case 2 . . . . .	72
A.1 Flowchart representation of Cowell Orbit Determination filters. . . . .	87
A.2 Flowchart representation of Semianalytical Orbit Determination filters. . . . .	88
D.1 Equinoctial system. . . . .	97

# Nomenclature

## Greek symbols

- $\beta, \delta$  Azimuth and elevation angle measurements, respectively.
- $\Delta \bar{\mathcal{E}}_k^l$  Semianalytical filter correction at time  $t_k$  given observations until time  $t_l$ .
- $\delta_{ij}$  *Kronecker delta*
- $\eta_i^j$  Short-periodic function of order  $j$  on element  $i$ .
- $\omega, \boldsymbol{\omega}$  Planet's rotational speed and velocity vector, respectively.
- $\Phi$  State Transition Matrix.
- $\sigma$  Standard deviation.
- $\theta$  Sidereal angle.

## Roman symbols

- $f$  Osculating orbital state function.
- $h_d$  Orbital measurement function.
- $r, v, a$  Position, velocity and acceleration vectors.
- $y$  Observation vector.
- $\bar{f}$  Mean orbital state function.
- $\mathcal{E}$  Equinoctial element set.
- $\mathcal{F}$  Fourier Series coefficient.
- $\mathbf{x}$  Orbital state vector.
- $A_i^j$  Mean element rate of change function of order  $j$  on element  $i$ .
- $j$  Imaginary unit.
- $J_2$  Second degree zonal harmonic coefficient.
- $s, \dot{s}$  Range and range-rate measurements, respectively.

$t$  Time variable.

$v_x, v_y, v_z$  Cartesian velocity components.

$X$  Discrete Fourier Transform coefficient.

$x, y, z$  Cartesian position components.

### Subscripts

$\oplus$  Earth parameter.

$d$  Discretized function.

$gc, gd$  geocentric and geodetic, respectively.

$i, f, s, w$  indexes of ECI, ECEF, ENZ and RSW frames, respectively.

$k$  Time sample index.

$Z, T, D$  Zonal, tesseral/sectoral and drag perturbations, respectively.

### Superscripts

$+$  *A posteriori.*

$-$  *A priori.*

T Transpose.

# Glossary

<b>ALFs</b>	Associated Legendre Functions.
<b>DFT</b>	Discrete Fourier Transform.
<b>DSST</b>	Draper Semianalytic Satellite Theory.
<b>ECEF</b>	Earth Centered Earth Fixed.
<b>ECI</b>	Earth Centered Inertial.
<b>EKF</b>	Extended Kalman Filter.
<b>ENZ</b>	East North Zenith.
<b>ESKF</b>	Extended Semianalytical Kalman Filter.
<b>EoM</b>	Equations of Motion.
<b>FFT</b>	Fast Fourier Transform.
<b>GNSS</b>	Global Navigation Satellite System
<b>GP</b>	General Perturbation orbital propagation.
<b>GRV</b>	Gaussian Random Variable.
<b>GS</b>	Ground Station.
<b>KF</b>	Kalman Filter.
<b>LEO</b>	Low Earth Orbit.
<b>MC</b>	Monte Carlo
<b>OD</b>	Orbit Determination
<b>PDEs</b>	Partial Differential Equations.
<b>RMSE</b>	Root Mean Square Error.
<b>SLR</b>	Satellite Laser Ranging.
<b>SP</b>	Special Perturbation orbital propagation.
<b>SRP</b>	Solar Radiation Pressure
<b>SST</b>	Semianalytic Satellite Theory.
<b>STM</b>	State Transition Matrix.
<b>SciPy</b>	Scientific Python package.
<b>TOD</b>	True of Date.
<b>UKF</b>	Unscented Kalman Filter.
<b>USKF</b>	Unscented Semianalytical Kalman Filter.
<b>VOP</b>	Variation-of-Parameters.

**WSL**    Weighted Statistical Linearization.

# Chapter 1

## Introduction

### 1.1 Motivation

Satellite Orbit Determination (OD) refers to the estimation of orbits (in terms of position and velocity, or other equivalent state spaces) of space objects (such as interplanetary spacecraft or Earth orbiting satellites), given applicable measurements [1, 2]. To do so, the object's motion is dynamically modeled using a set of differential equations, which are only approximate since they only consider the most relevant contributions to the motion.

OD has had a long and outstanding history. Its roots lie in astronomy and prediction of motion of planets and comets. Copernicus (1473-1543), Kepler (1571-1630), Newton (1642-1727), Lagrange (1736-1813) and Gauss (1777-1855) were some of the early pioneers of this science and their remarkable work still serves as the fundamental basis of modern methods and techniques [3]. From the launch in 1957 of Sputnik, which was tracked mostly by visual observations with a precision to only a few kilometers, the technological revolution of the last decades in terms of computational power and tracking sensors, has enabled OD solutions to reach the sub-centimeter accuracy [2].

A broad number of satellite space missions and applications require orbit information provided by OD, e.g., Earth observation, remote sensing, geodesy, telecommunications and weather missions [4]. Furthermore, in many cases the success of the mission relies on the satellite following a certain reference nominal orbit. Corrective maneuvers are therefore employed when the satellite deviates from that reference. This means that Orbit Control also relies on OD predictions [5, 6].

Orbit Determination is a very wide and general field. Different classes of problems exist within it, namely [1]: 1) Initial Orbit Determination (IOD) methods take a limited set of observations and usually operate on analytical and simplified orbital dynamics; 2) Batch Least-Squares Differential Correction methods take all the tracking data over an observation span and determine the best estimate of the satellite's state at a reference time, and are usually suitable for post-processing applications; and 3) Sequential methods are forward-time recursive sequential machines that repeat patterns of state prediction (orbit propagation) and measurement updates, and are typically associated with real-time applications. Furthermore, Precise Orbit Determination [4] is a type of OD that refers to the estimation of a satellite's

state with the highest possible accuracy, usually resorting to post-processing batch algorithms, implementing high-fidelity force models and relying on high-precision measurements from Global Navigation Satellite Systems (GNSS) [7] or Satellite Laser Ranging (SLR) [8]. POD research is very active [9, 10] and POD has been successfully implemented in many satellite missions, e.g. Sentinel-1A which has a post-processing position estimation error requirement of only 5 centimeters [11].

Space around the Earth is getting increasingly crowded, posing computational problems to accurately predict the positions of a large number of space objects [12]. This is very important, for instance, to safeguard payloads and astronauts when predicting close approaches. Morton and Roberts [13] estimate that the US Joint Space Operation Center performs about 40 000 track and object correlations per day, whereas the authors of [14] state that the Russian Space Surveillance Center performs about ten million orbit propagations every day to maintain their catalogs and provide collision warnings. In view of this, orbit propagation schemes must be both computationally efficient and accurate. Broadly speaking, these schemes may be classified as [3]: 1) Special Perturbation (SP) methods numerically and accurately propagate the Equations of Motion (EoM); 2) General Perturbation (GP) methods provide analytical approximations to the EoM; and 3) Semianalytical methods average the EoM, explicitly separating the short-term periodic motion from the long-term periodic and secular motion, which can then be numerically propagated with much larger integration step sizes when compared to SP. SP methods are the most accurate, but require the most computational resources, whereas GP methods are light and efficient but much more inaccurate [12]. Semianalytical methods provide a mix between SP and GP, having accuracies comparable with SP methods while retaining efficiencies closer to GP methods [12].

In view of the discussion above, Semianalytical propagations schemes within OD procedures are the most favorable schemes when it comes to keeping track of a large number of objects. Moreover, satellite guidance and orbit control benefit from the use of mean orbital elements (associated to both GP and Semianalytical methods), rather than osculating (non-averaged) elements (associated to SP methods) [5]. It makes more sense to have satellites track orbits defined by mean slowly-varying elements that capture the long-term evolution of the orbits, rather than having them respond to fast oscillations (with zero-mean time average) that do not change the shape and orientation of the orbit over time. Indeed, feeding mean elements to the orbit controller may reduce fuel consumption [5]. Moreover, satellite lifetime studies are more easily and straightforwardly conducted in the mean element state space [3], since these studies are only interested in the long-term behavior of the orbit. Furthermore, OD with Semianalytical methods is expected to be more stable, reliable and accurate than with SP schemes, since the linearization assumptions of nonlinear filtering algorithms are better satisfied with secular and long-periodic dynamics rather than with osculating dynamics [15].

This thesis focuses on the problem of sequential real-time OD of a Low Earth Orbit (LEO) satellite (to be used for instance in remote sensing applications) tracked by ground stations implementing both SP and Semianalytical propagation schemes. Although the use of GNSS is becoming more popular in this kind of mission scenario due to their low-cost, availability and proven ability to track LEO satellites [16], these systems might not always be reliable, for instance, they are susceptible to spoofing attacks [17]. In view of this, it may be necessary to provide auxiliary systems that safeguard OD procedures in



case of GNSS failure.

The development of a complete OD system can be broken into two phases: first, an orbit propagation scheme capable of handling state predictions (in view of some predefined accuracy and efficiency requirements) needs to be developed. Then, the propagator is coupled with estimation filtering algorithms, in view of some measurement model. While this coupling is straightforward for SP methods, Semianalytical schemes require additional operations to keep their aforementioned efficient implementation, namely, their large propagation step size.

There exist many alternative algorithms for sequential OD, which is a nonlinear filtering problem. This thesis focuses on sub-optimal solutions provided by Kalman filters. A comparison of many filtering schemes, namely, Extended and Unscented Kalman filters, particle filters, Gaussian mixtures or linear minimum mean-square error filters in the context of OD has been made in [18, 19].

## 1.2 Objectives and Deliverables

The objective and purpose of this work is to study and develop OD solutions that are able to estimate LEO satellites in the absence of GNSS sensors. The goal is to develop algorithms that are not only efficient for real-time implementation, but also reliable in terms of competitive accuracy requirements. In view of this, SP and Semianalytical propagation schemes are developed, based on reduced force models that capture the most important perturbations from the Earth's Gravitational Field and atmospheric drag. Then, these dynamics are coupled with established nonlinear and sub-optimal filters, namely, the Extended Kalman Filter (EKF) and the Unscented Kalman Filter (UKF).

While SP and GP propagation schemes have been extensively studied in the context of OD in reference textbooks [3, 20, 21] and recent research papers (SP in [18, 19, 22] and GP in [23, 24]), only a handful of research studies have been concerned with Semianalytical OD [15, 25, 26]. Andrew Green in [15] and Stephen Taylor in [25] proposed, implemented and tested (with both simulated and real data) the coupling of an EKF filter with Draper Semianalytical Satellite Theory (DSST) [27], which is one of the most popular and well documented Semianalytic Satellite Theories (SSTs). Besides reproducing this EKF, this thesis also aims at proposing a novel algorithm coupling an UKF with Semianalytical propagation. Furthermore, the present work also aims at delivering and testing a SST that can compete with DSST within OD problems, namely, a recent theory proposed and developed by Todd Ely in [28, 29]. When compared to DSST, Ely's theory is more flexible, easier to implement in computational applications and covers a wider range of problem domains, such as eccentric orbits.

To achieve the main goals proposed and mentioned above, the research has the following operational objectives:

- Study, implement and validate SP and Semianalytical propagation schemes. Validation of these schemes involves comparison with established and publicly available orbit propagation software;
- Perform accuracy and cost comparative studies between both propagation schemes, aiming at demonstrating and outlining their main characteristics and features;

- Understand the trade-off between propagation accuracy and computational cost with force model and propagation scheme selection;
- Study how different filtering algorithms, orbital state dynamical formulations, initial conditions and measurement types influence the overall accuracy of Orbit Determination.

If the outlined operational objectives are achieved, this thesis will provide, to the best knowledge of the author, the following original contributions to the literature:

- Prove and demonstrate the feasibility of Ely's SST in the context of Orbit Determination, departing from DSST-based solutions, which have already been studied before;
- Couple SST with an UKF, leading to the USKF. Comparing to the ESKF, the USKF is expected to be more robust to initialization errors and provide more accurate orbital state estimates.

## 1.3 Thesis Outline

In the present Chapter 1, a brief introduction to the work is provided. After presenting motivating reasons to study Orbit Determination, a brief definition of the proposed problems, objectives and contributions of this thesis is given.

Chapter 2 covers the background theory to be used throughout the work. A summary of the fundamentals of orbital mechanics, reference frames and orbital state-space representation is presented, followed by a brief review of nonlinear filtering theory and adopted filtering algorithms. Representation of signals in frequency domain is also overviewed. The chapter ends with a presentation of metrics for filter comparison.

Relying on previous concepts, Chapter 3 discusses implementation issues. It provides thorough and detailed discussion and development of Special Perturbation and Semianalytical orbital state dynamical models as well as measurement model schemes. Moreover, this chapter also discusses filter implementation of orbital systems associated to both SP and Semianalytical propagation schemes.

Chapter 4 presents the simulation studies conducted in this thesis, along with a brief discussion of the achieved results. The discussion starts with a performance evaluation of the orbital propagation schemes developed, in terms of accuracy and computational cost. Afterwards the implemented Cowell and Semianalytical filters are compared. The chapter also studies the influence that initialization errors and different measurement configurations have on the overall filtering performance.

Finally, Chapter 5 presents a summary of the conclusions drawn from the studies throughout this thesis, and identifies possible areas for further study and development of this work.

# Chapter 2

## Background

### 2.1 Satellite Navigation

The operation of a spacecraft in orbit is heavily dependent on the ability to locate and communicate with it. Also, satellite missions require active control of their orbit in accordance with specific mission requirements. Navigation is therefore an essential part of spacecraft operations, comprising the planning, determination, prediction and correction of a satellite's trajectory and orientation [20].

It is therefore essential to have access to measurements from which the satellite's position, velocity or attitude can be inferred. There exist different classes of satellite navigation sensors, depending on their nature. Inertial sensors (accelerometers and gyroscopes) measure motion parameters with respect to the inertial space [30]. Attitude sensors observe the spacecraft orientation with respect to a frame of reference characterized by visible references, namely, the Sun or other stars [31]. Orbit determination has even been studied with spaceborne magnetometers, which provide magnetic field measurements or gradiometers, which provide gravity gradients [32]. However, the most common way to perform orbit determination is through navigation tracking systems that measure spacecraft motion relative to known spatial references, such as ground stations or other satellites [20], which straightforwardly allow the satellite's position and velocity to be inferred. Fusing these sensors into multi-sensor solutions has also been studied in the past [33].

This work focuses on the determination/prediction phase of a satellite's trajectory (orientation determination is not analyzed), referring to the latter class of sensors described in the paragraph above, i.e., tracking systems. Typically these systems measure properties of electromagnetic wave propagation between a transmitter and a receiver [20]. The most common types of tracking systems used for spacecraft navigation are described next [20]:

- **Radar systems:** radar techniques have been used to observe spacecraft position and velocity since early times. Focusing on a simple configuration with one ground station and one satellite, three types of measurements are possible: 1) pseudo-range is the distance between satellite and station, computed from the round-trip travel time of a radar signal emitted from the ground station antenna to the satellite and radiated back to the station; 2) pointing angles are obtained

by measuring the direction of the maximum signal amplitude coming from the spacecraft and 3) range-rate of the spacecraft relative to the ground station can be derived from the Doppler shift of the round-trip radar signal<sup>1</sup>;

- **Laser systems:** Satellite Laser Ranging (SLR) systems provide highly accurate distance measurements by determining the turn-around flight time of laser pulses transmitted to a satellite and returned by a retro-reflector. It is a similar technique to radar tracking, but allows much better accuracies, that may achieve the order of centimeters. It is however noted that SLR does not allow autotracking of satellites and depends on the availability of *a priori* high-precision orbit knowledge for antenna pointing. Furthermore, SLR is strongly impacted by weather conditions. Therefore it is mainly used in high-precision geodetic scientific studies, namely, crustal dynamics, gravity field determination or Earth rotation parameter estimation;
- **Global Navigation Satellite Systems (GNSS):** Besides ground-based tracking systems, it is also possible to perform OD resorting to other satellite constellations, such as GNSS systems. These systems of satellites broadcast signals carrying precise information of their position and time, obtained from the on-board atomic-clock time-standard, which can then be used for autonomous absolute positioning through Multilateration techniques [4].

## 2.2 Reference Frames

Studying orbital motion involves applying Newton's laws of motion. These laws, which describe the relationship between a body and the forces acting upon it and its motion in response to those forces, are applied naturally in inertial reference frames [3]. An inertial reference frame in classical physics possesses the property that, in this frame of reference, a body with zero net force acting upon it does not accelerate, and thus is either at rest or moving with constant velocity [34].

Ground-based observations are obtained from observing sites on the surface of the Earth, which are therefore not at rest. In order to fuse orbit predictions with ground measurements, a concise definition of the different frames is required and their mutual relation has to be established [20].

### 2.2.1 Earth Centered Reference Systems

#### Earth Centered Inertial System

For near-Earth spacecraft, the most commonly adopted inertial frame is the Earth Centered Inertial (ECI) reference frame. This frame has its origin at the Earth's center of mass and therefore follows the Earth in its movement around the Sun. The acceleration that the Sun induces in the Earth is very similar (in direction and magnitude) to that induced in the satellite, since the Earth and the satellite are very close. Therefore, regarding the satellite's translation relative to the Earth, the acceleration applied by

---

<sup>1</sup>Few satellites are equipped with high precision time and frequency standards to perform one-way range and range-rate measurements, hence two-way ground-based measurements are considered instead.

the Sun on the two bodies can be discounted<sup>2</sup>. The result is then a geocentric inertial reference frame [35]. Its principal axes point in fixed directions with respect to the celestial sphere. The fundamental plane (plane  $xy$ ) is the Earth's equatorial plane. The  $x$  axis points towards the vernal equinox  $\Upsilon$ , the  $z$  axis points to the North Pole of Earth and the  $y$  axis is chosen so as to form a right-handed coordinate system.

In reality, neither the ecliptic (Earth's orbital plane) nor the Earth's equatorial plane are fixed with respect to the celestial sphere. Among others, the interaction of the Sun and the Moon with the Earth's equatorial bulge is responsible for small gyroscopic motion that induces oscillations on the equatorial plane [20]. In view of this, the time-dependent True of Date (TOD) realization of the ECI system is characterized by the instantaneous and up to date orientation of the ecliptic and equatorial planes [3]. By referring to a particular epoch and specifying how to transform vectors to and from that specified epoch, a more inertial frame may be defined. For that matter, the J2000 reference frame is defined according to the TOD frame orientation at 12:00 (UTC) on January 1st, 2000.

### Earth Centered Earth Fixed System

The Earth Centered Earth Fixed (ECEF) is centered at the Earth and rotates with it. The  $xy$  plane is also the equatorial plane, but now the  $x$  axis is directed towards the Greenwich meridian, with longitude  $\lambda = 0^\circ$ . The  $z$  axis coincides with the rotation axis of the Earth and points towards the North Pole of the Earth. The  $y$  axis is chosen so as to form a right-handed coordinate system (pointing to longitude  $\lambda = 90^\circ$ ). Considering the Earth's rotational speed  $\omega_\oplus$ , the ECI and ECEF frames are rotated by an angle of  $\theta_{GST}$  (Greenwich Sidereal Time) in the equatorial plane, as illustrated in Figure 2.1.

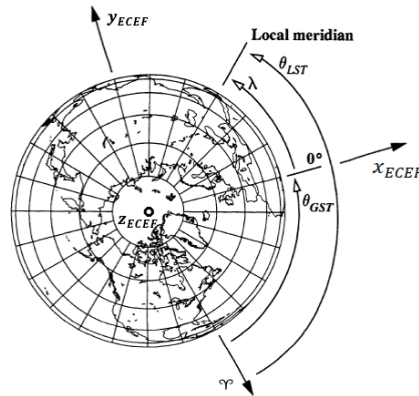


Figure 2.1: The figure illustrates the angular relation between ECI and ECEF frames. They are rotated by  $\theta_{GST}$  (Greenwich Sidereal Time) about the  $z$  axis (assumed parallel for ECI and ECEF), according to the angular speed  $\omega_\oplus$ . Figure adapted from [3].

Besides cartesian coordinates, spherical and geodetic coordinates are also useful to express a satellite's or ground station site's position. Spherical coordinates (also known as geocentric) are  $[r, \phi_{gc}, \lambda]^T$ , where  $r$  is the radial distance,  $\phi_{gc}$  is the geocentric latitude and  $\lambda$  is the longitude. Geodetic coordi-

<sup>2</sup>It is still possible to include this difference in the satellite's motion by considering a third body perturbation in the EoM due to the Sun

nates are relevant due to the ellipsoidal shape of the Earth, and are expressed as  $[h, \phi_{gd}, \lambda]^T$ , where  $h$  is height above local ground,  $\phi_{gd}$  is the geodetic latitude and  $\lambda$  the longitude. Let  $[x_f, y_f, z_f]^T$  denote ECEF cartesian coordinates of a point in space. The relation with spherical and geodetic coordinates is, respectively:

$$\begin{bmatrix} x_f \\ y_f \\ z_f \end{bmatrix} = \begin{bmatrix} r \cos \phi_{gc} \cos \lambda \\ r \cos \phi_{gc} \sin \lambda \\ r \sin \phi_{gc} \end{bmatrix}, \quad (2.1a)$$

$$\begin{bmatrix} x_f \\ y_f \\ z_f \end{bmatrix} = \begin{bmatrix} (N + h) \cos \phi_{gd} \cos \lambda \\ (N + h) \cos \phi_{gd} \sin \lambda \\ [N(1 - f_{\oplus})^2 + h] \sin \phi_{gd} \end{bmatrix}, \quad (2.1b)$$

where  $f_{\oplus}$  is the flattening parameter,  $R_{\oplus}$  is the Earth's mean equatorial radius and

$$N = \frac{R_{\oplus}}{\sqrt{1 - (2f_{\oplus} - f_{\oplus}^2) \sin^2 \phi_{gd}}}. \quad (2.2)$$

## 2.2.2 Satellite Coordinate System

The Satellite Coordinate System, denoted as RSW, is a satellite-based system centered on the satellite and moving with it along the orbit, and hence is not inertial. It is useful to analyze the relative motion of formation flying (for instance, in *rendezvous* maneuvers) or, in the context of this thesis, to analyze the position and velocity error of filtering estimates. The  $R$  axis (Radial direction) points from the Earth's center to the satellite. The  $S$  axis (Along-Track direction) is perpendicular to the  $R$  axis and lies on the orbital plane. The  $W$  axis (Cross-Track direction) is perpendicular to the orbital plane so as to form a right-handed coordinate system. Figure 2.2 illustrates the RSW frame.

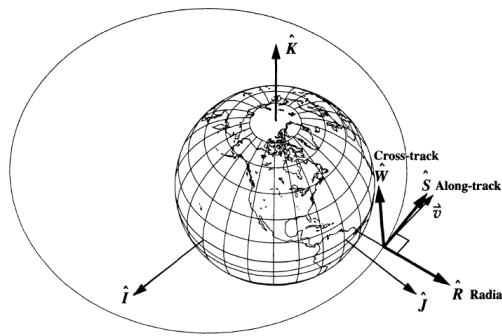


Figure 2.2: Illustration of the Satellite Coordinate System, also known as RSW. In the figure  $IKJ$  denote the ECI coordinate axes. Figure taken from [3].

## 2.2.3 Topocentric System

The topocentric system is useful to express satellite observations from ground-based sites. The fundamental plane of this system is the local horizon (plane tangent to the surface of the Earth at the

observer's site). In this thesis the East North Zenith (ENZ) realization [20] is used. The  $E$  axis is aligned with the local parallel (pointing East), the  $N$  axis is aligned with the local meridian (pointing North) and the  $Z$  axis points towards local Zenith.

Satellite measurements are usually obtained in spherical coordinates, and only then converted to cartesian. Let  $[s_E, s_N, s_Z]^T$  denote the cartesian coordinates of a point in the ENZ frame, and let  $[s, \beta, \delta]^T$  denote spherical coordinates, where  $s > 0$  is the radial distance,  $\beta \in [0, 2\pi)$  rad is the azimuth, measured from north clockwise to the satellite and  $\delta \in [-\pi/2, \pi/2]$  rad is the elevation, measured from the local horizon, positive up to the target. It is possible to relate cartesian and spherical coordinates using [20]

$$s = \sqrt{s_E^2 + s_N^2 + s_Z^2}, \quad (2.3a)$$

$$\beta = \arctan \frac{s_E}{s_N}, \quad (2.3b)$$

$$\delta = \arctan \frac{s_Z}{\sqrt{s_E^2 + s_N^2}}. \quad (2.3c)$$

## 2.2.4 Transformations

It is possible to transform vector quantities such as position  $\mathbf{r}$ , velocity  $\mathbf{v}$  or acceleration  $\mathbf{a}$  vectors between the defined reference systems. To clarify the notation, let  $\mathbf{A}_m$  be the vector quantity  $\mathbf{A}$  relative to frame  $m$ , i.e., written in terms of the basis vectors of frame  $m$ . Furthermore, when  $\mathbf{A}$  denotes the velocity or acceleration vector, the time derivative is also determined with respect to the subscripted frame. Subscript  $i$  denotes ECI,  $f$  denotes ECEF,  $s$  denotes ENZ and  $w$  denotes RSW. Below, only the transformations relevant to the work being developed are presented, which are paraphrased from [3, 20].

### ECI - ECEF

Since the ECI and ECEF systems have the same origin, changing the coordinates used to express vector quantities is achieved through a rotation matrix. However, special care must be taken when transforming velocity and acceleration vectors, since the motion of the Earth, and hence, the ECEF frame, relative to the ECI frame needs to be considered. Given the ECEF position  $\mathbf{r}_f$  and velocity  $\mathbf{v}_f$  vectors, the ECI counterparts,  $\mathbf{r}_i$  and  $\mathbf{v}_i$ , are given by:

$$\mathbf{r}_i = \mathbf{T}_f^i \mathbf{r}_f, \quad (2.4a)$$

$$\mathbf{v}_i = \mathbf{T}_f^i \mathbf{v}_f + \boldsymbol{\omega}_\oplus \times (\mathbf{T}_f^i \mathbf{r}_f), \quad (2.4b)$$

where  $\mathbf{T}_f^i$  is the rotation matrix from frame  $f$  (subscript) to frame  $i$  (superscript),  $\boldsymbol{\omega}_\oplus$  is the Earth's angular velocity vector and  $\mathbf{a} \times \mathbf{b}$  denotes the cross product between vectors  $\mathbf{a}$  and  $\mathbf{b}$ . The inverse relations are readily found by solving (2.4) for the ECEF quantities, noting that the inverse of the rotation matrix is

its transpose. The transformation between inertial and Earth-fixed accelerations is not of interest in the present work.

In the computational part of this thesis, the epoch dependent rotation matrices are computed using the Beyond Python Package<sup>3</sup> (Version 0.7.2). In addition, the package also outputs the Earth's sidereal angle  $\theta_{GST}(t)$  as a function of time. For simplicity, TOD and Pseudo Earth Fixed (PEF) are, respectively, the chosen frame realizations of the ECI and ECEF systems. These choices simplify the coordinate system transformations, since the  $z$  axes of the TOD and PEF frames are coincident, i.e.,  $z_{TOD} = z_{PEF}$ . The use of these frames greatly simplifies the computational burden of orbital propagations, but is not suitable for high-precision applications, where the J2000 is the chosen inertial frame and the International Terrestrial Reference Frame (ITRF) is usually the chosen Earth-fixed frame<sup>4</sup>.

## ECEF - ENZ

Transformation between ECEF and ENZ frames requires both a rotation and a translation, since the origin of the ENZ frame is at the local observation site. Let  $\mathbf{r}_f$  and  $\mathbf{v}_f$  be the position and velocity vectors of the target spacecraft and  $\mathbf{R}_f$  be the ground site in ECEF coordinates. The position and velocity vectors of the spacecraft in ENZ coordinates are thus<sup>5</sup>:

$$\mathbf{r}_s = \mathbf{T}_f^s (\mathbf{r}_f - \mathbf{R}_f), \quad (2.5a)$$

$$\mathbf{v}_s = \mathbf{T}_f^s \mathbf{v}_f, \quad (2.5b)$$

The rotation matrix  $\mathbf{T}_f^s$  is computed with the site's geodetic coordinates as

$$\mathbf{T}_f^s = \begin{bmatrix} -\sin \lambda & \cos \lambda & 0 \\ -\sin \phi_{gd} \cos \lambda & -\sin \phi_{gd} \sin \lambda & \cos \phi_{gd} \\ \cos \phi_{gd} \cos \lambda & \cos \phi_{gd} \sin \lambda & \sin \phi_{gd} \end{bmatrix}. \quad (2.6)$$

## ECI - RSW

RSW is an auxiliary frame used in this work for error analysis. Let  $\mathbf{e}_i$  denote an inertial error quantity (for instance, a position or velocity difference between the true and estimated state). This vector may be rotated to RSW with

$$\mathbf{e}_w = \mathbf{T}_i^w \mathbf{e}_i. \quad (2.7)$$

It is stressed that the RSW frame is not inertial, and hence, when  $\mathbf{e}$  is a velocity difference the present transformation simply rotates the error vector to RSW components, without changing the definition of

<sup>3</sup>The Beyond Python Package is an open-source orbital Python package, created by Jules David. It is available online at <https://pypi.org/project/beyond/> (last access on 16th December 2020). Software under MIT Licence.

<sup>4</sup>The ITRF frame models the motion of the Earth's rotational axis relative to its crust, which causes relative motion between the  $z$  axes of ECI and ECEF frames over time. This effect is commonly known as polar motion.

<sup>5</sup>The ENZ frame is also attached to the Earth, and thus, no time-derivative transformation between ECEF and ENZ is needed. Furthermore the ECEF time derivative of  $\mathbf{R}_f$  is null, since the station is at rest with respect to the Earth.



the time derivative. The rotation matrix is computed as

$$\mathbf{T}_i^w = \begin{bmatrix} \hat{\mathbf{R}} & \hat{\mathbf{S}} & \hat{\mathbf{W}} \end{bmatrix}^T, \quad (2.8)$$

where

$$\hat{\mathbf{R}} = \frac{\mathbf{r}_i}{\|\mathbf{r}_i\|}, \quad \hat{\mathbf{W}} = \frac{\mathbf{r}_i \times \mathbf{v}_i}{\|\mathbf{r}_i \times \mathbf{v}_i\|}, \quad \hat{\mathbf{S}} = \hat{\mathbf{W}} \times \hat{\mathbf{R}},$$

$\mathbf{r}_i$  and  $\mathbf{v}_i$  are the satellite's inertial true position and velocity.  $\|\mathbf{b}\|$  denotes the L2-norm of vector  $\mathbf{b}$ .

## 2.3 Orbital State-Space Models

Six unidimensional quantities are needed to define the state of spacecraft (without orientation), building up different orbital state-space models. Broadly speaking, these models typically fall into one of two categories:

- **State vector** - associated with position and velocity vectors;
- **Orbital element sets** - orbital elements are scalar magnitude and angular variables that specify the shape and orientation of the orbit and locate the satellite within it.

Either representation is equivalent and completely locates the body. In this thesis, the orbital state models are referenced to the ECI frame, however in GNSS applications they are usually tied to the ECEF frame [4].

### 2.3.1 State Vector

The orbital state vector is denoted by the symbol  $\mathbf{x}$  and comprises the inertial position and velocity vectors of a spacecraft, expressed in Cartesian coordinates

$$\mathbf{x} = [(\mathbf{r}_i)^T, (\mathbf{v}_i)^T]^T = [x_i, y_i, z_i, v_{x_i}, v_{y_i}, v_{z_i}]^T.$$

### 2.3.2 Keplerian Orbital Element Set

The Keplerian orbital elements allow for a much more intuitive and straightforward visualization of the orientation and shape of the orbit (conic section) in space, when compared to the state vector. The Keplerian elements are: semi-major axis  $a$ , eccentricity  $e$ , inclination  $i$ , Right Ascension of the Ascending Node (RAAN)  $\Omega$ , argument of perigee  $\varpi$  and true anomaly  $\nu$ . The full element set is represented by

$$\boldsymbol{\alpha} = [a, e, i, \Omega, \varpi, \nu]^T.$$

Sometimes, it is useful to relate the true anomaly  $\nu$  with the eccentric anomaly  $E$  and with the mean anomaly<sup>6</sup>  $M$ , for instance, when solving the two-body problem. The relation is the well-known Kepler's

<sup>6</sup>The mean anomaly  $M$  is the angular distance from the perigee which a fictitious body would have if it moved in a circular orbit, with constant speed, in the same orbital period as the actual body in its elliptical orbit [36].

Equation for elliptical form [3], given by

$$M = E - e \sin E, \quad (2.9)$$

which requires an iterative numerical method when solving for  $E$  (for instance, Algorithm 2 (*KepEqtnE*) of [3]). The relation between  $E$  and  $\nu$  is then

$$\tan \frac{E}{2} = \sqrt{\frac{1-e}{1+e}} \tan \frac{\nu}{2}. \quad (2.10)$$

### 2.3.3 Equinoctial Orbital Element Set

In many applications, satellite orbits are chosen to be near-circular or near-equatorial. While there is no inherent difficulty in calculating position and velocity vectors from known orbital elements with  $e$  or  $i$  close to zero, the reverse task may cause numerical problems [20]. To avoid these problems, alternative (more robust) sets may be employed. Among others, the equinoctial element set [27] is used. In this thesis, it is represented by the symbol  $\mathfrak{E}$ , and is defined as

$$\mathfrak{E} = \begin{bmatrix} a \\ h \\ k \\ p \\ q \\ \lambda \end{bmatrix} = \begin{bmatrix} a \\ e \sin(\varpi + \Omega) \\ e \cos(\varpi + \Omega) \\ \tan \frac{i}{2} \sin \Omega \\ \tan \frac{i}{2} \cos \Omega \\ M + \varpi + \Omega \end{bmatrix},$$

where  $a$  is the semi-major axis,  $h$  and  $k$  are components of the eccentricity vector in the equinoctial system,  $p$  and  $q$  are components of the line of nodes vector in the equinoctial system and  $\lambda$  is the mean longitude. Representation of the equinoctial system, as well as transformations between  $\mathbf{x}$  and  $\mathfrak{E}$ , are provided in Appendix D (Section D.1).

## 2.4 Orbital Mechanics

In order to study the dynamics of a spacecraft and predict its motion, which is the fundamental problem of this study, it is important to describe the mathematical and physical models of orbital mechanics (astrodynamics). Below, a simple review of orbital motion is provided, focusing on the two-body and the perturbed problems. A more detailed and in-depth development is provided in [3, 20, 34].

### 2.4.1 Kepler's Laws and Two-Body Problem

The fundamental laws that govern planetary motion were first empirically described and published by Johann Kepler between 1609 and 1619 in his works *Astronomia nova seu physica coelestis* and *Harmonices Mundi*. Later on the 17th century, Sir Isaac Newton formulated his laws of motion that

played a crucial role in providing the necessary mathematical background to support Kepler's findings, which only captured the kinematics of motion but not their dynamics.

According to Newton's laws, the simplest model for the motion of bodies in the Universe, such as the motion of planets around the Sun or satellites around the Earth, is that of a two-body problem. Figure 2.3 illustrates the two-body problem (between a satellite and the Earth), that is, a satellite  $m_{\text{sat}}$  with position  $\mathbf{r}_{\text{sat}}$  and the Earth  $m_{\oplus}$  with position  $\mathbf{r}_{\oplus}$ . As Vallado shows in [3], the satellite's acceleration relative to the Earth is given by

$$\frac{d^2 \mathbf{r}}{dt^2} = -\frac{G(m_{\oplus} + m_{\text{sat}})}{\|\mathbf{r}\|^3} \mathbf{r}, \quad (2.11)$$

where  $\mathbf{r} = \mathbf{r}_{\text{sat}} - \mathbf{r}_{\oplus}$ . If the satellite's mass is neglected and by replacing the standard gravitational parameter of the central body  $Gm_{\oplus}$  with  $\mu_{\oplus}$ , the equation above simplifies to

$$\frac{d^2 \mathbf{r}}{dt^2} = -\frac{\mu_{\oplus}}{\|\mathbf{r}\|^3} \mathbf{r}, \quad (2.12)$$

which is known as the two-body dynamics. Several assumptions are needed to apply this equation, namely [3]: 1) the mass of the satellite is neglected when compared to that of the attracting body; 2) both bodies are spherically symmetrical with uniform density (allowing for a point mass treatment); 3) the dynamical equation is implemented in an inertial frame centered on the central body; and 4) no other forces act on the system except for the gravitational forces acting along a line joining the centers of the two bodies.

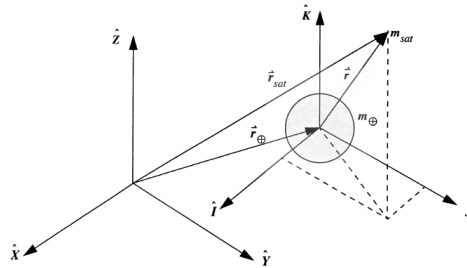


Figure 2.3: Illustration of the two-body problem between the Earth  $m_{\oplus}$  and the satellite  $m_{\text{sat}}$ . XYZ is an inertial frame. IJK is displaced from XYZ but is also inertial and does not rotate or accelerate, being centered at the Earth (ECI). Illustration taken from [3].

The solution of (2.12) leads to the well-known Keplerian orbits, which specify the trajectory of the satellite as a conic section curve. Figure 2.4 illustrates the possible geometries for Keplerian orbits (which are dependent on the initial conditions).

In the subsequent work, the analysis is confined to elliptic/circular motion. For these orbits, the solution of the two-body problem<sup>7</sup> is achieved, employing the Keplerian element set and given the initial conditions  $\mathbf{oe}(t_0) = [a_0, e_0, i_0, \Omega_0, \varpi_0, \nu_0]^T$ , by considering constant orbital shape and orientation, i.e.,  $a(t) = a_0$ ,  $e(t) = e_0$ ,  $i(t) = i_0$ ,  $\Omega(t) = \Omega_0$  and  $\varpi(t) = \varpi_0$ . The location of the satellite in the orbit is then

<sup>7</sup>The solution of the two-body problem is the time evolution of the satellite's position and velocity (or any other equivalent set of state variables) in an inertial frame.

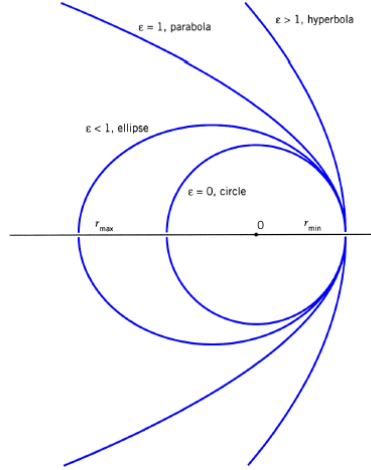


Figure 2.4: Different Keplerian orbits and the effect of the eccentricity  $e$  in the resulting conic curve. The possible geometries are: circular ( $e = 0$ ), ellipse ( $0 < e < 1$ ), parabola ( $e = 1$ ) and hyperbola ( $e > 1$ ). Illustration taken from [35].

found by implementing a linear law for the mean anomaly  $M$ , that is,

$$M(t) = M_0 + \sqrt{\frac{\mu}{a^3}}(t - t_0), \quad (2.13)$$

which may then be converted to true anomaly  $\nu$ , cf. Section 2.3.2.

## 2.4.2 Perturbed Motion

Applying (2.12) to Earth orbiting satellites neglects the effects of real-world perturbations. Perturbations are deviations from a nominal unperturbed motion, causing the satellite motion to drift apart from the two-body trajectory. Due to its daily rotation, the Earth is not a perfect sphere (nor does it have a constant density), but has the form of an oblate spheroid with an equatorial diameter that exceeds the polar one by about 20 km. Perturbations also arise from the gravitational attraction of other bodies such as the Sun, Moon or Jupiter. Satellites orbiting at low altitudes are also subject to non-conservative velocity-dependent forces caused by the resistance of the atmosphere (drag). Solar radiation pressure, thrust, magnetic fields, solid-Earth tides, ocean tides, Earth re-radiation, relativistic effects are other sources of perturbations [3, 20]. Figure 2.5 depicts the impact of various perturbations affecting Earth satellites as a function of their altitude.

To account for these effects, perturbing accelerations are added to Equation (2.12) in order to produce a more accurate model. This formulation is known as *Cowell's formulation* [3] and is given by

$$\frac{d^2 \mathbf{r}}{dt^2} = \frac{d^2 \mathbf{r}}{dt^2}_{\text{2-body}} + \frac{d^2 \mathbf{r}}{dt^2}_{\text{pert.}}. \quad (2.14)$$

Knowing the most important forces (in terms of intensity) that affect satellite motion is fundamental to properly tune the relevant state dynamics to be implemented in engineering problems.

To conclude, Table 2.1 presents some Earth's physical constants, which will be used throughout this

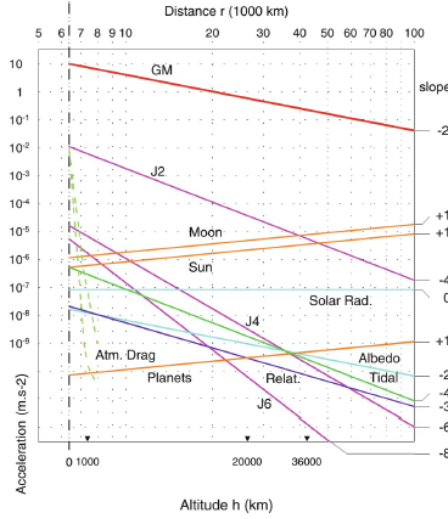


Figure 2.5: Magnitude of perturbing accelerations of Earth satellites as a function of their altitude. Graphic taken from [37].

work.

Earth's constant	Symbol	Value
Standard gravitational parameter	$\mu_{\oplus}$	$3.986004415 \times 10^5 \text{ km}^3/\text{s}^2$
Equatorial radius	$R_{\oplus}$	6378.1363 km
Flattening	$f_{\oplus}$	1/298.257
Second degree zonal coefficient	$J_2$	$1.08262693 \times 10^{-3}$
Angular velocity vector	$\omega_{\oplus}$	$[0, 0, \omega_{\oplus}]^T$
Rotational speed	$\omega_{\oplus}$	$7.292115 \times 10^{-5} \text{ rad/s}$

Table 2.1: Earth's physical constants. Values taken from [3].

## 2.5 Nonlinear Filtering

Nonlinear state estimation (also known as filtering) refers to a class of methods that are used for estimating the state of a time-varying system which is indirectly observed through noisy measurements. The state of the system is a collection of dynamic variables which fully describe it (position, velocity, orientation, etc.). Noisy measurements are not deterministic, but rather follow a probabilistic distribution. Let the system be modeled by the following nonlinear form [38]<sup>8</sup>

$$\dot{\mathbf{x}}(t) = \mathbf{f}(\mathbf{x}(t), t) + \mathbf{w}(t), \quad t > 0, \quad (2.15a)$$

$$\mathbf{y}(t_k) = \mathbf{h}_d(\mathbf{x}(t_k), t_k) + \mathbf{v}(t_k), \quad k = 0, 1, 2, 3, \dots, \quad (2.15b)$$

<sup>8</sup>The system described by (2.15) is commonly referred to in the literature as the continuous-discrete nonlinear system [38], since the state is modeled in continuous-time and the measurements are obtained at sampling discrete-times.

where  $\mathbf{x}(t) \in \mathbb{R}^n$  and  $\mathbf{w}(t) \in \mathbb{R}^n$  are, respectively, the state vector and the continuous process noise at time  $t$ ,  $\mathbf{y}(t_k) \in \mathbb{R}^m$  and  $\mathbf{v}(t_k) \in \mathbb{R}^m$  are, respectively, the measured output and the discrete measurement noise at discrete times  $t_k$ .  $\mathbf{f} : \mathbb{R}^n \times \mathbb{R}^+ \mapsto \mathbb{R}^n$  is the system dynamical function and  $\mathbf{h}_d : \mathbb{R}^n \times \mathbb{R}^+ \mapsto \mathbb{R}^m$  is the measurement function. Process noise represents both the disturbances that are unmodeled because they are unknown, as well as those intentionally excluded due to model complexity or computational limitations.

Process and measurement noise are assumed to be sequences of white, zero-mean, Gaussian noise and mutually uncorrelated with known covariance (power spectral density). These properties are summarized as [39]:

$$\mathbf{w}(t) \sim \mathcal{N}(\mathbf{0}, \mathbf{Q}(t)), \quad (2.16a)$$

$$\mathbf{v}(t_k) \sim \mathcal{N}(\mathbf{0}, \mathbf{R}_k), \quad (2.16b)$$

$$\mathbb{E}[\mathbf{w}(t)\mathbf{w}(\tau)^T] = \delta(t - \tau)\mathbf{Q}(t), \quad (2.16c)$$

$$\mathbb{E}[\mathbf{v}(t_k)\mathbf{v}(t_j)^T] = \delta_{kj}\mathbf{R}_k, \quad (2.16d)$$

$$\mathbb{E}[\mathbf{w}(t)\mathbf{v}^T(t_k)] = \mathbf{0}, \quad (2.16e)$$

where  $\delta_{ij} = 0$  for  $i \neq j$  and  $\delta_{ij} = 1$  for  $i = j$  is the *Kronecker delta*, and  $\delta(t - \tau)$  is the continuous *Dirac delta* (impulsive) function.  $\mathbb{E}[\cdot]$  denotes the expectation operator,  $\mathcal{N}$  the Gaussian distribution and the null vectors  $\mathbf{0}$  have the appropriate dimensions. Both the covariances  $\mathbf{Q}(t)$  and  $\mathbf{R}_k$  are positive definite matrices.

Furthermore, in computational applications it is possible to discretize the state dynamics via numerical solver routines of ordinary differential equations [40]. These methods replace the continuous-time dynamics of Equation (2.15a) with the following discretized form

$$\mathbf{x}(t_{k+1}) = \mathbf{f}_d(\mathbf{x}(t_k), t_k) + \mathbf{w}_d(t_k) \quad k = 0, 1, \dots \quad (2.17)$$

It should be noted that the discretization is being performed considering noiseless state dynamics, i.e., by setting  $\mathbf{w}(t) = \mathbf{0}$ , and then by explicitly adding the discretized process noise  $\mathbf{w}_d(t_k)$ . A more rigorous approach would be to discretize directly the stochastic differential system, which is crucial when a precise relation between continuous and discretized noise is intended. The concept of stochastic discretization of nonlinear dynamics in the context of Kalman estimation is further explored in [41].

### 2.5.1 Nonlinear Transformation of Random Variables

In this section the fundamental linearization procedures that are applied in the filtering algorithms to be implemented are reviewed. Let  $\mathbf{x}$  denote a Gaussian Random Variable (GRV) with mean  $\bar{\mathbf{x}} = \mathbb{E}[\mathbf{x}]$  and covariance  $\mathbf{P}_x = \mathbb{E}[(\mathbf{x} - \bar{\mathbf{x}})(\mathbf{x} - \bar{\mathbf{x}})^T]$  that undergoes an arbitrary nonlinear transformation

$$\mathbf{y} = \mathbf{g}(\mathbf{x}). \quad (2.18)$$

Given this formulation, the problem of nonlinear transformation of random variables tries to compute a Gaussian approximation of the statistics of the transformed Random Variable (RV)  $y$ , namely, its mean  $\bar{y}$  and covariance  $P_y$ . In the following, the first-order Taylor series linearization, the weighted statistical linearization and the Unscented Transform are reviewed.

### First-Order Taylor Series Linearization

In order to analytically calculate  $\bar{y}$  and  $P_y$ , a multidimensional Taylor series expansion around  $\bar{x}$  is employed (truncated to first order) [42, 43], that is,

$$y = g(x) \approx g(\bar{x}) + \left. \frac{\partial g(x)}{\partial x} \right|_{x=\bar{x}} (x - \bar{x}). \quad (2.19)$$

The statistics of  $y$  are then given by:

$$\bar{y} = g(\bar{x}), \quad (2.20a)$$

$$P_y = G_x(\bar{x}) P_x G_x^T(\bar{x}), \quad (2.20b)$$

where  $G_x(\bar{x}) = \left. \frac{\partial g(x)}{\partial x} \right|_{x=\bar{x}}$ . To conclude, function  $g$  can now be approximated, near the expansion point  $\bar{x}$ , by

$$y^{\text{lin.}} = g(\bar{x}) + G_x(\bar{x})(x - \bar{x}). \quad (2.21)$$

### Weighted Statistical Linearization and the Unscented Transform

The Weighted Statistical Linearization (WSL) is a linearization technique that takes into account the uncertainty or *probabilistic spread* of the prior RV when linearizing the function. By doing so, the resulting function is more accurate (in a statistical sense) than simply using a first-order Taylor series linearization around the mean of the RV. A complete analysis of the inaccuracy of the Taylor approximation when dealing with RVs and a comparison with the Unscented Transform and other Sigma-Point WSL transformations is given in [43].

Consider the nonlinear function  $y = g(x)$  which is evaluated in  $N$  points  $(\mathcal{X}_i, i = 1, \dots, N)$ , i.e.,  $\mathcal{Y}_i = g(\mathcal{X}_i)$ ,  $i = 1, \dots, N$ , where the points  $\mathcal{X}_i$  are chosen such that they capture certain statistical properties of  $x$ . The aim of WSL is to determine the linearization parameters  $A$  and  $b$  such that the function may be approximated by the following linearized regression

$$g(x) \approx y^{\text{lin.}} = Ax + b. \quad (2.22)$$

The following statistics can be computed [43]:

$$\bar{x} = \sum_{i=1}^N \mathcal{W}_i \mathcal{X}_i, \quad (2.23a)$$

$$P_x = \sum_{i=1}^N \mathcal{W}_i (\mathcal{X}_i - \bar{x})(\mathcal{X}_i - \bar{x})^T, \quad (2.23b)$$

$$\bar{\mathbf{y}} = \sum_{i=1}^N \mathcal{W}_i \mathbf{y}_i, \quad (2.23c)$$

$$\mathbf{P}_{\mathbf{y}} = \sum_{i=1}^N \mathcal{W}_i (\mathbf{y}_i - \bar{\mathbf{y}}) (\mathbf{y}_i - \bar{\mathbf{y}})^T, \quad (2.23d)$$

$$\mathbf{P}_{\mathbf{xy}} = \sum_{i=1}^N \mathcal{W}_i (\mathbf{x}_i - \bar{\mathbf{x}}) (\mathbf{y}_i - \bar{\mathbf{y}})^T \quad (2.23e)$$

where  $\mathcal{W}_i$  is a set of  $N$  scalar regression weights that sum to one. By minimizing the weighted sum of the squared errors  $\mathcal{E}_i = \mathbf{y}_i - (\mathbf{A}\mathbf{x}_i + \mathbf{b})$ , i.e.,

$$\{\mathbf{A}, \mathbf{b}\} = \operatorname{argmin} \sum_{i=1}^N \mathcal{W}_i \mathcal{E}_i^T \mathcal{E}_i, \quad (2.24)$$

one may compute the linearization parameters, which are given by:

$$\mathbf{A} = \mathbf{P}_{\mathbf{xy}}^T \mathbf{P}_{\mathbf{x}}^{-1}, \quad (2.25a)$$

$$\mathbf{b} = \bar{\mathbf{y}} - \mathbf{A}\bar{\mathbf{x}}. \quad (2.25b)$$

As shown in [43], the Unscented Transform (UT) makes implicit use of the WSL procedure to obtain an implied statistically linearized system model for use within nonlinear filtering processes. Hence, the UT is an example application of the WSL framework. Quoting from [44], "the UT is founded on the intuition that it is easier to approximate a probability distribution than it is to approximate an arbitrary nonlinear function or transformation". The UT forms a fixed number of Sigma Points, which capture the mean and covariance of the original distribution of  $\mathbf{x}$ . These points are then propagated through the non-linearity and finally the mean and covariance of the transformed variable are estimated from them<sup>9</sup>. The UT is presented in algorithmic-form in Algorithm 1, paraphrased from [43, 44].

In Equation (2.26)  $\sqrt{(n + \lambda)\mathbf{P}_{\mathbf{x}}}$  is the matrix square root of the weighted covariance (positive definite) matrix  $(n + \lambda)\mathbf{P}_{\mathbf{x}}$  and may be computed using efficient *Cholesky decomposition* methods [40]. In the experimental development of the thesis, this decomposition is performed using the Scientific Python (SciPy) Package<sup>10</sup> [45] (version 1.5.4).

Constants  $\alpha, \beta$  and  $\kappa$  are parameters of the transform.  $\alpha$  determines the spread of the sigma points around the mean  $\bar{\mathbf{x}}$  and is usually in the interval  $[10^{-4}, 1]$  [46],  $\kappa$  is a scaling parameter and is usually set to 0 (for state estimation) or  $3 - n$  (for parameter estimation) [47] and  $\beta$  is used to incorporate prior knowledge of the distribution of  $\mathbf{x}$  ( $\beta = 2$  is the optimal value for Gaussian distributions [46]). In the present work, these constants are set to  $\alpha = 1$ ,  $\beta = 2$  and  $\kappa = 0$ .

<sup>9</sup>Notice that the UT is different from Monte Carlo estimation (used in Particle Filters), because the selection of Sigma Points is deterministic.

<sup>10</sup>The SciPy package is available online at <https://www.scipy.org> (last access on 22nd December 2020)



---

**Algorithm 1** Unscented Transform (UT)

---

**Require:** GRV  $x \in \mathbb{R}^n$  characterized by the distribution  $x \sim \mathcal{N}(\bar{x}, P_x)$ ;

Nonlinear function  $g : \mathbb{R}^n \mapsto \mathbb{R}^r$ ;

$(\alpha, \beta, \kappa)$ .

$\triangleright$  parameters of the transform

**Ensure:** The UT is used for forming the Gaussian approximation

$$\begin{pmatrix} x \\ y \end{pmatrix} \sim \mathcal{N} \left( \begin{pmatrix} \bar{x} \\ \bar{y} \end{pmatrix}, \begin{pmatrix} P_x & P_{xy} \\ P_{xy}^T & P_y \end{pmatrix} \right)$$

of the joint distribution of  $x$  and  $y \in \mathbb{R}^r$  undergoing the transformation  $g(\cdot)$ .

1: **function** UT( $g, \bar{x}, P_x$ )

2: Form the set of  $2n + 1$  Sigma Points  $\mathcal{X}_i$  as follows:

$$\begin{aligned} \mathcal{X}_0 &= \bar{x}, \quad i = 0, \\ \mathcal{X}_i &= \bar{x} + [\sqrt{(n + \lambda)P_x}]_i, \quad i = 1, \dots, n, \\ \mathcal{X}_i &= \bar{x} - [\sqrt{(n + \lambda)P_x}]_{i-n}, \quad i = n + 1, \dots, 2n, \end{aligned} \tag{2.26}$$

where  $\lambda$  is a scaling parameter defined as

$$\lambda = \alpha^2(n + \kappa) - n, \tag{2.27}$$

and  $[\cdot]_i$  denotes the  $i$ th column of the matrix inside the brackets.

3: Compute the associated weights:

$$\begin{aligned} \mathcal{W}_0^{(m)} &= \lambda / (n + \lambda), \quad \mathcal{W}_i^{(m)} = 1 / [2(n + \lambda)], \quad i = 1, \dots, 2n, \\ \mathcal{W}_0^{(c)} &= \lambda / (n + \lambda) + (1 - \alpha^2 + \beta), \quad \mathcal{W}_i^{(c)} = 1 / [2(n + \lambda)], \quad i = 1, \dots, 2n, \end{aligned} \tag{2.28}$$

4: Transform each of the Sigma Points with

$$\mathcal{Y}_i = g(\mathcal{X}_i), \quad i = 0, \dots, 2n. \tag{2.29}$$

5: Finally, compute the statistics:

$$\bar{y} = \sum_{i=0}^{2n} \mathcal{W}_i^{(m)} \mathcal{Y}_i, \tag{2.30a}$$

$$P_y = \sum_{i=0}^{2n} \mathcal{W}_i^{(c)} (\mathcal{Y}_i - \bar{y})(\mathcal{Y}_i - \bar{y})^T, \tag{2.30b}$$

$$P_{xy} = \sum_{i=0}^{2n} \mathcal{W}_i^{(c)} (\mathcal{X}_i - \bar{x})(\mathcal{Y}_i - \bar{y})^T. \tag{2.30c}$$

6: **return**  $[\bar{y}, P_y, P_{xy}]$

7: **end function**

---

## 2.5.2 Kalman Filter

The purpose of Bayesian filtering is to compute the joint *a posteriori* probability density function (pdf) of the state  $x_k$  at each time step  $k$  given the history of measurements up to that time,  $p(x_k | y_{1:k})$ . The formal and exact solution of the continuous-time and discrete-time nonlinear problems (investigated in [48] and [39], respectively) is, in the general case, unfeasible because an explicit closed-form solution of the equations is intractable for most systems and an infinite amount of computational resources would be needed. Therefore, one resorts to sub-optimal algorithms, namely, Kalman filters [39].

The Kalman Filter (KF) was developed by Rudolph Kalman in 1960 [49] and consists of a linear,

discrete-time and time-varying system characterized by a finite-dimensional state vector and a sequence of noisy observations from which the state is inferred by minimizing a quadratic function of the estimation error. Proof and development of the KF equations may be found in [39, 49].

The filtering estimate at time  $t_k$  is computed as a two-step procedure. In the first step (predict step) the state model is used to predict the current (mean) state estimate and covariance based on knowledge up to the previous instant  $t_{k-1}$ . The *a priori* mean and covariance are defined as:

$$\hat{\mathbf{x}}_k^- = \mathbb{E}[\mathbf{x}_k | \mathbf{y}_{1:k-1}],$$

$$\mathbf{P}_k^- = \mathbb{E}[(\mathbf{x}_k - \hat{\mathbf{x}}_k^-)(\mathbf{x}_k - \hat{\mathbf{x}}_k^-)^T | \mathbf{y}_{1:k-1}].$$

where the superscript "-" indicates predicted estimates. Then, in the second step (update step) the measurement is used to correct the predicted state, producing the *a posteriori* distribution, denoted with the superscript "+", that is:

$$\hat{\mathbf{x}}_k^+ = \mathbb{E}[\mathbf{x}_k | \mathbf{y}_{1:k}],$$

$$\mathbf{P}_k^+ = \mathbb{E}[(\mathbf{x}_k - \hat{\mathbf{x}}_k^+)(\mathbf{x}_k - \hat{\mathbf{x}}_k^+)^T | \mathbf{y}_{1:k}].$$

Once initialized, the algorithm repeats its predict-update sequence for each time step  $k$ . A representative diagram of the algorithm is illustrated in Figure 2.6.

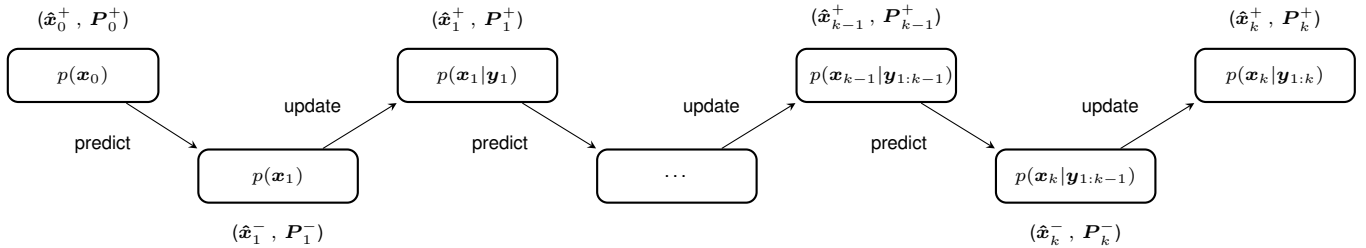


Figure 2.6: Consecutive predict and update steps on KF dynamics. Rather than propagating the entire pdf the KF only propagates the first (mean) and second (covariance) moments.

## Extended Kalman Filter

With nonlinear dynamics, the KF is no longer optimal<sup>11</sup>. Nonetheless, it is possible to extend the applicability of KFs to nonlinear systems. The Extended Kalman Filter (EKF) is one of such filters. The system is approximated by a linearized version around the state estimate (using first-order Taylor series expansion) and then the conventional KF algorithm is applied.

In this thesis the continuous-discrete EKF is considered, based on deductions made in [3, 20]. An explicit discretization of the state dynamics is therefore avoided.<sup>12</sup> The EKF is presented in Algorithm 2, adapted from the aforementioned references. This algorithm computes approximate solutions to filtering

<sup>11</sup>Under the assumptions that the system is fully linear, the initial state is Gaussian and the process and measurement noise sequences are white and Gaussian, then the filtering pdf  $p(\mathbf{x}_k | \mathbf{y}_{1:k})$  is Gaussian for all time instants and the KF is optimal [39].

<sup>12</sup>Modeling the dynamics directly in continuous-time is more accurate, since the laws that govern physical systems are naturally written in continuous-time.

problems as described by the system of Equations (2.15). Discretization of the process noise sequence in Equation (2.33a) is shown in Appendix B.

---

**Algorithm 2** Extended Kalman Filter (EKF)

---

**Require:** system functions  $f$ ,  $h_d$  and Jacobians  $F$  and  $H$ ;

$$\hat{x}_0^+ = \mathbb{E}[x(t_0)] = \bar{x}_0; \quad \triangleright \text{initial state estimate at } t_0$$

$$P_0^+ = \mathbb{E}[(x(t_0) - \bar{x}_0)(x(t_0) - \bar{x}_0)^T] = \bar{P}_0; \quad \triangleright \text{initial covariance estimate at } t_0$$

$$\{(t_k, y_k, R_k) : k = 1, \dots, T\}; \quad \triangleright \text{sequence of (possible infinite) observations to process}$$

$$Q(t) = Q, \quad \forall t. \quad \triangleright \text{constant continuous-time process noise covariance}$$

**Ensure:**  $\hat{X} = \{(\hat{x}_k^+, P_k^+) : k = 1, \dots, T\}$ .  $\triangleright$  estimated output trajectory (mean and covariance)

1: **procedure** EKF

2:     **Definitions:** The system Jacobians are defined as:

$$F(\hat{x}(t), t) = \left. \frac{\partial f(x(t), t)}{\partial x(t)} \right|_{x(t)=\hat{x}(t)}, \quad (2.31)$$

$$H(\hat{x}_k^-, t_k) = \left. \frac{\partial h_d(x_k, t_k)}{\partial x_k} \right|_{x_k=\hat{x}_k^-}.$$

3:     **for**  $t_k \in \{t_1, \dots, t_T\}$  **do**

4:         **Predict step:** Numerically propagate mean and State Transition Matrix (STM) dynamics in the interval  $t \in [t_{k-1}, t_k]$ , from the initial conditions  $\hat{x}(t_{k-1}) = \hat{x}_{k-1}^+$  and  $\Phi(t_{k-1}, t_{k-1}) = I_{n \times n}$  with:

$$\begin{aligned} \dot{\hat{x}}(t) &= f(\hat{x}(t), t), \\ \dot{\Phi}(t, t_{k-1}) &= F(\hat{x}(t), t)\Phi(t, t_{k-1}). \end{aligned} \quad (2.32)$$

The integration produces  $\hat{x}_k^- = \hat{x}(t)|_{t=t_k}$  and  $\Phi(t_k, t_{k-1}) = \Phi(t, t_{k-1})|_{t=t_k}$ .

5:         Discretize the process noise covariance and compute the propagated state covariance:

$$Q_d(t_k, t_{k-1}) = \Phi(t_k, t_{k-1}) Q \Phi^T(t_k, t_{k-1}) \cdot (t_k - t_{k-1}), \quad (2.33a)$$

$$P_k^- = \Phi(t_k, t_{k-1}) P_{k-1}^+ \Phi^T(t_k, t_{k-1}) + Q_d(t_k, t_{k-1}). \quad (2.33b)$$

6:         **Update step:** Compute the Kalman gain and update the state mean and covariance:

$$K_k = P_k^- H^T(\hat{x}_k^-, t_k) [H(\hat{x}_k^-, t_k) P_k^- H^T(\hat{x}_k^-, t_k) + R_k]^{-1}, \quad (2.34a)$$

$$\hat{x}_k^+ = \hat{x}_k^- + K_k(y_k - h_d(\hat{x}_k^-, t_k)), \quad (2.34b)$$

$$P_k^+ = [I_{n \times n} - K_k H(\hat{x}_k^-, t_k)] P_k^-. \quad (2.34c)$$

7:         Append updated state and covariance to the estimated trajectory  $\hat{X}$ .

8:     **end for**

9:     **return**  $\hat{X}$

10: **end procedure**

---

The process noise  $w(t)$  is assumed to have constant covariance  $Q(t) = Q$ , which implies that the stochastic process has infinite variance. Although this is only an idealized concept that is not physically realizable, it does serve as a useful mechanism for modeling system uncertainties and disturbances [50]. Furthermore, in (2.34c),  $I_{n \times n}$  denotes the identity matrix of dimension  $n$ , where  $n$  is the dimension of the state vector.

## Unscented Kalman Filter

The local linear approximations performed by the EKF may not work in problems with significant nonlinearities and thus may lead to large estimation errors or even filter divergence. In order to overcome these kinds of problems, Julier and Uhlmann proposed a new method in 1997 known as the Unscented Kalman Filter (UKF) [44].

The filtering pdf is also only approximate, but now its transformation through the nonlinear system relies on the UT (Section 2.5.1). In this work the discrete-discrete UKF is considered, resorting to the system characterized by (2.17) and (2.15b), cf. the original implementation of the UKF in [44]. Notice that the state dynamics are discretized beforehand. In recent years, the UKF has been extended to systems modeled directly in continuous-time [51]. This algorithm was not considered on this thesis and its implementation and comparison with the EKF and conventional UKF (both in performance and computational cost) on the context of OD is left as future work.

The UKF algorithm is presented in Algorithm 3, being paraphrased from [46, 51].

## 2.6 Signals in the Frequency Domain

In this section a brief overview of the Fourier Series and the Discrete Fourier Transform is given. The concepts presented here are based on those developed in [52] and will be useful when modeling the orbital dynamics in Chapter 3.

### 2.6.1 Fourier Series

#### 1-Dimensional

The Fourier Series (FS) is the frequency-domain representation of a continuous-time periodic signal  $x(t)$  with period  $T$  as a sum of an infinite set of harmonically related sinusoids in addition to a constant (DC) value. The signal's fundamental frequency is  $\omega_0 = \frac{2\pi}{T}$ . The frequencies of the other harmonics are integral multiples of  $\omega_0$ . The sufficient conditions that a signal has to satisfy so that it can be represented by a FS are: 1) the signal  $x(t)$  is absolutely integrable, i.e.,  $\int_0^T |x(t)| dt < \infty$ ; 2) the signal has a finite number of maxima and minima in one period; and 3) the signal has a finite number of discontinuities in one period. These are known as the Dirichlet conditions.

A real periodic signal  $x(t)$  satisfying the Dirichlet conditions can be decomposed as the following FS

$$x(t) = \sum_{k=-\infty}^{\infty} \mathcal{F}_{cs}(k) e^{jk\omega_0 t}, \quad (2.38)$$

where  $j = \sqrt{-1}$  is the imaginary unit and the FS frequency coefficients are defined as

$$\mathcal{F}_{cs}(k) = \frac{1}{T} \int_0^T x(t) e^{-jk\omega_0 t} dt, \quad k = -\infty, \dots, -1, 0, 1, \dots, \infty. \quad (2.39)$$

---

**Algorithm 3** Unscented Kalman Filter (UKF)

---

**Require:** system functions  $f_d, h_d$ ;

$$\hat{x}_0^+ = \mathbb{E}[x(t_0)] = \bar{x}_0; \quad \triangleright \text{initial estimate at } t_0$$

$$P_0^+ = \mathbb{E}[(x(t_0) - \bar{x}_0)(x(t_0) - \bar{x}_0)^T] = \bar{P}_0; \quad \triangleright \text{initial estimate covariance at } t_0$$

$$\{(t_k, y_k, R_k) : k = 1, \dots, T\}; \quad \triangleright \text{sequence of (possible infinite) observations to process}$$

$$Q(t) = Q, \quad \forall t. \quad \triangleright \text{constant continuous-time process noise covariance}$$

**Ensure:**  $\hat{X} = \{(\hat{x}_k^+, P_k^+) : k = 1, \dots, T\}$ .  $\triangleright$  estimated output trajectory (mean and covariance)

1: **procedure** UKF

2:   **for**  $t_k \in \{t_1, \dots, t_T\}$  **do**

3:     **Predict step:** Compute the mean and covariance (without process noise) of the discretized state dynamics using the UT (Algorithm 1)

$$[\hat{x}_k^-, P_k, P_{k,k-1}] = \text{UT}(f_d, \hat{x}_{k-1}^+, P_{k-1}^+). \quad (2.35)$$

4:     Compute the WSL equivalent of the STM, discretize the process noise covariance and compute the propagated state covariance:

$$A(t_k, t_{k-1}) = P_{k,k-1} (P_{k-1}^+)^{-1}, \quad (2.36a)$$

$$Q_d(t_k, t_{k-1}) = A(t_k, t_{k-1}) Q A^T(t_k, t_{k-1}) \cdot (t_k - t_{k-1}), \quad (2.36b)$$

$$P_k^- = P_k + Q_d(t_k, t_{k-1}). \quad (2.36c)$$

5:     **Update step:** Perform the UT of the measurement function, compute the Kalman gain and update the state mean and covariance:

$$[\bar{y}_k, \tilde{P}_y, P_{xy}] = \text{UT}(h_d, \hat{x}_k^-, P_k^-), \quad (2.37a)$$

$$P_y = \tilde{P}_y + R_k, \quad (2.37b)$$

$$K_k = P_{xy} P_y^{-1}, \quad (2.37c)$$

$$x_k^+ = x_k^- + K_k(y_k - \bar{y}_k), \quad (2.37d)$$

$$P_k^+ = P_k^- - K_k P_y K_k^T. \quad (2.37e)$$

6:     Append updated state and covariance to the estimated trajectory  $\hat{X}$ .

7:   **end for**

8:   **return**  $\hat{X}$

9: **end procedure**

---

## 2-Dimensional

The theory of 2-Dimensional (2-D) signals, for the most part, is a straightforward extension of the theory of 1-Dimensional (1-D) signals. In view of this, let  $x(t_1, t_2)$  be a periodic signal with periods  $T_1$  and  $T_2$  respectively in axes  $t_1$  and  $t_2$ . Its FS takes the form

$$x(t_1, t_2) = \sum_{k_1=-\infty}^{\infty} \sum_{k_2=-\infty}^{\infty} \mathcal{F}_{cs}(k_1, k_2) e^{j\omega_1 k_1 t_1} e^{j\omega_2 k_2 t_2}, \quad (2.40)$$

where  $\omega_1 = 2\pi/T_1$ ,  $\omega_2 = 2\pi/T_2$  and

$$\mathcal{F}_{cs}(k_1, k_2) = \frac{1}{T_1 T_2} \int_0^{T_1} \int_0^{T_2} x(t_1, t_2) e^{-j\omega_1 k_1 t_1} e^{-j\omega_2 k_2 t_2} dt_1 dt_2. \quad (2.41)$$

## 2.6.2 Discrete Fourier Transform

### 1-Dimensional

Let  $x(n)$  denote an  $N$  point finite-length signal. The Discrete Fourier Transform (DFT) transforms  $x(n)$  into a set of  $N$  frequency coefficients  $X(k)$ , which are the representation of the given time-domain sequence in the frequency domain. The Inverse Discrete Fourier Transform (IDFT) transforms the  $N$  frequency-domain coefficients back into the original time-domain sequence. The DFT and IDFT transformations are defined as:

$$X(k) = \sum_{n=0}^{N-1} x(n) e^{-j \frac{2\pi}{N} kn}, \quad k = 0, \dots, N-1 \quad \text{DFT equation,} \quad (2.42a)$$

$$x(n) = \frac{1}{N} \sum_{k=0}^{N-1} X(k) e^{j \frac{2\pi}{N} kn}, \quad n = 0, \dots, N-1 \quad \text{IDFT equation.} \quad (2.42b)$$

### 2-Dimensional

Let  $x(n_1, n_2)$  be a 2-D time-domain finite sequence where the directions  $n_1$  and  $n_2$  have length  $N_1$  and  $N_2$ , respectively. In this case, the DFT and IDFT are defined as:

$$X(k_1, k_2) = \sum_{n_1=0}^{N_1-1} \sum_{n_2=0}^{N_2-1} x(n_1, n_2) e^{-j \frac{2\pi}{N_1} k_1 n_1} e^{-j \frac{2\pi}{N_2} k_2 n_2}, \quad k_{1,2} = 0, \dots, N_{1,2} - 1 \quad \text{DFT equation,} \quad (2.43a)$$

$$x(n_1, n_2) = \frac{1}{N_1 N_2} \sum_{k_1=0}^{N_1-1} \sum_{k_2=0}^{N_2-1} X(k_1, k_2) e^{j \frac{2\pi}{N_1} k_1 n_1} e^{j \frac{2\pi}{N_2} k_2 n_2}, \quad n_{1,2} = 0, \dots, N_{1,2} - 1 \quad \text{IDFT equation.} \quad (2.43b)$$

## 2.6.3 Relation Between Discrete Fourier Transform and Fourier Series

### 1-Dimensional

In the real world, signals are mostly continuous and aperiodic. The accurate representation of such signals requires an infinite number of samples in the frequency and time domains. Both the DFT and FS serve the same purpose, that is, providing a representation of signals in the frequency domain. However, the FS analyzes a continuous-time signal (with possible infinite frequency components), whereas the DFT analyzes a discrete signal. Despite this, in actual applications it is much more practical to work with the DFT, due to its finite nature. Hence, an approximation of the DFT and FS frequency coefficients is desired.

The integral in (2.39) can be approximated using the rectangular rule of numerical integration. By dividing the period  $T$  into  $N$  intervals, each of those with width  $T_s = T/N$ , and sampling the signal at  $N$

points with  $x(t) = x(nT_s)$ , then (2.39) is approximated as

$$\begin{aligned}\mathcal{F}_{cs}(k) &\approx \frac{1}{T} \sum_{n=0}^{N-1} x(nT_s) e^{-jk\omega_0 nT_s} \frac{T}{N} \\ &= \frac{1}{N} \sum_{n=0}^{N-1} x(nT_s) e^{-j\frac{2\pi}{N}nk}, \quad k = 0, 1, \dots, N-1,\end{aligned}\tag{2.44}$$

where  $x(nT_s) = x(n)$  is the  $n$ th sample of the signal. By comparing (2.42a) and (2.44), the following relation is found the between FS and DFT coefficients:

$$\mathcal{F}_{cs}(k) = \frac{1}{N} X(k), \quad k = 0, \dots, N-1.\tag{2.45}$$

If  $x(t)$  is sampled with  $x(t) = x((n_0 + n)T_s)$ , i.e., with a delayed signal, then the following relation is found instead (using the DFT shifting theorem [52])

$$\mathcal{F}_{cs}(k) = e^{-j\frac{2\pi}{N}n_0k} \frac{X(k)}{N} = e^{-jk\omega_0 n_0 T_s} \frac{X(k)}{N}, \quad k = 0, \dots, N-1\tag{2.46}$$

A periodic signal can be represented uniquely by a spectrum with an infinite number of harmonically related sinusoids. Since in the DFT only a finite number of frequency components is produced, a problem known as *aliasing* arises. As shown in [52], when applying the DFT to a given signal, the sampling theorem states that the index of the highest frequency component the signal is composed of must be less than  $N/2$ , in order to represent the signal unambiguously with  $N$  DFT coefficients. On the other hand, if the signal contains frequency components greater than  $N/2$ , the DFT coefficients are corrupted and cannot recover the original time-domain signal. This usually leads to the high-frequency sinusoids being reconstructed as low frequency ones.

Hence, when approximating FS coefficients with DFT coefficients, one way to prevent or reduce *aliasing* is by ensuring that the signal is composed only of frequency components with index less than  $N/2$  (by prefiltering it with a low pass filter). Otherwise, the *aliasing* effect may corrupt the spectrum.

## 2-Dimensional

The relation between the 2-D FS and DFT is obtained in a similar way to the 1-D case. Sampling the signal  $x(t_1, t_2)$  at  $N_1 \times N_2$  points with  $x((n_{0,1} + n_1)T_{1s}, (n_{0,2} + n_2)T_{2s})$  and approximating the integral of (2.41) one obtains

$$\mathcal{F}_{cs}(k_1, k_2) = \frac{e^{-jk_1\omega_1 n_{0,1}T_{1s}} e^{-jk_2\omega_2 n_{0,2}T_{2s}}}{N_1 N_2} X(k_1, k_2),\tag{2.47}$$

where  $T_{1s} = T_1/N_1$ ,  $T_{2s} = T_2/N_2$  and  $n_{0,1}, n_{0,2}$  are the time delay indexes.

## 2.7 Performance Evaluation

Performance evaluation is a very important tool to develop, evaluate and compare algorithms. Different criteria may be employed depending on the applications and experiments. In the context of state estimation, commonly used criteria are accuracy and computational cost.

### 2.7.1 Accuracy

Accuracy is a measure of how well the produced estimates relate to the true state [39]. Throughout this work, the Root Mean Squared Error (RMSE) will become the basic tool to construct accuracy studies. Let  $\hat{x}(t_k)$  be a filter estimate of the true state  $x(t_k)$ , for  $k = 1, \dots, T$ , where  $t_T$  is the last instant of the estimation. In the context of this thesis,  $x$  may denote a position or velocity vector or any other orbital state variable.

The trajectory averaged RMSE is defined as [53]

$$\overline{\text{RMSE}} = \sqrt{\frac{1}{TN} \sum_{k=1}^T \sum_{i=1}^N \|x(t_k) - \hat{x}(t_k)\|^2}, \quad (2.48)$$

where  $N$  is the number of independent Monte Carlo estimation runs evaluated. The purpose of repeating the estimation algorithms  $N$  times, with realizations of the random noise variables independent from run to run, is to decrease the variability of the statistics and thus increase the power of the tests [39].

The previous definition represents the RMSE averaged along the trajectory in the interval  $t \in [t_1, t_T]$ . However, it is also possible to define the RMSE for each time instant  $t_k$ , as follows:

$$\text{RMSE}(t_k) = \sqrt{\frac{1}{N} \sum_{i=1}^N \|x(t_k) - \hat{x}(t_k)\|^2}, \quad (2.49)$$

which is a more convenient definition for plotting the error evolution over time.

### 2.7.2 Computational Cost

Cost metrics refer to how expensive a given method is to implement, either in terms of money, time or required technology. Sometimes, a less accurate algorithm may be preferred if it is less expensive to implement and still satisfies some predefined accuracy requirements. In this thesis the cost metric to be used in the simulation tests of the different algorithms implemented is CPU computation time. The algorithms and simulations were developed in the Python programming language<sup>13</sup> (version 3.7) and conducted on a laptop computer with 8GB RAM and Intel® i7-4210U, 1.7 GHz processor.

---

<sup>13</sup>Python Software Foundation. Python Language Reference available at <http://www.python.org> (last access on 17th December 2020).



## Chapter 3

# Modeling and Implementation

This chapter starts by modeling two orbital state dynamical formulations (osculating and mean dynamics), followed by the modeling of measurement model. Then, the orbital Kalman filters are assembled with these models.

### 3.1 Dynamics Modeling

As discussed in Section 2.4.2, satellite motion is affected by disturbing forces which cause the motion to drift away from pure two-body dynamics. There exist three main approaches in the literature to include the effects of perturbations in the dynamical model of orbital motion [3]:

- **Special Perturbation (SP)** - SP methods numerically integrate the EoM using high-fidelity force models that provide space operators with accurate predictions of positions and velocities of space objects. The major disadvantage of SP methods is their strictly sequential and step-wise processing when propagating from one epoch to another [12]. One of the most common formulations within SP methods is *Cowell's formulation*, modeled by (2.14). The General Mission Analysis Tool (GMAT)<sup>1</sup> is an example of an open-source software package implementing *Cowell* propagation;
- **General Perturbation (GP)** - GP methods replace the EoM with analytical approximations that capture the essential behavior of the motion over a limited time interval and which permits analytical integration, relying usually on series expansions of the perturbing accelerations [3]. They can be used to compute orbital positions at arbitrary times and the computational effort does not grow with the time between the epoch of the orbital elements and the epoch of the computation [12] (in contrast to SP methods). This trade-off speeds up computation but decreases accuracy. The SGP4 propagator [54] is an example of a GP propagator;
- **Semianalytical** - Semianalytical methods combine the best features of SP and GP methods to

---

<sup>1</sup>GMAT is an open-source orbital mission analysis tool provided by the National Aeronautics and Space Administration (NASA) agency. Available online at (last access on 12th December 2020) <https://software.nasa.gov/software/GSC-18094-1>. A physical and mathematical description of the GMAT's software may be found in <http://gmat.sourceforge.net/doc/R2020a/GMATMathSpec.pdf> (last access on 15th December 2020).

attain a superior mix of accuracy and efficiency. The Draper Semianalytic Satellite Theory (DSST) propagator [27] is one of the most popular and well-documented semianalytical propagators<sup>2</sup>.

SP methods provide the best accuracy, but are also the heaviest (in computational sense). Furthermore, they do not provide much about the qualitative behavior of the orbit, since the integration usually resorts to the orbital state vector. GP and Semianalytical methods, on the other hand, are modeled with orbital element sets [3].

The effects of perturbations on the time evolution of a generic orbital element are illustrated in Figure 3.1 and may be classified as: 1) secular effects, and 2) periodic effects. The former have an approximately linear variation over time. The latter are either long- or short-periodic, depending on the amount of time required for the effects to repeat. Short periodic-effects usually repeat on the order of the satellite's orbital period, whereas long-periodic effects have cycles longer than one period.

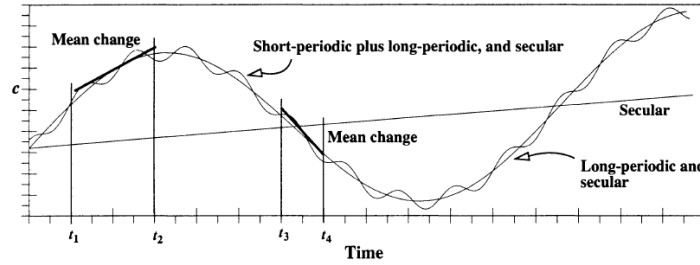


Figure 3.1: Effects of perturbing forces on the time evolution of a generic orbital element. The straight (linear-like) line shows secular effects. The oscillating (sinusoidal-like) lines show the evolution of the true (secular, long- and short-period) and short-period averaged (secular and long-period) dynamics. Image taken from [3].

It is useful to distinguish between osculating and mean orbital dynamics. The osculating dynamics are defined by the instantaneous position and velocity vectors and are therefore associated with SP methods. The osculating orbit is the true time-varying orbit, including both secular, long- and short-periodic effects. In contrast, mean dynamics, implemented in both GP and Semianalytical methods, are averaged over some selected time, thus excluding short-periodic effects. Depending on the average definition, these methods may even exclude long-periodic effects. The resulting mean orbit has smoothly-varying orbital elements.

In the thesis, both Cowell (osculating) and Semianalytical (mean) formulations are developed, modeled in Sections 3.1.1 and 3.1.2, respectively.

### 3.1.1 Osculating Dynamics

Figure 2.5 reveals that the most important perturbations affecting LEOs, which are the target orbits for this thesis, are the Earth's non-spherical and asymmetric gravitational field and drag. In view of this, *Cowell's Equation* (2.14) as adopted in this work reduces to

<sup>2</sup>The original development of DSST was provided in NASA's R&D Goddard Trajectory Determination System software. Recently, it was implemented in the open-source Orbit Extrapolation Kit (Orekit) Java flight dynamics library, available online at (last access on 12th December 2020) <https://www.orekit.org>.

$$\frac{d^2 \mathbf{r}_i(t)}{dt^2} = \frac{d^2 \mathbf{r}_i(t)}{dt^2}_{\text{Earth grav.}} + \frac{d^2 \mathbf{r}_i(t)}{dt^2}_{\text{drag}} = \mathbf{a}_i \text{ Earth grav.}(t) + \mathbf{a}_i \text{ drag}(t), \quad (3.1)$$

where the two-body acceleration  $\frac{d^2 \mathbf{r}_i}{dt^2}_{\text{2-body}}$  was replaced with the more general acceleration due to the Earth's asymmetric and non-spherical gravitational field, i.e.,  $\frac{d^2 \mathbf{r}_i}{dt^2}_{\text{Earth grav.}}$ .

Using the orbital state vector  $\mathbf{x}$ , the dynamical model governing the time evolution (propagation) of the system is given by

$$\dot{\mathbf{x}}(t) = \mathbf{f}(\mathbf{x}(t), t) + \mathbf{w}(t) = \begin{bmatrix} \mathbf{v}_i(t) \\ \mathbf{a}_i \text{ Earth grav.}(t) + \mathbf{a}_i \text{ drag}(t) \end{bmatrix} + \mathbf{w}(t). \quad (3.2)$$

It is emphasized that, although the acceleration is expressed in inertial components, the satellite sees the "true" orientation of the Earth over time. Therefore the gravitational terms should be evaluated in Earth-rotating coordinates and then rotated back to the ECI frame before integrating [3]. The remainder of this sections develops explicit models for the accelerations.

## Earth's Gravitational Field

The gravitational potential can be expressed in functional form as  $U = U(\mathbf{r}_f, \theta)$ , where  $\theta$  is a vector of model parameters. The spatial gradient of  $U$  with respect to  $\mathbf{r}_f$  yields [20]

$$\mathbf{a}_f \text{ Earth grav.} = \left[ \frac{\partial U(\mathbf{r}_f, \theta)}{\partial \mathbf{r}_f} \right]^T, \quad (3.3)$$

It is stressed that, when evaluating the Earth's gravity field using the expression above, one obtains the inertial acceleration in an inertial frame aligned with the instantaneous Earth-fixed frame, and not the acceleration in the non-inertial Earth-fixed frame [20, 55]. Thus, only a rotation is needed to obtain the acceleration in the ECI frame, i.e.,  $\mathbf{a}_i \text{ Earth grav.} = \mathbf{T}_f^i \mathbf{a}_f \text{ Earth grav.}$

Taking into account the Earth's non-uniform mass distribution and non-spherical shape and symmetry, it is possible to express the gravitational potential  $U$  as a spherical harmonic series [3, 56]

$$U = \frac{\mu_\oplus}{r} \left[ 1 + \sum_{n=2}^{N_n} C_{n,0} \left( \frac{a_e}{r} \right)^n P_{n,0}(\sin \phi_{gc}) + \sum_{n=2}^{N_n} \sum_{m=1}^{\min(n, N_m)} \left( \frac{a_e}{r} \right)^n P_{n,m}(\sin \phi_{gc}) [C_{n,m} \cos(m\lambda) + S_{n,m} \sin(m\lambda)] \right], \quad (3.4)$$

where  $U$  is expressed as a function of the orbiter's ECEF geocentric coordinates  $(r, \phi_{gc}, \lambda)$  (see Section 2.2.1). Each harmonic (term in the series) is specified by its degree and order  $(n, m)$ . In theory the series extends to infinity, however in computational applications it should be truncated at a certain finite degree  $N_n$  and order  $N_m$ .  $a_e$  is a reference radius (usually taken as the equatorial radius  $R_\oplus$ ).  $C_{n,m}$  and  $S_{n,m}$  are the gravitational coefficients. Finally  $\min(\cdot)$  returns the smallest of its arguments and  $P_{n,m}(u)$

are the Associated Legendre Functions (ALFs) of degree  $n$  and order  $m$ , defined as

$$P_{n,m}(u) = \frac{1}{2^n n!} (1 - u^2)^{m/2} \frac{d^{n+m}}{du^{n+m}} (u^2 - 1)^n. \quad (3.5)$$

The gravitational coefficients  $C_{n,m}$  and  $S_{n,m}$  are typically determined experimentally, and express the model dependence on the Earth's internal mass distribution. Depending on the degree and order, each harmonic is classified as zonal ( $m = 0$ ), sectoral ( $n = m$ ) or tesseral ( $n \neq m \neq 0$ ). Figure 3.2 illustrates representative examples of each type. These harmonics can be interpreted as eigenfunctions that constitute an independent basis for the gravitational model.

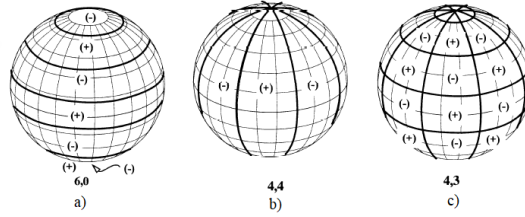


Figure 3.2: Visualization of spherical harmonics. a) represents the zonal (6, 0), b) the sectoral (4, 4), and c) the tesseral (4, 3). Zonals define potential dependence with latitudinal stripes, sectorals with longitudinal stripes and tesserals model specific regions on the Earth (latitudinal and longitudinal dependent). Source [3].

Coefficients  $C_{n,m}$  and  $S_{n,m}$  become very small as the harmonics' degree and order get large, which usually leads to truncation errors in computational applications. Frequently, normalized coefficients, denoted as  $\bar{C}_{n,m}$  and  $\bar{S}_{n,m}$ , are provided instead. These are obtained dividing the standard coefficients by a normalization factor,  $\bar{C}_{n,m} = C_{n,m}/N_{n,m}$  and  $\bar{S}_{n,m} = S_{n,m}/N_{n,m}$ , which is defined as

$$N_{n,m} = \sqrt{\frac{(n-m)! (2n+1) (2-\delta_{0,m})}{(n+m)!}}. \quad (3.6)$$

In order to complete the normalization, the ALFs must be normalized with the inverse normalization factor, i.e.,  $\bar{P}(u)_{n,m} = P_{n,m}(u)N_{n,m}$ . In (3.4) one may simply substitute the unnormalized quantities with the normalized ones, such that the products in each harmonic remain the same.

There exist some gravitational models that provide normalized coefficients  $\bar{C}_{n,m}$  and  $\bar{S}_{n,m}$ . The present work uses EGM96<sup>3</sup>, published by the Office of Geomatics at National Geospatial-Intelligence Agency in 1996. It implements the series up to  $N_n = N_m = 360$ .

Since  $U$  is modeled in spherical coordinates, the gravitational acceleration is given by (using the chain rule) [3]

$$\mathbf{a}_{f \text{ Earth grav.}} = \frac{\partial U}{\partial r} \left( \frac{\partial \mathbf{r}}{\partial \mathbf{r}_f} \right)^T + \frac{\partial U}{\partial \phi_{gc}} \left( \frac{\partial \phi_{gc}}{\partial \mathbf{r}_f} \right)^T + \frac{\partial U}{\partial \lambda} \left( \frac{\partial \lambda}{\partial \mathbf{r}_f} \right)^T. \quad (3.7)$$

<sup>3</sup>The EGM96 model coefficients are available at <https://earth-info.nga.mil/GandG/wgs84/gravitymod/egm96/egm96.html> (last access on 12th December 2020).

Finally, the acceleration vector  $\mathbf{a}_f \text{ Earth grav.} = [a_{x_f}, a_{y_f}, a_{z_f}]^T$  may be computed as [3]

$$a_{x_f} = \left[ \frac{1}{r} \frac{\partial U}{\partial r} - \frac{z_f}{r^2 \sqrt{x_f^2 + y_f^2}} \frac{\partial U}{\partial \phi_{gc}} \right] x_f - \left[ \frac{1}{x_f^2 + y_f^2} \frac{\partial U}{\partial \lambda} \right] y_f - \frac{\mu_{\oplus} x_f}{r^3} \quad (3.8a)$$

$$a_{y_f} = \left[ \frac{1}{r} \frac{\partial U}{\partial r} - \frac{z_f}{r^2 \sqrt{x_f^2 + y_f^2}} \frac{\partial U}{\partial \phi_{gc}} \right] y_f + \left[ \frac{1}{x_f^2 + y_f^2} \frac{\partial U}{\partial \lambda} \right] x_f - \frac{\mu_{\oplus} y_f}{r^3} \quad (3.8b)$$

$$a_{z_f} = \frac{1}{r} \frac{\partial U}{\partial r} z_f + \frac{\sqrt{x_f^2 + y_f^2}}{r^2} \frac{\partial U}{\partial \phi_{gc}} - \frac{\mu_{\oplus} z_f}{r^3}. \quad (3.8c)$$

Computational aspects of this equation (recursive algorithms for the ALFs and partials  $\partial U / \partial r$ ,  $\partial U / \partial \phi_{gc}$  and  $\partial U / \partial \lambda$ ) are discussed in Appendix C.

### Atmospheric Drag

Atmospheric drag is also a strong influence on the motion of LEO satellites [20]. The general equation for aerodynamic drag is

$$\mathbf{a}_{i \text{ drag}} = -\frac{1}{2} \frac{C_D A}{m} \rho \|\mathbf{v}_{\text{rel.}}\| \mathbf{v}_{\text{rel.}}, \quad (3.9)$$

where the coefficient of drag  $C_D$  is a dimensionless quantity that reflects the satellite's susceptibility to drag forces.  $A$  and  $m$  are, respectively, the cross-sectional area and mass of the satellite and  $\rho$  is the atmospheric density.  $\mathbf{v}_{\text{rel.}}$  is the velocity of the satellite relative to the local surrounding atmosphere. A reasonable approximation is to assume that the atmosphere co-rotates with the Earth [20], resulting in  $\mathbf{v}_{\text{rel.}} = \mathbf{v}_i - \boldsymbol{\omega}_{\oplus} \times \mathbf{r}_i$ .

There exist different models for atmospheric density. In this thesis the *Exponential Model* is used. There exist more complex models that take into consideration time-variant fluctuations of the atmospheric density, namely, due to solar or magnetic field activity, such as the Jacchia 1971 or the Harris-Priester models [3]. The *Exponential Model* postulates an exponential relationship between satellite's altitude  $h_{\text{ellp}}$  and density, that is,

$$\rho = \rho_0 \exp \left( -\frac{h_{\text{ellp}} - h_0}{H} \right), \quad (3.10)$$

where the coefficients  $\rho_0$ ,  $h_0$  and  $H$  are given in the table of Figure 3.3 as a function of the satellite's altitude.

### 3.1.2 Mean Dynamics

The underlying idea of Semianalytical methods is to separate short-term from long-term periodic and secular effects in the EoM, obtaining the mean dynamics, which can then be numerically integrated with much larger integration step sizes, when compared to SP Cowell propagation. If needed, the short-periodic effects can then be recovered by building the osculating state. This procedure belongs to a

Altitude $h_{ellp}$ (km)	Base Altitude $h_o$ (km)	Nominal Density $\rho_o$ (kg/m <sup>3</sup> )	Scale Height $H$ (km)	Altitude $h_{ellp}$ (km)	Base Altitude $h_o$ (km)	Nominal Density $\rho_o$ (kg/m <sup>3</sup> )	Scale Height $H$ (km)
0–25	0	1.225	7.249	150–180	150	$2.070 \times 10^{-9}$	22.523
25–30	25	$3.899 \times 10^{-2}$	6.349	180–200	180	$5.464 \times 10^{-10}$	29.740
30–40	30	$1.774 \times 10^{-2}$	6.682	200–250	200	$2.789 \times 10^{-10}$	37.105
40–50	40	$3.972 \times 10^{-3}$	7.554	250–300	250	$7.248 \times 10^{-11}$	45.546
50–60	50	$1.057 \times 10^{-3}$	8.382	300–350	300	$2.418 \times 10^{-11}$	53.628
60–70	60	$3.206 \times 10^{-4}$	7.714	350–400	350	$9.158 \times 10^{-12}$	53.298
70–80	70	$8.770 \times 10^{-5}$	6.549	400–450	400	$3.725 \times 10^{-12}$	58.515
80–90	80	$1.905 \times 10^{-5}$	5.799	450–500	450	$1.585 \times 10^{-12}$	60.828
90–100	90	$3.396 \times 10^{-6}$	5.382	500–600	500	$6.967 \times 10^{-13}$	63.822
100–110	100	$5.297 \times 10^{-7}$	5.877	600–700	600	$1.454 \times 10^{-13}$	71.835
110–120	110	$9.661 \times 10^{-8}$	7.263	700–800	700	$3.614 \times 10^{-14}$	88.667
120–130	120	$2.438 \times 10^{-8}$	9.473	800–900	800	$1.170 \times 10^{-14}$	124.64
130–140	130	$8.484 \times 10^{-9}$	12.636	900–1000	900	$5.245 \times 10^{-15}$	181.05
140–150	140	$3.845 \times 10^{-9}$	16.149	1000–	1000	$3.019 \times 10^{-15}$	268.00

Figure 3.3: Atmospheric Exponential Model parameters as a function of the satellite's altitude  $h_{ellp} = \|r\| - R_\oplus$ . Source [3].

broad field of averaging theory of dynamical systems called the Generalized Method of Averaging [57]. The theoretical framework that supports the Semianalytical propagation model implemented in this work is the same that supports other Semianalytical models. However, it diverges from traditional approaches, for instance DSST [27], in the computer implementation step — instead of relying on analytical expansions of the perturbing accelerations, the current model, proposed by Todd Ely in [28, 29], relies on numerical methods.

Before presenting the Semianalytical model, it is important to introduce new notation for clarity. Following the aforementioned references, the mean orbital dynamics are modeled using the equinoctial element set  $\mathcal{E}$ . Let  $\mathcal{E} = [\alpha, \lambda]^T$ , where vector  $\alpha = [a, h, k, p, q]$  denotes the slowly-varying equinoctial elements that indicate the orbit's shape and orientation, and  $\lambda$  denotes the fast variable that locates the satellite in the orbit. Furthermore, the osculating elements,  $\mathcal{E}$ , are distinguished from the mean elements,  $\bar{\mathcal{E}}$ , with an overbar. Moreover,  $\mathcal{E}_i$  denotes a generic equinoctial element, indexed with  $i = 1, \dots, 6$ , i.e.,  $\mathcal{E} = [\mathcal{E}_1, \mathcal{E}_2, \mathcal{E}_3, \mathcal{E}_4, \mathcal{E}_5, \mathcal{E}_6]^T$ .

The osculating equinoctial EoM are obtained by converting the cartesian EoM, (2.14), to equinoctial form. The Gaussian Variation-of-Parameters (VOP) formulation is given by [27]

$$\dot{\mathcal{E}}_i = \frac{d\mathcal{E}_i}{dt} = n(a)\delta_{i6} + \frac{d\mathcal{E}_i}{dv} \sum_{p \in P} \mathbf{q}_p, \quad i = 1, \dots, 6. \quad (3.11)$$

where  $\sum_{p \in P} \mathbf{q}_p$  is the vector sum of all perturbing inertial accelerations  $\mathbf{q}_p$ ,  $v$  is the inertial osculating satellite velocity and  $\delta_{i6}$  is the *Kronecker delta*. The perturbations to be considered are, as in Section 3.1.1, due to the Earth's gravitational field and drag, therefore the set of active perturbations is  $P = \{Z, T, D\}$ , where  $Z$  denotes the zonal harmonics,  $T$  the tesseral/sectoral harmonics and  $D$  drag. In the following, all vector quantities are inertial and expressed in ECI coordinates, therefore the subscript frame notation is dropped for convenience and should not be confused with the equinoctial element indexing. Partial derivatives  $\partial \mathcal{E} / \partial v$  are presented in Appendix D (Section D.3).  $n(a) = \sqrt{\mu_\oplus / a^3}$  is the osculating mean motion.

## Equations of Averaging

The mean EoM are found by averaging (3.11) with respect to the system's fast variables. While zonal and drag perturbations only need averaging with respect to  $\lambda$ , averaging the tesseral/sectoral acceleration requires special treatment, since these perturbations are also dependent on the Earth's orientation relative to the spacecraft. This dependence is usually expressed via the sidereal angle  $\theta = \theta_{GST}$  (introduced in Section 2.2.1). In the Earth's system, this is generally a fast dependence.

Let the osculating VOP (3.11) be rewritten as

$$\frac{d\mathcal{E}_i}{dt} = n(a)\delta_{i6} + \epsilon F_i(\boldsymbol{\alpha}, \lambda, \theta), \quad i = 1, \dots, 6, \quad (3.12)$$

where  $\epsilon F_i$  are the osculating element rates of change due to the perturbing accelerations and include all terms appearing on the last term of the right-hand side of (3.11).  $\epsilon$  is a small variational parameter that plays a crucial in deriving the mean EoM.

It is now assumed that the averaged EoM take the following VOP form

$$\frac{d\bar{\mathcal{E}}_i}{dt} = n(\bar{a})\delta_{i6} + \sum_{j=1}^{\infty} \epsilon^j A_i^j(\bar{\boldsymbol{\alpha}}), \quad i = 1, \dots, 6. \quad (3.13)$$

Functions  $\epsilon^j A_i^j$  give the mean element rates of change<sup>4</sup>, due to the perturbing forces, as a function of the slowly-varying mean elements, being independent on the fast variables.

SST further assumes a *near-identity transformation* between osculating and mean elements, that is,

$$\mathcal{E}_i = \bar{\mathcal{E}}_i + \sum_{j=1}^{\infty} \epsilon^j \eta_i^j(\bar{\boldsymbol{\alpha}}, \bar{\lambda}, \theta), \quad i = 1, \dots, 6, \quad (3.14)$$

where  $\epsilon^j \eta_i^j$  represents a small short-periodic variation of order  $j$  on element  $i$ . The short-periodic variations are assumed to contain all of the high-frequency components of the osculating elements  $\mathcal{E}_i$ , so that the mean elements  $\bar{\mathcal{E}}_i$  contain only secular and long-periodic variations.

The problem is now to relate the as yet undefined functions  $A_i^j$  and  $\eta_i^j$  with the known osculating  $F_i$  functions. The Semianalytical model implemented in this work truncates both Equations (3.13) and (3.14) to first order on  $\epsilon$ . The first-order Equations of Averaging of SST are modeled as

$$\mathcal{O}(\epsilon^1): \quad A_i^1(\bar{\boldsymbol{\alpha}}) + \frac{\partial \eta_i^1}{\partial \lambda} n(\bar{a}) + \frac{\partial \eta_i^1}{\partial \theta} \omega_{\oplus} = F_i(\bar{\boldsymbol{\alpha}}, \bar{\lambda}, \theta) - \frac{3}{2} \frac{\eta_1^1}{\bar{a}} n(\bar{a}) \delta_{i6}, \quad i = 1, \dots, 6, \quad (3.15)$$

where  $\omega_{\oplus} = \frac{d\theta}{dt}$  (the sidereal rate is the Earth's angular speed)<sup>5</sup>. It is emphasized that the dependence with  $\theta$  is dropped when treating zonal and drag perturbations. Proof of (3.15) may be found in [27]. A brief review of the derivation is provided Appendix E.

These equations relate the unknown functions  $A_i^1$  and the partial derivatives of  $\eta_i^1$  with the known  $F_i$  functions. However, the dependence with the fast variables,  $\bar{\lambda}$  and  $\theta$ , is still present. To eliminate it, the

<sup>4</sup>Notice that the superscript  $j$  in  $\epsilon^j$  designates a power and in  $A_i^j$  designates an index.

<sup>5</sup>Since  $\omega_{\oplus}$  is considered constant,  $\theta$  varies linearly with time, which means that  $\theta$  does not have short-periodic behavior, i.e.,  $\bar{\theta} = \theta$ .

equations needs to be averaged, with an averaging operator to be defined next.

### Averaging Operator

The averaging operator, denoted by  $\langle \cdot \rangle$ , is required to have the following properties

$$\langle \rho f + \sigma g \rangle = \rho \langle f \rangle + \sigma \langle g \rangle, \quad (3.16a)$$

$$\langle \langle f \rangle \rangle = \langle f \rangle, \quad (3.16b)$$

$$\langle \eta_i^j \rangle = 0, \quad (3.16c)$$

where  $\rho$  and  $\sigma$  are real numbers, and  $f$  and  $g$  are real continuous functions. The last condition ensures that the average of short-periodic functions is zero. Proofs of these properties can be found in [27].

For  $p \in \{Z, D\}$ , the osculating rate functions<sup>6</sup>  $F_{i,p}$  are small,  $2\pi$ -periodic in  $\bar{\lambda}$  and slowly-varying in time when the orbital elements are held fixed. Hence, the *single-averaging* operator is able to eliminate the dependence of  $F_{i,p}$  on  $\bar{\lambda}$ . It is defined as

$$\langle F_{i,p} \rangle (\bar{\alpha}) = \frac{1}{2\pi} \int_{\bar{\lambda}}^{\bar{\lambda}+2\pi} F_{i,p}(\bar{\alpha}, \xi) d\xi. \quad (3.17)$$

For  $p \in \{T\}$ ,  $F_{i,p}$  is small,  $2\pi$ -periodic in both  $\bar{\lambda}$  and  $\theta$  and slowly-varying in time when  $\theta$  and the orbital elements are held fixed. Hence, the *double-averaging* operator<sup>7</sup> eliminates the dependence on  $\bar{\lambda}$  and  $\theta$

$$\langle F_{i,p} \rangle (\bar{\alpha}) = \frac{1}{4\pi^2} \int_{\theta}^{\theta+2\pi} \int_{\bar{\lambda}}^{\bar{\lambda}+2\pi} F_{i,p}(\bar{\alpha}, \xi, \psi) d\xi d\psi. \quad (3.18)$$

For each perturbation  $p$ , (3.15) can be solved for the mean element rates by averaging both sides of the equation with (3.17) or (3.18). Since the functions  $\eta_i^1$  are  $2\pi$ -periodic and have zero mean, cf. (3.16c), they average out (as well as their partial derivatives) yielding, to first order:

$$A_{i,Z,D}^1(\bar{\alpha}) = \langle F_{i,Z,D}(\bar{\alpha}, \bar{\lambda}) \rangle, \quad (3.19a)$$

$$A_{i,T}^1(\bar{\alpha}) = \langle F_{i,T}(\bar{\alpha}, \bar{\lambda}, \theta) \rangle. \quad (3.19b)$$

Next, computational methods are described to compute  $A_{i,Z,D}^1$ ,  $A_{i,T}^1$ ,  $\eta_{i,Z,D}^1$  and  $\eta_{i,T}^1$ , cf. [28, 29].

### Zonal and Drag Perturbations

Both zonal and drag perturbations are treated in similar fashion, since both are averaged with the *single-averaging* operator, therefore they may be added up before averaging. The resultant perturbing osculating functions are then

$$F_{i,Z,D}(\bar{\alpha}, \bar{\lambda}) = F_{i,Z} + F_{i,D} = \frac{\partial \bar{\mathcal{G}}_i}{\partial \bar{\mathbf{v}}}(\mathbf{q}_Z + \mathbf{q}_D), \quad i = 1, \dots, 6. \quad (3.20)$$

<sup>6</sup> $F_{i,p}$  denotes the perturbing function  $F_i$  when only the perturbation  $p$  is considered in (3.11).

<sup>7</sup>Notice that if  $F_{i,p}$  is independent of  $\theta$ , the *double-averaging* operator reduces to the *single-averaging* one. Thus, without inconsistencies, one applies (3.17) to perturbations that do not depend on  $\theta$  and (3.18) to those that do depend.



The zonal harmonics acceleration  $q_Z$  is given by  $a_i$  Earth grav., however only the zonal terms of the gravitational series are considered (see Section 3.1.1).  $q_D$  is given by (3.9). Both the partials  $\partial \bar{\mathcal{E}}_i / \partial \bar{v}$  and the accelerations are to be evaluated with the available mean state at epoch.

### Computing $A_{iZ,D}^1$

According to (3.19a), the mean element rates  $A_{iZ,D}^1$  are obtained by integrating (averaging) the functions  $F_{iZ,D}$  with (3.17). Given the mean elements  $\bar{\mathcal{E}}(t)$  at a certain epoch  $t$ , the integration is performed in  $\bar{\lambda}$  from  $\bar{\lambda}(t)$  to  $\bar{\lambda}(t) + 2\pi$  (a single orbital revolution), while keeping the other elements  $\bar{\alpha}(t)$  fixed at the given epoch. The integral is computed numerically using a classic Gaussian fixed-order numerical quadrature technique [28]. To do so, the present work makes use of the SciPy package. The integration produces the mean element rates  $A_{iZ,D}^1(t)$  at the given epoch  $t$ .

### Computing $\eta_{iZ,D}^1$

Substituting (3.19a) in (3.15), and separating the slowly-varying mean elements  $\bar{\alpha}$  from  $\bar{\lambda} = \bar{\mathcal{E}}_6$ , one obtains the homological Partial Differential Equations (PDEs):

$$\langle F_{iZ,D} \rangle (\bar{\alpha}) + \frac{\partial \eta_{iZ,D}^1}{\partial \bar{\lambda}} n(\bar{a}) = F_{iZ,D}(\bar{\alpha}, \bar{\lambda}), \quad i = 1, \dots, 5, \quad (3.21a)$$

$$\langle F_{6Z,D} \rangle (\bar{\alpha}) + \frac{\partial \eta_{6Z,D}^1}{\partial \bar{\lambda}} n(\bar{a}) = F_{6Z,D}(\bar{\alpha}, \bar{\lambda}) - \frac{3}{2} \frac{\eta_{1Z,D}^1}{\bar{a}} n(\bar{a}). \quad (3.21b)$$

To solve these equations, it is assumed that the functions  $F_{iZ,D}(\bar{\alpha}, \bar{\lambda})$  can be expanded as a Fourier Series (FS) (2.38) in  $\bar{\lambda}$ , that is,

$$F_{iZ,D}(\bar{\alpha}, \bar{\lambda}) = \sum_{k=-\infty}^{\infty} \mathcal{F}_{iZ,D}^k(\bar{\alpha}) e^{jk\bar{\lambda}} = \langle F_{iZ,D} \rangle (\bar{\alpha}) + \sum_{\substack{k=-\infty \\ k \neq 0}}^{\infty} \mathcal{F}_{iZ,D}^k(\bar{\alpha}) e^{jk\bar{\lambda}}, \quad i = 1, \dots, 6, \quad (3.22)$$

where  $\mathcal{F}_{iZ,D}^k$  is the Fourier coefficient of order  $k$  on element  $i$ . Substituting (3.22) into (3.21) and integrating yields the following solution of the PDEs:

$$\eta_{iZ,D}^1(\bar{\alpha}, \bar{\lambda}) = \frac{1}{n(\bar{a})} \sum_{\substack{k=-\infty \\ k \neq 0}}^{\infty} \frac{\mathcal{F}_{iZ,D}^k(\bar{\alpha})}{jk} e^{jk\bar{\lambda}} + C_i(\bar{\alpha}), \quad i = 1, \dots, 5, \quad (3.23a)$$

$$\eta_{6Z,D}^1(\bar{\alpha}, \bar{\lambda}) = \frac{1}{n(\bar{a})} \sum_{\substack{k=-\infty \\ k \neq 0}}^{\infty} \left[ \frac{\mathcal{F}_{6Z,D}^k(\bar{\alpha})}{jk} + \frac{3}{2\bar{a}} \frac{\mathcal{F}_{1Z,D}^k(\bar{\alpha})}{k^2} \right] e^{jk\bar{\lambda}} + C_6(\bar{\alpha}), \quad (3.23b)$$

where  $C_i(\bar{\alpha})$  are the integration constants to be determined. Applying the average operator to both sides of (3.23) and performing the necessary simplifications yields  $C_i(\bar{\alpha}) = 0$ .

Contrary to other SSTs, where the Fourier coefficients are found analytically, in this work they are found numerically using DFT (introduced in Section 2.6.2), cf. [28]. The DFT of  $F_{iZ,D}$  is

$$X_{i_{Z,D}}^k = \sum_{n=0}^{N-1} F_{i_{Z,D}}(\bar{\alpha}, \bar{\lambda}_n) e^{-j \frac{2\pi}{N} kn}, \quad k = 0, \dots, N-1, \quad (3.24)$$

where  $N$  is the DFT length. Furthermore, functions  $F_{i_{Z,D}}$  are sampled with  $\bar{\lambda}_n = \bar{\lambda} + n \frac{2\pi}{N}$ , for  $n = 0, \dots, N-1$ . Notice that the sampling index  $n$  should not be confused with the mean motion  $n(\bar{a})$ . Referring to Section 2.6.3, the FS and DFT coefficients are related by

$$\mathcal{F}_{i_{Z,D}}^k = \frac{e^{-jk\bar{\lambda}} X_{i_{Z,D}}^k}{N}, \quad k = -\frac{N}{2}, \dots, 0, \dots, \frac{N}{2} - 1. \quad (3.25)$$

The frequency indices  $k$  have been limited to  $k < N/2$ , to avoid *aliasing* effects, that could otherwise corrupt the spectrum (as discussed in Section 2.6.3). On the other hand, it is still possible to recover the frequency information of the coefficients with greater  $k$ , by extending the range of indices  $k$  to the negative side up to  $-N/2$ . Indeed, the periodicity and symmetry properties of the DFT of real valued signals yield  $X_i^{-k} = X_i^{N-k}$  [52], which is useful for relating, with reduced *aliasing*, the results returned by the numerical DFT routines with the desired FS coefficients.

In the present study, the DFT coefficients are found efficiently using Fast Fourier Transform (FFT) algorithms, provided by the SciPy package. Given the mean elements  $\bar{\mathcal{G}}(t)$  at a certain epoch  $t$ , the FFT is provided a set of numerical samples of the osculating functions  $F_{i_{Z,D}}$ , evaluated with the current mean elements  $\bar{\alpha}(t)$  and sampled with mean mean longitude  $\bar{\lambda}_n = \bar{\lambda}(t) + n \frac{2\pi}{N}$  for  $n = 0, \dots, N-1$ .

### Tesseral/Sectoral Perturbations

The osculating perturbing functions due to tesseral/sectoral perturbations is given by (evaluated with mean elements)

$$F_{i_T}(\bar{\alpha}, \bar{\lambda}, \theta) = \frac{\partial \bar{\mathcal{G}}_i}{\partial \bar{v}} \mathbf{q}_T, \quad i = 1, \dots, 6. \quad (3.26)$$

$\mathbf{q}_Z$  is given by  $\mathbf{a}_i$  Earth grav., however only the tesseral/sectoral terms of the gravitational series are considered. The dependence between the sidereal angle  $\theta$  and mean mean longitude  $\bar{\lambda}$  is outlined, as shown in [27, 29], by decomposing  $F_{i_T}(\bar{\alpha}, \bar{\lambda}, \theta)$  as a 2-D FS of  $\bar{\lambda}$  and  $\theta$

$$F_{i_T}(\bar{\alpha}, \bar{\lambda}, \theta) = \sum_{k=-\infty}^{\infty} \sum_{\substack{m=-N_m \\ m \neq 0}}^{N_m} \mathcal{F}_{i_T}^{k,m}(\bar{\alpha}) e^{j(k\bar{\lambda} - m\theta)}, \quad i = 1, \dots, 6. \quad (3.27)$$

where  $m$  is the frequency index associated with the variable  $\theta$ , having the same meaning as in the gravitational potential series (3.4), that is, it represents the order of the harmonics [27]. Since the current analysis only considers tesseral/sectoral harmonics, zonal harmonics ( $m = 0$ ) are removed explicitly. Furthermore,  $N_m$  is the maximum order of the gravitational field under consideration.

Examination of (3.27) reveals the possibility of three perturbation problems, given the behavior of the two competing angles  $\lambda$  and  $\theta$ , with frequencies  $\dot{\lambda}$  and  $\dot{\theta}$ , respectively, [29]:

- **Resonant case** - Resonance is the appearance of otherwise unusual long-periodic contributions to the motion caused by the satellite's exposure to the Earth in a repeating orientation [3], due to

the commensurability of the two frequencies  $\dot{\theta}$  and  $\dot{\lambda}$ . It appears when there exists a rational ratio of two integers  $Q$  and  $P$  that satisfy (to a certain level of approximation) the relation  $\dot{\lambda}/\dot{\theta} \sim Q/P$ ;

- **Non-resonant case** - In this case the resonance condition is not met since the integers  $P$  and  $Q$  are too large to be significant, that is, the two frequencies are not commensurate;
- **Adiabatic case** - In this case  $\dot{\lambda}$  is much larger than  $\dot{\theta}$  and therefore the fast dependence with  $\theta$  can be dropped. While this is not the case for Earth, it is for Venus, since the sidereal period for this planet is  $\sim 243$  days.

In the present work, only the non-resonant case will be examined and employed. The addition of resonant harmonics in the Semianalytical propagation model is left as future work.

### Computing $A_{i_T}^1$

The mean element rates  $A_{i_T}^1$  are obtained by applying the *double-averaging* operator to the osculating functions  $F_{i_T}$ . However, by substituting the latter with the associated FS of Equation (3.27) and applying the averaging operator, one obtains:

$$\begin{aligned} A_{i_T}^1 &= \langle F_{i_T}(\bar{\alpha}, \bar{\lambda}, \theta) \rangle = \frac{1}{4\pi^2} \int_{\theta}^{\theta+2\pi} \int_{\bar{\lambda}}^{\bar{\lambda}+2\pi} \sum_{k=-\infty}^{\infty} \sum_{\substack{m=-N_m \\ m \neq 0}}^{N_m} \mathcal{F}_{i_T}^{k,m}(\bar{\alpha}) e^{j(k\xi - m\psi)} d\xi d\psi \\ &= \frac{1}{4\pi^2} \sum_{k=-\infty}^{\infty} \sum_{\substack{m=-N_m \\ m \neq 0}}^{N_m} \mathcal{F}_{i_T}^{k,m}(\bar{\alpha}) \left[ \frac{e^{-jm\psi} e^{jk\xi}}{mk} \right]_{\psi=\theta, \xi=\bar{\lambda}}^{\psi=\theta+2\pi, \xi=\bar{\lambda}+2\pi} \\ &= 0. \end{aligned} \quad (3.28)$$

This means that the non-resonant tesseral/sectoral harmonics yield no long-periodic or secular effects on the orbit. In fact, these harmonics only contribute to the short-periodic functions and may therefore be explicitly excluded from the mean element propagation.

### Computing $\eta_{i_T}^1$

By substituting (3.27) and  $A_{i_T}^1 = 0$  in (3.15), one obtains the homological PDEs for the current analysis:

$$n(\bar{a}) \frac{\partial \eta_{i_T}^1}{\partial \bar{\lambda}} + \omega_{\oplus} \frac{\partial \eta_{i_T}^1}{\partial \theta} = \sum_{k=-\infty}^{\infty} \sum_{\substack{m=-N_m \\ m \neq 0}}^{N_m} \mathcal{F}_{i_T}^{k,m}(\bar{\alpha}) e^{j(k\bar{\lambda} - m\theta)}, \quad i = 1, \dots, 5, \quad (3.29a)$$

$$n(\bar{a}) \frac{\partial \eta_{6_T}^1}{\partial \bar{\lambda}} + \omega_{\oplus} \frac{\partial \eta_{6_T}^1}{\partial \theta} = \sum_{k=-\infty}^{\infty} \sum_{\substack{m=-N_m \\ m \neq 0}}^{N_m} \mathcal{F}_{6_T}^{k,m}(\bar{\alpha}) e^{j(k\bar{\lambda} - m\theta)} - \frac{3n(\bar{a})}{2\bar{a}} \eta_{1_T}^1. \quad (3.29b)$$

Solving these PDEs yields:

$$\eta_{i_T}^1(\bar{\alpha}, \bar{\lambda}, \theta) = \sum_{k=-\infty}^{\infty} \sum_{\substack{m=-N_m \\ m \neq 0}}^{N_m} \frac{\mathcal{F}_{i_T}^{k,m}(\bar{\alpha})}{j(kn(\bar{a}) - m\omega_{\oplus})} e^{j(k\bar{\lambda} - m\theta)}, \quad i = 1, \dots, 5, \quad (3.30a)$$

$$\eta_{6T}^1(\bar{\alpha}, \bar{\lambda}, \theta) = \sum_{k=-\infty}^{\infty} \sum_{\substack{m=-N_m \\ m \neq 0}}^{N_m} \left( \frac{\mathcal{F}_{6T}^{k,m}(\bar{\alpha})}{j(kn(\bar{a}) - m\omega_{\oplus})} + \frac{3n(\bar{a})}{2\bar{a}} \frac{\mathcal{F}_{1T}^{k,m}(\bar{\alpha})}{(kn(\bar{a}) - m\omega_{\oplus})^2} \right) e^{j(k\bar{\lambda} - m\theta)}. \quad (3.30b)$$

Since resonance has been excluded from the current analysis, the denominator terms  $kn(\bar{a}) - m\omega_{\oplus}$  are not close to zero, hence the series does not diverge [29].

The 2-D FS series coefficients  $\mathcal{F}_{i_T}^{k,m}$  may be approximated by the 2-D DFT coefficients, using

$$\mathcal{F}_{i_T}^{k,m} = \frac{e^{-j(k\bar{\lambda} - m\theta)}}{NM} X_{i_T}^{k,m}, \quad k = -\frac{N}{2}, \dots, 0, \dots, \frac{N}{2} - 1, m = -\frac{M}{2}, \dots, -1, 1, \dots, \frac{M}{2} - 1. \quad (3.31)$$

The DFT coefficients are found efficiently using 2-D FFT algorithms from SciPy. In each call, for a given orbital state  $\bar{\mathcal{G}}(t)$  with known  $\theta(t)$  at epoch  $t$ , the FFT is provided a grid of numerical samples of the functions  $F_{i_T}$ , computed with the given mean elements. The slowly-varying elements  $\bar{\alpha}(t)$  are held constant to the current state, the mean mean longitude is sampled with  $\bar{\lambda}_{n_1} = \bar{\lambda}(t) + n_1 \frac{2\pi}{N}$  for  $n_1 = 0, \dots, N-1$  and the sidereal angle is sampled with  $\theta_{n_2} = \theta(t) + \frac{2\pi}{M} n_2$  for  $n_2 = 0, \dots, M-1$ , given DFT lengths  $N$  and  $M$ , respectively. The FFT routine returns, for all frequencies  $k$  and  $m$ , the associated coefficients  $X_{i_T}^{k,m}$  at the given epoch  $t$ .

### Forming the Semianalytical Propagator

Now that the averaged mean rates and short-periodic functions have been found, the first-order Semianalytical propagator can be assembled. Mean element propagation is given by

$$\frac{d\bar{\mathcal{G}}_i}{dt} = A_{i_{Z,D}}^1, \quad i = 1, \dots, 5, \quad (3.32a)$$

$$\frac{d\bar{\lambda}}{dt} = n(\bar{a}) + A_{6_{Z,D}}^1. \quad (3.32b)$$

The small parameter  $\epsilon$  has been useful to derive the Equations of Averaging, however in actual applications it is not explicitly computed, but rather is implicitly considered as part of the perturbing functions  $F_i$ . Hence, without loss of generality, it has been set to one [29]. These differential equations are numerically integrated with a given integration step size  $\Delta t$ . Since short-periodic terms have been averaged out, large integration steps can be used (typically on the order of one day for a LEO satellite). The integration solver returns the mean equinoctial state at the selected steps, i.e.  $\{\bar{\mathcal{G}}(t_k) : k = 1, 2, \dots\}$ . The integration process is initialized with a known initial mean state  $\bar{\mathcal{G}}(t_0)$ .

Equations (3.32) represent the Semianalytical orbital propagation model of the mean dynamics and therefore may be written in the same functional form of Equation (2.15a),

$$\dot{\bar{\mathcal{G}}}(t) = \frac{d\bar{\mathcal{G}}}{dt} = \bar{f}(\bar{\mathcal{G}}(t), t) + \bar{w}(t), \quad (3.33)$$

where  $\bar{w}(t)$  is the process noise, that accounts for the uncertainty error in the mean state propagation. The osculating state at integration times,  $\{\mathcal{G}(t_k) : k = 1, 2, \dots\}$ , is then recovered with the first-order

*near-identity transformation*, also called the *mean-to-osculating map*,

$$\mathcal{E}_i(t) = \bar{\mathcal{E}}_i(t) + \eta_{i_{Z,D}}^1(t) + \eta_{i_T}^1(t), \quad i = 1, \dots, 5, \quad (3.34a)$$

$$\lambda(t) = \bar{\lambda}(t) + \eta_{6_{Z,D}}^1(t) + \eta_{6_T}^1(t), \quad (3.34b)$$

where

$$\eta_{i_{Z,D}}^1 = \frac{1}{n(\bar{a})N_{Z,D}} \sum_{\substack{k=-N_{Z,D}/2 \\ k \neq 0}}^{N_{Z,D}/2-1} \frac{X_{i_{Z,D}}^k}{jk}, \quad (3.35a)$$

$$\eta_{i_T}^1 = \frac{1}{N_T M_T} \sum_{k=-N_T/2}^{N_T/2-1} \sum_{\substack{m=-M_T/2 \\ m \neq 0}}^{M_T/2-1} \frac{X_{i_T}^{k,m}}{j(kn(\bar{a}) - m\omega_\oplus)}, \quad (3.35b)$$

$$\eta_{6_{Z,D}}^1 = \frac{1}{n(\bar{a})N_{Z,D}} \sum_{\substack{k=-N_{Z,D}/2 \\ k \neq 0}}^{N_{Z,D}/2-1} \left( \frac{X_{6_{Z,D}}^k}{jk} + \frac{3}{2\bar{a}} \frac{X_{1_{Z,D}}^k}{k^2} \right), \quad (3.35c)$$

$$\eta_{6_T}^1 = \frac{1}{N_T M_T} \sum_{k=-N_T/2}^{N_T/2-1} \sum_{\substack{m=-M_T/2 \\ m \neq 0}}^{M_T/2-1} \left( \frac{X_{6_T}^{k,m}}{j(kn(\bar{a}) - m\omega_\oplus)} + \frac{3n(\bar{a})}{2\bar{a}} \frac{X_{1_T}^{k,m}}{(kn(\bar{a}) - m\omega_\oplus)^2} \right). \quad (3.35d)$$

$N_{Z,D}$  is the length of the DFT associated to zonal/drag perturbations.  $N_T$  and  $M_T$  are the lengths associated to tesseral/sectoral perturbations in the sampling of  $\bar{\lambda}$  and  $\theta$ , respectively.

The process described up to here only outputs the state at the selected integration times<sup>8</sup>. To obtain the orbital state in-between integration times, one may take advantage of the fact that the mean rates and short-periodic coefficients are slowly varying and thus, may be interpolated by relatively low order interpolation formulas, according to [15, 27].

In fact, the authors of [27] suggest the use of Lagrange interpolation, with four interpolation points to, interpolate the short-periodic coefficients. Between integration steps  $t_n$  and  $t_{n+1}$ , the interpolation routine is provided with  $\{(X_{i_{Z,D}}(t_k), X_{i_T}(t_k)) : k = n-1, n, n+1, n+2\}$ , where  $X_{i_{Z,D}}(t_k)$  and  $X_{i_T}(t_k)$  are sets containing all the  $N_{Z,D} - 1$  and  $(N_T - 1) \times (M_T - 1)$  DFT coefficients at  $t_k$ . Furthermore, the mean elements are interpolated with Hermite polynomials using three interpolation points. The routine is provided with  $\{(\bar{\mathcal{E}}(t_k), d\bar{\mathcal{E}}(t_k)/dt) : k = n-1, n, n+1\}$ . In the computational work of this thesis, Lagrange and Hermite interpolations are performed with Scipy. Figure 3.4 illustrates the assembled Semianalytical propagator.

Usually, the initial state is known in osculating form,  $\mathcal{E}(t_0)$ . In those cases, in order to correctly initialize the propagator with  $\bar{\mathcal{E}}(t_0)$ , the *near-identity transformation* must be inverted. A simple epoch point conversion is employed [3]

$$\begin{aligned} \bar{\mathcal{E}}_{i_{k+1}}(t_0) &= \mathcal{E}_i(t_0) - (\eta_{i_{Z,D}}^1(\bar{\alpha}_k, \bar{\lambda}_k) + \eta_{i_T}^1(\bar{\alpha}_k, \bar{\lambda}_k, \theta(t_0))) \\ \bar{\mathcal{E}}_{i_0}(t_0) &= \mathcal{E}_i(t_0) \end{aligned}, \quad i = 1, \dots, 6, \quad k = 1, 2, \dots \quad (3.36)$$

<sup>8</sup>Note that applying (3.32) and (3.34) with a small integration step is possible, but would be very inefficient, due to the many numerical quadrature and FFT computations.

**Reconstruct the *osculating* trajectory:**

1. Interpolate  $\bar{\mathcal{E}}(t)$  to the output epoch  $t_{i,n} \rightarrow \bar{\mathcal{E}}(t_{i,n})$ ;
2. Interpolate  $X_{Z,D}(t)$  and  $X_T(t)$  to the output epoch  $t_{i,n} \rightarrow X_{Z,D}(t), X_T(t_{i,n})$ ;
3. Use the *near-identity* transformation to get the *osculating* state at epoch  $\mathcal{E}(t_{i,n})$ .

Note: apply only step 3 at the integration times  $t_{i,0}$  since the interpolated quantities are already available.

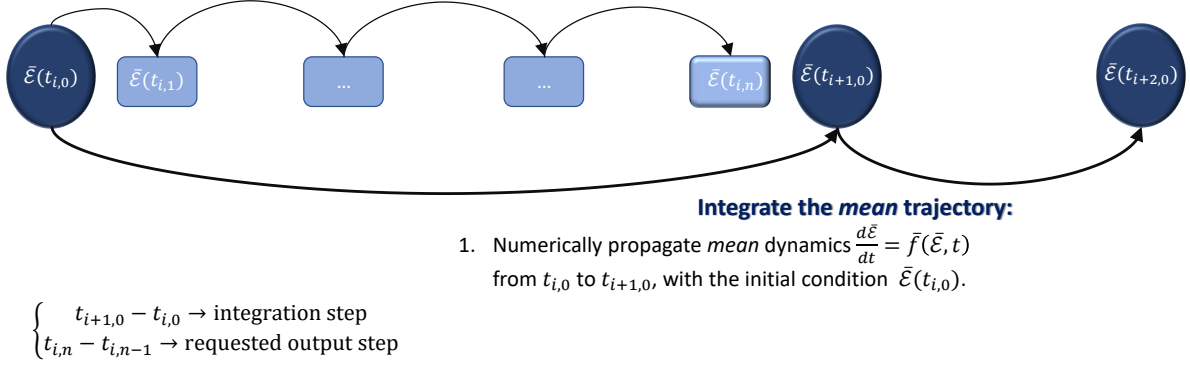


Figure 3.4: Semianalytical propagation. First, both the integration and output steps are defined. Then, the mean elements are integrated along the integration times. At each requested output time, both the mean dynamics and DFT coefficients are obtained from interpolation, and finally the *near-identity transformation* reconstructs the *osculating* equinoctial state. In the diagram, circles denote states obtained from integration, rectangles denote in-between interpolated states and the symbol  $\mathcal{E}$  denotes the equinoctial set.

The process is iterated in  $k$  until convergence (typically two or three iterations are enough [3]).

To summarize, the developed Semianalytical propagator diverges from traditional implementation of SST in the sense that: 1) the mean element rates are found with numerical quadrature, and 2) the coefficients of the Fourier Series are found via FFTs. In contrast, other Semianalytical models, such as the DSST, find these functions analytically via explicit formulae. This traditional approach requires detailed expansions and special treatment for each acceleration type, which is a difficult task that usually requires some form of truncation, typically in eccentricity, to make the problem tractable, hence limiting the range of orbits that can be considered and reducing the overall accuracy and performance of the method [28].

To conclude, the Semianalytical propagator adopted in this work is based on research proposed by Todd Ely: the FFT-based *near-identity transformation* is proposed in [29] and the numerical quadrature mean element propagation in [28].

## 3.2 Measurement Modeling

In this section satellite observations from ground stations are modeled, following [20]. These observations are processed in the topocentric frame. The most common ground measurement types, usually implemented by radar, telescope or laser systems, are range, range-rate, azimuth and elevation angles.

Let the topocentric position and velocity vectors of the satellite relative to Ground Station (GS)  $j$  at time  $t_k$  be  $\mathbf{r}_s^j(t_k)$  and  $\mathbf{v}_s^j(t_k)$ , respectively, according to (2.5).

The range measurement  $s^j(t_k)$  is computed by taking the norm of  $\mathbf{r}_s^j(t_k)$  and the range-rate measurement  $\dot{s}^j(t_k)$  by computing the time derivative<sup>9</sup> of the range measurement  $s^j(t_k)$ :

$$s^j(t_k) = \|\mathbf{r}_s^j(t_k)\| = \sqrt{\mathbf{r}_s^j(t_k) \cdot \mathbf{r}_s^j(t_k)}, \quad (3.37a)$$

$$\dot{s}^j(t_k) = \frac{d}{dt} (s^j(t_k)) = \frac{1}{2} \frac{1}{s^j(t_k)} ((\mathbf{r}_s^j)^T(t_k) \mathbf{v}_s^j(t_k) + (\mathbf{r}_s^j)^T(t_k) \mathbf{v}_s^j(t_k)) = \frac{(\mathbf{r}_s^j)^T(t_k) \mathbf{v}_s^j(t_k)}{s^j(t_k)}. \quad (3.37b)$$

The azimuth  $\beta^j(t_k)$  and elevation  $\delta^j(t_k)$  measurement angles (defined in Section 2.2.4), at time  $t_k$ , are computed, respectively, with (2.3b) and (2.3c). By concatenating these observations at time  $t_k$  for all available ground stations, the measurement Model (2.15b) is built

$$\mathbf{y}(t_k) = \mathbf{h}_d(\mathbf{x}(t_k), t_k) + \mathbf{v}(t_k), \quad (3.38)$$

where  $\mathbf{x}$  is the osculating state vector. A transformation of position and velocity vectors from inertial to topocentric frame, defined in Section 2.2.4, is therefore implicitly performed by  $\mathbf{h}_d$ , in accordance to the discussion above.

### 3.2.1 Measurement Noise

Measurement observations are not ideal, but rather corrupted with measurement noise that is inevitable in real-world sensors. Nevertheless, the technological evolution of the last decades has enabled sensors to become much more accurate in terms of noise statistics [2].

The measurement noise sequence  $\mathbf{v}(t_k)$  quantifies the random uncertainty associated with the observation error. It is assumed to be a sequence of white, zero-mean Gaussian noise with covariance  $\mathbf{R}_k$  at time  $t_k$ . It is reasonable to assume that all ground stations produce observations that are independent from one another. Hence, the covariance matrix is expressed by

$$\mathbf{R}_k = \text{diag}(\mathbf{R}_k^1, \dots, \mathbf{R}_k^r), \quad (3.39)$$

where  $\text{diag}(\cdot)$  is the diagonal matrix formed from the input parameters,  $r$  is the number of available stations at  $t_k$  and  $\mathbf{R}_k^j$  is the covariance associated to station  $j$ , which, in its turn, incorporates the standard deviations of the noise sequences associated to each unidimensional observation in the set  $\{s^j, \beta^j, \delta^j, \dot{s}^j\}$ , according to  $\mathbf{R}_k^j = \text{diag}((\sigma_s^j)^2, (\sigma_\beta^j)^2, (\sigma_\delta^j)^2, (\sigma_{\dot{s}}^j)^2)$ , where it was further assumed that each station produces independent unidimensional observations.

---

<sup>9</sup>The following derivative property is useful in the derivation  $\frac{\partial(\mathbf{u} \cdot \mathbf{v})}{\partial(\mathbf{w})} = \frac{\partial \mathbf{u}^T \mathbf{v}}{\partial \mathbf{w}} = \mathbf{u}^T \frac{\partial \mathbf{v}}{\partial \mathbf{w}} + \mathbf{v}^T \frac{\partial \mathbf{u}}{\partial \mathbf{w}}$ .

### 3.2.2 Station Availability

It was stated, previously, that the observation vector  $\mathbf{y}(t_k)$  is formed from measurements of all available ground stations. Therefore, a condition for station availability must be defined.

A GS is said to be unavailable when the line of sight path between the target satellite and the observer GS is blocked by the Earth. An availability (field of view) condition can be defined by assuming that the GS tracking capabilities are limited to some azimuth and elevation angular intervals. Therefore, measurements are only available when  $\beta \in [\beta_{\min}, \beta_{\max}]$  and  $\delta \in [\delta_{\min}, \delta_{\max}]$ .

In the present work, it is assumed that the whole azimuth domain is available ( $\beta_{\min} = 0$  rad and  $\beta_{\max} = 2\pi$  rad), but only positive elevation angles (above the local horizon plane) are available ( $\delta_{\min} = 0$  rad and  $\delta_{\max} = \pi$  rad).

## 3.3 Model Linearization

For the EKF, linearization of the state dynamics and measurement model requires computing the Jacobian of those models, i.e., their derivatives with respect to the state. Section 3.3.1 models the Jacobian of the osculating dynamics  $\mathbf{f}(\mathbf{x}, t)$ , Section 3.3.2 models the Jacobian of the mean dynamics  $\bar{\mathbf{f}}(\bar{\mathbf{g}}, t)$ , and Section 3.3.3 models the Jacobian of the measurement function  $\mathbf{h}_d(\mathbf{x}, t_k)$ , yielding:

$$\mathbf{F}(\mathbf{x}(t), t) = \frac{\partial \mathbf{f}(\mathbf{x}(t), t)}{\partial \mathbf{x}(t)}, \quad (3.40a)$$

$$\bar{\mathbf{F}}(\bar{\mathbf{g}}(t), t) = \frac{\partial \bar{\mathbf{f}}(\bar{\mathbf{g}}(t), t)}{\partial \bar{\mathbf{g}}(t)}, \quad (3.40b)$$

$$\mathbf{H}(\mathbf{x}(t_k), t_k) = \frac{\partial \mathbf{h}_d(\mathbf{x}(t_k), t_k)}{\partial \mathbf{x}(t_k)}. \quad (3.40c)$$

### 3.3.1 Osculating Dynamics Jacobian

In filtering applications, evaluating the state Jacobian usually requires much heavier computations than evaluating the state function itself. To reduce the computational burden in the linearization, it is common to simplify the force model of the Jacobian. This simplification seldom yields significantly worse results than those obtained with a full force model, and is much more lightweight [3, 20].

As an example, real-time implementation of an EKF on-board the Bispectral and Infrared Remote Detection (BIRD) satellite, launched in 2001, made use of a dynamic state model yielding a  $10 \times 10$  gravitational field, whilst the state dynamics Jacobian was evaluated considering only a point-mass ( $0 \times 0$ ) field [58].

In view of the discussion above, a  $2 \times 0$  gravitational field is chosen for the Jacobian computation since, cf. Figure 2.5, at LEO altitudes, the second-order zonal harmonic (commonly known as J2) is several orders of magnitude stronger than the following perturbation. The  $2 \times 0$  gravitational field is characterized by the following acceleration<sup>10</sup>[3]

<sup>10</sup>Notice that, since polar motion has been neglected (see Section 2.2.4), the  $z$  axes of ECI and ECEF coincide. Furthermore,



$$\mathbf{a}_i = \mathbf{a}_{i,0} + \mathbf{a}_{i,J2} = -\frac{\mu_{\oplus}}{r^3} \begin{bmatrix} x_i \\ y_i \\ z_i \end{bmatrix} - \frac{3J_2\mu_{\oplus}R_{\oplus}^2}{2r^5} \begin{bmatrix} \left(1 - \frac{5z_i^2}{r^2}\right) x_i \\ \left(1 - \frac{5z_i^2}{r^2}\right) y_i \\ \left(3 - \frac{5z_i^2}{r^2}\right) z_i \end{bmatrix}, \quad (3.41)$$

where  $\mathbf{a}_{i,0}$  and  $\mathbf{a}_{i,J2}$  are the two-body acceleration and the (2, 0) (J2) harmonic acceleration, respectively. The Jacobian is given, in matrix form, by

$$\frac{\partial \mathbf{f}(\mathbf{x}(t), t)}{\partial \mathbf{x}(t)} = \begin{bmatrix} \frac{\partial \mathbf{v}_i}{\partial \mathbf{r}_i} & \frac{\partial \mathbf{v}_i}{\partial \mathbf{a}_i} \\ \frac{\partial \mathbf{a}_i}{\partial \mathbf{r}_i} & \frac{\partial \mathbf{a}_i}{\partial \mathbf{v}_i} \end{bmatrix} = \begin{bmatrix} \mathbf{0}_{3 \times 3} & \mathbf{I}_{3 \times 3} \\ \frac{\partial \mathbf{a}_i}{\partial \mathbf{r}_i} & \mathbf{0}_{3 \times 3} \end{bmatrix}. \quad (3.42)$$

The partials  $\partial \mathbf{a}_i / \partial \mathbf{v}_i$  reduce to zero because the gravitational field acceleration does not depend on the state velocity and the partials  $\partial \mathbf{v}_i / \partial \mathbf{r}_i$  are zero because the state vectors are independent. The partials  $\partial \mathbf{a}_i / \partial \mathbf{r}_i$  are found by differentiating (3.41) with respect to the position vector components. The derivatives of  $\mathbf{a}_{i,0}$  are given, in matrix form, by

$$\frac{\partial \mathbf{a}_{i,0}}{\partial \mathbf{r}_i} = \begin{bmatrix} -\frac{\mu_{\oplus}}{r^3} + \frac{3\mu_{\oplus}x_i^2}{r^5} & \frac{3\mu_{\oplus}x_i y_i}{r^5} & \frac{3\mu_{\oplus}x_i z_i}{r^5} \\ \frac{3\mu_{\oplus}x_i y_i}{r^5} & -\frac{\mu_{\oplus}}{r^3} + \frac{3\mu_{\oplus}y_i^2}{r^5} & \frac{3\mu_{\oplus}y_i z_i}{r^5} \\ \frac{3\mu_{\oplus}x_i z_i}{r^5} & \frac{3\mu_{\oplus}y_i z_i}{r^5} & -\frac{\mu_{\oplus}}{r^3} + \frac{3\mu_{\oplus}z_i^2}{r^5} \end{bmatrix}, \quad (3.43)$$

and the derivatives of  $\mathbf{a}_{i,J2}$  are

$$\begin{aligned} \frac{\partial a_{x_i,J2}}{\partial x_i} &= k_1 \left[ \frac{1}{r^5} - \frac{5x_i^2}{r^7} - \frac{5z_i^2}{r^7} + \frac{35z_i^2 x_i^2}{r^9} \right], & \frac{\partial a_{x_i,J2}}{\partial y_i} &= k_1 \left[ \frac{-5x_i y_i}{r^7} + \frac{35z_i^2 x_i y_i}{r^9} \right], \\ \frac{\partial a_{x_i,J2}}{\partial z_i} &= k_1 \left[ \frac{-5x_i z_i}{r^7} + \frac{35z_i^3 x_i}{r^9} \right], & \frac{\partial a_{y_i,J2}}{\partial x_i} &= \frac{\partial a_{x_i,J2}}{\partial y_i}, \\ \frac{\partial a_{y_i,J2}}{\partial y_i} &= \frac{y_i}{x_i} \frac{\partial a_{x_i,J2}}{\partial y_i} + \frac{a_{x_i,J2}}{x_i}, & \frac{\partial a_{y_i,J2}}{\partial z_i} &= \frac{y_i}{x_i} \frac{\partial a_{x_i,J2}}{\partial z_i}, \\ \frac{\partial a_{z_i,J2}}{\partial x_i} &= k_1 \left[ \frac{-15z_i x_i}{r^7} + \frac{35z_i^3 x_i}{r^9} \right], & \frac{\partial a_{z_i,J2}}{\partial y_i} &= \frac{y_i}{x_i} \frac{\partial a_{z_i,J2}}{\partial x_i}, \\ & & \frac{\partial a_{z_i,J2}}{\partial z_i} &= k_1 \left[ \frac{3}{r^5} - \frac{30z_i^2}{r^7} + \frac{35z_i^4}{r^9} \right]. \end{aligned} \quad (3.44)$$

The constant  $k_1$  is equal to  $\frac{-3J_2\mu_{\oplus}R_{\oplus}^2}{2r^5}$ . These partials are then readily assembled together in matrix form

$$\frac{\partial \mathbf{a}_{i,J2}}{\partial \mathbf{r}_i} = \begin{bmatrix} \frac{\partial a_{x_i,J2}}{\partial x_i} & \frac{\partial a_{x_i,J2}}{\partial y_i} & \frac{\partial a_{x_i,J2}}{\partial z_i} \\ \frac{\partial a_{y_i,J2}}{\partial x_i} & \frac{\partial a_{y_i,J2}}{\partial y_i} & \frac{\partial a_{y_i,J2}}{\partial z_i} \\ \frac{\partial a_{z_i,J2}}{\partial x_i} & \frac{\partial a_{z_i,J2}}{\partial y_i} & \frac{\partial a_{z_i,J2}}{\partial z_i} \end{bmatrix}. \quad (3.45)$$

The derivative  $\partial \mathbf{a}_i / \partial \mathbf{r}_i$  is then given by the sum of (3.43) and (3.45).

---

since zonal acceleration does not depend on longitude (it only depends on latitude), Equation (3.41) can be evaluated directly in the True of Date (TOD) ECI frame [55], without the need to first compute the acceleration in the ECEF frame.

### 3.3.2 Mean Dynamics Jacobian

The Jacobian of the mean dynamics function  $\bar{\mathbf{f}}$  is approximated with finite differences. Let the Jacobian be divided into variational vectors of the six state variables  $\mathcal{E}_j$ , that is,

$$\bar{\mathbf{F}}(\bar{\mathcal{E}}(t), t) = \frac{\partial \bar{\mathbf{f}}(\bar{\mathcal{E}}(t), t)}{\partial \bar{\mathcal{E}}(t)} = \begin{bmatrix} \frac{\partial \bar{\mathbf{f}}}{\partial \mathcal{E}_1} & \cdots & \frac{\partial \bar{\mathbf{f}}}{\partial \mathcal{E}_6} \end{bmatrix}, \quad (3.46)$$

where each column-vector partial  $\partial \bar{\mathbf{f}} / \partial \bar{\mathcal{E}}_j$  is computed with the double-sided finite difference [15]

$$\frac{\partial \bar{\mathbf{f}}}{\partial \mathcal{E}_j} = \frac{\bar{\mathbf{f}}(\bar{\mathcal{E}} + \Delta \mathcal{E}_j, t) - \bar{\mathbf{f}}(\bar{\mathcal{E}} - \Delta \mathcal{E}_j, t)}{2 \|\Delta \mathcal{E}_j\|}. \quad (3.47)$$

$\Delta \mathcal{E}_j = [0, \dots, 0, \Delta \mathcal{E}_j, 0, \dots, 0]^T$  is the finite-difference vector of element  $j$ , that is, a null vector where the  $j$ th zero entry is replaced by the difference step  $\Delta \mathcal{E}_j$ .

In the present work, it was found that setting the finite-difference increments on each element according to  $\Delta \mathcal{E}_j \sim 10^{-5} \mathcal{E}_j$  yields good results, which is consistent with Green's findings in [15]. After propagating a representative LEO satellite, for a propagation period of a couple of days, the following increments were set:  $\Delta \mathcal{E}_1 = 10^{-3}$  km,  $\Delta \mathcal{E}_2 = 10^{-7}$ ,  $\Delta \mathcal{E}_3 = 10^{-7}$ ,  $\Delta \mathcal{E}_4 = 10^{-6}$ ,  $\Delta \mathcal{E}_5 = 10^{-6}$  and  $\Delta \mathcal{E}_6 = 10^{-6}$  rad.

Green also performed in [15] some comparison tests regarding the computation of the Semianalytical Jacobian. He found that the partial derivatives computed with the osculating (Cowell) formulation,  $\mathbf{F}(\mathbf{x}, t)$ , consistently agree with the ones obtained with the mean (Semianalytical) formulation,  $\bar{\mathbf{F}}(\bar{\mathcal{E}}, t)$ . It should be noted these derivatives are not with respect to the same state variables, therefore some state reduction transformations are needed. Green also compared the Semianalytical Jacobian using different force models and found that the use of simple models (low degree zonal harmonics) is sufficient.

In view of Green's findings, the finite-difference computation of the Jacobian considered in this work makes use of a reduced  $2 \times 0$  (J2) gravitational field, which is consistent with the one implemented for the osculating (Cowell) dynamics.

### 3.3.3 Measurement Model Jacobian

The Jacobian of the measurement model is, at a given state  $\mathbf{x}(t_k)$  and time  $t_k$ , given by

$$\frac{\partial \mathbf{h}_d(\mathbf{x}(t_k), t_k)}{\partial \mathbf{x}(t_k)} = \begin{bmatrix} \frac{\partial \mathbf{h}_d(\mathbf{x}(t_k), t_k)}{\partial \mathbf{r}_i} & \frac{\partial \mathbf{h}_d(\mathbf{x}(t_k), t_k)}{\partial \mathbf{v}_i} \end{bmatrix}. \quad (3.48)$$

It should be noted that the dimension of the matrix above is  $(r \cdot m) \times 6$ , where  $r$  is the number of available stations and  $m = 4$  is the number of unidimensional observations for each station. Below, the partials of the four measurement types for a generic station  $j$  (the superscript is dropped for simplicity) are presented. Concatenation of these partials to form the Jacobian is then straightforward.

Denoting by  $y$  a generic unidimensional measurement function, i.e., either one of  $s$ ,  $\beta$ ,  $\delta$  or  $\dot{s}$ , it can be shown that the partials with respect to the inertial state  $\mathbf{x}$  can be expressed in terms of the topocentric

quantities as [20]:

$$\frac{\partial y}{\partial \mathbf{r}_i} = \frac{\partial y}{\partial \mathbf{r}_s} \frac{\partial \mathbf{r}_s}{\partial \mathbf{r}_i} = \frac{\partial y}{\partial \mathbf{r}_s} \mathbf{T}_f^s \mathbf{T}_i^f, \quad (3.49a)$$

$$\frac{\partial y}{\partial \mathbf{v}_i} = \frac{\partial y}{\partial \mathbf{v}_s} \frac{\partial \mathbf{v}_s}{\partial \mathbf{v}_i} = \frac{\partial y}{\partial \mathbf{v}_s} \mathbf{T}_f^s \mathbf{T}_i^f. \quad (3.49b)$$

Differentiating (2.3b), (2.3c), (3.37a) and (3.37b) with respect to  $\mathbf{r}_s$  and  $\mathbf{v}_s$  yields

$$\begin{aligned} \frac{\partial s}{\partial \mathbf{r}_s} &= \frac{\mathbf{r}_s^T}{s}, & \frac{\partial s}{\partial \mathbf{v}_s} &= 0, \\ \frac{\partial \dot{s}}{\partial \mathbf{r}_s} &= \frac{s \mathbf{v}_s^T - \dot{s} \mathbf{r}_s^T}{s^2}, & \frac{\partial \dot{s}}{\partial \mathbf{v}_s} &= \frac{\mathbf{r}_s^T}{s}, \\ \frac{\partial \beta}{\partial \mathbf{r}_s} &= \left[ \frac{s_N}{s_E^2 + s_N^2}, \frac{-s_E}{s_E^2 + s_N^2}, 0 \right], & \frac{\partial \beta}{\partial \mathbf{v}_s} &= 0, \\ \frac{\partial \delta}{\partial \mathbf{r}_s} &= \left[ \frac{-s_E s_Z}{s^2 \sqrt{s_E^2 + s_N^2}}, \frac{-s_N s_Z}{s^2 \sqrt{s_E^2 + s_N^2}}, \frac{\sqrt{s_E^2 + s_N^2}}{s^2} \right], & \frac{\partial \delta}{\partial \mathbf{v}_s} &= 0. \end{aligned} \quad (3.50)$$

The derivatives above yield the Jacobian with respect to the state vector  $\mathbf{x}$ . For Semianalytical-based filtering algorithms, this Jacobian needs to be defined with respect to the mean equinoctial state  $\bar{\mathcal{E}}$ . The derivative  $\partial \mathbf{h}_d / \partial \bar{\mathcal{E}}$  is readily divided into the following partials, using the chain rule

$$\frac{\partial \mathbf{h}_d}{\partial \bar{\mathcal{E}}} = \frac{\partial \mathbf{h}_d}{\partial \mathbf{x}} \frac{\partial \mathbf{x}}{\partial \bar{\mathcal{E}}} \frac{\partial \bar{\mathcal{E}}}{\partial \bar{\mathcal{E}}}. \quad (3.51)$$

The first partial is the Jacobian of the measurement model with respect to the osculating state vector and its computation was already discussed above. The middle partial is the derivative of the state vector with respect to the equinoctial set and is provided in Appendix D (Section D.2). Finally, the last partial is obtained by differentiating the *near-identity transformation* (3.34) with respect to the mean state. First, the map is written in a more compact form:

$$\mathcal{E} = \bar{\mathcal{E}} + \eta(\bar{\mathcal{E}}). \quad (3.52)$$

The desired partial is then

$$\frac{\partial \mathcal{E}}{\partial \bar{\mathcal{E}}} = \mathbf{I} + \frac{\partial \eta}{\partial \bar{\mathcal{E}}}. \quad (3.53)$$

Finally, the derivative  $\frac{\partial \eta}{\partial \bar{\mathcal{E}}}$  is computed with finite differences, using the same procedure that was described in Section 3.3.2. Again, using a simplified J2 gravitational field force model, the following finite-difference increments were found:  $\Delta \mathcal{E}_1 = 10^{-5}$  km,  $\Delta \mathcal{E}_2 = 10^{-10}$ ,  $\Delta \mathcal{E}_3 = 10^{-10}$ ,  $\Delta \mathcal{E}_4 = 10^{-10}$ ,  $\Delta \mathcal{E}_5 = 10^{-10}$  and  $\Delta \mathcal{E}_6 = 10^{-10}$  rad.

### 3.4 Forming Orbit Determination Filters

Having modeled both the state dynamics and the measurement function, Orbit Determination filters are now introduced. Since two state dynamical formulations were developed — Cowell (osculating

dynamics) and Semianalytical (mean dynamics) — two kinds of filters are designed, in order to take advantage of both formulations. Cowell filters are described in Section 3.4.1 and Semianalytical filters are presented in Section 3.4.2. To conclude, Section 3.4.3 discusses filter initialization procedures.

In the implemented simulation environment, filters run at a fixed step of 5 seconds, that is, ground stations produce synchronized measurements every 5 seconds. It should be noted that ground stations will not always be available at the update-step times (cf. Section 3.2.2). Thus, in the absence of measurements, only the predict steps are used. Once the line of sight is restored, filters resume their predict-update cycle normally. On the other hand, if multiple stations are available at the same update time, since all stations produce independent observations, the update step is divided into separate update steps, to be performed in series, one for each station. This procedure, which reduces the dimension of the matrices involved and hence the computational cost, stems from the fact that all stations produce independent observations, hence global matrices for the recursive filter update are block diagonal and can be efficiently handled through sequential updates [59].

### 3.4.1 Cowell Filters

Cowell filters are one of the most common and straightforward ways to perform Orbit Determination. They have been extensively studied before, for instance, in Orbital Mechanics textbooks [3, 20, 21] and in recent research papers [18, 19]. They have also been successfully implemented in many orbital software packages, for instance, open-source software like GMAT or Orekit, or space-agency software like the Goddard Trajectory Determination System (GTDS) from NASA or the NAVigation Package for Earth Orbiting Satellites (NAPEOS) from the European Space Agency (ESA). In view of this, the Cowell filters presented in this section have already been studied and tested before, thus providing a convenient baseline comparison with Semianalytical filters in the context of this thesis.

In Cowell filters, the dynamical state is represented with the osculating state vector  $\mathbf{x}$ . Two filters are proposed: Extended Cowell Kalman Filter and Unscented Cowell Kalman Filter. These filters match the general implementation of EKF and UKF filters, respectively, as given by Algorithm 2 and Algorithm 3 in section 2.5.2. However, slight changes are needed, since update steps are skipped in the absence of station availability, as previously mentioned. The state dynamics are modeled with (3.2) and the measurement model resorts to (3.38). The Jacobian of the state and measurement models, needed for the EKF, are given, respectively, by (3.42) and (3.48). Furthermore, the UKF takes discretized dynamics, which are obtained by discretizing the continuous-time state dynamical model with (2.17). A flowchart illustrating an algorithmic implementation of Cowell filters, that is straightforwardly adapted to both EKF and UKF algorithms, is provided in Appendix A (Section A.3).

### 3.4.2 Semianalytical Filters

The idea of coupling Semianalytic Satellite Theory (SST) with filtering algorithms, namely, the Differential Correction Batch Least Squares and the Extended Kalman Filter, was first proposed and demonstrated in great detail in Green's thesis [15] in 1979, making use of the DSST propagator. Green hinted

that: 1) this coupling would increase computational speed of Orbit Determination systems due to the computational efficiency of the propagator and its large allowable integration step sizes, and 2) would also increase the accuracy because the linearization assumptions used in the algorithms would be better satisfied due to the near-linear behavior of the mean dynamics over an integration step. After Green's theoretical work, Taylor [25] designed, implemented, verified and tested in his thesis in 1982 the Extended Semianalytical Kalman Filter (ESKF), that couples a traditional EKF with DSST. More recently, in 2008, Folcik [26] coupled DSST with a Backward Smoothing Extended Kalman Filter. The aforementioned research relied on the original software implementation of DSST. In contrast, the present work departs from DSST reliance, and is supported, instead, by the Semianalytical theory introduced in Section 3.1.2, which is based on Ely's work.

The fundamental idea of coupling SST with filtering algorithms involves the following time frame definitions [25]:

- **Integration grid** - the time frame used by the semianalytical integrator, which means that the points on this grid are greatly spaced in time, according to the integrator step size;
- **Observation grid** - the time frame that contains the arrival times of the observations to be processed by the filter.

The operation of Semianalytical filters is the following: 1) the integrator propagates the trajectory along the integration grid points at times  $t_{k,0}$ , originating the nominal trajectory  $\bar{\mathcal{E}}_N(t_{k,0})$ ; 2) this nominal trajectory is then interpolated, in-between integration grid points, to the arrival times of observations, i.e., interpolated to observation times  $t_{k,i}$ , for  $i = 1, \dots, M$  (where  $M$  is the last point before the next integration time  $t_{k+1,0}$ ), originating the nominal trajectory  $\bar{\mathcal{E}}_N(t_{k,i})$  along the observation grid; 3) after the measurements at times  $t_{k,i}$  are processed, the filter corrections are propagated through the observation grid, without an explicit update of the nominal trajectory, which is then only updated at the next integration grid point at time  $t_{k+1,0}$ . By avoiding the relinearization of the nominal trajectory after every update step, large integration step sizes are possible, which evidences the efficiency of Semianalytical filters.

In the remainder of this section the Extended Semianalytical Kalman Filter (ESKF) is introduced. Next, a novel algorithm that couples the Semianalytical model with an Unscented Kalman Filter — the Unscented Semianalytical Kalman Filter (USKF) — is proposed. A flowchart illustrating an algorithmic implementation of Semianalytical filters is provided in Appendix A (Section A.3), for better visualization of the algorithms presented below.

### Extended Semianalytical Kalman Filter

In the ESKF proposed by Taylor, filter corrections are propagated with the State Transition Matrix (STM). To illustrate this process, let  $\bar{\mathcal{E}}_N(t_{k,i})$  be the nominal state at time  $t_{k,i}$  and let  $\Delta\bar{\mathcal{E}}_{k,i}^{k,i}$  be the filter correction after the measurement at  $t_{k,i}$ . The *a posteriori* state is then

$$\bar{\mathcal{E}}^+(t_{k,i}) = \bar{\mathcal{E}}_N(t_{k,i}) + \Delta\bar{\mathcal{E}}_{k,i}^{k,i}, \quad (3.54)$$

If the dynamics are linear, the *a priori* state at the next observation time  $t_{k,i+1}$  may be computed with

$$\begin{aligned}\bar{\mathcal{G}}^-(t_{k,i+1}) &= \Phi(t_{k,i+1}, t_{k,i}) \bar{\mathcal{G}}^+(t_{k,i}) \\ &= \Phi(t_{k,i+1}, t_{k,i}) \bar{\mathcal{G}}_N(t_{k,i}) + \Phi(t_{k,i+1}, t_{k,i}) \Delta \bar{\mathcal{G}}_{k,i}^{k,i} \\ &= \bar{\mathcal{G}}_N(t_{k,i+1}) + \Delta \bar{\mathcal{G}}_{k,i+1}^{k,i},\end{aligned}\tag{3.55}$$

where  $\Phi$  is the system's STM. The notation  $\Delta \bar{\mathcal{G}}_k^l$  indicates a filter correction at time  $t_k$  given observations until time  $t_l$ : if  $l < k$  it is a predicted correction and if  $l = k$  it is an updated correction.

Equation (3.55) states that the *a priori* state is computed by adding up the nominal state with the filter correction from the previous step, propagated in time with the STM. In order for this process to work, it is assumed that the system dynamics are linear (or, at least, near-linear in some sense), which is true for mean orbital dynamics. It is then evident that the solve-for dynamical state to be estimated with Semianalytical filters is the mean equinoctial set, in contrast to the solve-for osculating state vector of Cowell filters.

The complete operation of the ESKF, paraphrased from [25] but adapted to the present Semianalytical propagator, is provided in Appendix A (Section A.2).

Taylor verified with some simulations that the underlying assumptions that were made, namely, the near-linear behavior of the dynamics and the linearization of the measurement model in successive steps, are held true to some degree of precision [25]. Furthermore, Green demonstrated that Fourier coefficients have smooth variations over time and hence can be accurately interpolated [15].

### Unscented Semianalytical Kalman Filter

Although the ESKF makes use of continuous-time state dynamics, the novel USKF proposed in this work is implemented considering discretized state dynamics, according to the traditional implementation of UKFs, cf. Section 2.5.2. Discretization of (3.33), between integration grid nominal points at times  $t_{k,0}$  and  $t_{k+1,0}$ , with (2.17) yields

$$\bar{\mathcal{G}}_N(t_{k+1,0}) = \bar{f}_d(\bar{\mathcal{G}}_N(t_{k,0}), t_{k,0}) + \bar{\mathbf{w}}_{k,0}.\tag{3.56}$$

where  $\bar{\mathbf{w}}_{k,0}$  is the discretized process noise.

The underlying idea of coupling the Semianalytical propagator with the UKF is the same as with the ESKF, i.e., after each observation is processed, the filter correction is propagated with linearized dynamics around the nominal trajectory. However, in the present case, the linearization of the nominal trajectory is performed using Weighted Statistical Linearization (WSL), cf. Section 2.5.1. The WSL of (3.56) around the nominal state  $\bar{\mathcal{G}}_N(t_{k,i})$  enables propagation between observation grid time steps, that is,

$$\bar{\mathcal{G}}(t_{k,i+1}) \approx \mathbf{A}_{k,i}^N \bar{\mathcal{G}}(t_{k,i}) + \mathbf{b}_{k,i}^N + \bar{\mathbf{w}}_{k,i},\tag{3.57}$$

where  $\mathbf{A}_{k,i}^N$  and  $\mathbf{b}_{k,i}^N$  are the coefficients of the WSL. Replacing the first-order Taylor linearization used in

the filter correction propagation (3.55) with the WSL defined above gives

$$\begin{aligned}
\bar{\mathcal{E}}^-(t_{k,i+1}) &= \mathbf{A}_{k,i}^N \bar{\mathcal{E}}^+(t_{k,i}) + \mathbf{b}_{k,i}^N \\
&= \mathbf{A}_{k,i}^N \bar{\mathcal{E}}_N(t_{k,i}) + \mathbf{A}_{k,i}^N \Delta \bar{\mathcal{E}}_{k,i}^{k,i} + \mathbf{b}_{k,i}^N \\
&= \bar{\mathcal{E}}_N(t_{k,i+1}) + \mathbf{A}_{k,i}^N \Delta \bar{\mathcal{E}}_{k,i}^{k,i}.
\end{aligned} \tag{3.58}$$

The first term on the right-hand side of the last equality is the nominal state at  $t_{k,i+1}$ , which, according to the linearization just defined, is given as (neglecting process noise)  $\bar{\mathcal{E}}_N(t_{k,i+1}) = \mathbf{A}_{k,i}^N \bar{\mathcal{E}}_N(t_{k,i}) + \mathbf{b}_{k,i}^N$ . It is then observed that the filter correction may be propagated in time using the linearization coefficient  $\mathbf{A}_{k,i}^N$ , which is, therefore, the WSL equivalent of the STM.

The operations on the integration and observation grids of the proposed USKF are given below. The filter is initialized with an initial state estimate  $\bar{\mathcal{E}}(t_0)$ , covariance  $\mathbf{P}_0$  and with null corrections  $\Delta \bar{\mathcal{E}}_{0,0}^{0,0} = \mathbf{0}$ .

#### Operations on the Integration Grid

1. At the current time  $t_{k,0}$ , update the nominal state,  $\bar{\mathcal{E}}_{N_{\text{new}}}(t_{k,0})$ , for the new integration step, valid on the interval  $t \in [t_{k,0}, t_{k+1,0})$ , from the old nominal state  $\bar{\mathcal{E}}_{N_{\text{old}}}(t_{k,0})$  (defined on the previous integration grid step), using:

$$\bar{\mathcal{E}}_{N_{\text{new}}}(t_{k,0}) = \bar{\mathcal{E}}_{N_{\text{old}}}(t_{k,0}) + \Delta \bar{\mathcal{E}}_{k,0}^{k-1,M}, \tag{3.59}$$

where  $\Delta \bar{\mathcal{E}}_{k,0}^{k-1,M}$  is the filter correction from the last observation, at  $t_{k-1,M}$ , propagated until the current time  $t_{k,0}$ . Then, set the nominal covariance with  $\mathbf{P}_{k,0}^N = \mathbf{P}_{k,0}^-$ . The correction  $\Delta \bar{\mathcal{E}}_{k,0}^{k-1,M}$  and covariance  $\mathbf{P}_{k,0}^-$  are found by performing steps 2, 3, 4 and 5 of the observation grid (described below) from  $t_{k-1,M}$  to  $t_{k,0}$ ;

2. With  $\bar{\mathcal{E}}_N(t_{k,0})$  and  $\mathbf{P}_{k,0}^N$ , construct the  $2n+1$  nominal sigma points  $\mathcal{X}_i^N(t_{k,0})$  defined by (2.26), and store them in  $\mathcal{X}_S$ . To simplify the notation, the sigma points are concatenated in vector-form as  $\mathcal{X}^N(t_{k,0})$ . Then, initialize the filter correction:

$$\mathcal{X}_S = \mathcal{X}^N(t_{k,0}), \tag{3.60a}$$

$$\Delta \bar{\mathcal{E}}_{k,0}^{k,0} = \mathbf{0}; \tag{3.60b}$$

3. Numerically propagate the nominal trajectory and covariance using the UT (Algorithm 1 in Section 2.5.1) until time  $t = t_{k,0} + \Delta t = t_{k+1,0}$ , saving intermediate results at  $t = t_{k,\Delta t/3}$  and  $t = t_{k,2\Delta t/3}$ ,

$$[\bar{\mathcal{E}}_N(t_{k+1}), \mathbf{P}_{k+1}^N, -] = \text{UT}(\bar{\mathbf{f}}_d, \bar{\mathcal{E}}_N(t_k), \mathbf{P}_k^N), \tag{3.61}$$

from initial conditions  $\bar{\mathcal{E}}_N(t_{k,0})$  and  $\mathbf{P}_{k,0}^N$ , that were set on step 1. With the state and covariance at the requested times, construct the associated nominal sigma points. With the sets  $\mathcal{X}^N(t_{k,0})$ ,  $\mathcal{X}^N(t_{k,\Delta t/3})$ ,  $\mathcal{X}^N(t_{k,2\Delta t/3})$  and  $\mathcal{X}^N(t_{k+1,0})$ , set up a Lagrange interpolator for  $\mathcal{X}^N(t)$ .

#### Operations on the Observation Grid

The operations on the observation grid are triggered by receipt of a new observation. The observation grid procedure is followed in a loop-wise manner until no more observations are available or the next observation is not inside the interval  $[t_{k,0}, t_{k+1,0})$ , i.e., is later than the next integration time step. In that case, the integration step procedure described above is followed to advance the integration by one grid point. The observation grid is operated by the UKF: steps 2 to 5 belong to the filter's predict step and step 6 to the update step. It is noted that, if the first observation comes exactly at time  $t = t_{k,0}$ , then steps 2 to 5 are ignored for that observation, since the predicted state and covariance are already known from the integration step.

1. Obtain a new observation to process  $\mathbf{y}(t_{k,i})$  for  $i = 0, \dots, M$ , where  $\mathbf{y}(t_{k,M})$  is the last observation before the next integration point;
2. Interpolate  $\mathcal{X}^N(t)$  for  $t = t_{k,i}$ . Get  $\mathcal{X}^N(t_{k,i-1})$  from the value stored in  $\mathcal{X}_S$ . Compute the nominal state and covariance at the two epochs using:

$$\bar{\mathcal{G}}_N(t_{k,i}) = \sum_{j=0}^{2n} \mathcal{W}_j^{(m)} \mathcal{X}_j^N(t_{k,i}), \quad (3.62a)$$

$$\bar{\mathcal{G}}_N(t_{k,i-1}) = \sum_{j=0}^{2n} \mathcal{W}_j^{(m)} \mathcal{X}_j^N(t_{k,i-1}), \quad (3.62b)$$

$$\mathbf{P}_{k,i}^N = \sum_{j=0}^{2n} \mathcal{W}_j^{(c)} (\mathcal{X}_j^N(t_{k,i}) - \bar{\mathcal{G}}_N(t_{k,i})) (\mathcal{X}_j^N(t_{k,i}) - \bar{\mathcal{G}}_N(t_{k,i}))^T, \quad (3.62c)$$

$$\mathbf{P}_{k,i-1}^N = \sum_{j=0}^{2n} \mathcal{W}_j^{(c)} (\mathcal{X}_j^N(t_{k,i-1}) - \bar{\mathcal{G}}_N(t_{k,i-1})) (\mathcal{X}_j^N(t_{k,i-1}) - \bar{\mathcal{G}}_N(t_{k,i-1}))^T, \quad (3.62d)$$

$$\mathbf{P}_{k,i,i-1}^N = \sum_{j=0}^{2n} \mathcal{W}_j^{(c)} (\mathcal{X}_j^N(t_{k,i}) - \bar{\mathcal{G}}_N(t_{k,i})) (\mathcal{X}_j^N(t_{k,i-1}) - \bar{\mathcal{G}}_N(t_{k,i-1}))^T, \quad (3.62e)$$

where the UT weights are given by (2.28);

3. Compute the nominal WSL coefficient  $\mathbf{A}_{k,i}^N$

$$\mathbf{A}_{k,i}^N = \mathbf{P}_{k,i,i-1}^N (\mathbf{P}_{k,i-1}^N)^{-1}; \quad (3.63)$$

4. Obtain the predicted filter correction and compute the *a priori* mean state  $\bar{\mathcal{G}}^-(t_{k,i})$ :

$$\Delta \bar{\mathcal{G}}_{k,i}^{k,i-1} = \mathbf{A}_{k,i}^N \Delta \bar{\mathcal{G}}_{k,i-1}^{k,i-1}, \quad (3.64a)$$

$$\bar{\mathcal{G}}^-(t_{k,i}) = \bar{\mathcal{G}}_N(t_{k,i}) + \Delta \bar{\mathcal{G}}_{k,i}^{k,i-1}; \quad (3.64b)$$

5. Discretize the process noise covariance and compute the predicted covariance:

$$\begin{aligned} \mathbf{Q}_d(t_{k,i}, t_{k,i-1}) &= \mathbf{A}_{k,i}^N \mathbf{Q}(t_{k,i}) (\mathbf{A}_{k,i}^N)^T \cdot (t_{k,i+1} - t_{k,i}), \\ \mathbf{P}_{k,i}^- &= \mathbf{A}_{k,i}^N \mathbf{P}_{k,i-1}^+ (\mathbf{A}_{k,i}^N)^T + \mathbf{Q}_d(t_{k,i}, t_{k,i-1}), \end{aligned} \quad (3.65)$$

where  $\mathbf{Q}(t_{k,i})$  is the continuous-time noise covariance matrix;



6. Consider the measurement function with respect to the mean equinoctial state as

$$\bar{h}(\bar{\mathcal{E}}(t)) = \mathbf{h}_d \left( \mathbf{X}(\bar{\mathcal{E}}(t) + \boldsymbol{\eta}(\bar{\mathcal{E}}(t), \theta(t))), t \right), \quad (3.66)$$

where  $\mathbf{X}$  is the conversion from osculating  $\mathcal{E}$  to  $\mathbf{x}$  and  $\boldsymbol{\eta}$  is the *mean-to-osculating map* written in compact form. Do the traditional UKF update step:

$$[\hat{\mathbf{y}}(t_{k,i}), \mathbf{P}_y, \mathbf{P}_{\mathcal{E}y}] = \text{UT} \left( \bar{\mathbf{h}}, \bar{\mathcal{E}}^-(t_{k,i}), \mathbf{P}_{k,i}^- \right), \quad (3.67a)$$

$$\mathbf{K}_{k,i} = \mathbf{P}_{\mathcal{E}y} [\mathbf{P}_y + \mathbf{R}_{k,i}]^{-1}, \quad (3.67b)$$

$$\Delta \bar{\mathcal{E}}_{k,i}^{k,i} = \Delta \bar{\mathcal{E}}_{k,i}^{k,i-1} + \mathbf{K}_{k,i} (\mathbf{y}(t_{k,i}) - \hat{\mathbf{y}}(t_{k,i})), \quad (3.67c)$$

$$\bar{\mathcal{E}}^+(t_{k,i}) = \bar{\mathcal{E}}^-(t_{k,i}) + \Delta \bar{\mathcal{E}}_{k,i}^{k,i}, \quad (3.67d)$$

$$\mathbf{P}_{k,i}^+ = \mathbf{P}_{k,i}^- - \mathbf{K}_{k,i} \mathbf{P}_y \mathbf{K}_{k,i}^T. \quad (3.67e)$$

where  $\mathbf{R}_{k,i}$  is the measurement noise covariance matrix;

7. Save the sigma points of the current step into  $\mathcal{X}_S$ , i.e.,  $\mathcal{X}_S = \mathcal{X}^N(t_{k,i})$ , to be used in the next observation.

The algorithm provides, after each update step, the estimated equinoctial mean state and covariance  $\bar{\mathcal{E}}^+(t_{k,i})$  and  $\mathbf{P}_{k,i}^+$ . If requested, these quantities may be transformed to the osculating state vector space using [25]:

$$\mathbf{x}^+(t_{k,i}) = \mathbf{X} \left( \bar{\mathcal{E}}^+(t_{k,i}) + \boldsymbol{\eta}(\bar{\mathcal{E}}^+(t_{k,i}), \theta(t_{k,i})) \right), \quad (3.68a)$$

$$\mathbf{P}_{\mathbf{x}}^+(t_{k,i}) = \mathbf{G}(\bar{\mathcal{E}}^+(t_{k,i})) \mathbf{P}_{k,i}^+ \mathbf{G}^T(\bar{\mathcal{E}}^+(t_{k,i})), \quad (3.68b)$$

where  $\mathbf{G} = (\partial \mathbf{x} / \partial \bar{\mathcal{E}})$  ( $\partial \mathcal{E} / \partial \bar{\mathcal{E}}$ ). These partials were defined in Section 3.3.3.

The USKF proposed here avoids an additional linearization of  $\boldsymbol{\eta}$  around the nominal state, which, in contrast, is made in the ESKF (see (A.18b)), since the measurement function (3.66) is evaluated directly with the predicted state  $\bar{\mathcal{E}}^-(t_{k,i})$ . Although this is slightly computationally heavier, it should hold better results, especially for higher nonlinearities in the measurement model.

To conclude, Figure 3.5 shows the key idea of Semianalytical filters. The real, nominal and output (estimated) 3-D trajectories are plotted for a filtering simulation.

### 3.4.3 Filter Initialization

In Bayesian estimation, prior knowledge of the parameters being estimated is essential to the success of filtering algorithms and to avoid filter divergence issues. In this work, the initialization procedure is chosen to be self-starting, in the sense that the filter should be able to initialize taking into account only available ground station observational data. Following [1], a set of GS measurements is collected at the first  $N$  time instants of station availability, that is,  $\mathbf{Y}_{\text{init}} = \{(t_1, \mathbf{y}(t_1)), \dots, (t_N, \mathbf{y}(t_N))\}$ .

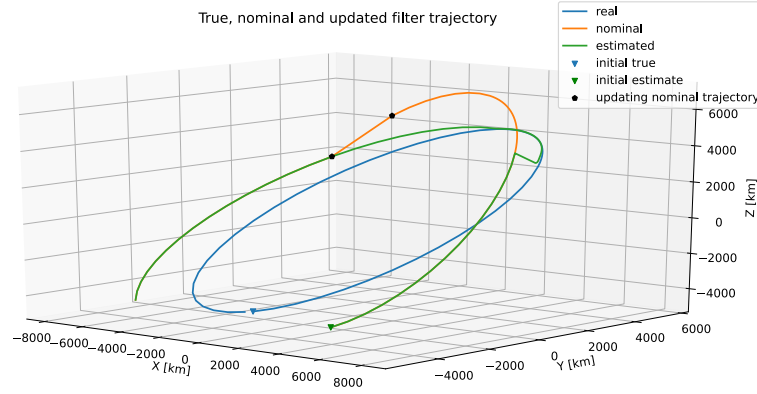


Figure 3.5: Illustration of real (blue), nominal (yellow) and output estimated (green) trajectories for a Semianalytical filtering simulation. At the beginning of the integration grid step (triangle marks), the nominal trajectory is defined with the current best filter knowledge. Then, as the measurements arrive, the estimated trajectory approaches the real one through the prediction-update cycle of the filter corrections. However, only when the integration grid step ends (black circle marks) is the nominal trajectory updated with the current filter knowledge, i.e., with the estimated state.

Then, Gauss's method of Initial Orbit Determination (IOD) is performed producing *crude* orbit estimates (without statistical treatment, i.e., without covariance estimation). The output of Gauss's method is then fed to a Differential Correction Batch Least Squares (LS) method that produces *refined* orbit,  $\hat{\mathbf{x}}^{(\text{LS})}(t_N)$ , and covariance,  $P_{\mathbf{x}}^{(\text{LS})}(t_N)$ , estimates at time  $t_N$ , that, in turn, initialize the filters.

The routine used in this work for Gauss's method is provided in Appendix D.15 of [34]. A straightforward translation to Python language was required. The batch LS method is presented in Appendix A (Section A.1) of this thesis and was readily adapted from [21].

It is emphasized that the described procedure produces an osculating prior estimate in state-vector form ( $\hat{\mathbf{x}}^{(\text{LS})}(t_N)$  and  $P_{\mathbf{x}}^{(\text{LS})}(t_N)$ ), which is sufficient for Cowell filter initialization. For Semianalytical filters, this initial distribution must be transformed to the mean equinoctial state-space. After converting the orbit estimate from (osculating) cartesian to equinoctial form,  $\hat{\mathbf{g}}^{(\text{LS})}(t_N) = \mathbf{X}(\hat{\mathbf{x}}^{(\text{LS})}(t_N))$ , the epoch point conversion of (3.36) yields  $\hat{\mathbf{g}}^{(\text{LS})}(t_N)$ . Covariance transformation is achieved by inverting the inverse transformation (from mean equinoctial to osculating state vector) proposed in (3.68b), that is,

$$P_{\mathbf{g}}^{(\text{LS})}(t_N) = G^{-1} P_{\mathbf{x}}^{(\text{LS})}(t_N) (G^T)^{-1}. \quad (3.69)$$

# Chapter 4

## Simulation Results

In this chapter, simulation results of propagation and filtering algorithms, which were modeled in the previous chapter, are presented and discussed, for a variety of test cases and experiments. The purpose of these experiments is to illustrate and outline the performance and main characteristics of both Cowell and Semianalytical propagation models and their coupling with OD filters. An evaluation of Cowell and Semianalytical propagation accuracy, computational cost and filter performance is thus envisioned. The influence of initialization errors and different measurement configurations are also analyzed.

Section 4.1 considers the case of a single ground station located in Lisbon and section 4.2 adopts the same simulation environment set up in [26], which considers four ground stations.

### 4.1 Test Case 1 - Single Ground Station

The simulation tests of this section consider a Sun-synchronous (SS) Low Earth Orbit (LEO) satellite. SS are near-polar orbits that maintain their constant orientation towards the Sun throughout the year, i.e., the satellite passes over any given point of the planet's surface at the same local mean solar time [3]. These orbits find many applications in Earth observation (remote sensing). Design of SS orbits requires selecting an inclination  $i$  that satisfies the following relation [3]

$$\dot{\Omega} = -\frac{3nJ_2R_{\oplus}^2}{2a^2(1-e^2)^2} \cos i = \frac{2\pi}{1 \text{ sidereal year}}, \quad (4.1)$$

where 1 sidereal year is  $\sim 365.2421897$  days [3]. The initial epoch and Keplerian state of the designed orbit are given in Table 4.1. Elements  $a$  and  $e$  were chosen in order to obtain a near-circular LEO.  $i$  was chosen to satisfy the relation above. Finally,  $\Omega$ ,  $\varpi$  and  $\nu$  were chosen arbitrarily.

Initial orbital conditions	
Epoch	6 April, 2000 11:00:00 UTC
Osculating Keplerian set	(7178 km, 0.03, 98.6°, 20°, 0°, 0°)
$(a, e, i, \Omega, \varpi, \nu)$	

Table 4.1: Initial epoch and orbital state for Test Case 1.

In order to conduct the simulation (propagation and filtering) studies, a reference (true) orbital trajectory needs to be defined. This reference trajectory is used to generate noisy observations, which are obtained by summing the true observations (computed with the true trajectory) with random noise, according to the defined measurement model and noise statistics. The random noise sequences are realizations of zero-mean GRVs, with predefined covariance. Table 4.2 shows the selected standard deviations of each individual measurement type. These noise statistics were chosen arbitrarily, but represent typical values of LEO radar tracking systems [3, 60]. Furthermore, the reference trajectory is also used for evaluating filter performance, through the Root Mean Square Error (RMSE) metric (cf. Section 2.7.1).

Measurement type	Standard deviation
Range	100 m
Azimuth and elevation	0.02°
Range-rate	10 cm s <sup>-1</sup>

Table 4.2: Sensor measurement noise standard deviations for Test Case 1.

The reference orbit (characterized by the initial conditions of Table 4.1) was generated using GMAT<sup>1</sup> Cowell propagation. Table 4.3 depicts the important configurations used set up the GMAT script. It is emphasized that the selected force model is much more accurate than the one to be implemented in the dynamical model of the filtering schemes. The idea is to simulate a realistic scenario where the satellite is indeed affected by a full-force model, whereas the filter dynamics only includes the most important perturbations. The difference in force models is accounted for by process noise. Figure 4.1 illustrates the reference trajectory, as obtained by the GMAT software.

GMAT Reference Orbit Generation		
<u>Force Model:</u>	Gravitational Field	180 × 180 field with EGM96 model
	Atmospheric drag	Jacchia Roberts atmospheric model
	Solar Radiation Pressure (SRP)	cannonball (spherical) model
	Third bodies	Sun, Moon and Jupiter (DE405 ephemeris file)
	Solid and pole Earth tides	
	Relativistic corrections	
<u>Integration Solver:</u>	Runge–Kutta 89 with Root Sum Square error control	
	Tolerance	9.999 × 10 <sup>-12</sup>
<u>Reference frames:</u>	Integration frame	J2000 (realization of ECI)
	Earth-fixed frame	ITRF (realization of ECEF)

Table 4.3: GMAT configuration for reference orbit generation.

The single ground station is located in Lisbon, with the following geodetic coordinates: altitude  $h =$

<sup>1</sup>GMAT is an open-source orbital mission analysis tool provided by NASA. Available online at (last access on 12th December 2020) <https://software.nasa.gov/software/GSC-18094-1>.

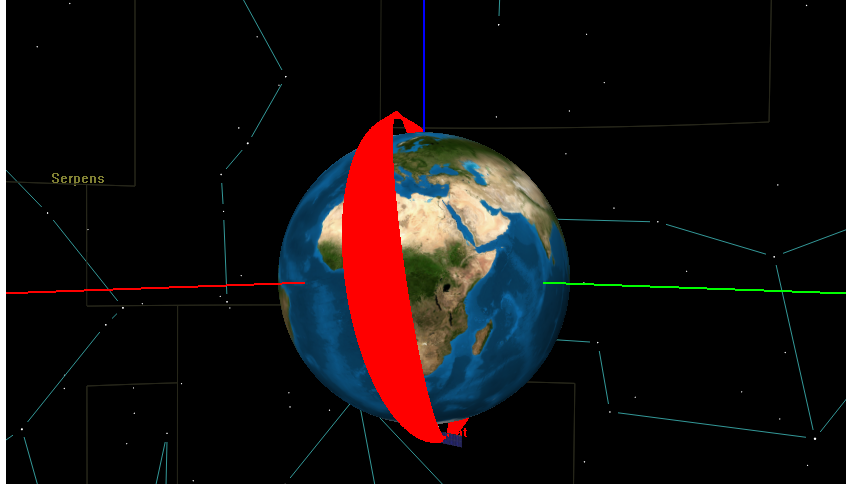


Figure 4.1: Illustration of the reference trajectory obtained with GMAT for Test Case 1. The red, green and blue axes are the  $x$ ,  $y$  and  $z$  axes of the J2000 frame, respectively. The simulation propagation period was 20 days. As evidenced in the figure, perturbations force oscillations in the orbit's shape and orientation.

0 m, latitude  $\phi_{gd} = 38.7^\circ$  and longitude  $\lambda = -9.2^\circ$ . Moreover, the physical properties of the satellite (held constant throughout the simulations) are shown in Table 4.4.

Satellite Property	Numerical Value
Mass $m$	25 kg
Drag Area $A_{drag}$	0.5 m <sup>2</sup>
Coefficient of Drag $C_D$	2.0
SRP Area $A_{SRP}$	0.5 m <sup>2</sup>
SRP Coefficient of Reflectivity $C_R$	1.5

Table 4.4: Physical properties of the satellite. It is noted that  $A_{SRP}$  and  $C_R$  are only defined to be used in the GMAT reference trajectory generation, since SRP is not part of the filter dynamical model.

### 4.1.1 Propagation Results

The tests conducted in this section try to find and establish a baseline comparison between the Cowell and Semianalytical propagators developed in this thesis and similar algorithms from known publicly available software packages. The objective is then to validate the software developed in this work, if the results are satisfactory. In view of this, the developed Cowell propagator is compared with GMAT propagator and the developed Semianalytical propagator is compared with DSST propagator, accessed through the Orekit<sup>2</sup> orbital package. Finally, this section concludes with a computation cost study, where the efficiency of semianalytical propagation is evidenced.

<sup>2</sup>The open-source Orbit Extrapolation Kit (Orekit) Java flight dynamics library is available online at (last access on 12th December 2020) <https://www.orekit.org>.

## Choosing the Numerical Integration Solver

First of all, it is crucial to select a numerical integration routine (solver), whose purpose is to numerically propagate/integrate the orbital dynamical models. Following a brief analysis of commonly used numerical solvers for orbital propagators performed in [20, 61], low-order Runge-Kutta methods were chosen, since the solver is intended to be implemented in OD problems, where integration steps between measurements are typically small. Use of higher order methods would reduce the integration efficiency, since the exploitation of large integration step sizes supported by them would not be used [4].

In view of this, two Runge-Kutta methods are analyzed: a Runge-Kutta 5(4) (denoted as RK45) [62], and a variation of an 8th order Runge-Kutta developed by Dormand & Prince known as DOP853 [63]. These solvers are accessed via the SciPy package. In both solvers, relative and absolute tolerances of numerical integration are set to  $10^{-3}$  and  $10^{-6}$ , respectively.

A straightforward test for the assessment of numerical integration performance is to apply the solver to a problem which has a known analytical solution. This comparison finds limitations of the numerical routines, by setting an absolute error according to the selected integration method, step size, accuracy and order [61]. The natural test concerning orbital propagators is the propagation of the Keplerian Orbit, whose analytical solution was discussed in Section 2.4.1. The numerical solution is found by integrating the osculating state dynamical model with perturbations turned off, i.e., considering two-body dynamics only. Results of RK45 and DOP853 numerical integration, in terms of trajectory averaged position and velocity  $\overline{\text{RMSE}}$ , are shown in Figure 4.2, as a function of different integration step sizes, for a 20-day propagation period, given the initial conditions of Table 4.1.

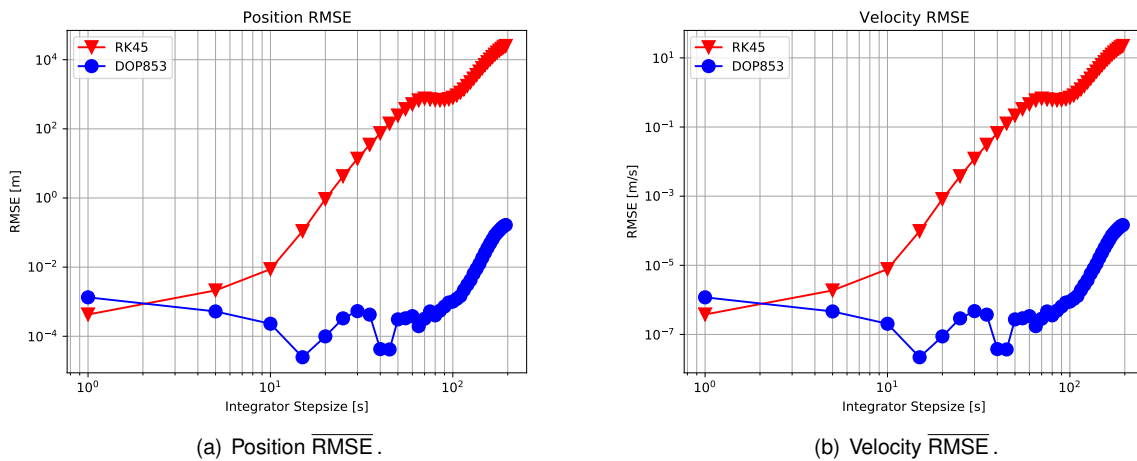


Figure 4.2: Averaged position and velocity  $\overline{\text{RMSE}}$  as a function of the integration step size, for a Keplerian Orbit propagation during 20 days. Numerical results are compared with the analytical true solution.

For the considered orbit, both integrators perform equally well with small integration step sizes (below 5 seconds). However, when the step increases DOP853 clearly outperforms RK45 by several orders of magnitude. For instance, for a step of 60 seconds, the position error of RK45 is around 800 m, whereas DOP853 has an error of approximately 0.1 mm.

As mentioned before, OD systems require small integration steps. In fact, the designed filtering

simulation environment is set up with a fixed step of 5 seconds. With this step, both integrators produce similar results, however RK45 is lighter [61]. Therefore, RK45 is the chosen numerical solver, to be used in the remaining simulations of this chapter, set with relative and absolute tolerances of  $10^{-3}$  and  $10^{-6}$ , respectively.

### Accuracy of Cowell Propagation

After defining the numerical solver, the developed Cowell propagator may now be compared to the GMAT's. Two experiments are conducted.

First, the implementation of the gravitational field is tested. Both propagators are set up with the same force model, that is, a low degree and order gravitational field, and propagate the same initial orbital conditions. The averaged position and velocity  $\overline{\text{RMSE}}$  between the two propagators is shown in Table 4.5, for a 20-day propagation period. It is observed that the two propagation schemes produce similar outputs, since the errors are small. Although there is no force model mismatch between the propagators, their computer implementation is different. Differences also arise in the implementation of the transformations between ECI and ECEF frames and due to the numerical solvers. However, it is noted that these differences are small.

Grav. Field	$\overline{\text{RMSE}}$	
	Position [m]	Velocity [m/s]
$2 \times 0$	3.1773	$3.3073 \times 10^{-3}$
$5 \times 0$	3.1846	$3.3146 \times 10^{-3}$
$5 \times 5$	3.1875	$3.3176 \times 10^{-3}$
$10 \times 0$	3.1857	$3.3157 \times 10^{-3}$
$10 \times 10$	3.1819	$3.3117 \times 10^{-3}$

Table 4.5: Gravitational field study. This table shows the averaged position and velocity  $\overline{\text{RMSE}}$  of the developed Cowell propagator compared to GMAT's. Both propagators are set up with the same force model.

Then, reduced force models implemented by the developed Cowell propagator are compared to the full-force model of the GMAT propagator. The time evolution of the position and velocity RMSE is depicted in Figure 4.3, for a variety of reduced force models, considering a  $10T$  propagation length, where the orbital period  $T$  is approximately 100 minutes. It is noted that higher gravitational fields yield smaller differences<sup>3</sup>.

The simulations performed in this section evidence that the mismatch between realistic and simplified force models is a great source of error in orbital propagation. Indeed, the use of simplified force models creates great uncertainty in the propagation model. Nevertheless, the purpose of using Orbit Determination filters is to relax the dynamical models, when non high-precision mission requirements are set, i.e., the inclusion of measurements enables a simplification in the state dynamics, since the uncertainty in the propagation is compensated by the observations.

<sup>3</sup>It is also noted that the atmospheric models are different: the developed Cowell propagator uses a simplistic exponential atmosphere, whereas GMAT uses a more sophisticated and accurate model (Jacchia Roberts).

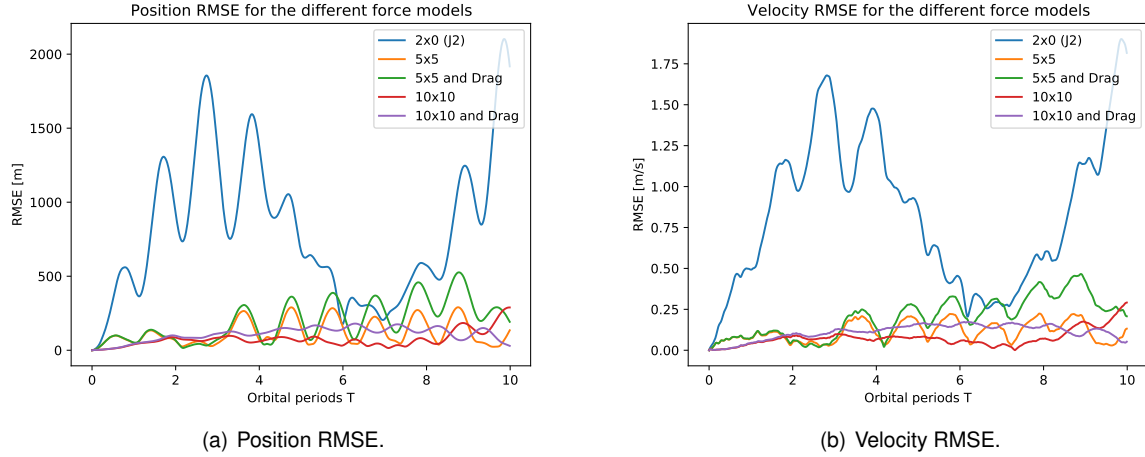


Figure 4.3: Full force model study. Time evolution of position and velocity RMSE for the developed Cowell propagator is plotted for a variety of force models.

The dynamical model of the Cowell propagator is therefore tuned with a  $5 \times 5$  gravitational field and atmospheric drag.

### Accuracy of Semianalytical Propagation

In this section, the developed Semianalytical propagator is tested. First, the length of the DFT associated to zonal/drag perturbations,  $N_{Z,D}$ , and the lengths  $N_T$  and  $M_T$  associated to tesseral/sectoral perturbations need to be tuned. The Semianalytical propagator is also tuned with a  $5 \times 5$  gravitational field and drag (note that if the model is changed, then the DFT lengths need to be retuned). Furthermore, the order of the quadrature  $N_{\text{quad}}$  also needs tuning. After these parameters are selected, comparative simulations with DSST are performed. For consistency, the DSST propagator is set up with an equivalent force model.

The numerical solver for the mean element propagation is also RK45, with relative and absolute tolerances set to  $10^{-3}$  and  $10^{-6}$ , respectively. However, the integration step size is now set to 1 day.

#### Tuning parameters

Following the tuning procedure described by Ely in [29], the parameters  $N_{Z,D}$ ,  $N_T$ ,  $M_T$  and  $N_{\text{quad}}$  are found by trial and error. The idea is to define a variation grid for these parameters and then propagate the same initial condition (defined in Table 4.1), with both Cowell and Semianalytical propagators, for all the selected cases. Then, by analyzing the difference between the osculating elements obtained with Cowell and Semianalytical propagation, one selects the smallest values for these parameters that provide the smallest differences. Note that the difference between the two propagation models is only associated to the averaging procedure (averaging the osculating element rates and computing the short-periodic functions), hence by minimizing this error, one obtains the best match between the two models. For example, Figure 4.4 illustrates the difference in semi-major axis  $a$ , referring to the zonal/drag perturbations, for different  $N_{Z,D}$  values. Clearly, the differences in  $a$  for  $N_{Z,D} = 16$  (subplot (b)) and  $N_{Z,D} = 32$



(subplot (c)) are very similar, however for  $N_{Z,D} = 8$  (subplot (a)) the error is much worse.

The following values were found to yield good results:  $N_{Z,D} = 16$ ,  $N_T = 16$ ,  $M_T = 16$  and  $N_{\text{quad}} = 20$ . Fortunately, once these parameters are found for a particular orbit, they remain relatively constant when analyzing nearby orbits [29], which is inevitable in estimation problems. It is noted that using greater values would not give better results but would be more computationally expensive. Moreover, DFT lengths are chosen to be powers of 2, since FFT evaluations are more efficient in such cases [52].

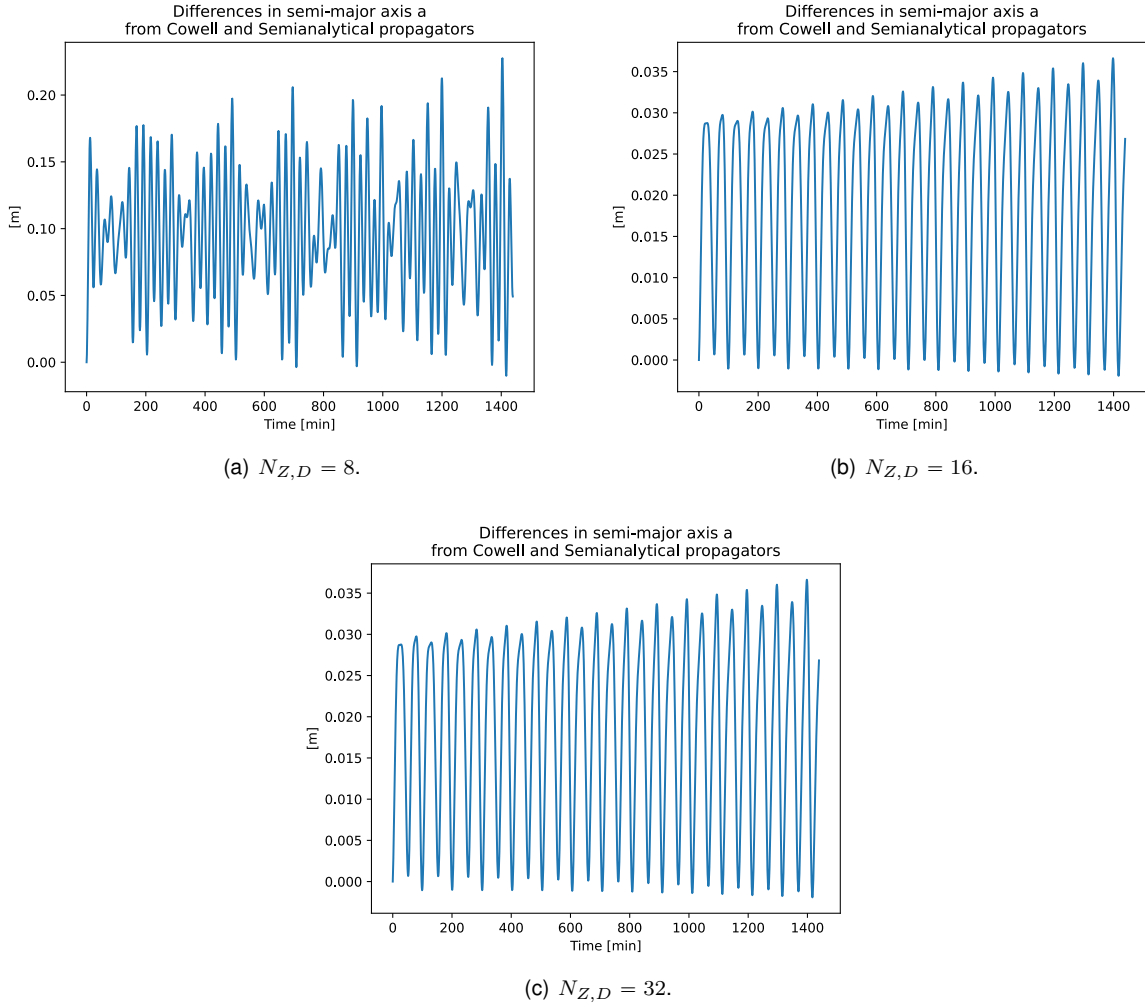


Figure 4.4: Tuning the DFT length for zonal (up to degree 5) and drag perturbations ( $N_{Z,D}$ ). Lengths of 8, 16 and 32 are respectively given in subplots (a), (b) and (c). It is clear that no visible differences exist between (b) and (c), which suggests that an order of 16 is sufficient. Results are shown for a 1-day simulation.

### Comparing with DSST

After tuning the DFT and numerical quadrature, the Semianalytical mean element propagation is compared with the equivalent propagation implemented in DSST. The averaged  $\overline{\text{RMSE}}$  for each equinoctial mean element is shown in Table 4.6 for a 7500-day propagation<sup>4</sup>. It is emphasized that, in this simulation

<sup>4</sup>Since the mean elements are slowly varying, yielding only secular and long periodic variations, the simulation interval needs to be extended for a large propagation period [64].

the outputs are compared in the mean space, rather than in the osculating. The results obtained are in line with similar propagation comparisons performed in [64].

Equinoctial mean element	RMSE
$\bar{a}$	$5.152 \times 10^{-10} \text{ km}$
$\bar{h}$	$2.033 \times 10^{-6}$
$\bar{k}$	$2.087 \times 10^{-6}$
$\bar{p}$	$4.665 \times 10^{-5}$
$\bar{q}$	$4.562 \times 10^{-5}$
$\bar{\lambda}$	$3.235 \times 10^{-6} \text{ rad}$

Table 4.6:  $\overline{\text{RMSE}}$  of mean element propagation of the developed Semianalytical propagator relative to DSST (assumed the true trajectory) for a 7500-day simulation period, using a  $5 \times 5$  gravitational field and drag.

Next, after validating the mean element propagation, the *mean-to-osculating map* is tested. With that in mind, the osculating trajectories outputted by the developed Semianalytical propagator and DSST are compared to the Cowell trajectory. The time evolution of the osculating position and velocity RMSE is depicted in Figure 4.5 for a 1-day propagation, taking the Cowell trajectory (obtained with the developed Cowell propagator) as reference. Furthermore, an analytical *mean-to-osculating map*, provided by Brouwer analytical theory (formulated in Appendix G of [65]), coupled with the mean element propagation provided by the developed Semianalytical propagator, is shown for comparison. This analytical map comprises a  $2 \times 0$  (J2) gravitational field.

The simulation evidences that the DSST's *mean-to-osculating map* captures the short-periodic variations better than the map from the developed Semianalytical model. This may be explained by the fact that the DSST's *mean-to-osculating map* is more complete, since it contains second-order treatment of some perturbations [12, 27], whereas the map of the developed propagator was truncated to first-order (cf. Section 3.1.2). Nevertheless, both maps clearly outperform the analytical *mean-to-osculating map* provided by the Brouwer analytical theory.

## Computation Cost

The propagation analysis is concluded with a CPU computation time test. As mentioned before, Semianalytical propagation is expected to be much lighter than Cowell propagation.

This efficiency is corroborated by the results obtained in this section. Figure 4.6 depicts the CPU computation time for different propagation periods (from 1-day to 7-day propagation). The trajectory is outputted at a requested fixed step of 60 seconds.

The efficiency of the Semianalytical scheme is further improved if the output interval is enlarged. Table 4.7 presents the Semianalytical propagation computation time for different requested output steps, for a 7-day simulation. With this configuration, Cowell propagation needs approximately 90 seconds (cf. Figure 4.6).

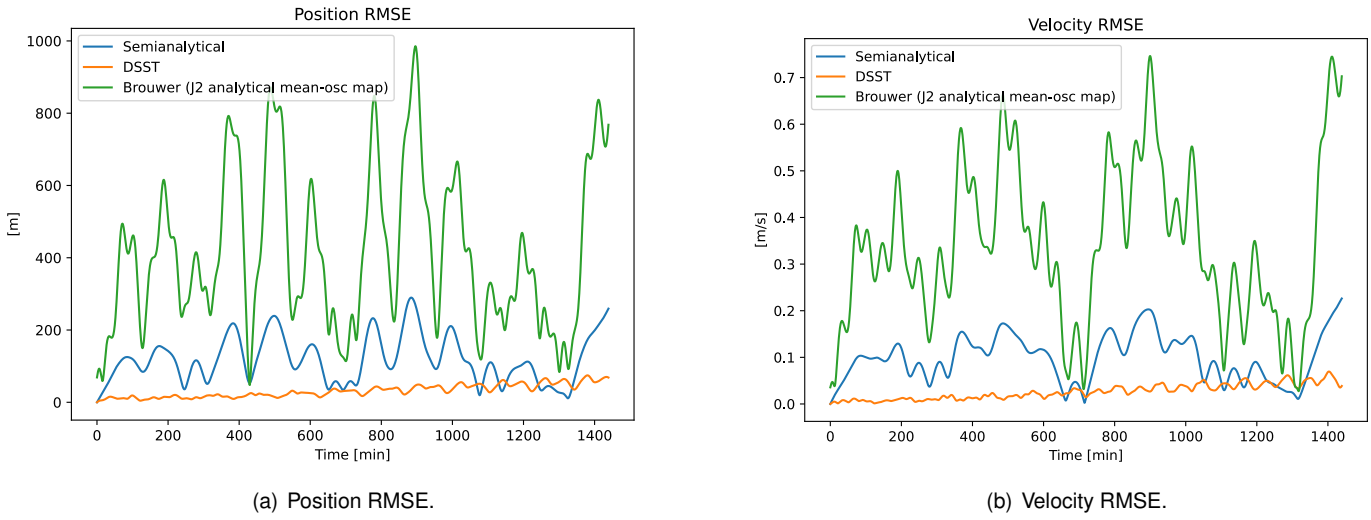


Figure 4.5: Position (a) and velocity (b) osculating RMSE of different propagation theories compared with Cowell propagation (assumed the true reference trajectory). Orange and blue lines represent, respectively, the error provided by DSST and the developed Semianalytical propagators, both tuned with a  $5 \times 5$  gravitational field and drag. The green line illustrates an analytical *mean-to-osculating map*, provided by the Brouwer analytical theory [65] (yielding only the J2 perturbation) coupled with the mean element propagation of the developed Semianalytical propagator. The initial osculating conditions are provided in Table 4.1. 1-day propagation.

These results are in line with similar comparisons made between Cowell and DSST propagation in [12].

To conclude, an efficient Semianalytical implementation was achieved. Nonetheless, this efficiency comes with a slight loss in accuracy, when compared to Cowell propagation with a similar force model. Depending on the mission requirements, the propagator that offers the best trade-off between accuracy and computational effort is to be favored.

Requested output step	Semianalytical propagation CPU computation time [s]
5 s	65.54
60 s	15.07
100 s	9.88
1000 s	2.38
10000 s	1.59
half a day	1.51
1 day (1 integration step)	1.50

Table 4.7: CPU computation time for Semianalytical propagation with different output requested steps, for a 7-day propagation period and force model comprising a  $5 \times 5$  gravitational field and drag.

Finally, both the developed Cowell and Semianalytical propagators seem to hold satisfactory results and acceptable performance when compared to propagation schemes from mature and established software packages, namely, GMAT's Cowell propagator and Orekit's DSST propagator.

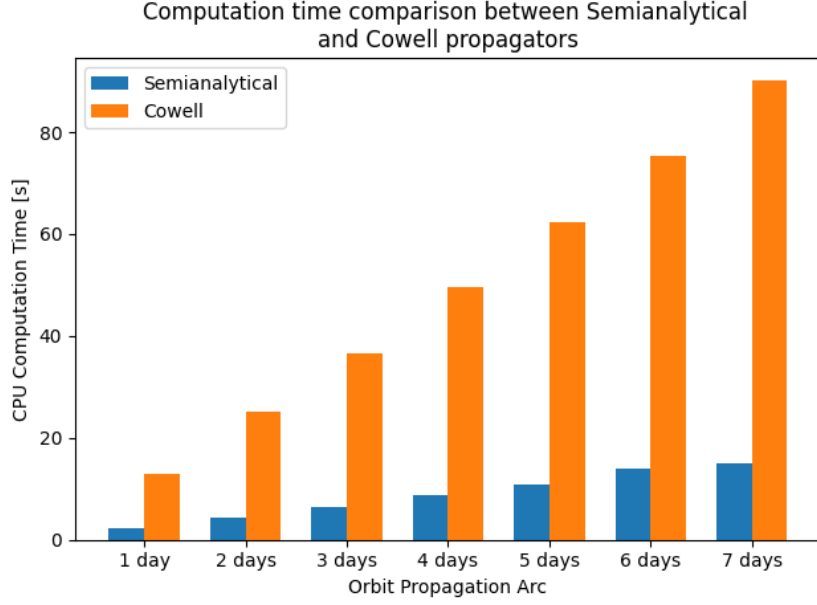


Figure 4.6: CPU computation time comparison between the developed Cowell and Semianalytical propagators, for propagation arcs from 1 day to 7 days. The orbital outputs are requested at a step of 60 seconds. The force model used is a  $5 \times 5$  gravitational field with drag.

#### 4.1.2 Filter Results

Having tested the performance of the propagation schemes alone, an evaluation of Orbit Determination filters is now sought. Two Cowell filters are considered: the Extended Cowell Kalman Filter (denoted as EKF) and the Unscented Cowell Kalman Filter (denoted as UKF). Similarly, two Semianalytical filters are considered: the Extended Semianalytical Kalman Filter (ESKF) and the Unscented Semianalytical Kalman Filter (USKF), introduced in Section 3.4. As mentioned before, the propagation dynamical force model comprises a  $5 \times 5$  gravitational field with atmospheric drag. The simulations run at a fixed step of 5 seconds, i.e., the ground stations produce, if available, observations once every 5 seconds.

The measurement noise covariance matrix  $R$ , to be used in the filter update steps, is readily constructed referring to the same standard deviations considered in the measurement noise generation of Table 4.2, in accordance to Section 3.2.1. Notice that, in this simulation setup, only one ground station, located in Lisbon, is considered.

Determination of the process noise covariance matrix  $Q$  is relatively more complex. A trial and error approach was used to define its diagonal entries, adjusting the values manually with the help of simulations. The following osculating covariance, in state-vector form, was found

$$Q_x = \begin{bmatrix} 10^{-9} I_{3 \times 3} & \mathbf{0}_{3 \times 3} \\ \mathbf{0}_{3 \times 3} & 10^{-12} I_{3 \times 3} \end{bmatrix}, \quad (4.2)$$

where the velocity and acceleration entries are given, respectively, in  $\text{km s}^{-1}$  and  $\text{km s}^{-2}$ .

This covariance is then transformed to the mean equinoctial space for use with Semianalytical filters, using a transformation similar to (3.69), yielding  $Q_{\bar{g}}$ . This transformation ensures equivalent process

noise between the osculating and mean dynamics, which allows for better comparison of filter results [25]. It is however noted that this transformation is time dependent, since the partials depend on the orbital state, hence it needs to be determined for each predict-step evaluation.

Filter initialization was described in Section 3.4.3. In order to choose the length  $N$  of the set of observations to be collected for the initialization procedure, simulations with different lengths were conducted. Figure 4.7 shows the position and velocity RMSE of the initial osculating estimate for different values of  $N$ . A total of 100 Monte Carlo (MC) runs were performed. Since the purpose of the initialization procedure is to simply produce a rough estimate that guarantees filter convergence, a length of  $N = 15$  was chosen.

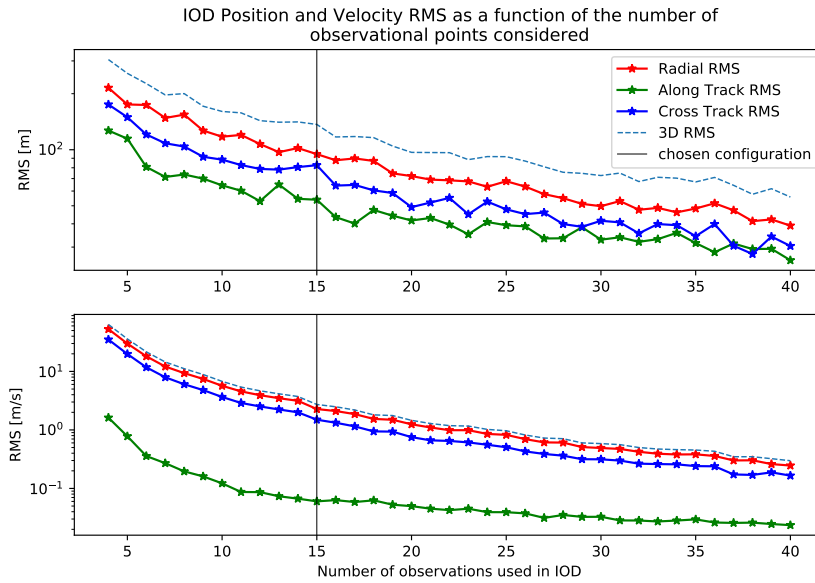


Figure 4.7: Position (above) and velocity (below) RMSE of the initialization procedure as a function of the length of the IOD observation set, for 100 MC runs. Besides the 3D position/velocity RMSE, the error in each component of RSW is also represented.

Figure 4.8 presents the time evolution of the position RMSE for a 7-day filtering simulation using the four filters described (only one MC run was made). Table 4.8 further provides the trajectory averaged position and velocity  $\overline{\text{RMSE}}$  for the same simulation, as well as the error in each component of the RSW frame. The errors are expressed in terms of the osculating quantities.

In this simulation, Semianalytical filters outperformed Cowell filters, with a position accuracy gain of approximately 300 m. The superiority of Semianalytical filters may be explained by the fact that the mean dynamics, estimated in Semianalytical filters, are more linear than the osculating dynamics, estimated in Cowell filters. Therefore, the former better satisfy the linearization procedures of filtering predict steps [15]. Furthermore, previous research has shown that propagating covariances in equinoctial elements is more advantageous than propagating them in cartesian state space [12]. Moreover, it is also evident that the UKF slightly outperforms the EKF and the USKF slightly outperforms the ESKF, which reveals the advantages of the Unscented Transform over the Taylor series linearization in the nonlinear transformation of the state and measurement models.

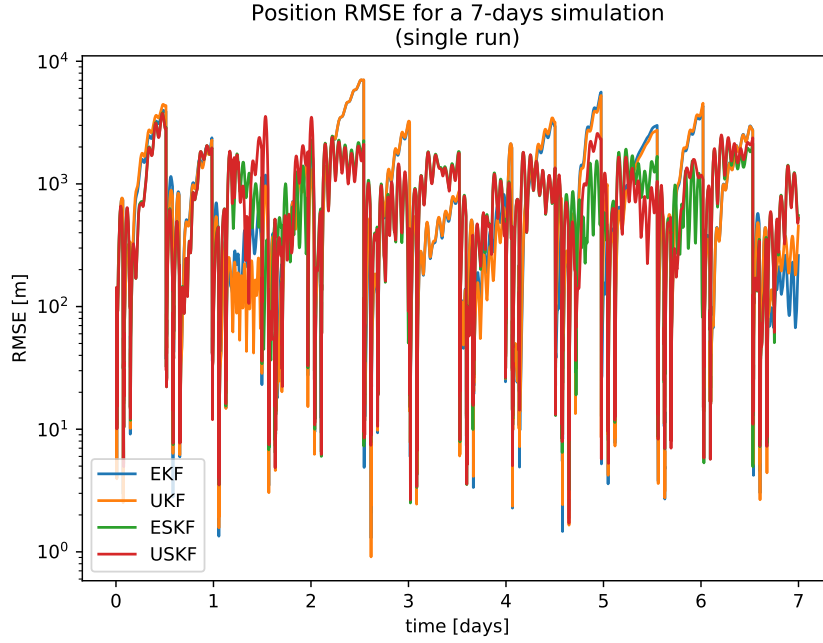


Figure 4.8: Time evolution of the position RMSE of the filtered trajectories outputted by EKF (blue), UKF (yellow), ESKF (green) and USKF (red) relative to the reference trajectory, for a single MC run. The simulation length is 7 days.

Filter	Position RMSE [m]				Velocity RMSE [m/s]			
	R	S	W	3D	R	S	W	3D
EKF	59.32	878.28	35.02	880.97	0.9198	0.0469	0.0717	0.9237
UKF	59.51	871.41	35.32	874.15	0.9143	0.0623	0.0756	0.9195
ESKF	94.91	582.10	89.81	596.59	0.6422	0.1017	0.1854	0.6761
USKF	92.44	537.54	88.86	552.62	0.5891	0.0999	0.1824	0.6248

Table 4.8: Trajectory averaged position and velocity  $\overline{\text{RMSE}}$  for the same simulation run illustrated in Figure 4.8. Besides the 3D position and velocity errors, the error is also provided in each component of the RSW frame (R - Radial, S - Along-Track and W - Cross-Track).

A closer examination of Table 4.8 reveals that the major contribution to the position error comes from the Along-Track (S) direction. This may be explained by asymmetries induced due to the atmospheric drag models, since the model used in the reference trajectory is much more accurate than the simplistic model used in the filter dynamics. It is emphasized that drag acts on the Along-Track direction, parallel to the velocity vector, and is one of the main orbital perturbations at low altitudes. Furthermore, when the satellite is at low elevation angles (relative to the ground station), small vertical errors can result in large uncertainty about where the satellite is in its orbit, which further accentuates the Along-Track error [3].

Figure 4.9 plots the time evolution of the position RMSE when the considered estimate is simply given by the inverse of the measurement function (using only range, azimuth and elevation observations), for a single ground station pass. Clearly, the error provided by these rough and unfiltered estimates is

substantially worse than the error obtained with filter estimates, which further exhibits the performance and relevance of the implemented algorithms.

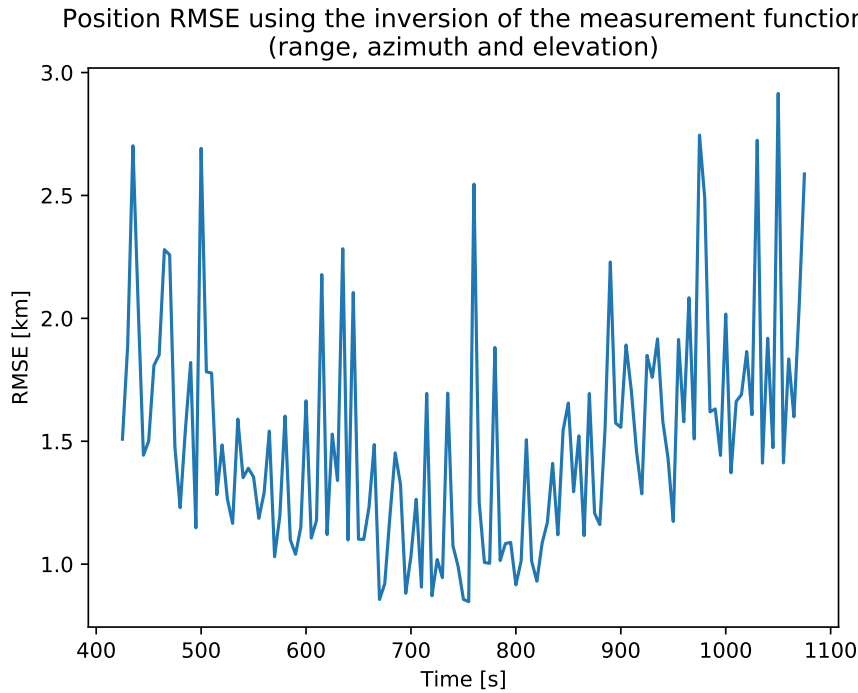


Figure 4.9: Position RMSE considering orbital estimates given directly by the inverse of the measurement function (with range, azimuth and elevation), for a single ground station pass. 100 MC simulations were ran.

### Initial Error Analysis

In the simulations provided above, the filters were initialized with the IOD initialization procedure, which provided initial position and velocity errors of approximately 200 m and  $2 \text{ m s}^{-1}$ , respectively, cf. Figure 4.7. Nonetheless, it is important to test how sensitive the filters are to initial errors, since bad estimates may lead to filter divergence. With that in mind, in the next simulation, the initialization procedure is replaced by direct initialization of the filters with known Gaussian distributions. For each MC run, the initial estimate  $\hat{\mathbf{x}}(t_0)$  of the state  $\mathbf{x}(t_0)$  is computed as a realization of the following distribution

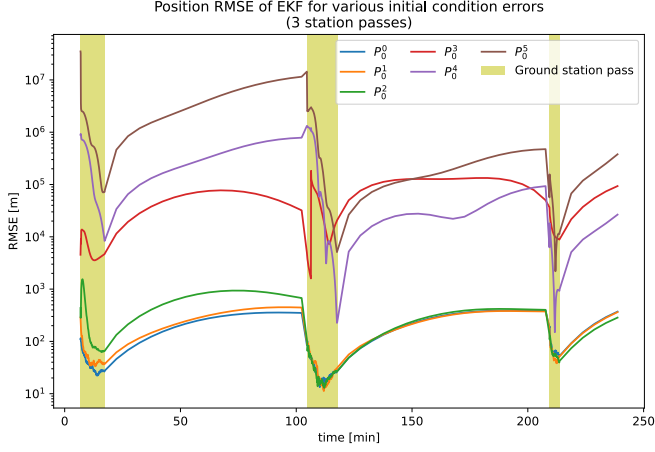
$$\hat{\mathbf{x}}(t_0) \sim \mathcal{N}(\mathbf{x}(t_0), \kappa \text{diag}(0.1, 0.1, 0.1, 10^{-4}, 10^{-4}, 10^{-4})) , \quad (4.3)$$

where the position and velocity components are expressed, respectively, in km and  $\text{km s}^{-1}$ . Different initial error covariances (simulation environments) are thus created by varying  $\kappa$ . The designed covariances ( $P^0$  to  $P^5$ ) are shown in Table 4.9 as a function of  $\kappa$ .

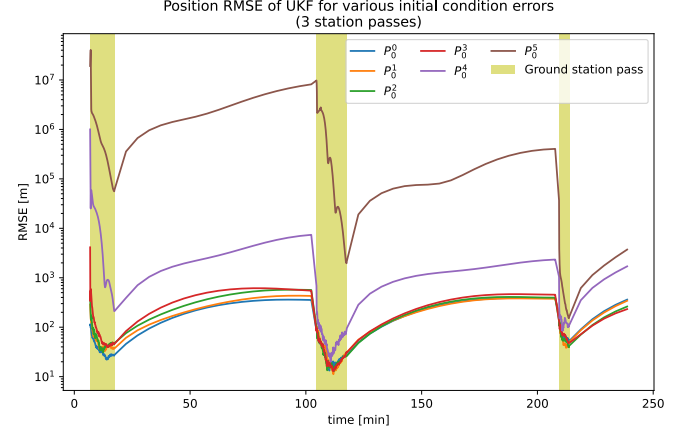
	$P^0$	$P^1$	$P^2$	$P^3$	$P^4$	$P^5$
$\kappa$	1	10	100	1000	10000	100000

Table 4.9: Simulation environments for the initial error analysis.

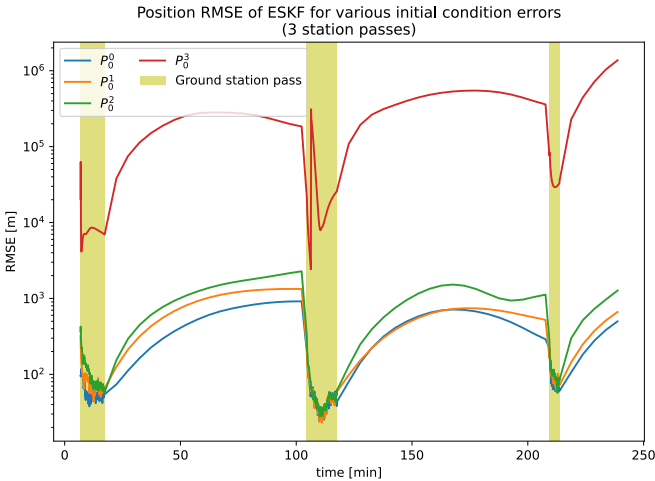
Simulation results for each environment are shown in Figure 4.10 for 50 MC runs, in terms of position RMSE. The simulations encompass a period of 3 ground station passes, highlighted in yellow in the plots.<sup>5</sup>



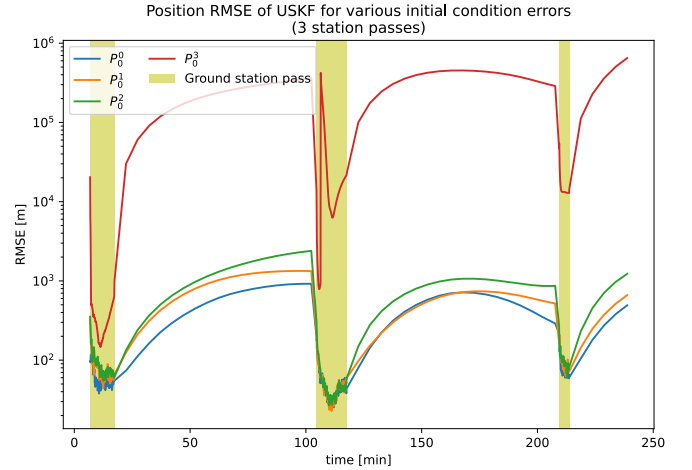
(a) EKF.



(b) UKF.



(c) ESKF.



(d) USKF.

Figure 4.10: Time evolution of the position RMSE for EKF (a), UKF (b), ESKF (c) and USKF (d), considering different initial error environments (cf. Table 4.9). The 50 MC runs last for 3 station passes (highlighted in yellow).  $P^0$  - blue lines,  $P^1$  - orange lines,  $P^2$  - green lines,  $P^3$  - red lines,  $P^4$  - purple lines,  $P^5$  - brown lines.

By looking at the time evolution of the RMSE, it is observed that, as expected, the error only reduces (the estimated trajectory converges to the true one) when there exists line of sight with the station. In the absence of update steps, the error grows due to the propagation uncertainty caused by the mismatch between the simplified and real force models. Comparing the EKF and UKF filters, it is seen that the latter is more robust to initial errors, since, at the end of the third passage, the error for all initial conditions converged to the interval between 50 m and 100 m, whereas the EKF produced an error of 10 km when

<sup>5</sup>The authors of [66] suggest that there are no substantial differences in accuracy and filter performance after three or four station passes.



$P^4$  and  $P^5$  were tested. Nevertheless, it is apparent that both Cowell filters are able to converge for all the tested cases, even if the initial estimate is poor.

On the other hand, the performance of the Semianalytical filters is not as good when adverse initial conditions are considered. It is noted that most of the simulations running with  $P^4$  and  $P^5$  did not converge, hence they are not shown. Nonetheless, when using  $P^0$ ,  $P^1$  and  $P^2$  the obtained results are similar to those of Cowell filters, with USKF slightly outperforming ESKF.

To conclude, the results suggest that Semianalytical filters need better initial estimates than Cowell filters. The existence of long periods of station unavailability may explain this drawback, since the estimation of mean trajectories in Semianalytical filters is expected to be slightly slower to converge than the estimation of osculating trajectories in Cowell filters, due to the inherent averaging procedure associated to the mean dynamics. It is noted that GS passes last, approximately, between 10 to 15 minutes, whereas one orbital period is  $\sim 100$  minutes.

### Observability Analysis

To better understand the importance of the measurement types and how they affect the performance of the different filtering algorithms, an observability analysis is conducted. With that in mind, three measurement functions are created, considering different configurations of measurement types. They are summarized in Table 4.10.

Sensor configurations	Model
Range, azimuth, elevation, range-rate	$\mathbf{h}_d^{(1)} = [s, \beta, \delta, \dot{s}]^T$
Range, azimuth, elevation	$\mathbf{h}_d^{(2)} = [s, \beta, \delta]^T$
Azimuth, elevation	$\mathbf{h}_d^{(3)} = [\beta, \delta]$

Table 4.10: Measurement models for the observability analysis.

Generally, a system is said to be observable if, for any possible evolution of state and control vectors, the current state can be estimated using only the available measurement information from sensors [67]. Observability of nonlinear systems is thoroughly studied in [68]. It is practically impossible to perform a global observability test for general nonlinear problems. For that reason, local observability about a given state or equilibrium point is usually considered [38]. Therefore, observability is evaluated by performing a test on the linearized time-variant system, which is only guaranteed within a neighborhood of the evaluated state, and, according to [38], is usually adequate for observability analyses.

The test consists of evaluating the observability of the state  $\mathbf{x}(t_0)$  by means of the observability Gramian,  $G$ , which is defined, considering a measurement interval of  $d + 1$  discrete observations from  $t_0$  to  $t_d$ , as [50]

$$G(\mathbf{x}(t_d)) = \sum_{k=0}^d \Phi^T(t_k, t_0) \mathbf{H}^T(\mathbf{x}(t_k), t_k) \mathbf{H}(\mathbf{x}(t_k), t_k) \Phi(t_k, t_0), \quad (4.4)$$

where  $\Phi(\cdot)$  is the State Transition Matrix (STM) of the linearized system (only osculating dynamics are considered in this analysis).

Then, the linearized system is said to be locally observable if and only if the Gramian  $G(\mathbf{x}(t_d))$  is nonsingular, i.e., full-rank  $n$  (where  $n$  is the dimension of the state vector). Notice in Equation (4.4) that, since the rank of each term in the summation is at most  $m$ , (where  $m$  is the dimension of the measurement vector), there is a minimum number  $d \geq n/m$  of measurements that must be processed before  $G(\mathbf{x}(t_d))$  may become full-rank  $n$ . However, once satisfied,  $G(\mathbf{x}(t_d))$  remains full rank. Consequently, this measure offers little insight once the rank condition is satisfied [50]. According to [67], the calculation of the matrix condition number provides a more robust determination of the inherent ill-conditioning for a given observability matrix. The matrix condition number coefficient  $\kappa$  is defined as

$$\kappa(\mathbf{x}(t_d)) = \frac{|\lambda_{\max}(\mathbf{G}, \mathbf{x}(t_d))|}{|\lambda_{\min}(\mathbf{G}, \mathbf{x}(t_d))|}, \quad (4.5)$$

where  $\lambda_{\max}(\mathbf{G}, \mathbf{x}(t_d))$  and  $\lambda_{\min}(\mathbf{G}, \mathbf{x}(t_d))$  denote the maximum and minimum eigenvalues of  $\mathbf{G}(\mathbf{x}(t_d))$ , respectively. A small value for  $\kappa$  indicates good observability (well-conditioned Gramian), whereas a large value indicates poor observability (ill-conditioned matrix). As  $\lambda_{\min}$  approaches zero,  $\kappa$  approaches infinity and the matrix becomes closer to being singular.

Figure 4.11 plots the time evolution of the Gramian's condition number for a single ground station pass, using the three measurement models defined in Table 4.10. Furthermore, Figure 4.12 shows the trajectory averaged steady-state RMSE in terms of RSW components after one ground station pass, using the same measurement models.

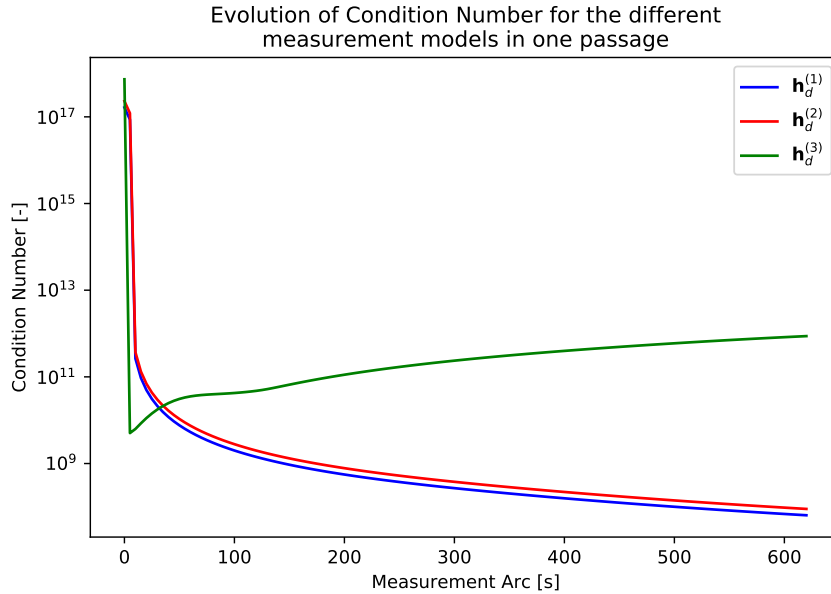
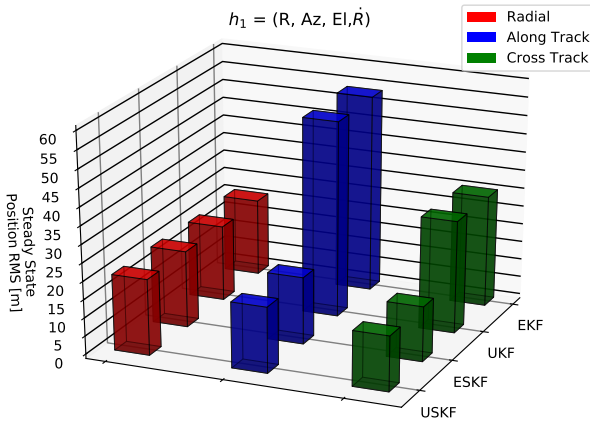


Figure 4.11: Evolution of the condition number of the observability Gramian in one passage versus the measurement arc length.

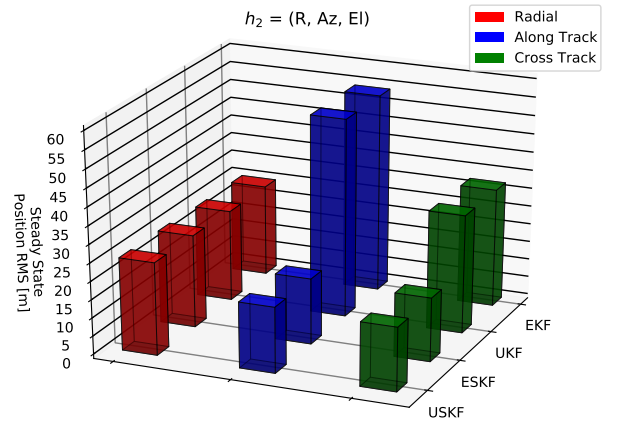
The condition number  $\kappa$  tends to stabilize for all three models. The most observable (in view of this metric) is  $h_d^{(1)}$  followed closely by  $h_d^{(2)}$ .  $h_d^{(3)}$  is, by several orders of magnitude, the least observable. One can glimpse that the lack of information on the Range direction greatly degrades filtering performance. Furthermore, since the condition numbers of  $h_d^{(1)}$  and  $h_d^{(2)}$  are quite close for all time instants, one

can infer that the addition of range-rate measurements does not provide any substantial gain, when compared to range alone.

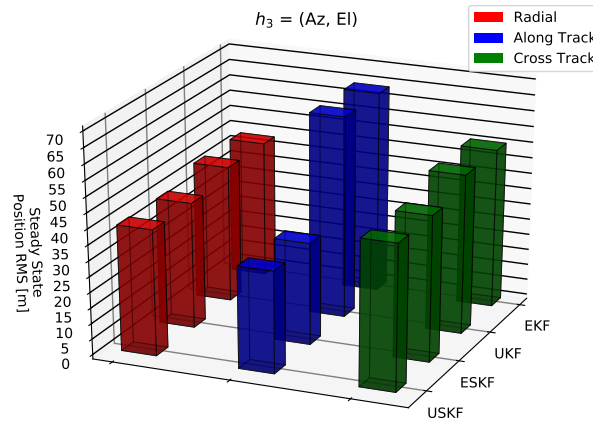
These remarks are corroborated by the  $\overline{\text{RMSE}}$  filtering results of Figure 4.12. In fact, the loss of range and range-rate data deteriorates the OD solution. Further studies could be made in order to infer the reason why the error on the Along-Track direction is substantially bigger for Cowell filters than it is for Semianalytical filters, when the same measurement model is considered.



(a) Measurement model  $\mathbf{h}_d^{(1)}$ .



(b) Measurement model  $\mathbf{h}_d^{(2)}$ .



(c) Measurement model  $\mathbf{h}_d^{(3)}$ .

Figure 4.12: Steady state averaged  $\overline{\text{RMSE}}$  after one ground station pass considering the measurement models of Table 4.10  $\mathbf{h}_d^{(1)}$  in (a),  $\mathbf{h}_d^{(2)}$  in (b) and  $\mathbf{h}_d^{(3)}$  in (c). Errors are provided for each component of the RSW frame. 50 MC runs.

## Final Remarks

There exist several strategies to improve filter performance. First, sensor noise statistics could be tweaked, as there is some latitude for choosing suitable covariances for tracking filters (effectively regarding them as "tuning knobs") when operating with a given set of sensors. In fact, experiences performed in [66] showed the existence of a zone with linear dependency between measurement noise level and Orbit Determination accuracy. However, there exists a technological limitation to how much these statistics may drop down. Another approach could be to use data from several ground stations [69]. However, the use of several stations may not significantly increase OD accuracy. In fact, sensors are often biased<sup>6</sup> and need to be properly tuned, which adds to the difficulty of accurate estimation. Another approach is to increase the frequency of measurement data [3]. Indeed, Fonte showed in [70] that dense, real-world observations from a single station could produce orbits accurate to less than 10 meters for a 12-hour prediction, at low altitudes.

Next, estimation results with multiple ground stations are presented.

## 4.2 Test Case 2 - Multiple Ground Stations

To conclude the simulation tests, a more in-depth comparison between the ESKF and USKF is furthered. The simulation environment contains four ground stations. Table 4.11 shows the geodetic location of these stations, as well as their associated measurement noise standard deviations. Range-rate is excluded. The initial mean orbital state and epoch are provided in Table 4.12. The modeling used to generate the reference true trajectory includes  $30 \times 30$  geopotential terms, Jacchia-Roberts atmospheric drag, lunar and solar point mass gravity and Earth polar motion. In this experiment, the reference mean and osculating trajectories are obtained with DSST. The reference mean trajectory is used to compare the accuracy of the filter estimates, and the reference osculating trajectory is used to compute the noisy measurements that are then fed to the filtering algorithms. The simulation setup just described is similar to the one defined in [26], where the performance of the ESKF, coupled with DSST, was evaluated. Hence, this experience is intended to serve as a baseline to compare the novel USKF proposed in this thesis with other existing Semianalytical filters.

The filtering dynamical model, satellite constants, initialization procedure and process noise covariances are the same as described for Test Case 1 in the previous section.

Figure 4.13 depicts the test case simulation results, in terms of absolute error time evolution of mean equinoctial elements (true elements minus estimate elements), for a 6-day simulation. Furthermore, the squared roots of the diagonal entries (associated to each element) of the estimated covariance are also shown. These values represent the filtering online estimation error.

It is noted that the behavior of the ESKF (in terms of the absolute error time evolution of the mean elements) is comparable to the one obtained in the referenced simulation test of [26].

---

<sup>6</sup>It is noted that, for simplicity, measurement bias was not considered in this thesis. In real-world applications bias is typically unknown, and hence usually included in the solve-for estimation state, which adds more uncertainty to the overall filter performance. Fonte suggests in [70] that using incorrect biases can produce almost a 70 meter error in LEO satellite filtering.

Ground Station	Geodetic Coordinates ( $h, \phi_{gd}, \lambda$ )	Measurement Noise Standard Deviations
1	( $h = 0 \text{ m}, \phi_{gd} = 42.617^\circ, \lambda = 288.509^\circ$ )	10 m for Range 0.005° for Azimuth and Elevation
2	( $h = 0 \text{ m}, \phi_{gd} = 42.623^\circ, \lambda = 288.512^\circ$ )	10 m for Range 0.005° for Azimuth and Elevation
3	( $h = 0 \text{ m}, \phi_{gd} = 9.395^\circ, \lambda = 167.479^\circ$ )	10 m for Range 0.005° for Azimuth and Elevation
4	( $h = 0 \text{ m}, \phi_{gd} = 21.572^\circ, \lambda = 201.733^\circ$ )	23 m for Range 0.019° for Azimuth and Elevation

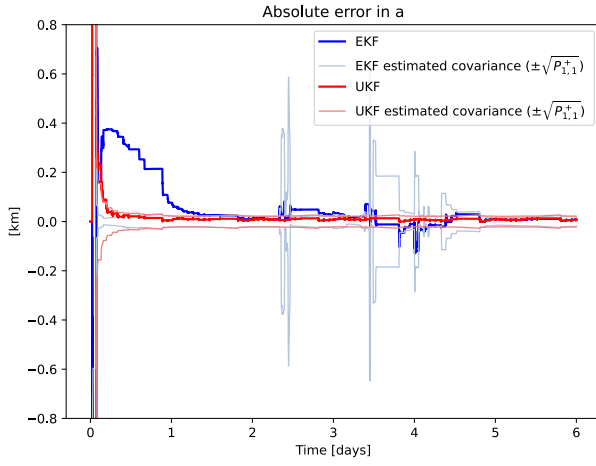
Table 4.11: Location of the ground stations and their associated measurement noise standard deviations for Test Case 2.

Initial orbital conditions	
Epoch	18 January, 2003 00:00:00 UTC
Mean Keplerian set	(6643 km, $8.9 \times 10^{-2}$ , $38^\circ$ , $214^\circ$ , $344^\circ$ , $74^\circ$ )
	( $\bar{a}, \bar{e}, \bar{i}, \bar{\Omega}, \bar{\omega}, \bar{\nu}$ )

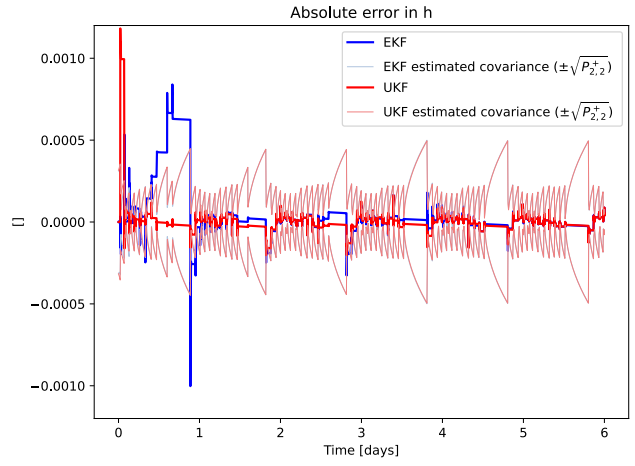
Table 4.12: Initial epoch and orbital state for Test Case 2.

Analyzing the estimation results of Figure 4.13, the superiority of the USKF is evidenced. On one hand, the USKF converged faster to the true trajectory, for instance, in the estimation of elements  $\bar{a}$  (Figure 4.13 (a)),  $\bar{k}$  (Figure 4.13 (c)) or  $\bar{\lambda}$  (Figure 4.13 (f)). On the other hand, the USKF filter estimates tended to suffer less destabilization than the ESKF estimates, indicating that the USKF is more stable and robust than the ESKF. Furthermore, it better propagated the mean trajectory in the periods of station unavailability.

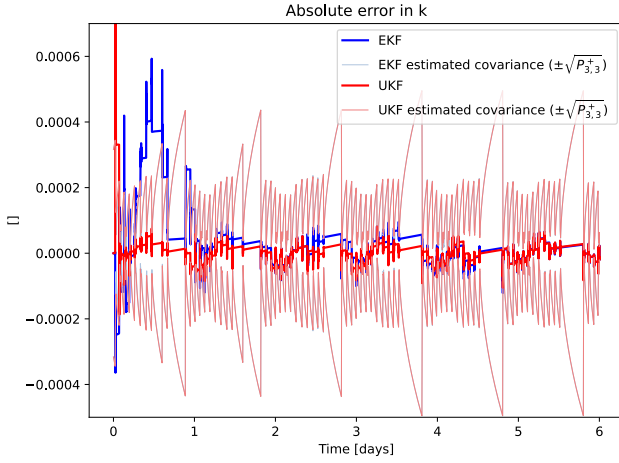
Note that Test Case 2 may not be compared directly with Test Case 1 since: 1) the simulation environment is not the same (in terms of measurement noise statistics and reference orbit generation), and 2) the filter results are not evaluated in terms of the same error quantities and metrics. However, the existence of more ground stations, distributed along the Earth's surface, creates more periods of station availability (and hence more observations to process), thus preventing filter estimates to further diverge from the true trajectory, in view of the reduced force model used in the filter dynamical model.



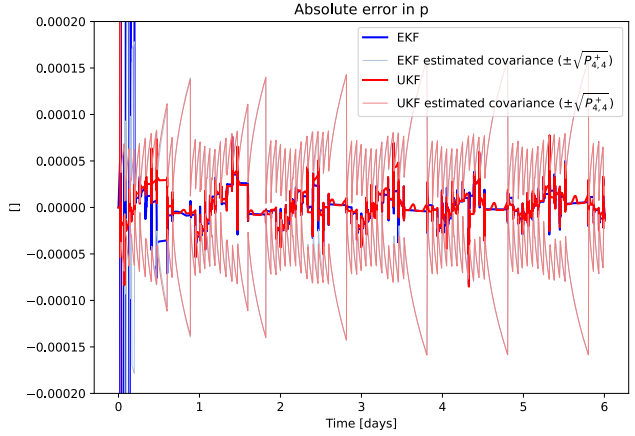
(a) Absolute error in  $\bar{a}$



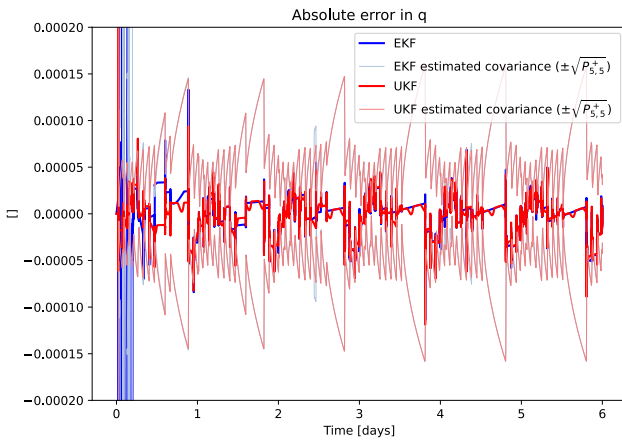
(b) Absolute error in  $\bar{h}$



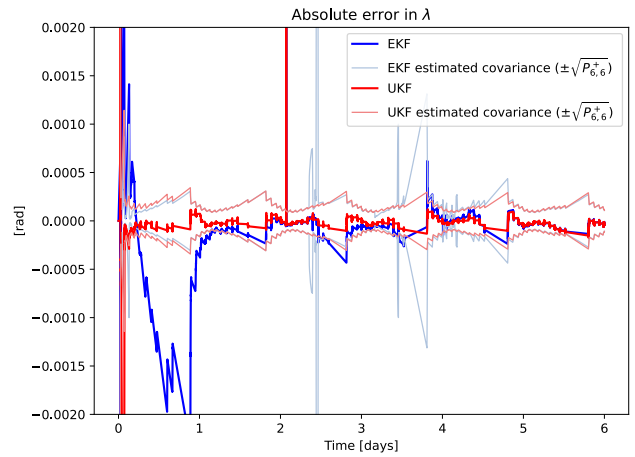
(c) Absolute error in  $\bar{k}$



(d) Absolute error in  $\bar{p}$



(e) Absolute error in  $\bar{q}$



(f) Absolute error in  $\bar{\lambda}$

Figure 4.13: Absolute error (true element minus estimated element) time evolution of the mean equinoctial elements, for a 6-day simulation. The diagonal entries of the covariance matrix associated to each element are also shown. Plots (a), (b), (c), (d), (e) and (f) represent the mean equinoctial elements  $\bar{a}$ ,  $\bar{h}$ ,  $\bar{k}$ ,  $\bar{p}$ ,  $\bar{q}$  and  $\bar{\lambda}$ , respectively. Blue lines depict ESKF error and red lines depict USKF error.

## Chapter 5

# Conclusions

This chapter summarizes the main conclusions that have been progressively presented throughout the text. Afterwards, the main achievements and contributions are outlined, as well as some proposals for possible future research paths linked to Orbit Determination in the context of Semianalytic Satellite Theory (SST).

As outlined in Section 1.2, the main purpose of this thesis was to develop Orbit Determination solutions to estimate LEO satellites in the absence of GNSS sensors, using only ground station data. The majority of the simulations conducted further considered Sun-synchronous orbits, due to their vast real-world applications. The sole use of ground station data is prone to difficulties in orbital filtering problems when compared to GNSS-based approaches. In fact: 1) the filter may spend most of the time propagating orbital states without measurement data, since the update step is only possible when there exists line of sight between the station and the target (station unavailability is especially severe in LEO missions tracked by a low number of stations, or with stations circumscribed to a limited land region), and 2) indirect observation of the dynamical state variables to estimate through nonlinear functions (although some GNSS-based filter solutions consider raw GNSS measurements — pseudorange and phase — it is common to filter directly the position and velocity fixes outputted by GNSS receivers). Furthermore, the present thesis also proposed to explore efficient implementation of filtering algorithms (in view of real-time implementation) through the use of SST.

In Chapter 3 the osculating and mean orbital state dynamics, associated, respectively, to Cowell and Semianalytical orbital propagators, were modeled. The process of averaging the osculating dynamics using the Generalized Method of Averaging of dynamical nonlinear systems was thoroughly described. The SST used in this work was proposed by Todd Ely and diverges from more common theories, namely, the Draper Semianalytic Satellite Theory (DSST), in the sense that: 1) averaging of the osculating element rates is achieved through numerical quadrature, and 2) computation of short-periodic effects, that enable reconstruction of the osculating trajectory, through the Fast Fourier Transform (FFT), as opposed to analytical derivations employed traditionally in other SSTs. When compared to DSST, Ely's theory is more flexible, easier to implement in computational applications and covers a wider range of problem domains, namely, eccentric orbits. However it is not as efficient, since numerical quadrature

and FFT evaluations require sampling of the orbital state function.

Results shown in Section 4.1.1 demonstrate the feasibility of the implemented Cowell and Semianalytical orbital propagators when compared to established propagators from publicly available software. Furthermore, it is shown that the Semianalytical propagator is able to compete with the Cowell propagator, yielding, respectively, position and velocity errors of at most 200 m and  $0.2 \text{ m s}^{-1}$  for a 1-day LEO propagation, whilst retaining a speed improvement factor (Cowell/Semianalytical CPU computation time) of  $\sim 6$ . However, it is noted that no optimization of the computer code was attempted (simulations were made in Python environment).

With the groundwork completed, Orbit Determination filters were then studied - Cowell and Semianalytical orbital propagators were coupled with Extended and Unscented Kalman Filters. Although the EKF usually prevails in most real-world estimation software, the UKF provided improved orbital estimates under all performed simulations. A comparison between the accuracy performance of Cowell and Semianalytical filters was made in Section 4.1.2. It was found that Semianalytical filters outperformed Cowell filters with accuracy gains of  $\sim 300 \text{ m}$  and  $\sim 0.3 \text{ m s}^{-1}$  in position and velocity vectors, respectively, for a 7-day simulation. These results are valid under the simulation environment described in the aforementioned section, especially noting that a simplified force model was used ( $5 \times 5$  gravitational field and atmospheric drag). In contrast, the authors of [12] found in their experiments that Cowell filters outperformed Semianalytical filters. However, the experiments performed in that article made use of more complex and accurate force models. It is hinted that the accuracy gain associated to the use of high-precision force models is overshadowed by the error associated to the averaging procedure underlying Semianalytical dynamics. On the other hand, when using simplified models, Semianalytical filters better capture the long-term evolution of the trajectory. Further research is needed to support these preliminary observations.

Sensitivity of the filters to initialization errors was also studied. This is relevant, since, in the context of sub-optimal Kalman filters, bad estimates may lead to filter divergence. It was found that Cowell filters are able to converge in much worse initialization environments than Semianalytical filters. For instance, Cowell filters converged with initial position errors of up to 10 000 km, whereas both Semianalytical filters started to diverge when position errors of 1000 km were provided. The existence of long periods of station unavailability may explain this drawback, since the estimation of mean trajectories in Semianalytical filters is slightly slower to converge than Cowell estimation of osculating trajectories, due to the inherent averaging procedure associated to the mean dynamics.

To conclude the filter performance examination, it is interesting to compare the RMSE (Figure 4.8) of the filter schemes with direct inversion of the measurement function (Figure 4.9). It is seen that all filters yield better estimates than the ones directly provided by the noisy observations. In fact, the propagation/filtering algorithms should always be able to enhance the accuracy of unprocessed observations. Moreover, the observability analysis that was conducted indicates that range-rate data could be excluded from the observation set with only minor loss in accuracy. Further exclusion of range data led to more significant degradation, however the algorithms still converged.



## 5.1 Contributions

To the best knowledge of the author, the present work proposes the following original contributions:

- Prove and demonstrate the feasibility of Ely's SST in the context of Orbit Determination, departing from DSST-based solutions, which have already been studied before;
- Couple SST with an UKF, leading to the USKF. Compared to the existing ESKF, the proposed USKF provided more accurate orbital state estimates, showed more robustness to destabilization and to initialization errors. It also required less time and fewer observations to converge, as shown in Section 4.2.

Ultimately, the fundamental conclusion to be drawn from this study is that substantial improvements in efficiency can be attained with minor impact on accuracy by the application of SST within Orbit Determination problems.

## 5.2 Future Work

The following suggestions are left for future development of the line of work pursued in this thesis:

- Model resonant tesseral and sectoral harmonics of the gravitational field, which were excluded from the present work, due to the lack of time. Resonant terms should be included in the propagation/filter dynamics when the problem requires the use of complete gravitational fields. For example, in LEO satellites (completing about 14 revolutions per day) the harmonics that resonate, considering a  $50 \times 50$  gravitational field, are  $(14, 14)$  through  $(50, 14)$  and  $(28, 28)$  through  $(50, 28)$  [12];
- The Orbit Determination solutions envisioned in this work are not autonomous, since they are dependent on ground stations. In recent years, autonomous OD solutions have gained popularity through the use of formation-flying constellations [71–73]. It would be interesting to couple Semi-analytical filters within formation-flying problems. The resultant solution would be both autonomous and computationally efficient;
- The present work adopts a greatly simplified measurement model. Realistic measurement models should not exclude effects like the finite speed of light, stellar aberration and media atmospheric corrections [20]. Furthermore, realistic sensors are usually affected by unknown sensor bias, which is typically included in the solve-for filtering state;
- When comparing the developed Semianalytical propagator with DSST, it was found that the *mean-to-osculating map* of DSST outperforms the one implemented in this work. Further research is needed to find the source of this divergence. A good starting point would be to augment Ely's work by building a second-order Semianalytical theory, as is done in DSST for some perturbations;

- Test the filters with real spacecraft data. The present work only considered simulation data, however there exist publicly available sets of observational data for some satellites, for instance, by the Satellite Laser Ranging network<sup>1</sup>;
- Study the impact of explicit discretization of the state dynamics employed by UKF. Recently, the UKF has been extended to systems modeled directly in continuous time [51].

---

<sup>1</sup>SLR observational data is available online, for some satellites, at <https://ilrs.gsfc.nasa.gov/index.html> (last access on 31st December 2020).

# Bibliography

- [1] J. R. Wright. Optimal orbit determination, 2002.
- [2] J. Vetter. Fifty years of orbit determination: Development of modern astrodynamics methods. *J. Hopkins Appl. Tech. D*, 27, January 2007.
- [3] D. Vallado and W. McClain. *Fundamentals of Astrodynamics and Applications*. Space Technology Library. Microcosm Press & Springer, 4<sup>th</sup> edition, 2013. ISBN 978-1881883180.
- [4] O. Montenbruck. Space applications. In *Springer Handbook of Global Navigation Satellite Systems*, pages 933–964. Springer International Publishing, May 2017. ISBN 978-3-319-42926-7. doi: 10.1007/978-3-319-42928-1\_32.
- [5] W. Zhong and P. Gurfil. Mean orbital elements estimation for autonomous satellite guidance and orbit control. *Journal of Guidance Control Dynamics*, 36:1624–1641, November 2013. doi: 10.2514/1.60701.
- [6] S. De Florio and S. D’Amico. Optimal autonomous orbit control of remote sensing spacecraft. February 2009.
- [7] S. D and M. Rothacher. Kinematic and reduced-dynamic precise orbit determination of low earth orbiters. *Advances in Geosciences*, 1, June 2003. doi: 10.5194/adgeo-1-47-2003.
- [8] D. Arnold, O. Montenbruck, S. Hackel, and K. Sośnica. Satellite laser ranging to low earth orbiters: orbit and network validation. *Journal of geodesy*, 93(11):2315–2334, 2019.
- [9] X. Li, J. Wu, K. Zhang, X. Li, Y. Xiong, and Q. Zhang. Real-time kinematic precise orbit determination for leo satellites using zero-differenced ambiguity resolution. *Remote Sensing*, 11(23), 2019.
- [10] N. Zehentner and T. Mayer-Gürr. Precise orbit determination based on raw GPS measurements. *Journal of Geodesy*, 90, November 2015. doi: 10.1007/s00190-015-0872-7.
- [11] H. Peter, A. Jäggi, J. Fernández, D. Escobar, F. Ayuga, D. Arnold, M. Wermuth, S. Hackel, M. Otten, W. Simons, et al. Sentinel-1A—First precise orbit determination results. *Advances in space research*, 60(5):879–892, 2017.
- [12] S. J. Setty, P. J. Cefola, O. Montenbruck, and H. Fiedler. Application of semi-analytical satellite theory orbit propagator to orbit determination for space object catalog maintenance. *Advances in Space Research*, 57(10):2218–2233, 2016.

- [13] M. Morton and T. Roberts. Joint space operations center (JSpOC) mission system (JMS). Technical report, AIR FORCE SPACE COMMAND PETERSON AFB CO, 2011.
- [14] V. F. Boikov, G. Makhonin, A. Testov, Z. N. Khutorovsky, and A. Shogin. Prediction procedures used in satellite catalog maintenance. *Journal of guidance, control, and dynamics*, 32(4):1179–1199, 2009.
- [15] A. J. Green. *Orbit determination and prediction processes for low altitude satellites*. PhD thesis, Massachusetts Institute of Technology, 1979.
- [16] O. Montenbruck. An epoch state filter for use with analytical orbit models of low earth satellites. *Aerospace Science and Technology*, 4(4):277–287, 2000.
- [17] N. O. Tippenhauer, C. Pöpper, K. B. Rasmussen, and S. Capkun. On the requirements for successful GPS spoofing attacks. In *Proceedings of the 18th ACM conference on Computer and communications security*, pages 75–86, 2011.
- [18] H. Chen, G. Chen, E. Blasch, and K. Pham. Comparison of several space target tracking filters. In *Sensors and Systems for Space Applications III*, volume 7330. International Society for Optics and Photonics, 2009.
- [19] A. Mashiku, J. Garrison, and J. R. Carpenter. Statistical orbit determination using the particle filter for incorporating non-gaussian uncertainties. In *AIAA/AAS Astrodynamics Specialist Conference*, 2012.
- [20] O. Montenbruck and E. Gill. *Satellite Orbits*, volume 1. Springer-Verlag GmbH, January 2000. ISBN 978-3-540-67280-7. doi: 10.1007/978-3-642-58351-3.
- [21] B. Schutz, B. Tapley, and G. H. Born. *Statistical orbit determination*. Elsevier, 2004.
- [22] B. O. Teixeira, M. A. Santillo, R. S. Erwin, and D. S. Bernstein. Spacecraft tracking using sampled-data Kalman filters. *IEEE Control Systems Magazine*, 28(4):78–94, 2008.
- [23] S. Spiridonova, M. Kirschner, and U. Hugentobler. Precise mean orbital elements determination for LEO monitoring and maintenance. 2014.
- [24] E. Gomez, P. Servidia, and M. España. Fast and reliable computation of mean orbital elements for autonomous orbit control. November 2019.
- [25] S. Taylor. Semianalytical satellite theory and sequential estimation. Master’s thesis, Massachusetts Institute of Technology, September 1981.
- [26] Z. J. Folcik. Orbit determination using modern filters/smoothers and continuous thrust modeling. Master’s thesis, Massachusetts Institute of Technology, 2008.
- [27] D. Danielson, C. Sagovac, B. Neta, and L. Early. Semianalytic satellite theory (SST): Mathematical algorithms. *Naval Postgraduate School*, January 1995.
- [28] T. A. Ely. Mean element propagations using numerical averaging. *The Journal of the Astronautical Sciences*, 61(3):275–304, 2014.
- [29] T. A. Ely. Transforming mean and osculating elements using numerical methods. *The Journal of*

- the Astronautical Sciences*, 62(1):21–43, 2015.
- [30] N. El-Sheimy and A. Youssef. Inertial sensors technologies for navigation applications: state of the art and future trends. *Satellite Navigation*, 1, January 2020. doi: 10.1186/s43020-019-0001-5.
  - [31] E. Canuto, C. Novara, L. Massotti, C. Perez Montenegro, and D. Carlucci. *Spacecraft dynamics and control: the Embedded Model Control approach*. March 2018. ISBN 978-0-08-100700-6. doi: 10.1016/C2016-0-00420-5.
  - [32] X. Sun, P. Chen, C. Macabiau, and C. Han. Autonomous orbit determination via Kalman filtering of gravity gradients. *IEEE Transactions on Aerospace and Electronic Systems*, 52(5):2436–2451, 2016. doi: 10.1109/TAES.2016.150387.
  - [33] V. Capuano, E. Shehaj, C. Botteron, P. Blunt, and P.-A. Farine. GNSS/INS/Star Tracker integration for real-time on-board autonomous orbit and attitude determination in LEO, MEO, GEO and beyond. September 2017.
  - [34] H. D. Curtis. *Orbital Mechanics for Engineering Students*. Butterworth-Heinemann, 1<sup>st</sup> edition, 2005. ISBN 978-0-7506-6169-0.
  - [35] P. Gil. *Elementos de Mecânica Orbital* [Elements of Orbital Mechanics]. IST, 2015; vs 1.1.6 vs 2020-01-11.
  - [36] O. Montenbruck and A. H. Armstrong. *Practical ephemeris calculations*. Springer-verlag Heidelberg, 1989.
  - [37] M. Capderou. *Handbook of satellite orbits: From kepler to GPS*. Springer Science & Business, 2014.
  - [38] K. R. Muske and T. F. Edgar. *Nonlinear State Estimation*, pages 311—370. Prentice-Hall, Inc., USA, 1997. ISBN 013625179X.
  - [39] Y. Bar-Shalom, X. R. Li, and T. Kirubarajan. *Estimation with applications to tracking and navigation: theory algorithms and software*. John Wiley & Sons, 2004. doi: 10.1002/0471221279.
  - [40] S. C. Chapra, R. P. Canale, et al. *Numerical methods for engineers*. Boston: McGraw-Hill Higher Education,, 2010.
  - [41] P. Frogerais, J.-J. Bellanger, and L. Senhadji. Various ways to compute the continuous-discrete extended Kalman filter. *IEEE Transactions on Automatic Control*, 57(4):1000–1004, 2011.
  - [42] F. Gustafsson and G. Hendeby. On nonlinear transformations of stochastic variables and its application to nonlinear filtering. In *2008 IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 3617–3620. IEEE, 2008.
  - [43] R. Van Der Merwe. *Sigma-point Kalman filters for probabilistic inference in dynamic state-space models*. PhD thesis, OGI School of Science & Engineering at OHSU, 2004.
  - [44] S. J. Julier and J. K. Uhlmann. New extension of the Kalman filter to nonlinear systems. In *Signal processing, sensor fusion, and target recognition VI*, volume 3068, pages 182–193. International Society for Optics and Photonics, 1997.

- [45] P. Virtanen, R. Gommers, T. E. Oliphant, M. Haberland, T. Reddy, D. Cournapeau, E. Burovski, P. Peterson, W. Weckesser, J. Bright, S. J. van der Walt, M. Brett, J. Wilson, K. J. Millman, N. Mayorov, A. R. J. Nelson, E. Jones, R. Kern, E. Larson, C. J. Carey, Í. Polat, Y. Feng, E. W. Moore, J. VanderPlas, D. Laxalde, J. Perktold, R. Cimrman, I. Henriksen, E. A. Quintero, C. R. Harris, A. M. Archibald, A. H. Ribeiro, F. Pedregosa, P. van Mulbregt, and SciPy 1.0 Contributors. SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python. *Nature Methods*, 17:261–272, 2020. doi: 10.1038/s41592-019-0686-2.
- [46] E. A. Wan, R. Van Der Merwe, and S. Haykin. The unscented Kalman filter. *Kalman filtering and neural networks*, 5(2007):221–280, 2001.
- [47] G. Terejanu, T. Singh, and P. Scott. Unscented Kalman filter/smoothing for a CBRN puff-based dispersion model. pages 1 – 8, August 2007. ISBN 978-0-662-45804-3. doi: 10.1109/ICIF.2007.4408076.
- [48] H. J. Kushner. On the differential equations satisfied by conditional probability densities of markov processes, with applications. *Journal of the Society for Industrial and Applied Mathematics, Series A: Control*, 2(1):106–119, 1964.
- [49] R. E. Kalman. A New Approach to Linear Filtering and Prediction Problems. *Journal of Basic Engineering*, 82(1):35–45, March 1960. ISSN 0021-9223. doi: 10.1115/1.3662552.
- [50] P. J. Huxel and R. H. Bishop. Navigation algorithms and observability analysis for formation flying missions. *Journal of guidance, control, and dynamics*, 32(4):1218–1231, 2009.
- [51] S. Sarkka. On unscented Kalman filtering for state estimation of continuous-time nonlinear systems. *IEEE Transactions on automatic control*, 52(9):1631–1641, 2007.
- [52] D. Sundararajan. *The Discrete Fourier Transform*. WORLD SCIENTIFIC, 2001. doi: 10.1142/4610.
- [53] W. Shu and Z. Zheng. Performance analysis of Kalman-based filters and particle filters for non-linear/non-gaussian bayesian tracking. *IFAC Proceedings Volumes*, 38(1):1131–1136, 2005.
- [54] F. R. Hoots and R. L. Roehrich. Spacetrack report no. 3: Models for propagation of NORAD element sets. Technical report, Aerospace Defense Center, Peterson Air Force Base, 1980. Online at (last access on 12th December 2020) <https://www.celestrak.com/NORAD/documentation/spacetrk.pdf>.
- [55] J. Cappellari, C. Vélez, and A. J. Fuchs. *Mathematical theory of the Goddard trajectory determination system*, volume 71106. Goddard Space Flight Center, 1976.
- [56] R. A. Eckman, A. J. Brown, and D. Adamo. Normalization of gravitational acceleration models. Available Online (last access on 12th December 2020) <https://ntrs.nasa.gov/api/citations/20110023121/downloads/20110023121.pdf>, 2011.
- [57] A. H. Nayfeh. *Perturbation methods*. John Wiley & Sons, 2008.
- [58] E. Gill, O. Montenbruck, and H. Kayal. The BIRD satellite mission as a milestone toward GPS-based autonomous navigation. *Navigation*, 48(2):69–75, 2001.
- [59] D. Willner, C. Chang, and K. Dunn. Kalman filter algorithms for a multi-sensor system. In 1976

- IEEE Conference on Decision and Control including the 15th Symposium on Adaptive Processes*, pages 570–574. IEEE, 1976.
- [60] G. Pecoraro, E. Cianca, G. Marino, and M. Ruggieri. Preliminary design of a small tracking RADAR for LEO space objects. In *2017 IEEE Aerospace Conference*, pages 1–11. IEEE, 2017.
  - [61] T. D. Papanikolaou and D. Tsoulis. Assessment of numerical integration methods in the context of low earth orbits and inter-satellite observation analysis. *Acta Geodaetica et Geophysica*, 51(4): 619–641, 2016.
  - [62] J. R. Dormand and P. J. Prince. A family of embedded Runge-Kutta formulae. *Journal of computational and applied mathematics*, 6(1):19–26, 1980.
  - [63] E. Hairer, S. Norsett, and G. Wanner. *Solving Ordinary Differential Equations I: Nonstiff Problems*, volume 8. January 1993. ISBN 978-3-540-56670-0. doi: 10.1007/978-3-540-78862-1.
  - [64] P. Cefola, Z. Folcik, R. Di-Costanzo, N. Bernard, S. Setty, and J. San-Juan. Revisiting the DSST standalone orbit propagator. In *AAS/AIAA Space Flight Mechanics Meeting*, 2014.
  - [65] H. Schaub and J. L. Junkins. *Analytical Mechanics of Space Systems*, volume 1. AIAA, 1<sup>st</sup> edition, 2003.
  - [66] *Orbit Determination Error Analysis for a future Space Debris Tracking Radar*, 2011. European Space Surveillance Conference 2011, 07-09 June 2011, Madrid.
  - [67] L. A. Aguirre, L. L. Portes, and C. Letellier. Structural, dynamical and symbolic observability: From dynamical systems to networks. *PLOS ONE*, 13(10), October 2018. doi: 10.1371/journal.pone.0206180.
  - [68] A. Isidori. *Nonlinear Control Systems*. Communications and Control Engineering. Springer London, 1995. ISBN 9783540199168.
  - [69] Z. Weixing, L. Wanke, and G. Xiaoying. Influence of ground station number and its geographical distribution on combined orbit determination of navigation satellite. *Procedia Environmental Sciences*, 10:2058–2066, 2011.
  - [70] D. J. Fonte. Implementing a  $50 \times 50$  gravity field model in an orbit determination system. Master's thesis, Massachusetts Institute of Technology, 1993.
  - [71] J. Sullivan and S. D'Amico. Nonlinear Kalman filtering for improved angles-only navigation using relative orbital elements. *Journal of Guidance, Control, and Dynamics*, 40(9):2183–2200, 2017.
  - [72] S. D'Amico. Relative orbital elements as integration constants of Hill's equations. *DLR, TN*, pages 05–08, 2005.
  - [73] L. Zhang, T. Li, H. Yang, S. Zhang, H. Cai, and S. Qian. Unscented Kalman filtering for relative spacecraft attitude and position estimation. *The Journal of Navigation*, 68(3):528–548, 2015.

# Appendix A

## Orbit Determination Filters

### A.1 Differential Correction Batch Least Squares

The Least Squares technique (LS) was initially developed for data fitting or estimation problems where the relations between the variables are linear. The problem consists of choosing a mathematical model and fitting the measurements as well as possible, by minimizing the sum of the squares of the residuals, which are defined as the difference between the observed and model-computed values of the dependent variables to estimate [3].

This idea of LS estimation may be extended to nonlinear dynamic problems. All observations are reduced (using the state dynamics) to the initial epoch  $t_0$  and then the estimator finds the best state  $\hat{x}_0^{(LS)}$  in the LS sense. Since the system is nonlinear, the solution is only approximate and thus, the process may be iterated for better convergence. Furthermore, this procedure implies the availability of a whole batch of observations [20].

Algorithm 4 presents the Differential Correction Batch Least Squares procedure. This algorithm is paraphrased and adapted from [21]. Similar procedures were developed and explained in [3, 20]. In this thesis, it is used for initialization of sequential Kalman filters and is intended to be coupled with osculating state dynamics (in cartesian form). Alternative algorithms exist for coupling LS with Semianalytical mean dynamics [15].

The stopping condition, defined in the *while statement* (line 6) is usually chosen to be the root mean square of the observation residuals. For each iteration  $j$  it is given by

$$\text{RMS}^{(j)} = \sqrt{\frac{\sum_{i=1}^T \mathbf{r}_i \mathbf{R}_i^{-1} \mathbf{r}_i^T}{N}}. \quad (\text{A.11})$$

$T$  is the total number of time instants used in the batch and  $N = T \times l$ , where  $l$  is the dimension of the observation vector  $\mathbf{y}$ . Convergence is assumed to be reached when:

$$\left| \frac{\text{RMS}^{(j-1)} - \text{RMS}^{(j)}}{\text{RMS}^{(j-1)}} \right| \leq \epsilon,$$



---

**Algorithm 4** Differential Correction Batch Least Squares

---

**Input:**  $\hat{x}_0^{(0)}$ ;  $\triangleright$  initialization (initial guess) of reference state trajectory  
 $\{(t_i, \mathbf{y}_i, \mathbf{R}_i) : i = 1, \dots, T\}$ .  $\triangleright$  sequence (*batch*) of observations (epoch, vector of observations, weighting matrix)

**Output:**  $\hat{x}_0^{(LS)}$ ;  $\triangleright$  iterated initial state  
 $\hat{P}_0^{(LS)}$ .  $\triangleright$  iterated initial covariance

1: **variables**  $\triangleright$  algorithm variables  
2:  $j = 1$ ;  $\triangleright$  iteration counter  
3:  $IterationMax = 10$ ;  
4:  $\mathbf{N} = \mathbf{0}_{n \times 1}$ ,  $\mathbf{\Lambda} = \mathbf{0}_{n \times n}$ .  $\triangleright$  LS variables  
5: **end variables**  
6: **while** (some stopping criterion is not met) **or** ( $j < IterationMax$ ) **do**  
7: **variables**  $\triangleright$  iteration variables  
8:  $\mathbf{x}^*(t_0) = \hat{x}_0^{(j-1)}$ ,  $\Phi(t_0, t_0) = \mathbf{I}_{n \times n}$ .  $\triangleright$  initialize reference trajectory and state transition matrix  
9: **end variables**  
10: **for**  $t_i \in \{t_1, \dots, t_T\}$  **do**  
11: Read next observation:  $(t_i, \mathbf{y}_i, \mathbf{R}_i)$ .  
12: Numerically integrate the reference trajectory and state transition matrix in  $t \in [t_{i-1}, t_i]$ , from the initial conditions  $\mathbf{x}^*(t_{i-1})$ ,  $\Phi(t_{i-1}, t_0)$ , with:  
$$\dot{\mathbf{x}}^*(t) = \mathbf{f}(\mathbf{x}^*(t), t), \quad (\text{A.1})$$
$$\dot{\Phi}(t, t_0) = \mathbf{F}(\mathbf{x}^*(t), t)\Phi(t, t_0), \quad (\text{A.2})$$

where

$$\mathbf{F}(\mathbf{x}^*(t), t) = \left. \frac{\partial \mathbf{f}(\mathbf{x}(t), t)}{\partial \mathbf{x}(t)} \right|_{\mathbf{x}(t) = \mathbf{x}^*(t)}.$$

The integration produces  $\mathbf{x}^*(t_i)$ ,  $\Phi(t_i, t_0)$ .

13: Accumulate current observation:  
$$\tilde{\mathbf{H}}_i = \left. \frac{\partial \mathbf{h}_d(\mathbf{x}(t_i), t_i)}{\partial \mathbf{x}_i} \right|_{\mathbf{x}(t_i) = \mathbf{x}^*(t_i)}, \quad (\text{A.3})$$
$$\mathbf{r}_i = \mathbf{y}_i - \mathbf{h}_d(\mathbf{x}^*(t_i), t_i), \quad (\text{A.4})$$
$$\mathbf{H}_i = \tilde{\mathbf{H}}_i \Phi(t_i, t_0), \quad (\text{A.5})$$
$$\mathbf{\Lambda} = \mathbf{\Lambda} + \mathbf{H}_i^T \mathbf{R}_i^{-1} \mathbf{H}_i, \quad (\text{A.6})$$
$$\mathbf{N} = \mathbf{N} + \mathbf{H}_i^T \mathbf{R}_i^{-1} \mathbf{r}_i. \quad (\text{A.7})$$

14: **end for**  
15: Solve the LS normal equation for the correction-term  $\Delta \hat{x}_0^{(j)}$   
$$\mathbf{\Lambda} \Delta \hat{x}_0^{(j)} = \mathbf{N}. \quad (\text{A.8})$$

16: Update the iteration reference state and compute the covariance  
$$\hat{x}_0^{(j)} = \hat{x}_0^{(j-1)} + \Delta \hat{x}_0^{(j)}, \quad (\text{A.9})$$
$$\mathbf{P}_0^{(j)} = \mathbf{\Lambda}^{-1}. \quad (\text{A.10})$$

17:  $j = j + 1$ .  $\triangleright$  new iteration  
18: **end while**

---

where  $\epsilon$  was set to 1.

## A.2 Extended Semianalytical Kalman Filter

The operations of the integration and observation grids of the ESKF are presented next, paraphrased from [25]. The filter is initialized with an initial state estimate  $\bar{\mathcal{G}}(t_0)$ , covariance  $P_0$  and  $\Delta\bar{\mathcal{G}}_{0,0}^{0,0} = 0$ .

### Operations on the Integration Grid

1. At the current time  $t_{k,0}$ , update the nominal state,  $\bar{\mathcal{G}}_{N_{\text{new}}}(t_{k,0})$ , for the new integration step, valid on the interval  $t \in [t_{k,0}, t_{k+1,0})$ , from the old nominal state  $\bar{\mathcal{G}}_{N_{\text{old}}}(t_{k,0})$  (defined on the previous integration grid step), using

$$\bar{\mathcal{G}}_{N_{\text{new}}}(t_{k,0}) = \bar{\mathcal{G}}_{N_{\text{old}}}(t_{k,0}) + \Delta\bar{\mathcal{G}}_{k,0}^{k-1,M}, \quad (\text{A.12})$$

where  $\Delta\bar{\mathcal{G}}_{k,0}^{k-1,M}$  is the filter correction from the last observation, at  $t_{k-1,M}$ , propagated until the current time  $t_{k,0}$ . Then set the covariance  $P_{k,0}^-$ . The correction  $\Delta\bar{\mathcal{G}}_{k,0}^{k-1,M}$  and covariance  $P_{k,0}^-$  are found by performing steps 2, 4, 5 and 7 of the observation grid (described below) from  $t_{k-1,M}$  to  $t_{k,0}$ ;

2. Initialize the filter correction and transition matrices:

$$\Delta\bar{\mathcal{G}}_{k,0}^{k,0} = 0, \quad (\text{A.13a})$$

$$\Phi(t_{k,0}, t_{k,0}) = I, \quad (\text{A.13b})$$

$$\Phi_S = \Phi^{-1}(t_{k,0}, t_{k,0}) = I, \quad (\text{A.13c})$$

where  $\Phi$  is the system's STM and  $\Phi_S$  is a local variable that stores the inverse of the STM;

3. Numerically propagate the nominal trajectory and STM dynamics until the next integration grid point at  $t = t_{k+1,0} = t_{k,0} + \Delta t$ , saving intermediate results at  $t = t_{k,\Delta t/2}$  and  $t = t_{k,\Delta 3t/4}$ , with:

$$\begin{aligned} \dot{\bar{\mathcal{G}}}_N(t) &= \bar{f}(\bar{\mathcal{G}}_N(t), t), \\ \dot{\Phi}(t, t_{k,0}) &= \bar{F}(\bar{\mathcal{G}}_N(t), t) \Phi(t, t_{k,0}), \end{aligned} \quad (\text{A.14})$$

from initial conditions  $\bar{\mathcal{G}}_N(t_{k,0})$  and  $\Phi(t_{k,0}, t_{k,0})$ . Functions  $\bar{f}$  and  $\bar{F}$  are defined in (3.33) and (3.46), respectively. Furthermore, at times  $t = t_{k,0}$ ,  $t = t_{k,\Delta t/2}$  and  $t = t_{k+1,0}$  save evaluations of the right-hand side of (A.14), invert the transition matrices and obtain the inverse STM dynamics using [25]

$$\dot{\Phi}^{-1}(t, t_0) = -\Phi^{-1}(t, t_0) \dot{\Phi}(t, t_0) \Phi^{-1}(t, t_0). \quad (\text{A.15})$$

With the values  $\bar{\mathcal{G}}_N(t_{k,0})$ ,  $\bar{\mathcal{G}}_N(t_{k,\Delta t/2})$ ,  $\bar{\mathcal{G}}_N(t_{k+1,0})$ ,  $\Phi(t_{k,0}, t_{k,0})$ ,  $\Phi(t_{k,\Delta t/2}, t_{k,0})$ ,  $\Phi(t_{k+1,0}, t_{k,0})$ ,  $\Phi^{-1}(t_{k,0}, t_{k,0})$ ,  $\Phi^{-1}(t_{k,\Delta t/2}, t_{k,0})$  and  $\Phi^{-1}(t_{k+1,0}, t_{k,0})$  and their respective time rates set up Hermite interpolators for  $\bar{\mathcal{G}}_N(t)$ ,  $\Phi(t, t_{k,0})$  and  $\Phi^{-1}(t, t_{k,0})$  (cf. Section 3.1.2);

4. Compute the short-periodic DFT coefficients  $X_{i_{Z,D}}$  and  $X_{i_T}$  at times  $t = t_{k,0}$ ,  $t = t_{k,\Delta t/2}$ ,  $t =$

$t_{k,3\Delta t/4}$  and  $t = t_{k+1,0}$  and set up Lagrange interpolators for  $\mathbf{X}_{iZ,D}(t)$  and  $\mathbf{X}_{iT}(t)$ <sup>1</sup>(cf. Section 3.1.2).

### Operations on the Observation Grid

The observation grid is operated by the EKF: steps 4 to 7 belong to the filter's predict step and steps 8 and 9 to the update step. It is noted that, if the first observation comes exactly at time  $t = t_{k,0}$ , then the filter predict step is ignored for that observation, since the predicted state and covariance are already known from the integration grid.

1. Obtain a new observation to process  $\mathbf{y}(t_{k,i})$  for  $i = 0, \dots, M$ , where  $\mathbf{y}(t_{k,M})$  is the last observation before the next integration grid;
2. Interpolate  $\bar{\mathcal{G}}_N(t)$  and  $\Phi(t, t_{k,0})$  for  $t = t_{k,i}$  obtaining the nominal state  $\bar{\mathcal{G}}_N(t_{k,i})$  and STM  $\Phi(t_{k,i}, t_{k,0})$  at the observation time. Get  $\Phi^{-1}(t_{k,i-1}, t_{k,0})$  from the value stored in  $\Phi_S$ , resorting to the previous observation;
3. Interpolate the DFT coefficients  $\mathbf{X}_{Z,D}(t)$  and  $\mathbf{X}_T(t)$  for  $t = t_{k,i}$  and use (3.34) to compute the nominal short-periodic function  $\boldsymbol{\eta}(\bar{\mathcal{G}}_N(t_{k,i}))$ ;
4. Compute the transition matrix

$$\Phi(t_{k,i}, t_{k,i-1}) = \Phi(t_{k,i}, t_{k,0}) \Phi^{-1}(t_{k,i-1}, t_{k,0}); \quad (\text{A.16})$$

5. Obtain the predicted filter correction

$$\Delta \bar{\mathcal{G}}_{k,i}^{k,i-1} = \Phi(t_{k,i}, t_{k,i-1}) \Delta \bar{\mathcal{G}}_{k,i-1}^{k,i-1}; \quad (\text{A.17})$$

6. Compute the *a priori* mean state  $\bar{\mathcal{G}}^-(t_{k,i})$  with the nominal state and predicted correction. Then, use it to compute the *a priori* osculating state estimate  $\bar{\mathcal{G}}^-(t_{k,i})$ , by summing the short-periodic function:

$$\bar{\mathcal{G}}^-(t_{k,i}) = \bar{\mathcal{G}}_N(t_{k,i}) + \Delta \bar{\mathcal{G}}_{k,i}^{k,i-1}, \quad (\text{A.18a})$$

$$\bar{\mathcal{G}}^-(t_{k,i}) = \bar{\mathcal{G}}^-(t_{k,i}) + \boldsymbol{\eta}(\bar{\mathcal{G}}_N(t_{k,i})) + \frac{\partial \boldsymbol{\eta}(\bar{\mathcal{G}}_N(t_{k,i}))}{\partial \bar{\mathcal{G}}} \Delta \bar{\mathcal{G}}_{k,i}^{k,i-1}. \quad (\text{A.18b})$$

Notice that the last two terms on the right-hand side of the second equation are a first-order approximation of  $\boldsymbol{\eta}(\bar{\mathcal{G}}^-(t_{k,i}))$  around the nominal state. The matrix  $\frac{\partial \boldsymbol{\eta}}{\partial \bar{\mathcal{G}}}$  is obtained by finite differences, cf. Section 3.3.3, and evaluated at  $\bar{\mathcal{G}}_N(t_{k,i})$ ;

7. Discretize the process noise covariance and compute the predicted covariance:

$$\begin{aligned} \mathbf{Q}_d(t_{k,i}, t_{k,i-1}) &= \Phi(t_{k,i}, t_{k,i-1}) \mathbf{Q}(t_{k,i}) \Phi^T(t_{k,i}, t_{k,i-1}) \cdot (t_{k,i} - t_{k,i-1}), \\ \mathbf{P}_{k,i}^- &= \Phi(t_{k,i}, t_{k,i-1}) \mathbf{P}_{k,i-1}^+ \Phi^T(t_{k,i}, t_{k,i-1}) + \mathbf{Q}_d(t_{k,i}, t_{k,i-1}), \end{aligned} \quad (\text{A.19})$$

<sup>1</sup>The middle points were defined arbitrarily but it is important to have them widely spaced in the interval for improved performance.

where  $Q(t_{k,i})$  is the continuous-time noise covariance matrix;

8. Convert the *a priori* osculating equinoctial state to the cartesian state vector  $\mathbf{x}$  and then compute the predicted observation:

$$\begin{aligned}\mathbf{x}^-(t_{k,i}) &= \mathbf{X}(\mathcal{E}^-(t_{k,i})), \\ \hat{\mathbf{y}}(t_{k,i}) &= \mathbf{h}_d(\mathbf{x}^-(t_{k,i}), t_{k,i});\end{aligned}\tag{A.20}$$

9. Do the EKF update step:

$$\mathbf{K} = \mathbf{P}_{k,i}^- \mathbf{H}^T \left[ \mathbf{H} \mathbf{P}_{k,i}^- \mathbf{H}^T + \mathbf{R}_{k,i} \right]^{-1}, \tag{A.21a}$$

$$\Delta \bar{\mathcal{E}}_{k,i}^{k,i} = \Delta \bar{\mathcal{E}}_{k,i}^{k,i-1} + \mathbf{K} (\mathbf{y}(t_{k,i}) - \hat{\mathbf{y}}(t_{k,i})), \tag{A.21b}$$

$$\bar{\mathcal{E}}^+(t_{k,i}) = \bar{\mathcal{E}}^-(t_{k,i}) + \Delta \bar{\mathcal{E}}_{k,i}^{k,i}, \tag{A.21c}$$

$$\mathbf{P}_{k,i}^+ = (\mathbf{I} - \mathbf{K} \mathbf{H}) \mathbf{P}_{k,i}^-. \tag{A.21d}$$

where  $\mathbf{H}$  is the measurement Jacobian with respect to the equinoctial mean state (3.51), evaluated with  $\bar{\mathcal{E}}^-(t_{k,i})$ , and  $\mathbf{R}_{k,i}$  is the measurement noise covariance matrix;

10. Interpolate  $\Phi^{-1}(t, t_{k,0})$  for  $t = t_{k,i}$  and save the result,  $\Phi^{-1}(t_{k,i}, t_{k,0})$ , into  $\Phi_S$ , to be used in the next observation.

### A.3 Filter Flowcharts

This section presents a flowchart representation of both Cowell and Semianalytical Orbit Determination filters. It is noted that these charts only give a general visualization of the algorithms, being appropriate for computer implementation of simulation studies, running with a fixed simulation step  $\delta t$ . They also assume perfect synchronization between the simulation step and the times of arrival of GS measurements, i.e., with the update-step times. Real-time implementation of these filters in realistic mission scenarios requires a more robust approach. In fact, measurements may not come at fixed known times and computation time of the predict and update procedures also needs to be taken into account. This exhaustive and precise implementation goes beyond the objectives proposed for this thesis. However, as an example, a realistic and meticulous implementation of an on-board filter may be found in [58] for the BIRD satellite mission.

Figure A.1 presents a flowchart of Cowell filtering and Figure A.2 presents a flowchart of Semianalytical filtering. The simulations runs at a fixed step of  $\delta t = 5\text{ s}$  for both filters, which is also the period of arrival of measurements, if available. Furthermore, the Semianalytical filter is tuned with an integration step size  $\Delta t$  of 1 day.

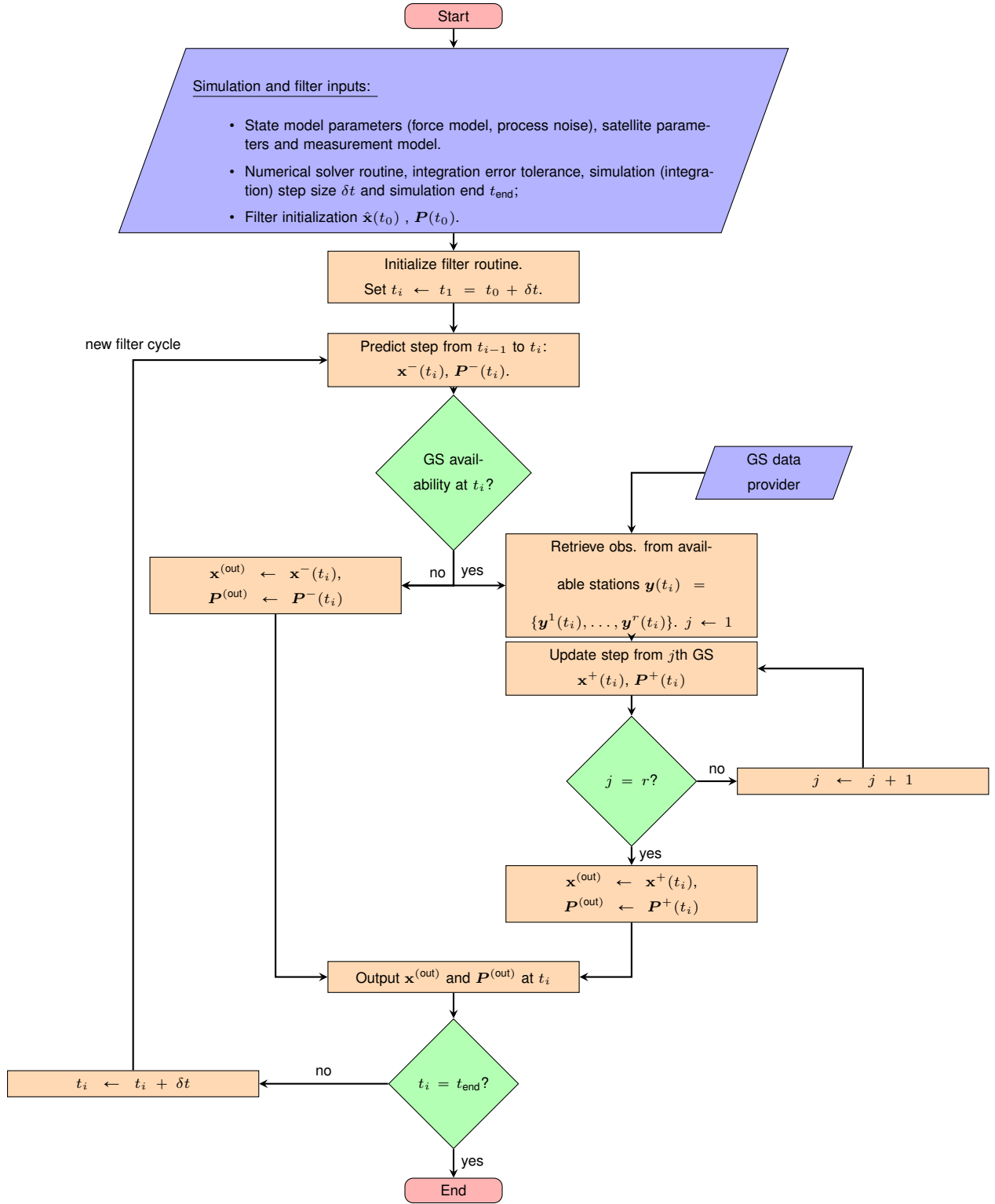


Figure A.1: Flowchart representation of Cowell Orbit Determination filters.

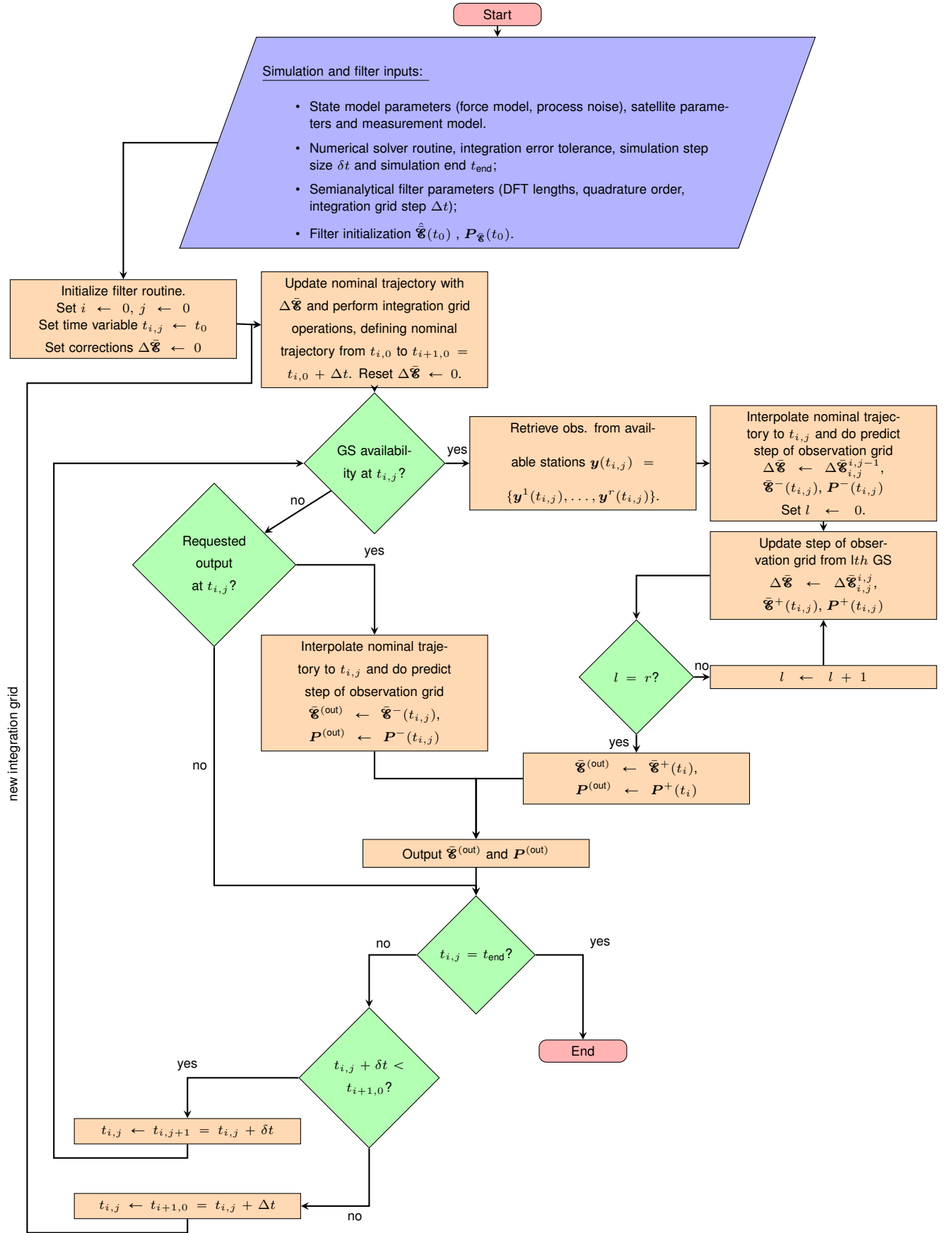


Figure A.2: Flowchart representation of Semianalytical Orbit Determination filters.

## Appendix B

# Linearization of Dynamical Systems and Discretization of Process Noise

Since the KF algorithms are implemented on a predict-update cycling fashion and the state dynamics (Equation (2.15a)) are modeled in continuous-time, a discretization of the process noise covariance matrix in-between update steps is needed, for instance, in Equations (2.33b) and (2.36c) of the EKF and UKF, respectively. Below, the linearization process of the predict step is described, resorting to the first-order Taylor series regression, following similar derivations made in [3, 20]. Moreover, this process is adaptable to the weighted statistical regression for use within the UKF framework.

Consider the state dynamics given by Equation (2.15a). The true value of the state  $\mathbf{x}$  is never known, but can only be estimated. The state estimate  $\hat{\mathbf{x}}(t)$  evolves according to

$$\frac{d\hat{\mathbf{x}}(t)}{dt} = \mathbf{f}(\hat{\mathbf{x}}(t), t). \quad (\text{B.1})$$

The estimation error is then

$$\mathbf{e}(t) = \mathbf{x}(t) - \hat{\mathbf{x}}(t). \quad (\text{B.2})$$

The first-order error dynamics are found by differentiating Equation (B.2) and expanding  $\mathbf{f}$  around  $\hat{\mathbf{x}}(t)$ , that is

$$\begin{aligned} \frac{d\mathbf{e}(t)}{dt} &= \mathbf{f}(\mathbf{x}(t), t) - \mathbf{f}(\hat{\mathbf{x}}(t), t) + \mathbf{w}(t) \\ &\approx \mathbf{F}(\hat{\mathbf{x}}(t), t) \mathbf{e}(t) \end{aligned} \quad (\text{B.3})$$

where

$$\mathbf{F}(\hat{\mathbf{x}}(t), t) = \left. \frac{\partial \mathbf{f}(\mathbf{x}(t), t)}{\partial \mathbf{x}(t)} \right|_{\mathbf{x}(t)=\hat{\mathbf{x}}(t)} \quad (\text{B.4})$$

The formal discrete-time solution of Equation (B.3) from  $t_{k-1}$  to  $t_k$  (filter update epochs) gives

$$\mathbf{e}(t_k) = \Phi(t_k, t_{k-1}) \mathbf{e}(t_{k-1}) + \mathbf{w}_d(t_k, t_{k-1}), \quad (\text{B.5})$$

where  $\Phi(t_k, t_{k-1})$  is obtained by integrating

$$\frac{d\Phi(t, t_{k-1})}{dt} = \mathbf{F}(\hat{\mathbf{x}}(t), t)\Phi(t, t_{k-1}) \quad (\text{B.6})$$

in the predict interval  $t \in [t_{k-1}, t_k]$ , with initial condition  $\Phi(t_{k-1}, t_{k-1}) = \mathbf{I}_{n \times n}$ .  $\Phi$  is the State Transition Matrix (STM). Notice that integration of the STM dynamics requires the companion integration of the state dynamics, as is performed in the EKF algorithm (Equation (2.32)). The sequence  $\mathbf{w}_d(t_k, t_{k-1})$  is the uncertainty in the dynamics used in the propagation of the estimation process and is defined from the white process noise  $\mathbf{w}(t)$  as

$$\mathbf{w}_d(t_k, t_{k-1}) = \int_{t_{k-1}}^{t_k} \Phi(t_k, \tau) \mathbf{w}(\tau) d\tau. \quad (\text{B.7})$$

In the predict step, not only the state estimate  $\hat{\mathbf{x}}_k^-$  is computed, but also the error covariance  $\mathbf{P}_k^-$ , providing a statistical measure of the quality of the estimate. Since  $\mathbf{e}(t_k)$  is assumed to be a zero-mean process<sup>1</sup>,  $\mathbb{E}[\mathbf{e}(t_k)] = 0$ , the covariance at  $t_k$  is computed by propagating the estimation covariance  $\mathbf{P}_{k-1}^+$ , at the previous time-step  $t_{k-1}$ , with

$$\mathbf{P}_k^- = \mathbb{E}[\mathbf{e}(t_k) \mathbf{e}^T(t_k)] = \Phi(t_k, t_{k-1}) \mathbf{P}_{k-1}^+ \Phi^T(t_k, t_{k-1}) + \mathbf{Q}_d(t_k, t_{k-1}), \quad (\text{B.8})$$

where the discretized process noise matrix  $\mathbf{Q}_d(t_k, t_{k-1})$  is given by

$$\mathbf{Q}_d(t_k, t_{k-1}) = \int_{t_{k-1}}^{t_k} \Phi(t_k, \tau) \mathbf{Q}(\tau) \Phi^T(t_k, \tau) d\tau. \quad (\text{B.9})$$

If the gap between measurements is small enough to assume a constant process noise, then the equation above is simplified to:

$$\mathbf{Q}_d(t_k, t_{k-1}) = \Phi(t_k, t_{k-1}) \mathbf{Q} \Phi^T(t_k, t_{k-1}) \cdot (t_k - t_{k-1}), \quad (\text{B.10})$$

which resembles a quadrature approximation of the integral of (B.9).

Regarding the UKF, the process is simpler, since the state dynamics are considered directly in discrete time,  $\mathbf{x}(t_k) = \mathbf{f}_d(\mathbf{x}(t_{k-1}), t_{k-1}) + \mathbf{w}_d(t_k)$ . The statistically linearized dynamics are given by, cf. (2.22),  $\mathbf{x}(t_k) \approx \mathbf{A}(t_k, t_{k-1})\mathbf{x}(t_{k-1}) + \mathbf{b}$ , where the linearization coefficient  $\mathbf{A}$  is given by  $\mathbf{A}(t_k, t_{k-1}) = \mathbf{P}_{k,k-1} (\mathbf{P}_{k-1}^+)^{-1}$ , cf. (2.25). The covariance  $\mathbf{P}_{k,k-1}$  describes the correlation between the states at time  $t_k$  and  $t_{k-1}$  and is obtained with  $[\hat{\mathbf{x}}_k^-, \mathbf{P}_k, \mathbf{P}_{k,k-1}] = \text{UT}(\mathbf{f}_d, \hat{\mathbf{x}}_{k-1}^+, \mathbf{P}_{k-1}^+)$  (predict step of UKF). Equation (B.10) is then rewritten as

$$\mathbf{Q}_d(t_k, t_{k-1}) = \mathbf{A}(t_k, t_{k-1}) \mathbf{Q} \mathbf{A}^T(t_k, t_{k-1}) \cdot (t_k - t_{k-1}), \quad (\text{B.11})$$

where  $\mathbf{A}(t_k, t_{k-1})$  plays the same role as the STM.

<sup>1</sup>It should be noted that, if systematic errors cause  $\mathbb{E}[\mathbf{e}(t_k)] \neq 0$ , then  $\mathbf{P}_k^-$  becomes the mean square error matrix, rather than the covariance.



## Appendix C

# Numerical Implementation of Earth's Gravitational Acceleration

In this appendix, numerical considerations for the computation of the acceleration vector  $\mathbf{a}_{f \text{ Earth grav.}}^i$  of Equation (3.7) are outlined.

### C.1 Recursive Computation of the ALFs

Implementation of the definition of ALFs (Equation (3.5)) is very inefficient for computer applications. Therefore, recursive algorithms may be used instead. A recursive implementation of unnormalized ALFs is presented by Vallado in [3]. It was adapted to the normalized version by following a normalization procedure similar to that provided in [56] (illustrated below). The resulting recursive formulae is (where the argument of  $P_{n,m}(\cdot)$  is dropped to simplify the notation):

$$\bar{P}_{n,0} = \frac{1}{n} \left[ (2n-1) \sin \phi_{gc} \bar{P}_{n-1,0} \frac{N_{n,0}}{N_{n-1,0}} - (n-1) \bar{P}_{n-2,0} \frac{N_{n,0}}{N_{n-2,0}} \right] \quad n \geq 2, \quad (\text{C.1a})$$

$$\bar{P}_{n,m} = \bar{P}_{n-2,m} \frac{N_{n,m}}{N_{n-2,m}} + (2n-1) \cos \phi_{gc} \bar{P}_{n-1,m-1} \frac{N_{n,m}}{N_{n-1,m-1}} \quad m \neq 0, m < n, \quad (\text{C.1b})$$

$$\bar{P}_{n,n} = (2n-1) \cos \phi_{gc} \bar{P}_{n-1,n-1} \frac{N_{n,n}}{N_{n-1,n-1}} \quad n \geq 2, \quad (\text{C.1c})$$

with the starting values

$$\bar{P}_{0,0} = P_{0,0} N_{0,0} = 1 N_{0,0},$$

$$\bar{P}_{1,0} = P_{1,0} N_{1,0} = \sin \phi_{gc} N_{1,0},$$

$$\bar{P}_{1,1} = P_{1,1} N_{1,1} = \cos \phi_{gc} N_{1,1}.$$

It should be noted that in Equation (C.1b),  $P_{n-2,m}$  reduces to zero on the recursions where  $m > n-2$ , because, in view of the definition of the ALFs,  $P_{n,m} = 0$  if  $m > n$ .

Next, the normalization procedure used to adapt the formulae above to the normalized version is outlined, based on [56]. Notice that, in order to obtain the aforementioned recursive formulae, the equations need to be written as a function of normalized polynomials, by replacing the conventional unnormalized ones with their normalized counterparts.

The process is shown for Equation (C.1a) (the same idea applies to the other two). Consider the unnormalized recursive formula, paraphrased from [3]

$$P_{n,0} = \frac{1}{n} [(2n-1) \sin \phi_{gc} P_{n-1,0} - (n-1) P_{n-2,0}] . \quad (\text{C.2})$$

The corresponding normalized Legendre polynomial  $\bar{P}_{n,0}$  is found by multiplying the equation above with the normalization factor  $N_{n,0}$

$$\bar{P}_{n,0} = N_{n,0} P_{n,0} = \frac{1}{n} [(2n-1) \sin \phi_{gc} N_{n,0} P_{n-1,0} - (n-1) N_{n,0} P_{n-2,0}] . \quad (\text{C.3})$$

By substituting the normalization definition  $P_{n,0} = \bar{P}_{n,0}/N_{n,0}$  in the equation above, one gets the normalized recursion

$$\bar{P}_{n,0} = \frac{1}{n} \left[ (2n-1) \sin \phi_{gc} \frac{N_{n,0}}{N_{n-1,0}} \bar{P}_{n-1,0} - (n-1) \frac{N_{n,0}}{N_{n-2,0}} \bar{P}_{n-2,0} \right] . \quad (\text{C.4})$$

According to [56], this normalization procedure is only computationally valid up to  $n + m < 171$  (therefore, valid for the tests conducted in this thesis), since for  $n + m \geq 171$  the computation of the normalization constant becomes prone to overflow errors (for 64-bit double precision real numbers). For these cases, there exist more efficient and computational friendly algorithms to implement the recursion of ALFs, namely, those developed by Samuel Pines, Bill Lear, and Robert Gottlieb.

## C.2 Recursive Computation of the Potential Partialials

Regarding the partials  $\frac{\partial U}{\partial r}$ ,  $\frac{\partial U}{\partial \phi_{gc}}$  and  $\frac{\partial U}{\partial \lambda}$ , they are computed recursively with:

$$\frac{\partial U}{\partial r} = -\frac{\mu}{r^2} - \frac{\mu}{r^2} \sum_{n=2}^{N_n} \sum_{m=0}^{\min(n, N_m)} \left( \frac{a_e}{r} \right)^n (n+1) \bar{P}_{n,m}(\sin \phi_{gc}) [\bar{C}_{n,m} \cos(m\lambda) + \bar{S}_{n,m} \sin(m\lambda)] , \quad (\text{C.5a})$$

$$\begin{aligned} \frac{\partial U}{\partial \phi_{gc}} &= \frac{\mu}{r} \sum_{n=2}^{N_n} \sum_{m=0}^{\min(n, N_m)} \left( \frac{a_e}{r} \right)^n \left[ \bar{P}_{n,m+1} \frac{N_{n,m}}{N_{n,m+1}} (\sin \phi_{gc}) - m \tan(\phi_{gc}) \bar{P}_{n,m}(\sin \phi_{gc}) \right] \\ &\quad \times [\bar{C}_{n,m} \cos(m\lambda) + \bar{S}_{n,m} \sin(m\lambda)] , \end{aligned} \quad (\text{C.5b})$$

$$\frac{\partial U}{\partial \lambda} = \frac{\mu}{r} \sum_{n=2}^{N_n} \sum_{m=0}^{\min(n, N_m)} \left( \frac{a_e}{r} \right)^n m \bar{P}_{n,m}(\sin \phi_{gc}) [\bar{S}_{n,m} \cos(m\lambda) - \bar{C}_{n,m} \sin(m\lambda)] . \quad (\text{C.5c})$$

These expressions were also adapted from the original unnormalized counterparts provided by Vallado in [3], by performing the same normalization procedure presented above.

## Appendix D

# Orbital State-Space Transformations

This appendix presents transformations to interchange between the cartesian state vector  $\mathbf{x} = [(\mathbf{r}_i)^T, (\mathbf{v}_i)^T]^T$  and the equinoctial element set  $\mathfrak{E} = [a, h, k, p, q, \lambda]^T$ . These conversions are widely used throughout this work. Furthermore, some partial derivatives of these transformations are also needed, and therefore are summarized below. All derivatives and transformations provided are paraphrased from [27].

Moreover, a conversion between state vector and Keplerian elements was required in the Keplerian orbit analytical vs. numerical propagation (Section 4.1.1). This conversion is not presented, but is readily available in many orbital mechanics textbooks, namely, on Algorithms 9 and 10 of [3] (*RV2COE* and *COE2RV*).

### D.1 Conversion Between State Vector and Equinoctial Elements

The equinoctial elements are defined with the help of the equinoctial system - basis vectors  $\mathbf{f}$ ,  $\mathbf{g}$  and  $\mathbf{w}$ , which are illustrated in Figure D.1. These basis vectors are computed in terms of  $x$ ,  $y$  and  $z$  components of the ECI frame as:

$$\mathbf{f} = \kappa \begin{bmatrix} 1 - p^2 + q^2 \\ 2pq \\ -2p \end{bmatrix}, \quad \mathbf{g} = \kappa \begin{bmatrix} 2pq \\ 1 + p^2 - q^2 \\ 2q \end{bmatrix}, \quad \mathbf{w} = \kappa \begin{bmatrix} 2p \\ -2q \\ 1 - p^2 - q^2 \end{bmatrix}, \quad (\text{D.1})$$

where

$$\kappa = \frac{1}{1 + p^2 + q^2}.$$

### D.1.1 From State Vector to Equinoctial Elements

The first step in converting from position and velocity to equinoctial elements is to compute the semi-major axis  $a$ , which is obtained by inverting the well-known energy integral for the two-body problem

$$a = \frac{1}{\frac{2}{\|\mathbf{r}_i\|} - \frac{\|\mathbf{v}_i\|^2}{\mu_\oplus}}. \quad (\text{D.2})$$

The second step is to compute the basis vectors ( $\mathbf{f}$ ,  $\mathbf{g}$ ,  $\mathbf{w}$ ) of the equinoctial system. The  $\mathbf{w}$  vector is obtained by normalizing the angular momentum vector, that is

$$\mathbf{w} = \frac{\mathbf{r}_i \times \mathbf{v}_i}{\|\mathbf{r}_i \times \mathbf{v}_i\|}. \quad (\text{D.3})$$

Equinoctial elements  $p$  and  $q$  are then given by:

$$p = \frac{w_x}{1 + w_z}, \quad (\text{D.4a})$$

$$q = -\frac{w_y}{1 + w_z}. \quad (\text{D.4b})$$

Vectors  $\mathbf{f}$  and  $\mathbf{g}$  are then computed using Equation (D.1).

The third step is to compute the eccentricity-related quantities. The eccentricity vector  $\mathbf{e}$  is given by

$$\mathbf{e} = -\frac{\mathbf{r}_i}{\|\mathbf{r}_i\|} + \frac{\mathbf{v}_i \times (\mathbf{r}_i \times \mathbf{v}_i)}{\mu_\oplus}. \quad (\text{D.5})$$

Equinoctial elements  $h$  and  $k$  are then given by:

$$h = \mathbf{e} \cdot \mathbf{g}, \quad (\text{D.6a})$$

$$k = \mathbf{e} \cdot \mathbf{f}. \quad (\text{D.6b})$$

The last step is to compute the mean longitude  $\lambda$ . First, compute the position coordinates of the satellite in the equinoctial system:

$$X = \mathbf{r}_i \cdot \mathbf{f}, \quad (\text{D.7a})$$

$$Y = \mathbf{r}_i \cdot \mathbf{g}. \quad (\text{D.7b})$$

Then, compute the eccentric longitude  $F$  with:

$$\sin F = h + \frac{(1 - h^2 b)Y - hkbX}{a\sqrt{1 - h^2 - k^2}}, \quad (\text{D.8a})$$

$$\cos F = k + \frac{(1 - k^2 b)X - hkbY}{a\sqrt{1 - h^2 - k^2}}, \quad (\text{D.8b})$$

where

$$b = \frac{1}{1 + \sqrt{1 - h^2 - k^2}}. \quad (\text{D.9})$$

Finally,  $\lambda$  is given by the equinoctial form of the Kepler's equation, where  $F$  is previously computed as  $F = \arctan 2(\sin F, \cos F)$ , that is

$$\lambda = F + h \cos F - k \sin F. \quad (\text{D.10})$$

### D.1.2 From Equinoctial Elements to State Vector

The first step in converting from equinoctial elements to position and velocity is to determine the equinoctial system basis vectors ( $\mathbf{f}$ ,  $\mathbf{g}$ ,  $\mathbf{w}$ ) using Equation (D.1).

The second step is to find the eccentric and true longitudes  $F$  and  $L$ , respectively.  $F$  is obtained by numerically solving Kepler's (D.10), using, for instance, the Newton's method. Then, compute  $b$  and mean motion  $n$  using, respectively, (D.9) and

$$n = \sqrt{\frac{\mu_{\oplus}}{a^3}}. \quad (\text{D.11})$$

The true longitude  $L$  is then computed using:

$$\sin L = \frac{(1 - k^2 b) \sin F + h k b \cos F - h}{1 - h \sin F - k \cos F}, \quad (\text{D.12a})$$

$$\cos L = \frac{(1 - h^2 b) \cos F + h k b \sin F - k}{1 - h \sin F - k \cos F}. \quad (\text{D.12b})$$

The third step is to compute the position and velocity components ( $X$ ,  $Y$ ) and ( $\dot{X}$ ,  $\dot{Y}$ ) of the satellite in the equinoctial system. The radial distance is given by

$$r = \frac{a(1 - h^2 - k^2)}{1 + h \sin L + k \cos L}. \quad (\text{D.13})$$

The position components are then given by:

$$X = r \cos L, \quad (\text{D.14a})$$

$$Y = r \sin L. \quad (\text{D.14b})$$

The velocity components are computed as:

$$\dot{X} = -\frac{na(h + \sin L)}{\sqrt{1 - h^2 - k^2}}, \quad (\text{D.15a})$$

$$\dot{Y} = \frac{na(k + \cos L)}{\sqrt{1 - h^2 - k^2}}. \quad (\text{D.15b})$$

The final step is to compute the position and velocity vectors using:

$$\mathbf{r}_i = X \mathbf{f} + Y \mathbf{g}, \quad (\text{D.16a})$$

$$\mathbf{v}_i = \dot{X} \mathbf{f} + \dot{Y} \mathbf{g}. \quad (\text{D.16b})$$

## D.2 Partial Derivatives of State Vector With Respect to the Equinoctial Elements

Let

$$A = \sqrt{\mu \oplus a}, \quad (D.17a)$$

$$B = \sqrt{1 - h^2 - k^2}, \quad (D.17b)$$

$$C = 1 + p^2 + q^2, \quad (D.17c)$$

and compute  $(X, Y)$  and  $(\dot{X}, \dot{Y})$  by algebraically transforming Equations (D.16) to:

$$\begin{aligned} X &= \mathbf{r}_i \cdot \mathbf{f}, & Y &= \mathbf{r}_i \cdot \mathbf{g}, \\ \dot{X} &= \mathbf{v}_i \cdot \mathbf{f}, & \dot{Y} &= \mathbf{v}_i \cdot \mathbf{g}. \end{aligned} \quad (D.18)$$

The partial derivatives of the position vector  $\mathbf{r}_i$  with respect to the equinoctial elements are given by:

$$\begin{aligned} \frac{\partial \mathbf{r}_i}{\partial a} &= \frac{\mathbf{r}_i}{a}, & \frac{\partial \mathbf{r}_i}{\partial h} &= \frac{\partial X}{\partial h} \mathbf{f} + \frac{\partial Y}{\partial h} \mathbf{g}, \\ \frac{\partial \mathbf{r}_i}{\partial k} &= \frac{\partial X}{\partial k} \mathbf{f} + \frac{\partial Y}{\partial k} \mathbf{g}, & \frac{\partial \mathbf{r}_i}{\partial p} &= \frac{2[q(Y\mathbf{f} - X\mathbf{g}) - X\mathbf{w}]}{C}, \\ \frac{\partial \mathbf{r}_i}{\partial q} &= \frac{2[p(X\mathbf{g} - Y\mathbf{f}) + Y\mathbf{w}]}{C}, & \frac{\partial \mathbf{r}_i}{\partial \lambda} &= \frac{\mathbf{v}_i}{n}, \end{aligned} \quad (D.19)$$

where  $n$  is computed with Equation (D.11) and where:

$$\begin{aligned} \frac{\partial X}{\partial h} &= -\frac{k\dot{X}}{n(1+B)} + \frac{aY\dot{Y}}{AB}, & \frac{\partial Y}{\partial h} &= -\frac{k\dot{Y}}{n(1+B)} - \frac{aX\dot{Y}}{AB} - a, \\ \frac{\partial X}{\partial k} &= \frac{h\dot{X}}{n(1+B)} + \frac{aY\dot{X}}{AB} - a, & \frac{\partial Y}{\partial k} &= \frac{h\dot{Y}}{n(1+B)} - \frac{aX\dot{X}}{AB}. \end{aligned} \quad (D.20)$$

The partial derivatives of the velocity vector  $\mathbf{v}_i$  with respect to the equinoctial elements are given by:

$$\begin{aligned} \frac{\partial \mathbf{v}_i}{\partial a} &= -\frac{\mathbf{v}_i}{2a}, & \frac{\partial \mathbf{v}_i}{\partial h} &= \frac{\partial \dot{X}}{\partial h} \mathbf{f} + \frac{\partial \dot{Y}}{\partial h} \mathbf{g}, \\ \frac{\partial \mathbf{v}_i}{\partial k} &= \frac{\partial \dot{X}}{\partial k} \mathbf{f} + \frac{\partial \dot{Y}}{\partial k} \mathbf{g}, & \frac{\partial \mathbf{v}_i}{\partial p} &= \frac{2[q(\dot{Y}\mathbf{f} - \dot{X}\mathbf{g}) - \dot{X}\mathbf{w}]}{C}, \\ \frac{\partial \mathbf{v}_i}{\partial q} &= \frac{2[p(\dot{X}\mathbf{g} - \dot{Y}\mathbf{f}) + \dot{Y}\mathbf{w}]}{C}, & \frac{\partial \mathbf{v}_i}{\partial \lambda} &= -\frac{na^3\mathbf{r}_i}{r^3}, \end{aligned} \quad (D.21)$$

where  $r = \|\mathbf{r}_i\|$  and where:

$$\begin{aligned} \frac{\partial \dot{X}}{\partial h} &= \frac{a\dot{Y}^2}{AB} + \frac{A}{r^3} \left( \frac{akX}{1+B} - \frac{Y^2}{B} \right), & \frac{\partial \dot{Y}}{\partial h} &= -\frac{a\dot{X}\dot{Y}}{AB} + \frac{A}{r^3} \left( \frac{akY}{1+B} + \frac{XY}{B} \right), \\ \frac{\partial \dot{X}}{\partial k} &= \frac{a\dot{X}\dot{Y}}{AB} - \frac{A}{r^3} \left( \frac{ahX}{1+B} + \frac{XY}{B} \right), & \frac{\partial \dot{Y}}{\partial k} &= -\frac{a\dot{X}^2}{AB} - \frac{A}{r^3} \left( \frac{ahY}{1+B} - \frac{X^2}{B} \right). \end{aligned} \quad (D.22)$$

These derivatives may then be concatenated in matrix form:

$$\frac{\partial \mathbf{x}}{\partial \mathbf{g}} = \begin{bmatrix} \frac{\partial \mathbf{r}_i}{\partial a} & \frac{\partial \mathbf{r}_i}{\partial h} & \frac{\partial \mathbf{r}_i}{\partial k} & \frac{\partial \mathbf{r}_i}{\partial p} & \frac{\partial \mathbf{r}_i}{\partial q} & \frac{\partial \mathbf{r}_i}{\partial \lambda} \\ \frac{\partial \mathbf{v}_i}{\partial a} & \frac{\partial \mathbf{v}_i}{\partial h} & \frac{\partial \mathbf{v}_i}{\partial k} & \frac{\partial \mathbf{v}_i}{\partial p} & \frac{\partial \mathbf{v}_i}{\partial q} & \frac{\partial \mathbf{v}_i}{\partial \lambda} \end{bmatrix}. \quad (\text{D.23})$$

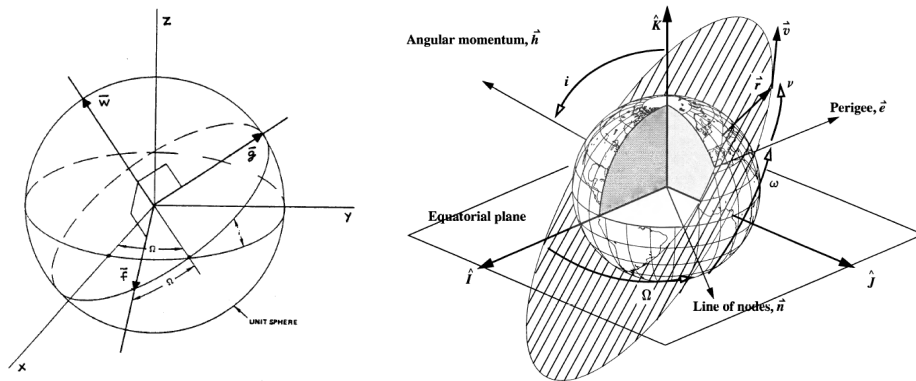
### D.3 Partial Derivatives of Equinoctial Elements With Respect to Velocity

Regarding the partial derivatives of the equinoctial elements with respect to the state vector, only the ones with respect to velocity are needed in this work (in the computation of the Gaussian VOP). Let  $\frac{\partial \mathcal{E}_j}{\partial \mathbf{v}_i}$  denote the vector, in cartesian coordinates, whose components in the ECI frame are the partial derivatives of the equinoctial element  $\mathcal{E}_j$  with respect to  $(v_x, v_y, v_z)$ , i.e.,

$$\frac{\partial \mathcal{E}_j}{\partial \mathbf{v}_i} = \begin{bmatrix} \frac{\partial \mathcal{E}_j}{\partial v_x} & \frac{\partial \mathcal{E}_j}{\partial v_y} & \frac{\partial \mathcal{E}_j}{\partial v_z} \end{bmatrix} \quad (\text{D.24})$$

The partial derivatives of the equinoctial elements with respect to velocity are given by:

$$\begin{aligned}
\frac{\partial a}{\partial \mathbf{v}_i} &= \frac{2\mathbf{v}_i}{n^2 a}, & \frac{\partial h}{\partial \mathbf{v}_i} &= \frac{(2\dot{X}Y - X\dot{Y})\mathbf{f} - X\dot{X}\mathbf{g}}{\mu_\oplus} + \frac{k(qY - pX)\mathbf{w}}{AB}, \\
\frac{\partial k}{\partial \mathbf{v}_i} &= \frac{(2X\dot{Y} - \dot{X}Y)\mathbf{g} - Y\dot{Y}\mathbf{f}}{\mu_\oplus} - \frac{h(qY - pX)\mathbf{w}}{AB}, & \frac{\partial p}{\partial \mathbf{v}_i} &= \frac{CY\mathbf{w}}{2AB}, \\
\frac{\partial q}{\partial \mathbf{v}_i} &= \frac{CX\mathbf{w}}{2AB}, & \frac{\partial \lambda}{\partial \mathbf{v}_i} &= -\frac{2\mathbf{r}_i}{A} + \frac{k\frac{\partial h}{\partial \mathbf{v}_i} - h\frac{\partial k}{\partial \mathbf{v}_i}}{1+B} + \frac{(qY - pX)\mathbf{w}}{A}.
\end{aligned} \tag{D.25}$$



(a) The equinoctial element set is associated with vectors  $\mathbf{f}$ ,  $\mathbf{g}$  and  $\mathbf{w}$ , which are the basis vectors of the equinoctial system.  $\mathbf{f}$  and  $\mathbf{g}$  lie in the orbital plane and  $\mathbf{w}$  is parallel to the angular momentum vector  $\mathbf{h}$ .  $h$  and  $k$  are, respectively, the  $\mathbf{f}$  and  $\mathbf{g}$  components of the eccentricity vector  $\mathbf{e}$ ,  $p$  and  $q$  are, respectively, the  $\mathbf{f}$  and  $\mathbf{g}$  components of the line of nodes vector  $\mathbf{n}$ . Source [27].

(b) Visualization of line of nodes vector  $\mathbf{n}$ , eccentricity vector  $\mathbf{e}$  and angular momentum vector  $\mathbf{h}$ , along with some Keplerian orbital elements, namely  $i$ ,  $\varpi$ ,  $\Omega$  and  $\nu$ . Source [3].

Figure D.1: Representation of the equinoctial system in (a) and representation of auxiliary vectors in (b).





# Appendix E

## Equations of Averaging

In this appendix, the first-order Equations of Averaging (3.15) of SST are derived, following [27]. Expanding  $F_i$  about the mean elements and  $n(a)$  about the mean semi-major axis, using Taylor series expansions, and rearranging the result as a power series of  $\epsilon$  yields:

$$F_i(\alpha, \lambda, \theta) = F_i(\bar{\alpha}, \bar{\lambda}, \theta) + \sum_{j=1}^{\infty} \epsilon^j f_i^j(\bar{\alpha}, \bar{\lambda}, \theta), \quad i = 1, \dots, 6, \quad (\text{E.1a})$$

$$n(a) = n(\bar{a}) + \sum_{j=1}^{\infty} \epsilon^j N^j(\bar{a}), \quad (\text{E.1b})$$

where the first-order terms are given by  $f_i^1 = \sum_{j=1}^6 \frac{\partial F_i}{\partial a_j} \eta_j^1$  and  $N^1 = -\frac{3}{2} \frac{\eta_1^1}{\bar{a}} n(\bar{a})$ . Thus, substituting (E.1) into (3.12), the osculating VOP may be expressed as a function of mean elements as

$$\frac{d\mathcal{E}_i}{dt} = \delta_{i6} \left( n(\bar{a}) + \sum_{j=1}^{\infty} \epsilon^j N^j(\bar{a}) \right) + \epsilon^1 F_i(\bar{\alpha}, \bar{\lambda}, \theta) + \sum_{j=1}^{\infty} \epsilon^{j+1} f_i^j(\bar{\alpha}, \bar{\lambda}, \theta), \quad i = 1, \dots, 6. \quad (\text{E.2})$$

Next, differentiating (3.14) with respect to  $t$  leads to another expression for the osculating VOP

$$\frac{d\mathcal{E}_i}{dt} = \frac{d\bar{\mathcal{E}}_i}{dt} + \sum_{j=1}^{\infty} \epsilon^j \left( \sum_{k=1}^6 \frac{\partial \eta_i^j}{\partial \mathcal{E}_k} \frac{d\bar{\mathcal{E}}_k}{dt} + \frac{\partial \eta_i^j}{\partial \theta} \frac{d\theta}{dt} \right), \quad i = 1, \dots, 6. \quad (\text{E.3})$$

Substituting (3.13) into (E.3) and rearranging terms yields

$$\frac{d\mathcal{E}_i}{dt} = n(\bar{a}) \delta_{i6} + \sum_{j=1}^{\infty} \epsilon^j \left( A_i^j + n(\bar{a}) \frac{\partial \eta_i^j}{\partial \lambda} + \sum_{k=1}^6 \frac{\partial \eta_i^j}{\partial \mathcal{E}_k} \sum_{l=1}^{\infty} \epsilon^l A_k^l + \frac{\partial \eta_i^j}{\partial \theta} \omega_{\oplus} \right), \quad i = 1, \dots, 6. \quad (\text{E.4})$$

Equating the right-hand sides of (E.4) and (E.2) yields an expression where each term is a power of  $\epsilon$ . Requiring the terms of the same power to be equal yields, for  $j = 1, 2, \dots$ , the Equations of Averaging of order 1, 2,  $\dots$ , respectively. The first-order dynamics are modeled by

$$\mathcal{O}(\epsilon^1): \quad A_i^1(\bar{\alpha}) + \frac{\partial \eta_i^1}{\partial \lambda} n(\bar{a}) + \frac{\partial \eta_i^1}{\partial \theta} \omega_{\oplus} = F_i(\bar{\alpha}, \bar{\lambda}, \theta) + N^1 \delta_{i6}. \quad (\text{E.5})$$