

Computational Methods for the Long-Term Propagation of Space Debris Orbits

Robin Hofsteenge

Master of Science Thesis

Computational Methods for the Long-Term Propagation of Space Debris Orbits

MASTER OF SCIENCE THESIS

For obtaining the degree of Master of Science in Aerospace Engineering at
Delft University of Technology

Robin Hofsteenge

December 1, 2013

Faculty of Aerospace Engineering · Delft University of Technology

Robin Hofsteenge, BSc

Chair of Astrodynamics and Space Missions
Faculty of Aerospace Engineering
Delft University of Technology
Kluyverweg 1, 2629 HS, Delft
The Netherlands

E-mail: R.Hofsteenge@student.tudelft.nl

Cover figure: Part of a destroyed solar panel in orbit around the Earth.

Final Version
Document Version 1.1
December 1, 2013

This document was typeset using L^AT_EX 2 ϵ .

Preface

The thesis project constitutes the final part of the Master of Science (MSc) program in Aerospace Engineering at Delft University of Technology. Specifically, this thesis project is about computational methods for the long-term propagation of space debris orbits. Prior to this project, a literature study was carried out. The present document continues on the findings of the literature study, but can be read as a stand-alone document. The report focuses on methods and concepts that have actually been implemented in the simulation code developed for the project. For a more complete treatment of the different perturbing forces acting on satellites or a more comprehensive overview of integration methods, the interested reader is encouraged to consult the report of the literature study (Hofsteenge, 2012).

I would like to take this opportunity to express my gratitude toward my thesis supervisor, Ron Noomen, for his helpful guidance during the project. Our weekly meetings not only made sure I remained on the right track, but were also useful in keeping me inspired, despite any difficulties encountered. I would also like to thank my fellow MSc students on the ninth floor of the Faculty of Aerospace Engineering for the pleasant working atmosphere. Finally, I would like to thank my parents for supporting me throughout my academic career.

Robin Hofsteenge

Delft, December 1, 2013

Abstract

Space debris poses a significant problem for the space sector. This problem relates to potential collisions of debris objects with active satellites, which in many cases will lead to catastrophic damage. Due to the absence of natural decay mechanisms in the higher regions of space, debris objects in these regions have very long orbital lifetimes. In order to assess the hazards posed to active satellites, it is relevant to be able to predict how the orbits of these debris objects behave on long timescales.

A simulation code in C++ has been created for this thesis project, capable of efficient propagation of space debris trajectories over long periods of time (typically a century or more), while taking into account various relevant perturbing forces. The simulation code can be applied to simulate the orbits of debris objects with a wide range of area-to-mass ratios, from intact satellites to tiny flecks of paint. The results produced with the simulation code have been verified to be consistent with results presented in recent research papers on space debris.

An extensive performance comparison has been made regarding the efficiency of different computational methods for carrying out accurate, long-term integrations of space debris orbits. Both traditional integration methods and symplectic integration methods were tested, the latter of which are interesting because of their energy conservation properties. All methods were also combined with different formulations of the equations of motion.

Of the methods tested, the Dormand-Prince 8(7) integration method combined with Gauss' form of Lagrange's planetary equations in modified equinoctial elements was found to be the most efficient. The performance of the symplectic integration methods was markedly less for this application than for the integration of completely Hamiltonian systems, though it was certainly acceptable.

The simulation code was also applied to predict the long-term orbital evolution for debris objects in GEO and GNSS graveyard orbits. While proposed GEO graveyard orbits were found to be safe, graveyard orbits in the GNSS region were found to be susceptible to resonances induced by the luni-solar perturbations, and hence, require a careful selection of the initial orbital parameters. In all cases, debris objects with high area-to-mass ratios were determined to be dangerous to active satellites, regardless of the initial conditions of the graveyard orbit.

Contents

Preface	5
Abstract	7
List of Symbols	13
List of Abbreviations	17
1 Introduction	19
2 Space Debris	21
2.1 History	21
2.2 Dangers Related to Space Debris	23
2.3 The Space Debris Environment	23
2.4 Sources of Space Debris	24
2.5 Characteristics of Debris Objects	26
2.6 Guidelines Regarding Space Debris	27
3 The GEO and GNSS Debris Environments	29
3.1 The GEO Debris Environment	29
3.1.1 Debris Population in the GEO Region	29
3.1.2 Existing Research	32
3.2 The GNSS Debris Environment	33
3.2.1 Debris Population in the GNSS Region	33
3.2.2 Existing Research	35
4 Force Model	37
4.1 Overview	37
4.2 Central Gravitational Force	40
4.3 Deviations in the Earth's Gravity Field	40
4.4 Gravitational Attraction of the Sun and the Moon	45
4.5 Solar Radiation Pressure	46

5	Orbit Formulation	53
5.1	Basic Laws of Astrodynamics	53
5.2	Coordinate Systems	54
5.2.1	Cartesian Coordinates	54
5.2.2	Orbital Elements	55
5.2.3	Modified Equinoctial Elements	57
5.3	Modeling Perturbed Orbits	58
5.3.1	Method of Cowell	59
5.3.2	Method of Encke	59
5.3.3	Method of Variation of Parameters	60
5.4	Choice of Propagation Methods	64
6	Traditional Integration Methods	67
6.1	Introduction	68
6.2	Accuracy and Efficiency	75
6.3	Fixed Step Size Methods	76
6.3.1	Runge-Kutta 4	76
6.3.2	General Runge-Kutta Methods	77
6.4	Adaptive Step Size Methods	78
6.4.1	General Aspects	78
6.4.2	Runge-Kutta-Fehlberg 5(6)	80
6.4.3	Dormand-Prince 8(7)	80
6.5	Choice of Integration Methods	82
7	Symplectic Integration Methods	85
7.1	Exponential Operators	85
7.2	Hamiltonian Mechanics	87
7.3	Symplectic Mappings	88
7.4	Symplectic Integrators	89
7.4.1	General Formulation	89
7.4.2	Hamiltonian Splitting	89
7.4.3	Kinoshita's Method	90
7.4.4	SABA _{2n} Methods	91
7.5	Methods for Splitting the Hamiltonian	93
7.5.1	T + V Splitting	93
7.5.2	Wisdom-Holman Splitting	94
8	Verification of the Simulation Code	95
8.1	Overview of the Simulation Code	95
8.2	Two-Body Problem	97
8.3	Positions of the Sun and the Moon	98
8.4	Individual Perturbations	100
8.4.1	J_2 -Term of the Earth's Gravity Field	101
8.4.2	Gravitational Attraction of the Sun	104

8.4.3	Gravitational Attraction of the Moon	105
8.4.4	Solar Radiation Pressure	106
8.5	Long-Term Behavior	110
9	Performance Comparison of Computational Methods	119
9.1	Overview	120
9.2	Testing Methodology	121
9.3	Two-Body Problem	123
9.3.1	Round-Off Errors	123
9.3.2	Error Growth	126
9.3.3	Performance for a Circular Orbit	129
9.3.4	Performance for a Highly Eccentric Orbit	132
9.4	Perturbed Problem	134
9.4.1	Important Remarks	134
9.4.2	Choice of Reference	138
9.4.3	Case I: Near-Circular GEO Graveyard Orbit – Low A/m	140
9.4.4	Case II: Near-Circular GEO Graveyard Orbit – High A/m	145
9.4.5	Case III: Highly Eccentric GPS Graveyard Orbit – Low A/m	150
9.4.6	Case IV: Near-Circular GPS Graveyard Orbit – High A/m	155
9.5	Discussion	160
10	Long-Term Debris Orbit Predictions	165
10.1	Introduction	165
10.2	GEO Graveyard Orbits	166
10.3	GNSS Graveyard Orbits	170
10.3.1	Low Altitude Eccentric Orbits	170
10.3.2	High Altitude Eccentric Orbits	174
10.3.3	Near-Circular Orbits	178
10.4	Discussion	182
11	Conclusions and Recommendations	185
11.1	Conclusions	185
11.2	Recommendations	188
A	JGM-3 Gravitational Coefficients	189
B	Routines for the Positions of the Sun and the Moon	191
C	Important Equations in Astrodynamics	195
	Bibliography	197

List of Symbols

A	Area [m ²]
\mathbf{a}	Acceleration vector of orbiting object [m/s ²]
a	Semi-major axis [m]
C_R	Radiation pressure coefficient [-]
$C_{n,m}$	Gravity field coefficient [-]
c	Speed of light, $c = 2.998792458 \cdot 10^8$ [m/s]
D	Difference between the mean longitudes of the Sun and the Moon [°]
d	Degrees of freedom of system
E	Eccentric anomaly [°]
E_ν	Energy of photon [J]
\mathbf{e}_r	Unit position vector of the object [-]
\mathbf{e}_\odot	Unit vector pointing from the object to the Sun [-]
\mathbf{e}_{Sun}	Unit vector pointing to the Sun [-]
e	Eccentricity [-]
ϵ	Local truncation error
F	Mean angular distance of the Moon from the ascending node [°]
f	Force function
\mathbf{f}	Collection of all perturbing accelerations that cannot be expressed as a potential [m/s ²]
\mathbf{f}_t	Total acceleration [m/s ²]
\mathbf{f}_N	Transverse component of perturbing acceleration [m/s ²]
\mathbf{f}_S	Radial component of perturbing acceleration [m/s ²]
\mathbf{f}_W	Normal component of perturbing acceleration [m/s ²]
G	Universal gravitational constant, $G = 6.67384 \cdot 10^{-11}$ [m ² /kg s ²]
\mathbf{g}'	Jacobian matrix
H	Angular momentum [m ² /s]
H_a	Altitude of apogee [m]
H_p	Altitude of perigee [m]
ΔH_{geo}	Difference in altitude with the geostationary altitude [m]

\mathcal{H}	Hamiltonian [m^2/s^2]
h	Integration step size [s]
i	Inclination [$^\circ$]
J	Symplectic identity [-]
$J_n (= -C_{n0})$	Zonal gravity field coefficient [-]
K_i	Slope function
\mathbf{k}_i	Slope function (vector)
L	Lagrangian [m^2/s^2]
L_0	Mean longitude of the Moon [$^\circ$]
l	Mean anomaly of the Moon [$^\circ$]
l'	Mean anomaly of the Sun [$^\circ$]
M	Mass of celestial body [kg]
M	Mean anomaly [$^\circ$]
M_0	Initial mean anomaly [$^\circ$]
M_\oplus	Mass of the Earth [kg]
m	Mass of object [kg]
m	Order of Legendre polynomial [-]
\mathbf{n}	Unit vector normal to surface A [-]
n	Mean motion [$^\circ/\text{s}$]
n	Degree of Legendre polynomial [-]
P_n	Legendre polynomial [-]
$P_{n,m}$	Associated Legendre polynomial [-]
P_r	Solar radiation pressure [N/m^2]
p	Semi-latus rectum [m]
p_j	Generalized momenta
p_v	Impulse carried by photon [$\text{N}\cdot\text{m}$]
q_j	Generalized coordinates
R	Radius of celestial body [m]
R	Perturbing potential [m^2/s^2]
R_\oplus	Equatorial radius of the Earth [m]
$\tilde{R} (= -R)$	Force function notation for perturbing potential [m^2/s^2]
\mathbf{R}_x	Matrix for rotation about the x-axis [-]
\mathbf{r}	Position vector of orbiting object [m]
$\dot{\mathbf{r}}$	Velocity vector of orbiting object [m/s]
$\ddot{\mathbf{r}}$	Acceleration vector of orbiting object [m/s^2]
$\Delta\mathbf{r}$	Deviation of actual trajectory from reference orbit [m]
\mathbf{r}_\odot	Vector from the object to the Sun [m]
\mathbf{r}_{Sun}	Position vector of the Sun [m]
r	Radius [m]
r_a	Radius of apocenter [m]
r_M	Distance of the Moon from the center of the Earth [m]
r_p	Radius of pericenter [m]
r_{Sun}	Distance of the Sun from the center of the Earth [m]
r_\odot	Distance of object from the Sun [m]

$S_{n,m}$	Gravity field coefficient [-]
s	Position vector of body of mass M [m]
s	Position vector of a point inside the Earth [m]
s	Distance from the center of the Earth to a point inside the Earth [m]
T	Kinetic energy [m ² /s ²]
T	Orbital period [s]
T	Number of Julian centuries since 1.5 January 2000 (J2000)
t	Time [s]
t_0	Reference time for mean anomaly [s]
U	Potential [m ² /s ²]
V	Velocity [m/s]
V	Potential energy [m ² /s ²]
v	Velocity vector of object [m/s]
v	Orbital velocity [m/s]
x, y, z	Cartesian position coordinates [m]
$\dot{x}, \dot{y}, \dot{z}$	Cartesian velocity components [m/s]
α	Right ascension [°]
β_M	Latitude of the Moon [°]
δ	Declination [°]
$\delta_{n,m}$	Kronecker delta [-]
\mathcal{E}	Orbital energy [J]
ε	Obliquity of the ecliptic [°]
ε_j	Emissivity of surface element j [-]
η	Approximate solution function
θ	True anomaly [°]
θ	Angle between the vector normal to surface A and the vector pointing to the Sun [°]
λ	Longitude [°]
λ_M	Longitude of the Moon with respect to the equinox and year 2000 ecliptic [°]
λ_{Sun}	Sun's ecliptic longitude [°]
$\mu (=GM)$	Standard gravitational parameter [m ³ /s ²]
v	Shadow function [-]
ρ	Position vector of object in case it would follow the unperturbed reference orbit [m]
ρ	Mass density [kg/m ³]
ρ	Reflection coefficient [-]
τ	Time of pericenter passage [s]
τ	Time step (symplectic integrators) [s]
Υ	Direction of the Vernal Equinox [-]
ϕ	Increment function
ϕ	Geocentric latitude [°]

ϕ	Solar flux [W/m ²]
Ω	Right ascension of the ascending node [°]
ω	Argument of pericenter [°]

List of Abbreviations

AN	Ascending Node
AU	Astronomical Unit, $1 \text{ AU} = 149.5978707 \cdot 10^6 \text{ [km]}$
CNES	Centre National d'Etudes Spatiale
CPU	Central Processing Unit
DOPRI8	Dormand-Prince 8(7)
EOL	End-Of-Life
ESA	European Space Agency
GEO	Geostationary Orbit
GNSS	Global Navigation Satellite System
GPS	Global Positioning System
IADC	Inter-Agency Space Debris Coordination Committee
ISS	International Space Station
JAXA	Japan Aerospace Exploration Agency
JD	Julian Date
JGM	Joint Gravity Model
JPL	Jet Propulsion Laboratory
LEO	Low Earth Orbit
MASTER	Meteoroid And Space Debris Terrestrial Environment Reference
MEE	Modified Equinoctial Elements
MEO	Medium Earth Orbit
MJD	Modified Julian Date
MLI	Multi-Layer Insulation
MSc	Master of Science
NASA	National Aeronautics and Space Administration
ODE	Ordinary Differential Equation
RAAN	Right Ascension of the Ascending Node
RAM	Random-Access Memory
RK4	Runge-Kutta 4
RKF56	Runge-Kutta-Fehlberg 5(6)
RSA	Russian Space Agency
SDAG	Space Debris Advisory Group

SI	Système International d'Unités
SRP	Solar Radiation Pressure
SSN	Space Surveillance Network
T+V	(Splitting in) kinetic and potential energy
Tudat	TU Delft Astrodynamics Toolbox
USSTRATCOM	United States Strategic Command
WH	Wisdom-Holman (splitting)

Chapter 1

Introduction

Space debris is a large problem for spaceflight, and is bound to become an even larger problem in the future. In fact, if no countermeasures are taken, vast increases in the amount of space debris might even make spaceflight in the vicinity of the Earth impossible in the future (Klinkrad, 2006). This makes space debris both an interesting and a relevant topic for research.

This thesis project focuses on simulating the long-term evolution of space debris orbits, primarily in two distinct regions of space: the geostationary region and the region in which the satellites of GPS and other global navigation satellite systems (GNSS) reside. In this report, these regions will simply be referred to as the GEO and GNSS regions, respectively. Since there is no atmosphere to speak of in these regions of space, there is practically no decay due to atmospheric drag. As a result, debris objects in these regions generally have very long orbital lifetimes.

Simulating the orbits of debris objects over long periods of time is not an easy task. The methods to be used for this must remain accurate over long timescales, but also need to be efficient. An important goal of this thesis project is to assess which methods are the most efficient for the computation of long-term space debris trajectories for specified accuracy requirements, and to quantify the differences in efficiency between methods with varying degrees of complexity. Of special interest here are a set of relatively new methods, called symplectic integration methods. Because of their underlying mathematical structure, they have excellent energy conservation properties when applied to Hamiltonian systems, like the n -body problem. Simulating the trajectories of space debris objects involves the integration of *perturbed* Hamiltonian systems, though, and it will be interesting to see how the performance of symplectic integrators holds up when applied to these systems.

To perform the simulations of space debris orbits and test the different computational methods, a simulation code in C++ was created for this thesis project, capable of propagating the orbits of debris objects over several centuries, while taking into account various relevant perturbing forces. The simulation code is able

to simulate the orbits of debris objects with diverse characteristics, ranging from full satellites and upper rocket stages to tiny flecks of paint and detached surface coatings. In addition to the assessment of different computational methods, the simulation code will also be used for testing the suitability of different graveyard orbits for preventing collision hazards with active satellites over long periods of time.

The report is structured as follows. In Chapter 2, a general introduction to the problem of space debris will be given and a number of important aspects related to it will be discussed. Then, Chapter 3 will provide an overview of the debris populations in the GEO and GNSS regions of space. Existing research regarding debris in these regions will also be considered in this chapter. After that, Chapter 4 will go into detail about the different perturbing forces that were included in the force model and how they were modeled. Chapter 5 continues with a short review of astrodynamics and introduces different ways in which the equations of motion can be formulated, as well as a number of relevant coordinate systems. Traditional integration methods will be treated in Chapter 6, along with a number of concepts that are relevant for all integration methods. Following that, symplectic integration methods will be covered in Chapter 7. Then, Chapter 8 will go into detail about the verification of the simulation code. After that, Chapter 9 will provide a detailed performance comparison of the computational methods that were implemented in the simulation code. Then, the long-term orbital developments of debris objects in graveyard orbits will be considered in Chapter 10. Finally, the main conclusions of this thesis project and a number of recommendations for future research will be presented in Chapter 11.

Chapter 2

Space Debris

Space debris or *orbital debris* has been defined by the Inter-Agency Space Debris Coordination Committee (IADC) as “all man-made objects including fragments and elements thereof, in Earth orbit or re-entering the atmosphere, that are non-functional” (IADC, 2002). This chapter will provide some background information on space debris. First, the historic development of the space debris population will be outlined, followed by a brief discussion on the dangers related to space debris. After that, a short overview of the space debris environment will be given. Then, the major sources of space debris will be identified and the characteristics of debris objects that are important in orbit simulations will be determined. Finally, the guidelines that have been defined regarding space debris will be discussed.

2.1 History

On October 4, 1957 Sputnik 1 was launched, becoming the first artificial satellite in orbit around the Earth. After that date, many launches into space would follow, mainly conducted by the United States and the (former) Soviet Union.

The first in-orbit break-up event in space history occurred on June 29, 1961, after the US Transit-4A satellite was launched on a Thor-Ablestar rocket. 77 minutes after the injection and separation of the Transit-4A and two additional payloads, the Ablestar upper stage exploded at an orbital altitude of around 900 km. This distributed its dry mass of 625 kg across at least 298 trackable fragments, of which nearly 200 were still in orbit 40 years later. This break-up event instantaneously increased the number of man-made objects in space by at least a factor 3.5 (Klinkrad, 2006). Since this event, space debris has been the largest contributor to the observable space object population, with in-orbit explosions being the largest source.

In August 1964 the first geostationary satellite, Syncom-3, was launched. Since then, over 800 objects have been placed in or near the geostationary orbit (GEO). The first spacecraft explosion in GEO occurred 14 years after the Syncom-3 launch,

in June 1978. As a reaction to this, the Czech astronomer Loboš Perek wrote a paper in 1979, titled "Outer Space Activities Versus Outer Space", and was the first to recommend space debris mitigation measures (Portree and Loftus, 1993). His recommended measures, which included the re-orbiting of GEO spacecraft into a disposal orbit at the end-of-life (EOL), are still applicable today.

Figure 2.1 shows the historic evolution of the number of cataloged orbital objects tracked by the Space Surveillance Network (SSN) of the United States Strategic Command (USSTRATCOM). As can be seen, the number of cataloged objects increased steadily over the years, with some smaller and larger peaks in between. It should be noted that the graph only indicates objects that are *trackable* by the 29 tracking stations of the SSN. For low Earth orbits (LEO), this means that only debris objects of diameters of 10 cm and larger are included, whereas for higher orbits the minimum detection diameter is larger (USSTRATCOM, 2011).

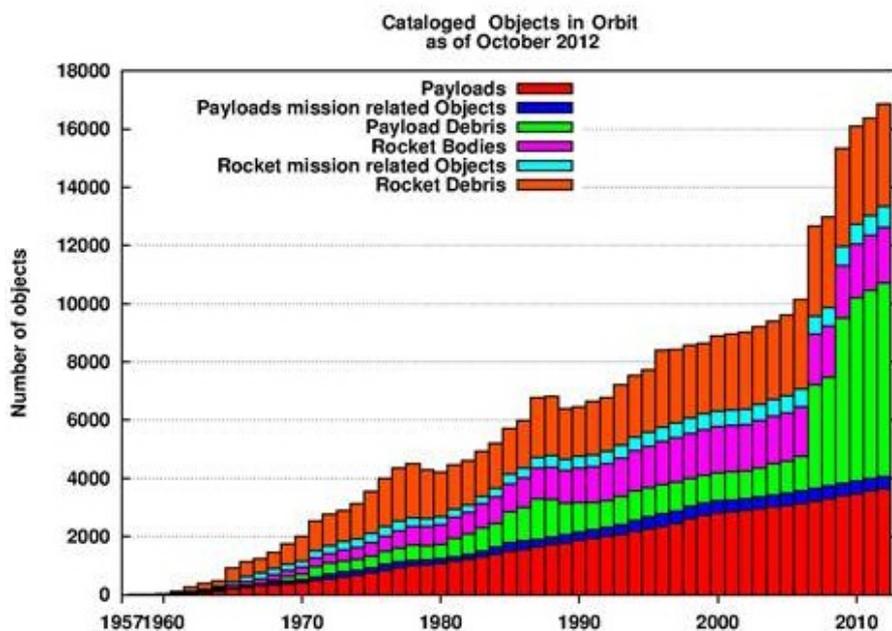


Figure 2.1: The evolution of the number of cataloged objects in orbit (ESA, 2013).

In recent years, some notable events happened with regard to space debris. In 2007, China performed an anti-satellite test, using a missile to destroy an old weather satellite. This event added more than 3,000 new objects to the space debris population (NASA, 2010). The abrupt rise in the number of trackable objects visible in Figure 2.1 for the year 2007 is a result of this test.

Furthermore, the first collision between two intact satellites in orbit around the Earth occurred on February 10, 2009. This collision involved a defunct Russian Kosmos satellite and a functioning US Iridium communications satellite and added over 2,000 pieces of trackable debris to the debris population. The effects of this

event are also visible in Figure 2.1.

2.2 Dangers Related to Space Debris

Debris objects pose a potential collision hazard to satellites and other spacecraft. Because the relative velocities between debris objects and spacecraft can be extremely high, ranging from 0 to 15 km/s, the energy involved in collisions between the two can be enormous. In many cases, the processes taking place do not fall into the categories normally known from solid-state physics. The materials of both the impacting object and the satellite wall may liquefy, evaporate or even ionize, and the metal droplets, solidifying again, will splash into the inside of the satellite. In addition, the accompanying shock waves can tear apart the entire satellite structure (Rex, 1998). In order to study such impacts in a laboratory environment on Earth, objects in the range of a centimeter need to be accelerated to velocities of around 10 km/s (or 36,000 km/h). Such shots are about the upper limit of what is possible in the military field of armor-piercing projectiles.

Consequently, shielding spacecraft against debris objects that exceed a certain size is virtually impossible. The International Space Station (ISS) has special shielding against space debris, but even these shields can only be effective in withstanding impacts of particles smaller than 1 cm (NASA, 2010). Collisions with larger objects need to be avoided by maneuvering the ISS in advance.

As the energy associated with collisions is often extremely large, the impact of a debris object of any significant size will generally result in the loss of the spacecraft. In addition, many new debris objects will be formed, which in turn increases the probability for future collisions. According to a scenario proposed by NASA scientists Kessler and Cour-Palais (1978), called the *Kessler Syndrome*, this may eventually lead to a cascade of collisions which could render space exploration, and even the use of satellites, unfeasible for many generations. Therefore, careful study of the space debris problem and the application of suitable counter-measures is considered essential for the future of spaceflight.

2.3 The Space Debris Environment

The higher the orbital altitude, the longer space debris objects will typically remain in orbit. If a debris object is left in an orbit below 600 km, the orbit will normally decay within several years due to aerodynamic drag and eventually the object will experience severe aerodynamic heating. Most debris objects do not survive this severe heating during re-entry and burn up in the atmosphere. Objects which do survive are most likely to fall into the oceans or onto sparsely populated regions of land. Up till now, no serious injury or property damage due to re-entering debris has been confirmed (NASA, 2011).

At altitudes of around 800 km, orbital decay generally takes several decades. Above 1,000 km, debris objects will normally continue to orbit the Earth for a

century or more, as the atmosphere at these altitudes is extremely tenuous.

A global overview of the cataloged objects in space is shown in Figure 2.2. As can be seen in the figure, most orbital debris resides within 2,000 km of the Earth's surface. Within this region, the amount of debris varies quite strongly with altitude. The largest concentrations of debris are found around 800 - 850 km (NASA, 2011). Moreover, another clear peak in the number of objects can be observed in the geostationary region, at an altitude of roughly 36,000 km. More information about the debris situation in this region will be given in Chapter 3. The debris environment of the region in which the satellites of GPS and other satellite navigation systems reside will also be discussed in that chapter.

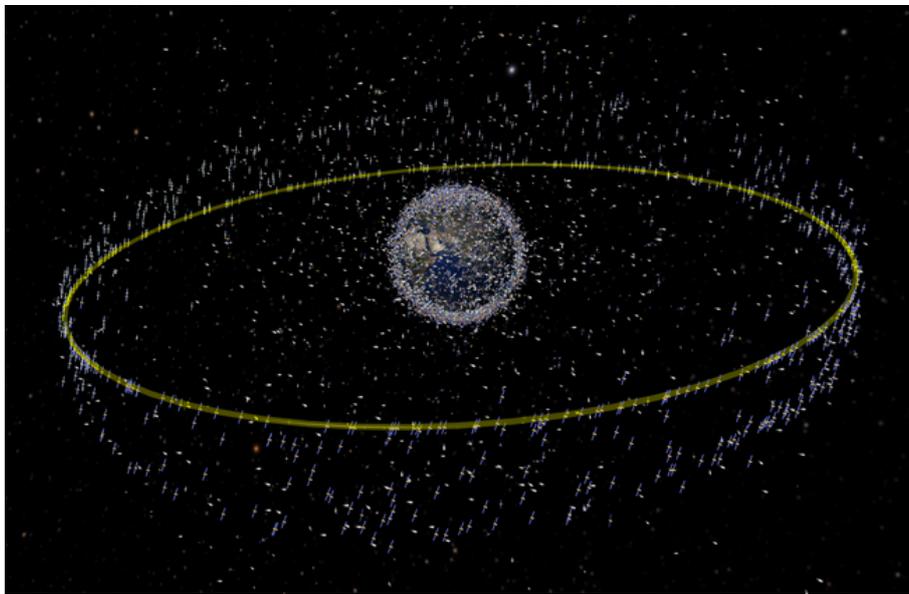


Figure 2.2: A global overview of the cataloged objects in space. The geostationary ring is indicated in yellow (ESA, 2009).

2.4 Sources of Space Debris

Space debris can originate from a number of different sources. The most important sources are (NASA, 2011):

- **Upper stages of launch vehicles:** When a satellite is brought into orbit, the upper stage of the launch vehicle usually has about the same final velocity as the satellite it was carrying and will remain in a similar orbit.
- **Derelict satellites:** Once a satellite is no longer functional it becomes a debris object. If no end-of-life de-orbit is carried out, it may remain in space for a very long time. This is especially the case for satellites in high Earth

orbits, where it is generally not feasible to apply an end-of-life impulse large enough for a de-orbit and where the atmosphere is so tenuous that there is almost no natural decay to speak of.

- **Explosion fragments:** Explosions of satellites and rocket bodies (see Figure 2.3) are the dominant source of in-orbit fragmentations, responsible for the vast majority of more than 200 registered break-up events (ESA, 2009). The main cause of in-orbit explosions is related to the residual fuel that remains in tanks or fuel lines of discarded satellites and rocket stages. Over time, the deteriorating effects of the harsh space environment on the mechanical integrity of components can lead to leaks and/or mixing of fuel components, which could trigger self-ignition.
- **Collision fragments:** Collisions also lead to in-orbit fragmentations, but up till now only a few registered collisions have occurred. The number of break-up events due to explosions far exceeds the amount of break-ups due to collisions (ESA, 2009). This may change, however, as the number of debris objects increases and the collision probabilities rapidly increase with it.
- **Solid rocket motor effluents:** When firing solid rocket motors, slag and dust particles are released. These particles are very small in size, ranging from a few micrometers to several millimeters.
- **Surface fragments:** Under the influence of ultra-violet radiation, atomic oxygen and impacting micro particles, the surfaces of objects in space start to degrade. This can lead to the detachment of paint and surface coatings, resulting in micrometer- to millimeter-sized debris.
- **Mission related objects:** Some objects are only needed for part of a space mission and are released into space once they are no longer required. A few examples of such objects are launch adapters, apogee kick-motors and covers for optical instruments.
- **Lost equipment:** Over the years, a number of different items have been lost by astronauts during extra-vehicular activities. These items include gloves, cameras, screwdrivers and quite a few other objects. However, most of them have burned up in the atmosphere relatively quickly.



Figure 2.3: Debris creation at the explosion of a rocket body (ESA, 2009).

2.5 Characteristics of Debris Objects

Debris objects can be very diverse, ranging from defunct satellites to used rocket stages and from paint flakes to residues from solid rocket motor firings. These objects all have different masses, sizes, reflection coefficients and other properties. In the simulations, which characteristics are of importance depends on the forces that are taken into account.

Gravitational forces depend on the mass of the object. However, it is the resulting acceleration of the object that is important, and this quantity is independent of the mass of the object. Furthermore, since all debris objects are negligibly small compared to the sizes of the celestial bodies that have any significant gravitational effect on them, debris objects are usually modeled as point masses as far as gravitational attraction is concerned. As a result, the physical properties of an object do not have an influence on how the object is affected by gravity in the simulations.

The effect that solar radiation pressure has on the trajectory of an object, on the other hand, is highly dependent on the physical properties of the object. As will be detailed in Section 4.5, the acceleration due to radiation pressure is linearly dependent on the *radiation pressure coefficient* C_R . Moreover, as solar radiation can be seen as a flux of energy, its effect on the acceleration of an object is directly proportional to the cross-sectional area of the object, A . Finally, the force on an object due to solar radiation is not dependent on the mass of the object, m . Hence, as a result of Newton's second law, the resulting acceleration is inversely proportional to m . The latter two properties are often combined to form the *area-to-mass ratio* A/m . Together with the radiation pressure coefficient, this ratio determines how the trajectory of an orbiting object will be affected by solar radiation pressure.

There are also other properties of debris objects that influence how certain

forces act on these objects. An example is the drag coefficient C_D , which scales the aerodynamic force. However, as will be discussed in Chapter 4, the aerodynamic force and many other forces are very small at GEO and GNSS altitudes compared to other perturbing forces and will be neglected in the simulations. Therefore, the properties that will characterize different debris objects will be the radiation pressure coefficient C_R and the area-to-mass ratio A/m . When varying object parameters in the simulation, these properties will be combined into one factor: $C_R \cdot A/m$. This factor is also called the *effective area-to-mass ratio* in the remainder of the report.

As will become clear in Section 4.5, C_R is defined such that it will always have a value between 1 and 2. A/m , on the other hand, can have values as low as 0.01 m²/kg for intact satellites and values as high as 40 m²/kg or more for pieces of foil used in multi-layer insulations of spacecraft (Schildknecht et al., 2008). Therefore, in literature often only the area-to-mass ratio is mentioned when discussing debris objects. In this report, the area-to-mass ratio will be used when generally discerning between different types of debris objects, while the exact value of $C_R \cdot A/m$ will be listed when simulation results are shown.

2.6 Guidelines Regarding Space Debris

According to Rex (1998), space debris reached the highest political levels for the first time in 1988, when US President Reagan announced the policy that “all space sectors will seek to minimize the creation of space debris”. This was after years of research in the US and awareness-raising conferences by various institutes. A year earlier, in 1987, ESA set up the Space Debris Working Group (currently called the ESA Space Debris Advisory Group (SDAG)) to focus on space debris related aspects of all ESA space projects. Realizing that all space-faring nations needed to cooperate on the problem, NASA, ESA and the Russian, Chinese, Japanese and other space agencies pooled their knowledge in the Interagency Space Debris Coordination Committee (IADC) in 1993. The primary purposes of the IADC are to exchange information on space debris research activities between member space agencies, to facilitate opportunities for cooperation in space debris research, to review the progress of ongoing cooperative activities and to identify debris mitigation options (IADC, 2011).

Up till now, no international treaty has been established yet that mandates behavior to minimize space debris. However, it is in the best interest of society to minimize the growth of the space debris problem. Therefore, many space agencies have defined guidelines regarding space debris. In the context of this thesis report, it is mainly important to consider the end-of-life guidelines that have been formulated for satellites in the GEO and GNSS regions of space. An overview of the guidelines for these regions employed by various space agencies is given in Tables 2.1 and 2.2. Although they do vary in details, the general objective of each of the guidelines is the same: at end-of-life, the spacecraft needs to be put in such an orbit

that it will not pose harm to other, still operational spacecraft.

Table 2.1: End-of-life guidelines for satellites in the GEO region, as defined by various space agencies (Klinkrad, 2006).

	GEO region
NASA	re-orbit above GEO so that $\Delta H_{1,geo}$ [km] $> 300 + 1,000 A/m$
JAXA	re-orbit above GEO so that $\Delta H_{2,geo}$ [km] $> 200 + 0.022 a \cdot C_R \cdot A/m$
CNES	re-orbit above GEO so that $\Delta H_{3,geo}$ [km] $> 235 + 1,000 C_R \cdot A/m$
RSA	re-orbit above GEO so that $\Delta H_{4,geo}$ [km] > 200

Table 2.2: End-of-life guidelines for satellites in 12-hour circular orbits (e.g. GPS satellites), as defined by various space agencies (Klinkrad, 2006).

	GNSS region
NASA	re-orbit so that either [$H_p > 2,500$ km and $H_a < 19,900$ km] or [$H_p > 20,500$ km and $H_a < 35,288$ km]
JAXA	re-orbit so that either [$H_p > 1,700$ km and $H_a < 19,900$ km] or [$H_p > 20,500$ km and $H_a < 35,288$ km]
CNES	no requirement for a disposal maneuver
RSA	only a general recommendation for EOL lifetime limitation

Chapter 3

The GEO and GNSS Debris Environments

As was mentioned in the introduction, this thesis project primarily focuses on debris in the GEO and GNSS regions of space. This chapter takes a closer look at the debris situations in these regions. First, the debris population in the GEO region will be examined, followed by an overview of existing research on the long-term orbital development of debris in that region. After that, the debris population in the GNSS region will be considered, and an overview will be given of existing research on the long-term behavior of orbits of debris objects in that region.

3.1 The GEO Debris Environment

Orbits with an altitude of 35,786 km above the equator have the unique property of having an orbital period equal to the rotational period of the Earth. Such orbits are called *geosynchronous* orbits. A special kind of geosynchronous orbit is the *geostationary* orbit, being a circular geosynchronous orbit with zero inclination. As the name implies, a satellite in a geostationary orbit remains at a fixed position with respect to the surface of the Earth. This is very convenient for a wide variety of applications, making the geostationary region arguably the most valuable region of space.

3.1.1 Debris Population in the GEO Region

The historic growth of the number of trackable objects in the GEO region is shown in Figure 3.1. In the figure, a distinction is made between geostationary and super-geostationary objects, the latter referring mainly to objects in super-GEO graveyard orbits. It can be seen that the number of objects in the GEO region steadily increases with about 30 objects per year.

Note that this concerns trackable objects. Objects which are too small to be detected – of which there are likely many more – are not included in the figures. Using telescopes built for detecting space debris objects as small as ~ 15 cm can be observed in the geostationary ring. However, it is hard to estimate object sizes from telescope observations, as objects change their brightness depending on attitude, phase angle and albedo (Jehn et al., 2006).

Much of the debris that cannot be detected has resulted from a number of in-orbit explosions in the geostationary region. The distribution of this debris is often taken into account using break-up models (Ikeda et al., 2008). As a result, the quantity of smaller objects in the population has a large uncertainty related to it.

Figure 3.2 shows the spatial density of objects larger than 10 cm in the GEO region that are included in the debris population of ESA's Meteoroid and Space Debris Terrestrial Environment Reference (MASTER) model.

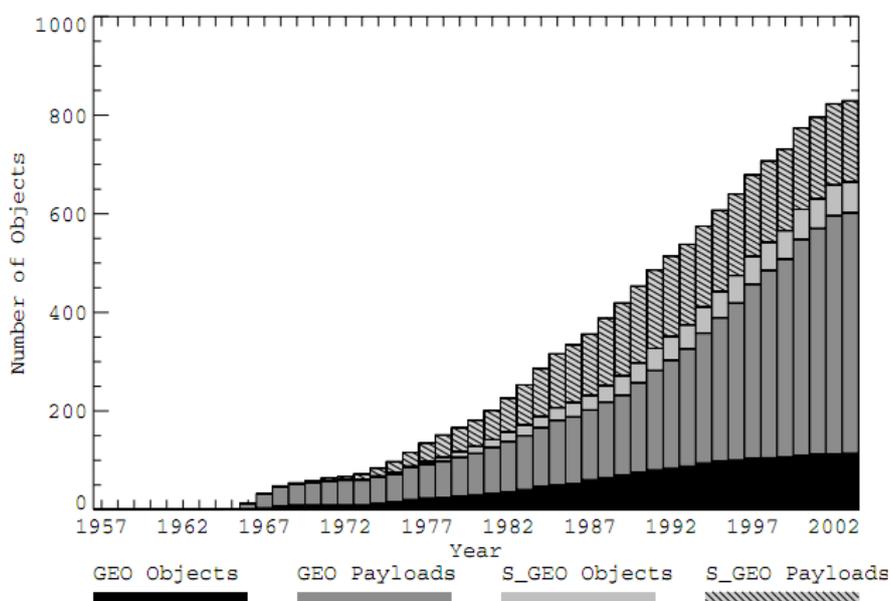


Figure 3.1: The evolution of the number of trackable GEO and super-GEO catalog objects over time. In the division used in the plot, the term 'payloads' indicates active and non-active satellites, whereas the term 'objects' refers to other objects, such as rocket bodies and boost motors (Klinkrad, 2006).

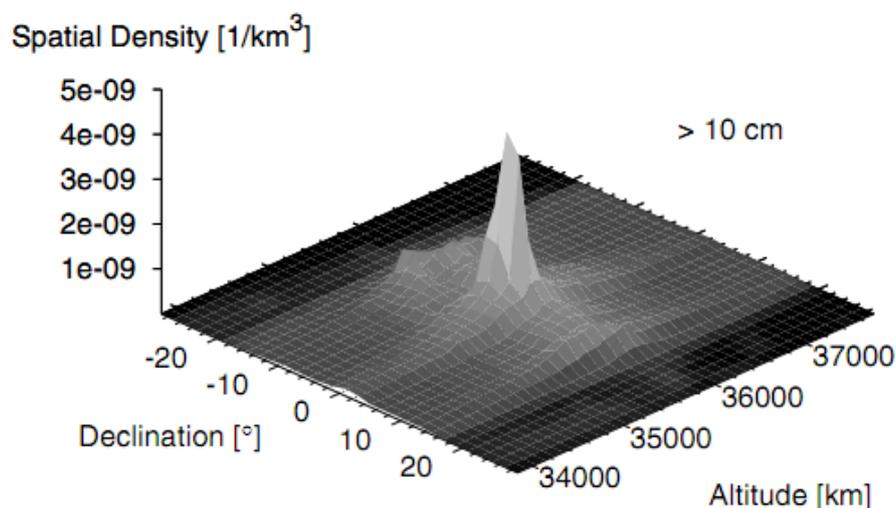


Figure 3.2: The spatial density of debris objects larger than 10 cm in GEO versus altitude and declination according to the MASTER-2001 model (Klinkrad, 2006).

As can be seen in Figure 3.2, the vast majority of objects larger than 10 cm is concentrated in or near geostationary orbit, with orbital altitudes of around 35,786 km and declinations in the range of zero degrees. According to Klinkrad (2006), objects which are no longer controlled undergo a cyclic variation in inclination with a period of 53 years and an amplitude of about 15° due to luni-solar perturbations in concert with stabilizing Earth oblateness effects. This explains the ridge in the figure across declinations of around $\pm 15^\circ$ at near-constant geosynchronous altitude.

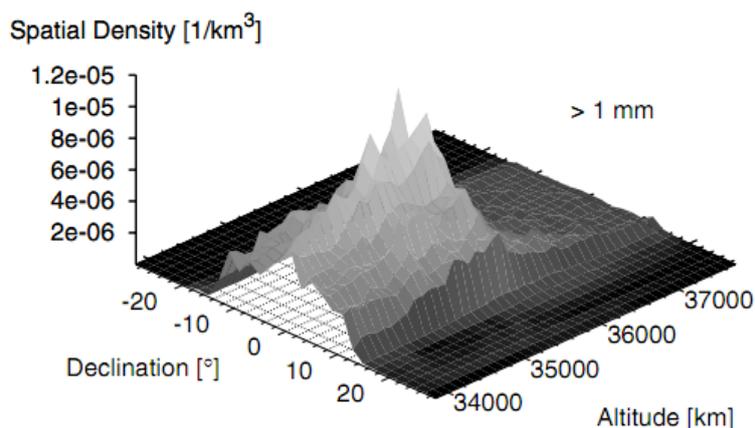


Figure 3.3: The spatial density of debris objects larger than 1 mm in GEO versus altitude and declination according to the MASTER-2001 model (Klinkrad, 2006).

Figure 3.3 shows the MASTER population for objects larger than 1 mm in the

GEO region. Clearly, the peak in object density is still at the conditions of the geostationary orbit, but the relative object density in surrounding areas is much larger than was the case for larger debris objects (cf. Figure 3.2). This is due to the fact that many small debris objects originate from in-orbit explosions, which lead to a large range of velocity vectors for the explosion fragments. Note that the object densities in Figure 3.3 are estimates based on simulations, rather than measured quantities. Also note that the object densities in the figure are about a factor 1,000 higher than in Figure 3.2.

3.1.2 Existing Research

An important point to consider regarding orbital debris in the GEO region is that the relative velocities between debris objects and satellites are significantly smaller than in LEO. This is a result of the fact that virtually all satellites in the GEO region have been launched into geostationary orbits, meaning they almost all started in prograde orbits with (roughly) the same altitude and inclination. Hence, typical collision velocities in GEO rarely exceed more than a few km/s (Matney, 2004).

In fact, the relative speed is usually a factor 5 or 10 smaller than in LEO, with the kinetic energy being many tens of times smaller. This means that, in general, catastrophic collisions in GEO require larger objects than in LEO. Objects of smaller sizes are often also harmful, but lead to a reduction in capability instead of destruction of the spacecraft. Nevertheless, given the unique characteristics of geostationary orbits, debris in the GEO region is carefully studied.

Anselmo and Pardini (2005) examined the dynamical evolution of objects released in geostationary orbits, with area-to-mass ratios ranging from 1 to 50 m²/kg. The study included both short-term (a few months) and long-term (54 years) simulations. The results indicated that the orbits of some objects developed eccentricities of up to 0.8, while still retaining mean motions of one revolution per day. It was concluded that for $A/m > 40$ m²/kg, orbital decay was attained in less than 40 months due to the eccentricity rise, while for lower values of A/m , a lifetime of at least two decades was found.

At NASA's Orbital Debris Program Office, a similar kind of study on the orbital evolution of debris in the GEO region was performed, using A/m values in the range of 0.1 to 35 m²/kg (Liou and Weaver, 2005). The simulation results indicated that solar radiation pressure could cause the orbit of a high A/m object to undergo a significant yearly variation of eccentricity, of which the amplitude would increase with increasing A/m values. Also, the orbits of debris objects with A/m values of 20 m²/kg or higher were found to yield maximum eccentricities of 0.55 or higher. Pieces of MLI blankets were identified as being the most likely sources of such debris. The existence of such debris objects with extremely high area-to-mass ratios (ranging from 1 to more than 40 m²/kg) has been confirmed using telescope observations (Schildknecht et al., 2008).

Next to the smaller debris objects discussed above, inactive satellites are also an important part of the debris population in the GEO region. While these debris

objects have significantly smaller area-to-mass ratios, they do have much larger masses than most other debris objects, resulting in a large damage potential in case of collisions.

Lewis et al. (2004) studied the stability of disposal orbits at super-GEO altitudes by simulating the evolution of these orbits over 200-year periods. The results indicated that the initial perigee and eccentricity of a disposal orbit were the most important factors for maintaining the orbit above the protected region. The evolution of the disposal orbit was also found to be sensitive to the initial lunar right ascension of the ascending node (RAAN) and the RAAN and argument of perigee of the disposal orbit, though these sensitivities were only minor.

At TU Delft, the long-term evolution of super-geostationary disposal orbits was also investigated by Van Kints (2005). In this study, it was found that circular disposal orbits for objects with properties typical for geostationary satellites ($A/m = 0.02 \text{ m}^2/\text{kg}$, $C_R = 1.3$) showed stable characteristics, with no secular effects for the eccentricity evolution. It was concluded that the recommended approach of the IADC was effective for preventing collision possibilities with active geostationary satellites, also over long periods of time.

3.2 The GNSS Debris Environment

The satellites of GPS and other global navigation satellite systems (GNSS) are located in a specific region of space, with altitudes of around 20,000 km above the Earth's surface. Given the fact that daily life becomes increasingly dependent on applications of GNSS, the debris situation in this region of space is highly relevant.

3.2.1 Debris Population in the GNSS Region

An overview of the number of cataloged super-LEO objects with the semi-major axes of their orbits is given in Figure 3.4. In this figure, it can be seen that there are relatively large concentrations of objects at semi-major axes of around 26,500 km and 42,500 km. The latter peak represents the geostationary region, which was discussed in Section 3.1.

The region around a semi-major axis of 26,500 km (or altitude of roughly 20,000 km) is the GNSS region, in which nearly all navigation satellites are located. The GNSS constellations consist of relatively large numbers of satellites that share the same altitude, but are divided over a number of orbital planes. This ensures that there are always enough satellites visible from any location on the Earth to accurately calculate the position of the user. Orbital data on the major operational and planned satellite constellations in the GNSS region is given in Table 3.1. Next to the GPS (United States), Glonass (Russian Federation) and Galileo (European Union) systems, China is developing the Beidou/Compass navigation system. This system will consist of a combination of geostationary satellites and MEO satellites and is aimed to provide global coverage by 2020 (Jin, 2013).

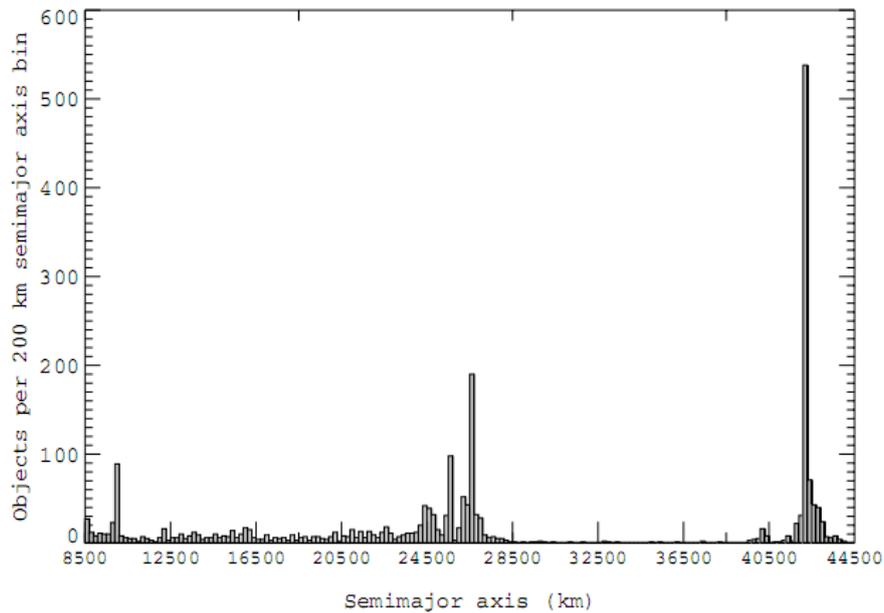


Figure 3.4: The number of cataloged super-LEO objects with their corresponding semi-major axes. The diagram is based on data from June 2003 (Klinkrad, 2006).

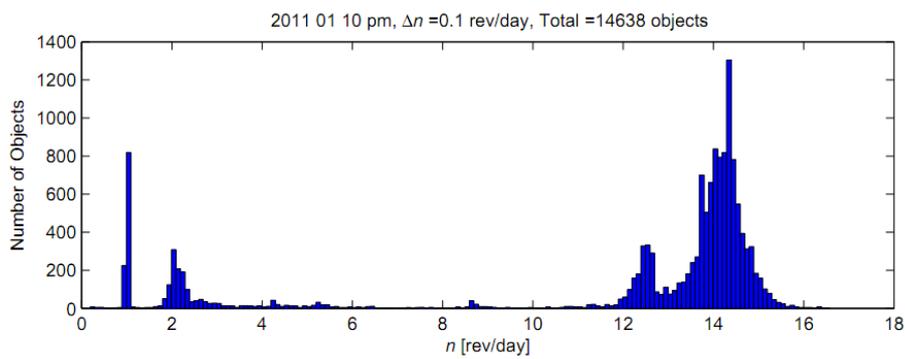


Figure 3.5: The number of cataloged objects per mean motion bin of 0.1 rev/day. The diagram is based on data from the two-line element catalog of January 2011 (Leloux, 2012).

Table 3.1: Orbital data on operational and planned satellite constellations in the GNSS region (Klinkrad, 2006), (Wang et al., 2009).

Constellation	H_p [km]	H_a [km]	i [°]	Satellites + spares	No. of planes	Satellites per plane
Glonass	19,132	19,132	64.8	24 + 3	3	8 (+1)
GPS/Navstar	20,182	20,182	55.0	24 + 3	6	4 (+0/1)
Beidou	21,500	21,500	55.0	27 + ?	3	9 (+?)
Galileo	23,616	23,616	56.0	27 + 3	3	9 (+1)

Similar to Figure 3.4, Figure 3.5 gives an overview of the number of cataloged objects in space. This figure is based on more recent data (status January 2011) provided by the Space Surveillance Network of the United States Strategic Command. As stated in Section 2.1, the data only includes objects of 10 cm or larger for LEO, while the detection threshold is significantly larger for the GNSS region.

Based on Figure 3.5, a rough estimate of the number of trackable objects in the GNSS region can be made. This region roughly spans the altitude range from 18,000 km to 24,000 km (see Table 3.1). This range approximately corresponds to a regime of mean motions between 1.6 and 2.3 rev/day. Adding the numbers of objects between these boundaries in Figure 3.5 yields that there were around 1,000 trackable objects in the GNSS region of space in January 2011. Though this is only a rough approximation, it does provide an order of magnitude regarding the number of relatively large objects in the orbital regime used by the GNSS satellites. Based on the plots, it can also be concluded that the GEO region contains about 2 to 3 times as many trackable debris objects as the GNSS region.

3.2.2 Existing Research

A number of different researchers have studied the long-term evolution of the orbits of debris objects in the GNSS region. Chao and Gick (2004) focused on the eccentricity evolution of disposal orbits of GPS, Glonass and Galileo satellites. They found that orbits that are initially near-circular may evolve into orbits with large eccentricities (up to 0.7 over 150 years). The underlying causes of this turned out to be resonances induced by the Sun/Moon and J_2 secular perturbations. These resonance effects were found to be strongly dependent on the inclination and altitude of the orbit. Another finding of the study was that Glonass satellites would start to enter the orbital regime used by the GPS constellation after 40 years.

A series of studies into debris behavior in the GNSS region was also carried out by Anselmo and Pardini. In one study, the long-term orbit behavior was investigated for debris objects with a large range of area-to-mass ratios, using different force model options (Anselmo and Pardini, 2009). It was found that for both low and high area-to-mass ratios, the orbital evolution was dominated by geopotential and luni-solar resonances. This confirms the findings of Chao and Gick (2004). Furthermore, it was noted that the direct solar radiation pressure would induce a

near-yearly oscillation of eccentricity and semi-major axis, whose amplitude would grow with increasing A/m values.

In a subsequent study (Anselmo and Pardini, 2010), it was stated that optical observations indicated that there was much more space debris with high area-to-mass ratios than previously anticipated. This concerned debris with A/m values hundreds or thousands of times greater than those of intact satellites. Simulations indicated that eccentricities of up to 0.7 could be reached for debris objects with such high A/m values, while the semi-major axes and mean motions of the orbits would remain close to the values of the original GPS orbits. Moreover, it was found that the lifetime of debris from GPS orbits with $C_R \cdot A/m$ values of up to $\sim 55 \text{ m}^2/\text{kg}$ exceeded one century. However, the lifetime would decrease rapidly to just a few months for values above this threshold.

In yet another study performed by Anselmo and Pardini (2011), the orbital evolution of the first upper stages for the Galileo and Beidou satellites was examined. It was concluded that the upper stage disposal strategies used for Beidou and Galileo are quite successful in preventing long-term interference with the navigation satellite constellations, provided that accidental breakups are prevented.

The results of the studies mentioned above were obtained using numerical simulations. Deleflie et al. (2011) used semi-analytical methods instead to study the long-term behavior of debris objects in the GNSS region. In the results, it is emphasized that the choice of initial conditions is critical. Especially the maximum eccentricity that can be reached is highly dependent on the initial conditions of the orbit, reaching a value of 0.7 or more in some cases, within a typical timescale of 200 years. This eccentricity growth corresponds well with the results obtained in the other studies. It should be noted that once such high eccentricities are reached, atmospheric drag starts to play a role and burn-up in the atmosphere is likely to follow.

Force Model

The orbit of a satellite or debris object about the Earth is primarily determined by the Earth's gravity field. To first approximation, the mass distribution of the Earth can be assumed to be radially symmetric. In that case, the orbit has the shape of a conic section and is called a Keplerian orbit (Wakker, 2010).

However, in reality the mass distribution of the Earth is not radially symmetric and furthermore, there are other forces than the Earth's gravity that affect the trajectory. These additional forces, including those resulting from the non-radially symmetric part of the Earth's mass distribution, make the real orbit deviate from a Keplerian one.

Since the additional forces are quite small compared to the main gravitational force, these forces are called *perturbing forces* or *perturbations*. The resulting orbit is called a *perturbed Keplerian orbit*.

In this chapter, an overview will be given of the various forces that play a role and a choice will be made regarding the forces to be taken into account for the simulation of orbits in the GEO and GNSS regions of space. After that, the forces that are taken into account will be discussed individually and it will be shown how they have been modeled in the simulation code.

4.1 Overview

In the literature survey preceding this thesis (Hofsteenge, 2012), the various perturbing forces acting on satellites and debris objects were discussed and compared. On the basis of this comparison, a choice has been made regarding the specific forces to be taken into account in the simulations. For completeness, a short overview of the forces that are acting will be provided in this section, including a short discussion on the forces to be included in the simulations. How these forces are modeled will be discussed in the subsequent sections.

An overview of the orders of magnitude of the forces that are acting, related to the distance from the center of the Earth, is given in Figure 4.1. Note that the

vertical axis has the unit $[\text{km}/\text{s}^2]$.

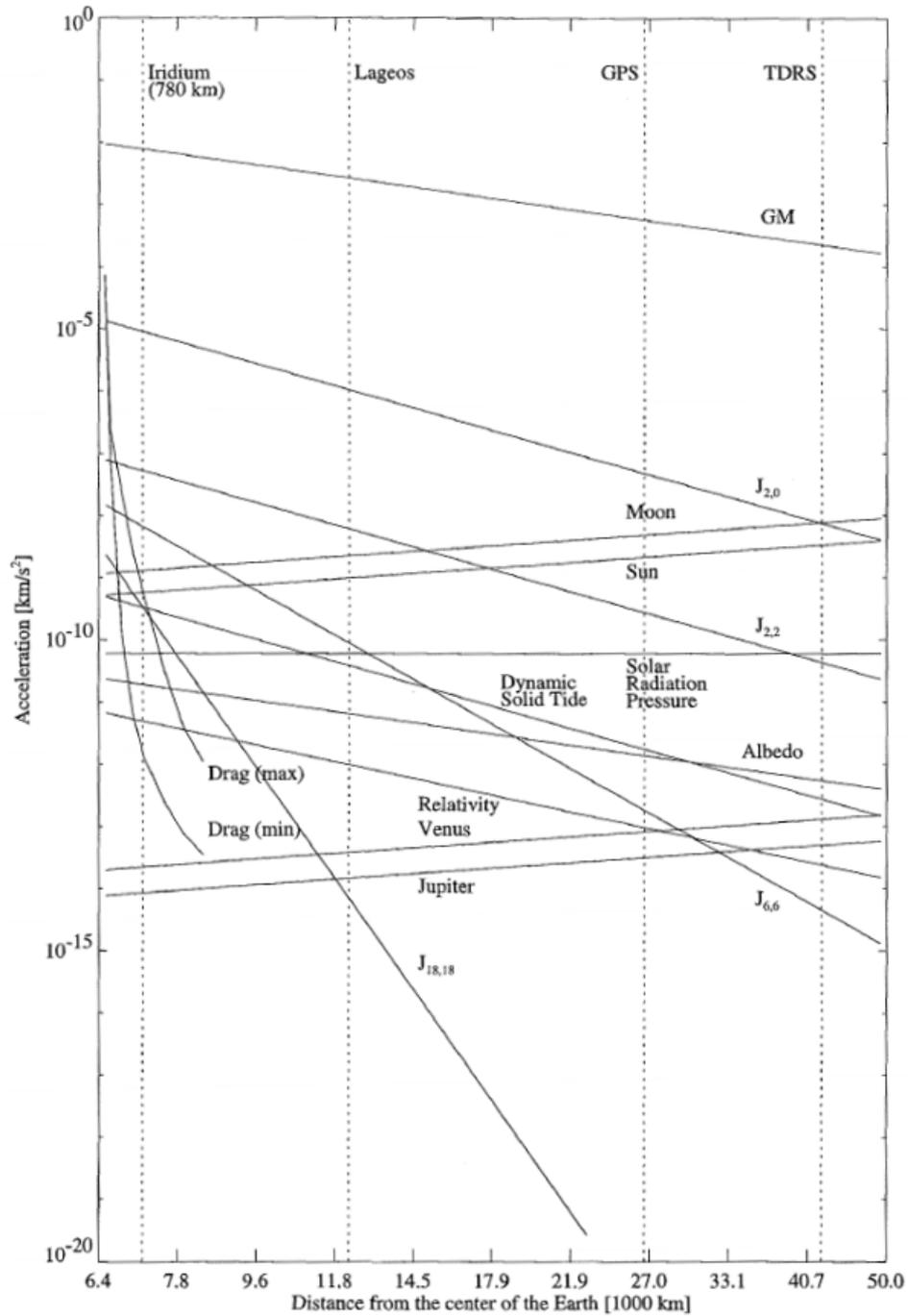


Figure 4.1: The magnitude of various perturbations as function of the distance from the center of the Earth (Montenbruck and Gill, 2005).

Since this thesis project is focused on the GEO and GNSS altitude regimes, the two dotted lines at the right-hand side of the figure indicate the most important areas of the graph. Clearly, all perturbations at these altitudes are at least three orders of magnitude smaller than the main gravitational acceleration of the Earth (indicated by GM). The $J_{2,0}$ -term of the Earth's gravity field yields the largest perturbing acceleration (also called the J_2 -effect), closely followed by the gravitational attractions of the Moon and the Sun. Note that the perturbing accelerations due to deviations in the Earth's gravity field decrease with increasing altitude, whereas the influence of the gravitational attraction of other celestial bodies increases with altitude.

The acceleration due to solar radiation pressure is one to two orders of magnitude smaller than the acceleration caused by the Moon and the Sun and is roughly in the same range as the acceleration due to the $J_{2,2}$ -term of the Earth's gravity field. However, this is for objects that have the characteristics of an intact satellite. For objects with very high area-to-mass ratios, such as pieces of MLI and paint flakes, solar radiation pressure will have a much larger perturbing effect. For these objects, the line for solar radiation pressure in Figure 4.1 may be shifted several orders of magnitude upwards.

The acceleration due to albedo at the GEO and GNSS regions is roughly two orders of magnitude smaller than the acceleration due to solar radiation pressure. In addition, there are many uncertainties associated with the modeling of albedo. Modeling its effect also requires a relatively high number of arithmetic operations due to the geometric aspects that are involved and will likely have a significant impact on the computation time required for simulations. For these reasons, albedo will not be included in the force model.

The perturbation caused by the Earth's dynamic solid tide is also multiple orders of magnitude smaller than the perturbation due to solar radiation pressure, and is usually only included for geodetic missions, which require extremely accurate orbit computations. Still smaller are the effects due to relativity and the gravity of Venus and Jupiter. These effects will also not be taken into account in the simulations.

Finally, the effect of atmospheric drag is completely negligible in the GEO and GNSS regions of space. Figure 4.1 illustrates this: even at an altitude of 10,000 km, the perturbations due to relativity and the perturbations due to the gravitational attraction of Venus and Jupiter will already be larger.

In some of the previous research mentioned in Sections 3.1.2 and 3.2.2, atmospheric drag was actually taken into account in case the orbit of the debris object would become highly eccentric and the perigee altitude would become smaller than 1,000 km. Generally, the object would then burn up in the atmosphere after a relatively short time. To take this effect into account, the simulation code developed for this thesis project continuously checks whether the altitude of the object is below 1,000 km. If this is the case, the simulation is stopped, based on the assumption that atmospheric drag will lead to reasonably fast decay. This way of handling drag removes the necessity of implementing a full-scale atmospheric model and

seems appropriate for the application of investigating the long-term developments of orbits that nominally remain within much higher altitude regimes.

Considering the above, the force model used in the simulations will include the main gravitational acceleration of the Earth, as well as a number of perturbations. These perturbations are the third-body perturbations due to the Sun and the Moon. The gravitational effects of other celestial bodies will not be taken into account, as they are simply too small to be worth the extra computation time. Solar radiation pressure will be included in the simulations, as its effect can be very noticeable, especially in case debris objects with high area-to-mass ratios are considered. Eclipses will also be taken into account. Deviations in the gravity field of the Earth will also be accounted for by means of an expansion in spherical harmonics. Based on the magnitudes of the geopotential coefficients and comparisons with other research, the maximum degree and order of the geopotential coefficients will be set to four in the simulations. However, this number can easily be increased or decreased in the simulation settings. The other perturbations mentioned in this section will not be modeled in the simulations, as their effect would be relatively unnoticeable and would not be worth the extra computational effort.

4.2 Central Gravitational Force

The central gravitational force is the force which is responsible for the main characteristics of the actual orbit. If only the central gravitational force is taken into account, an unperturbed Keplerian orbit will result, given appropriate initial conditions. In that case, it is assumed that the total mass of the Earth is concentrated at the center of the coordinate system, and the gravitational law can be stated as (Montenbruck and Gill, 2005)

$$\ddot{\mathbf{r}} = -\frac{GM_{\oplus}}{r^3}\mathbf{r} \quad (4.1)$$

in which \mathbf{r} is the position vector of the orbiting object from the center of the Earth, G is the universal gravitational constant, M_{\oplus} is the total mass of the Earth and r is the distance of the orbiting object from the center of the Earth.

Although completely Keplerian orbits are interesting from a theoretical point of view, in reality all orbits will deviate from them to some degree. Therefore, using only the formulation of Equation 4.1 as equation of motion will not result in sufficiently realistic trajectories. Though the right-hand side of Equation 4.1 will remain the most important acceleration, other acceleration components will have to be added to it, induced by various perturbing forces. These perturbing forces will be treated in the remaining sections of this chapter.

4.3 Deviations in the Earth's Gravity Field

As stated in the introduction of this chapter, the Earth does not have a radially symmetric mass distribution. In order to take this into account, it is convenient to

use a representation for the acceleration using the gradient of the gravity potential U (Montenbruck and Gill, 2005):

$$\ddot{\mathbf{r}} = \nabla U \quad (4.2)$$

For a radially symmetric mass distribution the potential is usually expressed as:

$$U = \frac{GM_{\oplus}}{r} \quad (4.3)$$

An expression valid for an arbitrary mass distribution can be obtained by summing up the contributions created by individual mass elements $dm = \rho(\mathbf{s})d^3\mathbf{s}$:

$$U = G \int \frac{\rho(\mathbf{s})d^3\mathbf{s}}{|\mathbf{r} - \mathbf{s}|} \quad (4.4)$$

In this equation, $\rho(\mathbf{s})$ represents the mass density at some point with position vector \mathbf{s} inside the Earth and $|\mathbf{r} - \mathbf{s}|$ is the distance of the orbiting object from this location (see Figure 4.2).

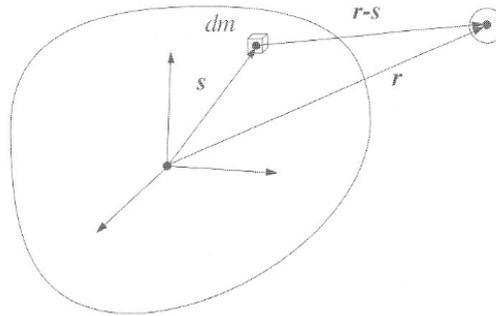


Figure 4.2: The contribution of a small mass element to the geopotential for an arbitrary mass distribution (Montenbruck and Gill, 2005).

In order to evaluate the integral in Equation 4.4, the inverse of the distance $|\mathbf{r} - \mathbf{s}|$ may be expanded using Legendre polynomials. For $r > s$ (with $s = \|\mathbf{s}\|$), which is true for all points \mathbf{r} outside a circumscribing sphere, the following expression holds (Montenbruck and Gill, 2005):

$$\frac{1}{|\mathbf{r} - \mathbf{s}|} = \frac{1}{r} \sum_{n=0}^{\infty} \left(\frac{s}{r}\right)^n P_n(\cos \gamma) \quad \text{with } \cos \gamma = \frac{\mathbf{r} \cdot \mathbf{s}}{rs} \quad (4.5)$$

In this equation, γ is the angle between \mathbf{r} and \mathbf{s} , and $P_n(u)$ is the Legendre polynomial of degree n , defined as

$$P_n(u) = \frac{1}{2^n n!} \frac{d^n}{du^n} (u^2 - 1)^n \quad (4.6)$$

By using the addition theorem of Legendre polynomials, the following expression can be obtained:

$$P_n(\cos \gamma) = \sum_{m=0}^n (2 - \delta_{0,m}) \frac{(n-m)!}{(n+m)!} P_{n,m}(\sin \phi) P_{n,m}(\sin \phi') \cos(m(\lambda - \lambda')) \quad (4.7)$$

in which λ and ϕ are the longitude and geocentric latitude corresponding to the object's position vector \mathbf{r} , respectively. Similarly, λ' and ϕ' indicate the same quantities for a point with position vector \mathbf{s} inside the Earth. Moreover, δ_{0m} represents the Kronecker delta $\delta_{n,m}$ for $n = 0$, which is equal to 1 if $m = 0$ and equal to 0 otherwise. Finally, $P_{n,m}$ indicates the associated Legendre polynomial of degree n and order m , which is defined as

$$P_{n,m}(u) = (1 - u^2)^{m/2} \frac{d^m}{du^m} P_n(u) \quad (4.8)$$

Using the previous expressions, the Earth's gravity potential can be written as:

$$U = \frac{GM_{\oplus}}{r} \sum_{n=0}^{\infty} \sum_{m=0}^n \frac{R_{\oplus}^n}{r^n} P_{n,m}(\sin \phi) (C_{n,m} \cos(m\lambda) + S_{n,m} \sin(m\lambda)) \quad (4.9)$$

with coefficients $C_{n,m}$ and $S_{n,m}$ given by

$$C_{n,m} = \frac{2 - \delta_{0,m}}{M_{\oplus}} \frac{(n-m)!}{(n+m)!} \int \frac{s^n}{R_{\oplus}^n} P_{n,m}(\sin \phi') \cos(m\lambda') \rho(\mathbf{s}) d^3 \mathbf{s} \quad (4.10)$$

$$S_{n,m} = \frac{2 - \delta_{0,m}}{M_{\oplus}} \frac{(n-m)!}{(n+m)!} \int \frac{s^n}{R_{\oplus}^n} P_{n,m}(\sin \phi') \sin(m\lambda') \rho(\mathbf{s}) d^3 \mathbf{s} \quad (4.11)$$

where R_{\oplus} is the equatorial radius of the Earth.

Geopotential coefficients with $m = 0$ are called zonal coefficients and describe the part of the potential that is independent of longitude. As a result of the definitions in Equation 4.11, $S_{n,0}$ is zero for all n . Often, the following notation is used for the other zonal coefficients:

$$J_n = -C_{n,0} \quad (4.12)$$

The remaining geopotential coefficients are called tesseral ($n > m$) and sectorial ($n = m$) coefficients.

Using the geopotential coefficients, the acceleration due to the Earth's gravity field can be written as

$$\ddot{\mathbf{r}} = \nabla \frac{GM_{\oplus}}{r} \sum_{n=0}^{\infty} \sum_{m=0}^n \frac{R_{\oplus}^n}{r^n} P_{n,m}(\sin \phi) (C_{n,m} \cos(m\lambda) + S_{n,m} \sin(m\lambda)) \quad (4.13)$$

Note that Equation 4.13 uses an infinite sum to compute the acceleration. However, it is neither realistic nor feasible to add an infinite number of terms. Therefore, the sum is usually truncated at a certain order m and degree n .

For calculating the Earth's gravity potential at a given location, recurrence relations are used for the evaluation of the Legendre polynomials. Making use of the addition theorems for Legendre polynomials, the computations can be done more efficiently using the following definitions (Montenbruck and Gill, 2005):

$$V_{n,m} = \left(\frac{R_{\oplus}}{r}\right)^{n+1} P_{n,m}(\sin \phi) \cos(m\lambda) \quad (4.14)$$

$$W_{n,m} = \left(\frac{R_{\oplus}}{r}\right)^{n+1} P_{n,m}(\sin \phi) \sin(m\lambda) \quad (4.15)$$

such that the gravity potential can be written as

$$U = \frac{GM_{\oplus}}{R_{\oplus}} \sum_{n=0}^{\infty} \sum_{m=0}^n (C_{n,m}V_{n,m} + S_{n,m}W_{n,m}) \quad (4.16)$$

To begin with, the first zonal terms are computed with

$$V_{0,0} = \frac{R_{\oplus}}{r} \quad (4.17)$$

$$W_{0,0} = 0 \quad (4.18)$$

$$V_{1,0} = \frac{zR_{\oplus}}{r^2} * V(0,0) \quad (4.19)$$

$$W_{1,0} = 0 \quad (4.20)$$

The other zonal terms (with $n \geq 2$) are calculated with the recurrence relations

$$V_{n,0} = \left(\frac{2n-1}{n}\right) \frac{zR_{\oplus}}{r^2} V_{n-1,0} - \left(\frac{n-1}{n}\right) \frac{R_{\oplus}^2}{r^2} V_{n-2,0} \quad (4.21)$$

$$W_{n,0} = 0 \quad (4.22)$$

The sectorial terms, in turn, are computed using

$$V_{m,m} = (2m-1) \left[\frac{xR_{\oplus}}{r^2} V_{m-1,m-1} - \frac{yR_{\oplus}}{r^2} W_{m-1,m-1} \right] \quad (4.23)$$

$$W_{m,m} = (2m-1) \left[\frac{xR_{\oplus}}{r^2} W_{m-1,m-1} - \frac{yR_{\oplus}}{r^2} V_{m-1,m-1} \right] \quad (4.24)$$

Finally, the tesseral terms are calculated with

$$V_{n,m} = \left(\frac{2n-1}{n-m}\right) \frac{zR_{\oplus}}{r^2} V_{n-1,m} - \left(\frac{n+m-1}{n-m}\right) \frac{R_{\oplus}^2}{r^2} V_{n-2,m} \quad (4.25)$$

$$W_{n,m} = \left(\frac{2n-1}{n-m}\right) \frac{zR_{\oplus}}{r^2} W_{n-1,m} - \left(\frac{n+m-1}{n-m}\right) \frac{R_{\oplus}^2}{r^2} W_{n-2,m} \quad (4.26)$$

Once the $V_{n,m}$ and $W_{n,m}$ terms have been computed, the acceleration components can be calculated by summing the partial accelerations for each component:

$$\ddot{x} = \sum_{n,m} \ddot{x}_{n,m} \quad (4.27)$$

$$\ddot{y} = \sum_{n,m} \ddot{y}_{n,m} \quad (4.28)$$

$$\ddot{z} = \sum_{n,m} \ddot{z}_{n,m} \quad (4.29)$$

where the partial accelerations are given by

$$\ddot{x}_{n,m} \stackrel{(m=0)}{=} \frac{GM}{R_{\oplus}^2} [-C_{n,0} V_{n+1,1}] \quad (4.30)$$

$$\begin{aligned} \ddot{x}_{n,m} \stackrel{(m>0)}{=} & \frac{GM}{R_{\oplus}^2} \frac{1}{2} [(-C_{n,m} V_{n+1,m+1} - S_{n,m} W_{n+1,m+1}) \\ & + \frac{(n-m+2)!}{(n-m)!} (C_{n,m} V_{n+1,m-1} + S_{n,m} W_{n+1,m-1})] \end{aligned} \quad (4.31)$$

$$\ddot{y}_{n,m} \stackrel{(m=0)}{=} \frac{GM}{R_{\oplus}^2} [-C_{n,0} W_{n+1,1}] \quad (4.32)$$

$$\begin{aligned} \ddot{y}_{n,m} \stackrel{(m>0)}{=} & \frac{GM}{R_{\oplus}^2} \frac{1}{2} [(-C_{n,m} W_{n+1,m+1} + S_{n,m} V_{n+1,m+1}) \\ & + \frac{(n-m+2)!}{(n-m)!} (-C_{n,m} W_{n+1,m-1} + S_{n,m} V_{n+1,m-1})] \end{aligned} \quad (4.33)$$

$$\ddot{z}_{n,m} = \frac{GM}{R_{\oplus}^2} (n-m+1) [-C_{n,m} V_{n+1,m} - S_{n,m} W_{n+1,m}] \quad (4.34)$$

A full derivation of the equations listed above can be found in Cunningham (1970). It should be noted that the acceleration is given in an Earth-fixed reference frame. In order to obtain the acceleration in an inertial reference frame, the acceleration vector needs to be multiplied with a time-dependent rotation matrix which accounts for the rotation of the Earth.

Gravity Model

The values of the geopotential coefficients are dependent on the gravity model that is used. Over the years, many gravity models have been created for the Earth and improvements are still being made. In the past decade, multiple satellites were launched with the specific purpose of determining the Earth's gravity field with a very high accuracy. Nevertheless, in the simulations only coefficients up to a

relatively low order and degree will be used, as adding more terms will have an increasingly large effect on the required computation time, while the gain in overall simulation accuracy significantly decreases once the degree and order of the coefficients get past a certain point. Therefore, the accuracy of the established JGM-3 gravity field model (Tapley et al., 1996) is deemed more than sufficient for this application. The coefficients of JGM-3, which have been used in the simulation code, can be found in Appendix A.

4.4 Gravitational Attraction of the Sun and the Moon

In addition to the gravitational attraction of the Earth, an object orbiting the Earth experiences gravitational attraction by all other bodies in the universe. By examining the gravitational law (Equation 4.1), it can be concluded that only bodies with a large mass and a relatively small distance from the object yield a gravitational force that is of any significance compared to the gravitational attraction of the Earth. As can be seen in Figure 4.1, the gravitational accelerations due to celestial bodies other than the Earth are much smaller than the gravitational acceleration caused by the Earth itself. Moreover, the perturbing accelerations produced by the Sun and the Moon are at least four orders of magnitude larger than the accelerations produced by the other celestial bodies. Hence, only the perturbing gravitational attraction of the Sun and the Moon will be taken into account. In literature, these perturbing accelerations are occasionally referred to as the *luni-solar perturbations*.

Using Newton's law of gravity, the gravitational attraction of a point mass M representing a celestial body can be stated as follows:

$$\ddot{\mathbf{r}} = GM \frac{\mathbf{s} - \mathbf{r}}{|\mathbf{s} - \mathbf{r}|^3} \quad (4.35)$$

In this equation, \mathbf{r} and \mathbf{s} are the geocentric coordinates of the orbiting object and point mass M , respectively.

It should be noted, however, that $\ddot{\mathbf{r}}$ represents the acceleration with respect to an inertial reference system (Montenbruck and Gill, 2005). In such a coordinate system, the Earth is not at rest, but experiences an acceleration due to M equal to

$$\ddot{\mathbf{r}} = GM \frac{\mathbf{s}}{|\mathbf{s}|^3} \quad (4.36)$$

Taking this into account leads to the following expression for the acceleration of the satellite relative to the center of the Earth, due to the gravitational force of body M (Montenbruck and Gill, 2005):

$$\ddot{\mathbf{r}} = GM \left(\frac{\mathbf{s} - \mathbf{r}}{|\mathbf{s} - \mathbf{r}|^3} - \frac{\mathbf{s}}{|\mathbf{s}|^3} \right) \quad (4.37)$$

Figure 4.3 illustrates this concept. In the top part of the figure, the accelerations with respect to a non-Earth-centered inertial reference frame are shown. Clearly,

the accelerations of both the orbiting object and the Earth are pointed toward the perturbing mass, with the magnitude of the acceleration decreasing as the distance from the perturbing mass increases. The lower part of Figure 4.3 shows the accelerations with respect to an Earth-centered frame. In this case, the orbiting object experiences an acceleration away from the Earth when it is on the line connecting the Earth and the perturbing body, but is pulled toward the Earth whenever it is in a position perpendicular to this line. Since the position of a satellite or debris object is generally described with respect to the Earth, the lower part of Figure 4.3 illustrates most clearly the effect of a perturbing mass on the trajectory of an object orbiting the Earth.

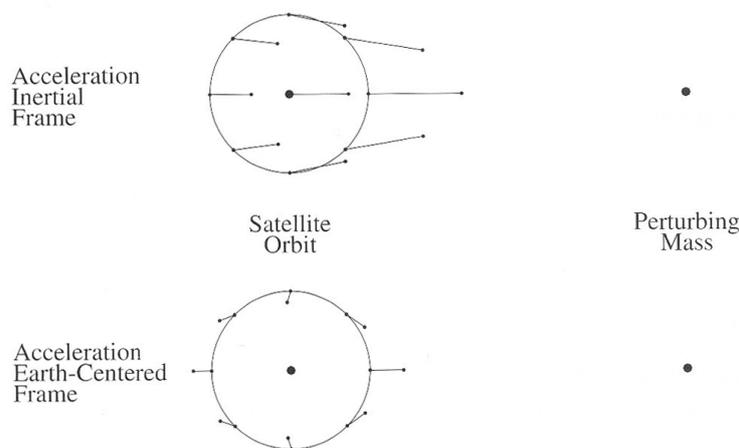


Figure 4.3: The acceleration due to a perturbing mass expressed in two different frames (Montenbruck and Gill, 2005).

Equation 4.37 can be used to calculate the perturbing acceleration due to the gravitational attraction of any celestial body. To use it, however, the position of the perturbing body needs to be known in the Earth-centered reference frame used for describing the motion of the satellite. In the simulation code, series expansions are used for approximating the coordinates of the Sun and the Moon in this reference frame at any point in time. A detailed description of the routines that are used for this is given in Appendix B.

4.5 Solar Radiation Pressure

When radiation emitted by the Sun reaches an object in orbit, the object experiences a small force, resulting from the absorption or reflection of photons. In contrast to gravitational perturbations, the perturbation due to solar radiation pressure depends on the object's surface area and mass. These quantities are often combined to form the area-to-mass ratio (A/m) of the object, which determines to a large extent how much the object will be affected by solar radiation pressure.

The magnitude of solar radiation pressure is dependent on the solar flux ϕ , which is the amount of energy ΔE that passes through an area A in a time interval Δt :

$$\phi = \frac{\Delta E}{A\Delta t} \quad (4.38)$$

The impulse carried by a single photon of energy E_ν is given by

$$p_\nu = \frac{E_\nu}{c} \quad (4.39)$$

in which c is the speed of light (Montenbruck and Gill, 2005). Hence, the total change in impulse of an absorbing body illuminated by the Sun in a time interval Δt is equal to

$$\Delta p = \frac{\Delta E}{c} = \frac{\phi}{c} A \Delta t \quad (4.40)$$

Therefore, the object experiences a force of

$$F = \frac{\Delta p}{\Delta t} = \frac{\phi}{c} A \quad (4.41)$$

resulting in a pressure of

$$P_r = \frac{\phi}{c} \quad (4.42)$$

The solar flux at a distance of 1 AU from the Sun amounts to

$$\phi \approx 1367 \text{ Wm}^{-2} \quad (4.43)$$

(McCarthy, 1996). Therefore, the solar radiation pressure near the Earth is approximately

$$P_r \approx 4.56 \cdot 10^{-6} \text{ Nm}^{-2} \quad (4.44)$$

Equation 4.44 is valid under the assumption that the object's surface absorbs all incoming photons and is perpendicular to the incoming radiation. In reality, the object's surface will have a reflection coefficient ρ with a value between 0 (complete absorption) and 1 (complete reflection). The force due to solar radiation pressure is shown for the two extreme cases in Figure 4.4. In this figure, the unit normal vector \mathbf{n} gives the orientation of the surface A . Furthermore, angle θ is the angle between this normal vector and the unit vector \mathbf{e}_\odot pointing from the object to the Sun.

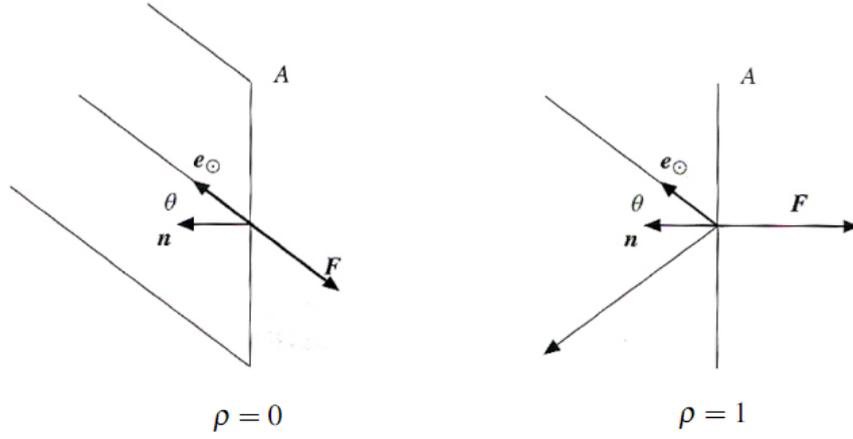


Figure 4.4: The force due to solar radiation pressure for the cases of complete absorption ($\rho = 0$) and complete (specular) reflection ($\rho = 1$) (Montenbruck and Gill, 2005).

The following expression can be used to calculate the force due to radiation pressure, taking into account a non-zero reflection coefficient (Montenbruck and Gill, 2005):

$$\mathbf{F} = -P_r \cos(\theta) A [(1 - \rho) \mathbf{e}_{\odot} + 2\rho \cos(\theta) \mathbf{n}] \quad (4.45)$$

In practice, the distance between the Earth and the Sun is not constant over the year, but varies between $147 \cdot 10^6$ km and $152 \cdot 10^6$ km. Accordingly, the solar radiation pressure also varies over the year. To account for this dependence, the following equation for the acceleration due to solar radiation pressure can be applied:

$$\ddot{\mathbf{r}} = -P_r \frac{(1\text{AU})^2 A}{r_{\odot}^2} \frac{1}{m} \cos(\theta) [(1 - \rho) \mathbf{e}_{\odot} + 2\rho \cos(\theta) \mathbf{n}] \quad (4.46)$$

in which r_{\odot} indicates the instantaneous distance of the object from the Sun (expressed in the same units as the term 1AU in the numerator).

In many cases, it can be assumed that the surface normal \mathbf{n} points to the Sun. In that case, Equation 4.46 can be simplified to

$$\ddot{\mathbf{r}} = -P_r C_R \frac{A}{m} \frac{\mathbf{r}_{\odot}}{r_{\odot}^3} (1\text{AU})^2 \quad (4.47)$$

where \mathbf{r}_{\odot} is the vector from the object to the Sun

$$\mathbf{r}_{\odot} = \mathbf{r} - \mathbf{r}_{Sun} \quad (4.48)$$

and C_R is the radiation pressure coefficient, given by

$$C_R = 1 + \rho \quad (4.49)$$

Eclipses

Eclipses play an important role in modeling the effects of solar radiation pressure. As long as the Sun is completely behind the Earth from the viewpoint of the orbiting object, no direct solar radiation will reach the object. Also, when the Sun is partially obscured, the solar radiation pressure will be reduced compared to the situation of full solar illumination.

Eclipses can be taken into account by modifying Equation 4.47 slightly:

$$\ddot{\mathbf{r}} = -v P_r C_R \frac{A}{m} \frac{r_\odot}{r_\odot^3} (1\text{AU})^2 \quad (4.50)$$

In Equation 4.50, v is a shadow function:

$$v = \begin{cases} 0 & \text{if the orbiting object is in umbra (full eclipse)} \\ \text{between 0 and 1} & \text{if the orbiting object is in penumbra (partial eclipse)} \\ 1 & \text{if the orbiting object is in sunlight} \end{cases}$$

Different shadow models can be used for eclipses. The two most common options are cylindrical and conical shadow models (see Figure 4.5). Although a cylindrical shadow model is easier to apply as it ignores penumbra, it lacks some of the accuracy of a conical shadow model.

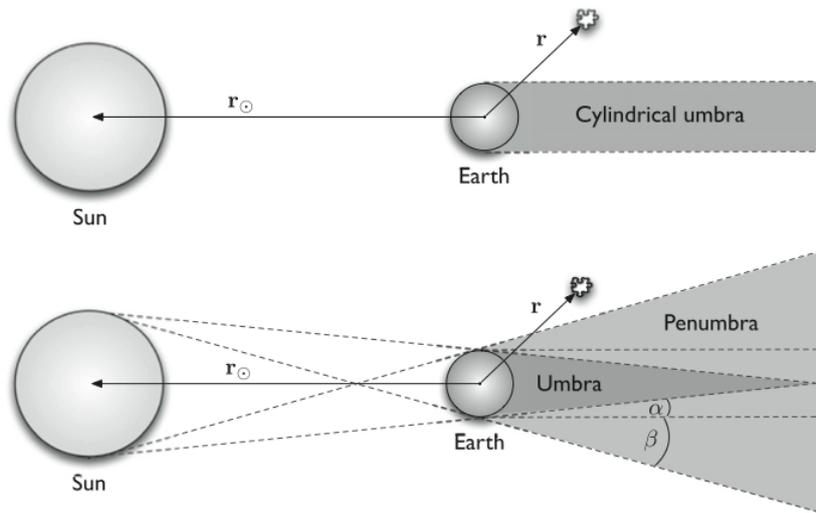


Figure 4.5: Cylindrical and conical shadow models (Hubaux et al., 2012).

Regarding Figure 4.5, it should be mentioned that the angles relating to the penumbra are highly exaggerated¹ and that in practice, the conical shadow model quite

¹In fact, $\alpha \approx 0.26^\circ$ and $\beta \approx 0.27^\circ$.

closely resembles the cylindrical shadow model. Additionally, the orbits that are to be simulated for this thesis project lie in the GEO and GNSS regions of space, and accordingly, eclipse conditions only occur during relatively small parts of the orbits. Therefore, it is decided to keep the eclipse modeling relatively simple and use a cylindrical shadow model for the simulations. This implies that partial eclipses are not taken into account and hence, the shadow function ν has a value of either 0 or 1.

To determine whether or not an orbiting object is in eclipse, vector calculus is used. First, the unit vector pointing to the Sun is determined:

$$\mathbf{e}_{Sun} = \frac{\mathbf{r}_{Sun}}{|\mathbf{r}_{Sun}|} \quad (4.51)$$

Then, the projection of the object's position vector on the position vector of the Sun is computed with

$$\begin{aligned} s &= \mathbf{r} \cdot \mathbf{e}_{Sun} \\ &= |\mathbf{r}| |\mathbf{e}_{Sun}| \cos(\theta) \\ &= r \cos(\theta) \end{aligned} \quad (4.52)$$

The condition

$$s > 0 \quad (4.53)$$

corresponds to

$$0 < \theta < 90^\circ$$

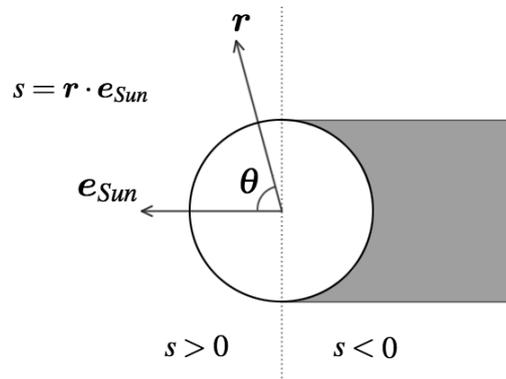
which means that the orbiting object is on the same side of the Earth as the Sun. In this case, the object is not in eclipse. This condition is illustrated in Figure 4.6(a).

The other condition that is used for checking whether the object is not in eclipse is given by

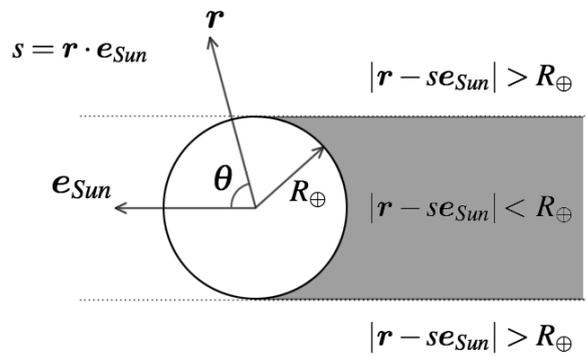
$$|\mathbf{r} - s\mathbf{e}_{Sun}| > R_\oplus \quad (4.54)$$

In case this condition is fulfilled, the object is outside a cylinder with the same radius as the Earth, which fully contains the shadow of the Earth, but also extends in the opposite direction (see Figure 4.6(b)).

Note that while fulfillment of either condition (4.53) or (4.54) is sufficient to prove that the object is not in eclipse, both conditions need to be false in order to deduce that the object is in eclipse. Hence, the simulation code checks for eclipses in the following way: if either condition (4.53) or (4.54) is satisfied, the object is assumed to be fully illuminated by the Sun and the illumination factor ν will get a value of 1. If neither condition is fulfilled, the object is determined to be in eclipse and ν will be set to 0.



(a) An illustration of the first condition. The object is not in eclipse if $s > 0$.



(b) An illustration of the second condition. The object is not in eclipse if $|r - se_{Sun}| > R_{\oplus}$.

Figure 4.6: A graphical representation of the conditions that are used to determine whether the orbiting object is in eclipse.

Chapter 5

Orbit Formulation

As a result of the force model treated in the previous chapter, the equation of motion has the following form:

$$\ddot{\mathbf{r}} = \ddot{\mathbf{r}}_{Main} + \ddot{\mathbf{r}}_{Geopotential} + \ddot{\mathbf{r}}_{Sun} + \ddot{\mathbf{r}}_{Moon} + \ddot{\mathbf{r}}_{SRP} \quad (5.1)$$

The trajectory of the space debris object to be modeled is the solution to this second-order differential equation. Before going into the topic of integration methods, it is worthwhile to consider in what ways the equation of motion can be formulated, as some ways are more practical and efficient than others.

This chapter starts with a short reminder of the basic laws of astrodynamics and will continue with a treatment of three coordinate systems that are convenient to use in astrodynamics. Finally, different methods of modeling perturbed orbits will be discussed, whose formulations can have a clear effect on the efficiency of long-term simulations.

5.1 Basic Laws of Astrodynamics

This section provides a brief review of the laws on which astrodynamics is based. In addition, an overview of some of the most fundamental equations in astrodynamics is given in Appendix C. It is assumed that the reader is familiar with astrodynamics, and hence, this section and Appendix C mainly serve to refresh the memory on some of the concepts and equations.

Newton's three laws of motion, formulated in his Principia (1687), read as:

1. Every particle continues in its state of rest or uniform motion in a straight line relative to an inertial reference frame, unless it is compelled to change that state by forces acting upon it.
2. The time rate of change of linear momentum of a particle relative to an inertial reference frame is proportional to the resultant of all forces acting upon that particle and is collinear with and in the direction of the resultant force.

3. If two particles exert forces on each other, these forces are equal in magnitude and opposite in direction.

Newton's law of gravity, governing the attraction between two point masses, is stated as:

Two particles attract each other with a force directly proportional to their masses and inversely proportional to the square of the distance between them.

Kepler's laws of planetary motion read as:

1. The orbit of a planet is an ellipse, of which the Sun is located in one of the foci.
2. The radius vector of the planet sweeps out equal areas in equal intervals of time.
3. The ratio between the square of the period and the cube of the major axis of an elliptical orbit is equal for all planets.

The laws listed above translate into (the equations for) unperturbed Keplerian orbits.

5.2 Coordinate Systems

There are multiple coordinate systems that can be used for orbit modeling. The choice of coordinate system can depend on multiple factors, such as ease of implementation, computational efficiency and geometric interpretation. In the next sections, three different coordinate systems that are convenient for orbital mechanics will be discussed.

5.2.1 Cartesian Coordinates

A Cartesian coordinate system consists of three axes that are mutually orthogonal. In most cases, these axes are called the x -, y -, and z -axis, respectively. In the description of objects orbiting the Earth, it is convenient to choose an equatorial coordinate system (see Figure 5.1), which is aligned with the Earth's rotation axis and equator. The reference frame is chosen such that the origin of the system is located at the center of the Earth, the x - y plane coincides with the equatorial plane and the z -axis points to the north pole. Furthermore, the x -axis is aligned to the Vernal Equinox (Υ), corresponding to the intersection of the equatorial plane with the Earth's orbital plane.

The position and velocity of an object in orbit around the Earth can now be specified using a set of six Cartesian coordinates: $[x, y, z, \dot{x}, \dot{y}, \dot{z}]$, where the last three parameters are time derivatives of the position coordinates. The combined

vector of these position coordinates and velocity components is often called a *state vector*, as it fully describes the state of a moving particle in 3D space.

An advantage of using Cartesian coordinates for orbit modeling is that it leads to relatively simple mathematical expressions, from which time derivatives can easily be obtained. Disadvantages are that the coordinates change relatively quickly, hence requiring many time steps when integrating, and the fact that it is hard to interpret what the orbit looks like from a set of position and velocity components.

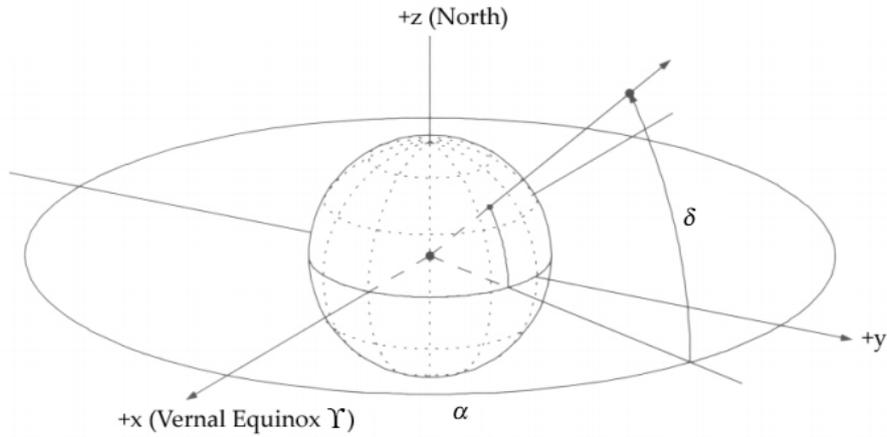


Figure 5.1: Definition of the equatorial coordinate system (Montenbruck and Gill, 2005).

5.2.2 Orbital Elements

When using Cartesian or polar coordinates, most of the parameters used for describing the orbit change quite drastically over time. However, it is also possible to choose certain parameters that remain constant over time for an unperturbed Keplerian orbit. This is the case for the orbital elements $[a, e, i, \Omega, \omega, \tau]$, also called the Keplerian elements.

The in-plane elements a and e are indicated in Figure 5.2. a is the semi-major of the ellipse¹, being equal to half the length of the longest axis of the ellipse. The eccentricity, e , determines the shape of the ellipse and is defined as

$$e = \frac{r_a - r_p}{r_a + r_p} \quad (5.2)$$

where r_a and r_p indicate the radii of apocenter and pericenter, respectively.

¹The orbital elements are also defined for parabolic ($e = 1$) and hyperbolic ($e > 1$) orbits, but these definitions will not be treated in this section.

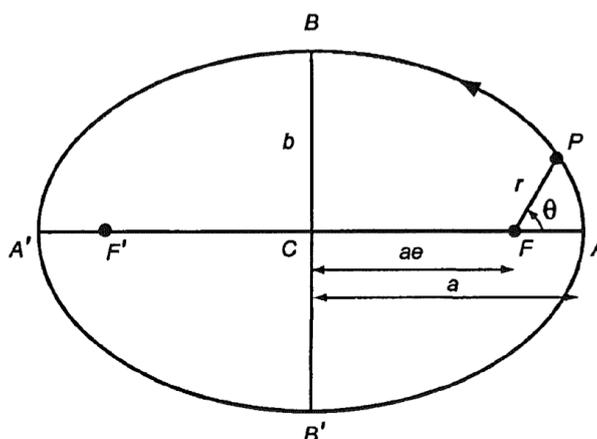


Figure 5.2: Definition of the in-plane orbital elements (Wakker, 2010).

The definition of the orbital elements i , Ω and ω is shown in Figure 5.3. i is the inclination, which is the angle between the orbital plane and the reference plane. Furthermore, Ω is called the right ascension of the ascending node (for an equatorial reference plane) and indicates the angle between the reference direction (Y) and the ascending node², measured in the reference plane. Next, the argument of pericenter ω is the angle from the ascending node to the pericenter, defining the orientation of the ellipse within the orbital plane. Finally, the time of pericenter passage τ is required to link time to the position in orbit. In order to have a parameter that is easier to interpret geometrically, the parameter τ is often replaced by the true anomaly θ , which represents the angle in the plane of the ellipse between the pericenter and the position of the orbiting object at any point in time. It should be emphasized, though, that θ does not remain constant over time for an unperturbed orbit, whereas τ does.

One of the main advantages of using orbital elements is that their physical meaning is much clearer than that of Cartesian coordinates. Additionally, the orbital elements (with the exception of θ) remain constant for an unperturbed Keplerian orbit and change only slowly for a perturbed orbit. This allows a smoother integration process than with Cartesian coordinates, which change quite rapidly over time. Hence, a larger integration step size can be applied when using orbital elements. However, the expressions that are involved are more complicated than the ones for Cartesian components, and do contain another disadvantage, which will be treated in the next section.

²For an object orbiting the Earth, the ascending node is the location where the object crosses the equatorial reference plane, moving from the Southern hemisphere to the Northern hemisphere.

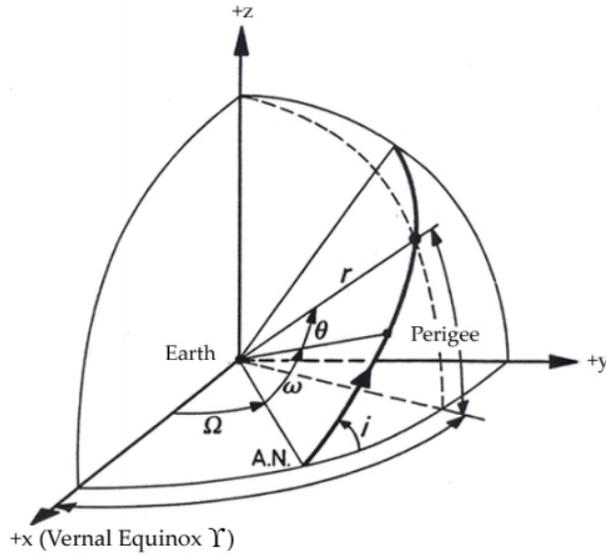


Figure 5.3: Definition of the orbital elements that define the orientation of the orbit in three-dimensional space (Wakker, 2010).

5.2.3 Modified Equinoctial Elements

Though the classical orbital elements have convenient properties, they do unfortunately have an important shortcoming. As will become clear in Section 5.3.3, some of the equations that are used for propagating the orbital elements become singular if either the inclination or eccentricity approaches zero. Since geostationary orbits ideally have both an inclination and an eccentricity of zero, the standard equations for advancing the orbital elements cannot be used in the simulation code, due to the existence of these singularities.

Therefore, another set of elements, called the modified equinoctial elements (MEE) is used. They are formed by combining the classical orbital elements in such a way that the singularities that arise in the equations of motion disappear, and hence, the new equations of motion can be used for all types of orbits³. The modified equinoctial elements are defined as (Walker et al., 1985):

$$p = a(1 - e^2) \quad (5.3)$$

$$f = e \cos(\bar{\omega}) \quad (5.4)$$

$$g = e \sin(\bar{\omega}) \quad (5.5)$$

$$h = \tan\left(\frac{i}{2}\right) \cos(\Omega) \quad (5.6)$$

³Strictly speaking, the new equations of motion do have a singularity at $i = \pi$, but this can be handled by an appropriate re-definition.

$$k = \tan\left(\frac{i}{2}\right) \sin(\Omega) \quad (5.7)$$

$$L = \bar{\omega} + \theta \quad (5.8)$$

where

$$\bar{\omega} = \omega + \Omega \quad (5.9)$$

Although the modified equinoctial elements are not as easy to interpret geometrically as the classical orbital elements, the advantage of having equations of motion that can also be used for the numerical propagation of near-circular orbits and orbits that lie in the equatorial plane outweighs this disadvantage, making the use of modified equinoctial elements the better choice for this thesis project.

5.3 Modeling Perturbed Orbits

As was discussed in Chapter 4, various perturbing forces change the orbits of satellites and other objects orbiting the Earth. To take these forces into account, the equation of motion of an orbiting object relative to a non-rotating geocentric equatorial reference frame can be written as (Wakker, 2010):

$$\frac{d^2\mathbf{r}}{dt^2} + \frac{\mu}{r^3}\mathbf{r} = -\nabla R + \mathbf{f} \quad (5.10)$$

In this equation, R is the perturbing or disturbing potential, which describes all perturbing accelerations that can be expressed by a potential function. \mathbf{f} , in turn, represents all perturbing accelerations that cannot be written as the gradient of a scalar function.

In general, Equation 5.10 cannot be solved analytically. Numerical integration techniques or approximative analytical methods are often used to find its solution instead. Methods depending on numerical integration techniques are called *special perturbation methods*, generating just one particular trajectory for an orbiting object, given its initial conditions. Methods employing approximative analytical methods, on the other hand, are called *general perturbation methods* and yield approximative solutions that are applicable for all orbiting objects and for all initial conditions.

Unfortunately, the use of general perturbation methods involves a tremendous amount of analytical labor (Wakker, 2010), especially if many perturbing forces are to be taken into account. Additionally, the achievable accuracy of these methods is significantly lower than that of special perturbation methods. For these reasons, general perturbation methods will be considered no further. Instead, the next sections will focus on the most important special perturbation techniques that exist in orbital mechanics.

5.3.1 Method of Cowell

The most straightforward method to compute perturbed satellite orbits is the method of Cowell (Wakker, 2010). In this method, the equation of motion is written in the form

$$\frac{d^2 \mathbf{r}}{dt^2} = \mathbf{f}_t \quad (5.11)$$

in which \mathbf{f}_t represents the total acceleration. Following from Equation 5.10, this total acceleration is given by

$$\mathbf{f}_t = -\frac{\mu}{r^3} \mathbf{r} - \nabla R + \mathbf{f} \quad (5.12)$$

Equation 5.11 is then integrated directly using a numerical integration method. More information about numerical integrators will be given in Chapters 6 and 7.

The most important advantages of Cowell's method are that it is easy to program due to its simple equations and that it is always applicable. No assumptions or substitutions have been made that restrict its use to specific situations.

An important drawback of the method, however, is that it makes no use of the fact that ∇R and \mathbf{f} are small, perturbing accelerations. The integration has to account for the effects of both the central Newtonian gravity field and the perturbations produced by the various perturbing forces. Hence, small integration steps are required in the numerical integration process. This leads to a relatively long computation time and a steadily growing integration error due to the accumulation of round-off errors.

5.3.2 Method of Encke

In contrast to the method of Cowell, the method of Encke makes use of a *reference orbit* and only the deviation with respect to that orbit is integrated numerically. The deviation of the actual trajectory with respect to the reference orbit at a time t is expressed as

$$\Delta \mathbf{r} = \mathbf{r} - \boldsymbol{\rho} \quad (5.13)$$

where \mathbf{r} is the actual position of the orbiting object and $\boldsymbol{\rho}$ denotes the position of the orbiting object in case it would follow the unperturbed reference orbit.

Setting up the equation of motion for $\Delta \mathbf{r}$ eventually leads to the following expression (Wakker, 2010):

$$\frac{d^2 \Delta \mathbf{r}}{dt^2} = \frac{\mu}{\rho^3} [(\boldsymbol{\rho} + \Delta \mathbf{r})f(q) - \Delta \mathbf{r}] - \nabla R + \mathbf{f} \quad (5.14)$$

where $f(q)$ can either be computed using a binomial series expansion:

$$f(q) = 3q \left(1 - \frac{5}{2}q + \frac{35}{6}q^2 - \frac{105}{8}q^3 + \dots \right) \quad (5.15)$$

or using a closed form:

$$f(q) = \frac{2q}{1+2q} \left[1 + \frac{1}{1+2q+\sqrt{1+2q}} \right] \quad (5.16)$$

Since in Encke's method only the perturbing accelerations are integrated numerically to obtain the deviation with respect to the reference orbit, a larger integration step size can generally be chosen than in Cowell's method. Yet, at each integration step Encke's method requires more computation time (Wakker, 2010). In general, however, Encke's method yields a more efficient computation process than Cowell's method for small but strongly varying perturbing forces.

If the perturbations accumulate, $\Delta \mathbf{r}$ may eventually become quite large. In that case, the reference orbit should be *rectified*, i.e. a new reference orbit should be chosen with $\Delta \mathbf{r} = 0$ and $\frac{d\Delta \mathbf{r}}{dt} = 0$ at the instant of rectification. This need for rectification is a drawback of Encke's method, especially if the orbit has to be rectified often during the propagation.

5.3.3 Method of Variation of Parameters

A third method to compute perturbed orbits is the method of variation of parameters, also called the method of variation of orbital elements. This method relies on the concept of *osculating Keplerian orbits*. An osculating orbit is the orbit that an orbiting object would follow if from that moment onward no perturbations would act on the object anymore (Wakker, 2010). This orbit touches the true (perturbed) orbit of the object at that particular moment in time, t , and the position and velocity are the same for both orbits in the point of contact. Since perturbing forces act on the object, another osculating orbit will be found at another instant in time, $t + \Delta t$.

In the method of variation of parameters, a perturbed orbit is viewed as a sequence of small parts of a series of osculating Keplerian orbits, of which the orbital elements vary continuously. This yields a set of first-order differential equations that describe the variation of the osculating orbital elements with time. By integrating these equations, the osculating orbital elements can be calculated for any moment in time. From these orbital elements, the position and velocity in the perturbed orbit at that time can be determined using the transformations valid for Keplerian orbits.

Since only the differences between the perturbed orbit and a Keplerian reference orbit are integrated, the method of variation of parameters is in fact very similar to the method of Encke. The main advantage of using the method of variation of parameters, however, is that the variation of orbital elements yields a much clearer picture of the geometric characteristics of the orbital perturbations than the variation of Cartesian position and velocity components. Furthermore, the method is used quite often to obtain approximative analytical solutions of the differential equations. A disadvantage of the method of variation of parameters is that the expressions that are used are quite complex.

Lagrange's Planetary Equations

One set of six simultaneous first-order differential equations that express the effects of perturbing forces on each of the orbital elements is given by *Lagrange's planetary equations* (Wakker, 2010):

$$\frac{da}{dt} = -2 \frac{a^2}{\mu} \frac{\partial \tilde{R}}{\partial \tau} \quad (5.17)$$

$$\frac{de}{dt} = -\frac{a(1-e^2)}{\mu e} \frac{\partial \tilde{R}}{\partial \tau} - \frac{1}{e} \sqrt{\frac{1-e^2}{\mu a}} \frac{\partial \tilde{R}}{\partial \omega} \quad (5.18)$$

$$\frac{di}{dt} = \frac{\cot i}{\sqrt{\mu a(1-e^2)}} \frac{\partial \tilde{R}}{\partial \omega} - \frac{1}{\sqrt{\mu a(1-e^2)} \sin i} \frac{\partial \tilde{R}}{\partial \Omega} \quad (5.19)$$

$$\frac{d\omega}{dt} = \frac{1}{e} \sqrt{\frac{1-e^2}{\mu a}} \frac{\partial \tilde{R}}{\partial e} - \frac{\cot i}{\sqrt{\mu a(1-e^2)}} \frac{\partial \tilde{R}}{\partial i} \quad (5.20)$$

$$\frac{d\Omega}{dt} = \frac{1}{\sqrt{\mu a(1-e^2)} \sin i} \frac{\partial \tilde{R}}{\partial i} \quad (5.21)$$

$$\frac{d\tau}{dt} = 2 \frac{a^2}{\mu} \frac{\partial \tilde{R}}{\partial a} + \frac{a(1-e^2)}{\mu e} \frac{\partial \tilde{R}}{\partial e} \quad (5.22)$$

In the equations listed above, a force function notation has been used, with $\tilde{R} = -R$, where R is the perturbing potential. It should be noted that Lagrange's planetary equations are only valid for perturbing forces that can be expressed through a perturbing potential.

Moreover, if $e = 0$ or $i = 0$, singularities will occur in the equations. Problems also occur if e and i are not equal to zero, but are very small in magnitude, making it impossible to find an analytical solution to the equations. These problems have nothing to do with the method in itself, but are a consequence of using orbital elements as the set of parameters that describe the orbit. If the orbit is likely to encounter these problems, other elements can be used that produce no singularities for the orbit that is considered, such as the modified equinoctial elements introduced in Section 5.2.3.

Gauss' Form of Lagrange's Planetary Equations

A limitation of Lagrange's planetary equations is that they can only be used for perturbations that can be described by a perturbing potential. This problem can be circumvented by deriving the equations in a slightly different way. The total perturbing acceleration is now decomposed into three orthogonal components: a radial component f_S , a component in the orbital plane perpendicular to the radius vector f_N , pointing in the direction of motion, and a component perpendicular to the orbital plane f_W , pointing in the direction of the angular momentum vector (see Figure 5.4).

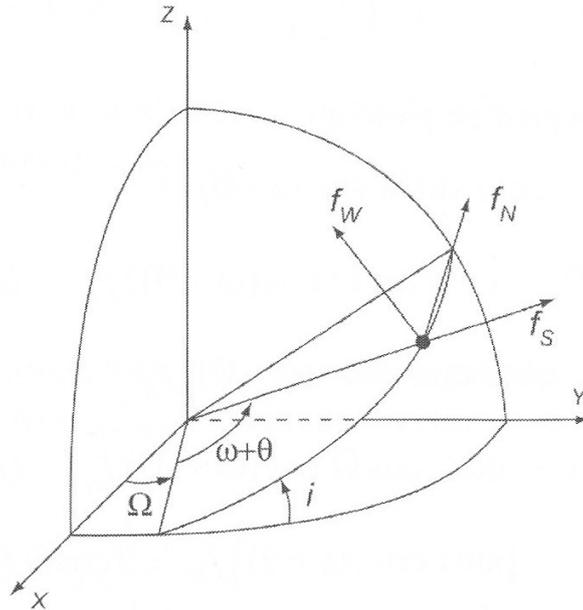


Figure 5.4: The geometry of the $f_S f_N f_W$ acceleration frame relative to the orbit and the equatorial frame (Wakker, 2010).

The conversion from Cartesian components to these components can be done by means of the following transformation:

$$\ddot{\mathbf{r}}_{SNW} = \mathbf{R} \cdot \ddot{\mathbf{r}}_{xyz} \quad (5.23)$$

where the rotation matrix \mathbf{R} is given by

$$\mathbf{R} = \begin{pmatrix} c_{\Omega}c_{(\omega+\theta)} - c_i s_{\Omega}s_{(\omega+\theta)} & s_{\Omega}c_{(\omega+\theta)} + c_i c_{\Omega}s_{(\omega+\theta)} & s_i s_{(\omega+\theta)} \\ -(c_{\Omega}s_{(\omega+\theta)} + c_i s_{\Omega}c_{(\omega+\theta)}) & -(s_{\Omega}s_{(\omega+\theta)} - c_i c_{\Omega}c_{(\omega+\theta)}) & s_i c_{(\omega+\theta)} \\ s_i s_{\Omega} & -s_i c_{\Omega} & c_i \end{pmatrix}$$

in which the following abbreviations have been used for the sines and cosines:

$$s_{\alpha} = \sin \alpha$$

$$c_{\alpha} = \cos \alpha$$

Using the radial (f_S), transverse (f_N) and normal (f_W) components of the acceleration, *Gauss' form of Lagrange's planetary equations* (also simply called *Gauss' equations*) can be obtained, being valid for all perturbing accelerations, regardless of whether these can be derived from a potential function or not (Wakker, 2010):

$$\frac{da}{dt} = 2 \frac{a^2}{\sqrt{\mu p}} \left[f_S e \sin \theta + f_N \frac{p}{r} \right] \quad (5.24)$$

$$\frac{de}{dt} = \sqrt{\frac{p}{\mu}} [f_S \sin \theta + f_N (\cos E + \cos \theta)] \quad (5.25)$$

$$\frac{di}{dt} = f_W \frac{r}{\sqrt{\mu p}} \cos u \quad (5.26)$$

$$\frac{d\omega}{dt} = -\sqrt{\frac{p}{\mu}} \left[f_W \frac{r}{p} \cot i \sin u + \frac{1}{e} \left\{ f_S \cos \theta - f_N \left(1 + \frac{r}{p} \right) \sin \theta \right\} \right] \quad (5.27)$$

$$\frac{d\Omega}{dt} = f_W \frac{r}{\sqrt{\mu p} \sin i} \sin u \quad (5.28)$$

$$\frac{dM}{dt} = n - f_S \left[\frac{2r}{\sqrt{\mu a}} - \frac{1-e^2}{e} \sqrt{\frac{a}{\mu}} \cos \theta \right] - f_N \frac{1-e^2}{e} \sqrt{\frac{a}{\mu}} \left(1 + \frac{r}{p} \right) \sin \theta \quad (5.29)$$

Note that in Equation 5.22, use is made of the time of pericenter passage τ , whereas in Equation 5.29 the mean anomaly M is used. Hence, Equations 5.22 and 5.29 are different, also in case of an unperturbed orbit (for which τ is constant and M varies according to $M = n(t - t_0)$).

Also note that the singularities arising if $e = 0$ or $i = 0$ are still present in the equations for the time derivatives of ω , Ω and M .

Gauss' Equations using Modified Equinoctial Elements

To prevent the singularities associated with the variation of parameters equations expressed using the orbital elements, the equations can be rewritten using modified equinoctial elements. Since the force model necessitates the use of Gauss' form of Lagrange's planetary equations, the modified equations will only be shown here for that form of the planetary equations. Following the work of Roth (1985), Gauss' equations using modified equinoctial elements are as follows:

$$\frac{dp}{dt} = 2 \sqrt{\frac{p^3}{\mu}} \frac{1}{W} f_N \quad (5.30)$$

$$\frac{df}{dt} = \sqrt{\frac{p}{\mu}} \frac{1}{W} [W \sin(L) f_S + A(L) f_N - g(h \sin(L) - k \cos(L)) f_W] \quad (5.31)$$

$$\frac{dg}{dt} = \sqrt{\frac{p}{\mu}} \frac{1}{W} [-W \cos(L) f_S + B(L) f_N + f(h \sin(L) - k \cos(L)) f_W] \quad (5.32)$$

$$\frac{dh}{dt} = \frac{1}{2} \sqrt{\frac{p}{\mu}} \frac{X}{W} \cos(L) f_W \quad (5.33)$$

$$\frac{dk}{dt} = \frac{1}{2} \sqrt{\frac{p}{\mu}} \frac{X}{W} \sin(L) f_W \quad (5.34)$$

$$\frac{dL}{dt} = \sqrt{\frac{\mu}{p^3}} W^2 + \sqrt{\frac{p}{\mu}} \frac{1}{W} (h \sin(L) - k \cos(L)) f_W \quad (5.35)$$

where the following abbreviations have been used:

$$s = (1 - f^2 - g^2)^{1/2} \quad (5.36)$$

$$X = 1 + h^2 + k^2 \quad (5.37)$$

$$W = 1 + f \cos(L) + g \sin(L) \quad (5.38)$$

$$A(L) = f + \cos(L)(1 + W) \quad (5.39)$$

$$B(L) = g + \sin(L)(1 + W) \quad (5.40)$$

5.4 Choice of Propagation Methods

The propagation methods discussed in Section 5.3 all have their respective advantages and disadvantages. Furthermore, the choice of the propagation method to use is linked to the choice of the coordinate system to use for the integration. The methods of Cowell and Encke are generally used in combination with Cartesian coordinates, whereas the method of variation of parameters relies on the use of orbital elements or similar parameters, like modified equinoctial elements.

Of the three propagation methods, the method of Cowell is the easiest one to implement, as it essentially implies a direct integration of the equations of motion in Cartesian coordinates. In addition to its ease of implementation, an important advantage of the method of Cowell is that it always works, given that a small enough integration step size has been chosen (Wakker, 2010). This makes it a convenient method to use in a general simulation tool, such as the one to be developed for this thesis project. Therefore, the Cowell formulation will be included as one of the options in the simulation code.

In the method of Encke, only the perturbations are integrated numerically, which allows a larger step size to be used than with the method of Cowell. A drawback, however, is that the reference orbit needs to be rectified if the deviations from it become too large. Since the orbits of objects with large area-to-mass ratios will also be modeled in this project, the simulated orbits can develop to be drastically different from the original orbits (see Sections 3.1.2 and 3.2.2), making it necessary to rectify the reference orbit fairly often. As a result, part of the advantage of using Encke's method is lost. Additionally, the method of Encke is fairly similar to the Wisdom-Holman splitting method for symplectic integrators,

which will be treated in Chapter 7. This particular method also does not have the drawback of needing periodic rectification of the reference orbit. Because of this, and in order to keep the number of methods in the simulation code manageable, the method of Encke will not be implemented. Instead, the Wisdom-Holman splitting method for symplectic integrators will be included in the simulation code.

The method of variation of parameters is also quite similar to the method of Encke, in the sense that only the effect of the perturbing forces is integrated numerically. However, in this case the direct effect on the orbital elements or similar parameters is integrated, instead of the effect on the Cartesian state vector. Because the orbital elements only show small variations under the effect of perturbing forces, this approach allows for a relatively stable integration process.

Since non-conservative forces are included in the force model, Gauss' form of Lagrange's planetary equations needs to be used instead of the classical set of equations. An important drawback of either set of equations is that they produce singularities for eccentricities or inclinations equal to zero, in case orbital elements are used for the propagation. As geostationary orbits are both circular and located in the equatorial plane, a different set of parameters needs to be used to prevent numerical problems in the simulations.

Because of its favorable properties, the method of variation of parameters will be used in the simulation code. In order to deal with the aspects mentioned above, the specific implementation will use Gauss' form of Lagrange's planetary equations combined with modified equinoctial elements.

Chapter 6

Traditional Integration Methods

The problem of predicting the trajectory of a space debris object from specified starting conditions is essentially an *initial value problem*, with the equation of motion being the central differential equation that needs to be solved. This equation of motion results from the force model (described in Chapter 4) and has the form

$$\ddot{\mathbf{r}} = \mathbf{a}(t, \mathbf{r}, \dot{\mathbf{r}}) \quad (6.1)$$

This second-order differential equation needs to be solved for a specified set of initial conditions. However, when perturbing forces are taken into account, the equation of motion cannot be solved analytically.

Hence, it is necessary to solve the equation numerically¹. The algorithms that are used for this purpose are called numerical integrators.

In this chapter, a number of commonly used integration methods will be described. These will be referred to as *traditional integration methods*. The next chapter will go into detail on a relatively new set of integration methods called *symplectic integration methods*.

This chapter on numerical integration methods is not meant to be exhaustive. There are many textbooks available on the subject, and a full discussion of the various methods that are in existence would require a book in itself. Instead, this chapter treats a number of concepts that are important in understanding the basics of how integration methods work and focuses primarily on the integration methods that have been implemented in the simulation code.

¹ Another option would be to approximate the solution using analytical approximations. However, analytical approximations are not deemed accurate enough for modeling space debris motion over periods of several centuries, and therefore these methods will not be considered any further in this report.

6.1 Introduction

The way in which numerical integration works is perhaps most clearly illustrated by means of an example. For visualization purposes, it is most convenient to introduce some of the concepts for a one-dimensional case. Afterwards, the notions can be extended to a three-dimensional situation quite easily.

Consider the following initial value problem:

$$\begin{aligned} \dot{y}(t) &= y(t)(2-t)t + t - 1 \\ y(0) &= 1 \end{aligned} \tag{6.2}$$

This problem does not have a closed-form solution. To approximate the solution, numerical integration methods can be used. In order to have a reference of the actual solution, a highly accurate numerical solution of the problem is shown in Figure 6.1.

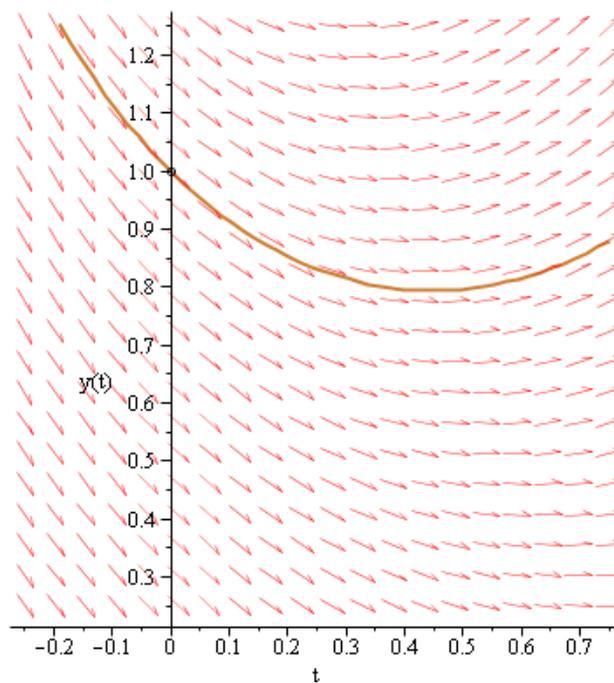


Figure 6.1: The solution to the initial value problem of Equation 6.2 (Harder, 2012).

Euler's Method

The simplest integration method is Euler's method. In this method, one sample of the time derivative is used to approximate the slope of the function:

$$K_1 = f(t_k, y_k) = \dot{y}(t_k, y_k) \quad (6.3)$$

This slope is used to approximate the next function value:

$$y_{k+1} = y_k + hK_1 \quad (6.4)$$

Equation 6.4 describes Euler's method. A single step with step size h taken using this method is often called an *Euler step*.

Calculating the function value of y at $t = 0.5$ for the example problem with Euler's method yields

$$K_1 = f(0, 1) = 1 \cdot (2 - 0) \cdot 0 + 0 - 1 = -1 \quad (6.5)$$

$$\begin{aligned} y(0.5) &\approx y_1 = y_0 + 0.5 \cdot (-1) \\ &= 1 - 0.5 = 0.5 \end{aligned} \quad (6.6)$$

This approximation of the solution is shown in Figure 6.2. Clearly, the correspondence with the actual solution is not very good, since the gradient of the function y has changed significantly between $t = 0$ and $t = 0.5$, while the approximation was based solely on the slope at $t = 0$.

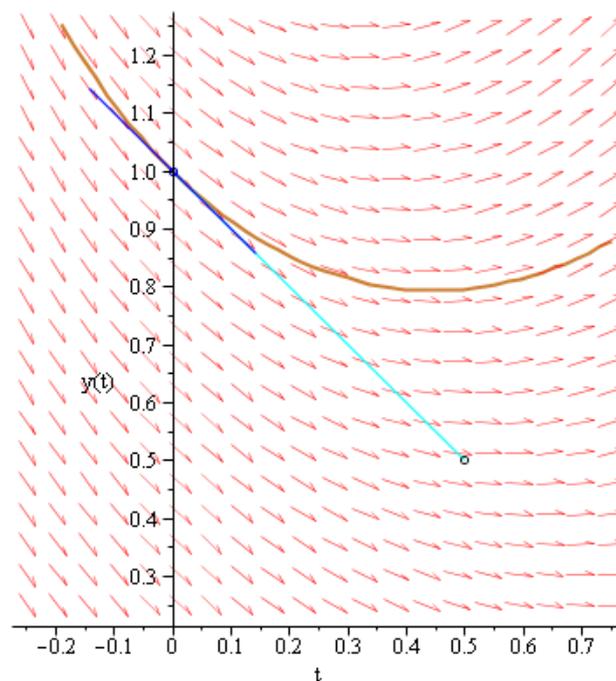


Figure 6.2: An approximation of $y(0.5)$ made using Euler's method (Harder, 2012).

Mid-Point Method

A slightly more advanced integration method is the mid-point method. In this method, the slope in the middle of the integration step (K_2) is used to approximate the next function value:

$$y_{k+1} = y_k + hK_2 \quad (6.7)$$

K_2 is calculated using the slope at the start of the integration step:

$$\begin{aligned} K_1 &= f(t_k, y_k) \\ K_2 &= f\left(t_k + \frac{1}{2}h, y_k + \frac{1}{2}hK_1\right) \end{aligned} \quad (6.8)$$

Computing the value of y at $t = 0.5$ for the example problem using the mid-point method gives

$$\begin{aligned} K_1 &= f(0, 1) = -1 \\ K_2 &= f(0.25, 1 + 0.25 \cdot (-1)) \\ &= f(0.25, 0.75) \\ &= -0.421875 \end{aligned} \quad (6.9)$$

$$\begin{aligned} y(0.5) &\approx y_1 = y_0 + 0.5 \cdot (-0.421875) \\ &= 0.7890625 \end{aligned} \quad (6.10)$$

As can be seen in Figure 6.3, the approximation made using the mid-point method is much better than the one made with Euler's method. However, the prediction can be improved even more by using a somewhat more complex integration method.

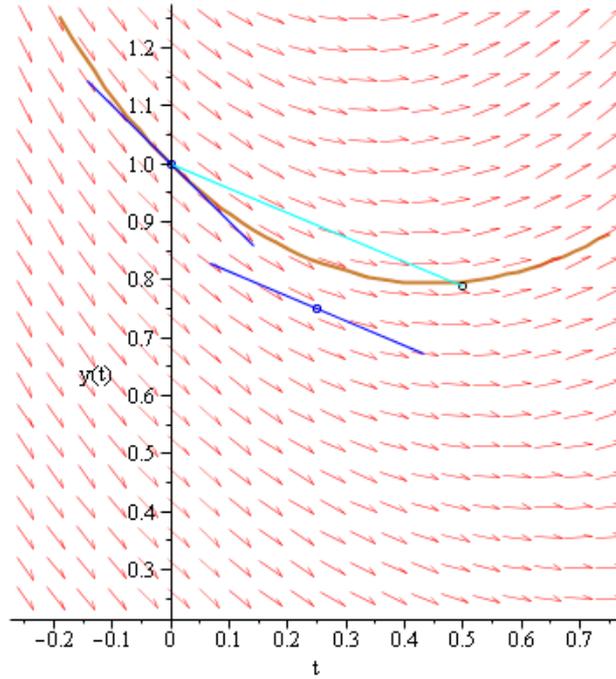


Figure 6.3: An approximation of $y(0.5)$ made using the mid-point method (Harder, 2012).

Runge-Kutta 4 Method

The Runge-Kutta 4 (RK4) method uses four slopes per integration step to calculate the solution. These different slopes are given by

$$K_1 = f(t_k, y_k) \quad (6.11)$$

$$K_2 = f\left(t_k + \frac{1}{2}h, y_k + \frac{1}{2}hK_1\right) \quad (6.12)$$

$$K_3 = f\left(t_k + \frac{1}{2}h, y_k + \frac{1}{2}hK_2\right) \quad (6.13)$$

$$K_4 = f(t_k + h, y_k + hK_3) \quad (6.14)$$

To approximate the value of the solution after time step h , the slopes are combined in a weighted average, where more weight is given to the slopes at the mid-point of the interval:

$$\phi = \frac{1}{6}(K_1 + 2K_2 + 2K_3 + K_4) \quad (6.15)$$

This effective slope is then used for taking the integration step:

$$y_{k+1} = y_k + h\phi \quad (6.16)$$

Working out the equations for the example problem yields

$$y(0.5) \approx y_1 = 0.7962062 \quad (6.17)$$

The different slopes used for taking the integration step for the example problem using the RK4 method are shown in Figure 6.4, along with the corresponding approximation for $f(0.5)$. Clearly, the approximation made with the RK4 method closely resembles the actual solution to the problem.

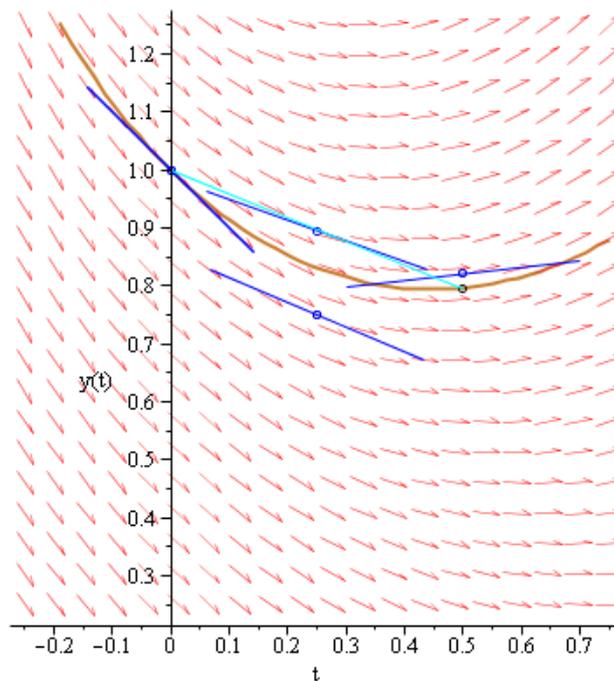


Figure 6.4: An approximation of $y(0.5)$ made using the RK4 method (Harder, 2012).

Comparison

A comparison of the approximations made using the different methods is shown in Table 6.1. Note that this concerns approximations made using a single integration step, as was the case in the previous examples. The *truncation error* is the error with the actual solution, which results from approximating the continuous integral by means of a discrete step.

Table 6.1: A comparison of the approximations of $y(0.5)$ made by the three integration methods *using a single integration step*. The true value of $y(0.5)$ is 0.7963901.

Method	Value for $y(0.5)$	Truncation error
Euler	0.5	0.2963901
Mid-point	0.7890625	0.0073276
RK4	0.7962062	0.0001839

Clearly, applying Euler's method results in the largest truncation error, while the RK4 method yields the smallest truncation error, and the mid-point method gives a truncation error that lies somewhere in between. Even though for all methods a step size of $h = 0.5$ was used, the RK4 method required four function evaluations for making the step, while the mid-point method needed two and Euler's method required only one.

When numerically integrating more complex systems, such as integrating the equations of motion of a satellite, the computation time required is governed to a large degree by the number of (force) function evaluations that are needed. Therefore, it would be most useful to compare different methods based on the accuracy attained when a similar number of function evaluations are used.

Figure 6.5 shows the approximation of the solution to the example problem made using Euler's method with four successive steps. Evidently, the prediction of the solution obtained in this way is still much worse than the prediction obtained using the RK4 method (cf. Figure 6.4).

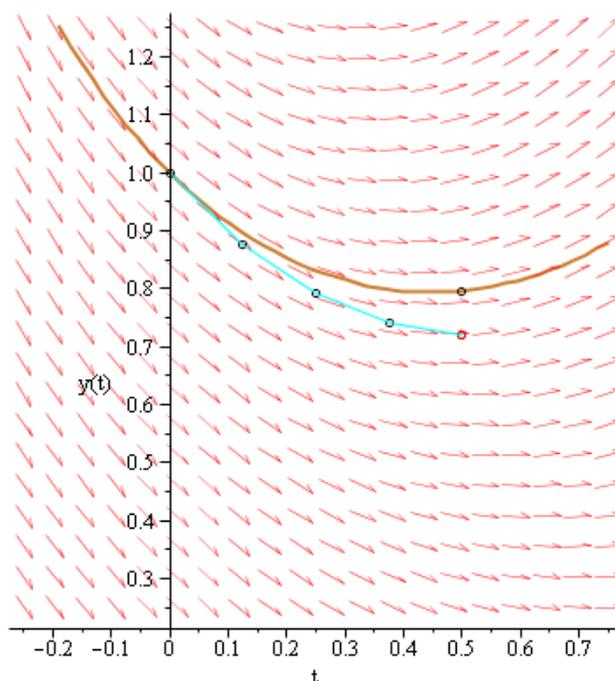


Figure 6.5: The result of approximating the solution to the example problem using Euler's method with four successive steps (Harder, 2012).

More specifically, Table 6.2 lists the approximations that are obtained by applying the different methods to the example problem, using an equal number of function evaluations. Even though both Euler's method and the mid-point method perform better than before, they are both still less accurate than the RK4 method. Hence, for solving this example problem, the RK4 method is clearly more efficient than both the mid-point method and especially Euler's method.

Table 6.2: A comparison of the approximations of $y(0.5)$ made by the three integration methods *using an equal number of function evaluations*. The true value of $y(0.5)$ is 0.7963901.

Method	# of steps	Value for $y(0.5)$	Truncation error
Euler	4	0.7190835	0.0773066
Mid-point	2	0.7957524	0.0006377
RK4	1	0.7962062	0.0001839

Although the previously introduced methods are relatively simple integration methods, the basic aspects of how the integration is performed remains the same for more advanced methods. In general, more advanced methods are better at dealing with complex systems than simpler methods. As a result, the choice of the integration method to use will often depend on the desired level of accuracy and the

characteristics of the problem to be solved.

Vector Equations

While the example problem was a one-dimensional problem, the task of modeling the motion of a space debris object in orbit around the Earth is of course a three-dimensional problem. Fortunately, the concepts explained before can easily be extended to a three-dimensional situation by using vectors.

For integration purposes it is convenient to state the equation of motion as a first-order differential equation of the form

$$\dot{\mathbf{y}} = \mathbf{f}(t, \mathbf{y}) \quad \text{with} \quad \mathbf{y}, \dot{\mathbf{y}}, \mathbf{f} \in \mathbb{R}^n \quad (6.18)$$

This form can always be obtained from the second-order differential equation for the acceleration

$$\ddot{\mathbf{r}} = \mathbf{a}(t, \mathbf{r}, \dot{\mathbf{r}}) \quad (6.19)$$

by combining position \mathbf{r} and velocity $\dot{\mathbf{r}}$ in the 6-dimensional state vector

$$\mathbf{y} = \begin{pmatrix} \mathbf{r} \\ \dot{\mathbf{r}} \end{pmatrix} \quad (6.20)$$

which satisfies the form of Equation 6.18 (Montenbruck and Gill, 2005):

$$\dot{\mathbf{y}} = \mathbf{f}(t, \mathbf{y}) = \begin{pmatrix} \dot{\mathbf{r}} \\ \mathbf{a}(t, \mathbf{r}, \dot{\mathbf{r}}) \end{pmatrix} \quad (6.21)$$

Equation 6.21 is a first-order differential equation, which is in general easier to integrate than a second-order differential equation like Equation 6.19. The methods described in the next sections will make use of the function $\mathbf{f}(t, \mathbf{y})$, as defined in Equation 6.21.

6.2 Accuracy and Efficiency

When performing a numerical integration, there are several sources that induce errors in the calculation. For long-term orbit integrations, the most important sources of errors are the following (Milani and Nobili, 1988):

- *Truncation* (or discretization) *errors*, caused by the replacement of continuous differential equations by finite-difference equations. Truncation errors can be mitigated by using smaller integration steps.
- *Round-off errors*, resulting from the limited accuracy with which computers store numbers. Round-off errors accumulate, meaning that the total error resulting from them increases if more integration steps are used.

- *Instabilities*, that can arise in two ways: either if the step size h is chosen too large or if the dynamical system happens to be chaotic.
- *Errors in the physical model*, which occur if forces or system parameters are approximated in an inaccurate way or if perturbations that have a noticeable effect on the orbital motion are neglected.

The main factors for assessing the performance of an integrator are *accuracy* and *speed*. These two factors often conflict with each other, as high accuracy, in general, requires a small step size, leading to long computation times.

A factor that takes both accuracy and speed into account, and is therefore a good measure for comparing different integrators, is *efficiency*. Usually, a certain accuracy is desired, implying that the total error over the integration should be kept below a certain (pre-defined) threshold. The integrator that satisfies this requirement and is able to complete the integration in the shortest amount of time is considered to be the most efficient integrator for that particular situation. Which integrator is the most efficient depends on the desired accuracy and on the properties of the orbit to be modeled. Hence, there is not one ideal integrator that is the best one to use in every single case.

6.3 Fixed Step Size Methods

Fixed step size methods use an integration step size h that is constant for the entire integration interval, making them relatively straightforward to implement. However, the fact that the step size is fixed implies that the integration method makes no distinction between parts of the system that are simple to integrate and parts that are harder. Consequently, the efficiency of fixed step size methods is generally not optimal for systems that exhibit a range of different behaviors over the integration interval.

6.3.1 Runge-Kutta 4

The RK4 method was already shown for a one-dimensional case in the introduction. Using the notation of Equation 6.21, the method can be extended to a three-dimensional situation. First of all, the approximate solution after a single integration step can be written in the general notation

$$\mathbf{y}(t_0 + h) \approx \mathbf{y}_0 + h \cdot \boldsymbol{\phi} = \boldsymbol{\eta}(t_0 + h) \quad (6.22)$$

In this notation, $\boldsymbol{\phi}$ is called the *increment function*. For the RK4 method, the increment function is calculated by taking a weighted average of four slopes calculated within the integration step:

$$\boldsymbol{\phi}_{\text{RK4}} = \frac{1}{6}(\mathbf{k}_1 + 2\mathbf{k}_2 + 2\mathbf{k}_3 + \mathbf{k}_4) \quad (6.23)$$

where the four slopes are calculated with

$$\mathbf{k}_1 = \mathbf{f}(t_0, \mathbf{y}_0) \quad (6.24)$$

$$\mathbf{k}_2 = \mathbf{f}(t_0 + h/2, \mathbf{y}_0 + h\mathbf{k}_1/2) \quad (6.25)$$

$$\mathbf{k}_3 = \mathbf{f}(t_0 + h/2, \mathbf{y}_0 + h\mathbf{k}_2/2) \quad (6.26)$$

$$\mathbf{k}_4 = \mathbf{f}(t_0 + h, \mathbf{y}_0 + h\mathbf{k}_3) \quad (6.27)$$

The RK4 method is designed to approximate the exact solution up to terms of order h^4 and is therefore called a fourth-order method (Montenbruck and Gill, 2005). The *local truncation error* of the method is bound by a term of order h^5 :

$$e_{RK4} = |\mathbf{y}(t_0 + h) - \boldsymbol{\eta}(t_0 + h)| \leq \text{const} \cdot h^5 \quad (6.28)$$

In literature, this statement about accuracy is often written in the following form:

$$\mathbf{y}(t_0 + h) = \mathbf{y}_0 + h \cdot \boldsymbol{\phi}_{RK4} + O(h^5) \quad (6.29)$$

where the O symbol gives the order of magnitude of the local truncation error. The nomenclature is such that a method of order p has a local truncation error on the order of $O(h^{p+1})$.

6.3.2 General Runge-Kutta Methods

Although there are many more Runge-Kutta methods than just RK4, they all share the same structure. In each of the methods, the slope is evaluated at multiple points within an integration step and a weighted average is taken to obtain the increment function $\boldsymbol{\phi}$.

For a general s -stage² Runge-Kutta method, the increment function is defined as

$$\boldsymbol{\phi} = \sum_{i=1}^s b_i \mathbf{k}_i \quad (6.30)$$

with slope functions

$$\mathbf{k}_1 = \mathbf{f}(t_0 + c_1 h, \mathbf{y}_0) \quad (6.31)$$

$$\mathbf{k}_i = \mathbf{f}(t_0 + c_i h, \mathbf{y}_0 + h \sum_{j=1}^{i-1} a_{ij} \mathbf{k}_j) \quad (i = 2 \dots s) \quad (6.32)$$

This leads to an approximation of

$$\boldsymbol{\eta}(t_0 + h) = \mathbf{y}_0 + h \cdot \boldsymbol{\phi} \quad (6.33)$$

Each Runge-Kutta method is fully described by the coefficients a_{ij} , b_i and c_i , which are chosen in such a way that the order of the local truncation error is as high as possible.

² s indicates the number of function evaluations that are required to form the increment function.

A common way of denoting the coefficients of Runge-Kutta methods is by means of a *Butcher tableau*. In such a table, all coefficients are listed in the following way:

$$\begin{array}{c|cccc}
 c_1 & & & & \\
 c_2 & a_{21} & & & \\
 c_3 & a_{31} & a_{32} & & \\
 \vdots & \vdots & \vdots & \ddots & \\
 c_s & a_{s1} & a_{s2} & \dots & a_{s,s-1} \\
 \hline
 & b_1 & b_2 & \dots & b_{s-1} & b_s
 \end{array}$$

For example, the Butcher tableau for the RK4 method looks as follows:

$$\begin{array}{c|cccc}
 c_i & & & & \\
 \hline
 0 & & & & \\
 \frac{1}{2} & \frac{1}{2} & & & \\
 \frac{1}{2} & 0 & \frac{1}{2} & & \\
 1 & 0 & 0 & 1 & \\
 \hline
 b_i & \frac{1}{6} & \frac{1}{3} & \frac{1}{3} & \frac{1}{6}
 \end{array}$$

Because all Runge-Kutta methods share the structure of Equations 6.30 to 6.32, each Runge-Kutta method can be fully defined by a single Butcher tableau.

6.4 Adaptive Step Size Methods

In addition to fixed step size methods, there are also methods which vary the step size during the integration. These *adaptive step size methods* change the step size such that each integration step contributes uniformly to the total integration error. As a result, these methods tend to be quite efficient, as no more time than is necessary is spent on parts that are easy to integrate, while the step size is automatically decreased for harder parts, making sure that the local error remains within the specified bounds.

6.4.1 General Aspects

Since the coefficients of an integration method are not uniquely determined by the condition of maximum order, it is possible to find various Runge-Kutta methods with an equal number of stages (Montenbruck and Gill, 2005). In addition, methods of neighboring order can be found which rely on the same set of function evaluations. These methods are called *embedded* Runge-Kutta methods, and they allow an easy estimation of the local truncation error, which can be used for efficient step size control.

An s -stage embedded method gives two independent approximations of orders p and $p + 1$:

$$\begin{aligned}\boldsymbol{\eta}(t_0 + h) &= \mathbf{y}_0 + h \cdot \sum_{i=1}^s b_i \mathbf{k}_i \\ \hat{\boldsymbol{\eta}}(t_0 + h) &= \mathbf{y}_0 + h \cdot \sum_{i=1}^s \hat{b}_i \mathbf{k}_i\end{aligned}\tag{6.34}$$

with local truncation errors

$$\begin{aligned}e &= |\mathbf{y}(t_0 + h) - \boldsymbol{\eta}(t_0 + h)| \leq ch^{p+1} \\ \hat{e} &= |\mathbf{y}(t_0 + h) - \hat{\boldsymbol{\eta}}(t_0 + h)| \leq \hat{c}h^{p+2}\end{aligned}\tag{6.35}$$

Because \hat{e} is smaller than e by the order of h , it is possible to get an estimate of the local truncation error of the method of order p from the difference of the two solutions:

$$e = |\mathbf{y} - \boldsymbol{\eta}| \approx |\hat{\boldsymbol{\eta}} - \boldsymbol{\eta}|\tag{6.36}$$

Although this could also be accomplished with arbitrary methods of neighboring order, embedded methods have the advantage of only requiring s evaluations of the force function instead of $2s - 1$, since the same set of function evaluations is used for both $\hat{\boldsymbol{\eta}}$ and $\boldsymbol{\eta}$.

The notation $\text{RK}p(q)$ is generally used for denoting embedded Runge-Kutta methods, implying that the specific method is of order p with an embedded method of order q for step size control.

Step Size Control

A common technique for step size control is based on limiting the local truncation error e , making use of the error estimate available with embedded methods. The process works as follows (Montenbruck and Gill, 2005):

Once a single integration step has been performed with step size h , the local truncation error is estimated using

$$e(h) \approx |\hat{\boldsymbol{\eta}} - \boldsymbol{\eta}|\tag{6.37}$$

If this error is larger than a specified tolerance ε , the integration step will be performed again using a smaller step size h^* . Since for the method of order p , $e(h)$ is proportional to h^{p+1} , the local truncation error for the new step size will be equal to

$$e(h^*) = e(h) \left(\frac{h^*}{h}\right)^{p+1} \approx |\hat{\boldsymbol{\eta}} - \boldsymbol{\eta}| \left(\frac{h^*}{h}\right)^{p+1}\tag{6.38}$$

This new truncation error needs to be smaller than ε . Solving Equation 6.38 for h^* results in the maximum allowed step size for repeating the step:

$$h^* = {}^{p+1}\sqrt{\frac{\varepsilon}{e(h)}}h \approx {}^{p+1}\sqrt{\frac{\varepsilon}{|\hat{\eta} - \eta|}}h \quad (6.39)$$

In practice, a safety factor is often used in order to avoid another unsuccessful step. In the simulation code, the value computed with Equation 6.39 is multiplied by 0.8 for this reason.

In addition, rapid oscillations of the step size are not desired either. Therefore, minimum and maximum values for the change in step size have been specified in the simulation code. These values make sure that the factor with which the step size changes always lies between 0.1 and 4.

6.4.2 Runge-Kutta-Fehlberg 5(6)

Runge-Kutta-Fehlberg 5(6) is a method by Fehlberg (1968), consisting of a method of order 5 with an embedded method of order 6 for step size control and a total of 8 stages. In this report, it is abbreviated as RKF56. The Butcher tableau for the RKF56 method is as follows:

c_i	a_{ij}							
0								
$\frac{1}{6}$	$\frac{1}{6}$							
$\frac{4}{15}$	$\frac{4}{15}$	$\frac{16}{75}$						
$\frac{2}{3}$	$\frac{5}{6}$	$-\frac{8}{3}$	$\frac{5}{2}$					
$\frac{4}{5}$	$-\frac{8}{5}$	$\frac{144}{25}$	-4	$\frac{16}{25}$				
1	$\frac{361}{320}$	$-\frac{18}{5}$	$\frac{407}{128}$	$-\frac{11}{80}$	$\frac{55}{128}$			
0	$-\frac{11}{640}$	0	$\frac{11}{256}$	$-\frac{11}{160}$	$\frac{11}{256}$	0		
1	$\frac{93}{640}$	$-\frac{18}{5}$	$\frac{803}{256}$	$-\frac{11}{160}$	$\frac{99}{256}$	0	1	
\hat{b}_i	$\frac{7}{1408}$	0	$\frac{1125}{2816}$	$\frac{9}{32}$	$\frac{125}{768}$	0	$\frac{5}{66}$	$\frac{5}{66}$
b_i	$\frac{31}{384}$	0	$\frac{1125}{2816}$	$\frac{9}{32}$	$\frac{125}{768}$	$\frac{5}{66}$	0	0

Note that because this is an embedded Runge-Kutta method, there are two sets of b -coefficients. The first row of coefficients (\hat{b}_i) gives the sixth-order accurate method, while the second row (b_i) gives the coefficients of the fifth-order accurate method. In this case, the lower-order method is used to propagate the solution, while the higher-order method is only used for step size control.

6.4.3 Dormand-Prince 8(7)

The Dormand-Prince 8(7) method, often abbreviated as DOPRI8, is a method by Prince and Dormand (1981). It uses 13 function evaluations to obtain an eighth-order approximation of the solution. The Butcher tableau for the DOPRI8 method is shown in Figure 6.6.

c_i	a_{ij}	b_j
0		
$\frac{1}{18}$		
$\frac{1}{12}$		
$\frac{1}{8}$		
$\frac{5}{16}$		
$\frac{3}{8}$		
$\frac{59}{400}$		
$\frac{93}{200}$		
$\frac{5490023248}{9719169821}$		
13		
20		
$\frac{1201146811}{1299019798}$		
1		
1		0
\hat{h}_i		
b_i		

Figure 6.6: The Butcher tableau for the DOPRI8 method (Montenbruck and Gill, 2005).

The first row of coefficients at the bottom of the Butcher tableau (\hat{b}_i) corresponds to the eighth-order method, whereas the second row (b_i) gives the coefficients of the seventh-order method. In contrast to the RKF56 method introduced in the previous section, the DOPRI8 method uses the higher-order method to propagate the solution, while the lower-order method is used for error estimation.

6.5 Choice of Integration Methods

The RK4, RKF56 and DOPRI8 methods have been implemented in the simulation code. All of these methods have the option of being used in combination with Cowell's method or with Gauss' form of Lagrange's planetary equations with modified equinoctial elements. This yields six different options, all of which are tested in the performance comparison of Chapter 9.

The RK4 method has been chosen because it is a very well-known method and serves well as a reference for comparison for the other methods that are tested. That being said, it is also a fairly simple method and it is not expected to be the most efficient one for integrating the relatively complex system.

DOPRI8, on the other hand, is a more sophisticated method, which, according to Montenbruck and Gill (2005) and Hairer et al. (1987), can be recommended as a general-purpose method for a wide range of applications. Also, because it is an adaptive step size method, it is expected to be efficient for different kinds of orbits and well able to deal with irregularities in the force model.

The RKF56 method lies somewhere in between the RK4 and DOPRI8 methods, both in terms of complexity and expected performance. Because the method used for propagating the solution is only one order higher than the RK4 method, it will be interesting to see how much of a difference it makes that the RKF56 method is an adaptive step size method, rather than a fixed step size method like RK4.

Of course, many more integration methods exist than have been treated in this chapter, which focused on the methods that have actually been implemented in the simulation code. An overview of integration methods that also includes a number of other methods can be found in the literature survey preceding this thesis report (Hofsteenge, 2012).

An interesting category of these other methods is posed by the multi-step methods. Multi-step methods use function values of the current step, as well as those from several previous steps to integrate to the next step. This requires storage of previous function values (called backpoints), but generally reduces the total number of function evaluations that are needed. Therefore, multi-step methods could certainly be considered for long-term integrations of space debris orbits.

However, multi-step methods require the force function $f(t, \mathbf{y})$ to be continuous and smooth throughout the set of backpoints (Berry, 2004). In this light, the force due to solar radiation pressure forms a problem, as it is discontinuous when encountering eclipse boundaries. Since solar radiation pressure is one of the most important perturbations in the situations to be modeled for this thesis project (pri-

marily when using high area-to-mass ratios), this aspect needs to be taken into account. One way to avoid the corresponding integration error is to restart the integration at the point of the discontinuity. Yet, these restarts would be required twice for each orbital revolution in which eclipse is encountered, as the orbiting object moves in and out of eclipse. Although this would not necessarily make multi-step methods poor methods to use, it would certainly have an impact on their performance.

Furthermore, one of the main research interests of this thesis lies in investigating how the performance of symplectic methods compares with the performance of more established traditional methods. For this reason, and the fact that within the time frame of the thesis project only a limited number of methods can be implemented and tested, multi-step methods and other methods such as extrapolation methods have been left out, in favor of having the option to include and test symplectic integration methods. These relatively new methods will be described in the next chapter.

Symplectic Integration Methods

Symplectic integration methods are quite unlike the methods described in the previous chapter. One of their major strengths lies in their energy preservation properties, making them particularly useful for long-term integrations. The main application of symplectic methods in orbital mechanics has been the integration of the n -body problem, which implies integration of a purely Hamiltonian system. In that application, symplectic integrators exactly conserve the angular momentum vector (Kinoshita and Nakai, 1992).

In this thesis project, however, symplectic methods will be applied to the integration of space debris orbits pertaining to the force model described in Chapter 4. This involves the integration of a *perturbed* Hamiltonian system. Since the application of symplectic methods to the modeling of space debris orbits is relatively new, it is largely unclear how these methods compare with the traditional methods. This makes symplectic methods an interesting category of integrators to investigate for this thesis project.

A detailed description of the theoretical background of symplectic methods could be given in this chapter. However, from the point of view of this thesis, the application is more important than the mathematical intricacies involved. Therefore, this chapter primarily describes the concepts that are necessary for understanding how symplectic integrators work and how they can be implemented. A more in-depth treatment of symplectic methods, including the underlying mathematics, can be found in the works of Hairer et al. (2006) or Fecko (2006).

7.1 Exponential Operators

Exponential operators are frequently used in the notation of symplectic integration methods. In order to facilitate the understanding of the subsequent sections, this section provides a short overview of the most important properties of exponential operators. In the equations below, \mathcal{A} , \mathcal{B} and \mathcal{C} are operators.

Important to note is that the commutative law generally does *not* hold for operators. Hence, in general, $\mathcal{A}\mathcal{B} \neq \mathcal{B}\mathcal{A}$. Therefore, it is convenient to define the quantity

$$[\mathcal{A}, \mathcal{B}] = \mathcal{A}\mathcal{B} - \mathcal{B}\mathcal{A} \quad (7.1)$$

which is called the commutator of \mathcal{A} and \mathcal{B} . Note that the order is important, so that $[\mathcal{A}, \mathcal{B}] = -[\mathcal{B}, \mathcal{A}]$. The operation of Equation 7.1 is also called Lie multiplication in literature (Eidelman, 2000). Lie multiplication also has the property of linearity

$$[\mathcal{A}, (\beta\mathcal{B} + \gamma\mathcal{C})] = \beta[\mathcal{A}, \mathcal{B}] + \gamma[\mathcal{A}, \mathcal{C}] \quad (7.2)$$

and the corresponding product rule is given by

$$[\mathcal{A}, (\mathcal{B}\mathcal{C})] = [\mathcal{A}, \mathcal{B}]\mathcal{C} + \mathcal{B}[\mathcal{A}, \mathcal{C}] \quad (7.3)$$

Operators which have these characteristics are called *Lie operators*.

The *exponential operator* of \mathcal{A} is defined as

$$e^{\mathcal{A}} = \mathcal{A}^0 + \frac{1}{1!}\mathcal{A}^1 + \frac{1}{2!}\mathcal{A}^2 + \dots = \sum_{n=0}^{\infty} \frac{\mathcal{A}^n}{n!} \quad (7.4)$$

Following the previous definition, the product of two exponential operators is given by

$$\begin{aligned} e^{\mathcal{A}}e^{\mathcal{B}} &= \left(\sum_{i=0}^{\infty} \frac{\mathcal{A}^i}{i!} \right) \left(\sum_{k=0}^{\infty} \frac{\mathcal{B}^k}{k!} \right) \\ &= \left(\mathcal{A}^0 + \frac{1}{1!}\mathcal{A}^1 + \frac{1}{2!}\mathcal{A}^2 + \frac{1}{3!}\mathcal{A}^3 + \dots \right) \\ &\quad \cdot \left(\mathcal{B}^0 + \frac{1}{1!}\mathcal{B}^1 + \frac{1}{2!}\mathcal{B}^2 + \frac{1}{3!}\mathcal{B}^3 + \dots \right) \\ &= I + \frac{1}{1!}\mathcal{A}^1 + \frac{1}{1!}\mathcal{B}^1 + \frac{1}{2!}\mathcal{A}^2 + \frac{1}{1!1!}\mathcal{A}^1\mathcal{B}^1 + \frac{1}{2!}\mathcal{B}^2 \\ &\quad + \frac{1}{3!}\mathcal{A}^3 + \frac{1}{2!1!}\mathcal{A}^2\mathcal{B}^1 + \frac{1}{1!2!}\mathcal{A}^1\mathcal{B}^2 + \frac{1}{3!}\mathcal{B}^3 + \dots \\ &= \sum_{i=0}^{\infty} \sum_{k=0}^i \frac{\mathcal{A}^{i-k}}{(i-k)!} \frac{\mathcal{B}^k}{k!} \end{aligned} \quad (7.5)$$

An important question is whether exponential operators commute, i.e.

$$e^{\mathcal{A}}e^{\mathcal{B}} = e^{\mathcal{B}}e^{\mathcal{A}} \quad ? \quad (7.6)$$

Checking this using Equation 7.5 yields that

$$e^{\mathcal{A}}e^{\mathcal{B}} - e^{\mathcal{B}}e^{\mathcal{A}} = \dots = [\mathcal{A}, \mathcal{B}] + \frac{1}{2!}[(\mathcal{A} + \mathcal{B}), [\mathcal{A}, \mathcal{B}]] + \dots \quad (7.7)$$

Hence, if \mathcal{A} and \mathcal{B} do not commute, then the exponentials do not commute either.

Another important question is whether the standard rule for multiplication of exponents is valid for exponential operators, i.e.

$$e^{\mathcal{A}}e^{\mathcal{B}} = e^{\mathcal{A}+\mathcal{B}} \quad ? \quad (7.8)$$

Using the power series definition of Equation 7.4 together with Equation 7.5 results in

$$e^{\mathcal{A}+\mathcal{B}} - e^{\mathcal{A}}e^{\mathcal{B}} = \dots = \frac{1}{2!}(\mathcal{A}\mathcal{B} + \mathcal{B}\mathcal{A}) - \mathcal{A}\mathcal{B} + \dots = -\frac{1}{2!}[\mathcal{A}, \mathcal{B}] + \dots \quad (7.9)$$

Thus, the standard rule for multiplication of Equation 7.8 is only valid if operators \mathcal{A} and \mathcal{B} commute.

7.2 Hamiltonian Mechanics

Symplectic integration methods are designed for the integration of Hamiltonian systems. These are systems which can be described by *Hamiltonian mechanics*, which essentially is a reformulation of classical mechanics. Since Hamiltonian mechanics is an extensive topic in itself, this section will only address some of the basics that are necessary to understand the integration methods.

Central to Hamiltonian mechanics is the concept of the Hamiltonian \mathcal{H} . This quantity is equal to the total energy of the system that is described. For closed systems, the Hamiltonian is equal to the sum of the kinetic and potential energy in the system:

$$\mathcal{H} = T + V \quad (7.10)$$

where T is the kinetic energy and V is the potential energy of the system.

A system with d degrees of freedom is described with the generalized coordinates

$$\mathbf{q} = (q_1, \dots, q_d)^T \quad (7.11)$$

These can be chosen as any set of coordinates. In a Cartesian coordinate frame, these will be x , y and z . Using generalized coordinates, the kinetic energy has the form

$$T \equiv T(\mathbf{q}, \dot{\mathbf{q}}) \quad (7.12)$$

and the potential energy is of the form

$$V \equiv V(\mathbf{q}) \quad (7.13)$$

The Lagrangian of the system is defined as

$$L = T - V \quad (7.14)$$

The generalized coordinates \mathbf{q} obey the following set of differential equations, which are called the Euler-Lagrange equations (Hand and Finch, 2008):

$$\frac{d}{dt} \left(\frac{\partial L}{\partial \dot{q}_j} \right) - \frac{\partial L}{\partial q_j} = 0 \quad (j = 1, \dots, d) \quad (7.15)$$

Consequently, the solution of these differential equations describes the motion of the system.

Using the Lagrangian, a more general definition of the Hamiltonian can be given:

$$\mathcal{H}(\mathbf{p}, \mathbf{q}) = \mathbf{p}^T \dot{\mathbf{q}} - L(\mathbf{q}, \dot{\mathbf{q}}) \quad (7.16)$$

where the notation \mathbf{p} has been used for the vector of generalized momenta, defined as

$$p_j = \frac{\partial L}{\partial \dot{q}_j} \quad (7.17)$$

The corresponding equations of motion of the system are described by the *Hamilton equations*:

$$\frac{dp_j}{dt} = - \frac{\partial \mathcal{H}}{\partial q_j} \quad (7.18)$$

$$\frac{dq_j}{dt} = \frac{\partial \mathcal{H}}{\partial p_j} \quad (7.19)$$

where q_j and p_j indicate generalized coordinates and generalized momenta, respectively, and $j = 1, \dots, d$.

It should be mentioned that Hamilton's equations are equivalent to the Euler-Lagrange equations, and both approaches lead to the same equations for the same generalized momentum. The main reason to use Hamiltonian mechanics instead of Lagrangian mechanics is that Hamiltonian systems have a symplectic structure, which can be utilized by means of symplectic integrators.

7.3 Symplectic Mappings

In symplectic integration, use is made of a $2d \times 2d$ matrix J , called the *symplectic identity* (Hairer et al., 2006):

$$J = \begin{bmatrix} 0 & I_d \\ -I_d & 0 \end{bmatrix} \quad (7.20)$$

where I_d is the $d \times d$ identity matrix.

Furthermore, a differentiable mapping $g : U \rightarrow \mathbb{R}^{2d}$ (where $U \subset \mathbb{R}^{2d}$ is an open set) is called *symplectic* if the Jacobian matrix $g'(p, q)$ has the following property at every point:

$$g'(p, q)^T J g'(p, q) = J \quad (7.21)$$

Symplectic mappings have the important property of *area preservation*. When applying symplectic mappings to problems in Hamiltonian mechanics, this implies that the total energy of the system is conserved by definition. Especially when long periods of time are considered, this property can give symplectic integrators a significantly higher accuracy than traditional integrators.

7.4 Symplectic Integrators

7.4.1 General Formulation

Symplectic integrators essentially integrate a trajectory by applying a sequence of symplectic mappings to a state in order to arrive at a later state. To perform the actual integration, the Hamiltonian vector field is written as

$$\dot{\mathbf{x}} = \{\mathbf{x}, \mathcal{H}\} = \sum_{j=1}^d \left(\frac{\partial \mathcal{H}}{\partial p_j} \frac{\partial \mathbf{x}}{\partial q_j} - \frac{\partial \mathcal{H}}{\partial q_j} \frac{\partial \mathbf{x}}{\partial p_j} \right) \quad (7.22)$$

where

$$\mathbf{x} = \begin{pmatrix} \mathbf{q} \\ \mathbf{p} \end{pmatrix} \in \mathbb{R}^{2d} \quad (7.23)$$

The notation is often shortened by making use of the differential operator $L_{\mathcal{H}}\mathbf{x}$, which is defined as $L_{\mathcal{H}}\mathbf{x} = \{\mathbf{x}, \mathcal{H}\}$. The formal solution of Equation 7.22 is then given by

$$\mathbf{x}(t) = \sum_{n=0}^{\infty} \frac{t^n}{n!} L_{\mathcal{H}}^n \mathbf{x}(t_0) = e^{tL_{\mathcal{H}}} \mathbf{x}(t_0) \quad (7.24)$$

It is not possible to express the operation $e^{tL_{\mathcal{H}}}\mathbf{x}$ in closed form, because the perturbed two-body problem is not analytically integrable. However, an explicit solution can be found by breaking the problem up into individually integrable parts. This solution preserves an approximate value of the Hamiltonian.

7.4.2 Hamiltonian Splitting

Symplectic integration methods for Hamiltonian systems are generally formulated by splitting the Hamiltonian into two parts:

$$\mathcal{H} = \mathcal{H}_A + \mathcal{H}_B \quad (7.25)$$

These two parts are then *integrated sequentially* using operators L_A and L_B , which are given by

$$L_A = \{\circ, \mathcal{H}_A\} = \sum_{j=1}^d \left(\frac{\partial \mathcal{H}_A}{\partial p_j} \frac{\partial}{\partial q_j} - \frac{\partial \mathcal{H}_A}{\partial q_j} \frac{\partial}{\partial p_j} \right) \quad (7.26)$$

and

$$L_B = \{\circ, \mathcal{H}_B\} = \sum_{j=1}^d \left(\frac{\partial \mathcal{H}_B}{\partial p_j} \frac{\partial}{\partial q_j} - \frac{\partial \mathcal{H}_B}{\partial q_j} \frac{\partial}{\partial p_j} \right) \quad (7.27)$$

The equation for an exact propagation of the solution along a time step τ may then be stated by

$$\mathbf{x}(t + \tau) = e^{\tau(L_A + L_B)} \mathbf{x}(t) \quad (7.28)$$

To integrate the two parts sequentially, the exponential is expanded and higher-order terms resulting from the general non-commutativity of L_A and L_B are truncated:

$$\begin{aligned} \mathbf{x}(t + \tau) &= \left(e^{\tau L_A} e^{\tau L_B} - \frac{\tau^2}{2} (L_A L_B - L_B L_A) + \dots \right) \mathbf{x}(t) \\ &= e^{\tau L_A} e^{\tau L_B} \mathbf{x}(t) + O(\tau^2) \end{aligned} \quad (7.29)$$

It is important to note that the order of evaluating a sequence of exponential operators is from right to left. Each operator applied to the state vector returns an updated state vector. The separate steps $e^{\tau L_A}$ and $e^{\tau L_B}$ can be computed exactly, on the condition that the Hamiltonian is split into integrable parts.

If the second row of Equation 7.29 is used for propagating the system over a time step τ , an error of order $O(\tau^2)$ is made, making it a first-order method. This essentially is a symplectic version of Euler's method.

In order to obtain a symplectic method accurate to order $O(\tau^k)$, a series of n propagations over the fractions c_i and d_i of a full time step τ may be chained together¹:

$$\mathbf{x}(t + \tau) = e^{d_n \tau L_B} e^{c_n \tau L_A} \dots e^{d_2 \tau L_B} e^{c_2 \tau L_A} \dots e^{d_1 \tau L_B} e^{c_1 \tau L_A} \mathbf{x}(t) + O(\tau^{k+1}) \quad (7.30)$$

The coefficients c_i and d_i need to be chosen carefully in order to maximize the order of the integrator. These c_i and d_i coefficients uniquely define a symplectic integrator. The specific symplectic integrators that have been implemented in the simulation code will be treated in the next sections. In addition, the two main methods for splitting the Hamiltonian will be discussed in Section 7.5.

7.4.3 Kinoshita's Method

This section concerns a symplectic method described by Kinoshita and Nakai (1992). As the method is not given a specific name in the paper itself, for brevity it will simply be called Kinoshita's method in this report. The corresponding coefficients can be found in Table 7.1. Because $d_4 = 0$, the method requires only three function evaluations per integration step. However, the local truncation error is of order τ^5 , making it a fourth-order symplectic method. Hence, Kinoshita's method requires one evaluation of the force model less than the RK4 method to produce a fourth-order approximation of the solution.

¹The mathematical proof for this can be found by using the Campbell-Baker-Hausdorff theorem (Bourbaki, 1972).

Table 7.1: The coefficients of Kinoshita's method (Kinoshita and Nakai, 1992). β has a value of $2^{1/3}$.

Coefficient	Exact value	Numerical Value
c_1	$\frac{1}{2(2-\beta)}$	0.675603596075797
c_2	$\frac{(1-\beta)}{2(2-\beta)}$	-0.175603596075797
c_3	$\frac{(1-\beta)}{2(2-\beta)}$	-0.175603596075797
c_4	$\frac{1}{2(2-\beta)}$	0.675603596075797
d_1	$\frac{1}{(2-\beta)}$	1.351207192151594
d_2	$\frac{-\beta}{(2-\beta)}$	-1.702414384303188
d_3	$\frac{1}{(2-\beta)}$	1.351207192151594
d_4	0	0

Note that some of the coefficients have negative values. As a result, the method uses both positive and negative steps in time when performing an integration step.

7.4.4 SABA_{2n} Methods

The SABA_{2n} methods are a set of symplectic methods developed by Laskar and Robutel (2001). In contrast to Kinoshita's method, these methods only use positive steps. According to the authors, this yields significantly better stability properties for large step sizes than is possible when negative steps are also included.

Laskar and Robutel actually present several sets of methods, of which the SABA_{2n} set is only one. The name refers to the order in which the sub-steps in each integration step are made. Since the performance of the different sets seems equivalent in the comparisons in the paper, only the SABA_{2n} set will be considered here. The prototype for all SABA_{2n} integrators is given by

$$e^{\tau L_K} = e^{c_1 \tau L_A} e^{d_1 \tau L_B} \dots e^{d_n \tau L_B} e^{c_{n+1} \tau L_A} e^{d_n \tau L_B} \dots e^{d_1 \tau L_B} e^{c_1 \tau L_A} \quad (7.31)$$

Note that this definition implies that the SABA_{2n} integrators are all symmetric. The index in the name gives the number of function evaluations that are necessary for each integration step. For this thesis project, three of the SABA_{2n} integrators have been implemented in the simulation code, being SABA₆, SABA₈ and SABA₁₀. Their coefficients are listed in Tables 7.2 to 7.4.

Table 7.2: The coefficients of SABA₆ (Laskar and Robutel, 2001).

Coefficient	Value
c_1	0.033765242898423986093849222753002695
c_2	0.135630063868443757075450979737044631
c_3	0.211295100191533802515448936669596706
c_4	0.238619186083196908630501721680711935
d_1	0.085662246189585172520148071086366447
d_2	0.180380786524069303784916756918858056
d_3	0.233956967286345523694935171994775497

Table 7.3: The coefficients of SABA₈ (Laskar and Robutel, 2001).

Coefficient	Value
c_1	0.019855071751231884158219565715263505
c_2	0.081811689541954746046003466046821277
c_3	0.135567033748648876886907443643292044
c_4	0.171048883710339590439131453414531184
c_5	0.183434642495649804939476142360183981
d_1	0.050614268145188129576265677154981095
d_2	0.111190517226687235272177997213120442
d_3	0.156853322938943643668981100993300657
d_4	0.181341891689180991482575224638597810

Table 7.4: The coefficients of SABA₁₀ (Laskar and Robutel, 2001).

Coefficient	Value
c_1	0.013046735741414139961017993957773973
c_2	0.054421580914093604672933661830479502
c_3	0.092826899194980052248884661654309736
c_4	0.123007087084888607717530710974544707
c_5	0.142260527573807989957219971018032089
c_6	0.148874338981631210884826001129719985
d_1	0.033335672154344068796784404946665896
d_2	0.074725674575290296572888169828848666
d_3	0.109543181257991021997767467114081596
d_4	0.134633359654998177545613460784734677
d_5	0.147762112357376435086946497325669165

7.5 Methods for Splitting the Hamiltonian

As was mentioned in Section 7.4.2, the Hamiltonian needs to be split up in individually integrable parts in order to perform the integration. These parts are then propagated sequentially using symplectic integration methods. Two of the main methods for splitting the Hamiltonian, which have both been implemented in the simulation code, will be treated next.

7.5.1 T + V Splitting

A commonly used splitting method consists of splitting the Hamiltonian in a part composed of the kinetic energy and a part composed of the potential energy of the system. Hence,

$$\mathcal{H} = T + V \quad (7.32)$$

or, in the notation of Equation 7.25,

$$\begin{aligned} \mathcal{H}_A &= T \\ \mathcal{H}_B &= V \end{aligned} \quad (7.33)$$

Using the force model described in Chapter 4, the components become

$$\begin{aligned} \mathcal{H}_A &= \mathcal{H}_{Kinetic} = T \\ \mathcal{H}_B &= \mathcal{H}_{Potential} = V_{TwoBody} + V_{Geopotential} + V_{Sun} + V_{Moon} + V_{SRP} \end{aligned} \quad (7.34)$$

Accordingly, applying the operator $e^{c_i \tau L_A}$ to the state vector results in the mapping

$$\begin{pmatrix} \mathbf{q} \\ \mathbf{p} \end{pmatrix} \mapsto \begin{pmatrix} \mathbf{q} + c_i \tau \frac{\partial T}{\partial \mathbf{p}}(\mathbf{p}) \\ \mathbf{p} \end{pmatrix} \quad (7.35)$$

whereas the operator $e^{d_i \tau L_B}$ yields the mapping

$$\begin{pmatrix} \mathbf{q} \\ \mathbf{p} \end{pmatrix} \mapsto \begin{pmatrix} \mathbf{q} \\ \mathbf{p} - d_i \tau \frac{\partial V}{\partial \mathbf{q}}(\mathbf{q}) \end{pmatrix} \quad (7.36)$$

when applied to the state vector. In practice, the mapping of Equation 7.35 amounts to propagating the position coordinates by adding the velocity components multiplied by a part of the time step to the old position coordinates. The mapping of Equation 7.36, on the other hand, implies adding the accelerations resulting from the force model multiplied by a different part of the time step to the momentum vector. Following Equation 7.30, the position and momentum vectors continue to be propagated sequentially until the time step is complete.

7.5.2 Wisdom-Holman Splitting

Another useful splitting method is the Wisdom-Holman (WH) method, introduced by Wisdom and Holman (1991). Using this method, the Hamiltonian is split into a two-body part \mathcal{H}_A and a smaller perturbing part \mathcal{H}_B . The perturbation strength ratio ε gives the relative magnitude of the perturbations:

$$\varepsilon = \mathcal{H}_B / \mathcal{H}_A \ll 1 \quad (7.37)$$

For the force model described in Chapter 4, the two parts become

$$\begin{aligned} \mathcal{H}_A &= \mathcal{H}_{Kepler} = T + V_{TwoBody} \\ \mathcal{H}_B &= \mathcal{H}_{Perturbations} = V_{Geopotential} + V_{Sun} + V_{Moon} + V_{SRP} \end{aligned} \quad (7.38)$$

The two-body part can be handled in different ways. In the simulation code, it is propagated using Kepler's equation

$$E(t) - e \sin E(t) = n(t - \tau) \quad (7.39)$$

Note that in this equation, τ indicates the time of pericenter passage, and not the integration step size. Since Kepler's equation cannot be solved analytically, it is solved by means of the Newton-Raphson method in the simulation code.

Conforming to Equation 7.30, applying the WH splitting method amounts to sequentially propagating the two-body part and adding the contribution from the perturbing forces.

Since the central gravitational force is by far the largest force that is acting (cf. Figure 4.1), and this force is effectively removed from the force model as far as direct acceleration is concerned, WH splitting allows the integration to remain stable for very large step sizes. As a result, the WH splitting method is expected to be able to yield relatively accurate results after long integration intervals within comparatively short computation times.

Chapter 8

Verification of the Simulation Code

Before applying the simulation code to the situations for which it was created, it is essential to verify that the code is working correctly and that the results produced using it accurately represent reality. This is no trivial task, as results that are not reflective of reality are meaningless from a practical point of view. Therefore, many aspects of the simulation code will be checked quite extensively in this chapter.

The chapter starts out with an overview of the simulation code, followed by a test consisting of integrating the two-body problem. After that, the routines for computing the positions of the Sun and the Moon at any point in time will be assessed. Then, all major perturbing forces included in the force model will be tested individually. Finally, long-term predictions made using the simulation code will be compared with data from existing research.

8.1 Overview of the Simulation Code

For performing the simulations and performance tests of this thesis project, a simulation code has been written in C++. The code has been built around the Tudat¹ framework and expands upon it. The simulation code can be used for long-term simulations of the orbits of satellites and debris objects with a large range of area-to-mass ratios, and is essentially applicable to simulations within all regions of space near the Earth, except LEO. This last exception is in place, because atmospheric drag plays an important role in LEO, and this particular force has not been taken into account directly. The reason for not including drag in the force model is because the main application for this thesis project is the simulation of orbits within the GEO and GNSS regions of space, in which atmospheric drag is com-

¹The TU Delft Astrodynamics Toolbox (Tudat) is a C++ library developed and maintained by staff and students at the Chair of Astrodynamics and Space Missions at the Faculty of Aerospace Engineering of Delft University of Technology. More information about Tudat can be found in the literature survey preceding this thesis project (Hofsteenge, 2012) or on the Tudat website, accessible at <http://tudat.tudelft.nl/>.

pletely negligible. In case a simulated orbit would develop in such a way that the altitude becomes less than 1,000 km, the simulation is automatically stopped and the user is notified that this altitude boundary has been crossed. For interpreting the results, it is then assumed that the debris object will decay relatively shortly afterwards. The research mentioned in Section 3.2.2 confirms the validity of this assumption.

A number of different settings and initial conditions can be specified in the simulation code. These include:

- The initial conditions of the orbit in classical orbital elements.
- The starting epoch in MJD. This is relevant for the initial positions of the Sun and the Moon, as well as for the orientation of the Earth, which has an effect on the accelerations resulting from the geopotential model.
- The simulation interval.
- The effective area-to-mass ratio $C_R \cdot A/m$ of the satellite or debris object.
- The perturbing forces to be included in the force model:
 - Geopotential force up to a specified degree and order.
 - Gravitational attraction of the Sun.
 - Gravitational attraction of the Moon.
 - Solar radiation pressure
- The integration method to be used:
 - Traditional methods:
 - Runge-Kutta 4.
 - Runge-Kutta-Fehlberg 5(6).
 - Dormand-Prince 8(7).

These methods can be used with either of these formulations:

- Cowell formulation (Cartesian components).
- Gauss' equations with modified equinoctial elements.

- Symplectic methods:

- Kinoshita's method.
- SABA₆.
- SABA₈.
- SABA₁₀.

These methods can be used with either of these splitting methods:

- T+V splitting.
- Wisdom-Holman splitting.

- The accuracy of the integration can be specified using:
 - Step size for fixed step size methods.
 - Relative and absolute error tolerances for adaptive step size methods. Minimum and maximum step sizes can also be specified.
- The format of the output data:
 - Orbital elements.
 - Cartesian state vectors.
- The output interval. A reasonable output interval is necessary to keep the sizes of data files manageable.

A number of functionalities from Tudat have been used, while other functions have been created specifically for this thesis project. Also, some features from the Boost and Eigen libraries have been used in the code. These libraries are also utilized in the code of Tudat itself. Specific parts that are used from Tudat can be found in the way that integration routines have been implemented and in the way that forces are handled, although the routines for modeling the individual perturbing forces have been created for this project. The implementations of RK4, RKF56 and DOPRI8 from Tudat have been used, though some bugs in Tudat have been found and corrected for the latter two methods (see Section 9.4.1). Tudat has also been used for coordinate transformations, while the code for the implementation of Gauss' equations with modified equinoctial elements has been partially adapted from existing beta code. Finally, the implementations of the symplectic methods, including the different splitting methods, have been created specifically for this project, in accordance with the Tudat structure for numerical integrators.

All parts of the code have been carefully tested and verified. The results of the verification can be found in this chapter.

8.2 Two-Body Problem

In the absence of perturbations, the trajectory computed by each propagation method should closely approximate an exact Keplerian orbit. This implies that the orbital elements should stay virtually constant over the full integration period.

All propagation methods described earlier have been tested for this condition by deactivating all perturbing forces in the force model. For some of the methods – the method of variation of parameters and symplectic integration with Wisdom-Holman splitting, to be specific – this is a trivial affair, as their formulations are based on numerical integration of the perturbations and an exact solution for the two-body part. Consequently, the results of these methods will automatically correspond to the exact solution of the two-body problem.

For the other methods, the degree of conformity with the exact solution is dependent on the integration step size. For small step sizes, all methods have been verified to produce results that are very close to the ideal Keplerian orbit.

When performing integrations over long periods of time, however, it is generally undesirable to use exceedingly small time steps for the integration, as doing so will result in extensive computation times. Therefore, efficiency should also be taken into account when evaluating propagation methods. This important aspect will be treated in Chapter 9.

8.3 Positions of the Sun and the Moon

The positions of the Sun and the Moon are used for the computation of their respective third-body perturbations. In addition, the position of the Sun is used for computing the force due to solar radiation pressure.

As mentioned in Section 4.4, series expansions are used in the simulation code to compute the positions of the Sun and the Moon at any point in time. Details about the routines used for this can be found in Appendix B. In order to produce meaningful simulation results, it is important to verify that the series expansions have been implemented correctly and yield positions that correspond well to the physical situation. To check this, the computed positions are compared with the JPL Horizons ephemeris². At each point in time, the relative error in position is determined according to

$$\Delta r_{relative} = \frac{|\mathbf{r}_{JPL} - \mathbf{r}_{computed}|}{|\mathbf{r}_{JPL}|} \cdot 100\% \quad (8.1)$$

where \mathbf{r}_{JPL} is the position vector supplied by JPL Horizons and $\mathbf{r}_{computed}$ is the position vector computed by the simulation code.

Table 8.1 shows the relative errors for the position of the Sun computed at arbitrary³ points in time. As can be seen, the position error starts out very small and slowly grows as time increases. It should be noted that the entries in the lower part of the table correspond to epochs that are quite far away from the starting epoch of 01-01-2014. For example, 50,000 days is roughly 137 years, whereas 200,000 days amounts to about 548 years. Seeing that the position error after 200,000 days is only slightly more than 3 percent, it can be concluded that the series expansion used for the position of the Sun yields position errors that are well within reasonable limits for the time frame envisioned for the simulations.

²The JPL Horizons ephemeris is accessible at <http://ssd.jpl.nasa.gov/?horizons>.

³Though the time epochs in the table are precise values chosen as thousands of days after the initial epoch, the chosen values have no real relation to the movement of the Sun and are in that sense arbitrary.

Table 8.1: Relative position errors for the position of the Sun computed with the series expansion routine used in the simulation code for different points in time.

Time after Jan. 1, 2014		Relative position error
[days]	[years]	[%]
1,000	2.74	0.0852
2,000	5.48	0.1102
3,000	8.21	0.1202
4,000	10.95	0.1381
5,000	13.69	0.1582
10,000	27.38	0.2228
20,000	54.76	0.3961
30,000	82.14	0.5328
40,000	109.51	0.6930
50,000	136.89	0.8577
100,000	273.79	1.6547
200,000	547.57	3.1861

In a similar way, Table 8.2 shows the relative position errors for the Moon. Clearly, the position errors for the Moon are even smaller than for the Sun, only reaching a value of slightly more than 0.1 percent after 200,000 days. This makes sense, as the expressions used for the position of the Moon are quite a bit more complex than those used for the position of the Sun. It could be argued that some terms can be removed from the series expansions for the position of the Moon (Equations B.18-B.20), but careful examination of the expressions reveals that many of the smaller terms are of the same order of magnitude. This makes it hard to determine which terms can be removed without removing too many of the underlying dynamics. Furthermore, it is questionable whether it is desirable to deliberately reduce the accuracy of the force model. Therefore, it is decided to keep the expressions for the position of the Moon the same as in Appendix B.

Table 8.2: Relative position errors for the position of the Moon computed with the series expansion routine used in the simulation code for different points in time.

Time after Jan. 1, 2014		Relative position error
[days]	[years]	[%]
1,000	2.74	0.0573
2,000	5.48	0.0112
3,000	8.21	0.0538
4,000	10.95	0.0130
5,000	13.69	0.0224
10,000	27.38	0.0304
20,000	54.76	0.0856
30,000	82.14	0.0516
40,000	109.51	0.0850
50,000	136.89	0.0531
100,000	273.79	0.0517
200,000	547.57	0.1244

8.4 Individual Perturbations

In this section, it is checked whether the simulation results for the individual perturbing forces are in line with the physical effects that are to be expected in reality. It should be emphasized that in reality, the perturbing forces never work in isolation as is the case in these simulations. Rather, the forces are working in unison and the combined effect of the forces on the orbital elements is generally different from the sum of the individual changes to the orbital elements.

The initial conditions for the simulations performed in this section are those listed in Table 8.3, unless specified otherwise. The simulation time for each of the plots is five days. Since the orbital period is almost one day ($23^{\text{h}}56^{\text{m}}04^{\text{s}}$ to be exact), this corresponds to approximately five orbital revolutions.

Table 8.3: The initial conditions for the simulation results in this section.

Orbital element	Initial value
a	42,164 km
e	0
i	0°
ω	0°
Ω	0°
θ	0°
Starting epoch	
Jan. 1, 2014	00:00:00

8.4.1 J_2 -Term of the Earth's Gravity Field

Figure 8.1 shows the effect of the J_2 -term of the Earth's gravity field on the orbital elements a , e and i . Evidently, a and e experience a periodic variation with a period of approximately one day. Wakker (2010), however, states that the maximum value of $|\Delta a|$ is limited to

$$|\Delta a|_{max} = 3J_2 \frac{R^2}{r_0} \sin^2 i_0 \quad (8.2)$$

This implies that for a zero initial inclination there should be no change in the semi-major axis due to J_2 . Nevertheless, Figure 8.1 shows a maximum deviation in a on the order of $2 \cdot 10^{-4}$ km. Though this is not a drastic deviation, it is still quite noticeable.

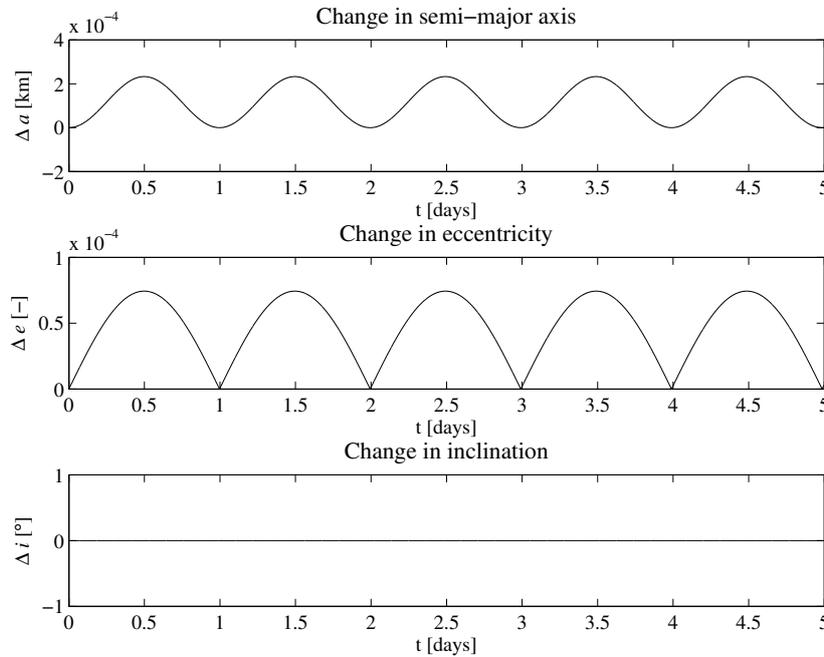


Figure 8.1: The effect of the J_2 -term of the Earth's gravity field on the orbital elements.

The appearance of the deviation in the semi-major axis is a result of the way the simulation code is set up. In the code, the initial orbital elements are specified at the beginning. From these elements the Cartesian state vector at the start of the simulation is computed. In this conversion, the gravitational parameter μ of the Earth is used. Without any perturbing forces, propagation of the state vector would result in an orbit that maintains the same orbital elements as specified in the initial conditions. However, with the J_2 -effect some mass is effectively added at the equator. Hence, for an equatorial orbit the gravitational pull is slightly stronger

than before. Consequently, a higher initial velocity is required to keep a circular orbit. In other words, the initial Cartesian state vector computed using μ no longer leads to a circular orbit with the orbital elements as specified, in case the J_2 -effect is taken into account.

To counteract this effect, the gravitational parameter that is used for computing the initial state vector can be slightly adjusted to include the effect of the J_2 -term. From the expressions in Section 4.3 it can be derived that the following expression can be used to achieve this:

$$\mu_{J_2\text{corrected}} = \mu \left(1 + \frac{3}{2} J_2 \frac{R_{\oplus}^2}{r^2} \right) \quad (8.3)$$

In case the gravitational parameter of Equation 8.3 is used for the computation of the initial state vector, the orbital elements behave as shown in Figure 8.2. Clearly, the deviations in semi-major axis and eccentricity are both negligibly small in this case, while the deviation in inclination is also still zero. Hence, for an equatorial orbit the J_2 -effect can be fully compensated for by slightly changing the initial state vector.

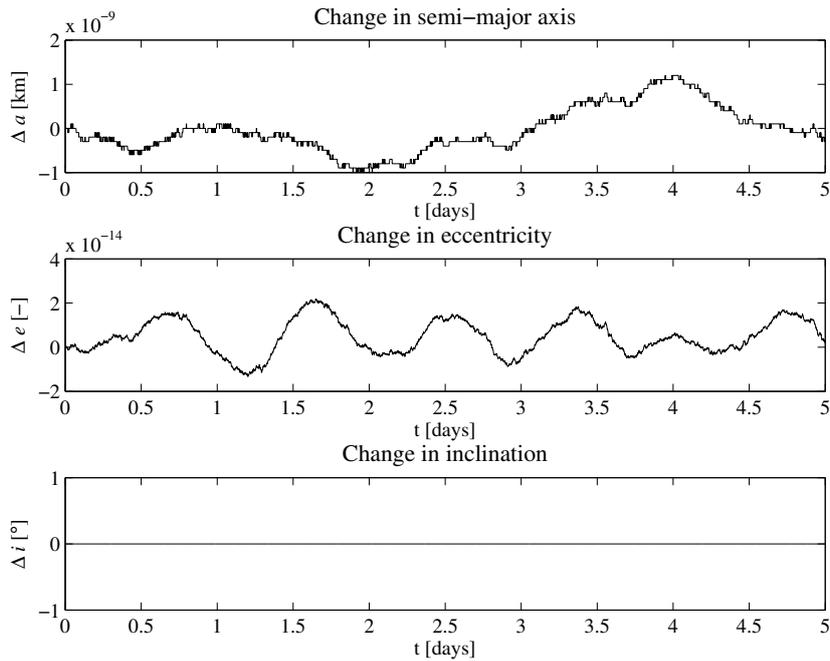


Figure 8.2: The effect of the J_2 -term of the Earth's gravity field on the orbital elements. For this plot, the initial state vector has been adjusted to account for the J_2 -effect.

For non-equatorial orbits, the J_2 -effect will lead to non-zero deviations in the orbital elements. Nevertheless, it will not result in secular or long-period changes, according to Wakker (2010). The maximum deviation in semi-major axis can be

computed with Equation 8.2, whereas the maximum change in inclination can be calculated using

$$|\Delta i|_{max} = \frac{3}{4} J_2 \frac{R^2}{r_0^2} \sin 2i_0 \quad (8.4)$$

If the initial conditions of Table 8.3 are used, except for the inclination, which is set at $i = 50^\circ$, the following maximum values are obtained for the semi-major axis and eccentricity using Equations 8.2 and 8.4:

$$|\Delta a|_{max} = 1.84 \text{ km}$$

$$|\Delta i|_{max} = 1.05 \cdot 10^{-3} \text{ }^\circ$$

The results of a simulation with the same initial values are shown in Figure 8.3. As can be seen, the semi-major axis and inclination oscillate with a period of roughly half a day, which corresponds to half the orbital period. Furthermore, the maximum deviations appear to be similar to the values listed above. A closer examination of the simulation output file has revealed that the maximum values are indeed identical to the computed values.

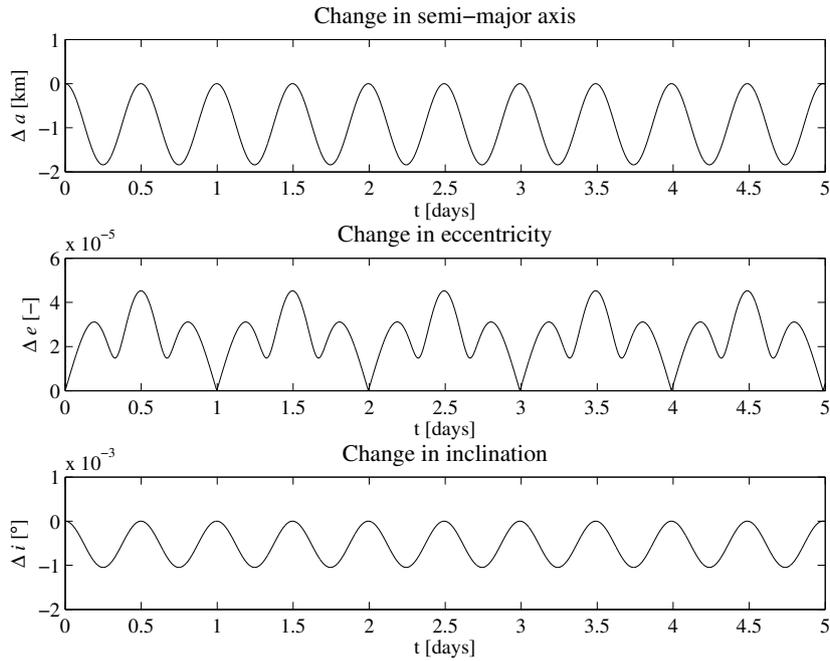


Figure 8.3: The effect of the J_2 -term of the Earth's gravity field on the orbital elements if the inclination has an initial value of 50° .

In addition to the effects described, J_2 also has an important effect on the right ascension of the ascending node Ω . In contrast to the effects described earlier, this concerns a secular variation, called the *regression of the nodes* (Wakker, 2010).

The corresponding change in Ω after one orbital revolution is given by

$$\Delta_{2\pi}\Omega = -3\pi J_2 \frac{R^2}{r_0^2} \cos i_0 \quad (8.5)$$

For a geostationary orbit, this comes down to a shift in longitude⁴ of $-48.2''$ per day. Though the effect is not visible in the plots shown in this section, its consequences will become clear in Section 8.5.

Other terms of the geopotential could also be included in this chapter. However, J_2 is by far the largest effect, overshadowing the effects of the other terms. Since there are so many terms, it does not seem worthwhile to individually verify the working of every term of the spherical harmonics data. In addition, there is almost no data available to perform said verification. Therefore, the verification of the other parts of the geopotential has been limited to carefully checking the expressions in the simulation code.

8.4.2 Gravitational Attraction of the Sun

The effect of the gravitational attraction of the Sun on the orbital elements is shown in Figure 8.4. The dynamics can perhaps be best understood by considering Figure 4.3 again. Essentially, an orbital revolution in this figure can be divided into four quarters. In two of these quarters, the perturbing mass (in this case, the Sun) is pulling the orbiting object in the direction of motion, while in the other two quarters, the perturbing mass is pulling in the direction opposite to the orbital motion. These two effects alternate, leading to a period in which the orbital velocity is increased, followed by a period in which the velocity is decreased, then a period in which the velocity is increased again, and so on. This leads to a periodic change in the semi-major axis of the orbit with two peaks per orbital revolution, *if* the perturbing body does not move during the orbital motion. In the case of the Sun, the coordinates of the perturbing body change only very little during one orbital revolution, and hence, the variation of the semi-major axis in Figure 8.4 has a period that is very close to half of the orbital period of one day.

The fact that the value of Δa is almost exclusively positive in the plot is a result of the initial conditions of the simulation. If the simulation is initiated with a different starting epoch, the mean value of the oscillation will shift, though the amplitude of the oscillation will remain roughly the same. The reason for this is that the plot shows the difference in a with the initial value of 42,164 km. If, due to the initial geometry, the Sun starts with pulling against the orbital motion and hence decreasing the orbital velocity, a will decrease first, leading to a negative value of Δa in at least the first part of the plot. If, on the other hand, the initial geometry is such that the Sun starts with increasing the orbital velocity, Δa will be positive in the first part of the plot. Under certain conditions, a will first increase for the maximum amount, after which it will decrease again by the same amount.

⁴As a result of its zero inclination, Ω is undefined for a geostationary orbit.

In this case, Δa will only be positive. This situation occurs almost completely in the first plot of Figure 8.4.

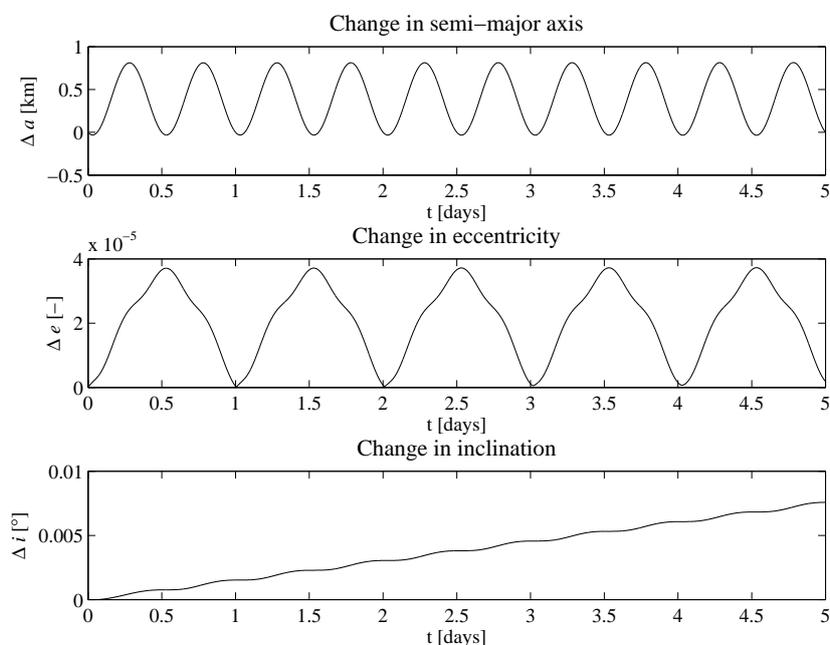


Figure 8.4: The effect of the gravity of the Sun on the orbital elements.

The simulation time of the plots in Figure 8.4 is only five days, and therefore only the short-period effects can be observed; long-period effects and potential secular effects cannot be checked using these plots. However, these aspects are implicitly included in Section 8.5, where the working of the force model is verified on longer timescales.

8.4.3 Gravitational Attraction of the Moon

Figure 8.5 shows the effect of the gravitational attraction of the Moon on the orbital elements. The situation is similar to the one pertaining to the gravitational attraction of the Sun, described in the previous section. It should be noted that the maximum deviation in the semi-major axis in the case of the Moon is about 2.4 km, whereas it is around 0.8 km for the Sun. Hence, the gravity of Moon clearly has a larger effect on the orbit than the gravity of the Sun. Following Wakker (2010), this was to be expected. Additionally, the period of the oscillation in the semi-major axis seems to be slightly, but noticeably more than half a day. The reason for this is likely that the position of the Moon changes notably during a single orbital revolution. This is also presumed to be the cause of the fact that two subsequent oscillations do not have the same amplitude. A subsequent check in which the motion of the Moon was deactivated has confirmed that these assertions

are correct.

Again, long-term effects cannot be verified based on Figure 8.5. These aspects will be addressed in Section 8.5, however, where the full force model is verified over a long period of time.

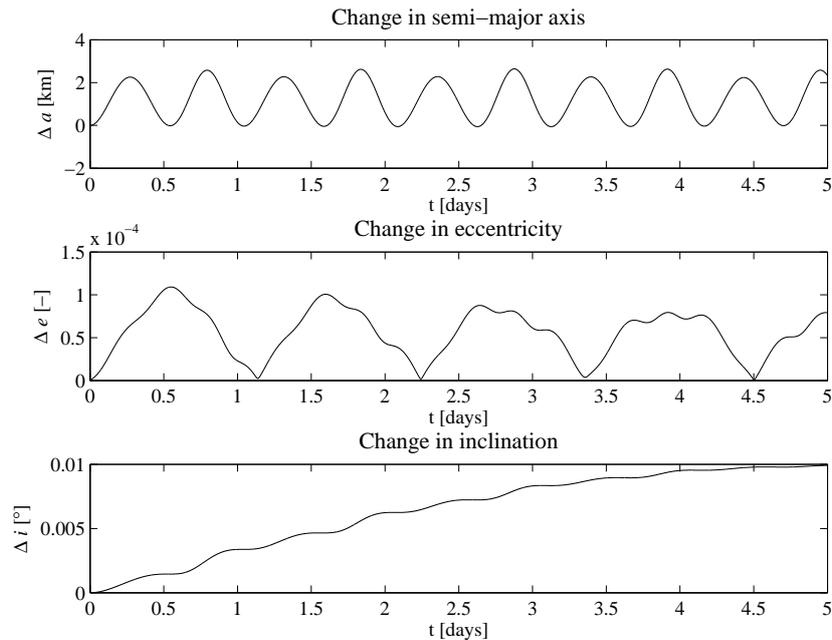


Figure 8.5: The effect of the gravity of the Moon on the orbital elements.

8.4.4 Solar Radiation Pressure

The effect of the perturbing force due to solar radiation pressure on an orbit with the initial conditions listed in Table 8.3 is shown in Figure 8.6. The area-to-mass ratio has a value of $0.01 \text{ m}^2/\text{kg}$ in this case. Clearly the semi-major axis shows a variation with a period equal to roughly one day, corresponding to one orbital period. This makes sense, as the force due to solar radiation pressure has a component which effectively pushes in the direction of the orbital motion for half of the orbit and works against the orbital motion for the other half, thereby increasing and decreasing the semi-major axis.

The inclination also shows a periodic variation with a period of approximately one day. Note that in this case Δi is only positive, because the initial inclination is zero and the inclination itself is defined such that it can only have positive values.

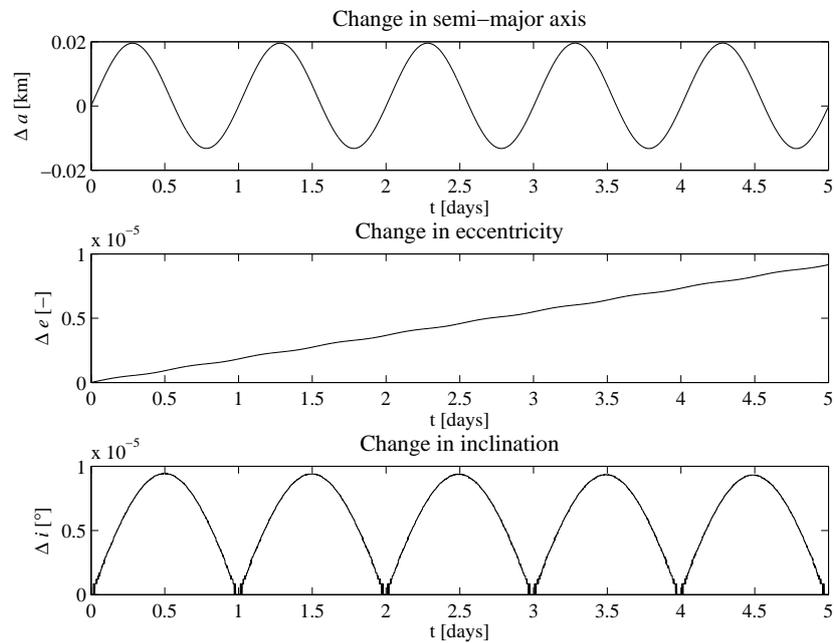


Figure 8.6: The effect of solar radiation pressure on the orbital elements for an area-to-mass ratio of $0.01 \text{ m}^2/\text{kg}$.

If the area-to-mass ratio is changed to $20 \text{ m}^2/\text{kg}$, while all other conditions remain the same, the results of Figure 8.7 are obtained. As can be seen, the results are qualitatively the same as before, but the amplitude of the variations has increased substantially. In fact, the area-to-mass ratio has increased by a factor of 2,000 and the magnitude of the deviations in a , e and i has increased by the same factor. This seems reasonable, as the force due to solar radiation pressure scales linearly with the area-to-mass ratio A/m (cf. Equation 4.50).

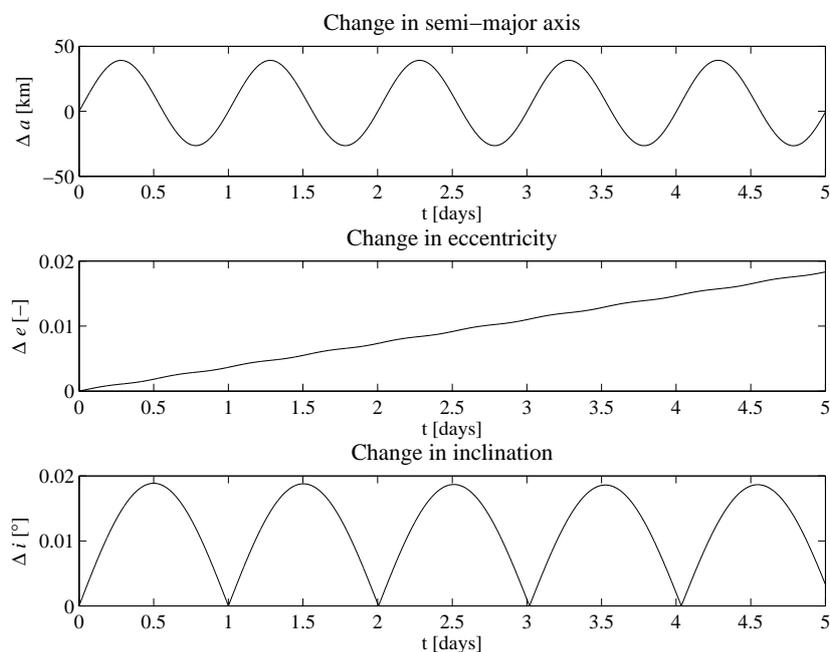


Figure 8.7: The effect of solar radiation pressure on the orbital elements for an area-to-mass ratio of $20 \text{ m}^2/\text{kg}$.

Note that there are no effects of eclipses visible in Figures 8.6 and 8.7. A subsequent check of the simulation has confirmed that there are indeed no eclipses taking place during the five-day period following the initial conditions of Table 8.3. Some simple geometric calculations reveal that due to its relatively large distance from the Earth, a geostationary satellite will only experience an eclipse during an orbital revolution if the angle the solar vector makes with the equatorial plane is smaller than 8.7° . Even in the most extreme case, with the solar vector being within the equatorial plane, a geostationary satellite will only be in eclipse for 4.8% of its orbital period. Hence, the effects of eclipses on a geostationary orbit are quite limited.

In order to verify the working of the eclipse routine, a simulation has been performed for a LEO satellite with an orbital altitude of 500 km. A lower altitude has been chosen in this case as the relative duration of the eclipse period is longer for low altitudes. Additionally, all orbital elements apart from the semi-major axis have been chosen to be equal to zero, resulting in a circular, equatorial orbit. To maximize the eclipse duration, the position of the Sun has been fixed in the equatorial plane. The change of the semi-major axis corresponding to the simulation is shown in Figure 8.8. For comparison, the same quantity is also plotted for a simulation in which eclipses have not been taken into account (see Figure 8.9).

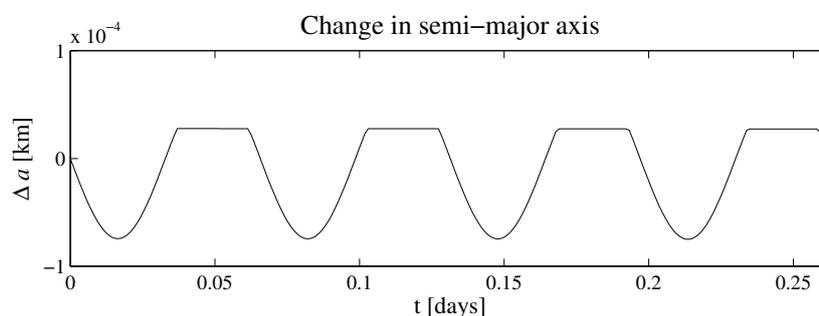


Figure 8.8: The effect of eclipses on the semi-major axis variation due to solar radiation pressure, for a satellite with an orbital altitude of 500 km and an area-to-mass ratio of $0.01 \text{ m}^2/\text{kg}$.

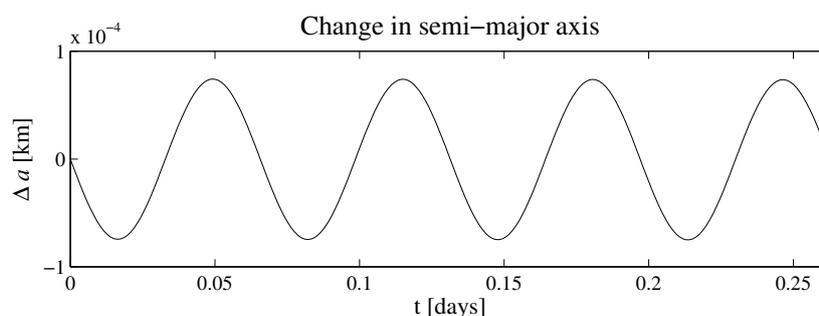


Figure 8.9: The semi-major axis variation for the same orbit as in Figure 8.8, without taking eclipses into account.

As can be seen in the figures, eclipses have the effect of "cutting off" the peaks of the oscillation in the semi-major axis. The reason for this is that the perturbing force due to solar radiation pressure simply stops acting on the satellite once it enters eclipse, immediately preventing further changes to the semi-major axis. Once the satellite leaves eclipse, the semi-major axis starts changing again. It should be noted that due to the fact that the shadow of the Earth always extends along the line opposite to the solar vector, the spacecraft always enters eclipse conditions after being accelerated by solar radiation and is always decelerated again after exiting eclipse. Hence, irrespective of the specifics of the orbit, an eclipse will always have the effect of cutting off the peak of the semi-major axis variation in the way shown in Figure 8.8.

A simple geometric calculation reveals that under the conditions of the simulation, the satellite should be in eclipse for approximately 38% of its orbital period. Inspection of Figure 8.8 shows that in the simulation, eclipses indeed have a duration of roughly $\frac{4}{10}$ of the orbital period, confirming that the eclipse routine has been implemented correctly.

8.5 Long-Term Behavior

Although the correct functioning of the individual perturbing forces has been verified in Section 8.4, it is also important to check whether the full force model results in trajectories that are accurate representations of reality. Therefore, the long-term behavior predicted by the simulation code will be assessed in this section.

According to Wertz (2009), the largest perturbation in GEO is a north-south drift or inclination change caused by the gravitational interaction with the Sun and the Moon. If not corrected, this perturbation will cause the inclination to vary between 0 and 15 degrees over a period of approximately 55 years.

Figure 8.10 shows the change in inclination during a 100-year simulation of a satellite which starts out in a nominal geostationary orbit. As can be seen, the inclination shows a periodic variation with values between 0 and roughly 15 degrees, with a period of approximately 53 years. This corresponds well with the descriptions from Wertz and other authors (Ikeda et al., 2008), (Matney, 2004), who also mention a maximum value of about 15 degrees and periods between 50 and 54 years.

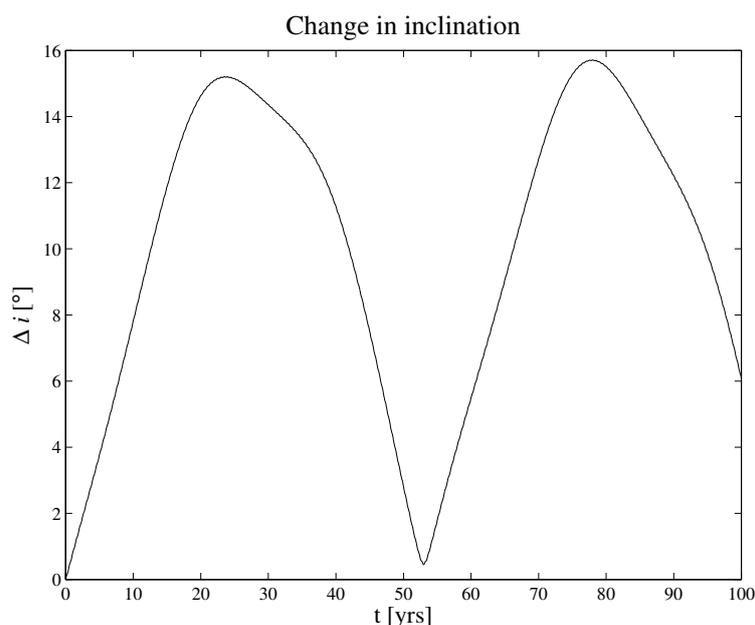


Figure 8.10: The change in inclination over time for a simulation of a satellite with $C_R \cdot A/m = 0.01 \text{ m}^2/\text{kg}$ departing from a nominal geostationary orbit.

For a more complete check of the long-term behavior predicted by the simulation code, it would be meaningful to compare simulation results with data from published research. However, even though there are a number of research papers available in which situations have been investigated which are similar to the ones that are of interest for this thesis project, most authors do not include the exact

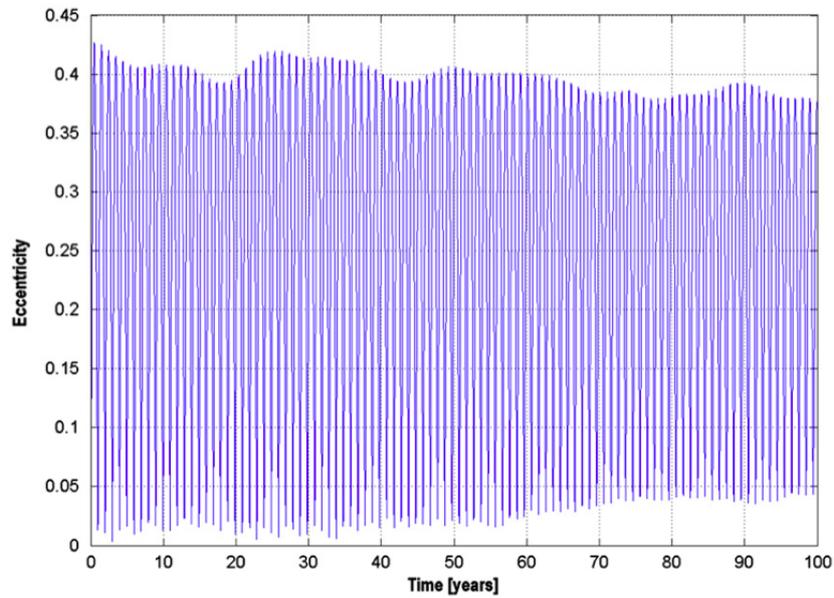
initial conditions used, making it very hard to reproduce the data.

Fortunately, Anselmo and Pardini (2010) have published a paper in which the long-term orbit developments of objects in GEO and GPS orbits are simulated, and have stated all initial conditions used for the simulations. The results from this paper will be used as test cases for verifying the simulation code.

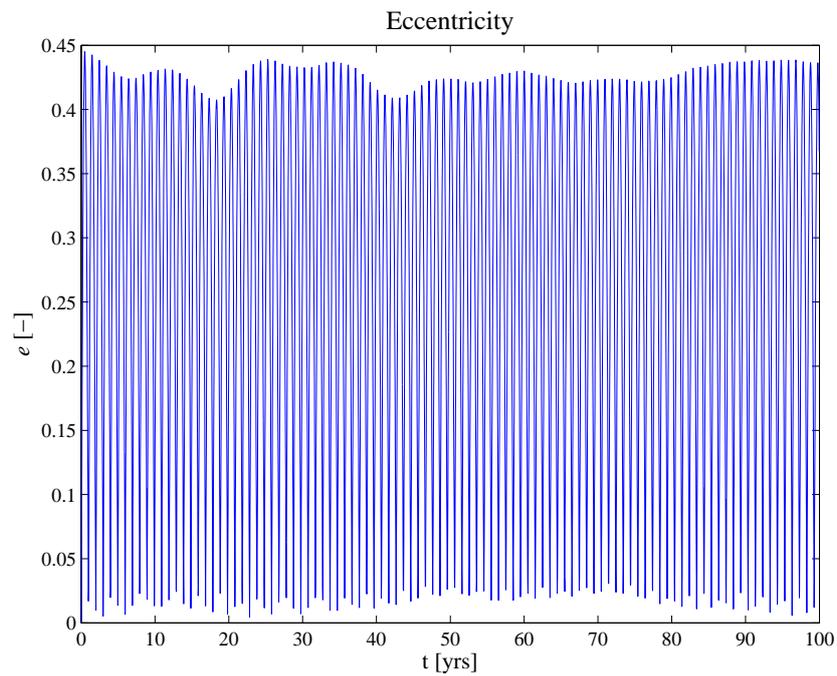
The initial conditions for the GEO test case used by Anselmo and Pardini are listed in Table 8.4. The object whose orbit was propagated was given a very high effective area-to-mass ratio of $20.4 \text{ m}^2/\text{kg}$. The results regarding the long-term evolution of eccentricity and inclination, as predicted both by Anselmo and Pardini and the simulation tool developed for this thesis project, are shown in Figures 8.11 and 8.12, respectively.

Table 8.4: The initial conditions for the GEO test case, as specified by Anselmo and Pardini (2010).

Starting epoch	
Dec. 22, 2005 00:00 UTC	
Orbital element	Initial value
a	42,164.465 km
e	0.0001
i	0.097°
ω	220.00°
Ω	50.00°
θ	301.22°
Physical property	Value
$C_R \cdot A/m$	$20.4 \text{ m}^2/\text{kg}$

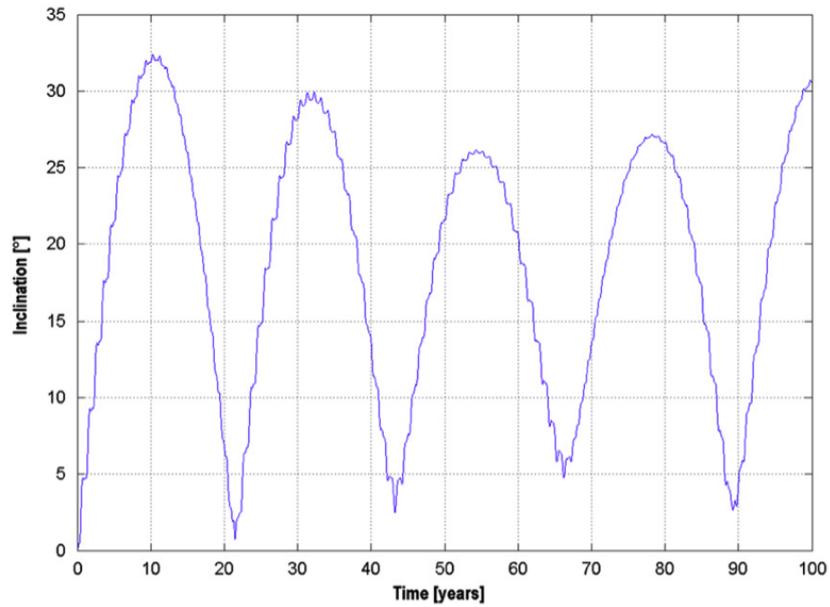


(a) Results from a numerical simulation carried out by Anselmo and Pardini (2010).

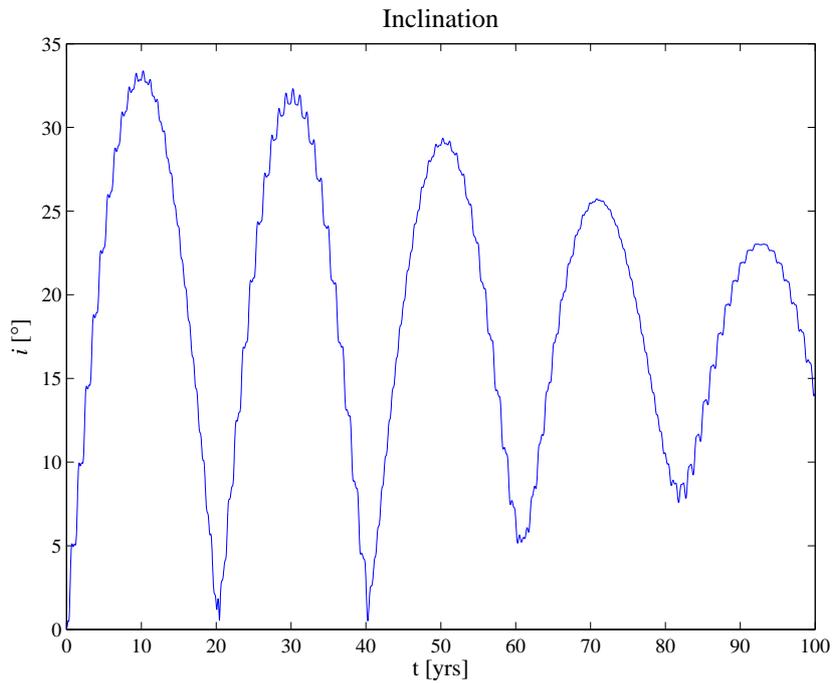


(b) Results produced using the simulation code developed for this thesis project.

Figure 8.11: The long-term evolution of the eccentricity for an object with $C_R \cdot A/m = 20.4 \text{ m}^2/\text{kg}$ released from a GEO orbit with the initial conditions of Table 8.4.



(a) Results from a numerical simulation carried out by Anselmo and Pardini (2010).



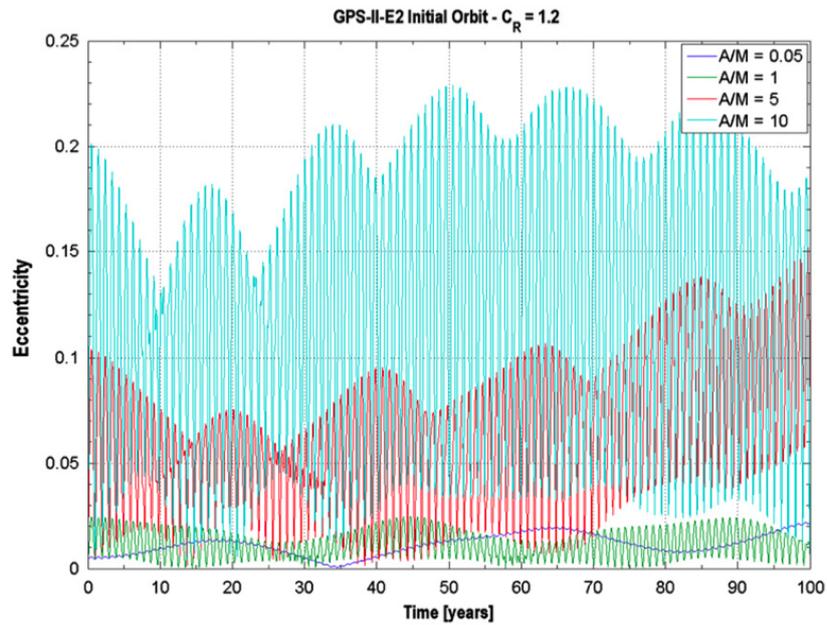
(b) Results produced using the simulation code developed for this thesis project.

Figure 8.12: The long-term evolution of the inclination for an object with $C_R \cdot A/m = 20.4 \text{ m}^2/\text{kg}$ released from a GEO orbit with the initial conditions of Table 8.4.

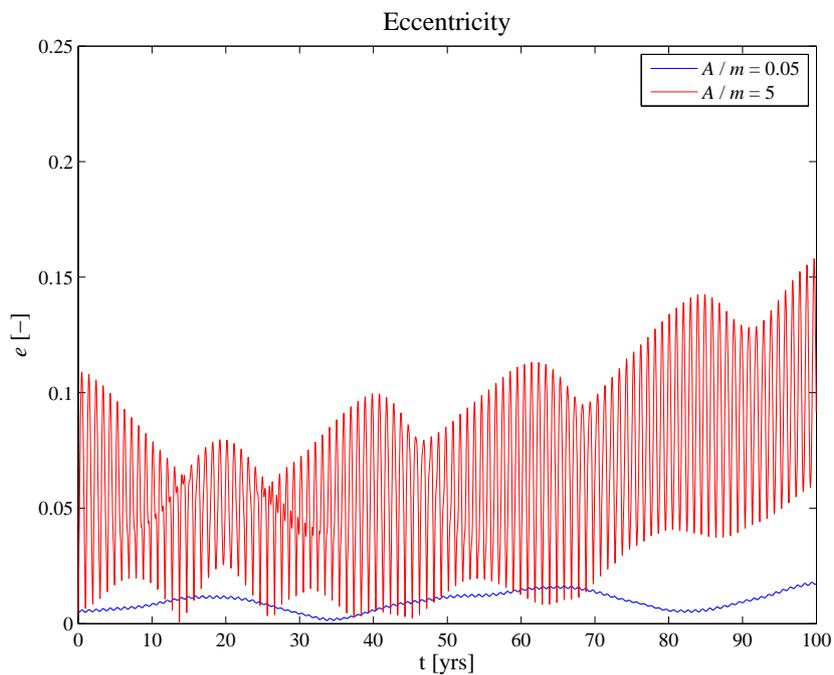
Similarly, the initial conditions used for the GPS test case are listed in Table 8.5. Anselmo and Pardini actually performed simulations with these starting conditions for objects with four different area-to-mass ratios. For the purpose of keeping the plots clear, only two area-to-mass ratios have been used for the simulations performed with the simulation code created for this project. The plots for the eccentricity and inclination development corresponding to the GPS test case can be found in Figures 8.13 and 8.14.

Table 8.5: The initial conditions for the GPS test case, as specified by Anselmo and Pardini (2010).

Starting epoch	
Apr. 16, 2007 17:29 UTC	
Orbital element	Initial value
a	26,560.432 km
e	0.0048672
i	54.5093°
ω	265.1898°
Ω	312.7360°
θ	94.2809°
Physical property	Values
$C_R \cdot A/m$	0.05 m ² /kg, 5 m ² /kg

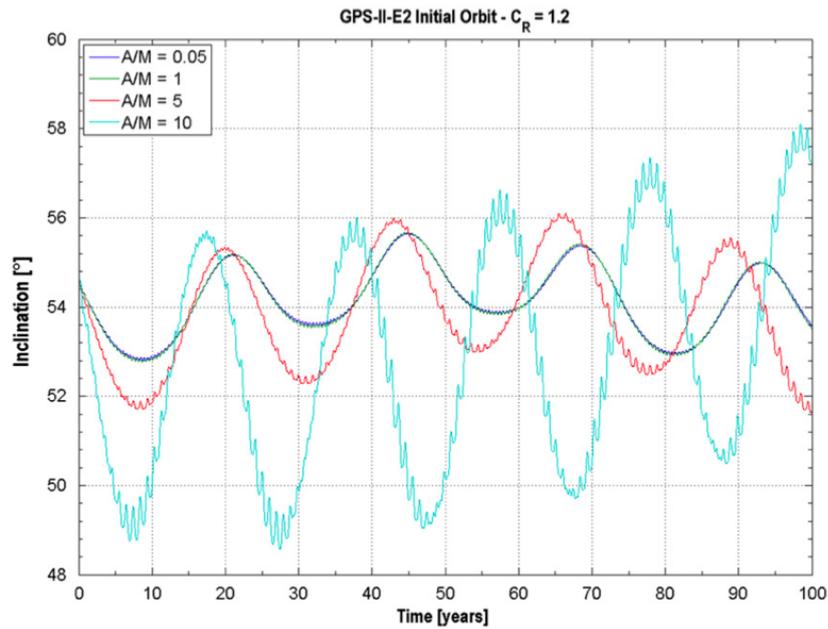


(a) Results from numerical simulations carried out by Anselmo and Pardini (2010).

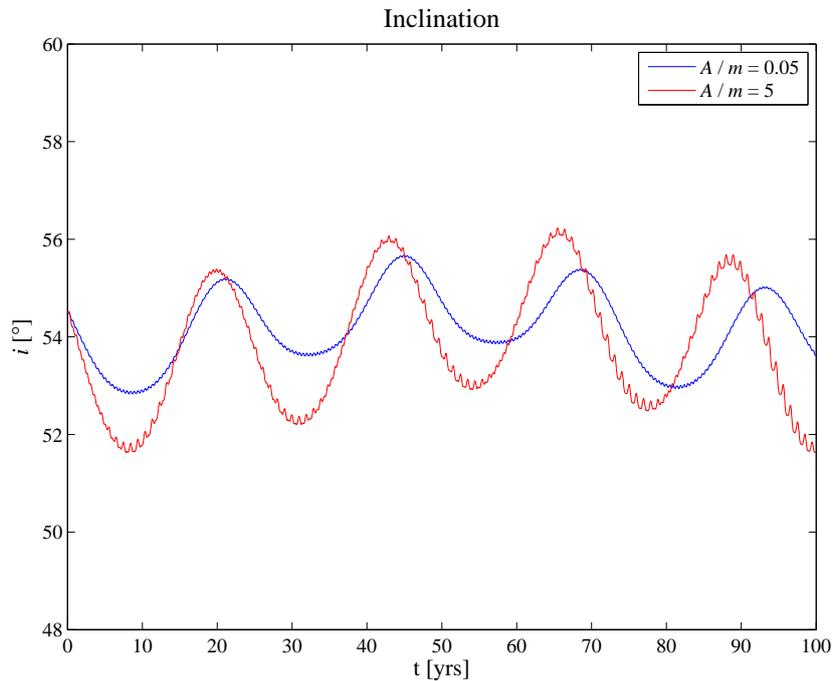


(b) Results produced using the simulation code developed for this thesis project.

Figure 8.13: The long-term evolution of the eccentricity for objects with different area-to-mass ratios released from a GPS orbit with the initial conditions of Table 8.5.



(a) Results from numerical simulations carried out by Anselmo and Pardini (2010).



(b) Results produced using the simulation code developed for this thesis project.

Figure 8.14: The long-term evolution of the inclination for objects with different area-to-mass ratios released from a GPS orbit with the initial conditions of Table 8.5.

As can be seen in Figures 8.11 and 8.12, both simulations for the GEO case predict the same general behavior, with some noticeable discrepancies that seem to increase over time. Unfortunately, not much information is given in the paper by Anselmo and Pardini (2010) about the way the forces were modeled. Any small differences in the force model may have an influence on the future trajectory. In addition, it should be kept in mind that the area-to-mass ratio for these particular simulations is very high, amplifying the effect of any discrepancies related to the modeling of solar radiation pressure. As a result, the cumulative effect may become quite noticeable over long integration periods. Because little information about the modeling techniques used for the reference simulations is available, there is no practical way of finding out what causes the differences. Most importantly, the differences are small enough that the conclusions from both simulations regarding the dangers related to such a debris object will generally be the same.

A more exact correspondence with the simulations by Anselmo and Pardini is found for the simulations for the GPS case, in which somewhat lower area-to-mass ratios have been used (see Figures 8.13 and 8.14). Almost no differences can be observed between the plotted data, while the simulation code for this thesis project has been created completely independently from the research by Anselmo and Pardini. This indicates that the simulation code is working correctly when propagating orbits over long periods of time with the full force model activated.

Now that the correct working of the simulation code has been verified using a range of different simulation cases, it is time to utilize the simulation code for the applications for which it was created, namely for testing the performance of different computational methods in predicting the long-term evolution of space debris orbits (Chapter 9) and interpreting the implications of such predictions for a number of specified graveyard orbits (Chapter 10).

Chapter 9

Performance Comparison of Computational Methods

The performance of the different computational methods¹ introduced earlier in this report for carrying out long-term simulations of space debris orbits is assessed in this chapter. The simulations need to be accurate, such that the predicted trajectories are actually representative of reality. In addition, the simulations must be carried out efficiently, as it is not feasible to conduct a large amount of simulation runs if each run takes an excessively long time to complete.

It will be interesting to quantify how much of a difference it makes to choose a suitable method for this specific application. Moreover, the efficiency of symplectic integrators for integrating space debris orbits is still largely unknown. The performance plots in this chapter will show how they perform compared to more traditional methods. The results of this chapter will also be convenient for conducting the long-term debris simulations of the next chapter.

This chapter is structured as follows. First, an overview is given of the different computational methods that are compared. Following that, the methodology that is used for the performance tests is discussed. Next, the performance for the two-body problem is considered, which also serves to showcase a number of important concepts. Then, the performance plots for multiple relevant test cases of long-term space debris simulations are presented. Finally, conclusions are drawn regarding the efficiency of all computational methods considered in this chapter and their suitability for carrying out long-term propagations of space debris orbits.

¹In this context, the term *computational methods* is used to refer to combinations of both integration methods and formulations of the equations of motion / Hamiltonian splitting methods.

9.1 Overview

In this chapter, the performance of the different computational methods introduced in the previous chapters is compared for a range of simulation cases. For clarity, a short overview of the methods that are compared is provided in this section, along with references to the parts of the report in which the methods were treated.

The computational methods that are compared are listed below. The part in brackets at the end of each entry shows the abbreviated notation that is used in the plots shown in the subsequent sections of this chapter.

- Runge-Kutta 4 with Cartesian components (RK4)
- Runge-Kutta 4 with modified equinoctial elements (RK4 MEE)
- Runge-Kutta-Fehlberg 5(6) with Cartesian components (RKF56)
- Runge-Kutta-Fehlberg 5(6) with modified equinoctial elements (RKF56 MEE)
- Dormand-Prince 8(7) with Cartesian components (DOPRI8)
- Dormand-Prince 8(7) with modified equinoctial elements (DOPRI8 MEE)
- Kinoshita's method with T+V splitting (Kinoshita)
- Kinoshita's method with Wisdom-Holman splitting (Kinoshita WH)
- SABA₆ with T+V splitting (SABA6)
- SABA₆ with Wisdom-Holman splitting (SABA6 WH)
- SABA₈ with T+V splitting (SABA8)
- SABA₈ with Wisdom-Holman splitting (SABA8 WH)
- SABA₁₀ with T+V splitting (SABA10)
- SABA₁₀ with Wisdom-Holman splitting (SABA10 WH)

In the items listed above, Cartesian components implies that Cowell's method is used, i.e. direct integration of the equations of motion formulated in Cartesian components (see Section 5.3.1). Furthermore, modified equinoctial elements means that Gauss' form of Lagrange's planetary equations is used, with modified equinoctial elements as element set in order to prevent singularities (see Section 5.3.3).

Continuing with the traditional integration methods, RK4 is the relatively simple and commonly used fixed step size integrator described in Section 6.3.1, whereas RKF56 and DOPRI8 are the more advanced adaptive step size methods treated in Sections 6.4.2 and 6.4.3, respectively.

Kinoshita's method and the $SABA_n$ methods are all symplectic integration methods, described in Sections 7.4.3 and 7.4.4, respectively. These symplectic methods are combined with either of the two splitting methods discussed in this report.

In short, T+V splitting implies splitting the Hamiltonian in a part consisting of all kinetic energy terms and a part consisting of all potential energy terms. Both parts are advanced sequentially during the integration step (see Section 7.5.1). Wisdom-Holman splitting, on the other hand, splits the Hamiltonian in a two-body part and a perturbations part. In the implementation of the simulation code, the integration is set up such that the two-body part is advanced by solving Kepler's equation, while the perturbations part is integrated numerically. Again, the advancement of the two parts is done sequentially using symplectic schemes (see Section 7.5.2).

Finally, a distinction is made in this chapter between simulations of the two-body problem and simulations of the perturbed two-body problem. In short, the two-body problem implies that only the central gravitational force has been taken into account, whereas the term perturbed problem indicates simulations in which the full force model described in Chapter 4 has been activated. Hence, these simulations include the geopotential force, luni-solar perturbations and the force due to solar radiation pressure (including eclipses), in addition to the central gravitational force.

9.2 Testing Methodology

The main goal of the performance tests conducted in this chapter is to assess how efficient different computational methods are at performing simulations of the long-term evolution of space debris orbits. The application of such simulations is to determine whether the orbits eventually develop in such a way that a collision risk with active satellites is created. Because of this application, errors in the *geometry* of the orbit are more important than errors regarding the exact *position* of the object *in* the orbit. Consequently, measuring the vector difference between a computed position and a reference position might lead to wrong conclusions, as an error in the position of the object within the orbit could lead to a large error, while the geometry of the computed orbit could in fact be very close to that of the actual orbit.

Therefore, the errors in semi-major axis, eccentricity and inclination are considered instead of the vector differences in position. The errors in the other orbital elements are not included in the performance comparison, because the remaining orbital elements have much less geometrical meaning for general predictions of potential collision hazards, and including them in the comparison would significantly increase the number of plots required.

When assessing the accuracy of the computational methods, the *maximum* errors in the orbital elements are used, as not only the final state of the simulation

is important, but also the trajectory before it. The maximum error gives an indication of the minimum accuracy that can be expected over the full simulation period. Another option, sometimes seen in literature, is to plot the differences of the orbital elements with those of the reference simulation as a function of time. Though this would provide somewhat more information, it is not considered feasible here, as the relatively large number of methods that are compared would require a vast amount of plots. Choosing to compare the maximum errors allows the performances of multiple methods to be combined in single plots, which saves a considerable amount of space and also facilitates an easier comparison between the different methods.

In the plots, the required computation time is the parameter that is shown together with the accuracy of each run. Some comparisons in literature use the step size or the number of function evaluations as a parameter to display. These parameters have the advantage that they do not depend on the computer setup that is used. However, they are not practical to use for the set of methods that are compared in this chapter. First of all, the step size can only be used as a parameter for fixed step size methods, whereas this comparison also includes variable step size methods. Moreover, both the step size and the number of function evaluations do not provide insight in the overhead costs in terms of computation time that different formulations have. For example, when using Gauss' equations with modified equinoctial elements, multiple conversions from modified equinoctial elements to Cartesian coordinates are required per integration step for evaluation of the force model. In addition to that, the Cartesian components of the force need to be converted to radial, tangential and normal components each time the force model is used. The Wisdom-Holman splitting method for symplectic integrators has similar requirements. To take all these aspects into account, the actual computation time required for performing the integration seems to be the best measure to use for assessing the efficiency of different methods.

In practice, the performance comparison has been performed as follows. First of all, a simulation run with specified initial conditions and integration settings is performed. During the integration, the state is saved at specific time intervals. After the integration has been completed, the computed orbital elements at these data points are compared with those of the reference data set. Once all errors with the reference data have been computed, the maximum errors in a , e and i are determined. These maximum errors are shown in performance plots, which show the maximum error in a specific orbital element versus the computation time required to perform the integration over the full simulation period. Hence, each data point in the performance plots represents a single 100-year simulation run with specific integration settings.

The computation time varies based on the simulation settings. In the plots, moving along the horizontal axis in the direction of increasing computation time implies decreasing the step size for fixed step size methods or choosing more stringent relative accuracy requirements for adaptive step size methods. The computation time shown in the performance plots is the time required for the actual integra-

tion process, and does not include input/output and post-processing of the results. The simulations were carried out on a system with a 3.4 GHz Intel Core i7 3770 CPU and 8 GB of RAM, with no other processes running in parallel.

Of course, the exact computation time will vary significantly based on the computer setup that is used, but the general pattern of how the different methods perform relative to each other will be the same regardless of the computer setup.

Regarding the performance plots, it is important to note that a double logarithmic scale has been used, in order to display data with a large range of magnitudes, both in the sense of computation time as in terms of accuracy. As a result, a small horizontal distance between two curves can easily imply a difference in efficiency of a factor two or more.

Finally, it should be mentioned that the simulations are quite specific and that changes in the dynamics of the system can significantly alter the relative performance of the methods. If, for example, other forces are added to the force model or another orbital regime is modeled (e.g. LEO), the results from this chapter do not necessarily translate to the new situation. Therefore, it should be kept in mind that the conclusions following from this chapter are mainly applicable to simulations that have similar characteristics as the simulations presented here.

9.3 Two-Body Problem

The two-body problem is a convenient test case for assessing the performance of different computational methods. Because there are no perturbations, the solution to the equations of motion is an exact Kepler orbit, with orbital elements that remain constant in time. As a result of this, the error at each data point can be computed exactly. Although this is very convenient, it should be noted that the performance of different methods for the two-body problem does not necessarily translate to the simulations of the perturbed two-body problem, which are the simulations of interest for this thesis project.

Nevertheless, the two-body problem is useful for illustrating a number of important concepts, such as round-off errors and error growth. In addition, the two-body problem forms an interesting test case for symplectic integrators, as it involves the integration of a completely Hamiltonian system: the exact situation for which symplectic integrators were originally designed. It will be interesting to see how the performance of symplectic integrators compares between the simulations of the two-body problem and the simulations of the perturbed two-body problem.

9.3.1 Round-Off Errors

In general, the accuracy of an integration can be increased by using a smaller step size, since the global truncation error will decrease as a result. However, there is a limit to this. Using smaller steps implies that the total number of steps required for the simulation increases. At each step, the state of the system is stored with a finite number of digits, inherent to the way in which computers store numbers. Because

the number of digits is limited, a small round-off error is introduced which adds to the total integration error.

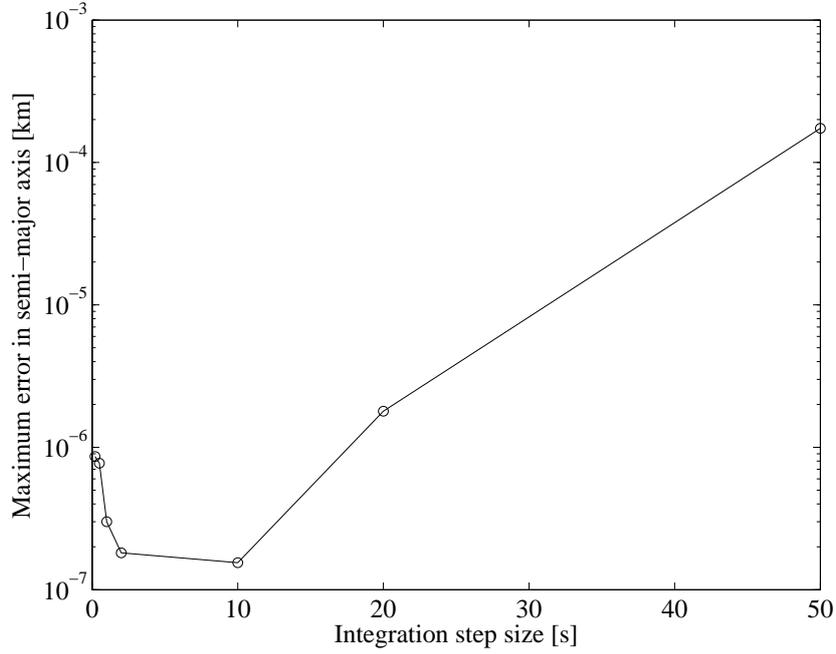
Although round-off errors are essentially random, they do accumulate over time. As a result, the contribution of round-off errors to the total integration error increases as smaller time steps are used. At a certain point, the accuracy gained by further decreasing the step size will become smaller than the increase in the total round-off error. Hence, though the computation time will increase substantially, the accuracy actually becomes worse.

Figure 9.1 shows the effect of round-off errors when performing a 100-year simulation of a nominal geostationary orbit with the RK4 method. As can be seen in the plots, the overall error decreases when the step size is decreased from 50 s to 10 s, but when further decreasing the step size, the error increases instead. The required computation times corresponding to the data points in the plots are shown in Table 9.1. Based on this, it can be observed that for this particular example, an integration with RK4 that takes less than 10 minutes to carry out results in a smaller maximum error over the 100-year integration period than an integration that takes more than 7 hours to complete.

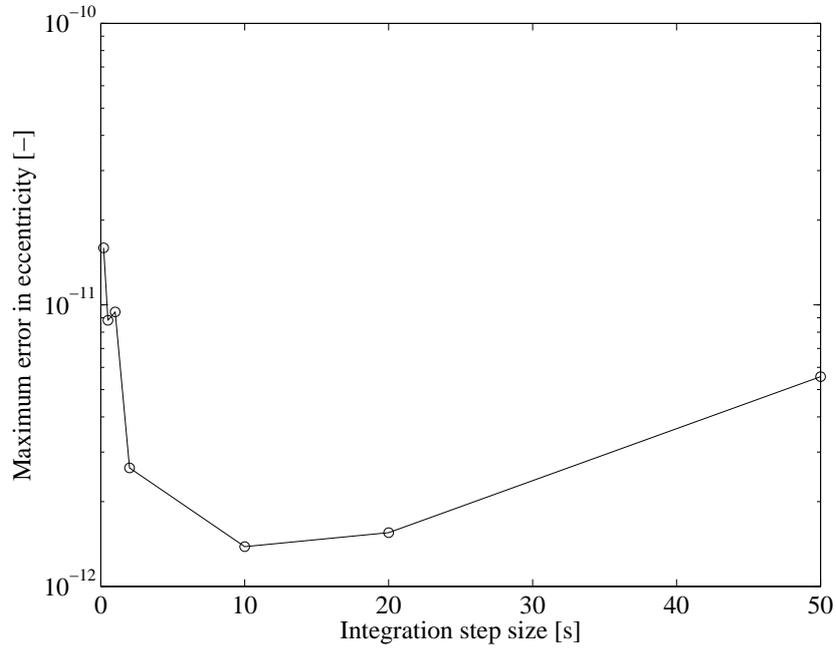
Though the smaller step sizes used in the example are not considered feasible for long-term simulations on account of their corresponding computation times alone, the existence of round-off errors is important to keep in mind. In essence, they make it impossible to come arbitrarily close to the solution of a complex problem by just using increasingly small integration steps. This complicates the matter of creating an accurate reference to use for the simulations of the perturbed problem, as will be discussed in Section 9.4.2.

Table 9.1: The computation times corresponding to the data points in the plots of Figure 9.1.

Integration step size [s]	Computation time
0.2	7 ^h 37 ^m 27 ^s
0.5	3 ^h 12 ^m 04 ^s
1	1 ^h 32 ^m 21 ^s
2	48 ^m 41 ^s
10	9 ^m 13 ^s
20	4 ^m 37 ^s
50	1 ^m 50 ^s



(a) The maximum error in semi-major axis for various integration step sizes.



(b) The maximum error in eccentricity for various integration step sizes.

Figure 9.1: An illustration of round-off errors for a 100-year simulation of a nominal geostationary orbit performed with the RK4 method.

9.3.2 Error Growth

An important property of the two-body problem is that the orbital energy², defined as

$$\mathcal{E} = -\frac{\mu}{2a} \quad (9.1)$$

remains constant. Therefore, the error in orbital energy, which can be derived from a , is a good measure to use for the comparison of different integration methods. In addition, symplectic integrators work in such a way that each sub-step is a mapping which exactly conserves the orbital energy, making it a good showcase for demonstrating what symplectic methods are capable of when integrating Hamiltonian systems.

In Figure 9.2, the error in orbital energy over time is shown for three different integration methods for a 100-year simulation of a nominal geostationary orbit. The number in the legend behind the method shows the integration step size (e.g. 50 s for RK4) or the relative error tolerance (10^{-14} for DOPRI8). As can be observed, the error in orbital energy grows linearly for the RK4 and DOPRI8 methods, while there is no error visible for Kinoshita's method on this scale.

The relation between error growth and step sizes (for fixed step size methods) or accuracy specifications (for adaptive step size methods) can be made visible by plotting the error in orbital energy over time for different integration settings on a logarithmic scale. This has been done for the plots shown in Figure 9.3. Note that because of the logarithmic scale, linear relations appear curved in the plots (similar to plots of logarithmic functions on a normal scale).

Clearly, for both RK4 and DOPRI8, the error growth directly scales with the step size or accuracy specifications used. The errors in orbital energy for Kinoshita's method, which is a symplectic method, do not seem to increase over time. The errors also remain many orders of magnitude smaller than the errors found for the RK4 and DOPRI8 methods. Also worth noting is that the errors for Kinoshita's method do not seem to increase when using a larger step size. In fact, the largest errors are encountered for the runs with the smallest step sizes. This is most likely a result of the accumulation of round-off errors.

It should be noted that although the step size does not seem to have an influence on the error in orbital energy in Figure 9.3(c), the step size does need to stay within reasonable bounds. If too large a step size is chosen (e.g. 5000 s), the error will become much larger.

²Strictly speaking, the energy considered here is the *specific* orbital energy. However, for brevity it will simply be called the orbital energy in this chapter.

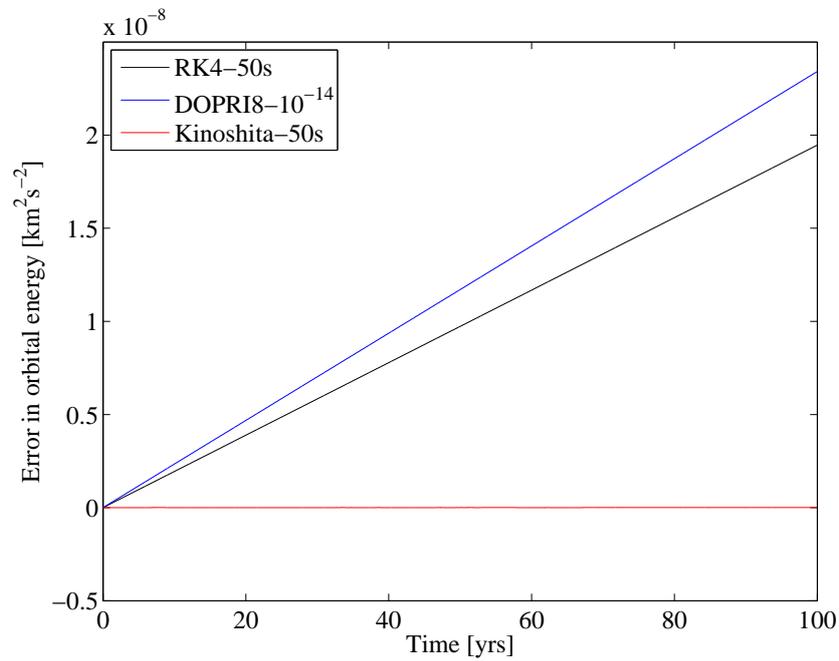
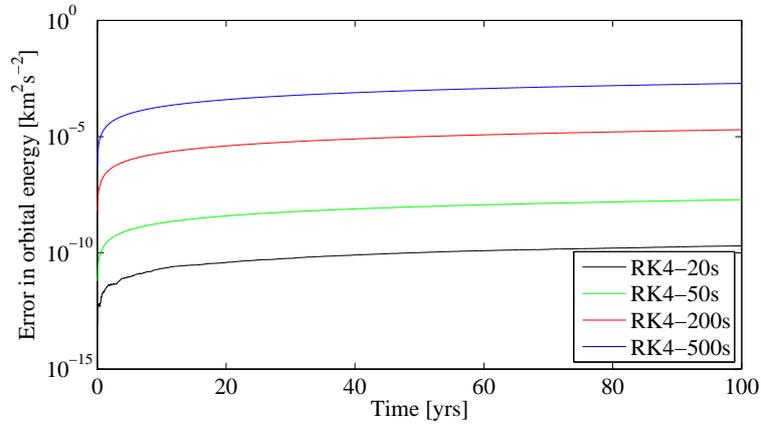
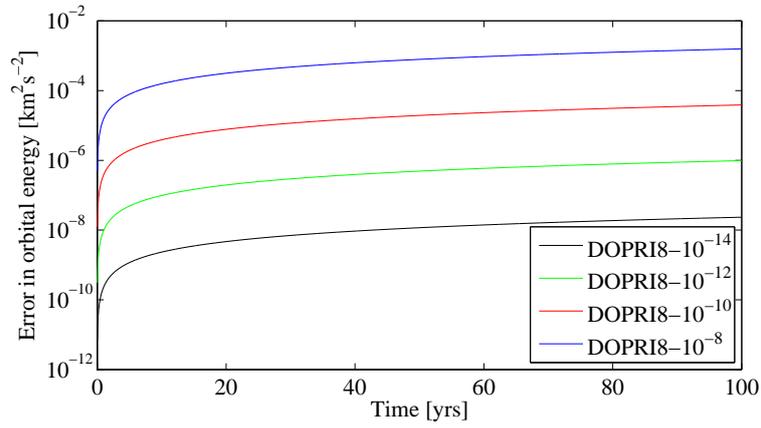


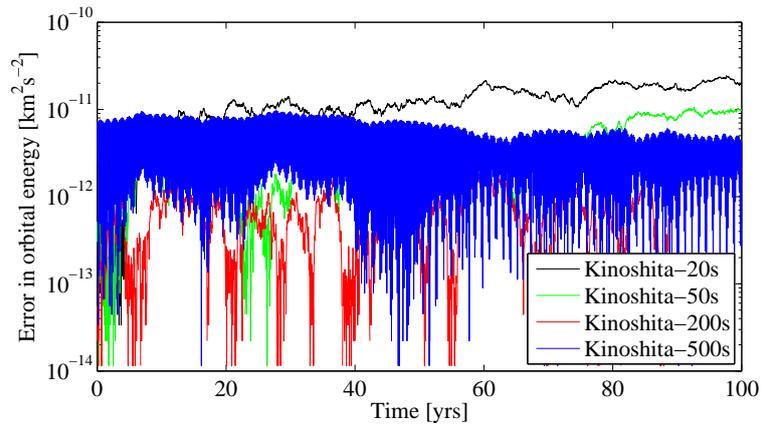
Figure 9.2: The growth of the error in orbital energy over time for three integration methods. The step sizes and accuracy requirements for the methods were merely chosen for illustration purposes. Even though RK4 and DOPRI8 show a similar error growth in this plot, the RK4 simulation took 116.3 s to run, whereas the DOPRI8 simulation only required 29.1 s of CPU time. The simulation with Kinoshita's method needed 70.4 s to complete, but would show a similar error pattern when using larger step sizes.



(a) The error in orbital energy over time when using the RK4 method with different step sizes.



(b) The error in orbital energy over time when using the DOPRI8 method with different accuracy settings.



(c) The error in orbital energy over time when using Kinoshita's method with different step sizes.

Figure 9.3: The performance of different methods for a 100-year simulation of a circular orbit without any perturbing forces.

9.3.3 Performance for a Circular Orbit

In this section, the performance of different methods for integrating a circular two-body problem is compared. The initial conditions used for the simulations are listed in Table 9.2. Because this concerns the integration of a two-body problem, some methods are not included in the comparison. For the traditional methods, the formulation using Gauss' equations with modified equinoctial elements is left out, as in that formulation only the perturbations are integrated. Since there are no perturbations in this case, the solution will automatically consist of a set of constant orbital elements equal to the initial conditions. Similarly, the Wisdom-Holman splitting method for the symplectic integrators will also be excluded from the comparison, since it solves the two-body part using Kepler's equation instead of via numerical integration.

Table 9.2: The initial conditions used for the simulations in this section.

Orbital element	Initial value
a	42,164 km
e	0
i	0°
ω	0°
Ω	0°
θ	0°

The performance plots resulting from the simulations are shown in Figure 9.4. The error in inclination has not been plotted, because for the two-body problem, there is no force acting in a direction perpendicular to the orbital plane. As a result, the inclination will remain the same as at the start of the simulation, irrespective of the integration method used.

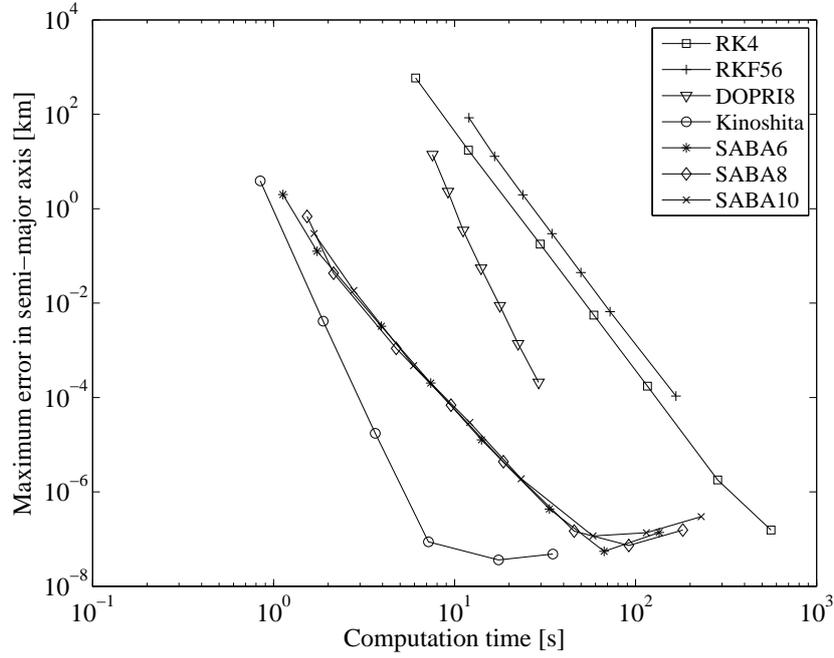
From the first plot, it is quite evident that the symplectic methods perform much better than the traditional methods regarding the error in semi-major axis. Also, Kinoshita's method seems to have a clear edge over the $SABA_{2n}$ methods, which all have roughly the same performance. Although for the same step size $SABA_{10}$ is more accurate than $SABA_6$, the fact that each step for $SABA_{10}$ takes longer to compute compensates for this. Hence, the data points, shown by the markings on the lines, do not overlap, but the lines created by connecting the data points are virtually identical for all $SABA_{2n}$ methods. Because of this, there seems to be no real advantage to using any of the $SABA_{2n}$ integrators over the others. Regarding the traditional methods, DOPRI8 is the most efficient, while the RK4 and RKF56 methods show a relatively similar performance.

When looking at the performance related to the error in eccentricity, the situation is different. This time, the traditional methods show better performance than the symplectic methods, with the DOPRI8 method being the most efficient and the RK4 and RKF56 methods showing roughly equal efficiency again. Of the symplectic methods, Kinoshita's method again clearly outperforms the $SABA_{2n}$

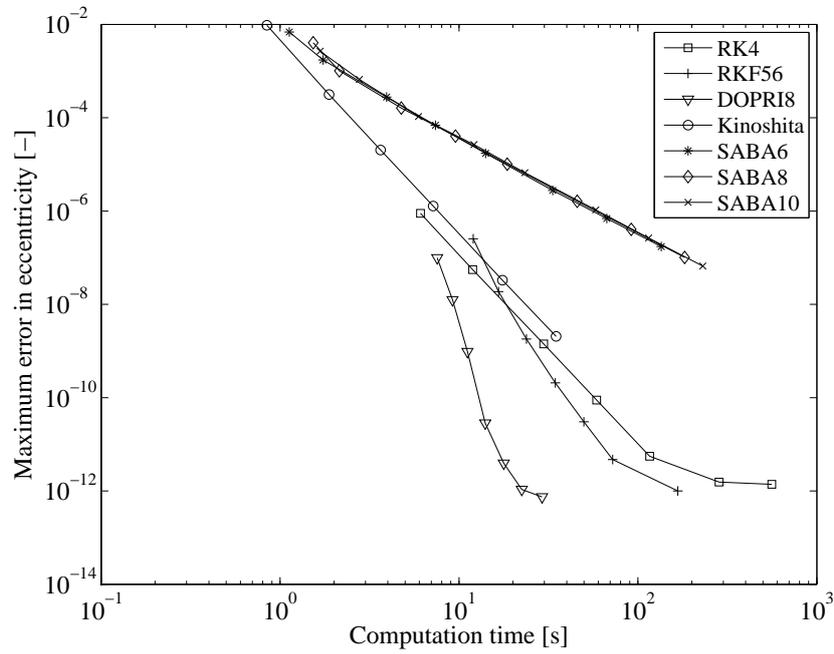
methods.

The dominance of the symplectic methods regarding the semi-major axis accuracy can easily be explained. As mentioned in Chapter 7, symplectic integrators actually apply a sequence of symplectic mappings to the state in order to arrive at a later state. Under the action of these mappings, the total energy of the system is conserved. Since the orbital energy only depends on the semi-major axis (cf. Equation 9.1), only the value of the semi-major axis is implicitly conserved by the mappings. As a result, the value of the eccentricity is by no means as well-preserved as the value of the semi-major axis.

In any case, the efficiency of the symplectic methods, and Kinoshita's method in particular, for integrating the circular two-body problem is remarkable. When only looking at the semi-major axis, millimeter-level accuracy can be reached for a 100-year simulation within less than 10 seconds of computation time by using Kinoshita's method. For reference, trying to reach this level of accuracy with the RK4 method would require over 300 seconds of computation time.



(a) Error in semi-major axis.



(b) Error in eccentricity.

Figure 9.4: The performance of different methods for a 100-year simulation of a circular orbit without any perturbing forces.

9.3.4 Performance for a Highly Eccentric Orbit

In this section, a two-body problem for a highly eccentric orbit is considered. The initial conditions (see Table 9.3) are the same as in the previous section, only the eccentricity has been changed to 0.7, leading to perigee and apogee distances of 12,649 km and 71,679 km, respectively. Although the orbital period remains the same as for the orbit of the previous section, the velocity is not uniform anymore over the orbit. Instead, the velocity when passing perigee (7.32 km/s) is much higher than the velocity at apogee (1.29 km/s). Because of this, it is expected that the adaptive step size methods will gain a relative advantage with respect to the fixed step size methods when comparing this situation to the previous one.

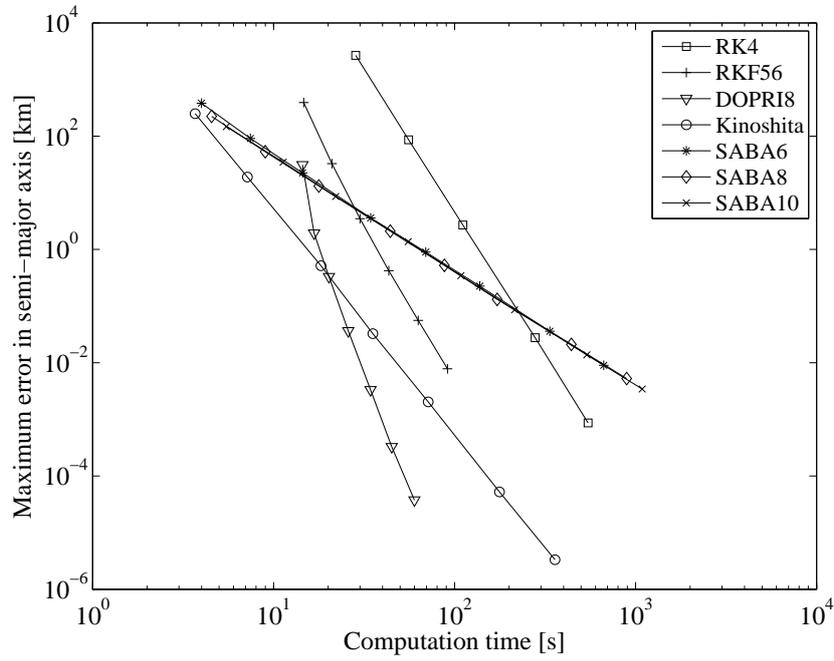
Table 9.3: The initial conditions used for the simulations in this section.

Orbital element	Initial value
a	42,164 km
e	0.7
i	0 °
ω	0 °
Ω	0 °
θ	0 °

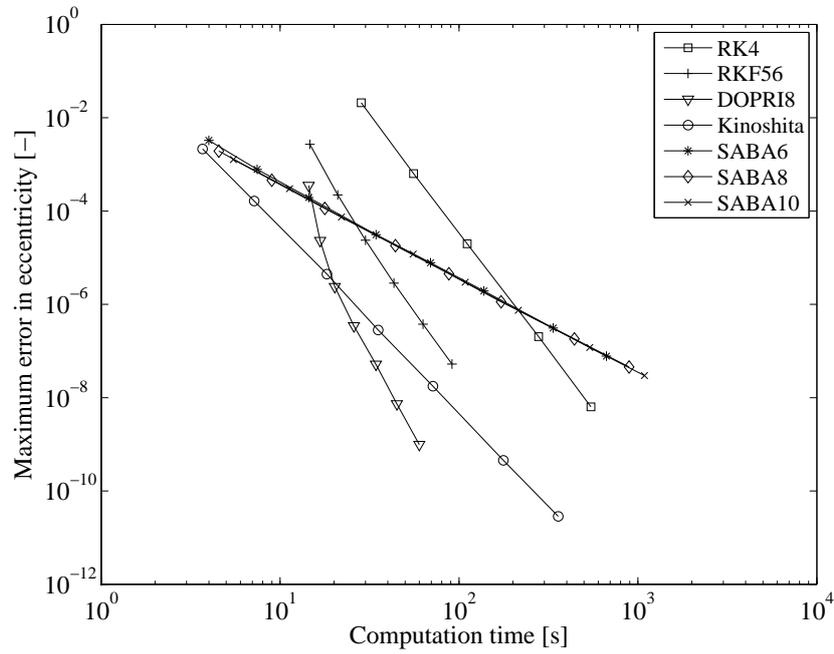
The results of the simulations are shown in Figure 9.5. As can be seen in the plots, the minimum computation time required for getting relatively accurate results has significantly increased for most methods. Notable exceptions are the RKF56 and DOPRI8 methods, which are both adaptive step size methods. Clearly, these are better at dealing with the variable behavior encountered over the orbit than the other methods, which are all fixed step size methods.

The symplectic methods still show strong performance, though they are not as dominant as they were for the circular case. In fact, for semi-major axis accuracy levels better than about 1 km, the DOPRI8 method is actually more efficient than Kinoshita's method. The $SABA_{2n}$ methods, in turn, are surpassed by the RKF56 method for relatively modest accuracy requirements. Additionally, the RKF56 method is clearly preferable over the RK4 method for this simulation case, while the performances of both methods were quite similar for the circular orbit.

Although the symplectic methods are designed for integrating Hamiltonian systems, such as the system in this example, the fact that they are fixed step size methods clearly has an impact on efficiency in this case. Relatively small integration steps need to be taken at the parts of the orbit near perigee to ensure small errors. Because the step size is fixed, the parts near apogee that are significantly easier to integrate need to be integrated with the same small step size. This is an inherent disadvantage of fixed step size methods. Adaptive step size methods, on the other hand, will optimize the step size based on the dynamics that are encountered, leading to a more efficient integration process. The advantages of this are clearly visible in the results of this section.



(a) Error in semi-major axis.



(b) Error in eccentricity.

Figure 9.5: The performance of different methods for a 100-year simulation of a highly eccentric orbit ($e = 0.7$) without any perturbing forces.

9.4 Perturbed Problem

Although the tests regarding the two-body problem are mainly interesting from a theoretical perspective, the performance tests for the perturbed situations have the most practical implications for this thesis project. These tests will actually show which methods are the most efficient for integration of the equations of motion including perturbations, which predict how the orbits of satellites and debris objects will develop over long periods of time.

First, some important remarks related to the simulations of the perturbed problem are made, followed by a discussion about the choice of suitable reference simulations. Then, the performance of all computational methods is tested by means of four distinct simulation cases, in which different perturbed orbits are propagated over a period of 100 years. The conclusions which can be drawn from these simulations, as well as the simulations of the two-body problem, are presented at the end of the chapter.

9.4.1 Important Remarks

Before considering the results of the simulations of the perturbed problem, some important remarks are in order for a better understanding of both the results and the implementation process.

Note on Wisdom-Holman Splitting

As described in Section 7.5.2, the Wisdom-Holman splitting method for symplectic integrators has been implemented in such a way that the two-body part is solved by means of Kepler's equation while the perturbations part is integrated numerically. The state vector is propagated by sequentially advancing both parts multiple times during each integration step.

Since the force model is formulated in Cartesian coordinates, integration of the accelerations resulting from the perturbing forces requires the state to be expressed in Cartesian coordinates as well. Propagation of the two-body part using Kepler's equation, on the other hand, necessitates a formulation of the state in orbital elements. Because both parts are advanced successively, conversions between Cartesian coordinates and orbital elements are required in between all sub-steps of each integration step.

Although this way of performing the integration allows for the two-body part to be solved exactly, it also adds some overhead to the computations by means of the large number of coordinate transformations. As it turns out, the coordinate transformations also have a negative effect on the accuracy of the simulations, which becomes apparent when integrations are performed over long periods of time. As a result, the simulation results produced using symplectic integrators with the Wisdom-Holman splitting method appear to decrease in accuracy when smaller integration steps are used, after reaching a certain accuracy level.

The reason for this is closely related to the concept of round-off errors, discussed in Section 9.3.1. Each individual coordinate transformation is composed of a number of steps, all of which introduce small round-off errors in the state. Although small individually, these errors accumulate over time, and the contribution of this source to the global integration error starts to become relevant once a sufficient number of coordinate transformations have been applied. Consequently, using smaller integration steps, which increases the total number of coordinate transformations and round-off errors, can eventually increase the global integration error.

Accordingly, choosing an appropriate step size for methods using the Wisdom-Holman splitting technique becomes a balancing act between minimizing the truncation error resulting from integration of the perturbed part and keeping the contribution of round-off errors, which mainly result from the required coordinate transformations, within bounds.

Note on Adaptive Step Size Methods in Tudat

As was mentioned in Section 8.1, the routines for adaptive step size methods from Tudat have been implemented in the simulation code. Specifically, this concerns the implementation of the RKF56 and DOPRI8 methods. Unfortunately, the Tudat routines for these methods were not completely correct, which has led to some considerable delays in the later stages of the thesis project.

The first error concerned the Tudat implementation of the DOPRI8 method. As defined by Prince and Dormand (1981), this is an eighth-order embedded Runge-Kutta method. The notation Dormand-Prince 8(7) implies that the eighth-order method is used for propagating the solution, while the seventh-order method is only used for step size control. Careful inspection of the original Tudat implementation, however, revealed that the method was implemented the other way around, i.e. the eighth-order method was used for controlling the step size, whilst the seventh-order method was used for advancing the solution. Although this difference would not have a huge impact on the accuracy of the solution, from a theoretical perspective and for the performance comparison of this chapter it is important that the method is implemented correctly. The error in Tudat was fixed shortly after it was found.

Another error in Tudat, which would have more far-reaching consequences for the thesis project, involved the working of the Tudat Core routine *integrateTo*. As explained in Section 9.2, the performance comparisons are carried out by comparing the orbital elements computed by a method at a large number of fixed output points with those of a highly accurate reference simulation. This approach makes it necessary to integrate exactly to fixed points in time and to save the data at these points. Since adaptive step size methods use steps that are optimized based on local error estimates, the steps taken using these methods will generally not coincide with predefined time instants. In order to force the integrator to integrate exactly to intermediate output points, the *integrateTo* routine was used. With this routine, the end of a time interval can be specified. The integration method then takes steps

and makes sure the end of the last step exactly coincides with the specified time.

During simple tests of the simulation code, the adaptive step size methods appeared to be working correctly. However, when assessing the performance of all integration methods for the simulations of the perturbed two-body problem of this chapter, the RKF56 and DOPRI8 methods showed rather unexpected results. The accuracy attained with both methods was quite mediocre, also when using the most stringent accuracy requirements. In fact, both methods were outperformed by the RK4 method when the formulation in modified equinoctial elements was used. Subsequent checks of the Tudat implementations of RKF56, DOPRI8 and adaptive step size methods in general revealed no errors. The fact that both RKF56 and DOPRI8 appeared to be working correctly in the simulations of the two-body problem further seemed to confirm that the Tudat code was correct.

The source of the erratic behavior finally became apparent when the development of errors over time was considered (see Figure 9.6). The peaks in the plot hinted at large local errors that were somehow not propagated with the rest of the solution. When the step size of DOPRI8 was artificially limited to a maximum of 300 s, the plot of Figure 9.7 was obtained. As can be observed, there are much less peaks present than in the previous results, but there are still some. Because the performance plots are based on *maximum* errors, the results shown in Figures 9.6 and 9.7 would result in maximum errors on the order of 10^{-2} km instead of 10^{-6} km.

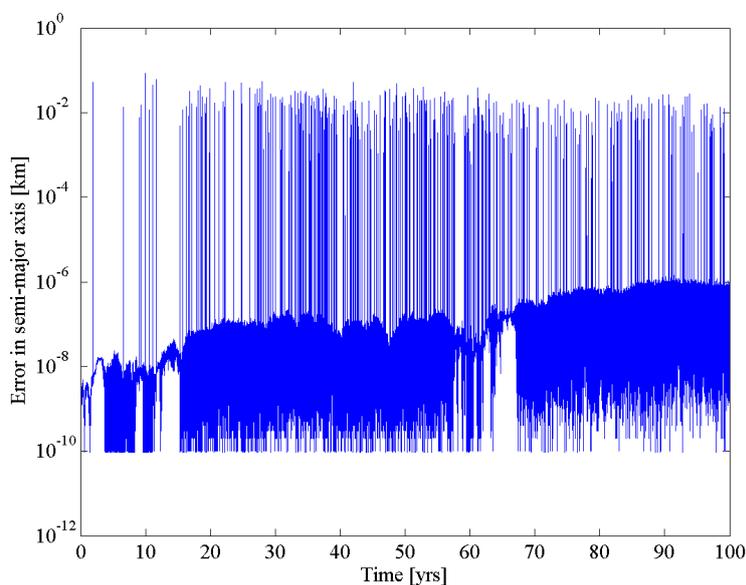


Figure 9.6: The error in semi-major axis over time for a 100-year simulation of a perturbed geostationary orbit, using DOPRI8 with modified equinoctial elements in combination with the old Tudat implementation of *integrateTo*.

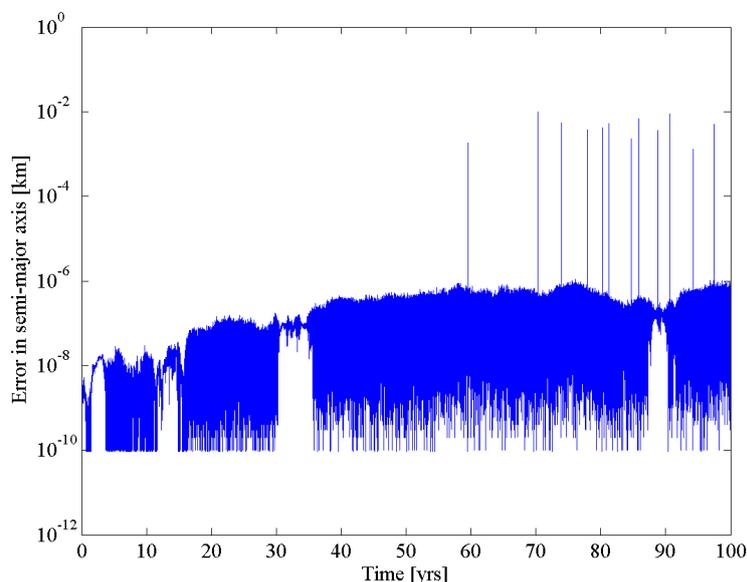


Figure 9.7: The error in error semi-major axis over time for the same simulation settings as used for the results in Figure 9.6, but with a specified *maximum* step size of 300 s.

The cause for the unusually large local errors turned out to be the following. Whenever *integrateTo* was used, the last integration step before the output time would be truncated, such that the end of the step would exactly land on the specified instant. Whenever the last step size would be accepted by the routine for step size control of the adaptive step size method, everything would work exactly as intended and the last integration step would be made to the specified output time. If, however, the truncated step size would not be accepted by the routine for step size control, a new, smaller step size would be computed and the last integration step would be taken using this new step size. In this case, the *integrateTo* routine would not continue the integration, leaving a gap in time between the time of the last integrated state and the desired output time. As a result, the state at a point in time before the desired output time would actually be saved as if it were the state at the output time. For example, the state at $t = 9,500$ s could be recorded as if it were the state at $t = 10,000$ s.

This particular oversight in the code of *integrateTo* was what caused the peaks in the plots, and consequently, the large maximum errors computed for the performance plots. Because the integrators in Tudat are actually C++ objects which store the state and time internally, the correct time corresponding to the state would be used when integrating the trajectory further. As a result, the errors related to the incorrect functioning of the *integrateTo* routine would not be propagated, hence the isolated peaks in the plots.

Once the error in *integrateTo* was located, it could be fixed quite easily. The

result of a simulation performed after the fix is shown in Figure 9.8. Clearly, the peaks are no longer present in the results and the maximum error is in the range of 10^{-6} km instead of 10^{-2} km. After implementing the fix, RKF56 and DOPRI8 finally showed the performance that was expected of the methods. The solution has since been adopted in Tudat Core. It should be noted that because *integrateTo* is a generic function in Tudat, the fix solves the working of *integrateTo* for all adaptive step size methods in Tudat, not only RKF56 and DOPRI8.

Unfortunately, because the error in *integrateTo* was found relatively late in the project, it had quite some implications. Many simulation runs had to be run again in the end, but before that, conclusions were made based on incorrect simulations. Because some of the conclusions were ultimately found hard to believe, much effort was put into finding any potential errors. Fortunately, the bug in *integrateTo* was eventually found, and consequently, all simulations could be conducted using correctly functioning implementations of the adaptive step size methods.

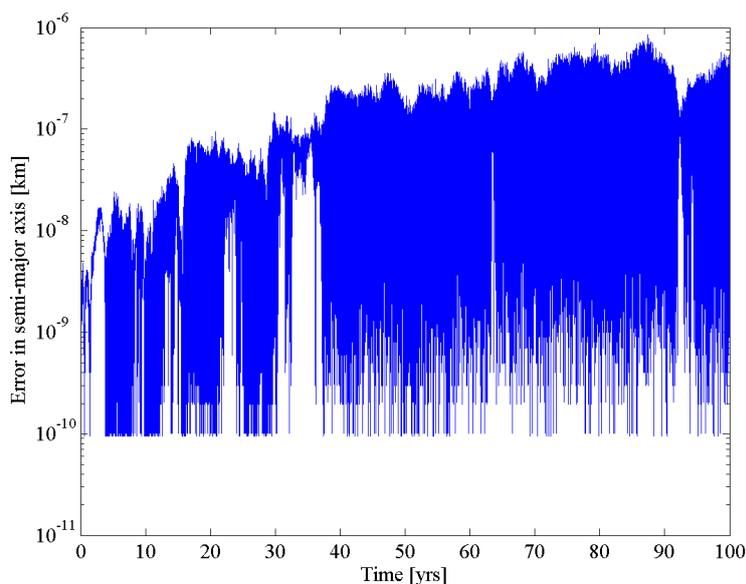


Figure 9.8: The error in error semi-major axis over time for the same simulation settings as used for the results in Figure 9.6, after correcting the implementation of the *integrateTo* routine.

9.4.2 Choice of Reference

In contrast to the two-body problem, no exact solution is available for simulations for which the full force model has been activated. This makes it significantly more difficult to quantify the performance of computational methods for carrying out simulations of the perturbed two-body problem. In order to provide a good estimate of the accuracy of simulations performed with different methods, a highly accurate

reference is needed. Choosing such a reference is a delicate task, as one can never be fully certain that a specific reference presents a solution that closely resembles the actual solution of the system.

As was discussed in Section 9.3.1, due to round-off errors it is generally not possible to get extremely close to the actual solution of a complex system by just using exceedingly small integration steps. Especially when integrating over long periods of time, round-off errors can become quite relevant. Originally, Kinoshita's method combined with Wisdom-Holman splitting was chosen as a possible method for generating the references for the simulations of the perturbed problem. However, due to the accumulation of round-off errors discussed in the note on Wisdom-Holman splitting in Section 9.4.1, the method is not suitable for producing extremely accurate solutions³. Additionally, it would not be clear beforehand which step size would result in the best solution, as a smaller truncation error comes at the cost of a larger total round-off error.

When choosing between the other methods, the choice was made to focus on the traditional methods in combination with Gauss' form of Lagrange's planetary equations with modified equinoctial elements. Both the symplectic methods with T+V splitting and the traditional methods with Cartesian coordinates would result in formulations in which the central gravitational force would contribute significantly to the local truncation errors. Using Gauss' equations would allow integration of the effects of the perturbing forces only, which intuitively would result in a better approximation of the perturbed trajectory. A number of numerical tests confirmed this intuition.

The RK4, RKF56 and DOPRI8 methods in combination with Gauss' equations were all tested in their abilities to produce accurate references. This was done by computing the maximum errors of other simulation runs with the runs that were tested as references. References that resulted in small maximum errors for all other methods were considered to be more accurate than references which had larger maximum errors for the other runs.

In the end, DOPRI8 with the most stringent accuracy requirements and RK4 with a very small step size showed the best results when used as reference. Based on its track record for a wide range of applications (Montenbruck and Gill, 2005), DOPRI8 was eventually chosen as the method to produce the references with. Essentially, DOPRI8 should be able to handle most systems well, without having to know the exact dynamical behavior of the system beforehand. Since RK4 is a fixed step size method without any form of error control, it does not necessarily handle irregularities well, making it less suitable for producing reliable references than DOPRI8.

³In fact, this characteristic of Wisdom-Holman splitting in the implementation of the simulation code was discovered when testing different references.

9.4.3 Case I: Near-Circular GEO Graveyard Orbit – Low A/m

This first test case considers the integration of a near-circular GEO graveyard orbit for a satellite with an effective area-to-mass ratio of $0.02 \text{ m}^2/\text{kg}$. Due to the low area-to-mass ratio, solar radiation pressure can be expected to only have a minor effect in this simulation. Because the orbit is also near-circular, this simulation case is considered to be the easiest one to integrate of the cases that are tested. The initial conditions for this case are listed in Table 9.4.

Table 9.4: The initial conditions for simulation case I.

Starting epoch	
Jan. 1, 2014 00:00:00	
Orbital element	Initial value
a	42,464 km
e	0.001
i	0.1°
ω	0°
Ω	0°
θ	0°
Physical property	Value
$C_R \cdot A/m$	$0.02 \text{ m}^2/\text{kg}$

The long-term evolution of the orbital elements for this simulation case, based on the reference simulation, is shown in Figure 9.9. The performance plots for the different computational methods corresponding to the simulation case are shown in Figures 9.10 to 9.12.

Note that each of the figures contains two plots. For each figure, the upper plot indicates the performance of the traditional methods and Kinoshita's method, while the lower plot shows the performance of the symplectic methods. Two separate plots have been used, since combining all data into a single plot would result in a plot that was very hard to interpret due to the abundance of data. The inclusion of Kinoshita's method in both plots should allow for a relatively easy comparison between the traditional and symplectic methods. Also note that Figure 9.10 shows the maximum errors in semi-major axis, while Figures 9.11 and 9.12 display the maximum errors in eccentricity and inclination, respectively. This approach for displaying the performance data is maintained for all simulation cases in the remainder of this chapter. The results of the different simulation cases are discussed at the end of the chapter.

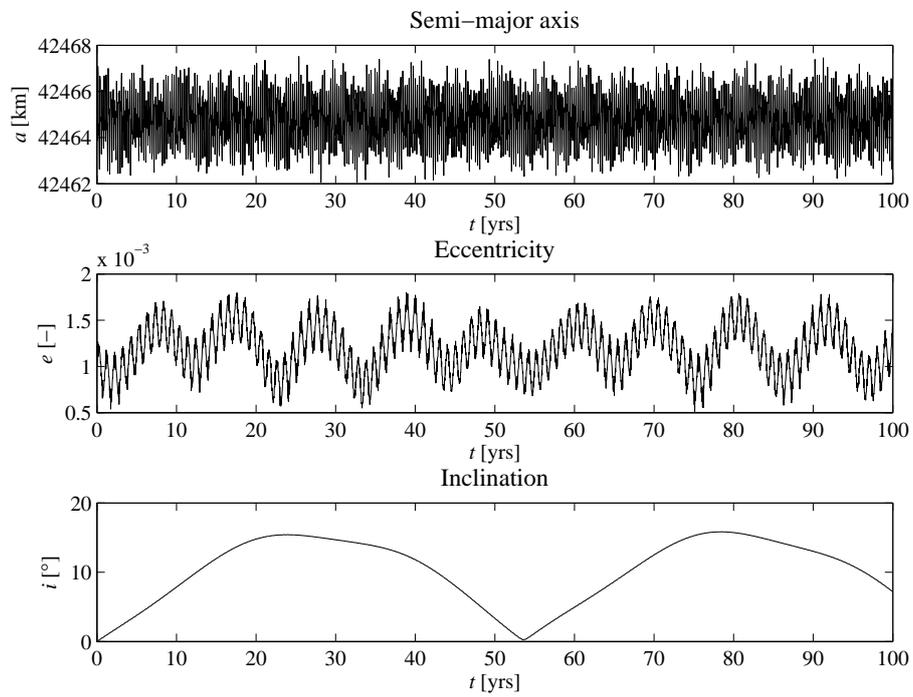
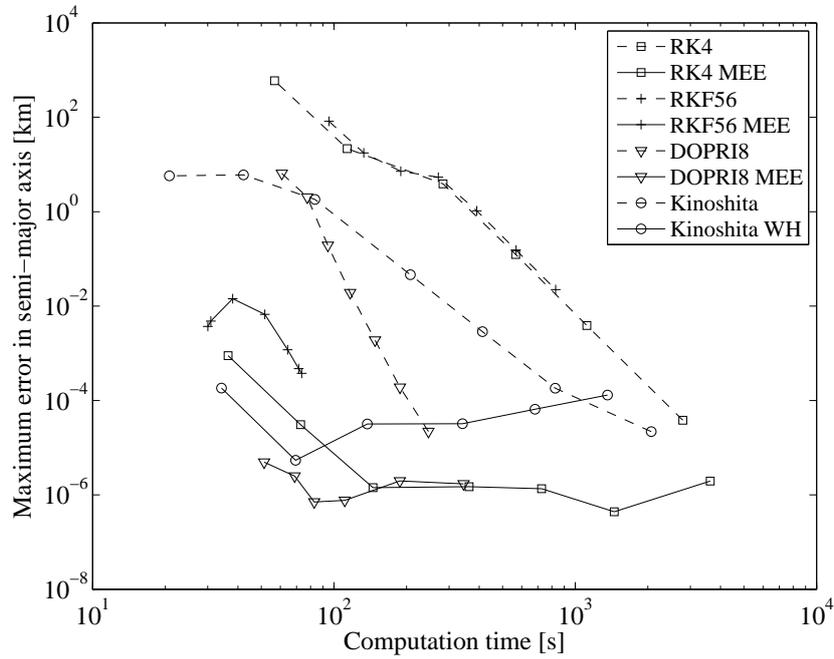
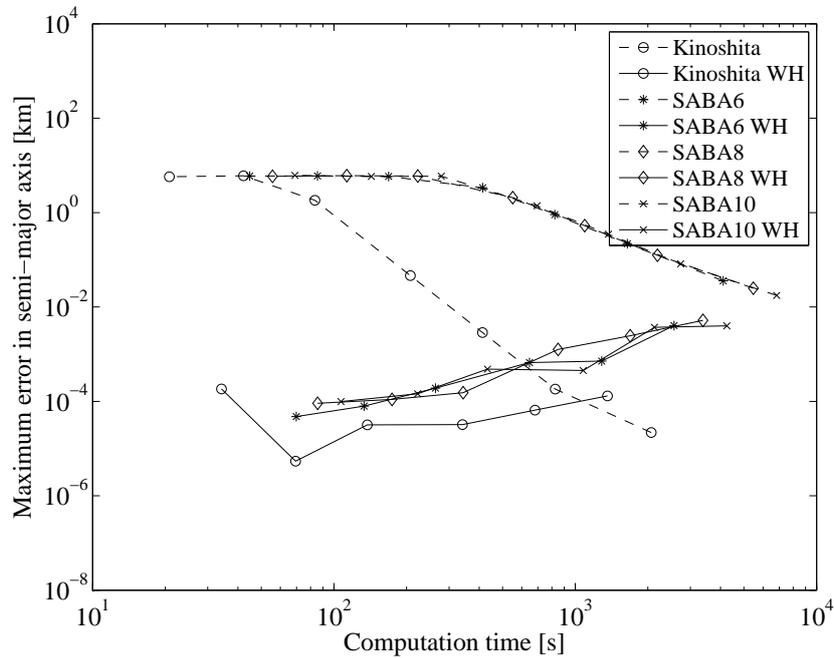


Figure 9.9: The long-term evolution of the orbital elements for simulation case I.

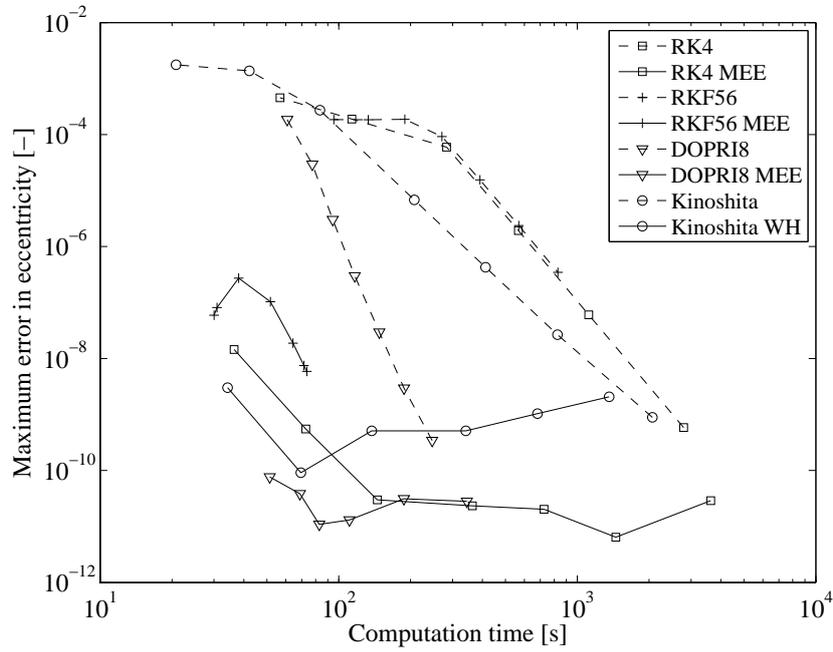


(a) Traditional methods and Kinoshita's method.

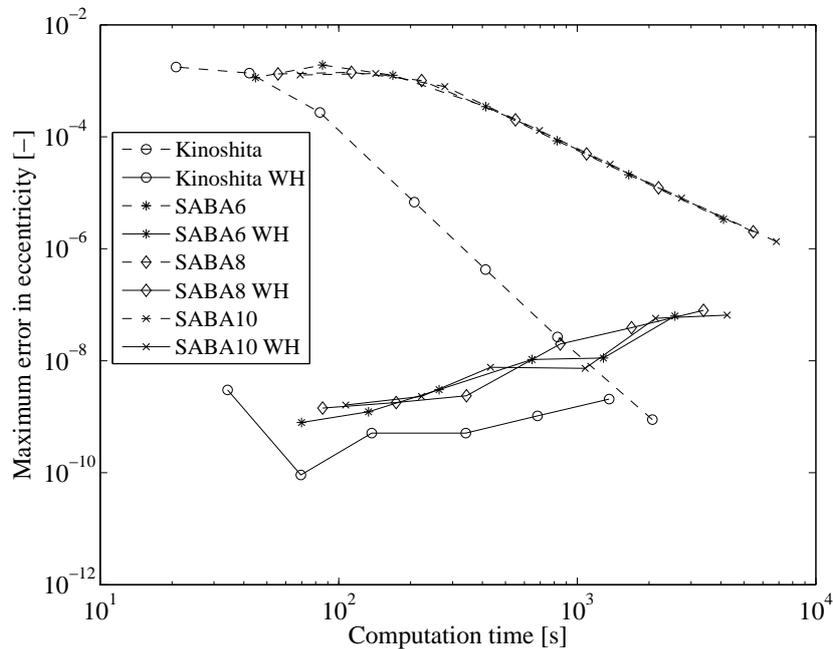


(b) Symplectic methods.

Figure 9.10: The performance regarding the semi-major axis for different computational methods, for a 100-year propagation of a near-circular GEO graveyard orbit of an object with a low area-to-mass ratio.

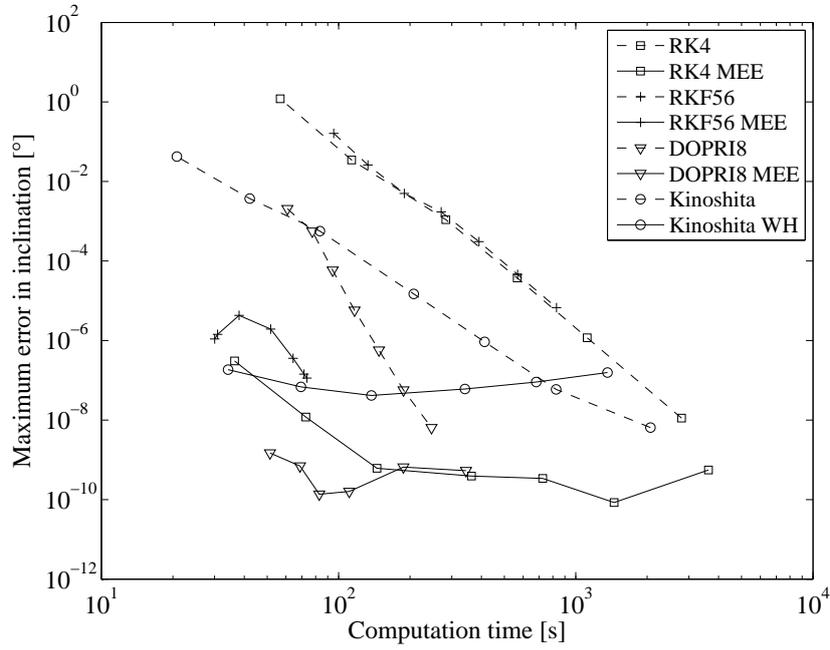


(a) Traditional methods and Kinoshita's method.

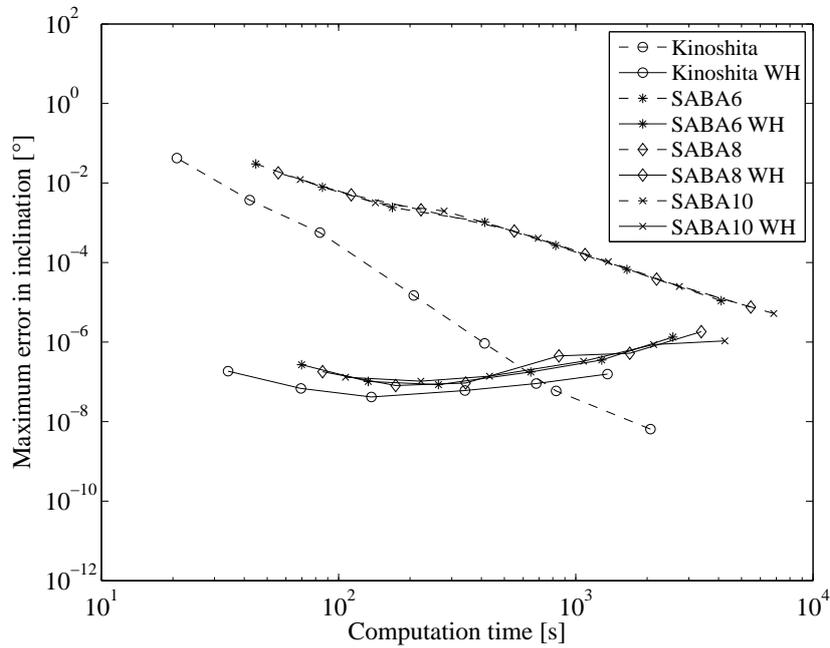


(b) Symplectic methods.

Figure 9.11: The performance regarding the eccentricity for different computational methods, for a 100-year propagation of a near-circular GEO graveyard orbit of an object with a low area-to-mass ratio.



(a) Traditional methods and Kinoshita's method.



(b) Symplectic methods.

Figure 9.12: The performance regarding the inclination for different computational methods, for a 100-year propagation of a near-circular GEO graveyard orbit of an object with a low area-to-mass ratio.

9.4.4 Case II: Near-Circular GEO Graveyard Orbit – High A/m

Similar to the first simulation case, case II concerns the performance for simulating the long-term orbital development of a near-circular GEO graveyard orbit. However, in this case, the object for which the trajectory is simulated has a very high effective area-to-mass ratio of $40 \text{ m}^2/\text{kg}$. As a result, the force due to solar radiation pressure will have a major impact on the trajectory, which will make the system harder to integrate accurately over long periods of time. An overview of the initial conditions used for this simulation case is given in Table 9.5.

Table 9.5: The initial conditions for simulation case II.

Starting epoch	
Jan. 1, 2014 00:00:00	
Orbital element	Initial value
a	42,464 km
e	0.001
i	0.1°
ω	0°
Ω	0°
θ	0°
Physical property	Value
$C_R \cdot A/m$	$40 \text{ m}^2/\text{kg}$

The results of the reference simulation for this case are shown in Figure 9.13. The corresponding performance plots are displayed in Figures 9.14 to 9.16.

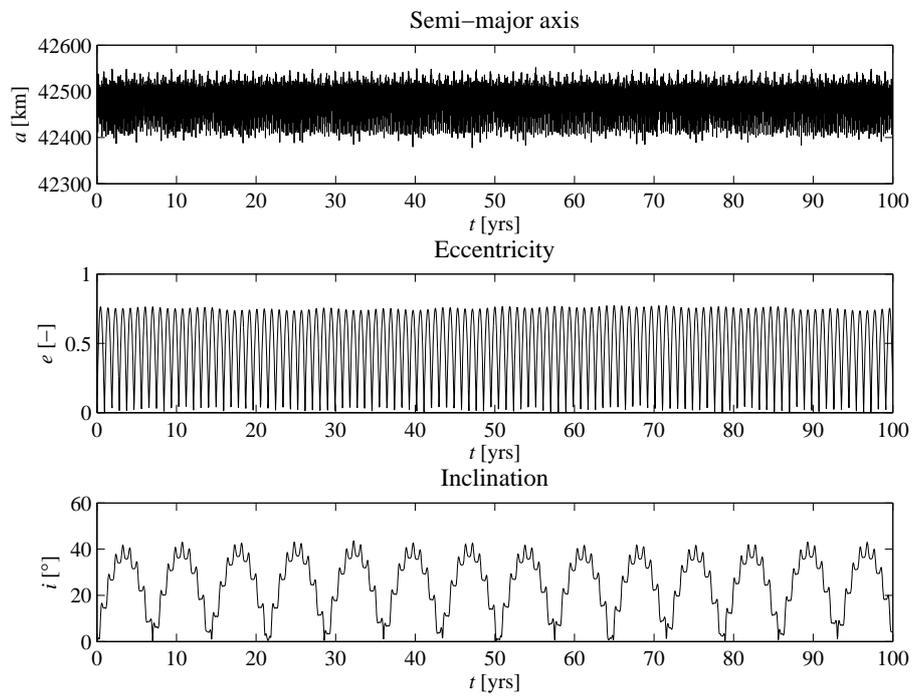
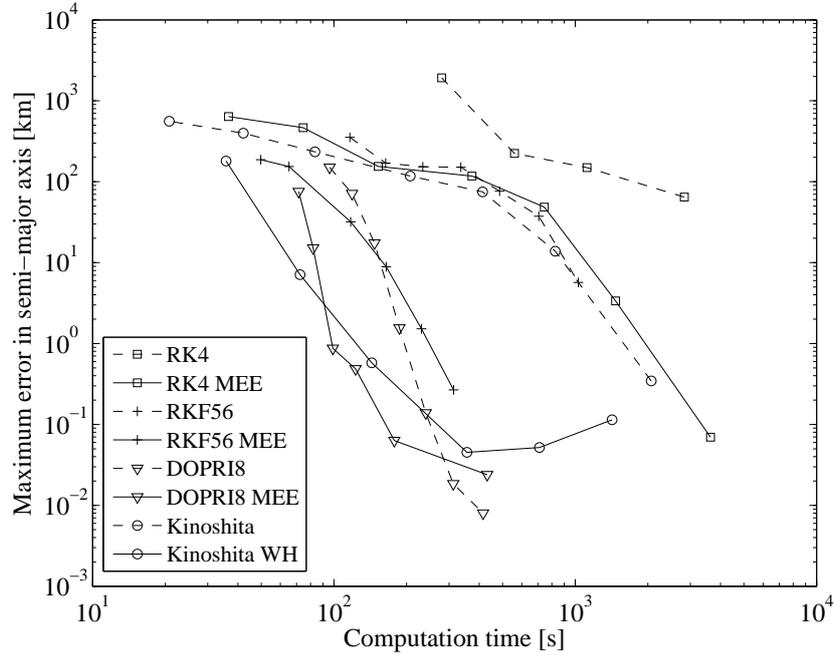
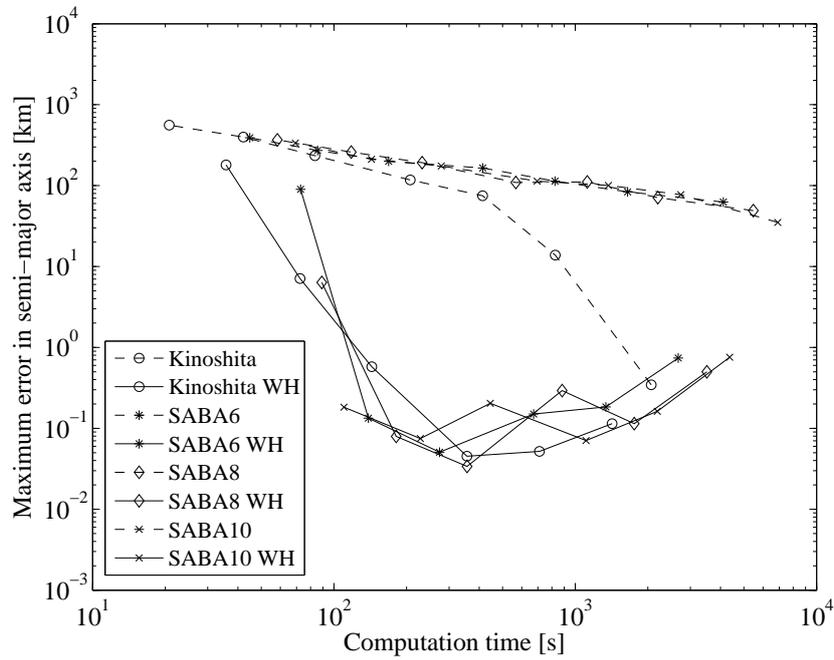


Figure 9.13: The long-term evolution of the orbital elements for simulation case II.

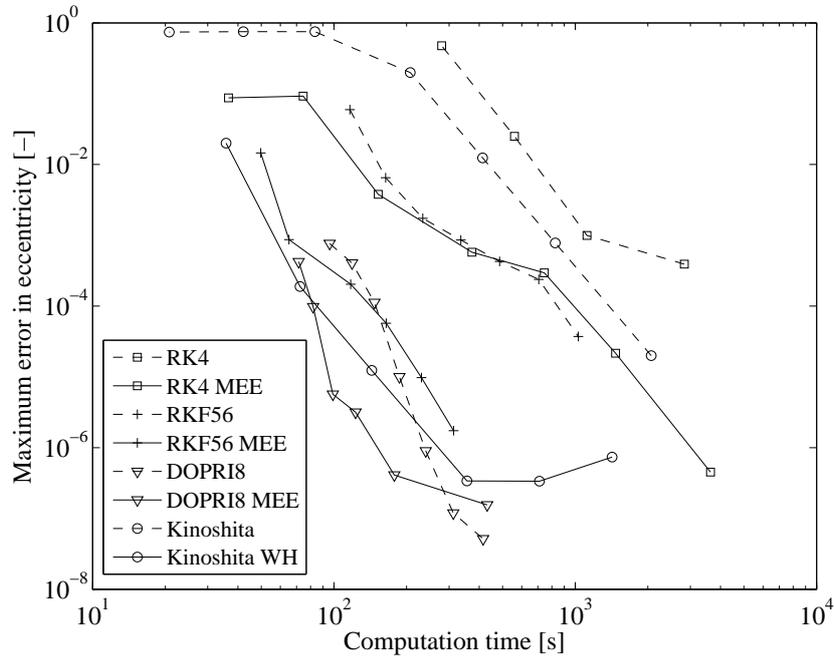


(a) Traditional methods and Kinoshita's method.

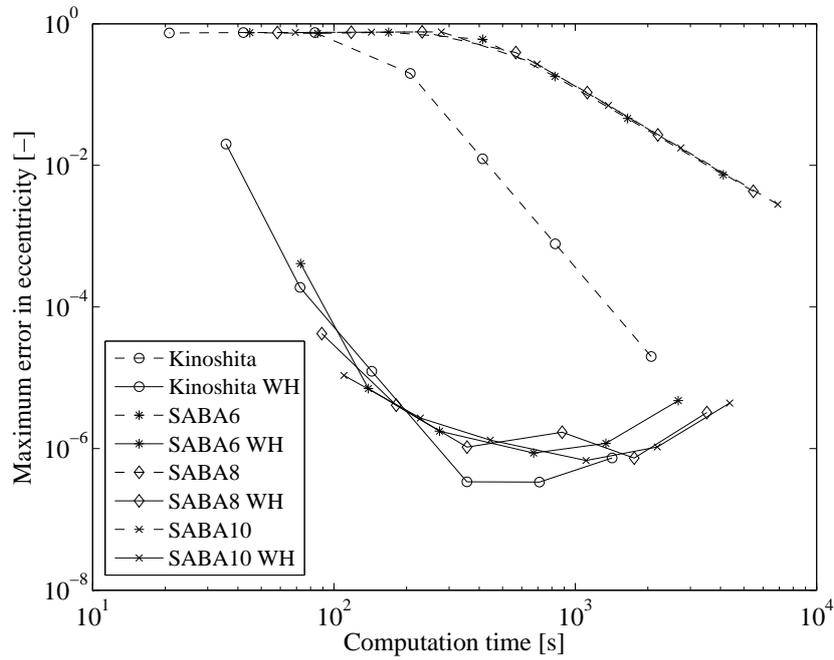


(b) Symplectic methods.

Figure 9.14: The performance regarding the semi-major axis for different computational methods, for a 100-year propagation of a near-circular GEO graveyard orbit of an object with a high area-to-mass ratio.

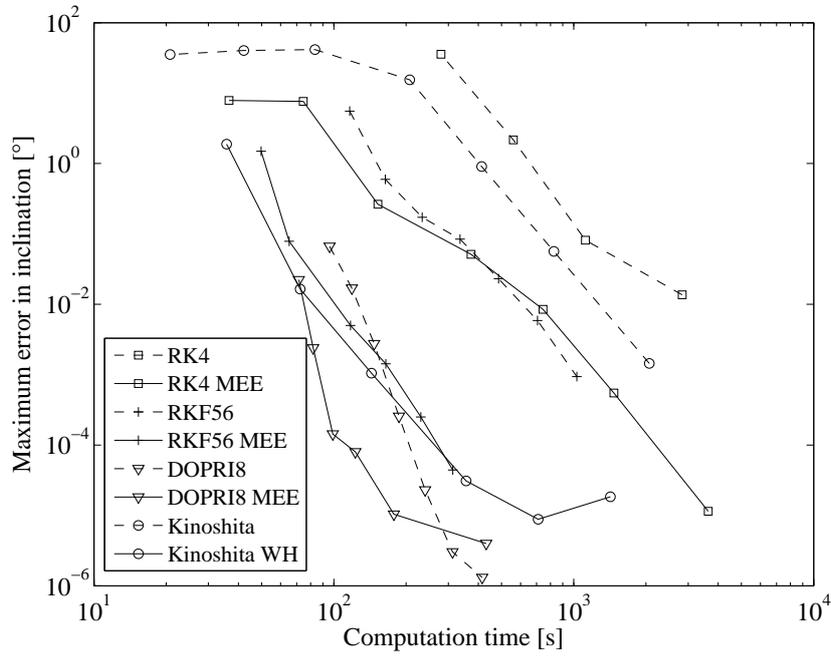


(a) Traditional methods and Kinoshita's method.

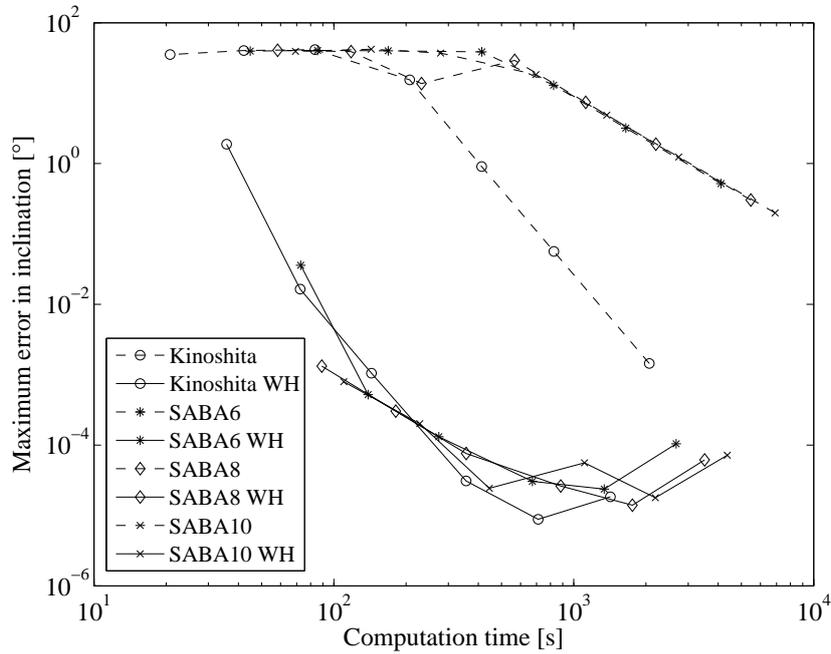


(b) Symplectic methods.

Figure 9.15: The performance regarding the eccentricity for different computational methods, for a 100-year propagation of a near-circular GEO graveyard orbit of an object with a high area-to-mass ratio.



(a) Traditional methods and Kinoshita's method.



(b) Symplectic methods.

Figure 9.16: The performance regarding the inclination for different computational methods, for a 100-year propagation of a near-circular GEO graveyard orbit of an object with a high area-to-mass ratio.

9.4.5 Case III: Highly Eccentric GPS Graveyard Orbit – Low A/m

In this test case, a highly eccentric GPS graveyard orbit is simulated for a satellite with a low area-to-mass ratio. The initial orbit has been based on the end-of-life guidelines listed in Table 2.2, and has perigee and apogee altitudes of respectively 2,500 km and 19,900 km. This presents the most eccentric GPS graveyard orbit that is possible under the guidelines formulated by NASA. The corresponding initial conditions are listed in Table 9.6.

A highly eccentric orbit has deliberately been chosen, because the dynamics related to it are quite distinct from the dynamics of the near-circular orbits considered in the previous two test cases. Additionally, the lower overall altitude implies that the perturbations related to the gravity field of the Earth will become more important, whereas the accelerations resulting from the gravitational attraction of the Sun and Moon will slightly decrease in magnitude. Moreover, the force model will change in nature depending on the distance from the Earth, governed by the position in the (eccentric) orbit. Because of the low area-to-mass ratio, solar radiation pressure will only play a minor role in this simulation case.

Table 9.6: The initial conditions for simulation case III.

Starting epoch	
Jan. 1, 2014 00:00:00	
Orbital element	Initial value
a	17,578 km
e	0.4949
i	55 °
ω	0 °
Ω	0 °
θ	0 °
Physical property	Value
$C_R \cdot A/m$	0.02 m ² /kg

The long-term evolution of the orbital elements for this simulation case is given in Figure 9.17. The corresponding performance plots for the different methods are shown in Figures 9.18 to 9.20.

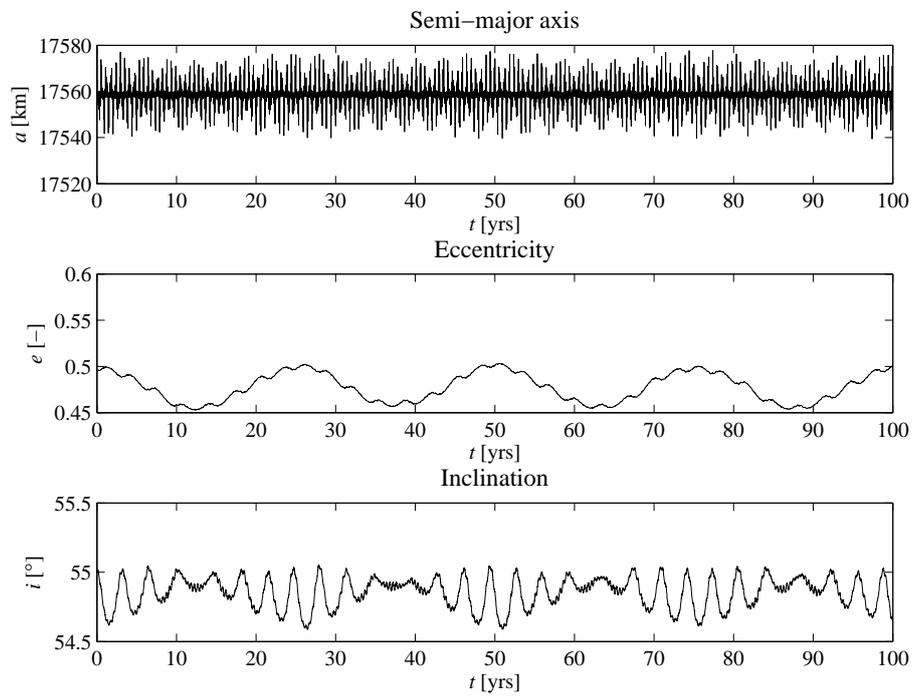
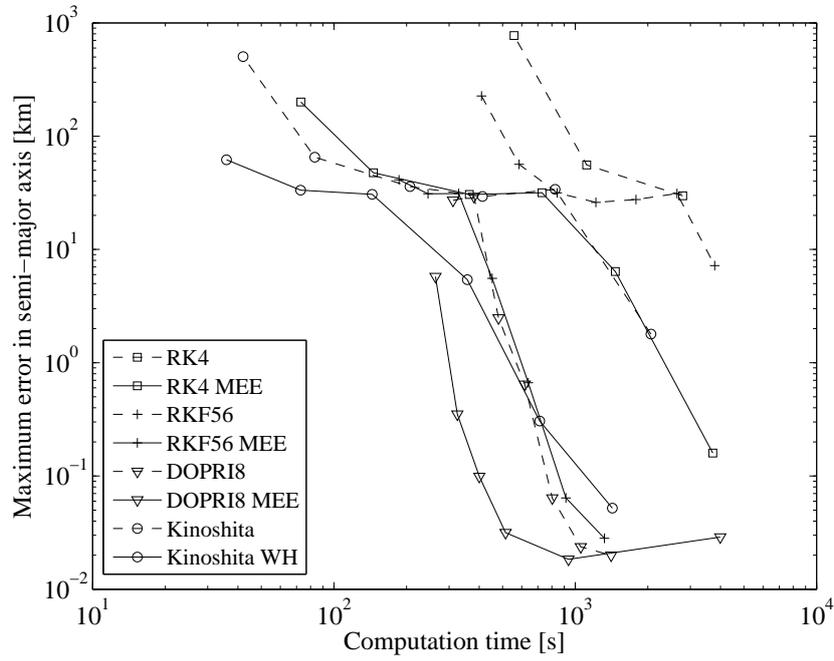
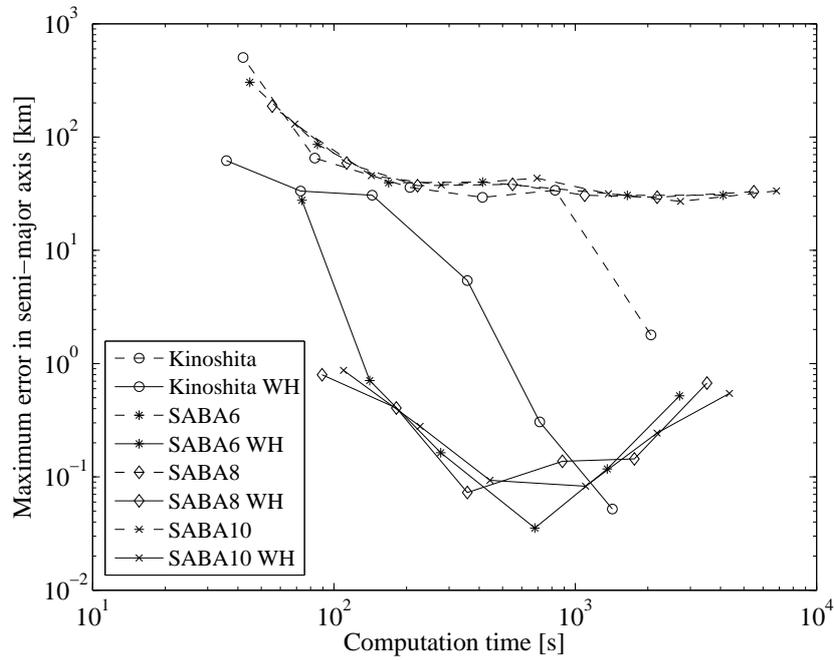


Figure 9.17: The long-term evolution of the orbital elements for simulation case III.

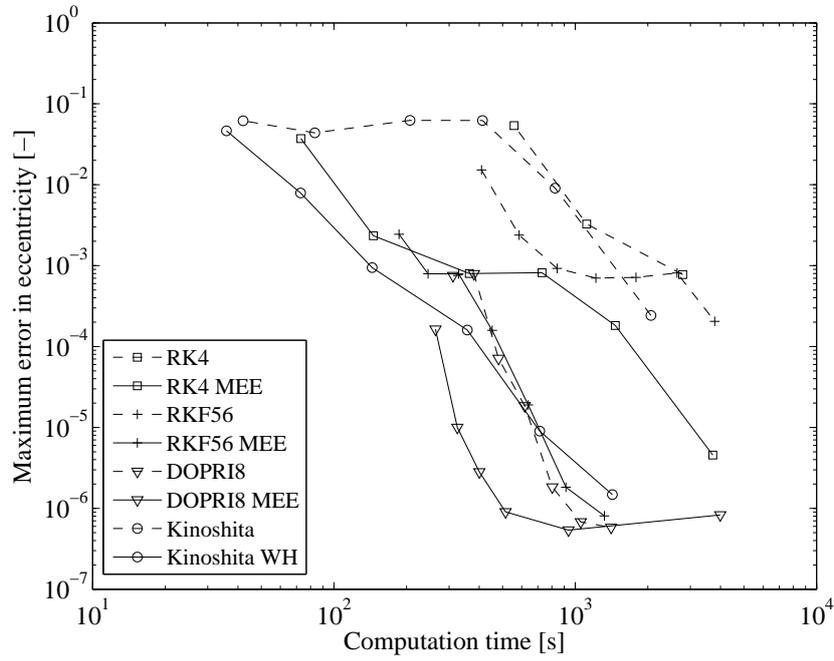


(a) Traditional methods and Kinoshita's method.

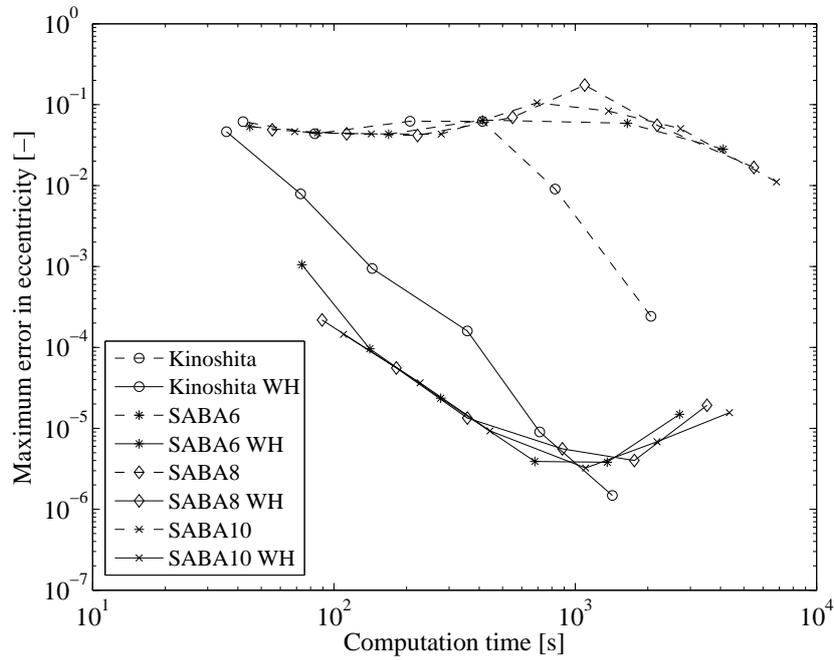


(b) Symplectic methods.

Figure 9.18: The performance regarding the semi-major axis for different computational methods, for a 100-year propagation of a highly eccentric GPS graveyard orbit of an object with a low area-to-mass ratio.

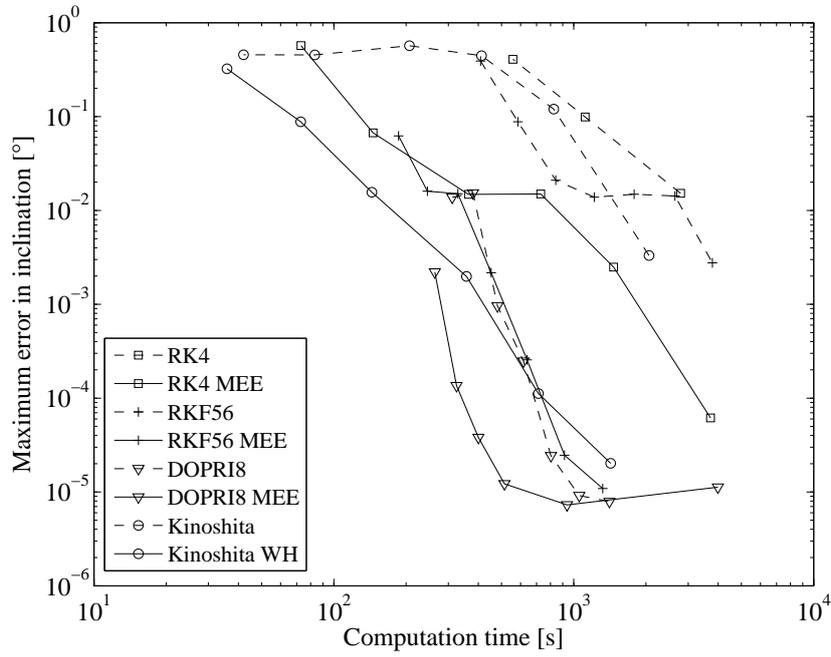


(a) Traditional methods and Kinoshita's method.

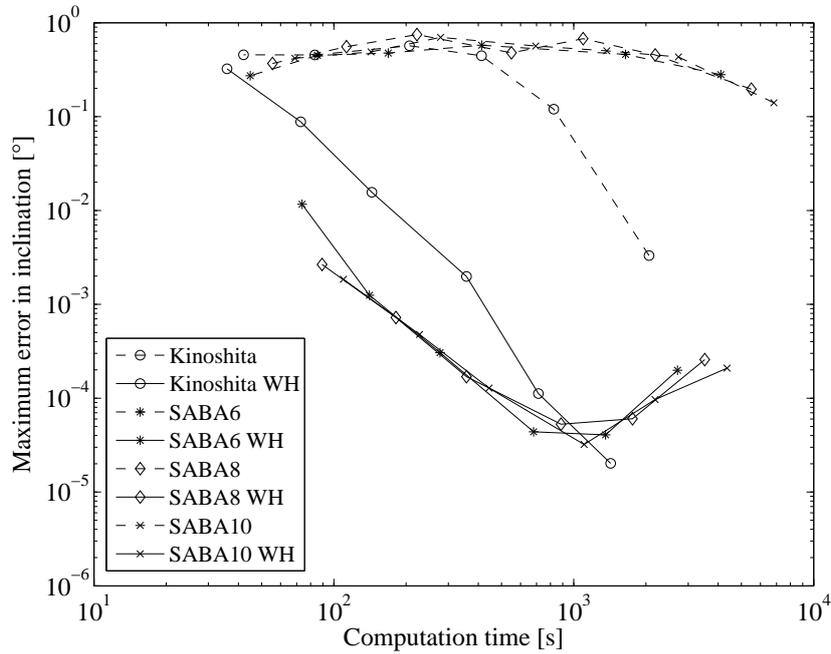


(b) Symplectic methods.

Figure 9.19: The performance regarding the eccentricity for different computational methods, for a 100-year propagation of a highly eccentric GPS graveyard orbit of an object with a low area-to-mass ratio.



(a) Traditional methods and Kinoshita's method.



(b) Symplectic methods.

Figure 9.20: The performance regarding the inclination for different computational methods, for a 100-year propagation of a highly eccentric GPS graveyard orbit of an object with a low area-to-mass ratio.

9.4.6 Case IV: Near-Circular GPS Graveyard Orbit – High A/m

In this final test case, a near-circular GPS graveyard orbit is considered for an object with a high area-to-mass ratio. Simulations have shown that the initial conditions of the previous case will lead to decay within a year if the trajectory of an object with a high area-to-mass ratio is modeled (see Figure 10.8(c) on page 173). Therefore, these initial conditions combined with a high-area to mass ratio do not represent a good test case for long-term integration performance. Instead, the initial conditions listed in Table 9.7 will be used for the final test case.

Of the other test cases that were considered, this test case is most similar to case II. However, in this case, the initial orbit lies in the GNSS region of space and has a relatively high initial inclination. As a result, the system that is integrated will still have significantly different characteristics than case II. It will be interesting to see how the performances of the computational methods compare between the two cases.

Table 9.7: The initial conditions for simulation case IV.

Starting epoch	
Jan. 1, 2014 00:00:00	
Orbital element	Initial value
a	27,260 km
e	0.001
i	55 °
ω	0 °
Ω	0 °
θ	0 °
Physical property	Value
$C_R \cdot A/m$	40 m ² /kg

The results of the reference simulation for case IV are shown in Figure 9.21. In addition, the performance plots for the different computational methods are presented in Figures 9.22 to 9.24.

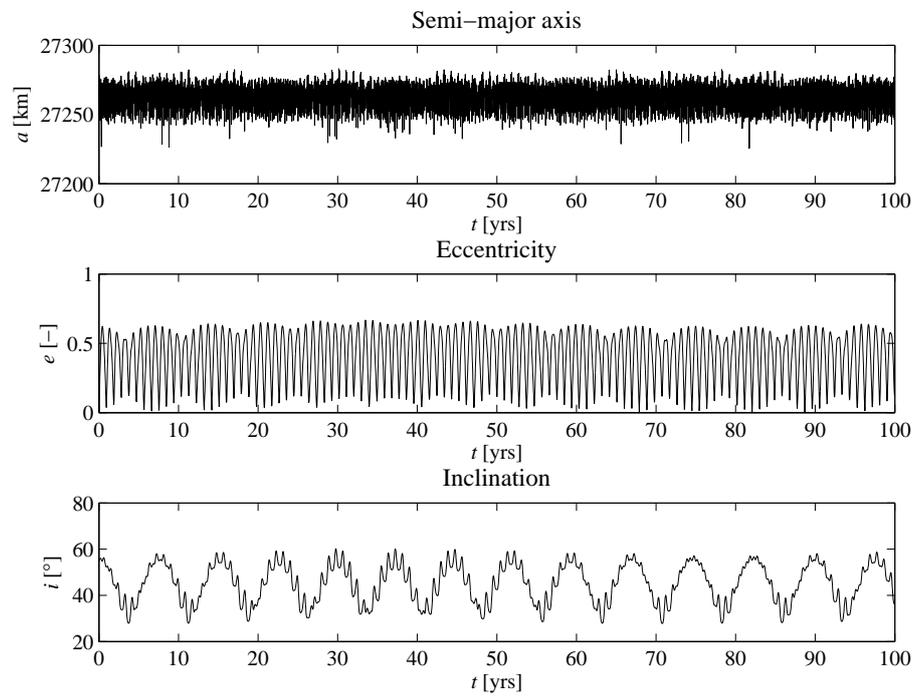
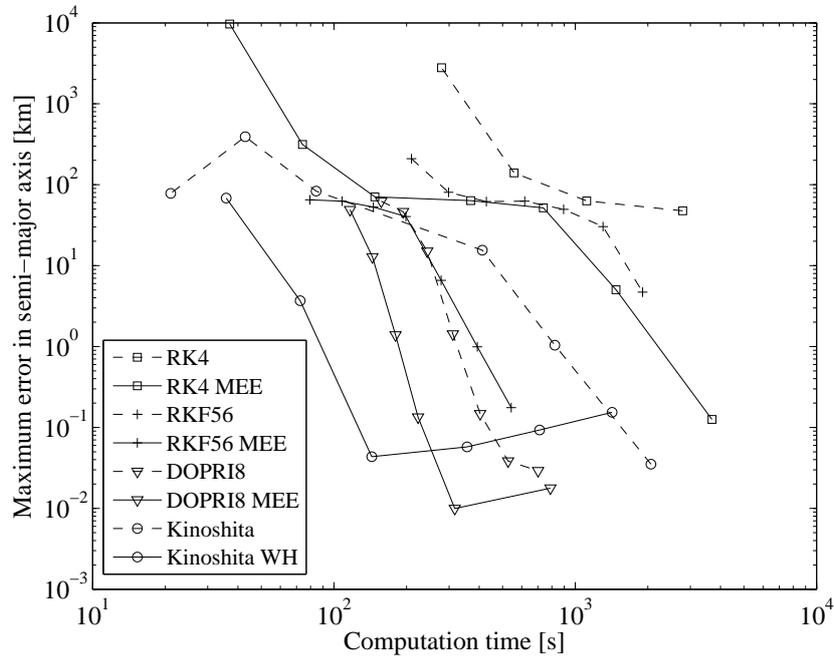
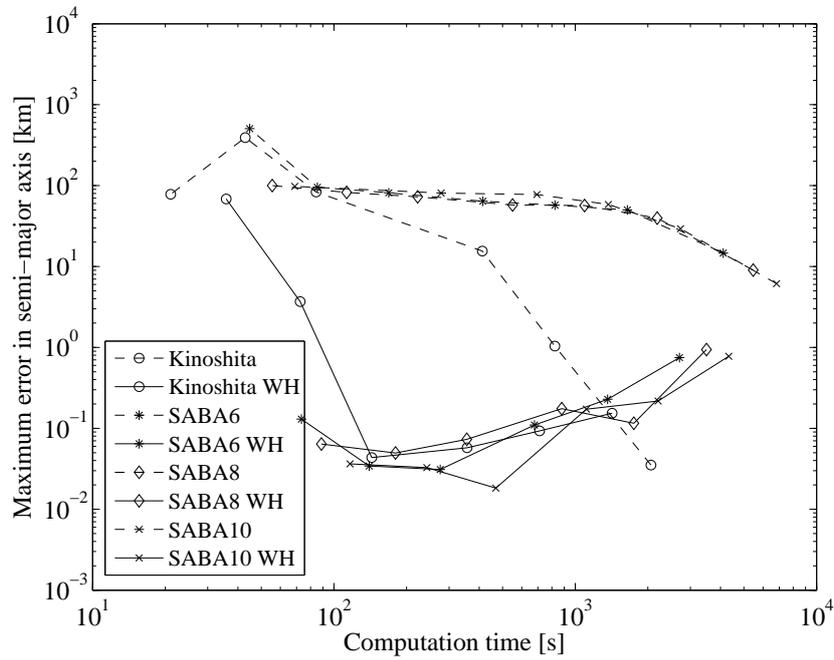


Figure 9.21: The long-term evolution of the orbital elements for simulation case IV.

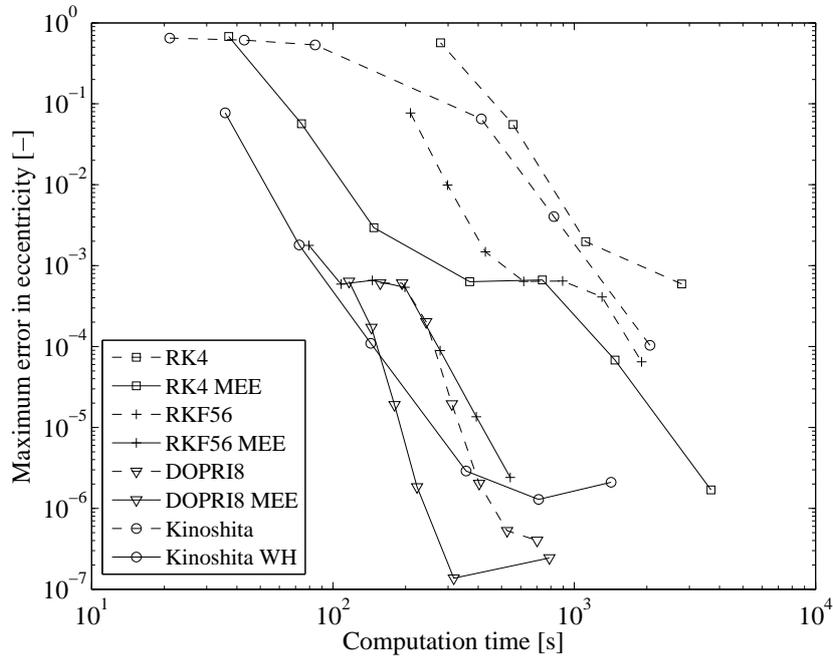


(a) Traditional methods and Kinoshita's method.

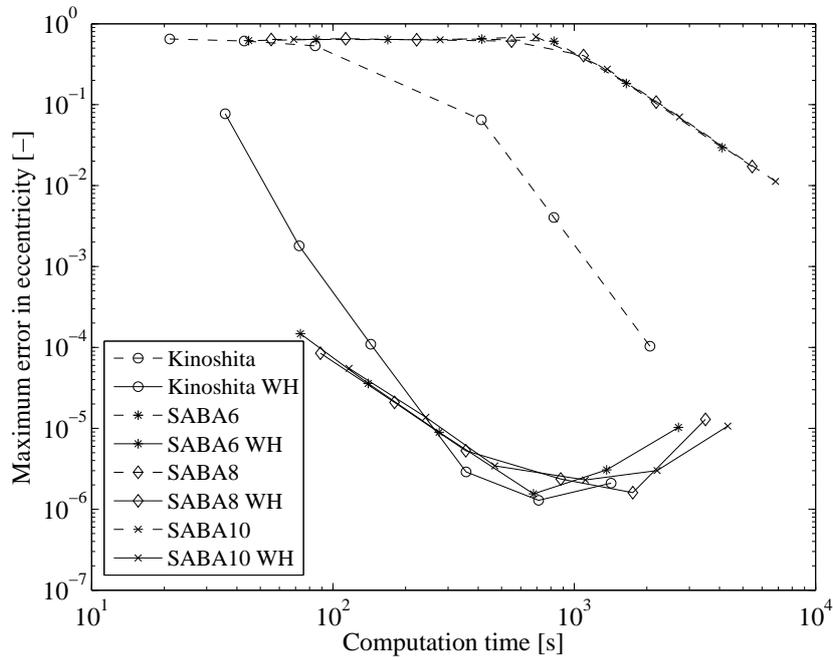


(b) Symplectic methods.

Figure 9.22: The performance regarding the semi-major axis for different computational methods, for a 100-year propagation of a near-circular GPS graveyard orbit of an object with a high area-to-mass ratio.

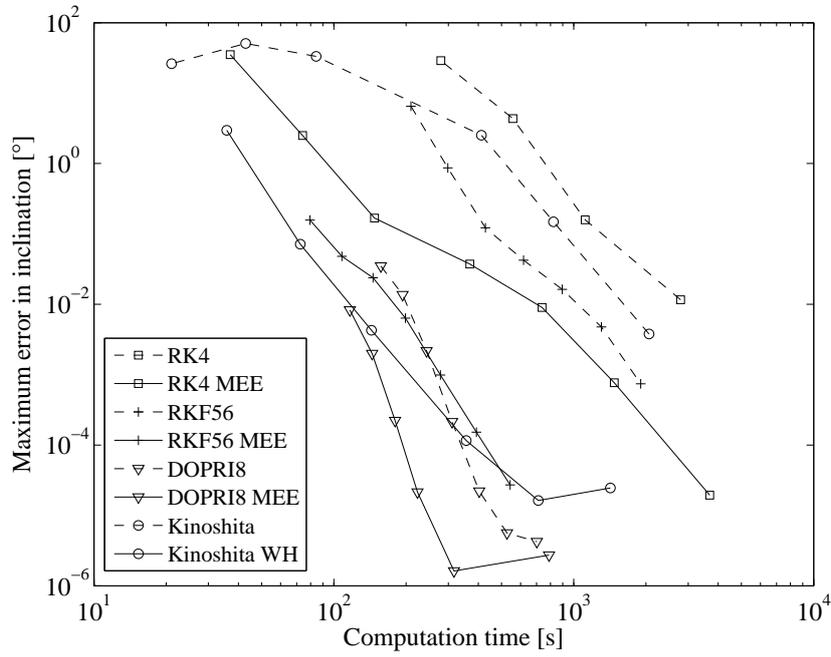


(a) Traditional methods and Kinoshita's method.

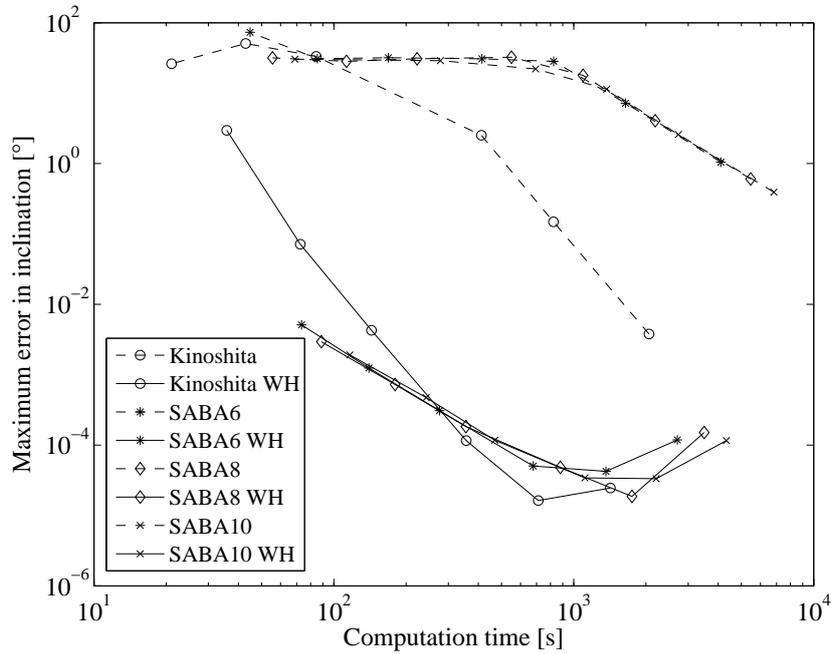


(b) Symplectic methods.

Figure 9.23: The performance regarding the eccentricity for different computational methods, for a 100-year propagation of a near-circular GPS graveyard orbit of an object with a high area-to-mass ratio.



(a) Traditional methods and Kinoshita's method.



(b) Symplectic methods.

Figure 9.24: The performance regarding the inclination for different computational methods, for a 100-year propagation of a near-circular GPS graveyard orbit of an object with a high area-to-mass ratio.

9.5 Discussion

When assessing the performance of different computational methods for a particular application, the specific requirements of that application should be kept in mind. In this case, the application is predicting how the orbits of space debris objects develop over long periods of time and whether or not collision hazards with functional satellites will arise in the future. This specific application has implications for the accuracy that is required for the simulations. Since most graveyard orbits initially have an altitude difference in the order of a few hundred kilometers with respect to active satellite orbits, a simulation accuracy of e.g. 100 km is not acceptable, as that level of accuracy might lead to incorrect conclusions. On the other hand, millimeter-level accuracy is also not required for this particular application.

During the course of the thesis project, it has been established that a maximum in-plane error on the order of a few kilometers is deemed acceptable for predicting potential collision hazards with active satellite constellations. Practically, this means that in the performance plots of the previous sections, errors in semi-major axis of 10^0 km or smaller combined with errors in eccentricity of roughly 10^{-4} or smaller will result in an acceptable in-plane accuracy. The opposite is also true: data points which lie above these error boundaries correspond to simulation runs which are not considered to be accurate enough for this specific application.

Regarding the accuracy of the simulations, it should be noted that a maximum error in semi-major axis of 10^0 km in the performance plots means that during the full 100-year simulation, the prediction of the semi-major axis made using that specific method did not deviate more than 1 km from the reference *at any point* in the 100-year period. For a geostationary orbit, such an error corresponds to a maximum error of roughly 0.0024 % in semi-major axis, which is of course very small. However, as mentioned above, for the intended application, errors in this range are still acceptable, but larger errors are not. Therefore, integration methods are required which maintain a high level of accuracy over long periods of time. In addition, relatively short computation times are desirable, making efficient methods which also satisfy the accuracy requirements the methods of choice. In practice, this implies that methods which have data points in the lower left of the performance plots in Section 9.4 are the best methods for integrating the specific simulation cases.

Case I

Looking at the results for simulation case I, it can be concluded that for the traditional methods, the formulation using Gauss' equations leads to much better performance than the relatively simple Cowell formulation. Across the board, DOPRI8 MEE and RK4 MEE perform best, reaching a high accuracy relatively fast. RKF56 MEE performs somewhat worse in this case, though it still produces moderately accurate results in a very short computation time. Kinoshita's method with T+V splitting performs better than RK4 and RKF56 using the Cowell formu-

lation, but is surpassed by DOPRI8.

With Wisdom-Holman splitting, the performance of Kinoshita's method is comparable with RK4 MEE and DOPRI8 MEE, although its results do not get more accurate if smaller steps are used. This particular behavior of symplectic methods using Wisdom-Holman splitting was discussed in Section 9.4.1 and is present for all methods that use Wisdom-Holman splitting, not just for Kinoshita's method.

Considering the symplectic integrators in specific, Kinoshita WH performs best. Just like in the simulations of the two-body problem of Section 9.3, the performance curves are roughly the same for all $SABA_{2n}$ methods tested ($2n = 6, 8, 10$). Using a higher value for n will generally result in a more accurate prediction, but at the cost of additional computation time. This is not only true for simulation case I, but also for the other simulation cases. Therefore, in this discussion the $SABA_{2n}$ methods will be treated as a single set of methods, rather than as individual methods. For simulation case I, the $SABA_{2n}$ methods perform notably worse than Kinoshita's method, especially when using T+V splitting. If Wisdom-Holman splitting is used, there is less of a difference, though Kinoshita's method still performs better in that case.

Taking into account the above, for integrating a GEO graveyard orbit for an object with a low area-to-mass ratio, RK4 MEE, DOPRI8 MEE or Kinoshita with Wisdom-Holman splitting are considered the best methods, reaching a high accuracy while requiring only a short computation time. Using Cowell's formulation is not recommended, except when using DOPRI8. Using Gauss' equations with modified equinoctial elements, however, is clearly the better option.

Case II

When simulating a GEO graveyard orbit for an object with a high area-to-mass ratio, as was done for test case II, the situation clearly changes. The maximum error over the simulation period increases across the board, with the best methods having a maximum error in semi-major in the order of tens of meters, whereas previously millimeter-level accuracy was reached. This is likely because the system has become much more volatile due to the vastly increased influence of solar radiation pressure, and as a result, has become harder to integrate accurately.

Noteworthy is that while RK4 MEE was considered one of the best methods for the low area-to-mass ratio case, it performs much worse in the situation with a high area-to-mass ratio, only reaching relatively good accuracy when using very small integration steps, and as a result requiring a substantial amount of CPU time. RK4 integration using the Cowell formulation is essentially not usable for the high area-to-mass ratio simulation, as errors on the order of hundreds of km in semi-major axis are reached, even when taking a considerable amount of computation time. DOPRI8 performs very well for this particular case, both when using Gauss' equations and when using the Cowell formulation, although accurate results can still be produced faster using Gauss' equations.

Kinoshita's method also performs quite well when using Wisdom-Holman split-

ting. A similar performance is obtained using the $SABA_{2n}$ methods with Wisdom-Holman splitting. A difference is that when using T+V splitting, Kinoshita's method eventually reaches acceptable accuracy, while the $SABA_{2n}$ methods never reach a sufficient level of accuracy.

Recommended methods for simulation case II are DOPRI8 combined with either the Cowell formulation or Gauss' equations, or any of the symplectic methods with Wisdom-Holman splitting.

Case III

For the integration of a highly eccentric GPS graveyard orbit for an object with a low area-to-mass ratio, which was tested in simulation case III, DOPRI8, DOPRI8 MEE, Kinoshita WH and RKF56 MEE all perform well, with DOPRI8 MEE having a slight edge. The other traditional methods all perform significantly worse, though RK4 MEE can still attain a reasonable level of accuracy, given enough computation time.

Regarding the use of the symplectic methods for simulation case III, it is notable that the $SABA_{2n}$ methods with Wisdom-Holman splitting perform better than Kinoshita's method with the same splitting method, whereas in the previous cases, Kinoshita's method showed superior performance. Hence, for this case, the $SABA_{2n}$ methods are actually preferable over Kinoshita's method.

Methods that can be recommended for simulation case III are DOPRI8, DOPRI8 MEE, RKF56 MEE and the $SABA_{2n}$ methods with Wisdom-Holman splitting. Of these methods, DOPRI8 MEE shows the best performance.

Case IV

Simulation case IV considered the integration of a near-circular GPS graveyard orbit for an object with a high area-to-mass ratio. This simulation case is relatively similar to case II. Both involve the propagation of near-circular orbits for objects with high area-to-mass ratios. The performance plots for both test cases are also relatively similar, although there is somewhat more spread in the performance of the traditional methods for test case IV.

Specifically, RK4 performs worst for case IV, while DOPRI8, DOPRI8 MEE and RKF56 MEE all show good performance. Also, all symplectic integration methods with Wisdom-Holman splitting demonstrate a performance comparable to that of the best traditional methods. Regarding the symplectic methods with T+V splitting, Kinoshita's method can reach good accuracy if sufficient computation time is allowed, while the $SABA_{2n}$ methods will converge to an accurate solution much more slowly when decreasing the step size.

For test case IV, the best overall performance is achieved by DOPRI8 MEE and all of the symplectic integrators with Wisdom-Holman splitting.

Wrapping Up

All things considered, the combination of Gauss' form of Lagrange's planetary equations with DOPRI8 as integrator performs as one of the best, if not the best method in all test cases. The advantage of error control should not be overlooked. Thanks to it, DOPRI8 can be applied relatively safely to any problem without knowing the specific dynamics of the system beforehand. If the accuracy specifications are set properly, DOPRI8 MEE will typically give an accurate prediction of the trajectory that is simulated within a relatively short amount of time.

For the non-adaptive step size methods, the user needs to choose a proper step size him/herself. If too large a step size is chosen, significant errors will be made in the propagation. If, on the other hand, too small a step size is chosen, the computation will be relatively inefficient. DOPRI8, due to its step size control, will essentially ensure an efficient integration. RKF56 is also an adaptive step size method, but its performance for this application is generally worse than DOPRI8's. That being said, RKF56 shows clearly superior performance over RK4 for most test cases.

In general, it can be concluded that the more clever methods, which make use of the knowledge that the two-body part stays constant due to the central gravitational force and that only the perturbations have an effect on the orbital elements, perform significantly better than the more straightforward methods. Specifically, this concerns the formulation using Gauss' equations with modified equinoctial elements for the traditional methods and the Wisdom-Holman splitting method for the symplectic methods, both of which clearly outperform the methods which use the Cowell formulation and the symplectic methods which use T+V splitting.

Methods like DOPRI8 combined with Gauss' equations and symplectic integrators with Wisdom-Holman splitting are more elaborate to implement than simpler methods like RK4 combined with a simple Cowell formulation, but the differences in efficiency are vast. Even when DOPRI8 is used with a Cowell formulation, the integration can be performed around 15 times as fast as with an RK4 integrator, for a similar level of accuracy. Moreover, for systems that are harder to integrate, for example due to a high area-to-mass ratio or a high eccentricity, a simple RK4 implementation using a Cowell formulation will essentially never reach a good accuracy for any feasible computation time. In these cases, more sophisticated methods are certainly required. All in all, the conclusion can be drawn that for performing long-term integrations of trajectories of space debris objects, it is definitely worthwhile to implement more intricate methods over simpler methods, despite the extra programming effort required. The gains in accuracy and efficiency are so significant that simple methods like RK4 combined with a Cowell formulation should essentially never even be considered for use.

When comparing the symplectic methods with the traditional methods, it can be stated that the symplectic methods with Wisdom-Holman splitting show competitive performance with the best traditional methods that were tested. However, the choice of step size is important, and what is a proper step size is dependent

on the characteristics of the system. This makes the symplectic integrators harder to apply to an unknown system than adaptive step size methods such as DOPRI8, which will automatically ensure an accurate and efficient integration process. Adjusting the symplectic integrators in such a way that they can use variable step sizes is unfortunately not an option, as this will result in a clear degradation of accuracy, and more importantly, will make them lose their symplectic properties (Skeel and Biesiadecki, 1994). Although there are ways around this, they require entirely new methods to be constructed with variable step size in mind from the beginning. In addition, because in the Wisdom-Holman splitting method only the perturbations, which are partially non-Hamiltonian, are integrated numerically, it is quite questionable whether newly implemented variable step size symplectic integrators would perform better for this problem than a normal adaptive step size integrator of high order like DOPRI8 combined with Gauss' equations in modified equinoctial elements.

For integrating purely Hamiltonian systems, in this case variants of the two-body problem, the performance of symplectic integrators was found to be excellent. On the other hand, the symplectic integrators did not show a clear advantage over the traditional methods in the simulations of the perturbed two-body problem. It should also be taken into account that because symplectic methods are relatively new, they will generally not be available in existing simulation tools for satellite orbits. Furthermore, they are harder to understand and implement than traditional methods, if one is not familiar with the underlying mathematics beforehand.

Generally speaking, for integrating the solar system or simulating the three-body problem, symplectic integrators may be noticeably more efficient than most traditional methods. However, for propagating perturbed satellite and debris orbits, such as the ones considered in this thesis report, the advantage of using symplectic integrators is limited. Therefore, for these applications it is hard to recommend symplectic integration methods over established methods such as DOPRI8.

Chapter 10

Long-Term Debris Orbit Predictions

At the end of their operational lifetimes, satellites in the higher orbital regions are generally put in graveyard orbits, in order to prevent possible collisions with active satellites. The guidelines for graveyard orbits for GEO and GPS satellites, as specified by various space agencies, were listed in Tables 2.1 and 2.2.

Assuming that satellites are re-orbited to graveyard orbits, it is interesting to consider how these orbits develop over time. Therefore, long-term simulations are carried out in this chapter to predict how the orbits of debris objects originating from graveyard orbits evolve, and whether collision possibilities with active satellites will arise in the future.

10.1 Introduction

Ideally every re-orbited satellite would stay well removed from the region in which the operational satellites are located. Even so, events such as in-orbit explosions do occur, leading to debris objects with a wide range of characteristics. Hence, even if intact satellites in specific graveyard orbits would stay safely away from the active satellite constellations, these new debris objects could still be dangerous. As was mentioned in Section 2.5, the only physical property of debris objects that has an effect on the way the perturbing forces influence their trajectories is the effective area-to-mass ratio, $C_R \cdot A/m$. For an intact satellite, this factor has a value of roughly $0.02 \text{ m}^2/\text{kg}$. However, objects such as explosion fragments and flakes of paint can have much higher area-to-mass ratios, which dramatically increases the effect solar radiation pressure has on their trajectories. To take these objects into account as well, simulations will also be carried out for objects with effective area-to-mass ratios of $1 \text{ m}^2/\text{kg}$ and $40 \text{ m}^2/\text{kg}$.

This chapter should not be viewed as an in-depth assessment of all different options for graveyard orbits of geostationary satellites and satellites in the GNSS

constellations. Rather, it should provide insight into the long-term developments of a selection of graveyard orbits possible under the guidelines formulated by various space agencies (cf. Tables 2.1 and 2.2). As part of that, it will become clear how the various perturbing forces influence the orbits of objects with a range of area-to-mass ratios, as well as how the effects of these forces differ for the orbital regimes that are considered.

In the performance comparison of the previous chapter, the DOPRI8 method in combination with Gauss' form of Lagrange's planetary equations in modified equinoctial elements was determined to be the most efficient and robust method for performing long-term propagations of space debris orbits. Accordingly, this method is used for carrying out the simulations for this chapter. Also, to account for any effects that might occur on longer timescales, the integration time for the simulations is set to 400 years.

In the next sections, the results of the simulations will be presented. In addition to figures showing the evolution of the orbital elements for the simulations corresponding to different area-to-mass ratios, plots of the perigee and apogee altitudes over time will also be provided. These plots can be used to determine whether the regions of space in which the active satellites reside are crossed by the trajectories of the debris objects that are simulated, and hence, whether collisions would be possible. For convenience, these plots also include the altitudes of the relevant satellite constellations. At the end of the chapter, a number of important conclusions will be drawn based on the simulation results presented here.

10.2 GEO Graveyard Orbits

In the end-of-life guidelines for satellites in the geostationary region listed in Table 2.1, three of the four space agencies included the area-to-mass ratio in the computation for the minimum change in altitude required for the graveyard orbit. As a result, satellites that are affected more by solar radiation pressure will be given a higher graveyard orbit, in order to compensate for the accompanying orbital fluctuations. It should be kept in mind, however, that the guidelines are only for intact satellites, which have the ability to be re-orbited at end-of-life. Generally speaking, these satellites have relatively low area-to-mass ratios, leading to suggested re-orbit altitudes of roughly 200 to 300 km above GEO.

Accordingly, a semi-major axis that is 300 km larger than the nominal radius of the geostationary orbit will be used for the simulations of the GEO graveyard orbits of this section. The full list of initial conditions can be found in Table 10.1.

Table 10.1: The initial conditions for the simulation results in this section.

Starting epoch	
Jan. 1, 2014 00:00:00	
Altitude	Initial value
Apogee	36,128 km
Perigee	36,044 km
Orbital element	Initial value
a	42,464 km
e	0.001
i	0.1°
ω	0°
Ω	0°
θ	0°

The simulation results for the long-term evolution of the GEO graveyard orbits of objects with effective area-to-mass ratios of 0.02, 1 and 40 m^2/kg are shown in Figures 10.1 to 10.3, respectively. The corresponding perigee and apogee altitudes over time are displayed in Figure 10.4. The results of these simulations, as well as those of the simulations presented in the next sections, will be discussed at the end of the chapter.

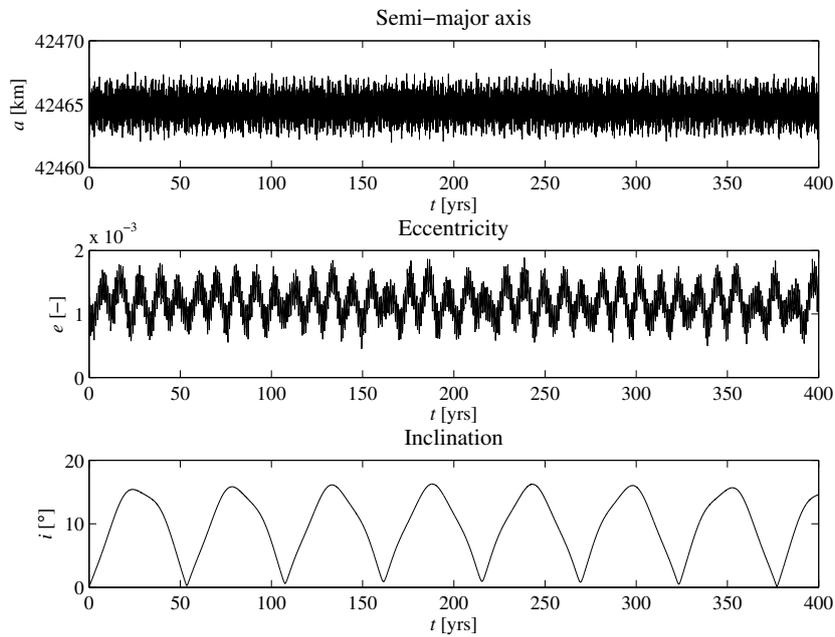


Figure 10.1: The long-term development of a GEO graveyard orbit for an object with a $C_R \cdot A/m$ value of 0.02 m^2/kg .

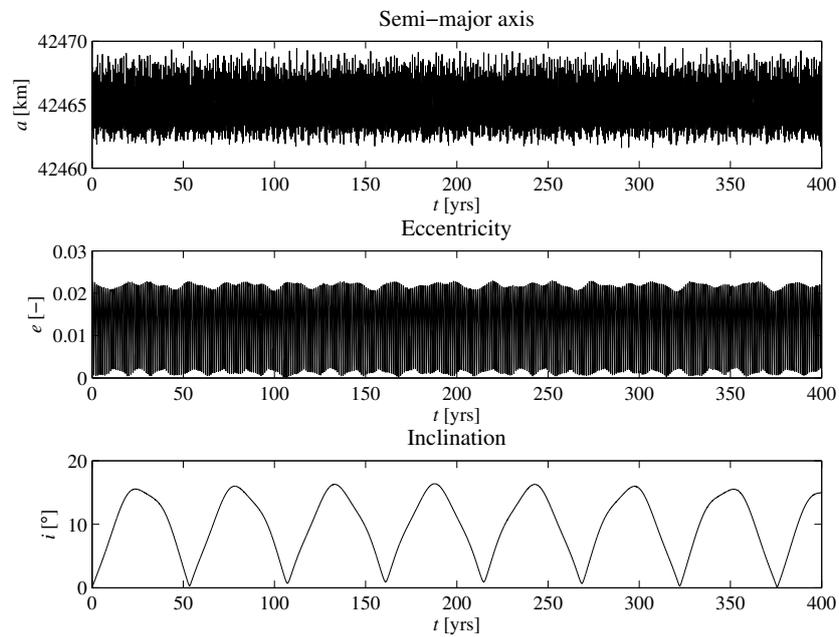


Figure 10.2: The long-term development of a GEO graveyard orbit for an object with a $C_R \cdot A/m$ value of $1 \text{ m}^2/\text{kg}$.

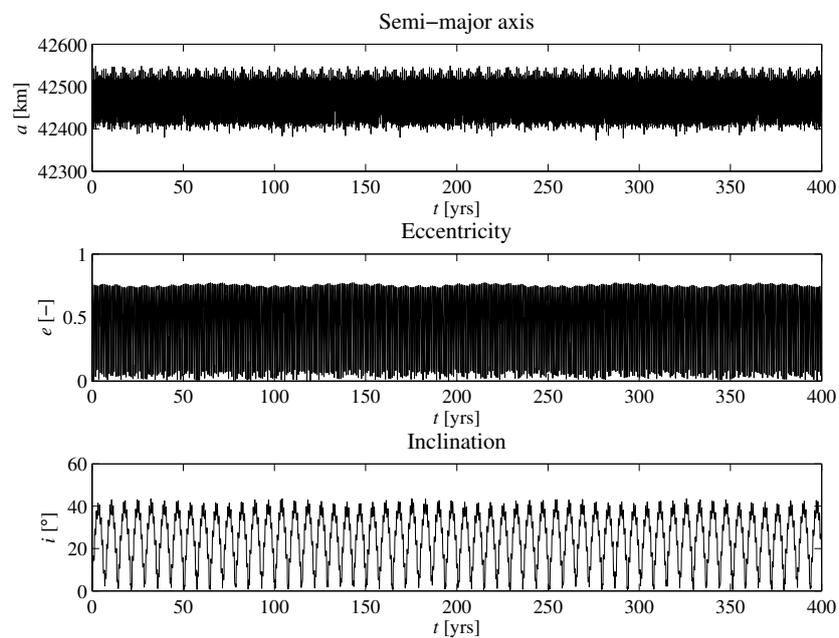


Figure 10.3: The long-term development of a GEO graveyard orbit for an object with a $C_R \cdot A/m$ value of $40 \text{ m}^2/\text{kg}$.

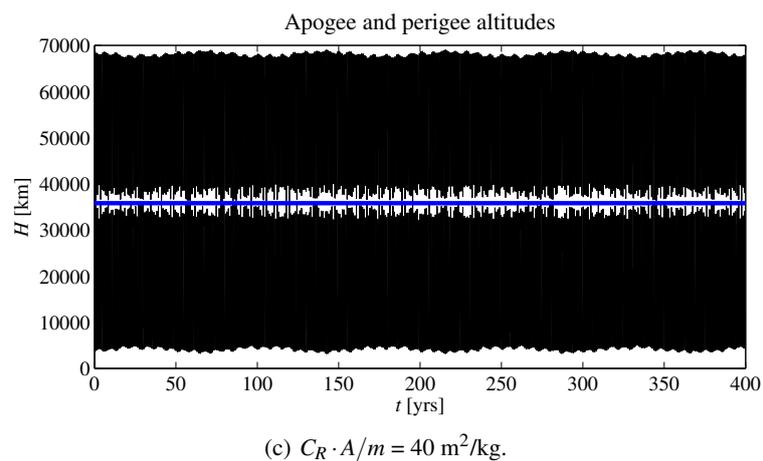
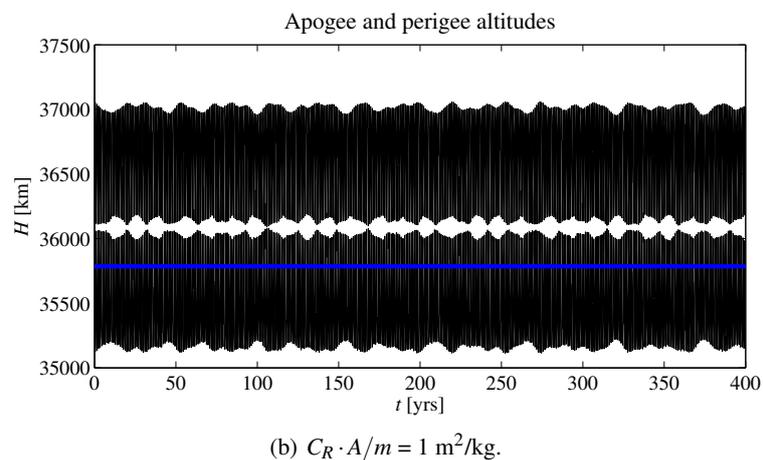
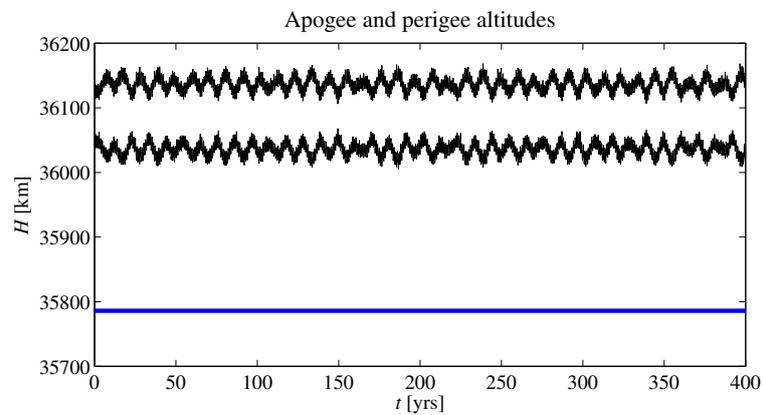


Figure 10.4: The apogee and perigee altitudes for debris objects with various area-to-mass ratios starting in GEO graveyard orbits. The blue line indicates the nominal altitude of geostationary satellites.

10.3 GNSS Graveyard Orbits

The end-of-life guidelines for satellites in 12-hour circular orbits, such as GPS satellites, allow for more different options for graveyard orbits than the guidelines for GEO satellites (see Table 2.2). To take into account the most distinct options, two boundary cases for eccentric orbits will be considered, as well as a more basic, near-circular option.

10.3.1 Low Altitude Eccentric Orbits

The initial conditions for the simulation results in this section correspond to the most eccentric GPS graveyard orbit that is possible under the NASA end-of-life guidelines. The entire orbit lies below the nominal GPS altitude and has initial perigee and apogee altitudes of 2,500 and 19,900 km, respectively. The full initial conditions are listed in Table 10.2.

Table 10.2: The initial conditions for the simulation results in this section.

Starting epoch	
Jan. 1, 2014 00:00:00	
Altitude	Initial value
Apogee	19,900 km
Perigee	2,500 km
Orbital element	Initial value
a	17,578 km
e	0.4949
i	55 °
ω	0 °
Ω	0 °
θ	0 °

The long-term evolution of orbits with these initial conditions for objects with a range of area-to-mass ratios are shown in Figures 10.5 to 10.7. The related perigee and apogee altitudes are provided in Figure 10.8.

It should be noted that as a result of the low initial perigee, the rapid growth in eccentricity for a debris object with an effective area-to-mass ratio of 40 m²/kg leads to decay within a short amount of time. Therefore, the timescales for the corresponding plots (Figures 10.7 and 10.8(c)) are much shorter than those for the other plots.

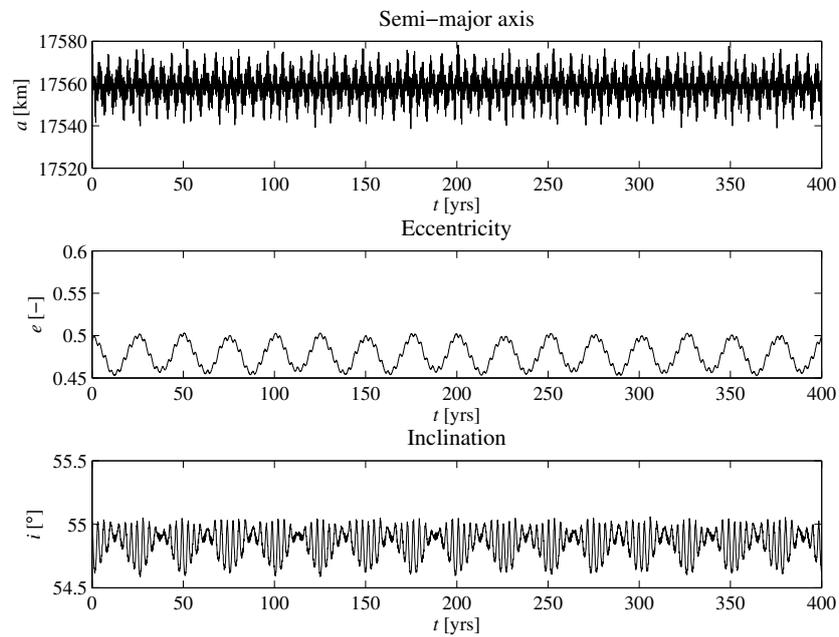


Figure 10.5: The long-term development of a low altitude eccentric GPS graveyard orbit for an object with a $C_R \cdot A/m$ value of $0.02 \text{ m}^2/\text{kg}$.

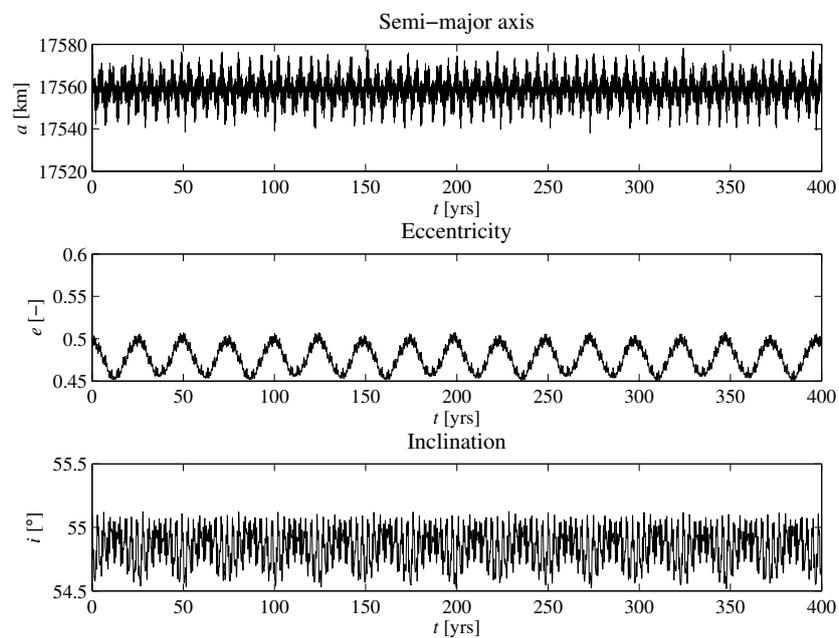


Figure 10.6: The long-term development of a low altitude eccentric GPS graveyard orbit for an object with a $C_R \cdot A/m$ value of $1 \text{ m}^2/\text{kg}$.

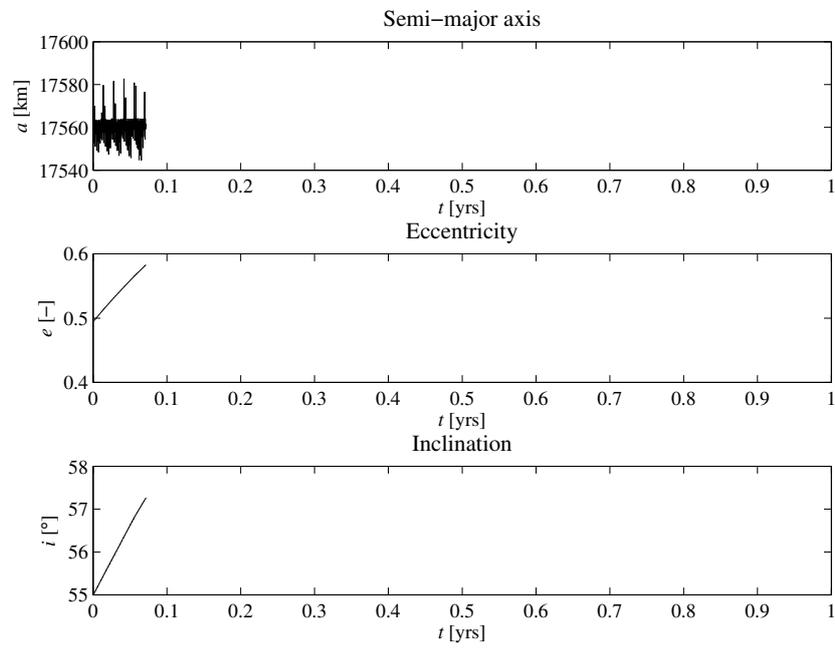
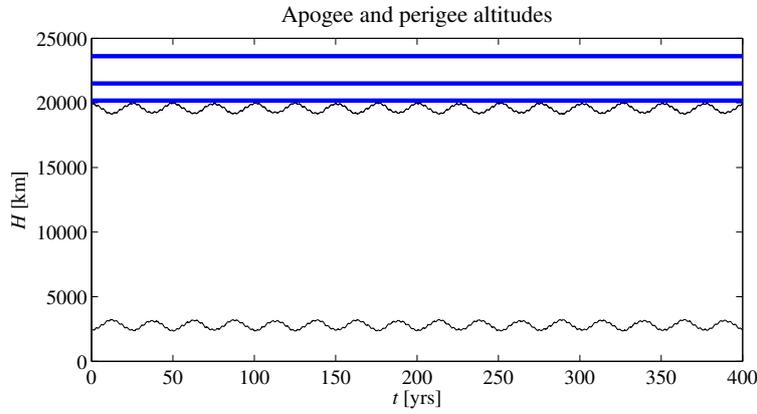
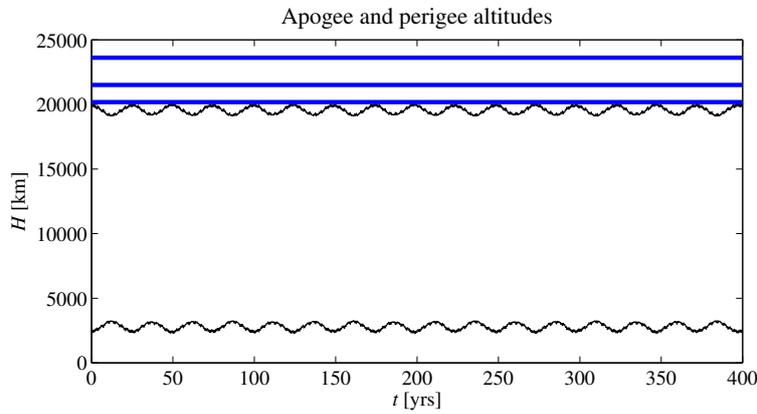


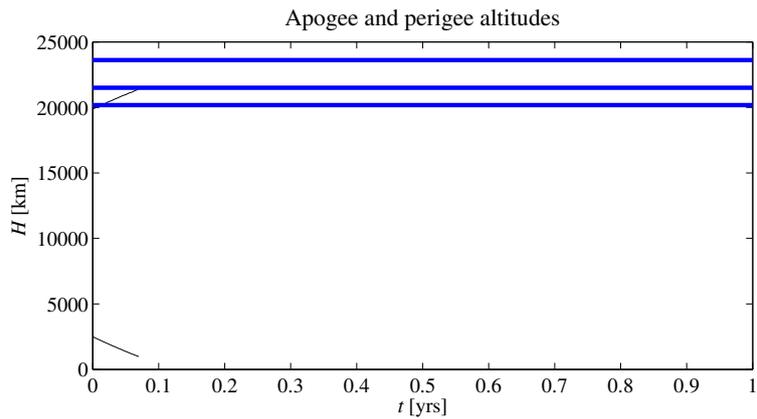
Figure 10.7: The development of a low altitude eccentric GPS graveyard orbit for an object with a $C_R \cdot A/m$ value of $40 \text{ m}^2/\text{kg}$. Due to the influence of solar radiation pressure the object will enter the atmosphere within a month.



(a) $C_R \cdot A/m = 0.02 \text{ m}^2/\text{kg}$.



(b) $C_R \cdot A/m = 1 \text{ m}^2/\text{kg}$.



(c) $C_R \cdot A/m = 40 \text{ m}^2/\text{kg}$.

Figure 10.8: The apogee and perigee altitudes for debris objects with various area-to-mass ratios starting in low altitude eccentric GPS graveyard orbits. Going from lower altitude to higher altitude, the blue lines indicate the nominal altitudes of satellites of the GPS, Beidou and Galileo constellations, respectively.

10.3.2 High Altitude Eccentric Orbits

While in the previous section eccentric graveyard orbits below the nominal GPS altitude were considered, in this section the long-term developments of eccentric graveyard orbits that lie completely above the GPS regime are considered. The initial conditions used for the simulations are listed in Table 10.3. Specifically, they result in the most eccentric graveyard orbits above the nominal GPS altitude possible under the NASA end-of-life guidelines.

Table 10.3: The initial conditions for the simulation results in this section.

Starting epoch	
Jan. 1, 2014 00:00:00	
Altitude	Initial value
Apogee	35,288 km
Perigee	20,500 km
Orbital element	Initial value
a	34,272 km
e	0.2157
i	55 °
ω	0 °
Ω	0 °
θ	0 °

The results of the long-term simulations with these initial conditions for objects with different area-to-mass ratios are shown in Figures 10.9 to 10.11. The corresponding perigee and apogee altitudes are included in Figure 10.12.

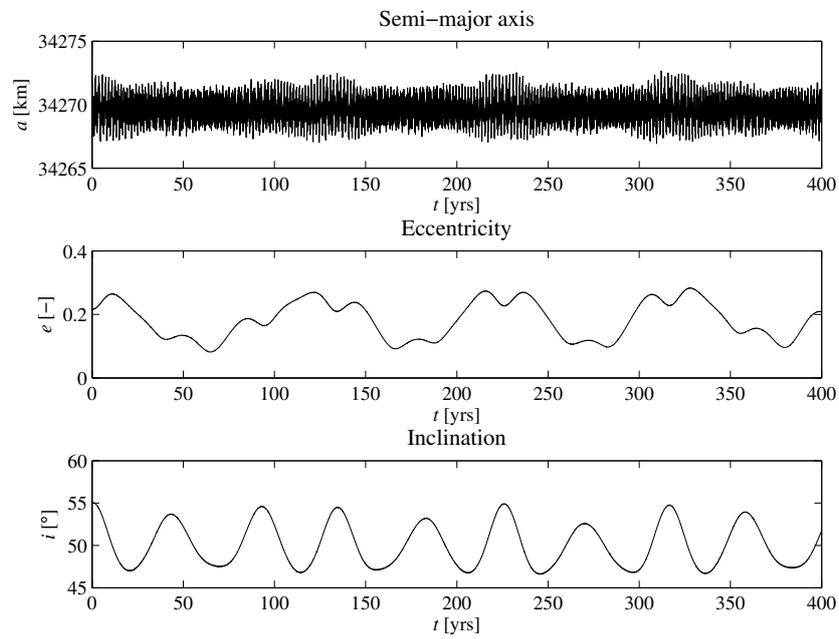


Figure 10.9: The long-term development of a high altitude eccentric GPS graveyard orbit for an object with a $C_R \cdot A/m$ value of $0.02 \text{ m}^2/\text{kg}$.

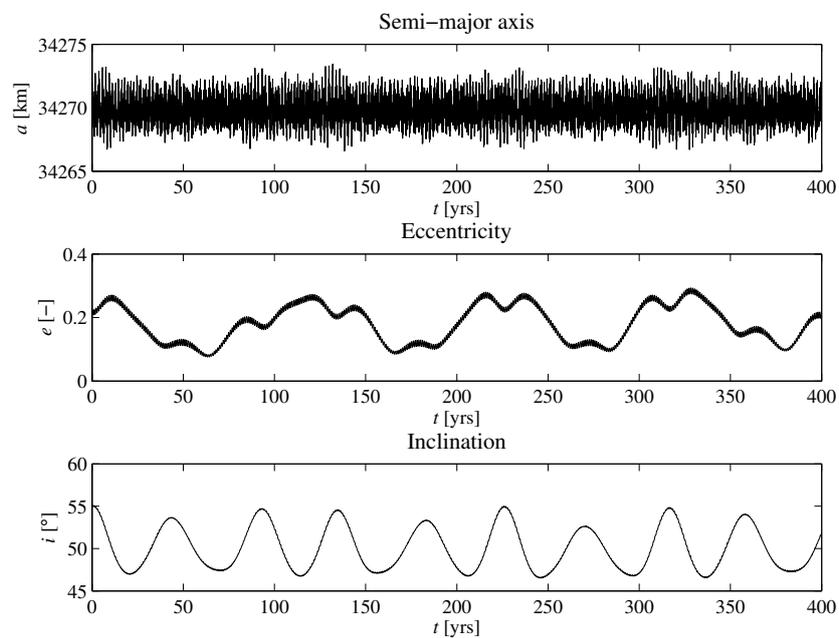


Figure 10.10: The long-term development of a high altitude eccentric GPS graveyard orbit for an object with a $C_R \cdot A/m$ value of $1 \text{ m}^2/\text{kg}$.

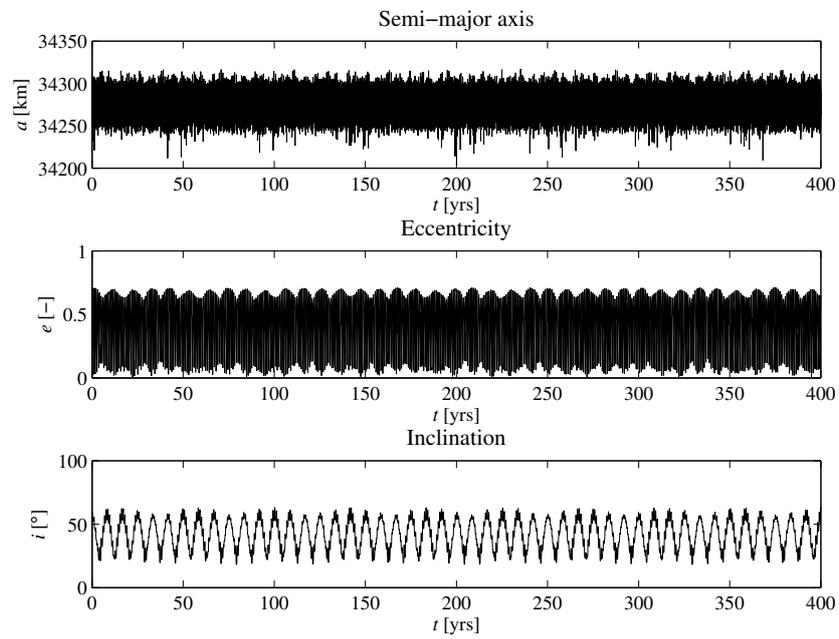


Figure 10.11: The long-term development of a high altitude eccentric GPS graveyard orbit for an object with a $C_R \cdot A/m$ value of $40 \text{ m}^2/\text{kg}$.

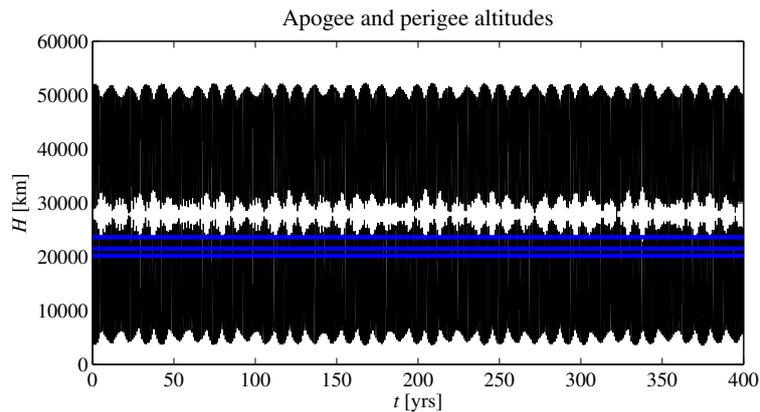
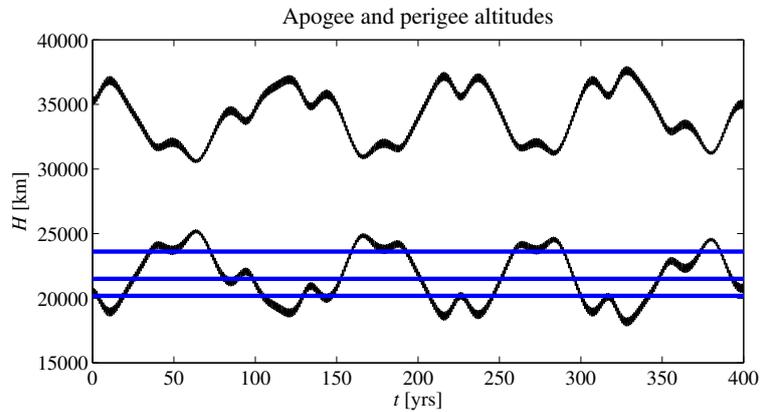
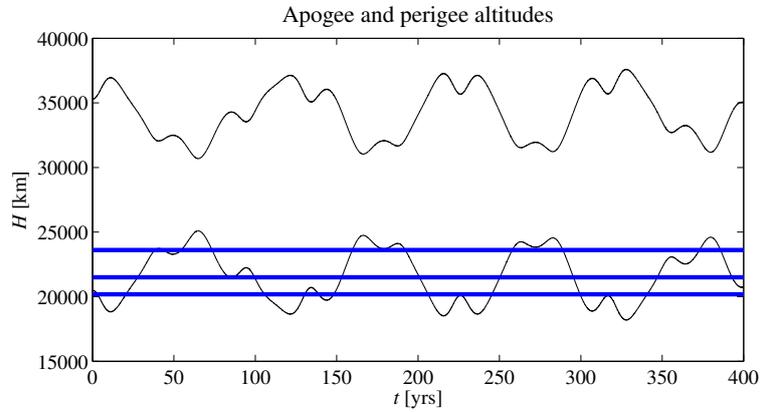


Figure 10.12: The apogee and perigee altitudes for debris objects with various area-to-mass ratios starting in high altitude eccentric GPS graveyard orbits. Going from lower altitude to higher altitude, the blue lines indicate the nominal altitudes of satellites of the GPS, Beidou and Galileo constellations, respectively.

10.3.3 Near-Circular Orbits

In this final set of simulations, near-circular GPS graveyard orbits are considered with a semi-major axis that is 700 km larger than the semi-major axis of a nominal GPS orbit. Because of this, the altitudes used by the graveyard orbits lie almost exactly in between the GPS and Beidou constellations. The initial conditions for this situation are listed in Table 10.4.

Table 10.4: The initial conditions for the simulation results in this section.

Starting epoch	
Jan. 1, 2014 00:00:00	
Altitude	Initial value
Apogee	20,909 km
Perigee	20,855 km
Orbital element	Initial value
a	27,260 km
e	0.001
i	55 °
ω	0 °
Ω	0 °
θ	0 °

The long-term evolution of the orbital elements for debris objects with different area-to-mass ratios are shown in Figures 10.13 to 10.15. The accompanying plots of the perigee and apogee altitudes are provided in Figure 10.16.

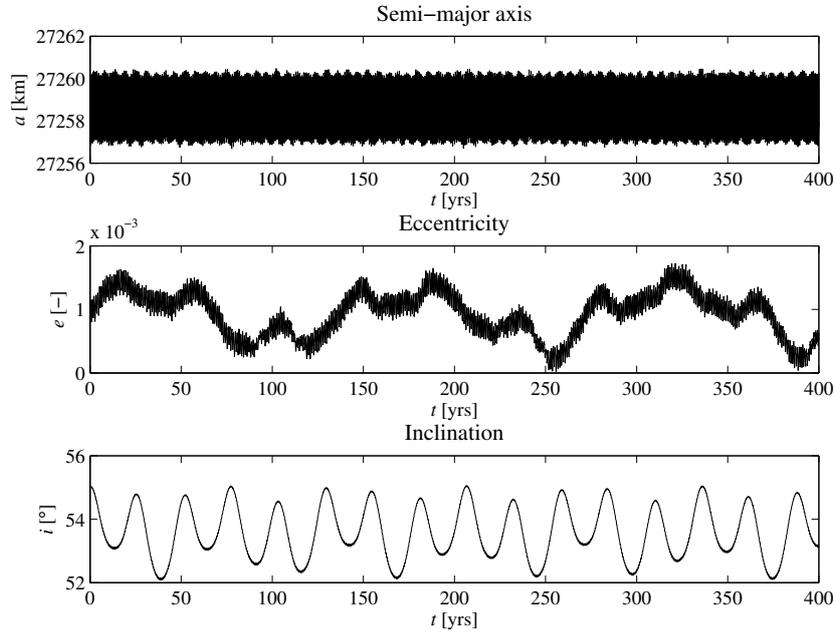


Figure 10.13: The long-term development of a near-circular GPS graveyard orbit for an object with a $C_R \cdot A/m$ value of $0.02 \text{ m}^2/\text{kg}$.

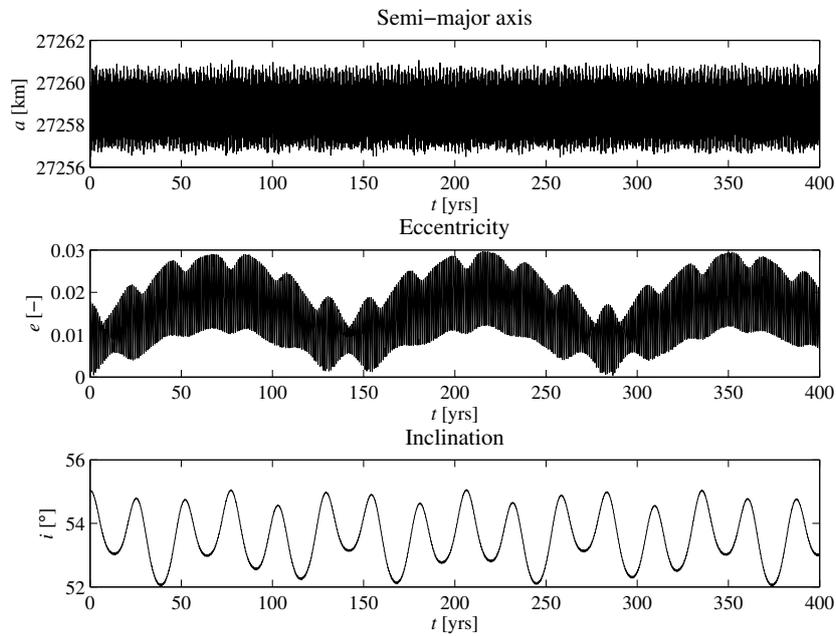


Figure 10.14: The long-term development of a near-circular GPS graveyard orbit for an object with a $C_R \cdot A/m$ value of $1 \text{ m}^2/\text{kg}$.

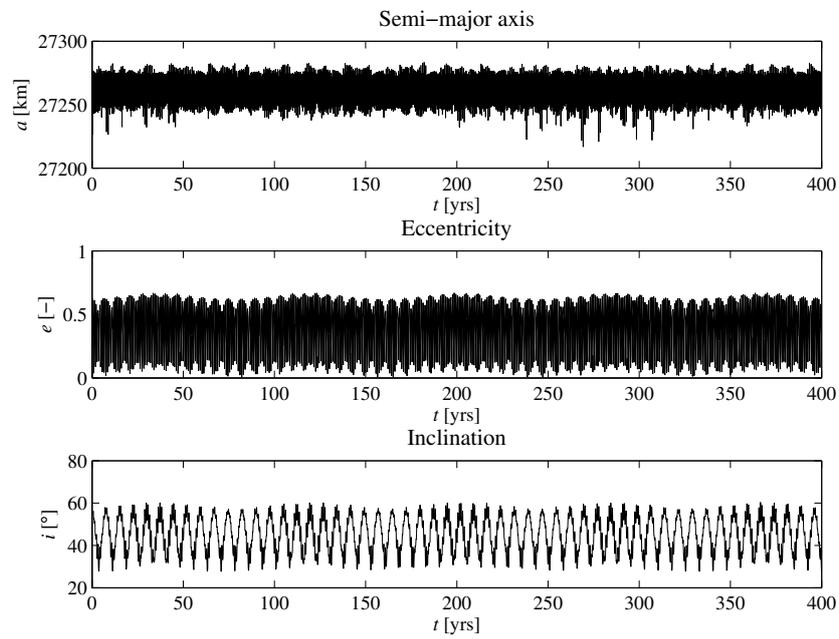


Figure 10.15: The long-term development of a near-circular GPS graveyard orbit for an object with a $C_R \cdot A/m$ value of $40 \text{ m}^2/\text{kg}$.

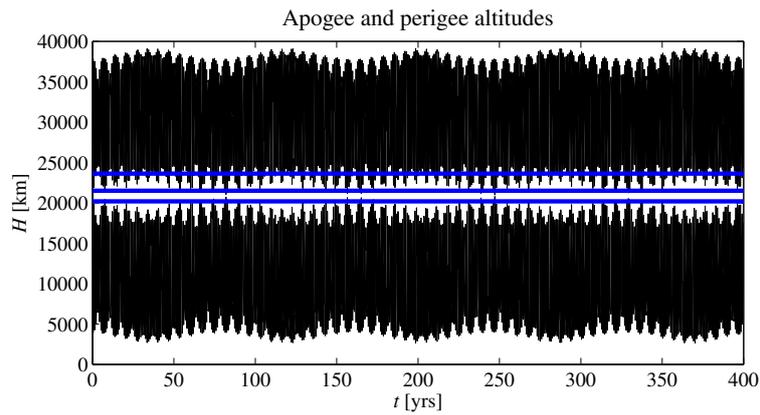
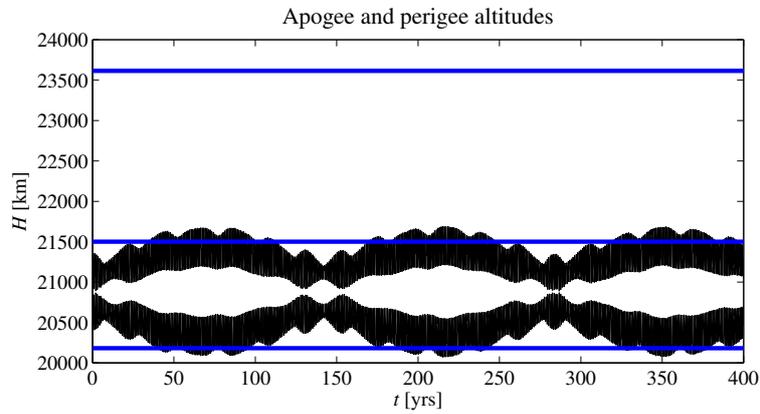
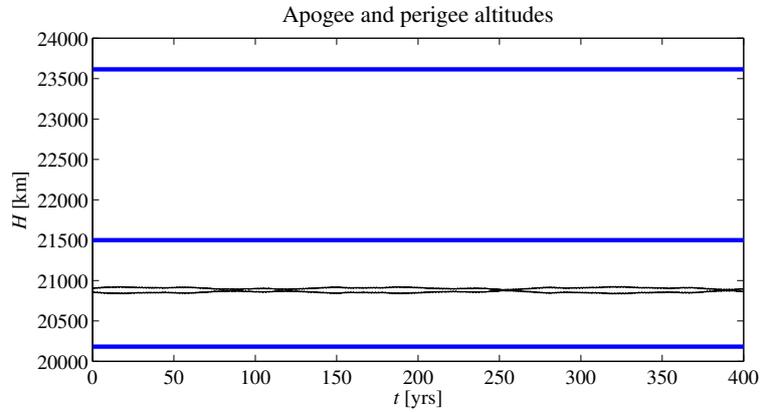


Figure 10.16: The apogee and perigee altitudes for debris objects with various area-to-mass ratios starting in near-circular GPS graveyard orbits. Going from lower altitude to higher altitude, the blue lines indicate the nominal altitudes of satellites of the GPS, Beidou and Galileo constellations, respectively.

10.4 Discussion

GEO Graveyard Orbits

As can be seen in Figure 10.1, the orbit of a satellite starting in a typical GEO graveyard orbit experiences only relatively small variations in semi-major axis and eccentricity. As was already seen in Section 8.5, the inclination shows a periodic behavior with a maximum inclination of roughly 15 degrees and a period of about 53 years. This behavior in the inclination is also present for a debris object with an effective area-to-mass ratio of $1 \text{ m}^2/\text{kg}$. However, for an object with a $C_R \cdot A/m$ value of $40 \text{ m}^2/\text{kg}$, the pattern is clearly different. Specifically, the oscillation both has a shorter period and a higher maximum inclination. This same behavior was also found by Anselmo and Pardini (2008), who determined that the 53-year periodic behavior with maxima of about 15 degrees caused by the luni-solar perturbations was present for $C_R \cdot A/m$ values up to roughly $1 \text{ m}^2/\text{kg}$ and noted that higher area-to-mass ratios would lead to shorter periods and higher peaks, as a result of an increasingly large perturbing acceleration due to solar radiation pressure.

For higher area-to-mass ratios, the variations in semi-major axis also become larger, but it is the behavior of the eccentricity under the influence of solar radiation pressure that is noteworthy. The eccentricity experiences yearly oscillations, with much smaller long-period modulations. Though the initial graveyard orbit only has a small eccentricity of 0.001, eccentricities of over 0.02 are reached for objects with $C_R \cdot A/m = 1 \text{ m}^2/\text{kg}$, while eccentricity values of up to 0.75 are attained for objects with $C_R \cdot A/m = 40 \text{ m}^2/\text{kg}$. These results are consistent with those of Anselmo and Pardini (2008). The oscillations in eccentricity lead to large fluctuations in the perigee and apogee altitudes, as can be observed in Figure 10.4. Consequently, collision hazards with active geostationary satellites start to arise for objects with higher area-to-mass ratios. The simulation results show that for the graveyard orbit considered in Section 10.2, debris objects with effective area-to-mass ratios lower than $1 \text{ m}^2/\text{kg}$ already start to cross the geostationary ring, which could potentially result in collisions with active satellites. Debris objects with substantially higher area-to-mass ratios will also cross the altitude regimes of MEO satellites and potentially even LEO satellites.

All in all, the simulations for the GEO region indicate that for intact satellites, a typical graveyard orbit which follows the guidelines formulated by the space agencies will impose no collision hazards with active satellites in the geostationary ring, not even after long periods of time. This conclusion is confirmed by the research of Van Kints (2005). Moreover, other initial conditions that might have an influence on the orbital development of the graveyard orbit were found to be of minor influence in a sensitivity study by Lewis et al. (2004). Hence, the former conclusion regarding the absence of collision hazards related to graveyard orbits formulated under the previously mentioned end-of-life guidelines can be expected to be true in general.

On the other hand, debris objects with moderate to high area-to-mass ratios will

in fact impose a danger to active geostationary satellites. The effect of solar radiation pressure on susceptible pieces of debris is so large that even if a much higher graveyard orbit would be chosen, objects with sufficiently high area-to-mass ratios would still be able to reach the geostationary ring, and potentially other regions of space as well.

GNSS Graveyard Orbits

The conclusions regarding the graveyard orbits for satellites in the GNSS region are partially the same as those for satellites in the GEO region. In both cases, objects with higher area-to-mass ratios will experience large oscillations in eccentricity under the influence of solar radiation pressure, leading to collision possibilities with active satellites. However, the simulations of the GNSS graveyard orbits also revealed other dynamics, which can have significant implications for the suitability of certain graveyard orbits.

The simulations of the low-altitude eccentric orbits of Section 10.3.1 show periodic variations in the eccentricity with a maximum variation of about 0.05, present even for objects with low area-to-mass ratios. Although these variations in eccentricity are significant, the maximum apogee altitudes resulting from them remains near the boundaries of what can be considered dangerous for active GPS satellites. In the case of a very high area-to-mass ratio, the low initial perigee of the graveyard orbit results in rapid decay of the debris object.

In the simulations of the high-altitude eccentric orbits (Section 10.3.2), the eccentricity displays peculiar behavior in the form of long-period variations with minimum and maximum values of approximately 0.1 and 0.3. Furthermore, significant periodic variations in the inclination are found. The plots of the perigee and apogee altitudes show that even for low area-to-mass ratios, the altitude regimes of the GPS, Beidou and Galileo constellations will all be crossed by the simulated debris objects, even though the inclinations of these objects vary as well.

For the near-circular orbits considered in Section 10.3.3, the trajectories of the simulated debris objects show more desirable behavior. Although the inclination varies with roughly three degrees, the eccentricity remains relatively stable, only showing the familiar dependence on the area-to-mass ratio. Because the initial orbit was situated almost exactly in between the GPS and Beidou constellations, a growth in eccentricity results in perigee and apogee altitudes that are removed almost equal distances from both constellations. The simulations point out that the altitude regimes of the active satellites will be crossed only by debris objects with effective area-to-mass ratios of roughly $1 \text{ m}^2/\text{kg}$ and higher. Hence, for intact GPS satellites the graveyard orbit with the initial conditions of the simulations will be safe, even on long timescales.

After investigation, the remarkable variations in eccentricity for the eccentric simulation cases were found to be caused by the combined effects of the luni-solar perturbations and the irregularities in the Earth's gravity field. Apparently, resonances induced by these perturbing forces arise in specific situations.

These phenomena in the GNSS region have been the topic of quite some recent research, for example by Ineichen et al. (2003), Rossi (2008) and Deleflie et al. (2011). It turns out that the orbits in the region of space where the GNSS constellations are located can suffer from inclination-dependent luni-solar resonances. These resonances are mainly responsible for the secular developments in eccentricity and strongly influence the stability of orbits near the GNSS constellations. In the paper by Rossi, it is emphasized that the initial conditions have a large effect on the long-term development of disposal orbits. Therefore, a careful selection of the orbital parameters is advised, particularly of the initial eccentricity and resonant angle $2\omega + \Omega$.

Which resonances play a role and when is a topic of study on its own. Many papers have been published on this topic, and it would be beyond the scope of this thesis project to extensively investigate the resonant dynamics of the system. In addition, resonances are often investigated using analytical relations, as it can be easier to determine when they arise using such methods than by performing a large number of relatively time-intensive numerical simulations. Nevertheless, any resonances will in fact be taken into account in the predictions made using the simulation code developed for this project, granted that the resonances are caused by perturbing forces included in the force model. Consequently, the simulation code can accurately predict the long-term developments of orbits starting from specified initial conditions, but may not necessarily be the best tool to study the existence and causes of resonances with.

Conclusions

To conclude, it can be stated that the suggested GEO graveyard orbits lead to desirable behavior over long periods of time, in which the defunct satellites will never pose a collision hazard for the active geostationary satellites. For the graveyard orbits in the GNSS region, resonances can play an important role. Because these resonances are heavily dependent on the initial conditions of the orbit, it would be recommended to carefully consider the different options for graveyard orbits for satellites in this region on a case-by-case basis.

Regarding GEO and GNSS graveyard orbits alike, it can be concluded that debris objects with high area-to-mass ratios can be hazardous for active satellites, no matter where these objects originate. Even if graveyard orbits are chosen that are farther removed from the active satellite constellations, debris objects with sufficiently high area-to-mass ratios will still find their way to the orbital regimes used by the operational satellites, as a result of large increases in eccentricity caused by solar radiation pressure. Accordingly, it is recommended to focus on preventing high area-to-mass ratio debris from being created at all.

Conclusions and Recommendations

11.1 Conclusions

Simulation Code

A simulation code was developed for performing long-term propagations of the orbits of debris objects in the GEO and GNSS regions of space, and, most importantly, for assessing the performance of different computational methods for said application.

To this end, the various perturbing forces acting on satellites were considered and their relative magnitudes in the GEO and GNSS regions of space were compared. It turned out that for the aforementioned regions, by far the most important perturbing accelerations are caused by irregularities in the gravity field of the Earth, the third-body perturbations due to the Sun and the Moon, and the force due to solar radiation pressure. These perturbing forces are also the ones used in the majority of research papers on debris prediction above LEO. Accordingly, these particular forces were included in the force model of the simulation code.

A range of different computational methods were implemented in the simulation code. The traditional methods that were implemented include the classic RK4 method and the adaptive step size methods RKF56 and DOPRI8. These methods all have the option to be used in combination with either Cowell's method or Gauss' form of Lagrange's planetary equations. In the last formulation, modified equinoctial elements were used in order to prevent singularities.

The simulation code also includes a number of symplectic methods, in the form of Kinoshita's method and the set of SABA_{2n} methods. These integration methods can be combined with either of two splitting methods for the Hamiltonian: splitting into kinetic and potential energy components, or Wisdom-Holman splitting, in which the Hamiltonian is split into a two-body part and a perturbations part.

The complete simulation code was tested extensively. In the testing phase, it was verified that the simulation code produces long-term simulation results consistent with those presented in recent research papers on space debris.

Performance of Computational Methods

An extensive comparison between the different computational methods has been made regarding their performance for carrying out long-term simulations of space debris trajectories. Overall, the best method was found to be the DOPRI8 method combined with Gauss' form of Lagrange's planetary equations in modified equinoctial elements. In each of the test cases, this particular combination performed as one of the best, if not the best method. The fact that the step size is adapted based on the dynamics of the system ensures that the integration is performed efficiently. Moreover, as a result of the error control, the method can be applied reliably to almost any system without knowing the exact dynamics of that system beforehand. With fixed step size methods, on the other hand, the user will have to specify a step size, and if this step size is chosen too small, the integration will be inefficient, while too large a step size will lead to inaccurate results. The aforementioned advantages of adaptive step size methods also hold for the RKF56 method, which performed as one of the better methods. However, DOPRI8 clearly had an edge over RKF56 in terms of efficiency for the cases that were considered.

Another conclusion is that, in general, using the more clever approaches, in which only the perturbations are integrated numerically, will lead to large gains in efficiency compared to using the more straightforward formulations. Specifically, Gauss' equations and the Wisdom-Holman splitting method allow for a vastly more efficient integration process than the Cowell formulation and the T+V splitting method. A notable exception to this is the DOPRI8 method, which also performs reasonably well combined with the Cowell formulation.

Generally speaking, for the accurate, long-term integration of space debris orbits, it is definitely worthwhile, if not essential, to implement fairly complex propagation techniques instead of simpler methods. To illustrate this: a direct integration of the force model in Cartesian components (i.e. the Cowell formulation) with the RK4 method can give results of sufficient accuracy for a simulation of a GEO orbit of an object with a low area-to-mass ratio. However, it will take much longer to complete the integration than when more intricate methods are used. For the other test cases, in which the initial orbit was either eccentric or the debris object had a high area-to-mass ratio, acceptable results were not attained at all with the RK4 method combined with the Cowell formulation, not even when very small integration steps were used. So, although more complicated methods may take additional time to implement, for long-term simulations the associated gain in efficiency is so vast that the more simple methods should essentially not even be considered for use.

Regarding the symplectic methods, it can be stated that they are very efficient for long-term integrations of Hamiltonian systems. This was shown using integrations of the two-body problem, where the symplectic methods were quite dominant. Their results for the integration of perturbed systems were somewhat less impressive. That is not to say that their performance was bad. In fact, Kinoshita's method with Wisdom-Holman splitting was among the best methods for most test

cases. However, the vast majority of symplectic methods are fixed step size methods, with no form of error control. As a result, their performance is dependent on the user's choice of step size, which will rarely be optimal. Moreover, symplectic methods are relatively hard to implement if one is not familiar with their mathematical background and the nomenclature used in literature beforehand. Hence, while symplectic methods may be clearly the best choice for the integration of perfectly Hamiltonian systems, for the perturbed systems considered in this report, they are merely quite a good option. As such, their advantages are not considered substantial enough to recommend them over established methods such as DOPRI8.

Long-Term Evolution of Space Debris Orbits

To investigate the long-term behavior of space debris orbits, a number of 400-year simulations were carried out for debris objects with various area-to-mass ratios. The simulations were focused on debris originating from graveyard orbits, which were chosen conforming to the end-of-life guidelines defined by various space agencies.

Based on the simulations, it can be concluded that for satellites in the GEO region, the suggested end-of-life approach that involves re-orbiting satellites to graveyard orbits a few hundred kilometers above the geostationary ring is a safe strategy. Intact satellites in these graveyard orbits will not pose collision hazards to active geostationary satellites, not even after long periods of time.

Graveyard orbits in the GNSS region are another matter, though. Orbits in that region of space were found to be vulnerable to resonances, induced by the luni-solar perturbations. Whether or not an orbit will suffer from these effects is highly dependent on the initial conditions of the orbit. Of the GPS graveyard orbits that were tested, a near-circular graveyard orbit located almost exactly in between the GPS and Beidou constellations was determined to be a safe choice over long periods of time. However, due to the dependence on initial conditions, it is recommended to treat graveyard orbits in the GNSS region on a case-by-case basis, as the resonant dynamics may be difficult to predict beforehand.

For both the GEO and GNSS regions of space, it was found that debris objects with high area-to-mass ratios would be dangerous and would start crossing the orbital regimes used by active satellites within the time frame of a year. Essentially, the choice of graveyard orbit does not matter much for debris objects with sufficiently high area-to-mass ratios. Due to their extreme susceptibility to the effects of solar radiation pressure, they will find their way to the active satellite constellations regardless and will pose potential collision hazards. Therefore, it is recommended to focus efforts on preventing high area-to-mass ratio debris from being created at all, rather than trying to mitigate any collision probabilities related to it by means of unfeasibly propellant-intensive end-of-life maneuvers.

11.2 Recommendations

There are several options for expanding upon the findings of this thesis project. The most important ones are listed below.

- In this thesis project, a number of different computational methods were compared. For practical reasons, only a selection of the full spectrum of methods in existence could be included in the comparison. That being said, there are also other methods which would be interesting to consider in a future comparison. Some of these methods are multistep methods and extrapolation methods. In addition, variable step size symplectic methods would be an interesting choice. Although examples of these methods found in literature were all of relatively low order, the advantages in efficiency and reliability related to the use of adaptive step sizes are hard to overlook.
- The attitude of a debris object will likely have an impact on how its trajectory is affected by solar radiation pressure. In the simulation code, however, an average value for $C_R \cdot A/m$ has been used and it has been assumed that the surface normal always points to the Sun. More realistic behavior could potentially be attained if rotation would also be included in the simulations. Additionally, eclipses could be modeled slightly more accurately by means of a conical shadow model. In the current implementation, both of these approaches were not deemed necessary, but considering that solar radiation pressure can have such large effects on the trajectories of debris objects with high area-to-mass ratios, it would be interesting to see how much of an influence these adjustments will have on the predicted trajectories.
- The initial conditions of an orbit of a debris object are of importance for the future trajectory of that object. In this project, the assumption was made that debris objects originating from graveyard orbits would have the same initial orbits as the original re-orbited satellites. In reality, however, debris objects resulting from explosions or collisions can have significantly different velocity vectors. For more complete assessments of the safety provided by graveyard orbits, these differences in initial conditions could be taken into account in the simulations.
- The dangers related to space debris are all connected to collision possibilities. Predicting the actual probabilities of collisions is a topic on its own, and it was decided at the start of the project not to include this particular topic. Nevertheless, it would be useful for the assessment of graveyard orbits to be able to quantify how large the risks posed to active satellites would be. Consequently, adding a routine capable of providing estimates of the probabilities of collisions occurring over specified periods of time would definitely be helpful for evaluating different end-of-life options.

Appendix A

JGM-3 Gravitational Coefficients

The coefficients of the Joint Gravity Model 3 (JGM-3), up to degree and order 8, are listed in Table A.1. The coefficients in the table are the conventional (non-normalized) coefficients.

Table A.1: JGM-3 gravitational coefficients (Tapley et al., 1996).

n	m	C_{nm}	S_{nm}
2	0	-0.10826360229840e-02	0
3	0	0.25324353457544e-05	0
4	0	0.16193312050719e-05	0
5	0	0.22771610163688e-06	0
6	0	-0.53964849049834e-06	0
7	0	0.35136844210318e-06	0
8	0	0.20251871520885e-06	0
2	1	-0.24140000522221e-09	0.15430999737844e-08
3	1	0.21927988018965e-05	0.26801189379726e-06
4	1	-0.50872530365024e-06	-0.44945993508117e-06
5	1	-0.53716510187662e-07	-0.80663463828530e-07
6	1	-0.59877976856303e-07	0.21164664354382e-07
7	1	0.20514872797672e-06	0.69369893525908e-07
8	1	0.16034587141379e-07	0.40199781599510e-07
2	2	0.15745360427672e-05	-0.90386807301869e-06
3	2	0.30901604455583e-06	-0.21140239785975e-06
4	2	0.78412230752366e-07	0.14815545694714e-06
5	2	0.10559053538674e-06	-0.52326723987632e-07
6	2	0.60120988437373e-08	-0.46503948132217e-07
7	2	0.32844904836492e-07	0.92823143885084e-08

Table A.1: JGM-3 gravitational coefficients – *continued*.

n	m	C_{nm}	S_{nm}
8	2	0.65765423316743e-08	0.53813164055056e-08
3	3	0.10055885741455e-06	0.19720132389889e-06
4	3	0.59215743214072e-07	-0.12011291831397e-07
5	3	-0.14926153867389e-07	-0.71008771406986e-08
6	3	0.11822664115915e-08	0.18431336880625e-09
7	3	0.35285405191512e-08	-0.30611502382788e-08
8	3	-0.19463581555399e-09	-0.87235195047605e-09
4	4	-0.39823957404129e-08	0.65256058113396e-08
5	4	-0.22979123502681e-08	0.38730050770804e-09
6	4	-0.32641389117891e-09	-0.17844913348882e-08
7	4	-0.58511949148624e-09	-0.26361822157867e-09
8	4	-0.31893580211856e-09	0.91177355887255e-10
5	5	0.43047675045029e-09	-0.16482039468636e-08
6	5	-0.21557711513900e-09	-0.43291816989540e-09
7	5	0.58184856030873e-12	0.63972526639235e-11
8	5	-0.46151734306628e-11	0.16125208346784e-10
6	6	0.22136925556741e-11	-0.55277122205966e-10
7	6	-0.24907176820596e-10	0.10534878629266e-10
8	6	-0.18393642697634e-11	0.86277431674150e-11
7	7	0.25590780149873e-13	0.44759834144751e-12
8	7	0.34297618184624e-12	0.38147656686685e-12
8	8	-0.15803322891725e-12	0.15353381397148e-12

Appendix B

Routines for the Positions of the Sun and the Moon

As was mentioned in Section 4.4, approximate methods are used in the simulation code for computing the position coordinates of the Sun and the Moon at any point in time. The methods used for this are described below.

Position of the Sun

An approximate way of describing the coordinates of the Sun at any given time is outlined in (Montenbruck and Gill, 2005). This approach uses the J2000 epoch as a starting point. First, time is expressed as

$$T = (JD - 2451545.0)/36525.0 \quad (\text{B.1})$$

where T represents the number of Julian centuries since 1.5 January 2000 (J2000) and JD is the Julian Date at the time considered.

Then, the following values are used for the mean orbital elements that approximate the Sun's orbit with respect to the Earth and the ecliptic:

$$a = 149,600,000 \text{ km} \quad (\text{B.2})$$

$$e = 0.016709 \quad (\text{B.3})$$

$$i = 0.0000^\circ \quad (\text{B.4})$$

$$\Omega + \omega = 282.9400^\circ \quad (\text{B.5})$$

$$M = 357.5256^\circ + 35999.049T^\circ \quad (\text{B.6})$$

The position coordinates of the Sun can be determined using the relations for Keplerian orbits. However, because the inclination and eccentricity are both small, series expansions can be used without any significant loss of accuracy (Montenbruck and Gill, 2005), speeding up the computation process. This leads to the following expressions for the Sun's distance r_{Sun} and ecliptic longitude λ_{Sun} :

$$r_{Sun} = (149.619 - 2.499 \cos M - 0.021 \cos 2M) \cdot 10^6 \text{ km} \quad (\text{B.7})$$

$$\lambda_{Sun} = \Omega + \omega + M + 6892'' \sin M + 72'' \sin 2M \quad (\text{B.8})$$

The third coordinate of the spherical coordinate system, the ecliptic latitude β_{Sun} , vanishes within an accuracy of 1' (Montenbruck, 1989). It is therefore not listed above.

The spherical coordinates can be converted into Cartesian coordinates by applying the transformation

$$\mathbf{r}_{Sun} = \mathbf{R}_x(-\varepsilon) \begin{bmatrix} r_{Sun} \cos \lambda_{Sun} \cos \beta_{Sun} \\ r_{Sun} \sin \lambda_{Sun} \cos \beta_{Sun} \\ r_{Sun} \sin \beta_{Sun} \end{bmatrix} \quad (\text{B.9})$$

in which $\mathbf{R}_x(\phi)$ represents the rotation matrix for rotation about the x-axis

$$\mathbf{R}_x(\phi) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(\phi) & \sin(\phi) \\ 0 & -\sin(\phi) & \cos(\phi) \end{bmatrix} \quad (\text{B.10})$$

and

$$\varepsilon = 23.43929111^\circ \quad (\text{B.11})$$

is the obliquity of the ecliptic¹, being the inclination of the ecliptic relative to the equatorial plane of the Earth. In the case of Equation B.9, $\mathbf{R}_x(-\varepsilon)$ is used to rotate from the ecliptic plane to the equatorial plane. The vector involving the angles λ_{Sun} and β_{Sun} , on the other hand, is applied to transform from spherical to Cartesian coordinates.

Since $\beta_{Sun} = 0$ (cf. Equation B.4), Equation B.9 may be simplified to give the following vector for the position of the Sun:

$$\mathbf{r}_{Sun} = \begin{bmatrix} r_{Sun} \cos \lambda_{Sun} \\ r_{Sun} \sin \lambda_{Sun} \cos \varepsilon \\ r_{Sun} \sin \lambda_{Sun} \sin \varepsilon \end{bmatrix} \quad (\text{B.12})$$

In the simulation code, this is the relation that is used to transform from spherical coordinates with respect to the ecliptic plane to Cartesian coordinates with the equatorial plane as reference plane.

Position of the Moon

Just like with the position of the Sun, a series expansion can be used to approximate the position of the Moon at any given time. Since the motion of the Moon is strongly influenced by both the Earth and the Sun, a larger number of terms is needed in this case. In the approach described by Montenbruck and Gill (2005), the following five arguments are required:

$$L_0 = 218.31617^\circ + 481267.88088T^\circ - 1.3972T^\circ \quad (\text{B.13})$$

¹The obliquity of the ecliptic is assumed to be constant.

$$l = 134.96292^\circ + 477198.86753T^\circ \quad (\text{B.14})$$

$$l' = 357.52543^\circ + 35999.04944T^\circ \quad (\text{B.15})$$

$$F = 93.27283^\circ + 483202.01873T^\circ \quad (\text{B.16})$$

$$D = 297.85027^\circ + 445267.11135T^\circ \quad (\text{B.17})$$

In these relations, L_0 is the mean longitude of the Moon, l is the Moon's mean anomaly, l' is the Sun's mean anomaly, F is the mean angular distance of the Moon from the ascending node and D is the difference between the mean longitudes of the Sun and the Moon. The longitude of the Moon with respect to the equinox and the year 2000 ecliptic can now be expressed as:

$$\begin{aligned} \lambda_M = & L_0 + 22640'' \sin(l) + 769'' \sin(2l) \\ & - 4586'' \sin(l - 2D) + 2370'' \sin(2D) \\ & - 668'' \sin(l') - 412'' \sin(2F) \\ & - 212'' \sin(2l - 2D) - 206'' \sin(l + l' - 2D) \\ & + 192'' \sin(l + 2D) - 165'' \sin(l' - 2D) \\ & + 148'' \sin(l - l') - 125'' \sin(D) \\ & - 110'' \sin(l + l') - 55'' \sin(2F - 2D) \end{aligned} \quad (\text{B.18})$$

Furthermore, the latitude of the Moon can be calculated using

$$\begin{aligned} \beta_M = & 18520'' \sin(F + \lambda_M - L_0 + 412'' \sin(2F) + 541'' \sin(l')) \\ & - 526'' \sin(F - 2D) + 44'' \sin(l + F - 2D) \\ & - 31'' \sin(-l + F - 2D) - 25'' \sin(-2l + F) \\ & - 23'' \sin(l' + F - 2D) + 21'' \sin(-l + F) \\ & + 11'' \sin(-l' + F - 2D) \end{aligned} \quad (\text{B.19})$$

Finally, the distance of the Moon from the center of the Earth can be expressed as:

$$\begin{aligned} r_M = & (385000 - 20905 \cos(l) - 3699 \cos(2D - l) \\ & - 2956 \cos(2D) - 570 \cos(2l) + 246 \cos(2l - 2D) \\ & - 205 \cos(l' - 2D) - 171 \cos(l + 2D) \\ & - 152 \cos(l + l' - 2D)) \text{ km} \end{aligned} \quad (\text{B.20})$$

The expressions stated above yield the position of the Moon in spherical ecliptic coordinates. These coordinates can be converted to equatorial Cartesian coordinates with the following transformation:

$$\mathbf{r}_M = \mathbf{R}_x(-\varepsilon) \begin{bmatrix} r_M \cos \lambda_M \cos \beta_M \\ r_M \sin \lambda_M \cos \beta_M \\ r_M \sin \beta_M \end{bmatrix} \quad (\text{B.21})$$

Similar to Equation B.9, $\mathbf{R}_x(-\varepsilon)$ is used in Equation B.21 to rotate from the ecliptic plane to the equatorial plane, while the vector involving λ_M and β_M is utilized to transform from spherical to Cartesian coordinates.

Appendix C

Important Equations in Astrodynamics

In this appendix, a number of basic equations in astrodynamics related to the two-body problem are summarized (Wakker, 2010).

Newton's law of gravitation (two-body problem):

$$\ddot{\mathbf{r}} = -\frac{GM}{r^3}\mathbf{r} \quad (\text{C.1})$$

Vis-viva equation:

$$v^2 = GM \left(\frac{2}{r} - \frac{1}{a} \right) \quad (\text{C.2})$$

Kinetic energy:

$$T = \frac{1}{2}v^2 \quad (\text{C.3})$$

Potential energy:

$$V = -\frac{GM}{r} \quad (\text{C.4})$$

Total energy:

$$\begin{aligned} \mathcal{E} &= T + V \\ &= \frac{1}{2}v^2 - \frac{GM}{r} \\ &= -\frac{1}{2}\frac{GM}{a} \end{aligned} \quad (\text{C.5})$$

Gravitational parameter:

$$\mu = GM \quad (\text{C.6})$$

Conic section equation:

$$r = \frac{p}{1 + e \cos \theta} \quad (\text{C.7})$$

Semi latus-rectum:

$$p = a(1 - e^2) \quad (\text{C.8})$$

Ellipse ($0 \leq e < 1$):

$$r_p = a(1 - e) \quad (\text{C.9})$$

$$r_a = a(1 + e) \quad (\text{C.10})$$

Orbital period:

$$T = 2\pi \sqrt{\frac{a^3}{\mu}} \quad (\text{C.11})$$

Mean motion:

$$n = \sqrt{\frac{\mu}{a^3}} \quad (\text{C.12})$$

Kepler's equation:

$$E(t) - e \sin E(t) = n(t - \tau) \quad (\text{C.13})$$

Mean anomaly:

$$M = n(t - \tau) \quad (\text{C.14})$$

Bibliography

- Anselmo, L. and Pardini, C. (2005). Orbital evolution of geosynchronous objects with high area-to-mass ratios, *Proceedings of the 4th European Conference on Space Debris*, ESA/ESOC, Darmstadt, Germany. 18-20 April, 2005.
- Anselmo, L. and Pardini, C. (2008). Long-term evolution of geosynchronous orbital debris with high area-to-mass ratios, *Trans. Japan Soc. Aero. Space Sci.* (51): 22–27.
- Anselmo, L. and Pardini, C. (2009). Dynamical evolution of high area-to-mass ratio debris released into GPS orbits, *Advances in Space Research* (43): 1491–1508.
- Anselmo, L. and Pardini, C. (2010). Long-term dynamical evolution of high area-to-mass ratio debris released into high Earth orbits, *Acta Astronautica* (67): 204–216.
- Anselmo, L. and Pardini, C. (2011). Orbital evolution of the first upper stages used for the new European and Chinese navigation satellite systems, *Acta Astronautica* (68): 2066–2079.
- Berry, M. (2004). *A variable-step double-integration multi-step integrator*, Ph.D. Dissertation, Virginia Polytechnic Institute and State University, Blacksburg, Virginia, USA.
- Bourbaki, N. (1972). *Elements of Mathematics: Lie Groups and Lie Algebras*, Springer Verlag, Heidelberg, Germany.
- Chao, C. and Gick, R. (2004). Long-term evolution of navigation satellite orbits: GPS/GLONASS/GALILEO, *Advances in Space Research* (34): 1221–1226.
- Cunningham, L. (1970). On the Computation of the Spherical Harmonic Terms needed during the Numerical Integration of the Orbital Motion of an Artificial Satellite, *Celestial Mechanics* (2): 207–216.

- Deleflie, F., Rossi, A., Portmann, C., Métris, G. and Barlier, F. (2011). Semi-analytical investigations of the long-term evolution of the eccentricity of Galileo and GPS-like orbits, *Advances in Space Research* (47): 811–821.
- Eidelman, Y. (2000). *Introduction to Lie Operators for Accelerator Physics*, Argonne National Laboratory, Argonne, IL, USA.
- ESA (2009). Space Debris Environment, ESA Space Debris Office. http://www.esa.int/esaMI/Space_Debris/SEMQQ8VPXPF_0.html. Accessed on 21-03-2012.
- ESA (2013). Space Debris - Analysis and Prediction, ESA Space Debris Office. http://www.esa.int/Our_Activities/Operations/Space_Debris/Analysis_and_prediction. Accessed on 11-11-2013.
- Fecko, M. (2006). *Differential Geometry and Lie Groups for Physicists*, Cambridge University Press, Cambridge, UK.
- Fehlberg, E. (1968). Classical fifth-, sixth-, seventh-, and eight-order Runge-Kutta formulas with stepsize control, George C. Marshall Space Flight Center, Huntsville, Alabama, USA.
- Hairer, E., Lubich, C. and Wanner, G. (2006). *Geometric Numerical Integration*, Springer Verlag, Heidelberg, Germany.
- Hairer, E., Nørsett, S. and Wanner, G. (1987). *Solving Ordinary Differential Equations I*, Springer-Verlag, Berlin-Heidelberg-New York.
- Hand, L. and Finch, J. (2008). *Analytical Mechanics*, Cambridge University Press, Cambridge, UK.
- Harder, D. W. (2012). 4th-order Runge-Kutta and the Dormand-Prince Methods, University of Waterloo, Waterloo, Ontario, Canada.
- Hofsteenge, R. (2012). *Long-Term Orbit Development of Space Debris in the GEO and GPS Regions*, Literature Survey, Delft University of Technology, Delft, The Netherlands.
- Hubaux, C., Lemaître, A., Delsate, N. and Carletti, T. (2012). Symplectic integration of space debris motion considering several Earth's shadowing models, *Advances in Space Research* (49): 1472–1486.
- IADC (2002). IADC Space Debris Mitigation Guidelines. Issue 1, rev. 1.
- IADC (2011). Inter-Agency Space Debris Coordination Committee, IADC Website. <http://www.iadc-online.org/index.cgi>. Accessed on 24-04-2012.
- Ikeda, H., Hanada, T. and Yasaka, T. (2008). Searching for lost fragments in GEO, *Acta Astronautica* (63): 1312–1317.

- Ineichen, D., Beutler, G. and Hugentobler, U. (2003). Sensitivity of GPS and GLONASS orbits with respect to resonant geopotential parameters, *Journal of Geodesy* (77): 478–486.
- Jehn, R., Ariafar, S., Schildknecht, T., Musci, R. and Oswald, M. (2006). Estimating the number of debris in the geostationary ring, *Acta Astronautica* (59): 84–90.
- Jin, S. (2013). Recent progresses on Beidou/COMPASS and other global navigation satellite systems (GNSS), *Advances in Space Research* (51): 941.
- Kessler, D. and Cour-Palais, B. (1978). Collision Frequency of Artificial Satellites: The Creation of a Debris Belt, *Journal of Geophysical Research* (83): 2637–2646.
- Kinoshita, H. and Nakai, H. (1992). New Methods for Long-Time Numerical Integration of Planetary Orbits, *Proceedings of the 152nd Symposium of the International Astronomical Union*, Angra dos Reis, Brazil. 15-19 July, 1991.
- Klinkrad, H. (2006). *Space Debris - Models and Risk Analysis*, Praxis Publishing Ltd, Chichester, UK.
- Laskar, J. and Robutel, P. (2001). High order symplectic integrators for perturbed Hamiltonian systems, *Celestial Mechanics and Dynamical Astronomy* (80): 39–62.
- Leloux, J. (2012). *Filtering techniques for orbital debris conjunction analysis*, MSc. Thesis, Delft University of Technology, Delft, The Netherlands.
- Lewis, H., Swinerd, G., Martin, C. and Campbell, W. (2004). The stability of disposal orbits at super-synchronous altitudes, *Acta Astronautica* (55): 299–310.
- Liou, J. and Weaver, J. (2005). Orbital dynamics of high area-to-mass ratio debris and their distribution in the geosynchronous region, *Proceedings of the 4th European Conference on Space Debris*, ESA/ESOC, Darmstadt, Germany. 18-20 April, 2005.
- Matney, M. (2004). Toward a comprehensive GEO debris measurement strategy, *Acta Astronautica* (55): 291–298.
- McCarthy, D. (1996). *IERS Technical Note 21*, IERS Conventions, Central Bureau of IERS, Observatoire de Paris, Paris, France.
- Milani, A. and Nobili, A. (1988). Integration error over very long time spans, *Celestial Mechanics* (43): 1–34.
- Montenbruck, O. (1989). *Practical Ephemeris Calculations*, Springer Verlag, Heidelberg, Germany.

- Montenbruck, O. and Gill, E. (2005). *Satellite Orbits - Models, Methods and Applications*, Springer Verlag, Heidelberg, Germany. Corrected 3rd Printing.
- NASA (2010). Orbital Debris and the ISS, NASA Orbital Debris Program Office. http://www.nasa.gov/mission_pages/station/news/orbital_debris.html. Accessed on 20-04-2012.
- NASA (2011). NASA Orbital Debris FAQ, NASA Orbital Debris Program Office. http://www.nasa.gov/news/debris_faq.html. Accessed on 20-04-2012.
- Portree, D. and Loftus, J. (1993). Orbital Debris and Near-Earth Environmental Management: A Chronology, Technical Report 1320, NASA Reference Publication.
- Prince, P. and Dormand, J. (1981). High order embedded Runge-Kutta formulae, *Journal of Computational and Applied Mathematics* (7).
- Rex, D. (1998). Will space run out of space? The orbital debris problem and its mitigation, *Space Policy* (14): 95–105.
- Rossi, A. (2008). Resonant Dynamics of Medium Earth Orbits: Space Debris Issues, *Celestial Mechanics and Dynamical Astronomy* (100): 267–286.
- Roth, E. (1985). The Gaussian Form of the Variation-of-Parameter Equations Formulated in Equinoctial Elements - Applications: Airdrag and Radiation Pressure, *Acta Astronautica* (12): 719–730.
- Schildknecht, T., Musci, R. and Flohrer, T. (2008). Properties of the high area-to-mass ratio space debris population at high altitudes, *Advances in Space Research* (41): 1039–1045.
- Skeel, R. and Biesiadecki, J. (1994). Symplectic Integration with Variable Stepsize, *Annals of Numerical Mathematics* (1): 1–9.
- Tapley, B. D., Watkins, M. M., Ries, J. C., Davis, G. W., Eanes, R. J., Poole, S. R., Rim, H. J., Schutz, B. E., Shum, C. K., Nerem, R. S., Lerch, F. J., Marshall, J. A., Klosko, S. M., Pavlis, N. K. and Williamson, R. G. (1996). The Joint Gravity Model 3, *Journal of Geophysical Research* (101): 28029–28049.
- USSTRATCOM (2011). USSTRATCOM Space Control and Space Surveillance, United States Strategic Command. http://www.stratcom.mil/factsheets/USSTRATCOM_Space_Control_and_Space_Surveillance/. Accessed on 22-03-2012.
- van Kints, M. (2005). *Optimization of super-geostationary disposal orbits*, MSc. Thesis, Delft University of Technology, Delft, The Netherlands.
- Wakker, K. (2010). *Astroynamics I and II - Lecture Notes of Course AE4-874*, Delft University of Technology, Delft, The Netherlands.

-
- Walker, M., Ireland, B. and Owens, J. (1985). A Set of Modified Equinoctial Orbit Elements, *Celestial Mechanics* (36): 409–419.
- Wang, X., Wu, B., Hu, X., Zhu, Y., Qu, W., Yang, F. and Appleby, G. (2009). Impact of SLR Tracking on COMPASS/Beidou, *International Technical Laser Workshop on SLR Tracking of GNSS Constellations*, Metsovo, Greece. 14-19 Sept. 2009.
- Wertz, J. (2009). *Orbit & Constellation Design & Management*, Microcosm Press, Hawthorne, CA, USA.
- Wisdom, J. and Holman, M. (1991). Symplectic maps for the N -body problem, *The Astronomical Journal* (102(4)): 1528–1538.