Orbit Uncertainty Propagation in Space Situational Awareness applied to Conjunction Assessment

MSc Thesis

Justin Bos

Delft University of Technology

TU Delft

Orbit Uncertainty Propagation in Space Situational Awareness applied to Conjunction Assessment

MSc Thesis

by

Justin Bos

to obtain the degree of Master of Science
at the Delft University of Technology,
to be defended publicly on Thursday July 3, 2025 at 14:00.

Student number: 5079268
Project duration: September 2, 2024 – July 3, 2025
Thesis committee:

Dr. ir. E.J.O. Schrama	Chair
Dr. S. Gehly	Supervisor
M.Ş. Uludağ	External examiner

Cover: Modified version of an image generated with Dall·E 3.

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

TU Delft

Preface

This MSc thesis marks the end of my study period at TU Delft, through which I have learnt invaluable knowledge, gained incredible experiences, and made new lifelong friends, something that I am very grateful for.

I would like to extend my deepest gratitude to my thesis supervisor, Dr. Steve Gehly, for his crucial guidance and support throughout the thesis, as well as for all his efforts and patience prior to the thesis. I am proud to say that you have been the best supervisor I could have wished for.

I would also like to extend my heartfelt thanks to my parents, for their unwavering support and trust, and providing me with all the opportunities I could think of, it means the world to me. Additionally, to my brothers, and extended family, thank you for all your support, interest and advise, it is truly appreciated. Finally, to all my friends, thank you for all the academic support, the laughter, the reliefs of stress and overall the creation of good memories.

Justin Johannes Petrus Bos
Delft, September 2024

Executive Summary

Space Situational Awareness is becoming increasingly important as the number of space objects increases, and the risk of collisions between objects rises. A collision between two satellites will lead to even more fragmented objects cluttering the space around the Earth, which in turn causes more risk of collisions, and inevitably leads to the doom scenario of the Kessler syndrome, where our access to Space is prevented by the large density of space debris. To prevent this doom scenario, Conjunction Assessment aims to predict the probability of collision between objects ahead of time, to allow for mitigating measures. Conjunction assessment between space objects requires an accurate propagation of the uncertainty of both objects from a measurement to the time of closest approach. To predict collisions sufficiently ahead of time requires long propagations of the uncertainties through highly nonlinear dynamics, which often result in non-Gaussian final uncertainties. This research aims to improve upon the calculation of the collision probability by comparing existing and novel uncertainty propagation methods, and applying a promising novel uncertainty propagation method to conjunction assessments.

In the first part of this research, 8 different uncertainty propagation methods are implemented and compared to each other, by propagating the uncertainty in various challenging test cases. These test cases include two low earth orbits, two highly elliptical orbits and a geostationary orbit, and are used with varying propagation times, such as to analyse the use of these uncertainty propagation methods with highly non-Gaussian uncertainties. The implemented methods are Monte Carlo (MC) sampling, the linearised covariance (LinCov) method, the unscented transform (UT), the multi-fidelity (MF) method, Gaussian Mixture Models (GMMs) combined with the UT as well as with the MF to propagate the components, Adaptive Entropy-based Gaussian-mixture Information Synthesis (AEGIS) and Polynomial Chaos Expansions (PCEs).

It is found that LinCov is computationally fastest of the implemented methods, but only performs well when the final uncertainty remains Gaussian. Similarly the UT performs well for final Gaussian uncertainties. When the final uncertainty becomes non-Gaussian, i.e. for long propagation times or highly elliptical orbits, both PCEs and MF perform well, where MF has a higher computational efficiency. With the settings used in this research, GMMs are unable to accurately approximate the tails of non-Gaussian final uncertainties. Finally, AEGIS is too computationally inefficient compared to other methods, and becomes impractical for long propagation times.

In the second part of this research, the promising MF method has been used to propagate the uncertainties of two space objects to the time of closest approach (TCA) in 4 different conjunction assessment scenarios. The realistic conjunction scenarios are based on actual close encounters and include both high and low relative velocity encounters, as well as short and long propagation times. The uncertainties are propagated to the TCA from an initial epoch $48h$ or $96h$ earlier, using various different settings of MF, as well as using Monte Carlo samples to be used as a baseline. With the propagated uncertainties, the collision probability is calculated by comparing the closest distance for random generated sample pairs at the TCA, to the hard-body radius of the combined satellites.

It is found that when when enough important samples (8 or more) are propagated in the MF method, the collision probability remains within the 95% confidence interval, even when the low-fidelity dynamics used are as simple as 2-body Keplerian dynamics. Using the 2-body Keplerian dynamics, a computational speed-up of up to 4 orders of magnitude can be achieved.

Contents

List of Tables

List of Figures

List of symbols and abbreviations

Abbreviations

AEGIS	Adaptive Entropy-based Gaussian-mixture Information Synthesis
CA	Conjunction Analysis
CARA	Conjunction Risk Analysis team
CI	Confidence interval
DA	Differential Algebra
DCA	Distance at Closest Approach.
DISCOS	Database and Information System Characterising Objects in Space
ECI	Earth Centred Inertial
ESA	European Space Agency
GEO	Geostationary Earth Orbit
GMM	Gaussian Mixture Model
GMM_MF	Gaussian Mixture Models combined with Multi-Fidelity
GMM_UT	Gaussian Mixture Models combined with the Unscented Transform
HBR	Hard-Body Radius
HEO	Highly Elliptical Orbit
HF	High-Fidelity
LEO	Low Earth Orbit
LF	Low-Fidelity
LinCov	Linearised Covariance method
MC	Monte Carlo
MEE	Modified Equinoctial Elements
MF	Multi-Fidelity
NASA	National Aeronautics and Space Administration
PCE	Polynomial Chaos Expansion
PDF	Probability density function
RSW	Radial, Along-track and Cross-track
SRP	Solar Radiation Pressure
SSA	Space Situtational Awareness
STM	State Transition Matrix
STT	State Transition Tensors
TCA	Time of Closest Approach
Tudat	TU Delft Astrodynamics Toolbox
UP	Uncertainty Propagation
UT	Unscented Transform
UTC	Universal Time Coordinated

Greek Symbols

α_i	Weight of the i'th component in a Gaussian mixture model	$[-]$
β	Confidence level	$[-]$
$\boldsymbol{\chi}_i$	State vector of the i'th sigma point	$[m, ms^{-1}]$
$\boldsymbol{\eta}$	Vector pointing towards the ascending node and the true anomaly	$[-]$
$\boldsymbol{\Phi}(t, t_0)$	State transition matrix from t_0 to t	$[-]$
$\boldsymbol{\xi}$	Random input vector	$[m, ms^{-1}]$
ϵ_a	Absolute tolerance level of variable step integrators	$[-]$
ϵ_r	Relative tolerance level of variable step integrators	$[-]$
ϵ_θ	Tolerance level of the true anomaly	$[rad]$
ϵ_{DCA}	Tolerance level of the distance at closest approach	$[m]$
ϵ_{ent}	Tolerance level of the change in entropy	$[-]$

ϵ_{lf}	Tolerance level of the estimation of the low fidelity samples	$[m]$
ϵ_{num}	Lower numerical limit, set to a small value like 10^{-10}	$[-]$
ϵ_{TCA}	Tolerance level of the time of closest approach	$[s]$
κ	Tuning parameter for the unscented transform	$[-]$
μ	Gravitational constant of the Earth	$\left[m^3 s^{-2}\right]$
Ω	Right ascension of the ascending node of the orbit	$[rad]$
ω	Argument of periapsis of the orbit	$[rad]$
$\Psi(\boldsymbol{\xi})$	Basis function in a polynomial chaos expansion	$[-]$
$\psi_i^j(\boldsymbol{\xi})$	The j'th polynomial function for the i'th variable	$[-]$
σ	Standard deviation of a univariate probability distribution	$[-]$
$\sigma_{\hat{a}}$	Standard deviation of a multivariate probability distribution along the unit vector $\hat{a}$	$[-]$
θ	True anomaly	$[rad]$
$\underline{\Xi}$	Set of important samples	$[-]$
$\underline{\xi}$	Important sample	$[-]$
Ξ	Set of sample points	$[-]$
ξ_i	Random input of the i'th variable	$[-]$

Latin Symbols

$\hat{\boldsymbol{a}}$	Unit vector of the splitting direction	$[-]$
$\overline{\boldsymbol{x}}$	Mean state vector	$\left[m, ms^{-1}\right]$
$\boldsymbol{A}$	Permutation matrix due to pivoting	$[-]$
$\boldsymbol{C_h}$	Lower triangular matrix in the Cholesky decomposition	$[-]$
$\boldsymbol{c}_m$	Matrix containing coefficients in the multi-fidelity method	$[-]$
$\boldsymbol{c}_p$	Matrix containing coefficients in the Polynomial Chaos Expansions method	$[-]$
$\boldsymbol{d}_{rel}$	Vector of Euclidean distances between pairs of state vectors	$[m]$
$\boldsymbol{e}$	Eccentricity vector	$[-]$
$\boldsymbol{G^L}$	Gramian matrix of the low fidelity propagated states	$[-]$
$\boldsymbol{G}$	Matrix characterising the diffusion of a Brownian motion process	$\left[m^2, m^2 s^{-2}\right]$
$\boldsymbol{g}^L$	Matrix defined by dot products between low fidelity propagated states of specific samples	$[-]$
$\boldsymbol{h}$	Specific angular momentum vector	$\left[m^2 s^{-1}\right]$
$\boldsymbol{J}$	Jacobian matrix	$[-]$
$\boldsymbol{J}^*(t)$	Jacobian matrix of the reference trajectory	$[-]$
$\boldsymbol{P}$	Covariance matrix	$\left[m^2, m^2 s^{-1}, m^2 s^{-2}\right]$
$\boldsymbol{P}^r$	Position covariance matrix	$\left[m^2\right]$
$\boldsymbol{Q}$	Covariance matrix of a Brownian motion process	$\left[m^2, m^2 s^{-2}\right]$
$\boldsymbol{r}$	Position vector in Cartesian coordinates	$[m]$
$\boldsymbol{R}_A^B$	Rotation matrix from frame A to frame B	$[-]$
$\boldsymbol{r}_{rel}$	Relative position of the primary object with respect to the secondary object	$[m]$
$\boldsymbol{S}$	Square root matrix of the covariance matrix	$\left[m, ms^{-1}\right]$
$\boldsymbol{V}$	Copy of the low fidelity propagated states	$\left[m, ms^{-1}\right]$
$\boldsymbol{v}$	Velocity vector in Cartesian coordinates	$\left[ms^{-1}\right]$
$\boldsymbol{w}_c$	Vector used as intermediate step in the pivoted Cholesky decomposition	$[-]$
$\boldsymbol{x}$	State vector (in Cartesian coordinates unless specified otherwise)	$\left[m, ms^{-1}\right]$
$\boldsymbol{x}^*$	State vector of a reference orbit	$\left[m, ms^{-1}\right]$
$\boldsymbol{X}^0(\Xi)$	Initial states belonging to the generated samples	$\left[m, ms^{-1}\right]$
$\boldsymbol{X}^H(\Xi)$	States propagated using high fidelity dynamics	$\left[m, ms^{-1}\right]$
$\boldsymbol{X}^k(\Delta \boldsymbol{t})$	States propagated using 2-body Keplerian dynamics	$\left[m, ms^{-1}\right]$
$\boldsymbol{X}^L(\Xi)$	States propagated using low fidelity dynamics	$\left[m, ms^{-1}\right]$
$\boldsymbol{X}^Q$	Set of state vectors of points in a quadrature grid	$\left[m, ms^{-1}\right]$
$\boldsymbol{x}_i$	State vector of the i'th sample	$\left[m, ms^{-1}\right]$

Δr	Relative Euclidean distance between two state vectors	$[m]$
Δt_{prop}	Length of time between the propagation start time and mean time of closest approach	$[s]$
Δv	Relative velocity of the primary object with respect to the secondary object	$[ms^{-1}]$
$\Delta \boldsymbol{t}$	Vector with time epochs relative to t_0 in the propagation	$[s]$
$\Delta \boldsymbol{x}$	Difference in state vectors between a true trajectory and reference orbit	$[m, ms^{-1}]$
$\dot{\boldsymbol{x}}$	First time derivative of a state vector	$[ms^{-1}, ms^{-2}]$
$\dot{H}_e(\boldsymbol{x})$	Time derivative of the differential entropy of a probability distribution	$[-]$
$\hat{\boldsymbol{R}}$	Unit vector of the radial direction of the RSW coordinate frame	$[m]$
$\hat{\boldsymbol{S}}$	Unit vector of the along-track direction of the RSW coordinate frame	$[m]$
$\hat{\boldsymbol{W}}$	Unit vector of the cross-track direction of the RSW coordinate frame	$[m]$
$\hat{\boldsymbol{X}}$	Unit vector of the x-axis of the conjunction plane in Cartesian coordinates	$[m]$
$\hat{\boldsymbol{Y}}$	Unit vector of the y-axis of the conjunction plane in Cartesian coordinates	$[m]$
$\hat{\boldsymbol{Z}}$	Unit vector of the z-axis of the conjunction plane in Cartesian coordinates	$[m]$
$\hat{K}$	Unit vector of the third axis in a 3-dimensional coordinate system	$[-]$
$[\boldsymbol{c}_m]_\ell$	Coefficients relating all samples to the ℓ'th important sample	$[-]$
$[\boldsymbol{c}_p]_j$	The coefficient of the j'th polynomial in the expansion	$[-]$
$[\tilde{\boldsymbol{c}}_p]_j$	Optional coefficient of the j'th polynomial in the expansion	$[-]$
$\mathscr{I}$	Set of indices	$[-]$
$\mathscr{N}(\boldsymbol{x} : \overline{\boldsymbol{x}}, \boldsymbol{P})$	Multivariate Gaussian distribution with mean vector $\overline{\boldsymbol{x}}$ and covariance $\boldsymbol{P}$	$[-]$
$\mathscr{N}(0, \mathrm{I}_6)$	Multivariate standard normal distribution	$[-]$
$\overline{x}$	Mean of variable x in a univariate distribution	$[-]$
$\tilde{\boldsymbol{X}^H}$	Estimated states with high fidelity dynamics	$[m, ms^{-1}]$
$\tilde{\boldsymbol{X}^L}$	Estimated states with low fidelity dynamics	$[m, ms^{-1}]$
$\tilde{h}$	Step size used for the interpolation of the nonlinearity	$[-]$
$\tilde{V}$	Integration volume	$[m^3]$
$\underline{\mathscr{I}}$	Ordered list of indices corresponding to important samples	$[-]$
a	Semi major axis of the orbit	$[m]$
A_P	Average cross sectional area of the primary object.	$[m^2]$
A_S	Average cross sectional area of the secondary object.	$[m^2]$
A_j	Area of the joint radii of the two objects in a conjunction.	$[m^2]$
C_D	Drag coefficient of a satellite.	$[-]$
C_R	Solar Radiation coefficient of a satellite.	$[-]$
DCA_{mean}	Distance between sampled pair of state vectors at $t_{TCA,mean}$	$[m]$
DCA_{true}	True distance at closest approach between the sampled pair of state vectors	$[m]$
E	Eccentric anomaly	$[rad]$
e	Eccentricity of the orbit	$[-]$
e_c	Error level used to evaluate iterations in the pivoted Cholesky decomposition	$[-]$
e_m	Mathematical constant, approximately equal to 2.71828	$[-]$
e_{pc}	Relative error of the collision probability	$[-]$
f	x-component of the eccentricity vector in the orbital frame	$[-]$
$F(\boldsymbol{x}, t)$	Derivative function of a state vector dependent on time	$[-]$
g	y-component of the eccentricity vector in the orbital frame	$[-]$
H	Angular momentum	$[kg \cdot m^2 s^{-1}]$
h	x-component of the ascending node vector in the orbital frame	$[-]$
$h(x, y)$	Combined PDF of two objects in the conjunction plane	$[-]$
H_e	Differential entropy of a distribution	$[-]$
$H_{initial}$	Nonlinear differential entropy of a probability distribution evaluated at the initial time	$[-]$
H_{linear}	Linear differential entropy of a probability distribution	$[-]$
$H_{nonlinear}$	Nonlinear differential entropy of a probability distribution	$[-]$
I	Retrograde factor	$[-]$

i	Inclination of the orbit	$[rad]$
k	y-component of the ascending node vector in the orbital frame	$[-]$
L	True longitude	$[rad]$
L_2	L_2 distance between two probability distributions	$[-]$
M	Mean anomaly	$[rad]$
m	Mass of the satellite	$[kg]$
N/P	Ratio of number of samples per coefficient in a polynomial chaos expansion	$[-]$
N_c	Number of samples resulting in a collision	$[-]$
N_g	Number of components in a Gaussian mixture model	$[-]$
N_t	Total number of samples generated in a conjunction assessment	$[-]$
N_{bisect}	Number of points to divide each interval by, in the bisection algorithm	$[-]$
N_{grid}	Number of points in a quadrature integration grid	$[-]$
N_{orbit}	Number of orbital revolutions between the start and end time of the propagation	$[-]$
$N_{samples}$	Number of samples generated	$[-]$
n_{state}	Size of the state vector	$[-]$
NL	Nonlinearity measure of a function	$[-]$
NL_2	Normalised L_2 distance between two probability distributions	$[-]$
p	Semi-latus rectum	$[m]$
$p(\boldsymbol{x})$	Probability density function of a multivariate distribution	$[-]$
$p(x)$	Probability density function of a univariate variable	$[-]$
P_C	Collision probability	$[-]$
P_e	Expansion order of a polynomial chaos expansion	$[-]$
$P_{C,i}$	Instantaneous collision probability	$[-]$
P_{coef}	Number of coefficients in a polynomial chaos expansion	$[-]$
Q	Quadrature order of the numerical integration grid	$[-]$
R	Radius of an object	$[m]$
r	Distance	$[m]$
r_{mf}	Rank of the multi-fidelity solution defining the number of important samples	$[-]$
T	Orbital period	$[s]$
t	Time epoch	$[-]$
t_0	Initial time epoch of the propagation	$[-]$
t_f	Final time epoch of the propagation	$[-]$
t_{window}	Length of time window around $t_{TCA,mean}$ in which to find $t_{TCA,true}$	$[s]$
v	Velocity	$[ms^{-1}]$
V'	Volume swept out by a hardball sphere during a close encounter	$[m^3]$
V_{max}	Eigenvector corresponding to the maximum eigenvalue	$[m, ms^{-1}]$
v_{ratio}	Ratio between the desired relative velocity and the old relative velocity	$[-]$
W_i	Weight of the i'th sigma point	$[-]$
w_i	Weight of the i'th sample	$[-]$
w_q	Weight of the q'th point in a quadrature grid	$[-]$

Subscripts

$_1$	From the first probability distribution in a comparison of uncertainties
$_2$	From the second probability distribution in a comparison of uncertainties
$_x$	x-coordinate of vector in Cartesian coordinates
$_y$	y-coordinate of vector in Cartesian coordinates
$_z$	z-coordinate of vector in Cartesian coordinates
$_{1,i}$	From the i'th component of the first Gaussian mixture model
$_{2,i}$	From the i'th component of the second Gaussian mixture model
$_{cart}$	In Cartesian coordinates

J	From the joint probability distribution in a conjunction assessment
mee	In modified equinoctial elements
new	After a transformation
old	Before a transformation
P	From the primary object in a conjunction assessment
sc	From the scaled Gaussian probability distribution
S	From the secondary object in a conjunction assessment
t_0	At the initial time epoch
t_f	At the final time epoch
$TCA,mean$	At the time of closest approach between the mean state vectors
$TCA,true$	At the actual time of closest approach between the sampled pair of state vectors
TCA	At the time of closest approach

1

Introduction

Space Situational Awareness.

With the ever growing number of space objects being launched, and the exponential rise of objects in orbit around Earth [1], space situational awareness becomes increasingly important to prevent doom scenarios such as that described by the Kessler syndrome [1]. Collisions between spacecrafts have sparked additional attention to the concepts of collision avoidance, conjunction analysis and space debris tracking [2, 3]. Among many other space objectives, these methods require an adequate characterization of the uncertainty in the orbital state of the catalogued space objects [4], which entails determining the object's state moments (such as the mean and covariance matrix) or associated probability density function (PDF) [5]. Due to the increase in the ratio of objects in space to telescopes on the ground, and the already vast amount of tracked space objects, the tracking of catalogued objects uses long propagation times of multiple orbits and possibly spanning several days, to provide orbital state information in between sparse measurements [6, 7]. These long propagation times and strong accuracy requirements define the problem in uncertainty propagation (UP) in astrodynamics.

Uncertainty propagation methods.

The problem of propagating the PDF of a space object in the two-body problem can be described by the Fokker-Planck equation [8]. However, solutions to this equation are generally intractable and no solution is known for this equation in the astrodynamics domain [5, 9]. Conventional approaches to solve the equation numerically scale exponentially with the size of the state vector, and thus suffer from the 'curse of dimensionality'.

To handle the large and complex state-space of orbit uncertainty problems, Sequential Monte Carlo (MC) methods were introduced [10, 11]. The Monte Carlo method has seen many improvements [12–16], and has been widely used in the late 20th century for space missions. However, to achieve a high accuracy with MC, a large number of samples is needed, which can be computationally expensive for orbit propagation problems [5]. Nevertheless, due to its easy implementation and the reliability of its results, which approach the true probability distribution when the number of samples goes to infinity, it is still often used as a comparison to determine the accuracy performance of other methods[17].

An alternative to the computationally intensive MC is to linearise the orbital mechanics, and assume that the PDF is a Gaussian distribution, such that propagating this PDF is reduced to integrating the state transition matrix (STM)[5, 18–20]. While considerably faster than MC, this method becomes inaccurate when used over long periods for the highly nonlinear orbital systems. In a linear system, a Gaussian distribution remains Gaussian over time. However, in the non-linear orbital dynamics this is not the case, and long propagations lead to non-Gaussian uncertainties, even if the initial distribution is Gaussian [21]. The linear uncertainty propagation and STM can not account for this.

This trade-off between accuracy and computational intensity has led to alternative methods being actively investigated. One of these methods is the unscented transform (UT) proposed by [22], which keeps the nonlinear dynamics of the system, while approximating the PDF using a Gaussian distribution represented by a limited set of weighted samples [23–25]. This method only requires the propagation of $2 \cdot n_{state} + 1$ points for an accurate approximation of the first two moments. However, this approach is limited to Gaussian distributions.

To approximate non-Gaussian PDFs, a Gaussian mixture model (GMM) can be used [26–30]. This method approximates the non-Gaussian PDF by a sum of multiple weighted Gaussian distribution components, which can be determined using various optimisation and splitting methods [27, 28, 30–32]. Extensions of the GMM include methods to adaptively split components online, such as to keep an accurate representation of the PDF as the non-Gaussianity increases [26, 29, 33]. The GMMs provide an accurate way to spread the propagation of large non-Gaussian PDFs in non-linear systems over multiple small problems, which can be easily combined with many of the other existing uncertainty propagation methods. However, for large dimensionalities, GMMs

may require many components to obtain the required accuracy, and determining and updating the weights of the Gaussian components can be a complex process [5].

The state transition tensor (STT) method for orbit uncertainty propagation has been proposed in [34], which has been extended on in [35–39]. This method applies Taylor expansions to the deviation of the state with respect to a nominal trajectory, where a first order Taylor expansion corresponds to solving for the STM in the aforementioned linearised method, and solving for higher order solutions can provide better results for cases with strong nonlinearity [34]. These STT based methods do not require random sampling and also deliver accurate results in comparison to the MC method, while providing a semi-analytic method to non-linearly propagate the uncertainties efficiently. However, these methods require continuous and differentiable underlying dynamics, and the derivation of higher order STTs often results in computational complexities [5].

To avoid the derivation of complex partial derivatives, differential algebra (DA), proposed in [40], can be used to propagate the orbit uncertainty nonlinearly [41, 42]. The DA technique allows for the computation of a function's Taylor expansion up to an arbitrary order, which can then be used to efficiently compute the PDF [42]. DA methods are able to propagate uncertainties nonlinearly without having to integrate the variational equations to obtain high-order expansions. However, like STTs, DA also requires continuous and differentiable dynamics, which in orbit determination is not always the case [5].

Another alternative faster than the MC method, which is able to represent higher moments of the PDF, is using polynomial chaos expansions (PCEs), first proposed by [43]. This method represents the inputs and outputs of a system using series of approximations with random variables [5]. The coefficients can be determined using intrusive and non-intrusive methods. The latter treats the dynamics as a black box, making the method suitable for astrodynamics applications with pre-existing orbit propagation tools [5], and PCEs have been applied to many orbit UP problems [7, 44–49]. PCEs allow for fast propagation and conversion, but also suffer from the curse of dimensionality [50].

This work focuses on a new non-intrusive method based on stochastic collocation, Multi-fidelity (MF), proposed by [51] and first applied to an astrodynamics problem in [52]. MF leverages the speed performance of low fidelity propagation methods to propagate many samples, while achieving accurate results by correcting the propagated states with few carefully chosen high-fidelity (HF) samples [52]. The MF method can be combined with other methods listed above (GMMs, DA, MC) and can provide reduced runtimes for both particle and Gaussian mixture representations of the PDF, however, at a cost of reduced accuracy [53].

Multi-fidelity orbit uncertainty propagation.

Multi-fidelity methods are not unique to orbit uncertainty propagation problems, or even uncertainty quantification in general, but have been used in various fields and come in varying versions of the method [54]. The common factor of these techniques is the advantageous combination of high-fidelity models to achieve a high accuracy and low-fidelity models to produce fast results. Narayan et al. [51] proposed a non-intrusive version of the MF method applicable to stochastic collocation problems, built upon in [55], which has then been used by Jones et al. [52] for orbit uncertainty propagation problems, starting a series of developments in the MF method for astrodynamics.

Jones et al. [52] applies the MF method to a set of test cases in which an initial Gaussian PDF is propagated to a non-Gaussian PDF. The test cases include two different low earth orbit (LEO) satellites and a Molniya orbit, where different acceleration models are analysed for the low-fidelity model. The MF method demonstrates a significant decrease in position error when the sampled states propagated with low-fidelity (LF) dynamics are corrected using the subset of points propagated with high-fidelity (HF) dynamics. Here the position error is determined by directly comparing the propagated / corrected points with the same initial point propagated using HF dynamics. At the same time, the states propagated with MF dynamics can be generated with much faster run times compared to propagating all samples using HF dynamics. The MF method is also combined with GMMs for the same scenarios, where the MF method provides similar results but with faster run times compared to using GMMs with the UT for each component, by speeding up the propagation of the sigma points. Finally, MF is applied to a two-line element case, where the MF correction improves the orbit determination accuracy of the low-fidelity model using SGP4 [56].

Jones et al. [53] develops a method to determine the systematic errors induced by the MF method, which is tested using the scenarios described in [52]. A new type of filter is developed using MF and GMMs, in which the systematic error is accounted for, thus producing smaller errors than what is normally achievable with the MF method. The filter is demonstrated on a double pendulum problem, and applied to a space object tracking scenario. This work shows that the MF filter is faster than using a HF filter, and converges to the HF

filter's results. In the updated work [57], a method is introduced to automatically split the domain over which the MF methods are used to achieve an arbitrary accuracy without significantly increasing the computation time.

Fossà et al. [58] develops a new method combining MF with DA and GMMs. The GMM representation of the PDF is used with automatic splitting, where in the low-fidelity step, the non-linearity index is determined by Taylor expansions computed with DA. The centres of the Taylor polynomials are then propagated as sample points using high-fidelity models. The new method is demonstrated with three test cases containing a low, medium and high Earth orbit. The MF method is compared to MC samples and a good agreement is found between the PDFs, with significant speed-up compared to the HF method.

Jia et al. [59] applies the MF in a different way to assess and leverage the correlation between the uncertainties after different numbers of orbital revolutions, and uses it for efficient long-term uncertainty propagation. While this uses the same methods as the other MF studies, to combine a higher accuracy model with a more efficient model, this study uses the description of low-fidelity and high-fidelity differently. The method comprises a short propagation and a longer propagation representing the low- and high-fidelity model respectively, which are used as a MF method to predict the long term uncertainty propagation. Results for a test scenario, using a highly elliptical orbit in the equatorial plane, with long-term propagation times of both 5 and 10 orbital periods, demonstrate that with very few sampling data the MF method can achieve comparable accuracy to MC, and is especially effective for long-time uncertainty propagations.

Wolf et al. [60] applies the MF method developed in [52] to the cis-lunar space, and determines the effectiveness of the method in these highly nonlinear environments, and compares the effect of different low-fidelity models in the three-body problem. Test case scenarios include a low and medium Lunar orbit, and a near rectilinear Halo orbit. Results lead to similar conclusions as in [52], with improvements on accuracy compared to the low-fidelity model and better runtime performance than the high-fidelity models.

Fossà et al. [61] builds upon [58] and decouples the propagation of the initial uncertainty from the process noise effects, to combine them a posteriori into a single orbit state estimation accounting for both uncertainty sources. The method is applied to two test cases taken from [52] and an extra scenario in which a continuous thrust is applied to a spacecraft in a geostationary transfer orbit. Results indicate similar accuracy to the high-fidelity methods with improved runtime performances.

Conjunction Assessment

Conjunction Assessment (CA) is one of the most important applications of uncertainty propagation. CA consists of a number of tasks, aiming to asses the risk of collision between two space objects. Important tasks of CA are propagating the positions and uncertainties of space objects to the time of closest approach (TCA), and computing the collision probability P_C [62]. For the computation of P_C, many different methods have been developed with varying assumptions, levels of accuracy and computational efficiencies [63].

The MC simulation has been widely used to calculate the instantaneous collision probability [64–67]. This method is considered the most reliable and accurate, but also the most computationally expensive method [63]. Many improvements have been developed to MC to improve its computational efficiency, both to the MC method itself[68–70] and to the underlying sampling methods used for the propagation [7, 45, 71–74].

Due to the long runtimes required for MC methods, alternative methods to calculate the instantaneous P_C have been developed, both analytical and numerical: Transforming the integral form into simpler expressions [75], combining PCEs with MC [7, 45, 71], using the conjugate unscented transform [76] and splitting the position PDF into Gaussian components that can be integrated over a hard sphere. [77]

More interesting than the instantaneous collision probability, is the cumulative collision probability P_C, which is the instantaneous collision probability integrated over the time of the close encounter, and is often distinguished between short-term and long-term encounters. Many different methods exist to calculate P_C for short-term encounters, such as numerical methods [78–80] as well as various analytical methods [75, 81–84]. These short term methods make a number of assumptions, including constant velocities and constant uncertainties over the time of the encounter. Using these assumptions, the time integral of the instantaneous collision probability can be transformed to a volume integral, which can be reduced to an area integral in the encounter plane [63]. Foster's method [78] uses polar coordinates in the encounter plane [85], Alfano [79] develops a series representation, and Patera [80] uses a coordinate rotation and a scale change to reduce the problem to a 1-dimensional line integral.

In cases of low relative velocities, the assumptions made for the short-term methods can no longer be used,

and long-term methods are needed to determine P_C. Patera [86] develops an integration method to estimate P_C that includes changes in the position, velocity and covariances throughout the conjunction. Coppola et al. [87] derives a formula based on a spherical surface integral, which can be used for both short- and long-term encounters. Alfano [88] presents three more numerical approaches using a discretisation of time, and in [89] the nonlinear motion is linearised to enable easier evaluation of the integrations. These methods all assume Gaussian uncertainties, but other developments can also be used with non-Gaussian uncertainties. Vittaldev et al. [90] uses a GMM to represent a non-Gaussian distribution and uses all-on-all analysis of the components to compute P_C, DeMars et al. [91] expands on this by using Coppola's method to include the velocity uncertainty, Vittaldev et al. [30] uses a multivariate GMM to approach non-Gaussian distributions better and Shelton et al. [92] approximates non-Gaussian PDFs with an adaptive GMM to combine with Coppola's methods.

In addition to the research of new collision probability computation methods, different orbit uncertainty propagation methods have been combined with P_C computation methods to provide faster or more accurate alternatives. A combination of GMMs and STT is developed in [93] to apply as an uncertainty propagation method for conjunction analysis. Khatri et al. [94] extends this by relaxing the assumption of one object with a known position and using more realistic dynamics. This is expanded on in [95–97] by demonstrating that in a long-term conjunction each component's interaction with other components can be modelled as a short-term encounter, thus extending the applications to long-term conjunction analysis. Balducci et al. [98] applies separated representations with MC to calculate P_C. Jones et al. [45] uses PCEs to generate the samples required for a multi-epoch MC computation of P_C for a short term conjunction, and avoids the computation of the time of closest approach (TCA) by assuming a dense grid over the conjunction period. This work is also extended to the conjunction analysis after a translational manoeuvrer in [7].

Research Motivation

Previous studies [52, 53, 57–61] have shown the effectiveness of the Multi-Fidelity method in efficiently propagating non-Gaussian uncertainties over long time windows and in challenging scenarios, while still providing good agreement with the results obtained from the MC runs . This raises the question whether the Multi-Fidelity can be used to propagate the uncertainties to the TCA in a conjunction analysis to provide a significant improvement in computation time, without deviating too much from the real probability distributions such that the collision probability becomes inaccurate.

Current methods to compute the collision probability each have their advantages and disadvantages [63]. Many of the methods assume Gaussian uncertainties even at the TCA, which is often not the case. Other methods have reduced accuracy or require too many assumptions on the scenarios. Using Multi-Fidelity (which can accurately propagate non-Gaussian uncertainties too), or a combination of the Multi-Fidelity method and Gaussian Mixture Models, a better alternative can be found for the computation of the collision probability. A more accurate can result in the decrease of unnecessary collision avoidance manoeuvrers, whereas a more efficient method facilitates an increase in the number of space objects that can be tracked for collisions for the same computational power available. A higher computational efficiency also means that longer propagation times can be used within the same computation times. Thus, if the method used can accurately predict the uncertainty for long propagation times, the time window of conjunction assessments can be extended to predict collisions further in advance, which gives satellite operators more time to take mitigating action.

The importance of conjunction analysis in space situational awareness, the certain need for an uncertainty propagation method to compute the collision probability, and the absence of previous research into the application of Multi-Fidelity to conjunction analysis, all together lead to a key research gap, to be filled with by the work described in this research proposal. In this research, the relatively new method of Multi-Fidelity orbit uncertainty propagation is applied to the realm of conjunction analysis, to see if the method can be a feasible alternative for the slow method of MC simulations. Due to the various number of methods to propagate the uncertainty of space objects, there is plenty of literature available on the application of other uncertainty propagation methods in conjunction analysis, which can be beneficial for the set up of the test scenarios, useful for guidance on the methodology, and can provide applicable validation and comparison data.

The research performed in this report is outlined in Chapter 2, where the objective, main research question and sub-questions are documented. Following this, Chapter 3 provides more information about UP methods and Chapter 4 goes into more detail on the computation of the collision probability. A comparison study between several UP methods for various test case scenarios is described in Chapter 5, after which the results are presented in Chapter 6. Subsequently, a conjunction analysis study with the MF method is set up in Chapter 7, followed by the results in Chapter 8. Finally, the report is concluded by answering the research question in Sections 9.1 and 9.2, a discussion in Section 9.3, and ended with several recommendations in Section 9.4.

2

Research Outline

To aid the development in the field of uncertainty propagation and conjunction analysis and to improve the problem described in Chapter 1, a research will be conducted with the following objective:

Reduce the safety risks of space debris by improving the efficiency of computationally expensive tasks in conjunction analysis.

In order to achieve this, a preliminary research is outlined, with the focus on the recently developed methods of orbit uncertainty propagation, such as Multi-Fidelity. The research is described by the following research question:

How can novel orbit uncertainty propagation methods contribute to the improvement upon the calculation of the collision probability between two space objects?

By investigating the following set of sub-questions related to this main research question, a better understanding of orbit uncertainty propagation environment will be developed, and a promising candidate for the uncertainty propagation method is analysed, with the aim to answer the main research question:

- **How do different orbit uncertainty propagation methods compare to each other in terms of accuracy and computational efficiency?**
- **Is Multi-Fidelity a feasible orbit uncertainty propagation method to use for conjunction analysis?**
- **How do challenging scenarios such as long time horizons, highly elliptical orbits and low relative velocities, affect the performance of High-Fidelity orbit uncertainty propagation in conjunction assessment?**

3

Uncertainty Propagation Methods

3.1. Monte Carlo

An uncertainty's PDF can be very precisely propagated using the Monte Carlo method. By generating a sufficient amount of random samples from the initial distribution, the propagated states of these random samples will accurately represent the final distribution, even when the final distribution is not a Gaussian distribution.

Starting from an initially Gaussian uncertainty, $N_{samples}$ number of samples can be generated from this multivariate normal distribution via [52]

$$\boldsymbol{x}_{t_0}(\boldsymbol{\xi}) = \overline{\boldsymbol{x}}_{t_0} + \boldsymbol{S}\boldsymbol{\xi} \tag{3.1}$$

where $\overline{\boldsymbol{x}}_{t_0}$ is the mean state vector at t_0, $\boldsymbol{S}$ is the square root matrix of the covariance matrix, such that $\boldsymbol{P} = \boldsymbol{S}\boldsymbol{S}^T$, and

$$\boldsymbol{\xi} \sim \mathcal{N}(0, \mathrm{I}_6) \tag{3.2}$$

are random input vectors, generated using a pseudo-random number simulator.

These samples can then be propagated individually to their final states, where they can be used as a sample set to show the final uncertainty. From these samples, a Gaussian mean and covariance can also be derived according to:

$$\overline{\boldsymbol{x}} = \sum_{i=1}^{N_{samples}} w_i \boldsymbol{x}_i \tag{3.3}$$

$$\boldsymbol{P} = \sum_{i=1}^{N_{samples}} w_i \left(\boldsymbol{x}_i - \overline{\boldsymbol{x}}\right)\left(\boldsymbol{x}_i - \overline{\boldsymbol{x}}\right)^T \tag{3.4}$$

where for all Monte Carlo samples, $w_i = 1/N_{samples}$.

The accuracy of the Monte Carlo method depends on the number of samples used. The higher the number of samples, the more accurate the representation of the true uncertainty will be, and although a large sample set is computationally expensive, it is often used to represent the truth in a comparison with other uncertainty propagation methods.

3.2. Linearised Covariance

A quick and computationally efficient method to propagate an uncertainty is to use the linearised covariance (LinCov) method, which approximates the nonlinear dynamics with a linear model. The true trajectory of an object in space, with states $\boldsymbol{x}(t)$, can be represented by the difference with respect to a reference orbit, with states defined by $\boldsymbol{x}^*(t)$. If the reference orbit is sufficiently close to the true trajectory, the difference between states, $\Delta\boldsymbol{x}(t) = \boldsymbol{x}(t) - \boldsymbol{x}^*(t)$, will behave linearly [18]. An exaggerated illustration is shown in Figure 3.1.

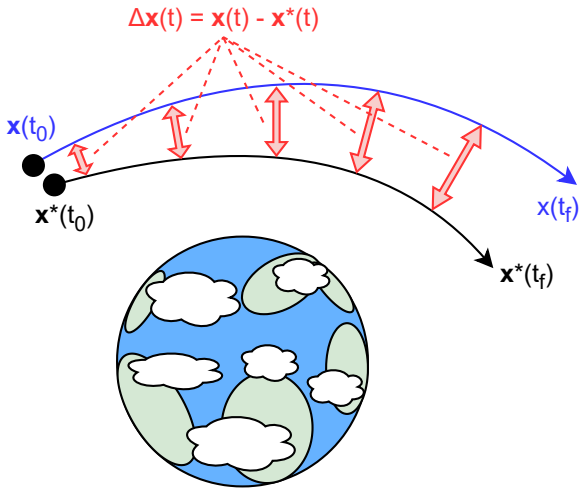


Figure 3.1: *Illustration of the relation between a true trajectory and a reference orbit.*

Given the nonlinear dynamics of the true trajectory [18]:

$$\dot{\boldsymbol{x}}(t) = F(\boldsymbol{x}, t) \tag{3.5}$$

the Taylor expansion can be expanded about the reference trajectory:

$$\dot{\boldsymbol{x}}(t) = F(\boldsymbol{x}, t) = F(\boldsymbol{x}^*, t) + \left[\frac{\delta F(t)}{\delta \boldsymbol{x}(t)}\right]^* \left[\boldsymbol{x}(t) - \boldsymbol{x}^*(t)\right] + \ldots \tag{3.6}$$

By neglecting higher order terms, this can be simplified to:

$$\Delta\dot{\boldsymbol{x}}(t) = \boldsymbol{J}^*(t)\Delta\boldsymbol{x}(t) \tag{3.7}$$

where $\boldsymbol{J}^*(t)$ represents the Jacobian of the reference trajectory at epoch t:

$$\boldsymbol{J}^*(t) = \left[\frac{\delta F(t)}{\delta \boldsymbol{x}(t)}\right]^* \tag{3.8}$$

The general solution of this system can then be expressed as:

$$\Delta\boldsymbol{x}(t) = \boldsymbol{\Phi}(t, t_k)\Delta\boldsymbol{x}(t_k) \tag{3.9}$$

where $\boldsymbol{\Phi}(t, t_k)$ is the state transition matrix (STM) from epoch t_k to t.

The mean state vector and covariance matrix are defined as the expected value of the state, and the expected squared difference with respect to the mean state [99]:

$$\overline{\boldsymbol{x}} = \int_{\infty} \boldsymbol{\xi}\, p(\boldsymbol{\xi}) d\boldsymbol{\xi} \tag{3.10}$$

$$\boldsymbol{P} = \int_{\infty} (\boldsymbol{\xi} - \overline{\boldsymbol{x}})(\boldsymbol{\xi} - \overline{\boldsymbol{x}})^T p(\boldsymbol{\xi}, t) d\boldsymbol{\xi} \tag{3.11}$$

Assuming that the differences between the true trajectory and the reference trajectory remains small, the STM in Equation 3.9 can be used with Equations 3.10 and 3.11 to propagate the mean state vector and covariance matrix linearly from the initial epoch t_0 to epoch t:

$$\overline{\boldsymbol{x}}(t) = \boldsymbol{\Phi}(t, t_0)\overline{\boldsymbol{x}}(t_0) \tag{3.12}$$

$$\boldsymbol{P}(t) = \boldsymbol{\Phi}(t, t_0)\boldsymbol{P}(t_0)\boldsymbol{\Phi}(t, t_0)^T + \boldsymbol{G}(t)\boldsymbol{Q}(t)\boldsymbol{G}(t)^T \tag{3.13}$$

Here $\boldsymbol{G}(t)$ is a matrix characterising the diffusion of a Brownian motion process with covariance matrix $\boldsymbol{Q}(t)$. Both $\boldsymbol{G}(t)$ and $\boldsymbol{Q}(t)$ can be used to account for mismodelling of the acceleration models. However, mismodelling will not be considered for any of the UP methods in this study, since this research focusses more on the

evolution of the uncertainty in the nonlinear dynamical system, and on comparing various uncertainty propagation methods to each other based on their computational efficiencies and accuracies in estimating the final uncertainty. Thus, the new formula to propagate the covariance matrix forward becomes:

$$\boldsymbol{P}(t) = \boldsymbol{\Phi}(t, t_0)\boldsymbol{P}(t_0)\boldsymbol{\Phi}(t, t_0)^T \tag{3.14}$$

With this method, only the mean vector of the probability distribution has to be propagated to solve the variational equations and obtain the STM, which can be used in Equation 3.14 to calculate the final covariance matrix. Since the method only requires one state vector to be propagated, it is very computationally efficient. However, the method assumes a Gaussian distribution for both the initial and final epoch, and linearises the dynamics close to the reference trajectory. Since the true orbital dynamics are nonlinear and a Gaussian uncertainty becomes non-Gaussian for long propagation times, this leads to inaccuracies, which are small for short propagation times but will grow when the propagation time increases.

3.3. Unscented Transform

The idea of the Unscented Transform (UT) is to avoid linearising or approximating the dynamics, by instead approximating the probability distribution itself. This is done by using a set of carefully chosen sigma points from which the statistical properties of the distribution can be determined. These sigma points can then be transformed using a nonlinear function, after which the new distribution is determined from the transformed points. An illustration is shown in Figure 3.2.

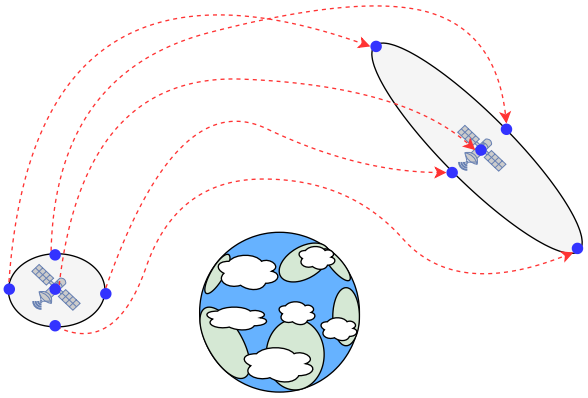


Figure 3.2: *A 2D illustration of the Unscented transform.*

There are various sets of points and weights available to use for the UT. In this study, the following set given by Julier et al. [100] will be used:

$$\begin{aligned}
\boldsymbol{\chi}_0 &= \overline{\boldsymbol{x}} & W_0 &= \kappa / (n_{state} + \kappa) \\
\boldsymbol{\chi}_i &= \overline{\boldsymbol{x}} + \left(\sqrt{(n_{state} + \kappa)\boldsymbol{P}}\right)_i & W_i &= 1/(2 \cdot (n_{state} + \kappa)) \quad \text{for } i = 0, \ldots, n_{state} \\
\boldsymbol{\chi}_{i+n_{state}} &= \overline{\boldsymbol{x}} - \left(\sqrt{(n_{state} + \kappa)\boldsymbol{P}}\right)_i & W_{i+n_{state}} &= 1/(2 \cdot (n_{state} + \kappa))
\end{aligned} \tag{3.15}$$

where $\overline{\boldsymbol{x}}$ is the mean vector of the distribution, $\left(\sqrt{(n_{state} + \kappa)\boldsymbol{P}}\right)_i$ is the i'th column of the square root matrix of the scaled covariance matrix, n_{state} is the size of the state vector and κ is a tuning parameter, often set to $n_{state} + \kappa = 3$ for Gaussian distributions.

To transform a set of sigma points $\boldsymbol{\chi}_i$ and weights W_i back into statistical moments, the following equations can be used [100]:

$$\overline{\boldsymbol{x}} = \sum_{i=0}^{2 \cdot n_{state}} W_i \boldsymbol{\chi}_i \tag{3.16}$$

$$\boldsymbol{P} = \sum_{i=0}^{2 \cdot n_{state}} W_i \left(\boldsymbol{\chi}_i - \overline{\boldsymbol{x}}\right)\left(\boldsymbol{\chi}_i - \overline{\boldsymbol{x}}\right)^T \tag{3.17}$$

3.4. Multi-Fidelity Orbit Uncertainty Propagation

The Multi-Fidelity (MF) method is based on using a low-fidelity (LF) model to propagate a large number of samples in an efficient manner, and combining this with a correction based on a small set of samples propagated with a high-fidelity (HF) model, to improve the accuracy of the propagated points. The method relies on the assumption that the relation between sample points propagated using LF dynamics, will not be much different from the relation between sample points propagated using HF dynamics. In this way, all the sample points propagated using LF dynamics are mapped to a small set of 'important samples'. The important samples are then propagated using the HF dynamics, and the map is used to transform the remaining LF samples to approximately the states they would assume under the HF dynamics. An illustration of this process is shown in Figure 3.3.

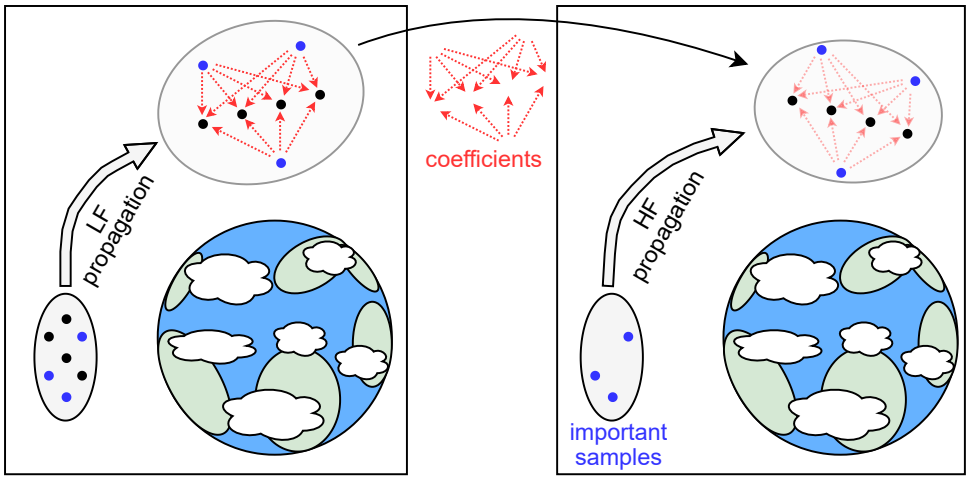


Figure 3.3: *Illustration of the Multi-Fidelity orbit uncertainty propagation method.*

Given a set of $N_{samples}$ sample points $\Xi = \left\{\xi_1, \ldots, \xi_{N_{samples}}\right\}$, the initial states $\boldsymbol{X^0}(\Xi) = \left[\boldsymbol{x}_{t_0}(\xi_1), \cdots, \boldsymbol{x}_{t_0}(\xi_{N_{samples}})\right]$ are created from an initial Gaussian distribution using:

$$\boldsymbol{x}(\xi) = \overline{\boldsymbol{x}} + \boldsymbol{S}\boldsymbol{\xi} \tag{3.18}$$

Where $\overline{\boldsymbol{x}}$ is the mean state vector, $\boldsymbol{S}$ is the square root matrix of the covariance matrix, such that $\boldsymbol{P} = \boldsymbol{S}\boldsymbol{S}^T$, and

$$\boldsymbol{\xi} \sim \mathcal{N}(0, \mathrm{I}_6) \tag{3.19}$$

are generated using a pseudo-random number simulator.

The next step is to propagate all samples from t_0 to t_f with LF dynamics to get $\boldsymbol{X^L}(\Xi) = \left[\boldsymbol{x}^L_{t_f}(\xi_1), \cdots, \boldsymbol{x}^L_{t_f}(\xi_{N_{samples}})\right]$. With these LF states, Algorithm 1, included in Appendix A, can be used to select a smaller set of r_{mf} important samples $\underline{\Xi} = \left\{\underline{\xi}_1, \cdots, \underline{\xi}_{r_{mf}}\right\}$, while also generating an array of coefficients $\boldsymbol{c}_m$, where for each important sample $\underline{\xi}_\ell$ the coefficients $[\boldsymbol{c}_m]_\ell$ relate all LF states $\boldsymbol{X^L}(\Xi)$ to the LF state vector of this important sample: $\boldsymbol{x}^L\left(\underline{\xi}_\ell\right)$. This process is explained in more detail further in this section. For now it suffices that the important samples and coefficients are generated such that the LF states $\boldsymbol{X^L}(\Xi)$ can be approximated by the important samples using an expansion of the form:

$$\tilde{\boldsymbol{x}}^L(\boldsymbol{\xi}) = \sum_{\ell=1}^{r_{mf}} [\boldsymbol{c}_m]_\ell(\boldsymbol{\xi}) \boldsymbol{x}^L\left(\underline{\xi}_\ell\right) \tag{3.20}$$

To determine the required rank r_{mf}, an initial value of $r = 1$ is chosen, which is increased by 1 at each step while repeating Algorithm 1 to determine the important samples, until the generated set of important samples satisfies an accuracy condition. The accuracy condition can be defined by comparing the approximation $\tilde{\boldsymbol{X}}^L(\Xi)$ from Equation 3.20 with the actual states $\boldsymbol{X^L}(\Xi)$. In this work r_{mf} is selected such that the maximum difference

between the approximated states and the actual states is smaller than a user-defined tolerance ϵ_{lf}:

$$\max\left|\tilde{\boldsymbol{X}}^L - \boldsymbol{X}^L\right| < \epsilon_{lf} \tag{3.21}$$

The small set of important samples $\underline{\Xi}$ is then propagated using HF dynamics to generate $\boldsymbol{X}^H\left(\underline{\Xi}\right)$. The assumption is made that the same coefficients $\boldsymbol{c}_m$ used with $\boldsymbol{X}^L\left(\underline{\Xi}\right)$ to approximate $\tilde{\boldsymbol{X}}^L(\Xi)$ in Equation 3.20 can be used in an expansion with $\boldsymbol{X}^H\left(\underline{\Xi}\right)$ to approximate $\tilde{\boldsymbol{X}}^H(\Xi)$ [52]. With the HF states of the important samples $\boldsymbol{X}^H\left(\underline{\Xi}\right)$, the state vector of each sample can be estimated with HF dynamics using:

$$\tilde{\boldsymbol{x}}^H(\boldsymbol{\xi}) = \sum_{\ell=1}^{r_{mf}} [\boldsymbol{c}_m]_\ell(\boldsymbol{\xi})\boldsymbol{x}^H\left(\underline{\boldsymbol{\xi}}_\ell\right) \tag{3.22}$$

To select the important samples and generate the coefficients, Equation 3.23 can be solved iteratively to find the next important sample, by maximising the distance between the state of the next important sample $\boldsymbol{x}\left(\underline{\xi}_k\right)$ and the states of the existing set of important samples $\boldsymbol{X}\left(\underline{\Xi}^{k-1}\right)$.

$$\underline{\xi}_k = \underset{\underline{\xi}\in\underline{\Xi}}{\arg\max}\ \mathrm{dist}\left(\boldsymbol{x}(\underline{\xi}), \boldsymbol{X}\left(\underline{\Xi}^{k-1}\right)\right) \tag{3.23}$$

Where

$$\mathrm{dist}(\boldsymbol{x},\mathbb{X}) \equiv \inf_{\boldsymbol{y}\in\mathbb{X}} \|\boldsymbol{x}-\boldsymbol{y}\| \qquad \text{and} \qquad \underline{\Xi}^k = \underline{\Xi}^{k-1} \cup \left\{\underline{\xi}_k\right\} \tag{3.24}$$

However, solving this is generally intractable [60], and a linear algebra method from [51] can be used to determine the set of important samples $\underline{\Xi}$ and the coefficients $\boldsymbol{c}_m$ used in Equation 3.20 by solving the pivoted Cholesky decomposition [71]:

$$\left[\boldsymbol{X}^L\right]^T \boldsymbol{G}^L \left[\boldsymbol{X}^L\right] = \boldsymbol{A}^T \boldsymbol{C_h} \left[\boldsymbol{C_h}\right]^T \boldsymbol{A} \tag{3.25}$$

Here $\boldsymbol{G}^L$ is the Gramian matrix of the LF samples $\boldsymbol{X^L}$, where $\left[\boldsymbol{G}^L\right]_{i,j} = \boldsymbol{x}^L(\xi_i)\cdot\boldsymbol{x}^L\left(\xi_j\right)$, and $\boldsymbol{A}$ is a permutation matrix based on pivoting that orders samples based on the optimization problem shown in Equation 3.23, and thus directly indicates the indices of the important samples $\underline{\Xi}$ in the full sample set Ξ.

Using Algorithm 1 from Appendix A, $\boldsymbol{A}$ and $\boldsymbol{C_h}$ can be obtained from Equation 3.25 without computing the full Cholesky decomposition [71]. The $\boldsymbol{C_h}$ matrix can then be used to generate $\boldsymbol{c}_m$ using [71]:

$$\boldsymbol{C_h}\left[\boldsymbol{C_h}\right]^{\boldsymbol{T}} \boldsymbol{c}_m = \boldsymbol{g}^L \tag{3.26}$$

where

$$\left[\boldsymbol{g}^L\right]_\ell = x^L(\xi)\cdot x^L\left(\underline{\xi}_\ell\right) \qquad \ell = 1,\ldots,r \tag{3.27}$$

The entire process is summarised in a flowchart presented in Figure 3.4

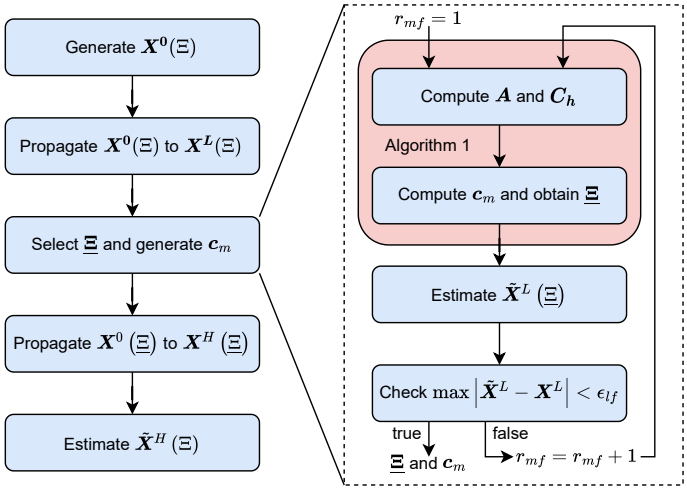


Figure 3.4: *Flowchart of the process of propagating an uncertainty with MF*

Equation 3.25 requires $n_{state} \geq r_{mf}$ in order to generate a large enough matrix $\boldsymbol{A}$ to find the order of indices for the r_{mf} important samples. This means that for this MF uncertainty propagation method, which uses Equation 3.25 in Algorithm 1 (documented in Appendix A), the maximum number of important samples is limited to the size of the state vector. For most astrodynamics problems, the state vector $\boldsymbol{x}(\xi)$ is comprised of the position and velocity vector in 3D:

$$\boldsymbol{x}(\xi) = \begin{bmatrix} \boldsymbol{r}_t(\boldsymbol{\xi}) \\ \boldsymbol{v}_t(\boldsymbol{\xi}) \end{bmatrix} \tag{3.28}$$

Using this as the state vector, results in $n_{state} = 6$ which limits the number of import samples too much. However, n_{state} can be increased by including the position and velocity vectors at multiple moments of t into one new state vector:

$$\boldsymbol{x}(\boldsymbol{\xi}) = \begin{bmatrix} \boldsymbol{r}_{t_0}(\boldsymbol{\xi}) \\ \boldsymbol{v}_{t_0}(\boldsymbol{\xi}) \\ \vdots \\ \boldsymbol{r}_{t_f}(\boldsymbol{\xi}) \\ \boldsymbol{v}_{t_f}(\boldsymbol{\xi}) \end{bmatrix} \tag{3.29}$$

This formulation allows $r_{mf} > 6$ and will be used in this further analysis. Since the propagation of states from t_0 to t_f already requires integrating the position and velocity vector over multiple intermediate steps, the generation of this extended state vector causes no significant increase to the computation time.

3.5. Gaussian Mixture Models

Gaussian Mixture Models are useful to represent a non-Gaussian PDF as a weighted sum of Gaussian distributions. The PDF is split up into smaller components that each have an associated weight and their own Gaussian PDF with a covariance and a mean. By combining enough of these components, any general PDF can be accurately approximated [101]. By splitting the non-Gaussian PDF into smaller Gaussian distributions, the individual components can conveniently be used for uncertainty propagation and collision probability computation methods that assume Gaussian distributions [30, 90–92, 94–97]. An example illustration of how a GMM representation can use Gaussian components to evolve to a non-Gaussian PDF is shown in Figure 3.5 [96].

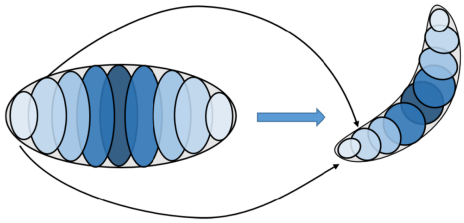


Figure 3.5: *A propagated GMM representation of an initially Gaussian PDF to a non-Gaussian PDF, image from [96].*

For a univariate Gaussian distribution with mean $\overline{x}$ and standard deviation σ, the PDF is given by [102]:

$$f(x) = \frac{1}{\sqrt{2\pi}\sigma}\exp\left[-\frac{(x-\overline{x})^2}{2\sigma^2}\right] \tag{3.30}$$

For a multivariate Gaussian distribution, with mean vector $\overline{\boldsymbol{x}}$ and covariance matrix $\boldsymbol{P}$, this is represented by [27]:

$$\mathcal{N}(\boldsymbol{x}:\overline{\boldsymbol{x}},\boldsymbol{P}) = \frac{1}{\sqrt{\det(2\pi\boldsymbol{P})}}\exp\left[-\frac{1}{2}(\boldsymbol{x}-\overline{\boldsymbol{x}})^T\boldsymbol{P}^{-1}(\boldsymbol{x}-\overline{\boldsymbol{x}})\right] \tag{3.31}$$

The PDF of a GMM with N_g components is then given by:

$$p(\boldsymbol{x}) = \sum_{i=1}^{N_g}\alpha_i\mathcal{N}(\boldsymbol{x}:\overline{\boldsymbol{x}}_i,\boldsymbol{P}_i) \tag{3.32}$$

where $\overline{\boldsymbol{x}}_i$ and $\boldsymbol{P}_i$ are the mean and covariance matrix of the component i for $i = 1,2,...,N_g$.

There are various versions of GMMs available. These versions depend on the split size, the split directions, the splitting rules used for the resulting components, and the algorithms used to determine when and how to split a component. For a select number of components and algorithms, various pre-computed splitting libraries have been developed. These splitting libraries provide the weights α_i, means $\overline{x}_i$ and standard deviations σ_i for a specific number of elements of a GMM that approaches the standard normal distribution. Using the standard deviation and mean of the non-Gaussian PDF, this GMM approximating the standard normal distribution can then be shifted and scaled to approximate the non-Gaussian PDF.

The splitting of a PDF into a GMM is done by generating a fixed number of Gaussian components N_g with their predefined covariance matrix $\boldsymbol{P}_i$, and creating an optimisation problem for the weights α_i and means $\overline{\boldsymbol{x}}_i$ of the components. In the optimisation problem, a metric that defines the accuracy of the PDF approximation, such as the L_2 distance, can be used to set up a cost function, and symmetry and the total summation of the weights can be used as constraints. To form the predefined covariance matrices $\boldsymbol{P}_i$, the original covariance matrix $\boldsymbol{P}$ of the non-Gaussian PDF is taken, and a new predefined standard deviation σ is applied to the column of the splitting direction in the square-root $\boldsymbol{S}$ of the covariance matrix $\boldsymbol{P}$ according to $\boldsymbol{P} = \boldsymbol{S}\boldsymbol{S}^T$. The standard deviation values in the splitting direction can be chosen according to different splitting rules [103]:

$$\begin{aligned} 1:&\quad \sigma^2 = (1/N_g) \\ 2:&\quad \sigma^2 = (1/N_g)^{3/4} \\ 3:&\quad \sigma^2 = (1/N_g)^{1/2} \end{aligned}$$

To keep the number of components limited to save computation time, the number of directions to split a GMM in is often limited to 1 or 2. This direction is carefully chosen, sometimes based on the size of the covariance in the different directions, determined by the eigenvalues of the covariance matrix [28], or in other cases the sensitivity to nonlinearity at t_0 [103], determined with a metric described by Equation 3.33.

$$NL(\hat{\boldsymbol{a}}) = \frac{f(\overline{\boldsymbol{x}}+\tilde{h}\sigma_{\hat{a}}\hat{\boldsymbol{a}}) + f(\overline{\boldsymbol{x}}-\tilde{h}\sigma_{\hat{a}}\hat{\boldsymbol{a}}) - 2f(\overline{\boldsymbol{x}})}{2\tilde{h}^2} \tag{3.33}$$

Here $\hat{\boldsymbol{a}}$ is the unit vector of the splitting direction, $\sigma_{\hat{a}}$ is the standard deviation along that unit vector, and $\tilde{h}$ is the step size of the interpolation, often chosen as $\sqrt{3}$ [103]. In cases of multiple splitting directions, the directions can be ranked based on these metrics and the top 2 or 3 directions are chosen to split in. Since the nonlinearity in orbital dynamics applies to all directions in the Cartesian frame, splitting the GMM to 6 components becomes intractable. Therefore, converting the coordinates to Keplerian elements (or better Equinoctial elements to avoid singularities), where 5 of the 6 orbital elements evolve linearly, reduces the needed splitting directions to 1 and provides better approximations of the non-Gaussian PDF [27].

After an uncertainty has been split into a GMM, the propagation of the GMM can be done by various other uncertainty propagation methods, such as MC, LinCov, UT and MF, since each component can be treated as a Gaussian distribution and can be propagated separately, or in parallel. The most logical here are the UT or LinCov, since the purpose of using GMMs is to transform non-Gaussian uncertainties into smaller uncertainties that remain Gaussian, thereby allowing the use of the most efficient methods that assume final Gaussian distributions.

Besides splitting an initial PDF into a GMM that keeps the number of components constant during the uncertainty propagation, the GMM can also be split online by checking at each step whether the nonlinearity passes a threshold, and applying a split to the current set of components in the direction that becomes too nonlinear. This is called automatic domain splitting, and has various different implementations possible. The methods differ in their ways of checking when to split a component and how to split the component.

A commonly used method for adaptive splitting is Adaptive Entropy-based Gaussian-mixture Information Synthesis (AEGIS) [28], where the differential entropy of a solution flow is calculated using the linear LinCov method and nonlinear UT, such that the difference can be used to determine whether or not to split the GMM further. Another method compares the linearised extended Kalmann filter (EKF) solution with higher order solutions to determine the size of the truncation error due to linearisation, and applies splitting accordingly [104].

3.6. Adaptive Entropy-based Gaussian-mixture Information Synthesis

The AEGIS method propagates an uncertainty as a GMM (for an initially Gaussian distribution this can be initialised as a GMM with 1 single component), and continues to split this GMM further along the propagation, at moments where the nonlinear behaviour reaches a user-defined threshold. The idea behind this, is that whenever the dynamics are so nonlinear that it affects the solution more than desired, the splitting will decrease the size of the Gaussian components, and thus reduce the effect of nonlinear behaviour on the components.

The nonlinear behaviour is assessed by comparing the difference between two calculations of the differential entropy of the Gaussian component, a linear and nonlinear version. The differential entropy $H_e(\boldsymbol{x})$ of a Gaussian distribution $\mathcal{N}\left(\boldsymbol{x}:\overline{\boldsymbol{x}},\boldsymbol{P}\right)$ can be determined using [28]:

$$H_e(\boldsymbol{x}) = \frac{1}{2}\log|2\pi e_m \boldsymbol{P}| \tag{3.34}$$

where $\boldsymbol{P}$ is the covariance matrix, $\boldsymbol{x}$ is the current state and e_m is the mathematical constant, equal to approximately 2.718. For a linearised system, the time derivative of this differential entropy can be determined by [105]:

$$\dot{H}_e(\boldsymbol{x}) = \text{trace}\left(\boldsymbol{J}\left(\overline{\boldsymbol{x}}(t),t\right)\right) \tag{3.35}$$

where $\boldsymbol{J}\left(\overline{\boldsymbol{x}}(t),t\right)$ is the Jacobian matrix of the time derivatives of the state as a function of the mean $\overline{\boldsymbol{x}}(t)$.

The linear differential entropy can be determined by starting with an initial differential entropy calculated by Equation 3.34, and integrating the derivative in Equation 3.35 over time. Whereas the nonlinear differential entropy can be directly calculated using Equation 3.34 after propagating the covariance using a nonlinear technique, often the UT [105]. When the normalised difference (with respect to the initial entropy of the component) reaches a threshold, the propagation is stopped. The component is then split into new components and the entropies are recalculated before continuing with the propagations of the new components. The difference in entropy is assessed according to:

$$\frac{|H_{nonlinear} - H_{linear}|}{H_{initial}} > \epsilon_{ent} \tag{3.36}$$

Where $H_{initial}$ is the nonlinear entropy at the initial epoch of the component's propagation, and ϵ_{ent} is a tunable parameter $0 < \epsilon_{ent} < 1$.

In this way, each component is propagated (usually with the UT, which can be directly used to calculate the nonlinear differential entropy along the propagation) until all components have reached the end of the time domain. This can be done in parallel, stopping the entire propagation each time a component requires splitting, and continuing with a slightly bigger set of components. It can also be done sequentially, by adding newly split components to the queue, and propagating each component individually until all components in the queue have reached the final epoch.

3.7. Polynomial Chaos Expansions

Non-intrusive Polynomial Chaos Expansions (PCEs) approximate the final solution by a sum of polynomial functions, built around a number of sampled points. The sampled points are selected based on a specific grid, and can be propagated using a black box propagator, to provide sample states of the final solution. These states are then used to generate a PCE, which can be used to estimate the entire probability distribution, as well as for other purposes such as sensitivity analyses.

The final uncertainty can be represented by [106]:

$$\boldsymbol{x}_{t_f}(\boldsymbol{\xi}) \approx \sum_{j=0}^{P_e} \left[\boldsymbol{c}_p\right]_j \Psi_j(\boldsymbol{\xi}) \tag{3.37}$$

where

$$\Psi_j(\boldsymbol{\xi}) = \prod_{i=1}^{n_{state}} \psi_i^j(\xi_i) \tag{3.38}$$

Here p denotes the polynomial order, d the dimension of the state vector, or in other words the number of variables in each polynomial, $\psi_i^j(\boldsymbol{\xi})$ is the j'th polynomial function for the i'th variable with respect to the random input ξ_i of the i'th variable. $\boldsymbol{c}_p$ are the coefficients of the expansion, to be found at the epoch of interest t. In total, a number of P_{coef} coefficients are needed for the expansion, where P_{coef} follows from [44]:

$$P_{coef} = \frac{(P_e + n_{state})!}{P_e! \cdot n_{state}!} \tag{3.39}$$

where P_e is the order of the expansion, and n_{state} is the dimension of the state vector $\boldsymbol{x}$.

To solve Equation 3.37 for the coefficients $\boldsymbol{c}_p$, two different methods are used: Pseudo-spectral projection, and least squares regression.

Pseudo-spectral projection

In the Pseudo-spectral projection approach, the final solution is projected against each basis function. Since the polynomials form orthonormal basis functions, the coefficients can be found by the projection of $\boldsymbol{x}(\boldsymbol{\xi})$ onto each basis function $\psi_j(\boldsymbol{\xi})$ [106]:

$$\left[\boldsymbol{c}_p\right]_j = \frac{1}{\langle \Psi_j \rangle^2} \int \boldsymbol{x}(\boldsymbol{\xi}) \Psi_j(\boldsymbol{\xi})\, p(\boldsymbol{\xi})\, d\boldsymbol{\xi} \tag{3.40}$$

Where $p(\boldsymbol{\xi})$ represents the probability density of the solution, integrated over the full probability space, and $\langle \Psi_j \rangle^2$ is the norm squared of the multivariate polynomial, which can be found by the product of the norm squared of each univariate polynomial function [106]:

$$\langle \Psi_j \rangle^2 = \prod_{i=1}^{n_{state}} \langle \psi_i^j \rangle^2 \tag{3.41}$$

In pseudo-spectral projection, this integral is approximated by a finite set of points in a quadrature grid of order Q: $\boldsymbol{X}^Q = \left\{\boldsymbol{x}(\boldsymbol{\xi}_1), \ldots, \boldsymbol{x}\left(\boldsymbol{\xi}_{N_{grid}}\right)\right\}$, such that the numerical integration becomes:

$$c_j = \frac{1}{\langle \Psi_j \rangle^2} \sum_{q=1}^{N_{grid}} x(\boldsymbol{\xi}_q) \Psi_j(\boldsymbol{\xi}_q)\, w_q \tag{3.42}$$

where w_q are the weights associated to the nodes q of a quadrature grid, and N_{grid} is the number of points in the quadrature grid, which depends on the choice of grid structure. For the choice of grid points, several quadrature rules are available. This study includes the Gaussian quadrature rules, where the grid points are distributed based on the PDF of each random input ξ_i. To reduce the effects of dimensionality on the number of grid points, sparse Smolyak grids are used to preserve the 1D integration properties, while requiring significantly fewer grid points [44].

Least Squares Regression

In the regression approach, random samples are used to find a set of coefficients that minimises the sum of the squared differences between the solution and estimation of each sample. The samples are generated at t_0 according to:

$$\boldsymbol{x}_{t_0}(\boldsymbol{\xi}) = \overline{\boldsymbol{x}}_{t_0} + \boldsymbol{S}\boldsymbol{\xi} \quad \text{with} \quad \boldsymbol{P} = \boldsymbol{S}\boldsymbol{S}^T \tag{3.43}$$

And propagated to t to produce $\boldsymbol{x}_t(\boldsymbol{\xi})$. To find the coefficients $\boldsymbol{c}_p$ that minimise the sum of squared differences, the following cost function is used [44]:

$$\left[c_p\right]_j \approx \arg\min_{\left\{\left[\tilde{c}_p\right]_j(t)\right\}} \frac{1}{N_{samples}} \sum_{i=1}^{N_{samples}} \left(\boldsymbol{x}_t(\boldsymbol{\xi}_i) - \sum_{j}^{P_e} \left[\tilde{c}_p\right]_j(t) \Psi_j(\boldsymbol{\xi}_i) \right)^2 \tag{3.44}$$

where $N_{samples}$ is the total number of samples and P_e is the order of the expansion used. The accuracy of the least squares regression approximation depends on the ratio between the number of samples used $N_{samples}$ and the number of coefficients P_{coef}, which is further denoted as N/P.

Chaospy

The process of generating an expansion, creating a quadrature integration grid or random samples, and solving for the coefficients using pseudo-spectral projection or least squares regression, can all be done efficiently using the Python package Chaospy [107]. In this study, the entire propagation of uncertainty using PCEs is done via Chaospy, where only the expansion order P_e, quadrature order Q and sample ratio N/P are adjusted as settings.

3.8. Comparing Propagated Probability Distributions

The various methods described above produce different types of PDFs at t_f. Studying the effectiveness of the various methods requires a way to compare these final uncertainties. A metric that can compare two probability distributions, and can be applied to different types of PDFs, is the L_2 distance, which gives a measure of how different two distributions are from each other.

L_2 Distance

The L_2 distance between two probability distributions can be calculated by the following formula [108]:

$$L_2(p_1, p_2) = \int_{\mathbb{R}^n} \left(p_1(\boldsymbol{x}) - p_2(\boldsymbol{x})\right)^2 d\boldsymbol{x} \tag{3.45}$$

The L_2 distance is both symmetric and satisfies the triangle inequality $L_2(p_1, p_2) \leq L_2(p_1) + L_2(p_2)$. The symmetry here comes from the square in the integral, and the triangle inequality can be proven by rewriting Equation 3.45 to:

$$L_2(p_1, p_2) = \int_{\mathbb{R}^n} \left(p_1(\boldsymbol{x}) - p_2(\boldsymbol{x})\right)^2 d\boldsymbol{x} = \int_{\mathbb{R}^n} p_1(\boldsymbol{x})^2 d\boldsymbol{x} + \int_{\mathbb{R}^n} p_2(\boldsymbol{x})^2 d\boldsymbol{x} - 2\int_{\mathbb{R}^n} p_1(\boldsymbol{x})\, p_2(\boldsymbol{x})\, d\boldsymbol{x} \tag{3.46}$$

$$= L_2(p_1) + L_2(p_2) - 2\int_{\mathbb{R}^n} p_1(\boldsymbol{x})\, p_2(\boldsymbol{x})\, d\boldsymbol{x} \tag{3.47}$$

where $p_1(\boldsymbol{x})$ and $p_2(\boldsymbol{x})$ are the PDFs of the two probability distributions and are both defined as positive in the entire domain of integration.

Using this triangle property, the L_2 distance can be normalised to a value $0 \leq NL_2(p_1, p_2) \leq 1$ as follows:

$$NL_2(p_1, p_2) = \frac{\int_{\mathbb{R}^n} \left(p_1(\boldsymbol{x}) - p_2(\boldsymbol{x})\right)^2 d\boldsymbol{x}}{\int_{\mathbb{R}^n} p_1(\boldsymbol{x})^2 d\boldsymbol{x} + \int_{\mathbb{R}^n} p_2(\boldsymbol{x})^2 d\boldsymbol{x}} = 1 - \frac{2\int_{\mathbb{R}^n} p_1(\boldsymbol{x})\, p_2(\boldsymbol{x})\, d\boldsymbol{x}}{\int_{\mathbb{R}^n} p_1(\boldsymbol{x})^2 d\boldsymbol{x} + \int_{\mathbb{R}^n} p_2(\boldsymbol{x})^2 d\boldsymbol{x}} \tag{3.48}$$

By substituting the PDF for a GMM from Equation 3.32 into each term from Equation 3.46, the equation for the

L_2 distance between two GMMs can be found:

$$\int_{\mathbb{R}^n} p_1(\boldsymbol{x})\, p_2(\boldsymbol{x})\, d\boldsymbol{x} = \int_{\mathbb{R}^n} \left[\sum_{i=1}^{k_1} w_{1,i}\mathcal{N}\left(\boldsymbol{x}:\overline{\boldsymbol{x}}_{1,i},\boldsymbol{P}_{1,i}\right)\right]\left[\sum_{j=1}^{k_2} w_{2,j}\mathcal{N}\left(\boldsymbol{x}:\overline{\boldsymbol{x}}_{2,j},\boldsymbol{P}_{2,j}\right)\right] d\boldsymbol{x} \tag{3.49}$$

$$= \int_{\mathbb{R}^n} \sum_{i=1}^{k_1}\sum_{j=1}^{k_2} w_{1,i}\, w_{2,j}\mathcal{N}\left(\boldsymbol{x}:\overline{\boldsymbol{x}}_{1,i},\boldsymbol{P}_{1,i}\right)\mathcal{N}\left(\boldsymbol{x}:\overline{\boldsymbol{x}}_{2,j},\boldsymbol{P}_{2,j}\right) \tag{3.50}$$

$$= \sum_{i=1}^{k_1}\sum_{j=1}^{k_2} w_{1,i}\, w_{2,j}\int_{\mathbb{R}^n} \mathcal{N}\left(\boldsymbol{x}:\overline{\boldsymbol{x}}_{1,i},\boldsymbol{P}_{1,i}\right)\mathcal{N}\left(\boldsymbol{x}:\overline{\boldsymbol{x}}_{2,j},\boldsymbol{P}_{2,j}\right) \tag{3.51}$$

Similarly for the other two terms:

$$\int_{\mathbb{R}^n} p_1(\boldsymbol{x})^2\, d\boldsymbol{x} = \sum_{i=1}^{k_1}\sum_{j=1}^{k_1} w_{1,i}\, w_{1,j}\int_{\mathbb{R}^n} \mathcal{N}\left(\boldsymbol{x}:\overline{\boldsymbol{x}}_{1,i},\boldsymbol{P}_{1,i}\right)\mathcal{N}\left(\boldsymbol{x}:\overline{\boldsymbol{x}}_{1,j},\boldsymbol{P}_{1,j}\right) \tag{3.52}$$

$$\int_{\mathbb{R}^n} p_2(\boldsymbol{x})^2\, d\boldsymbol{x} = \sum_{i=1}^{k_2}\sum_{j=1}^{k_2} w_{2,i}\, w_{2,j}\int_{\mathbb{R}^n} \mathcal{N}\left(\boldsymbol{x}:\overline{\boldsymbol{x}}_{2,i},\boldsymbol{P}_{2,i}\right)\mathcal{N}\left(\boldsymbol{x}:\overline{\boldsymbol{x}}_{2,j},\boldsymbol{P}_{2,j}\right) \tag{3.53}$$

The product of two Gaussian PDFs can be computed with a single scaled Gaussian PDF [108]:

$$\mathcal{N}\left(\boldsymbol{x}:\overline{\boldsymbol{x}}_1,\boldsymbol{P}_1\right)\mathcal{N}\left(\boldsymbol{x}:\overline{\boldsymbol{x}}_2,\boldsymbol{P}_2\right) = \mathcal{N}\left(\overline{\boldsymbol{x}}_1:\overline{\boldsymbol{x}}_2,\boldsymbol{P}_1+\boldsymbol{P}_2\right)\mathcal{N}\left(\boldsymbol{x}:\overline{\boldsymbol{x}}_{sc},\boldsymbol{P}_{sc}\right) \tag{3.54}$$

where the new Gaussian PDF can be computed with:

$$\overline{\boldsymbol{x}}_{sc} = \boldsymbol{P}_{sc}\left(\boldsymbol{P}_1^{-1}\overline{\boldsymbol{x}}_1 + \boldsymbol{P}_2^{-1}\overline{\boldsymbol{x}}_2\right) \quad \text{and} \quad \boldsymbol{P}_{sc} = \left(\boldsymbol{P}_1^{-1}+\boldsymbol{P}_2^{-1}\right)^{-1} \tag{3.55}$$

Since the integral of a Gaussian PDF over the entire probability domain is always 1, the integral of the product of two GMMs then reduces to the scaling factor:

$$\int_{\mathbb{R}^n} \mathcal{N}\left(\boldsymbol{x}:\overline{\boldsymbol{x}}_1,\boldsymbol{P}_1\right)\mathcal{N}\left(\boldsymbol{x}:\overline{\boldsymbol{x}}_2,\boldsymbol{P}_2\right) = \mathcal{N}\left(\overline{\boldsymbol{x}}_1:\overline{\boldsymbol{x}}_2,\boldsymbol{P}_1+\boldsymbol{P}_2\right) \tag{3.56}$$

$$= \frac{1}{\sqrt{\det\left(2\pi\left(\boldsymbol{P}_1+\boldsymbol{P}_2\right)\right)}}\exp\left[-\frac{1}{2}\left(\overline{\boldsymbol{x}}_1-\overline{\boldsymbol{x}}_2\right)^T\left(\boldsymbol{P}_1+\boldsymbol{P}_2\right)^{-1}\left(\overline{\boldsymbol{x}}_1-\overline{\boldsymbol{x}}_2\right)\right] \tag{3.57}$$

Thus, the L_2 distance between two GMMs becomes:

$$L_2\left(p_1,p_2\right) = d_{1,1} + d_{2,2} - 2d_{1,2} \tag{3.58}$$

and the NL_2 distance:

$$NL_2\left(p_1,p_2\right) = 1 - \frac{2d_{1,2}}{d_{1,1}+d_{2,2}} \tag{3.59}$$

where

$$d_{1,2} = \sum_{i=1}^{k_1}\sum_{j=1}^{k_2} w_{1,i}\, w_{2,j}\mathcal{N}\left(\overline{\boldsymbol{x}}_{1,i}:\overline{\boldsymbol{x}}_{2,j},\boldsymbol{P}_{1,i}+\boldsymbol{P}_{2,j}\right) \tag{3.60}$$

$$d_{1,1} = \sum_{i=1}^{k_1}\sum_{j=1}^{k_1} w_{1,i}\, w_{1,j}\mathcal{N}\left(\overline{\boldsymbol{x}}_{1,i}:\overline{\boldsymbol{x}}_{1,j},\boldsymbol{P}_{1,i}+\boldsymbol{P}_{1,j}\right) \tag{3.61}$$

$$d_{2,2} = \sum_{i=1}^{k_2}\sum_{j=1}^{k_2} w_{2,i}\, w_{2,j}\mathcal{N}\left(\overline{\boldsymbol{x}}_{2,i}:\overline{\boldsymbol{x}}_{2,j},\boldsymbol{P}_{2,i}+\boldsymbol{P}_{2,j}\right) \tag{3.62}$$

Gaussian Mixture Model Fitting on Data

A final uncertainty with a probability distribution that consists of a large set of points, such as that generated by MC, can not be compared using the closed form expressions of the NL_2 found above. However, a set of points can be approximated with a GMM by using the expectation-maximisation algorithm to fit a Gaussian mixture model to the data. This study uses Scikit-learn [109] to fit GMMs to the samples of the probability distributions, such that these uncertainties can be compared to other Gaussian distributions and GMMs.

4

Conjunction Assessment

Once a close encounter between two space objects is identified [62], there are various methods to determine the probability of a collision. The collision probability can be determined at a single epoch, which is defined as the instantaneous collision probability. More often, a cumulative collision probability is determined, which integrates the instantaneous collision probabilities over the duration of the close encounter. These close encounters are separated into two categories: short-term encounters and long-term encounters. The short-term encounters consider a high relative velocity between the two objects (in the range of kilometres per second) and often make a number of assumptions on the situation to simplify the computation of the collision probability. The long-term encounters consider low relative velocities resulting in longer time periods of the close encounter and thus fewer simplifying assumptions can be used.

To calculate the instantaneous collision probability, the position covariances of both objects (if their uncertainties are uncorrelated) are summed to a joint covariance matrix [63]:

$$\boldsymbol{P}_J^r = \boldsymbol{P}_P^r + \boldsymbol{P}_S^r \tag{4.1}$$

where $\boldsymbol{P}_P^r$ and $\boldsymbol{P}_S^r$ are the position covariance matrices of the primary and secondary space object respectively. Then, the instantaneous collision probability of the two space objects is given by the integration of the corresponding joint Gaussian PDF, centred on the secondary object, over the volume of a hardball sphere with the radius HBR $= R_P + R_S$ centred on the primary object:

$$P_{C,i}(t) = \iiint_{\tilde{V}} \mathcal{N}\left(\boldsymbol{r}; \boldsymbol{r}_{rel}, \boldsymbol{P}_J^r\right) \mathrm{d}V \tag{4.2}$$

Here $\mathcal{N}\left(\boldsymbol{r}; \boldsymbol{r}_{rel}, \boldsymbol{P}_J^r\right)$ represents the 3-dimensional combined Gaussian PDF, $\boldsymbol{r}_{rel}$ denotes the relative position of the primary object with respect to the secondary object and R_P and R_S are the radii of the primary and secondary object respectively. An illustration of the integration volume $\tilde{V}$ is shown in Figure 4.1.

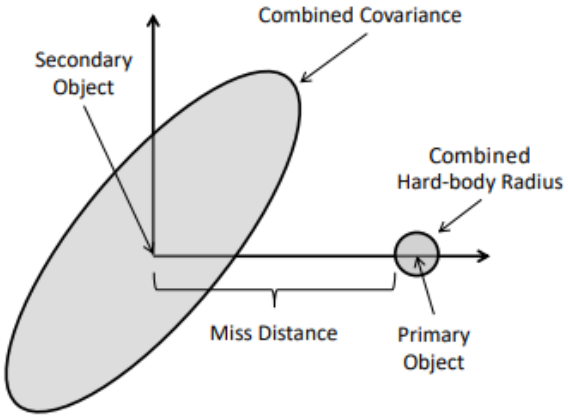


Figure 4.1: *2D rendering of the integration to P_C in the conjunction plane, image from [110].*

To determine the cumulative P_C, the instantaneous collision probability $P_{C,i}$ can be integrated over the duration of the conjunction [63]:

$$P_C = \int_{t_1}^{t_2} P_{C,i}(t)\, \mathrm{d}t \tag{4.3}$$

Where t_1 and t_2 define the duration of the close encounter. Usually t_1 is chosen at an epoch far enough away to assume a cumulative collision probability of 0 up to that point [87].

4.1. Analytical method

For short-term encounters, analyses to compute the collision probability generally make the following assumptions [63]:

- The relative velocity of the two space objects during the close approach is considered constant.
- There is no uncertainty in the velocity of the space objects during the encounter.
- The position uncertainty is constant during the encounter and equal to the uncertainty at the TCA.
- The position uncertainty of the two objects are uncorrelated and follow Gaussian distributions.
- The two objects are spherically shaped.

Using these assumptions, the time integral can be converted to a 3-dimensional volume integral, similar to the formula described in Equation 4.2 [63]:

$$P_C = \iiint_{V'} \mathcal{N}\left(\boldsymbol{r}; \boldsymbol{r}_{rel,TCA}, \boldsymbol{P}^r_{J,TCA}\right) \mathrm{d}V \tag{4.4}$$

Except now, V' refers to the volume swept out by a hardball sphere (with radius HBR = $R_P + R_S$) centred on the primary object, and the combined uncertainty is constant and taken as the uncertainty at the time of closest approach $t = t_{TCA}$. An illustration of the swept out volume is given in Figure 4.2.

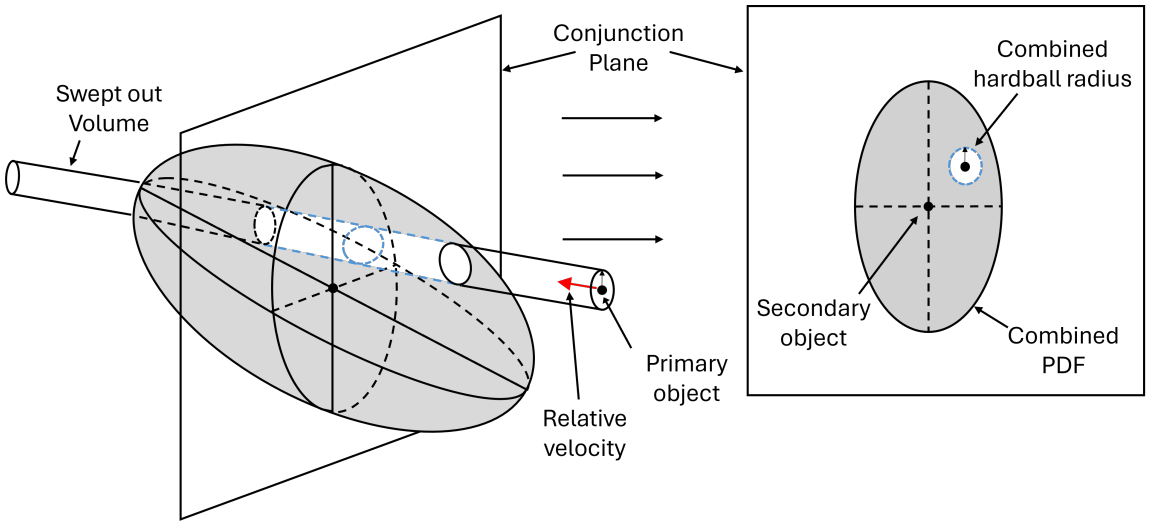


Figure 4.2: *3D and 2D schematic of the integration to P_C in a short-term close encounter.*

From Figure 4.2 it can be seen that when the relative velocity and the combined position uncertainty are constant throughout the short-term encounter, the 3-dimensional volumetric integral can be reduced to a 2-dimensional surface integral on the conjunction plane, by integrating along the relative velocity direction. This simplifies the collision probability computation problem to Equation 4.5 and is the basis for various numerical and analytical methods that expand on this with further derivations and methods [63].

$$P_C = \iint_{A_j} h(x, y) \, \mathrm{d}S \tag{4.5}$$

Where $h(x, y)$ is the combined PDF in the conjunction plane, obtained by integrating the combined PDF over the relative velocity direction, and A_j represents the area of the combined radii hardball sphere projected on the conjunction plane. The conjunction plane is defined as the plane orthogonal to the relative velocity vector, passing through the mean position of the secondary object.

4.2. Monte Carlo Collision Probability

In cases of low relative velocities, the encounter becomes a long-term encounter and the assumptions made for short-term encounters no longer hold. The computation of P_C in long-term encounters is more complex than that for the short-term [63]. Most methods assume a Gaussian distribution for both space objects' uncertainties. By dividing a non-Gaussian PDF into a number of Gaussian components like by the GMMs described in Chapter 3, these methods can be used on each Gaussian component.

To compute P_C accurately without requiring simplifying assumptions, both for instantaneous, short-term and long-term collision probabilities, MC simulations can be used. To compute the P_C using MC simulation, a sufficient number of random samples must be produced from the PDFs, each of which are propagated through the close encounter. By determining for each pair of samples if a collision has occurred, the collision probability can be approximated with [63]:

$$P_C = \frac{N_c}{N_t} \tag{4.6}$$

where N_c is the number of samples resulting in a collision and N_t is the total number of samples generated. The collision can be determined by comparing the combined radii of the space objects with the distance between both objects at the time of interest for the instantaneous collision probability, and by comparing it with the distance at closest approach (DCA) for the cumulative collision probabilities. The standard deviation of the estimated P_C is then [63]:

$$\sigma = \sqrt{\frac{P_C\,(1-P_C)}{N_t}} \tag{4.7}$$

To ensure that the standard deviation of the result is low enough to make the estimate accurate, sufficiently many runs should be generated. The number of samples required also increases when the collision probability becomes small, which is often around 10^{-4} or lower for space objects [45]. Figure 4.3 shows the number of samples required to obtain various relative error levels for a given confidence level of $\beta = 97\%$ [63].

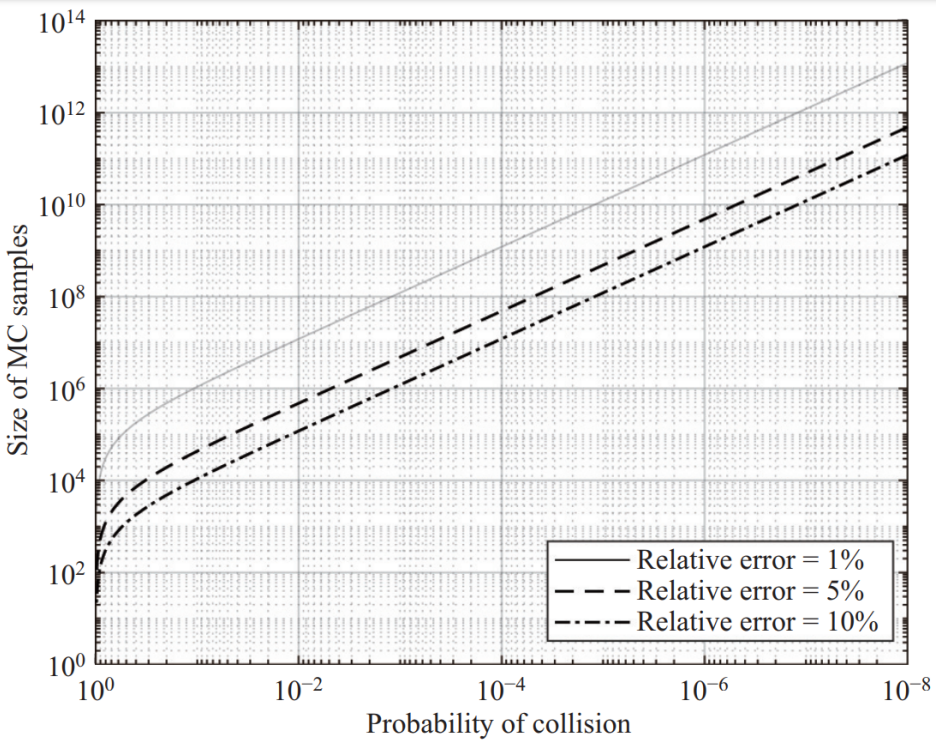


Figure 4.3: *The number of MC samples required to compute P_C with $\beta = 97\%$, image from [63].*

Two concepts of computing P_C are using an all-on-all or one-on-one analysis. With MC simulations, it is common to use the one-on-one technique. By choosing random pairs of samples and evaluating these pairs for a

collision, the collision probability can be estimated, which approaches the true value if enough random pairs are evaluated [111]. Some more complex P_C computation techniques include all-on-all analyses, such as when working with GMMs. In an all-on-all analysis, each component is paired with each other component to evaluate its collision probability. This can be useful when the number of points or components is much smaller. With GMMs this technique is often used to evaluate two PDFs modelled by a Gaussian Mixture, where each pair of Gaussian components can be evaluated using more advanced numerical or analytical computation methods [95].

To determine whether a collision occurs between two sampled states, both initial states are propagated up to t_{TCA}. If the distance between the position of both states at t_{TCA} is smaller than the HBR, then a collision is counted. However, for each sample pair, the actual moment at which the distance between both positions is smallest can deviate from the t_{TCA} found for the mean states. Therefore, a distinction is made between the t_{TCA} found for the mean states: $t_{TCA,mean}$, and the actual t_{TCA} for each sample pair: $t_{TCA,true}$. The count of collisions can be determined by evaluating the distance between all pairs at the same time epoch of $t_{TCA,mean}$, or by evaluating the distance between the pairs at the $t_{TCA,true}$, which is a different time epoch for each sample pair.

The $t_{TCA,true}$ can be found by propagating both sampled states backwards and forwards from the $t_{TCA,mean}$ over a small time window, and applying an algorithm to find the exact time at which the distance is minimum, for which multiple options are available. The original brute-force Monte Carlo method developed by the NASA Conjunction Assessment Risk Analysis (CARA) team used equally spaced bins in time [112], which has later been updated with an algorithm to find all local minima more accurately [113]. Another often used method is Denenberg's algorithm, which can find the 4 extrema of each orbital period by the use of a fitted polynomial function [114].

In this study, a bisection method, described in Algorithm 2 in Appendix A, is used to find the local minimum efficiently. The algorithm is based on the algorithm used by the CARA team, and works under the assumption that only one minimum is present in the bounded search interval. This assumption can be used when the interval is much smaller than the smallest orbital period of the two objects, and sufficiently large to ensure the minimum is found within the interval. The backward and forward propagation from $t_{TCA,mean}$ in these algorithms is often done with a simpler and more efficient propagation model than the model used to propagate the uncertainty to $t_{TCA,mean}$, for example using Keplerian dynamics [112].

Instead of propagating the state vector of both satellites from t_0 to $t_{TCA,mean}$ for all $N_{samples}$ points, the states can also be sampled from a Gaussian distribution created at $t_{TCA,mean}$ from the propagated sampled points using Modified Equinoctial Elements (MEE). The state vectors at $t_{TCA,mean}$ of both objects for all propagated points are first converted from Cartesian coordinates to MEE using Equations B.11 to B.16 from Appendix B. Then, the mean vector and covariance matrix are obtained using:

$$\overline{\boldsymbol{x}}_{mee} = \frac{1}{N_{samples}} \sum_{i=0}^{N_{samples}-1} \boldsymbol{x}_{i,mee} \tag{4.8}$$

$$\boldsymbol{P}_{mee} = \frac{1}{N_{samples}} \sum_{i=0}^{N_{samples}-1} \left(\boldsymbol{x}_{i,mee} - \overline{\boldsymbol{x}}_{mee}\right)\left(\boldsymbol{x}_{i,mee} - \overline{\boldsymbol{x}}_{mee}\right)^T \tag{4.9}$$

With the mean vector and covariance matrix in MEE, $N_{samples}$ state vectors can be sampled using

$$\boldsymbol{x}_{mee}(\boldsymbol{\xi}) = \overline{\boldsymbol{x}}_{mee} + \boldsymbol{S}_{mee}\boldsymbol{\xi} \tag{4.10}$$

where $\boldsymbol{S}$ is the square root matrix of the covariance matrix in MEE, such that $\boldsymbol{P}_{mee} = \boldsymbol{S}_{mee}\boldsymbol{S}_{mee}^T$, and $\boldsymbol{\xi} \sim \mathcal{N}(0, \mathrm{I}_6)$. Finally, all sampled state vectors are converted to Cartesian coordinates using Equations B.11 to B.16.

To summarise the available methods of computing the P_C with MC samples, there are three main choices to be made. The samples can be evaluated in an all-on-all or one-on-one analysis, where this research will implement a one-on-one analysis. Additionally, the samples can be evaluated at $t_{TCA,mean}$, or at the calculated $t_{TCA,true}$ for each sample pair. The latter option is used further in this study, where the $t_{TCA,true}$ is calculated using Algorithm 2, described in Appendix A. Finally, the required samples at t_{TCA} can be propagated from t_0, or sampled from a propagated PDF at t_{TCA}. Again, the latter option is used in the conjunction assessment study in this research, such that the PDFs can be propagated with fewer samples than the number of samples needed to calculate P_C.

5

Uncertainty Propagation Analysis Scenarios

The various uncertainty propagation (UP) methods described in Chapter 3 are evaluated with different settings for a set of test case scenarios, such as to compare their efficiency and accuracy in propagating different types of uncertainties with orbital dynamics. The efficiency and accuracy are determined by the computation times of the uncertainty propagation and the NL_2 distance between the propagated uncertainty and an uncertainty propagated with 10^5 MC samples used as a baseline. The test case scenarios are based on currently active satellites, orbiting around the Earth in unique types of orbits. The initial uncertainty of each test case scenario is based on the measurement uncertainty published by Privateer Wayfinder [115], and represents a realistic uncertainty, to be used in the uncertainty propagations of RSOs.

A total of five objects are selected, of which two objects are in a low Earth orbit (LEO), one object is in a geostationary orbit (GEO) and two objects have a highly elliptical orbit (HEO). Important specifications of the five satellites and their orbits are presented in Table 5.1. The mass and average cross section are obtained from ESA DISCOS (Database and Information System Characterising Objects in Space)[1].

Name	Orbit type	a $[km]$	e $[-]$	i $[deg]$	T $[s]$	m $[kg]$	Average cross section $[m^2]$
VELOX C1	LEO	6857.2	0.0020	14.9786	5652	123	0.348
ONEWEB-0416	LEO	7605.7	0.0012	87.9960	6600	148	2.032
COSMOS 2518	Molniya	26548.0	0.7191	63.0748	43050	2550	14.304
SPIRALE B	HEO	20311.8	0.6725	1.9085	28800	117	1.379
METEOSAT 12	GEO	42165.9	0.0002	0.7341	86172	3760	19.366

Table 5.1: *Characteristic information about the different test case scenarios.*

In this study, a pre-developed propagation tool is used to propagate the initial states forward in time. This propagator is used to analyse the scenario, as well as to propagate any sample points required in the uncertainty propagation methods. The propagation software used is the TU Delft Astrodynamics Toolbox (Tudat)[2]. In Tudat, the position ephemerides of third-bodies in the propagation, as well as the rotation model of the central body, are retrieved from SPICE [116, 117].

In the following sections, each of the five test case scenarios is described in more detail. This includes illustrations of the orbital trajectories, as well as the time epochs for the start and end of each propagation. Each section also includes the initial distribution of the uncertainty of each test case scenario, provided by the mean vector and the covariance matrix. Finally, for each test case scenario, a summary of the settings used for the propagation is included.

[1] https://discosweb.esoc.esa.int/
[2] https://docs.tudat.space/en/latest/

5.1. Velox scenario

Figure 5.1 shows an illustration of the orbit of the VELOX C1 satellite, on which the Velox test case scenario is based.

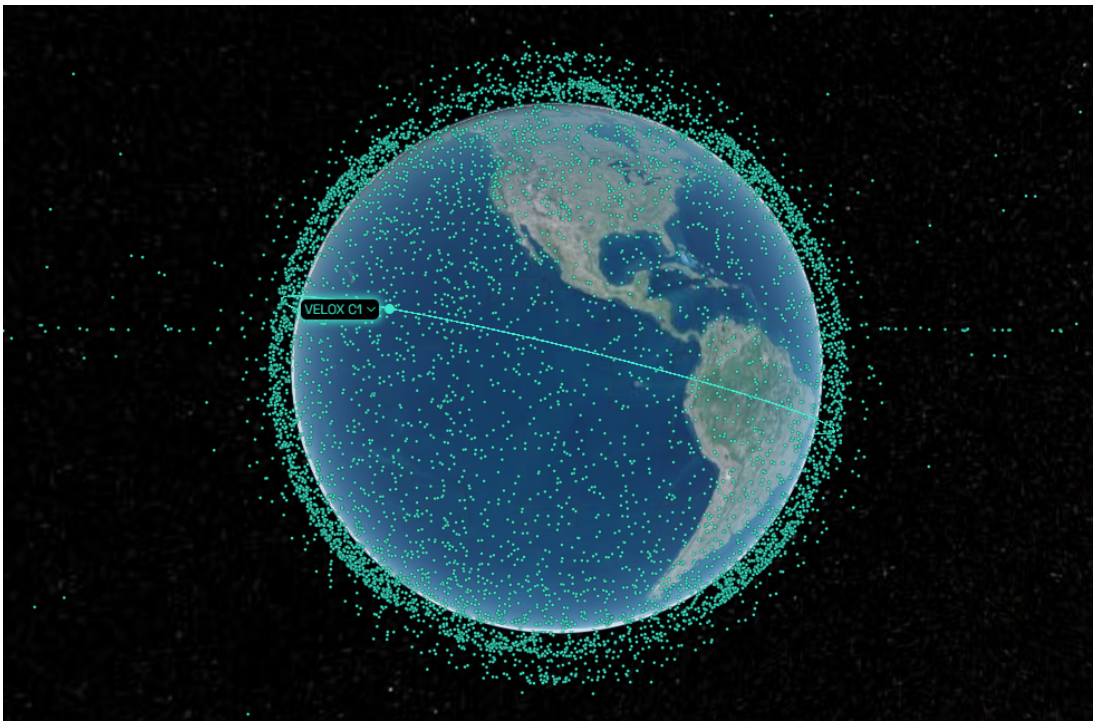


Figure 5.1: *Illustration of the orbit of the VELOX C1 satellite, at 18-02-2025 20:00:00 UTC via Wayfinder [115].*

Based on the available data of the satellite's state vectors and covariance matrices, the initial state of the test case scenario is selected at t_0 = 12-02-25 21:45:41.733 UTC, resulting in an initial state vector close to the orbit's perigee. The initial uncertainty, taken from data from Wayfinder [115], is characterised by a Gaussian distribution with the following values in the ECI coordinate frame:

$$\bar{\boldsymbol{x}} = \begin{bmatrix} -5.365E+6 & -4.249E+6 & 4.120E+4 & 4.593E+3 & -5.780E+3 & 1.965E+3 \end{bmatrix} \begin{bmatrix} m & m/s \end{bmatrix}$$

$$\boldsymbol{P} = \begin{bmatrix} 9.725E+3 & -4.153E+2 & 2.444E+2 & 1.183E+1 & 4.463E+0 & 6.775E-1 \\ -4.153E+2 & 8.767E+3 & -6.952E+1 & 3.254E-1 & 1.865E+0 & 2.313E-2 \\ 2.444E+2 & -6.952E+1 & 8.809E+3 & 1.559E+0 & 6.881E-1 & 1.816E+0 \\ 1.183E+1 & 3.254E-1 & 1.559E+0 & 1.915E-2 & 2.767E-3 & -7.530E-3 \\ 4.463E+0 & 1.865E+0 & 6.881E-1 & 2.767E-3 & 4.925E-2 & 3.177E-3 \\ 6.775E-1 & 2.313E-2 & 1.816E+0 & -7.530E-3 & 3.177E-3 & 6.039E-2 \end{bmatrix} \begin{bmatrix} m^2 & \frac{m^2}{s} \\ \frac{m^2}{s} & \frac{m^2}{2^2} \end{bmatrix}$$

Using Tudat, a single propagated orbit, starting from this initial state, is shown in Figure 5.2. The duration of this propagation is set to T = 5652.0 seconds, resulting in t_f = 12-02-25 23:19:53.732 UTC. For a simulation lasting N_{orbit} orbital revolutions, the propagation end time is determined by $t_f = t_0 + N_{orbit} \cdot T$, also shown in Table 5.2 for the various number of revolutions chosen to simulate.

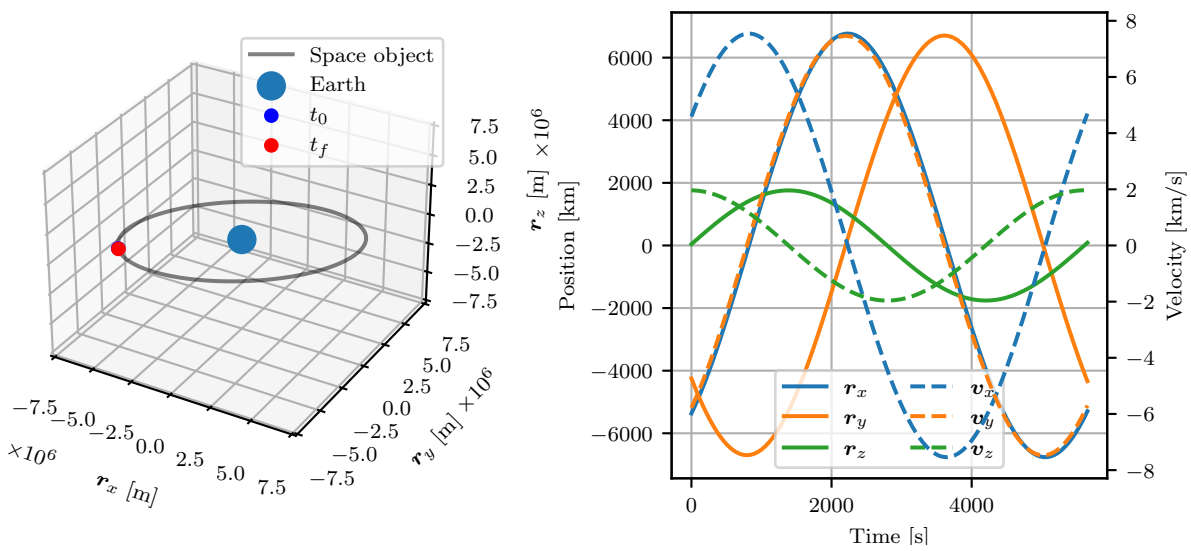


Figure 5.2: *A single propagated orbit of the Velox test case scenario.*

N_{orbit} [−]	t_0 [UTC]	t_f [UTC]
1	12-02-25 21:45:41.733	12-02-25 23:19:53.732
2	12-02-25 21:45:41.733	13-02-25 00:54:05.732
10	12-02-25 21:45:41.733	13-02-25 13:27:41.732
20	12-02-25 21:45:41.733	14-02-25 05:09:41.732
50	12-02-25 21:45:41.733	16-02-25 04:15:41.732

Table 5.2: *Propagation start and end times for different numbers of orbital revolutions simulated, for the Velox scenario.*

The propagator and integrator settings for the Velox scenario have been analysed for the maximum number of revolutions, $N_{orbit} = 50$, such that both the integration error and the model error (compared to the highest fidelity model available in Tudat) induced by the propagation are kept below $1m$. The full analysis is documented in Appendix C, and the final settings are repeated here for the high-fidelity and low-fidelity dynamics in Table 5.3 and Table 5.4.

Acceleration type	
Central body gravity (Earth)	100 x 100 (GOCO05c)
3rd body perturbations (point mass)	Sun, Moon
Solar radiation pressure	C_R = 1.5 (Cannonball)
Atmospheric drag	C_D = 2.0 (US76)
Environment models	
Central body rotation model	Constant rotation rate determined at t_0
3rd body position model (Moon)	Keplerian 2-body orbit determined at t_0
3rd body position model (Sun)	Retrieved every epoch from Spice
Propagator settings	
Integrator	variable rkdp87 (abs. tolerance: 2E-6)
Propagator	Cowell

Table 5.3: *Propagation settings used for the high-fidelity dynamics of the Velox scenario*

Acceleration type	
Central body gravity (Earth)	2 x 2 (GOCO05c)
Atmospheric drag	C_D = 2.0 (US76)
Environment models	
Central body rotation model	Constant rotation rate determined at t_0
Propagator settings	
Integrator	variable rkdp87 (abs. tolerance: 2E-6)
Propagator	Cowell

Table 5.4: *Propagation settings used for the low-fidelity dynamics of the Velox scenario*

5.2. Oneweb scenario

Figure 5.3 shows an illustration of the orbit of the ONEWEB-0416 satellite, on which the Oneweb test case scenario is based.

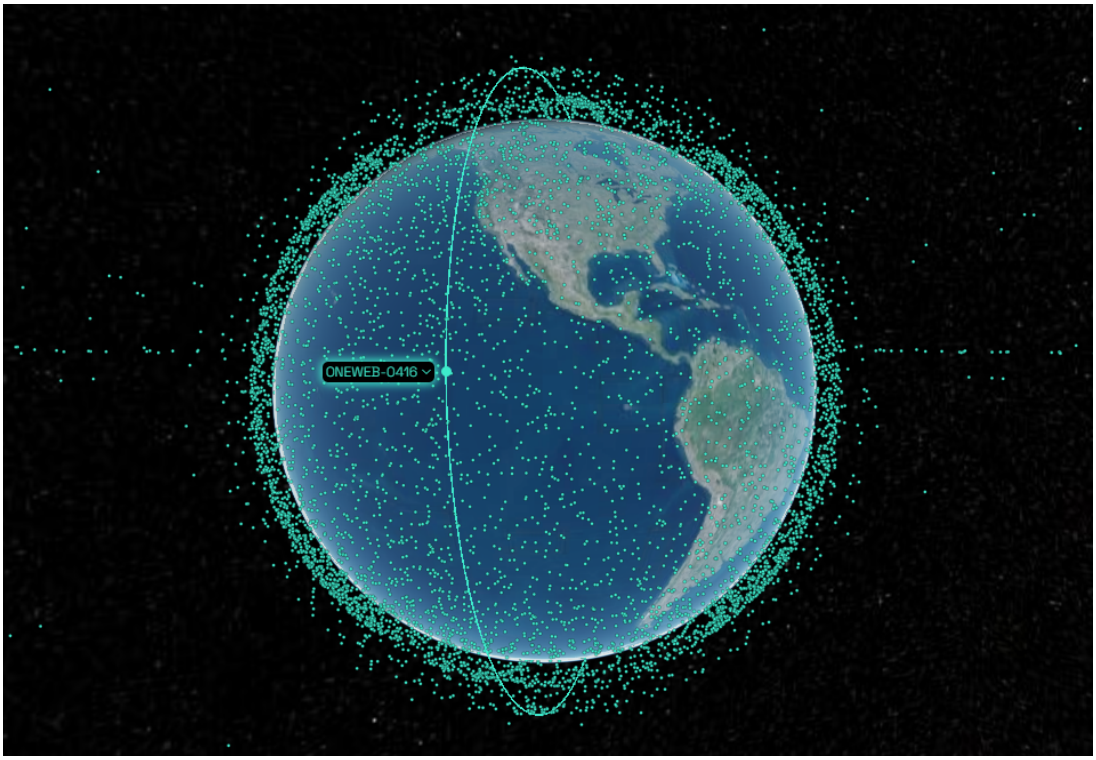


Figure 5.3: *Illustration of the orbit of the ONEWEB-0416 satellite, at 18-02-2025 20:00:00 UTC via Wayfinder [115].*

Based on the available data of the satellite's state vectors and covariance matrices, the initial state of the test case scenario is selected at t_0 = 13-02-25 06:37:02.992 UTC, resulting in an initial state vector close to the orbit's perigee. The initial uncertainty is characterised by a Gaussian distribution with the following values:

$$\bar{\boldsymbol{x}} = \begin{bmatrix} -3.044E+5 & -6.542E+4 & 7.585E+6 & 4.973E+3 & -5.267E+3 & 1.535E+2 \end{bmatrix} \begin{bmatrix} m & m/s \end{bmatrix}$$

$$\boldsymbol{P} = \begin{bmatrix} 9.167E+3 & -3.331E+2 & -3.901E+2 & 3.412E+0 & -4.321E-1 & -3.537E+0 \\ -3.331E+2 & 9.165E+3 & 3.976E+2 & -5.312E-1 & 3.223E+0 & 3.380E+0 \\ -3.901E+2 & 3.976E+2 & 9.126E+3 & -1.587E+0 & 1.318E+0 & 9.914E+0 \\ 3.412E+0 & -5.312E-1 & -1.587E+0 & 3.482E-2 & 6.663E-3 & 8.223E-3 \\ -4.321E-1 & 3.223E+0 & 1.318E+0 & 6.663E-3 & 3.487E-2 & -7.790E-3 \\ -3.537E+0 & 3.380E+0 & 9.914E+0 & 8.223E-3 & -7.790E-3 & 2.377E-2 \end{bmatrix} \begin{bmatrix} m^2 & \frac{m^2}{s} \\ \frac{m^2}{s} & \frac{m^2}{2^2} \end{bmatrix}$$

A single propagated orbit, starting from this initial state, is shown in Figure 5.4. The duration of this propagation is set to $T = 6600.0$ seconds, resulting in t_f = 13-02-25 08:27:02.991 UTC. For a simulation lasting N_{orbit} orbital revolutions, the propagation end time is determined by $t_f = t_0 + N_{orbit} \cdot T$, also shown in Table 5.5 for the various number of revolutions chosen to simulate.

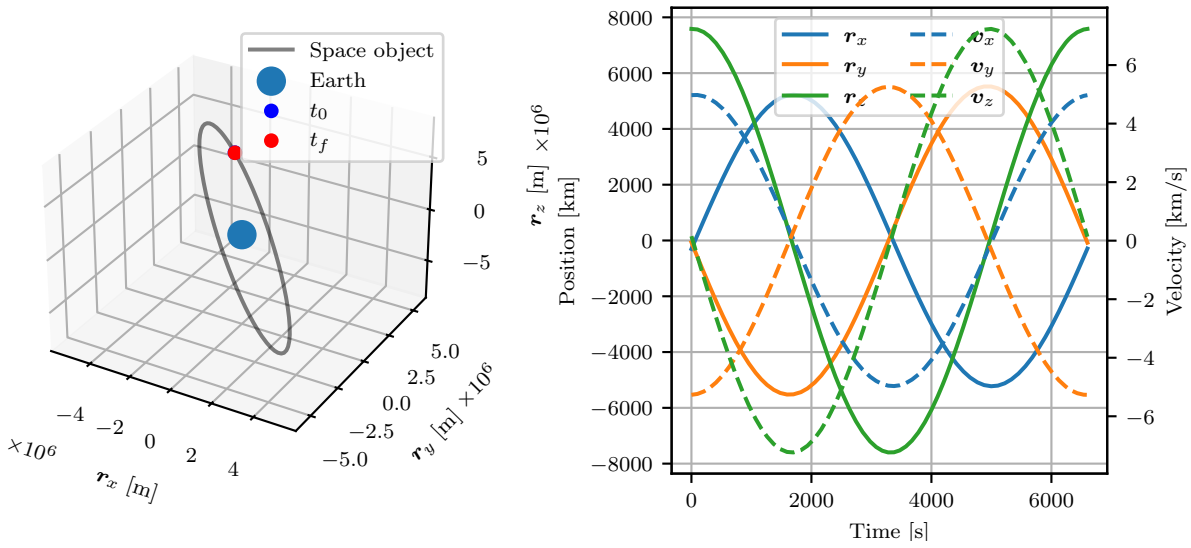


Figure 5.4: *A single propagated orbit of the Oneweb test case scenario.*

N_{orbit} [−]	t_0 [UTC]	t_f [UTC]
1	13-02-25 06:37:02.992	13-02-25 08:27:02.991
2	13-02-25 06:37:02.992	13-02-25 10:17:02.991
10	13-02-25 06:37:02.992	14-02-25 00:57:02.991
20	13-02-25 06:37:02.992	14-02-25 19:17:02.991
50	13-02-25 06:37:02.992	17-02-25 02:17:02.991

Table 5.5: *Propagation start and end times for different numbers of orbital revolutions simulated, for the Oneweb scenario.*

The propagator and integrator settings for the Oneweb scenario have been analysed for the maximum number of revolutions, $N_{orbit} = 50$, such that both the integration error and the model error (compared to the highest fidelity model available in Tudat) induced by the propagation are kept below $1m$. The full analysis is documented in Appendix C, and the final settings are repeated here for the high-fidelity and low-fidelity dynamics in Table 5.6 and Table 5.7.

Acceleration type	
Central body gravity (Earth)	100 x 100 (GOCO05c)
3rd body perturbations (point mass)	Sun, Moon
Solar radiation pressure	C_R = 1.5 (Cannonball)
Atmospheric drag	C_D = 2.0 (US76)
Environment models	
Central body rotation model	Constant rotation rate determined at t_0
3rd body position model (Moon)	Keplerian 2-body orbit determined at t_0
3rd body position model (Sun)	Retrieved every epoch from Spice
Propagator settings	
Integrator	variable rkdp87 (abs. tolerance: 2E-6)
Propagator	Cowell

Table 5.6: *Propagation settings used for the high-fidelity dynamics of the Oneweb scenario*

Acceleration type	
Central body gravity (Earth)	2 x 2 (GOCO05c)
Environment models	
Central body rotation model	Constant rotation rate determined at t_0
Propagator settings	
Integrator	variable rkdp87 (abs. tolerance: 2E-6)
Propagator	Cowell

Table 5.7: *Propagation settings used for the low-fidelity dynamics of the Oneweb scenario*

5.3. Cosmos scenario

Figure 5.5 shows an illustration of the orbit of the COSMOS 2518 satellite, on which the Cosmos test case scenario is based.

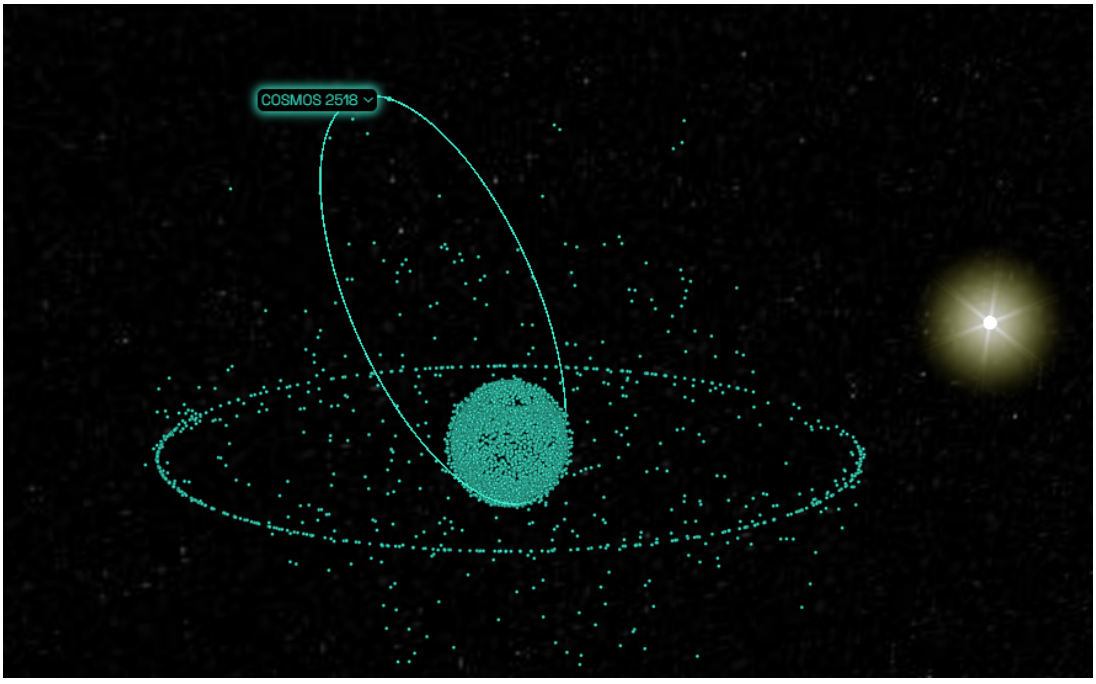


Figure 5.5: *Illustration of the orbit of the COSMOS 2518 satellite, at 18-02-2025 20:00:00 UTC via Wayfinder [115].*

Based on the available data of the satellite's state vectors and covariance matrices, the initial state of the test case scenario is selected at t_0 = 13-02-25 02:28:47.055 UTC, resulting in an initial state vector close to the orbit's perigee. The initial uncertainty is characterised by a Gaussian distribution with the following values:

$$\bar{\boldsymbol{x}} = \begin{bmatrix} -2.546E+5 & -4.494E+6 & -6.260E+6 & 6.861E+3 & -6.054E+3 & 2.150E+3 \end{bmatrix} \begin{bmatrix} m & m/s \end{bmatrix}$$

$$\boldsymbol{P} = \begin{bmatrix} 8.990E+3 & -1.610E+2 & 5.496E+2 & 4.011E+0 & 1.324E+0 & 3.056E+0 \\ -1.610E+2 & 8.625E+3 & -3.708E+1 & 5.416E-1 & 3.538E+0 & 1.350E+0 \\ 5.496E+2 & -3.708E+1 & 9.233E+3 & 2.027E+0 & 2.502E+0 & 9.390E+0 \\ 4.011E+0 & 5.416E-1 & 2.027E+0 & 2.670E-2 & 1.678E-3 & -7.867E-3 \\ 1.324E+0 & 3.538E+0 & 2.502E+0 & 1.678E-3 & 2.949E-2 & -2.846E-3 \\ 3.056E+0 & 1.350E+0 & 9.390E+0 & -7.867E-3 & -2.846E-3 & 1.822E-2 \end{bmatrix} \begin{bmatrix} m^2 & \frac{m^2}{s} \\ \frac{m^2}{s} & \frac{m^2}{2^2} \end{bmatrix}$$

A single propagated orbit, starting from this initial state, is shown in Figure 5.6. The duration of this propagation is set to $T = 43050.0$ seconds, resulting in t_f = 13-02-25 14:26:17.054 UTC. For a simulation lasting N_{orbit} orbital revolutions, the propagation end time is determined by $t_f = t_0 + N_{orbit} \cdot T$, also shown in Table 5.8 for the various number of revolutions chosen to simulate.

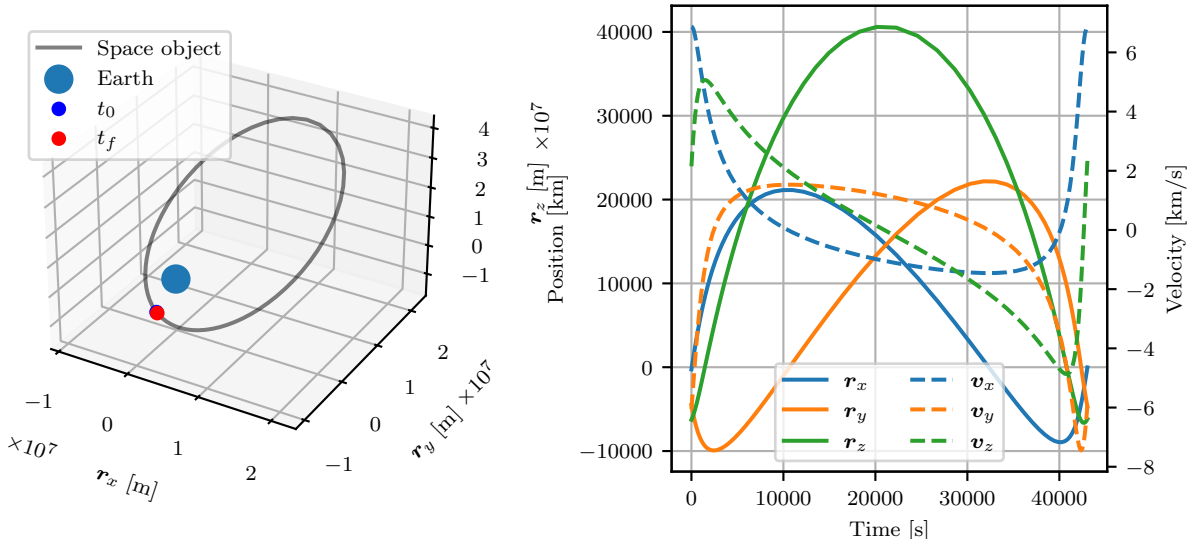


Figure 5.6: *A single propagated orbit of the Cosmos test case scenario.*

N_{orbit} [−]	t_0 [UTC]	t_f [UTC]
0.5	13-02-25 02:28:47.055	13-02-25 08:27:32.054
1	13-02-25 02:28:47.055	13-02-25 14:26:17.054
1.5	13-02-25 02:28:47.055	13-02-25 20:25:02.054
2	13-02-25 02:28:47.055	14-02-25 02:23:47.054

Table 5.8: *Propagation start and end times for different numbers of orbital revolutions simulated, for the Cosmos scenario.*

The propagator and integrator settings for the Cosmos scenario have been analysed for the maximum number of revolutions, $N_{orbit} = 2$, such that both the integration error and the model error (compared to the highest fidelity model available in Tudat) induced by the propagation are kept below $1m$. The full analysis is documented in Appendix C, and the final settings are repeated here for the high-fidelity and low-fidelity dynamics in Table 5.9 and Table 5.10.

Acceleration type	
Central body gravity (Earth)	100 x 100 (GOCO05c)
3rd body perturbations (point mass)	Sun, Moon, Venus
Solar radiation pressure	C_R = 1.5 (Cannonball)
Atmospheric drag	C_D = 2.0 (US76)
Relativistic Correction	Sun (Schwarzschild)
Environment models	
Central body rotation model	Constant rotation rate determined at t_0
3rd body position model (Moon, Venus)	Keplerian 2-body orbit determined at t_0
3rd body position model (Sun)	Retrieved every epoch from Spice
Propagator settings	
Integrator	variable rkdp87 (abs. tolerance: 1E-3)
Propagator	Cowell

Table 5.9: *Propagation settings used for the high-fidelity dynamics of the Cosmos scenario*

Acceleration type	
Central body gravity (Earth)	2 x 2 (GOCO05c)
3rd body perturbations (point mass)	Sun, Moon
Environment models	
Central body rotation model	Constant rotation rate determined at t_0
3rd body position model (Moon)	Keplerian 2-body orbit determined at t_0
3rd body position model (Sun)	Retrieved every epoch from Spice
Propagator settings	
Integrator	variable rkdp87 (abs. tolerance: 1E-3)
Propagator	Cowell

Table 5.10: *Propagation settings used for the low-fidelity dynamics of the Cosmos scenario*

5.4. Spirale scenario

Figure 5.7 shows an illustration of the orbit of the SPIRALE B satellite, on which the Spirale test case scenario is based.

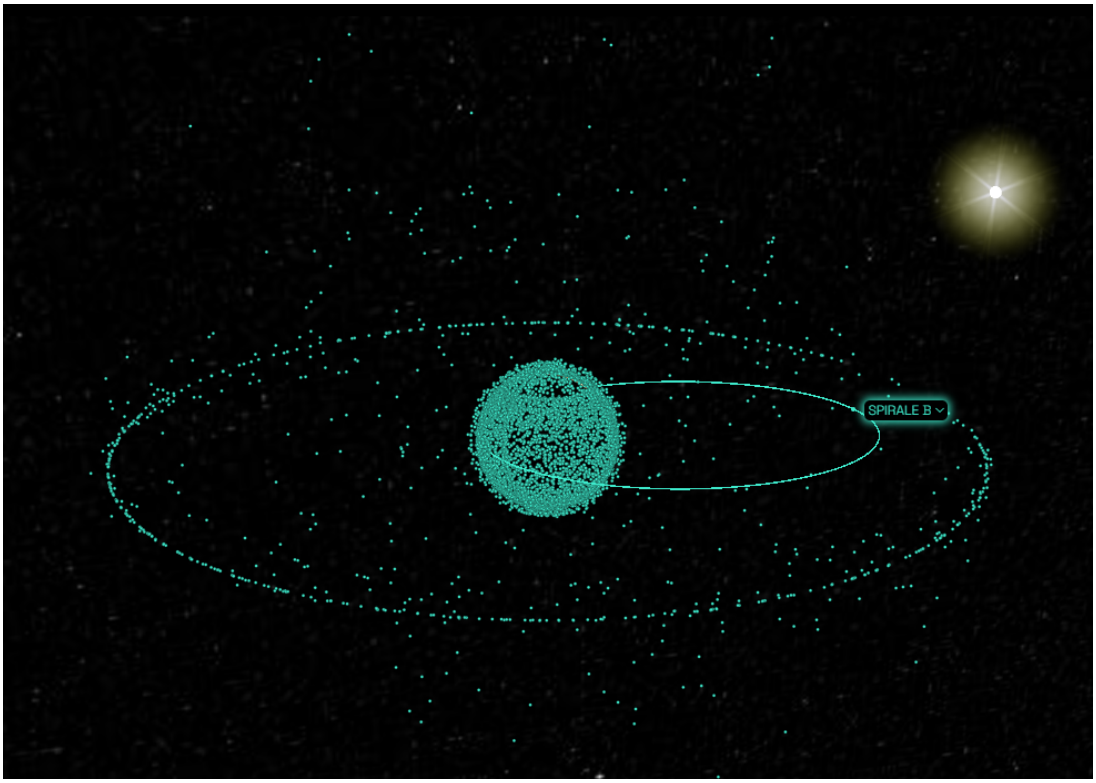


Figure 5.7: *Illustration of the orbit of the SPIRALE B satellite, at 18-02-2025 20:00:00 UTC via Wayfinder [115].*

Based on the available data of the satellite's state vectors and covariance matrices, the initial state of the test case scenario is selected at t_0 = 13-02-25 06:24:19.159 UTC, resulting in an initial state vector close to the orbit's perigee. The initial uncertainty is characterised by a Gaussian distribution with the following values:

$$\bar{\boldsymbol{x}} = \begin{bmatrix} 2.772E+6 & 6.216E+6 & 1.960E+5 & -9.483E+3 & 2.745E+3 & -2.042E+2 \end{bmatrix} \begin{bmatrix} m & m/s \end{bmatrix}$$

$$\boldsymbol{P} = \begin{bmatrix} 9.522E+3 & 2.765E+2 & 2.915E+1 & 7.296E+0 & 5.152E+0 & 2.237E-1 \\ 2.765E+2 & 8.865E+3 & 1.175E+1 & 3.673E+0 & 7.006E+0 & 1.827E-1 \\ 2.915E+1 & 1.175E+1 & 8.649E+3 & 1.948E-1 & 2.229E-1 & 2.777E+0 \\ 7.296E+0 & 3.673E+0 & 1.948E-1 & 1.896E-2 & -6.324E-3 & -5.099E-4 \\ 5.152E+0 & 7.006E+0 & 2.229E-1 & -6.324E-3 & 2.498E-2 & -3.324E-4 \\ 2.237E-1 & 1.827E-1 & 2.777E+0 & -5.099E-4 & -3.324E-4 & 3.316E-2 \end{bmatrix} \begin{bmatrix} m^2 & \frac{m^2}{s} \\ \frac{m^2}{s} & \frac{m^2}{2^2} \end{bmatrix}$$

A single propagated orbit, starting from this initial state, is shown in Figure 5.8. The duration of this propagation is set to $T = 28800$ seconds, resulting in t_f = 13-02-25 14:24:19.158 UTC. For a simulation lasting N_{orbit} orbital revolutions, the propagation end time is determined by $t_f = t_0 + N_{orbit} \cdot T$, also shown in Table 5.11 for the various number of revolutions chosen to simulate.

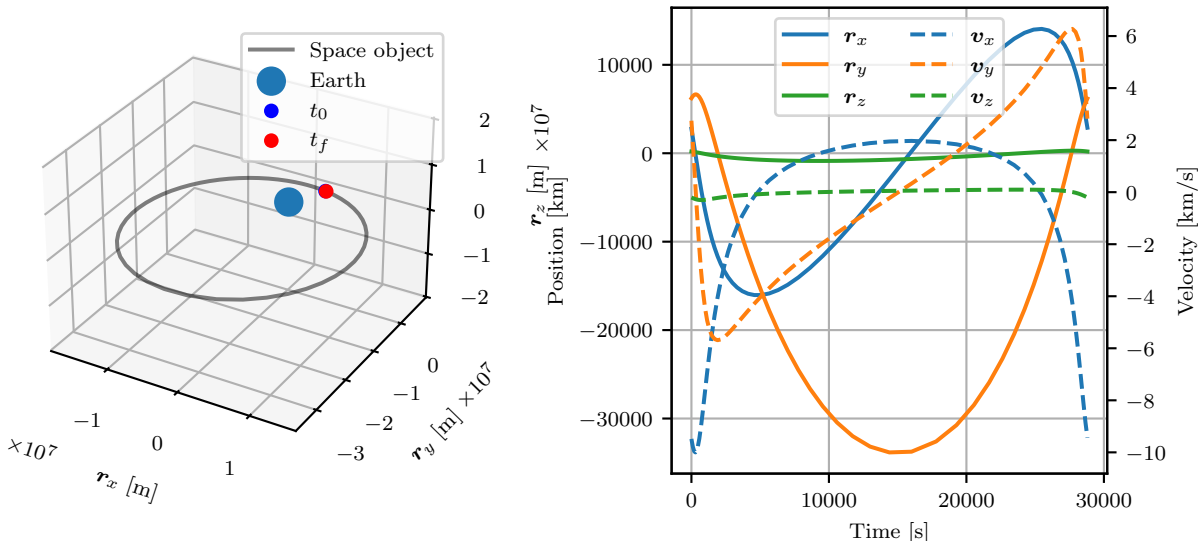


Figure 5.8: *A single propagated orbit of the Spirale test case scenario.*

N_{orbit} [−]	t_0 [UTC]	t_f [UTC]
0.5	13-02-25 06:24:19.159	13-02-25 10:24:19.158
1	13-02-25 06:24:19.159	13-02-25 14:24:19.158
1.5	13-02-25 06:24:19.159	13-02-25 18:24:19.158
2	13-02-25 06:24:19.159	13-02-25 22:24:19.158

Table 5.11: *Propagation start and end times for different numbers of orbital revolutions simulated, for the Spirale scenario.*

The propagator and integrator settings for the Spirale scenario have been analysed for the maximum number of revolutions, $N_{orbit} = 2$, such that both the integration error and the model error (compared to the highest fidelity model available in Tudat) induced by the propagation are kept below $1m$. The full analysis is documented in Appendix C, and the final settings are repeated here for the high-fidelity and low-fidelity dynamics in Table 5.12 and Table 5.13.

Acceleration type	
Central body gravity (Earth)	100 x 100 (GOCO05c)
3rd body perturbations (point mass)	Sun, Moon, Mars, Jupiter
Solar radiation pressure	C_R = 1.5 (Cannonball)
Atmospheric drag	C_D = 2.0 (US76)
Relativistic Correction	Sun (Schwarzschild)
Environment models	
Central body rotation model	Constant rotation rate determined at t_0
3rd body position model (Moon, Mars, Jupiter)	Keplerian 2-body orbit determined at t_0
3rd body position model (Sun)	Retrieved every epoch from Spice
Propagator settings	
Integrator	variable rkdp87 (abs. tolerance: 2E-3)
Propagator	Cowell

Table 5.12: *Propagation settings used for the high-fidelity dynamics of the Spirale scenario*

Acceleration type	
Central body gravity (Earth)	2 x 2 (GOCO05c)
3rd body perturbations (point mass)	Sun, Moon
Atmospheric drag	C_D = 2.0 (US76)
Environment models	
Central body rotation model	Constant rotation rate determined at t_0
3rd body position model (Moon)	Keplerian 2-body orbit determined at t_0
3rd body position model (Sun)	Retrieved every epoch from Spice
Propagator settings	
Integrator	variable rkdp87 (abs. tolerance: 2E-3)
Propagator	Cowell

Table 5.13: *Propagation settings used for the low-fidelity dynamics of the Spirale scenario*

5.5. Meteosat scenario

Figure 5.9 shows an illustration of the orbit of the METEOSAT 12 satellite, on which the Meteosat test case scenario is based.

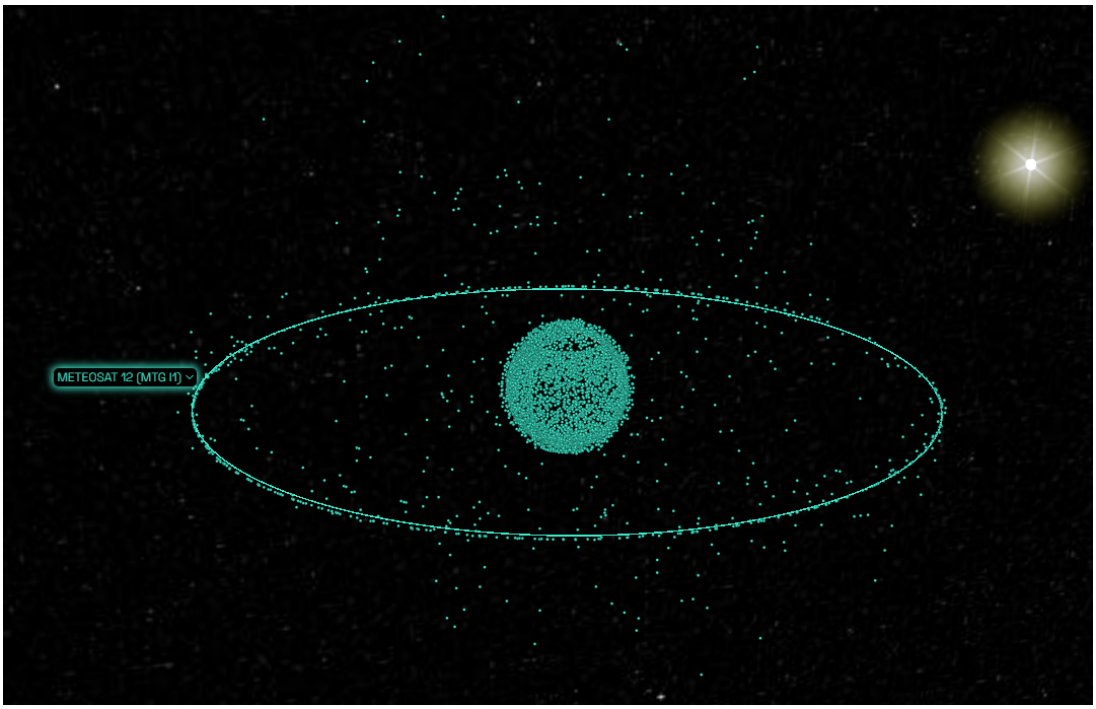


Figure 5.9: *Illustration of the orbit of the METEOSAT 12 satellite, at 18-02-2025 20:00:00 UTC via Wayfinder [115].*

Based on the available data of the satellite's state vectors and covariance matrices, the initial state of the test case scenario is selected at t = 12-02-25 10:51:32.136 UTC, resulting in an initial state vector close to the orbit's perigee. The initial uncertainty is characterised by a Gaussian distribution with the following values:

$$\bar{\boldsymbol{x}} = \begin{bmatrix} 2.421E+7 & -3.450E+7 & -5.241E+5 & 2.517E+3 & 1.767E+3 & -9.749E+0 \end{bmatrix} \begin{bmatrix} m & m/s \end{bmatrix}$$

$$\boldsymbol{P} = \begin{bmatrix} 9.866E+3 & -5.327E-1 & -9.708E-3 & 4.211E+0 & -2.893E-2 & -3.945E-4 \\ -5.327E-1 & 9.863E+3 & -1.238E-2 & -2.776E-2 & 4.272E+0 & 9.126E-4 \\ -9.708E-3 & -1.238E-2 & 9.865E+3 & -3.848E-4 & 9.239E-4 & 4.195E+0 \\ 4.211E+0 & -2.776E-2 & -3.848E-4 & 1.866E-2 & 7.944E-4 & 1.125E-5 \\ -2.893E-2 & 4.272E+0 & 9.239E-4 & 7.944E-4 & 1.751E-2 & -2.127E-5 \\ -3.945E-4 & 9.126E-4 & 4.195E+0 & 1.125E-5 & -2.127E-5 & 1.913E-2 \end{bmatrix} \begin{bmatrix} m^2 & \frac{m^2}{s} \\ \frac{m^2}{s} & \frac{m^2}{2^2} \end{bmatrix}$$

A single propagated orbit, starting from this initial state, is shown in Figure 5.10. The duration of this propagation is set to $T = 86172.0$ seconds, resulting in t_f = 13-02-25 10:47:44.135 UTC. For a simulation lasting N_{orbit} orbital revolutions, the propagation end time is determined by $t_f = t_0 + N_{orbit} \cdot T$, also shown in Table 5.14 for the various number of revolutions chosen to simulate.

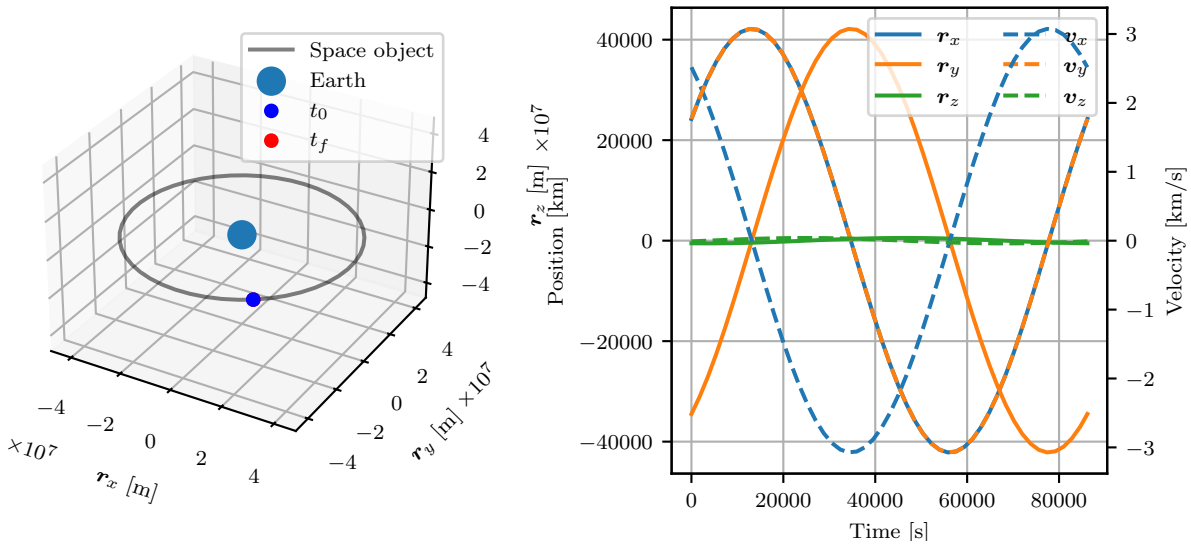


Figure 5.10: *A single propagated orbit of the Meteosat test case scenario.*

N_{orbit} [−]	t_0 [UTC]	t_f [UTC]
1	12-02-25 10:51:32.136	13-02-25 10:47:44.135
2	12-02-25 10:51:32.136	14-02-25 10:43:56.135

Table 5.14: *Propagation start and end times for different numbers of orbital revolutions simulated, for the Meteosat scenario.*

The propagator and integrator settings for the Meteosat scenario have been analysed for the maximum number of revolutions, $N_{orbit} = 2$, such that both the integration error and the model error (compared to the highest fidelity model available in Tudat) induced by the propagation are kept below $1m$. The full analysis is documented in Appendix C, and the final settings are repeated here for the high-fidelity and low-fidelity dynamics in Table 5.15 and Table 5.16.

Acceleration type	
Central body gravity (Earth)	100 x 100 (GOCO05c)
3rd body perturbations (point mass)	Sun, Moon
Solar radiation pressure	C_R = 1.5 (Cannonball)
Relativistic Correction	Sun (Schwarzschild)
Environment models	
Central body rotation model	Constant rotation rate determined at t_0
3rd body position model (Moon)	Keplerian 2-body orbit determined at t_0
3rd body position model (Sun)	Retrieved every epoch from Spice
Propagator settings	
Integrator	fixed rkf8 (time step: 1800s)
Propagator	Cowell

Table 5.15: *Propagation settings used for the high-fidelity dynamics of the Meteosat scenario*

Acceleration type	
Central body gravity (Earth)	2 x 2 (GOCO05c)
Environment models	
Central body rotation model	Constant rotation rate determined at t_0
Propagator settings	
Integrator	fixed rkf8 (time step: 1800s)
Propagator	Cowell

Table 5.16: *Propagation settings used for the low-fidelity dynamics of the Meteosat scenario*

6

Uncertainty Propagation Analysis Results

For each of the scenarios and the number of orbital revolutions described in Chapter 5, the initial uncertainty is propagated using the different UP methods described in Chapter 3, with various different settings. Using each method, a final probability distribution is obtained, which is compared to the distribution obtained using MC with 10^5 samples, to obtain an accuracy metric for each set of method and settings. Each uncertainty propagation is also timed from start to finish, to use as the computation time metric in the evaluation. For each scenario, the UP methods are all run simultaneously on the DelftBlue supercomputer [118], to minimise the effect of varying factors on the run time of the uncertainty propagations.

The various settings used for each UP method are summarised in Table 6.1. For methods with multiple settings, the values for the different settings are cross-matched, resulting in 21 MF, 14 PCE, 15 GMM_UT, 30 GMM_MF and 30 AEGIS propagations, per scenario per number of revolutions. For AEGIS, a range of ϵ_{ent} values is determined for each scenario and number of revolutions separately, based on the relative difference in entropy of the covariances calculated at multiple epochs along a single UT and LinCov propagation. A custom range of values is selected that ensures a wide range of resulting number of splits in the propagations.

Method	**Setting parameter**	**Values used**
LinCov	N.A.	N.A.
UT	N.A.	N.A.
MF	$N_{samples}$	$\{10^3, 5\cdot10^3, 10^4\}$
	ϵ_{lf}	{1, 5, 10, 50, 100, 500, 1000} $[m]$
PCEs (regression)	P_e	{2, 3}
	N/P	{0.9, 1, 2}
PCEs (projection)	P_e	{1, 2}
	Q	{2, 3, 4}
GMM_MF	Split size	{(3,1), (5,1), (3, 3), (5, 5), (5, 3)}
	Split direction	$\{(V_{max}, V_{max}), (\hat{\boldsymbol{a}}_{r,r}, \hat{\boldsymbol{a}}_{r,v}), (\hat{\boldsymbol{a}}_{v,v}, \hat{\boldsymbol{a}}_{v,r})\}$
	ϵ_{lf}	{1, 10} $[m]$
GMM_UT	Split size	{(3,1), (5,1), (3, 3), (5, 5), (5, 3)}
	Split direction	$\{(V_{max}, V_{max}), (\hat{\boldsymbol{a}}_{r,r}, \hat{\boldsymbol{a}}_{r,v}), (\hat{\boldsymbol{a}}_{v,v}, \hat{\boldsymbol{a}}_{v,r})\}$
AEGIS	Split size	{3, 5}
	ϵ_{ent}	Custom range of 15 values

Table 6.1: *The various values used for the different settings for each UP method.*

For the splitting directions for GMMs in Table 6.1, V_{max} refers to the eigenvector that corresponds to the maximum eigenvalue of the uncertainty's covariance matrix. The other directions represent specific vectors in the Cartesian 3-dimensional space, where the first subscript indicates whether the split is applied to the position or the velocity uncertainty, and the second subscript references the vector, according to:

$$\hat{\boldsymbol{a}}_{r,r} = \begin{bmatrix} \boldsymbol{r}_x \\ \boldsymbol{r}_y \\ \boldsymbol{r}_z \\ 0 \\ 0 \\ 0 \end{bmatrix} / \left|\overline{\boldsymbol{r}}\right| \quad \hat{\boldsymbol{a}}_{r,v} = \begin{bmatrix} \boldsymbol{v}_x \\ \boldsymbol{v}_y \\ \boldsymbol{v}_z \\ 0 \\ 0 \\ 0 \end{bmatrix} / \left|\overline{\boldsymbol{v}}\right| \quad \hat{\boldsymbol{a}}_{v,r} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ \boldsymbol{r}_x \\ \boldsymbol{r}_y \\ \boldsymbol{r}_z \end{bmatrix} / \left|\overline{\boldsymbol{r}}\right| \quad \hat{\boldsymbol{a}}_{v,v} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ \boldsymbol{v}_x \\ \boldsymbol{v}_y \\ \boldsymbol{v}_z \end{bmatrix} / \left|\overline{\boldsymbol{v}}\right| \tag{6.1}$$

Note that, unlike the eigenvector V_{max}, which is a 6-dimensional vector, these splitting directions are originally 3-dimensional, and are extended with zeros to be applied to a 6-dimensional covariance matrix. This means that in the other 3 dimensions, for which the splitting direction vector contains zeros, the uncertainty is not split and in these directions all new components have the same variances and covariances as in the original covariance matrix.

The splitting directions are chosen based on the nonlinearity metric $NL(\hat{\boldsymbol{a}})$ defined in Equation 3.33. The tested splitting directions are chosen as the unit vectors of the ECI and RSW coordinate frames, applied to both the position and velocity [119]:

$$\begin{bmatrix} \hat{\boldsymbol{a}}_{r,x} \\ \hat{\boldsymbol{a}}_{r,y} \\ \hat{\boldsymbol{a}}_{r,z} \\ \hat{\boldsymbol{a}}_{v,x} \\ \hat{\boldsymbol{a}}_{v,y} \\ \hat{\boldsymbol{a}}_{v,z} \\ \hat{\boldsymbol{a}}_{r,r} \\ \hat{\boldsymbol{a}}_{r,v} \\ \hat{\boldsymbol{a}}_{r,h} \\ \hat{\boldsymbol{a}}_{v,r} \\ \hat{\boldsymbol{a}}_{v,v} \\ \hat{\boldsymbol{a}}_{v,h} \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ \overline{\boldsymbol{r}}_x & \overline{\boldsymbol{r}}_y & \overline{\boldsymbol{r}}_z & 0 & 0 & 0 \\ \overline{\boldsymbol{v}}_x & \overline{\boldsymbol{v}}_y & \overline{\boldsymbol{v}}_z & 0 & 0 & 0 \\ \overline{\boldsymbol{h}}_x & \overline{\boldsymbol{h}}_y & \overline{\boldsymbol{h}}_z & 0 & 0 & 0 \\ 0 & 0 & 0 & \overline{\boldsymbol{r}}_x & \overline{\boldsymbol{r}}_y & \overline{\boldsymbol{r}}_z \\ 0 & 0 & 0 & \overline{\boldsymbol{v}}_x & \overline{\boldsymbol{v}}_y & \overline{\boldsymbol{v}}_z \\ 0 & 0 & 0 & \overline{\boldsymbol{h}}_x & \overline{\boldsymbol{h}}_y & \overline{\boldsymbol{h}}_z \end{bmatrix} / \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ |\overline{\boldsymbol{r}}| \\ |\overline{\boldsymbol{v}}| \\ |\overline{\boldsymbol{h}}| \\ |\overline{\boldsymbol{r}}| \\ |\overline{\boldsymbol{v}}| \\ |\overline{\boldsymbol{h}}| \end{bmatrix} \tag{6.2}$$

where the division by the second matrix is applied uniformly per row, such that all vectors are normalised to unit vectors. The directions with the highest $NL(\hat{\boldsymbol{a}})$ are the $\hat{\boldsymbol{a}}_{r,r}$ and $\hat{\boldsymbol{a}}_{v,v}$, which are chosen as the first splitting direction. For GMM settings with two splits at t_0, the second splitting direction is chosen to complement the first direction, such that the position or velocity is split along both the $\boldsymbol{r}$ and the $\boldsymbol{v}$ direction.

To compare the accuracy and computational efficiency of the various uncertainty propagation methods, the NL_2 distance is used to define the accuracy of the propagated uncertainties at t_f, where a value of 0 represents a perfect agreement, and a value of 1 indicates that the two distributions are completely different. The NL_2 distance is calculated between each propagated uncertainty using the various methods and settings, and the propagated uncertainty using 10^5 MC samples. For uncertainties that consist of a set of sample points, such as when using MC samples, but also with MF or PCEs, a GMM is fitted on the sample points, consisting of 9 components, as described in Section 3.8, which is used to calculate the NL_2 distance.

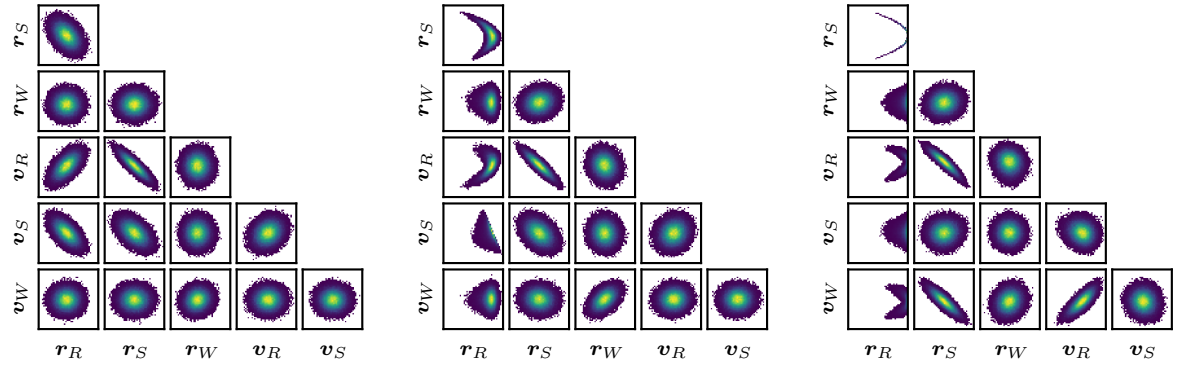
By showing the NL_2 distance of the propagated uncertainty versus the computation time of the propagation, the trade-off between accuracy and computationally efficiency can be analysed for the different methods. An optimal uncertainty propagation method shows a low NL_2 distance, as well as a low computation time. In the following sections, only a selection of figures is shown to demonstrate the most important results, leading to an asynchronous discussion between the different scenarios. For a complete set of figures showing the NL_2 distance versus the computation time of the propagation methods, for all scenarios and number of revolutions, the reader is referred to Appendix E.

The following section discusses the effects of increasing propagation times in Section 6.1, followed by analyses of the performance of the different uncertainty propagation methods in Sections 6.2 to 6.6. Afterwards, Section 6.7 discusses the HEO cases in more detail. Finally, the chapter concludes with an overview of the most optimal uncertainty propagation methods in Section 6.8

6.1. Influence of Propagation Time

The PDF of the propagated uncertainty using 10^5 MC samples is shown in Figure 6.1 for increasing propagation times of the Velox scenario. These distributions are used as the true uncertainty and are used to calculate the NL_2 distances for each propagated uncertainty. Therefore, it is important to know how these uncertainties are distributed before analysing the accuracy of other propagation methods. From Figure 6.1 it can be seen that for a LEO orbit, the uncertainty becomes increasingly non-Gaussian after a higher number of orbital revolutions. For long propagation times, the PDF turns into a banana shape, positioned along the along track (S) direction with its curvature towards the negative radial (R) direction, that becomes thinner as the propagation time extends.

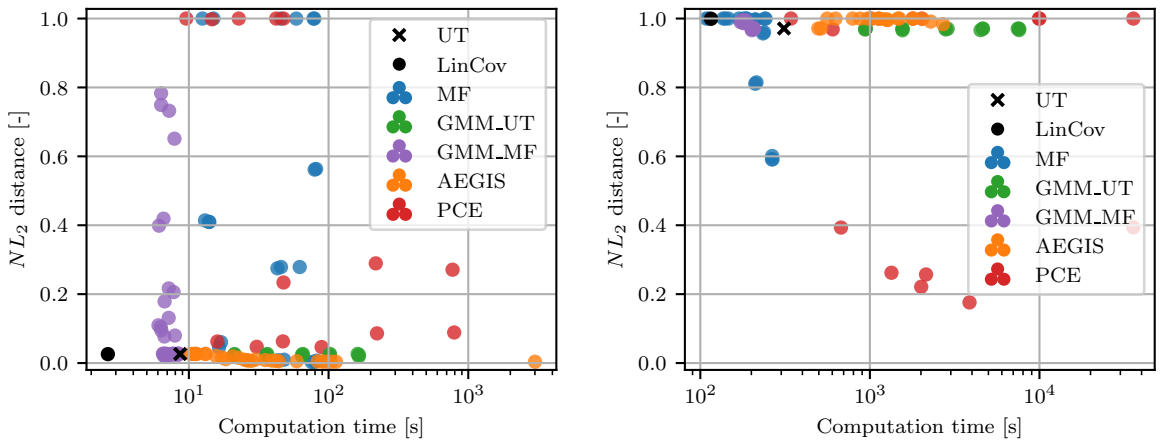
This banana-like shape is caused by a difference in the phase along the orbit of the various sample points. The different state vectors cause a change in the orbital period of each sample. However, since all samples are propagated with the same propagation time (corresponding to a full revolution of the mean orbit), a difference in orbital periods results in a difference in the phase of the sample points at t_f. This difference in phase accumulates after multiple orbital revolutions, where for long propagation times this difference in state vectors becomes significantly larger than the initial deviation in state vectors at t_0, thus turning the PDF into a thin banana shape along the orbit trajectory.



(a) *1 revolution.* **(b)** *10 revolutions.* **(c)** *50 revolutions.*

Figure 6.1: *Correlations between each of the 6 dimensions in the RSW coordinate frame, shown with 10^5 propagated MC samples for the Velox scenario after 1, 10 and 50 revolutions.*

Figure 6.2 shows the accuracy and computational efficiency of the various methods for a short and long propagation time of the Oneweb scenario. In this figure, the NL_2 distance, which is a normalised metric ranging from 0 (indicating no overlap with the MC samples) to 1 (indicating identical distributions), represents the accuracy of the propagated uncertainties, where lower values indicate higher accuracies. The results are shown for the different uncertainty propagation methods with various settings, and are grouped per method.
In Figure 6.2 it can be seen that for a short propagation time, i.e. after 1 orbital revolution, all methods are able to approximate the distribution relatively well, with NL_2 values tending towards 0. Only for some settings of MF (including GMM_MF) and PCEs, the solutions have not converged and the resulting NL_2 distances are larger, with some even at 1.0. For the higher number of revolutions, the trend is the opposite, where most results are clustered near an NL_2 distance of 1, and only the NL_2 distances of MF and PCEs reach towards 0 depending on the settings used. This behaviour is expected, since for longer propagation times the final uncertainty becomes non-Gaussian, which is more difficult to approximate, especially for methods that assume Gaussian distributions.



(a) *1 revolution.* **(b)** *50 revolutions.*

Figure 6.2: *NL_2 distance compared against MC samples, versus the computation time for different UP methods with various settings, for the Velox scenario after 1 and 50 revolutions.*

6.2. Linearised Covariance and the Unscented Transform

Both LinCov and the UT are methods that assume a final Gaussian distribution, and their approximations of the probability distribution are also Gaussian. These methods show a steep rise in the NL_2 distance as the propagation time increases, shown in Figure 6.3. For longer propagation times, the true PDF (approximated by MC) becomes non-Gaussian and the approximations of LinCov and the UT start to deviate more. For 50 revolutions of the Velox scenario, shown in Figure 6.2, LinCov has almost no overlap with the MC samples, where UT performs slightly better.

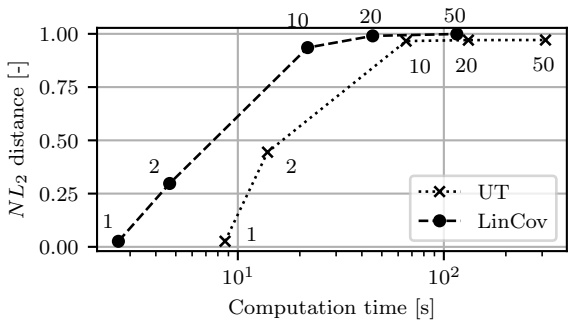
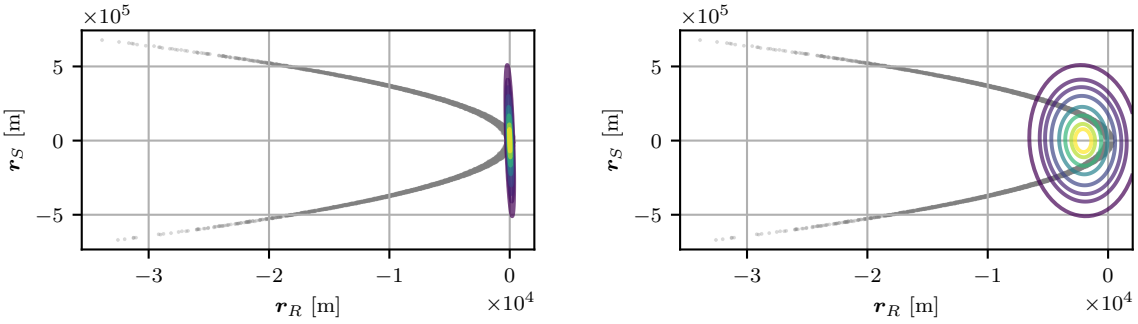


Figure 6.3: *The NL_2 distance and computation time of the LinCov and UT for different numbers of revolutions of the Velox scenario.*

This difference can also be seen by comparing the final position uncertainty in the R-S frame of both LinCov and the UT with the MC samples, shown in Figure 6.4. The LinCov method captures the size of the spread at the mean quite well, with a very 'thin' shape, while the UT covers more area of the distribution's tails with a more spread out approximation, which is slightly favoured by the calculation of the NL_2 distance metric.



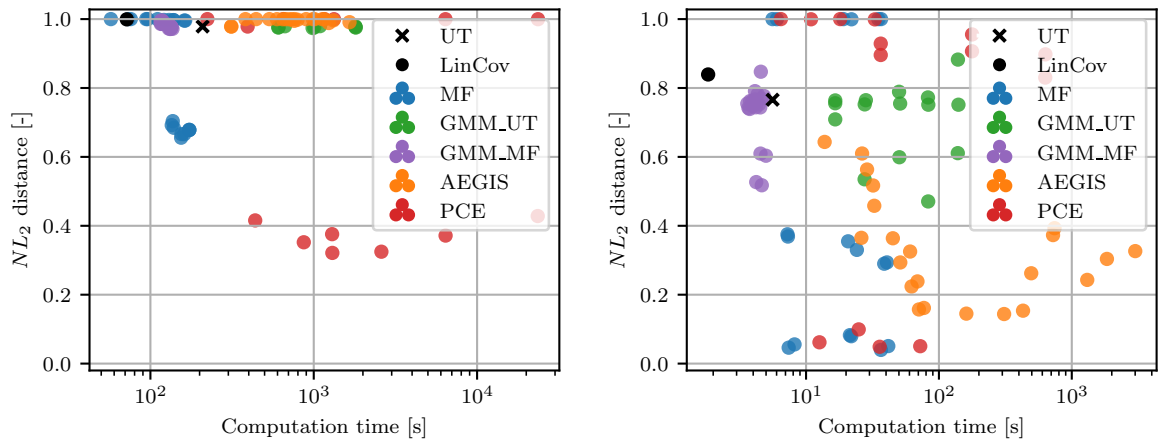
(a) *Linearised Covariance.*

(b) *Unscented Transform.*

Figure 6.4: *The position uncertainty in the Radial and Along-track frame compared against MC samples, after 50 revolutions of the Velox scenario, for the LinCov and UT method.*

6.3. Polynomial Chaos Expansions

PCEs are able to approximate the final uncertainty with high accuracy, even for non-Gaussian cases. Figure 6.5 shows the high performance of PCEs in a case where most methods are not able to approximate the non-Gaussian uncertainty, as well as in a case where multiple methods perform well. In both cases, PCEs stand out as the best scoring method on the NL_2 distance, with some settings reaching towards 0 even for long propagation times, though at the cost of computation time. Similar results can be found for almost all other scenarios, documented in Appendix E.

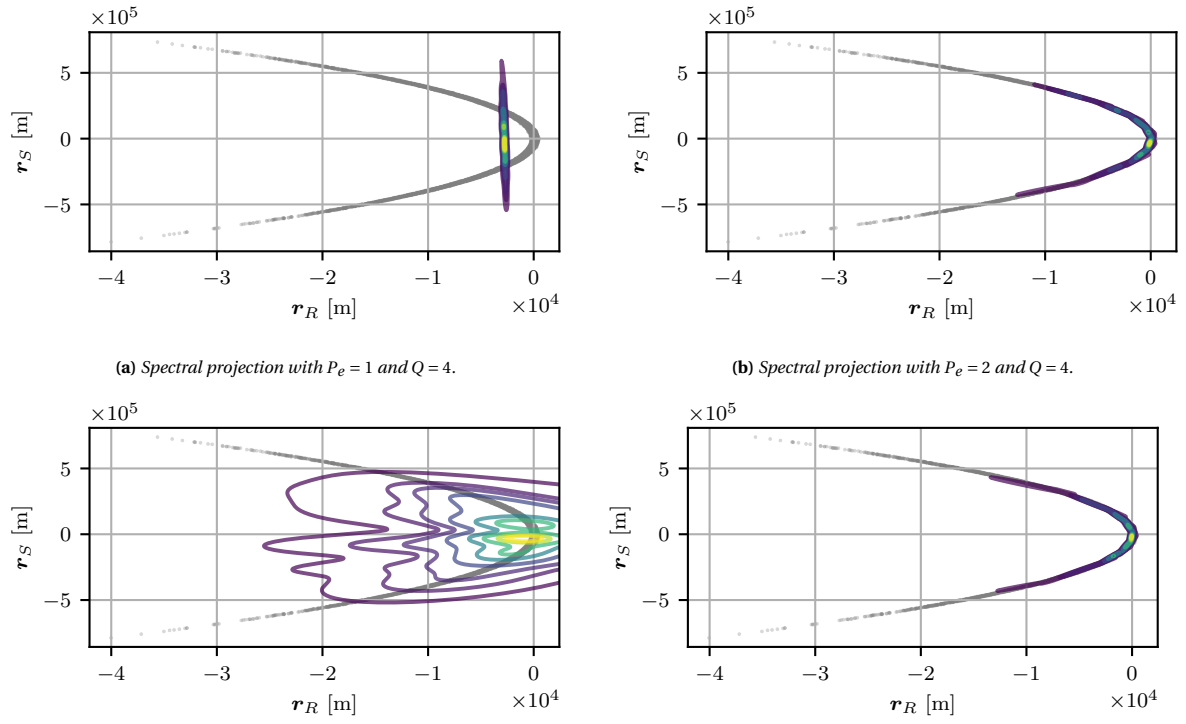


(a) *Oneweb at 50 revolutions.*

(b) *Meteosat at 2 revolutions.*

Figure 6.5: *NL_2 distance compared against MC samples, versus the computation time for different UP methods with various settings, for two different scenarios after the maximum number of orbital revolutions.*

There are some settings for PCEs, which perform very poorly on the NL_2 distance, while also taking long computation times. These are the settings for which the expansion can not approximate the distribution, or for which the least squares regression does not converge. Figure 6.6 shows this behaviour for the non-Gaussian final distribution of Oneweb after 50 orbits, where the $P_e = 1$ expansion can not approximate a non-Gaussian distribution, and the $N/P = 0.9$ regression is not converged yet, while their counterparts with $P_e = 2$ and $N/P = 1$ respectively provide very similar results to MC.



(a) *Spectral projection with $P_e = 1$ and $Q = 4$.*

(b) *Spectral projection with $P_e = 2$ and $Q = 4$.*

(c) *Least Squares Regression with $P_e = 2$ and $N/P = 0.9$.*

(d) *Least Squares Regression with $P_e = 2$ and $N/P = 1$.*

Figure 6.6: *The position uncertainty in the Radial and Along-track frame compared against MC samples, after 50 revolutions of the Oneweb scenario, for different settings of the PCEs.*

6.4. Multi-Fidelity

In Figure 6.5, shown in the previous section, it can be seen that the MF method also performs well for both cases. While it scores slightly worse on the NL_2 metric compared to PCEs, it takes less computation time to do so. For almost all cases, especially with long propagation times, MF can provide better results than LinCov or UT in terms of the NL_2 distance. At the same time, in most cases the MF method also has shorter computation times than PCEs, even for the settings that reach the very low NL_2 distances. Figure 6.7 clearly shows this difference in computation time for comparable NL_2 distances. The reader is reminded that the x-axis uses a log scale, thus the MF method typically has a computation time that is an order of magnitude smaller than that of PCEs. It can be seen that the difference in computation time also increases when the propagation time becomes longer. For the most optimal settings of both methods, MF requires the propagation of fewer sample points with HF dynamics than PCEs. When the propagation time becomes long, the propagation of samples becomes the most important factor in the computation time, leading to the increased difference in computational efficiency.

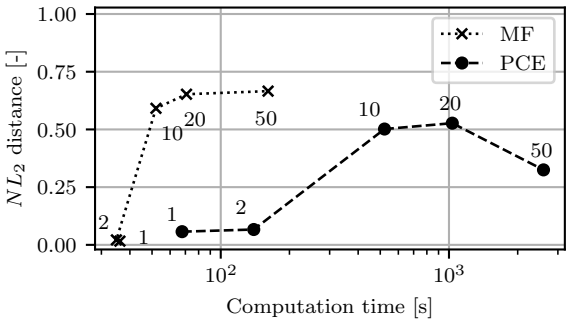
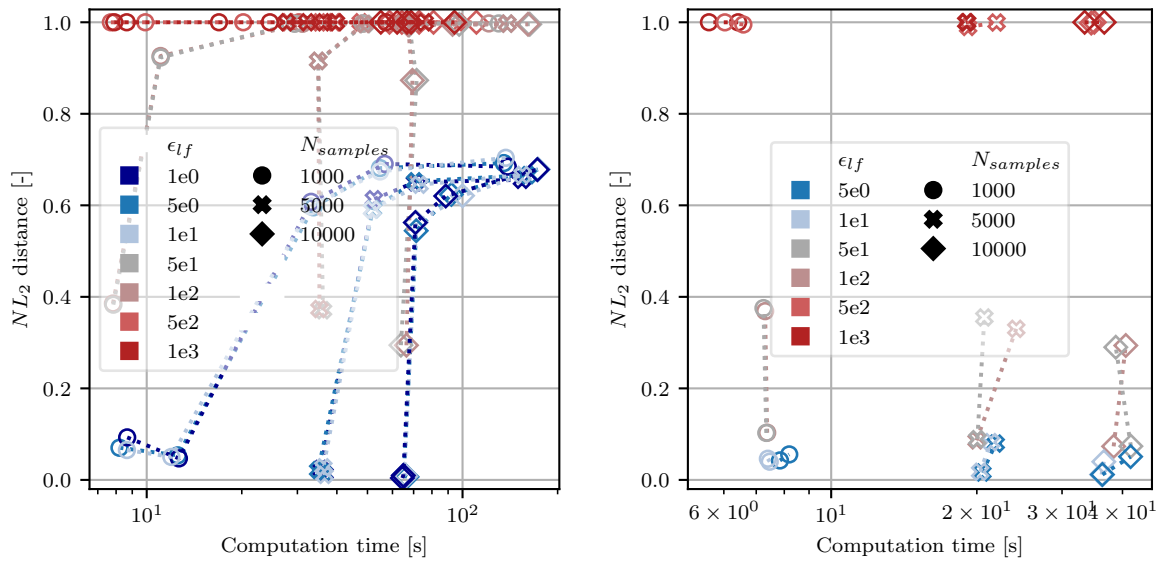


Figure 6.7: *The NL_2 distance and computation time of MF with $N_{samples} = 5000$ and $\epsilon_{lf} = 5m$, and a PCE with $P_e = 3$ and $N/P = 2$, for different numbers of revolutions of the Oneweb scenario.*

The two main setting parameters of MF that have been tuned in this analysis are: the low-fidelity tolerance ϵ_{lf} and the total number of sample points propagated $N_{samples}$. Since the rank of the MF method is increased until the ϵ_{lf} is satisfied, a lower value for ϵ_{lf} generally results in a higher number of important samples used. The effects of using different settings on the computation time and NL_2 distance can be seen when plotting the NL_2 distance against the computation time again, but this time only for the MF method, and grouping the results based on the settings used. In Figure 6.8, it is seen that most values of ϵ_{lf} result in an NL_2 distance near 1. This is due to there not being enough important samples to accurately map the relation to all samples in the distribution. However, for smaller values of ϵ_{lf}, the NL_2 distance starts around 0 for short propagation times, and ends around 0.7 for long propagation times, indicating that with more important samples, the approximation of the final uncertainty improves.

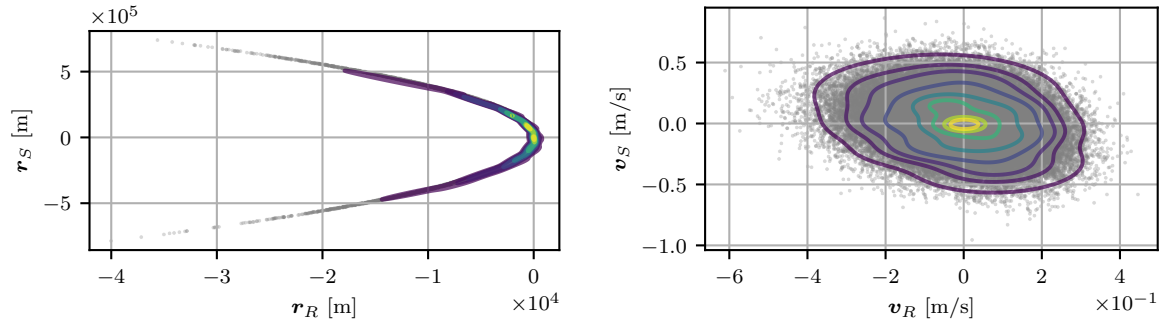
In Figure 6.8 it can be also be seen that for a larger $N_{samples}$, the computation time also increases, and the NL_2 distance slightly decreases. Although a higher $N_{samples}$ leads to slightly lower NL_2 distances, adjusting the ϵ_{lf} leads to much bigger changes in the NL_2 distance, with smaller differences in the computation time. It should be noted that in this analysis not all settings of every method are explored, thus this NL_2 distance of 0.6 is not necessarily the limit of the MF method. For instance, tuning the complexity and accuracy of the HF and LF models could provide propagations with shorter computation times or lower NL_2 distances.



(a) *Oneweb.* **(b)** *Meteosat.*

Figure 6.8: *The NL_2 distance and computation time of MF with various settings, where the various data points along the plotted lines correspond to different numbers of revolutions, for the Oneweb and Meteosat scenario.*

Figure 6.9 shows the distribution of one of the optimal settings, meaning a low ϵ_{lf} and a high $N_{samples}$, which results in a NL_2 distance of 0.7 for the Oneweb scenario after 50 revolutions. From the combination of Figure 6.8a and Figure 6.9, which both concern the same scenario, it is clear that even though the NL_2 distance is not close to 0, the shape of the MC samples is still very well approximated by the MF propagation.

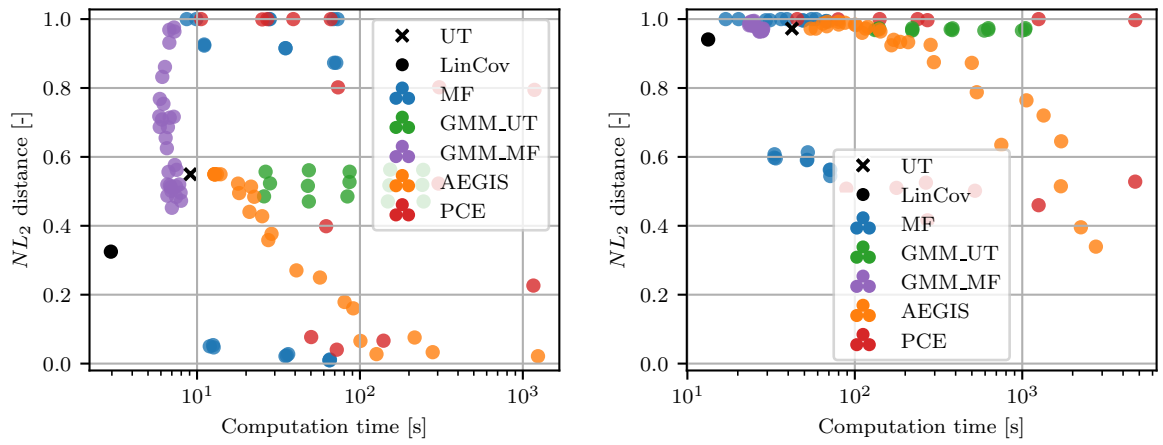


(a) *Position uncertainty.* **(b)** *Velocity uncertainty.*

Figure 6.9: *The position and velocity uncertainty in the Radial and Along-track frame compared against MC samples, after 50 revolutions of the Oneweb scenario, for the MF method with $\epsilon_{lf} = 1$ and $N_{samples} = 10000$.*

6.5. Gaussian Mixture Models

In almost all scenarios, the GMMs (both with UT and with MF) perform similar to or worse than LinCov in terms of NL_2 distance, while having much longer computation times. Figure 6.10a demonstrates that the GMMs already fall behind when the LinCov can still approximate the MC propagated uncertainty with a low NL_2 distance. Additionally, Figure 6.10b demonstrates that when the uncertainty becomes non-Gaussian, the GMMs can still not perform any better than the LinCov, and also have NL_2 distances of near 1.



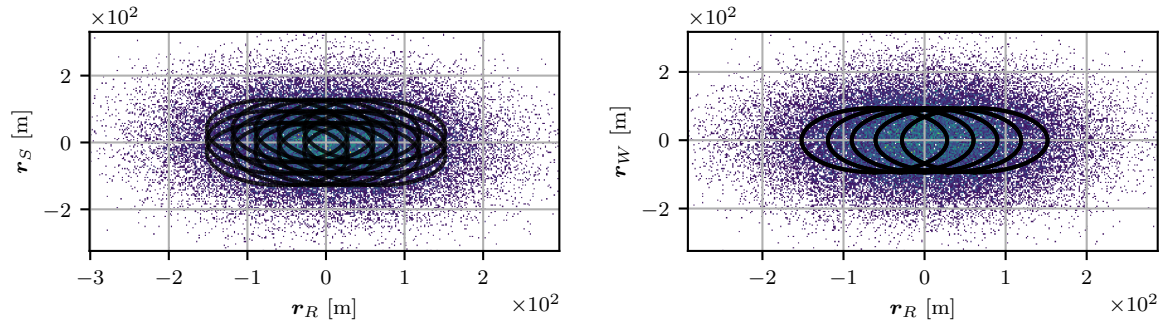
(a) *Oneweb at 2 revolutions.*

(b) *Oneweb at 10 revolutions.*

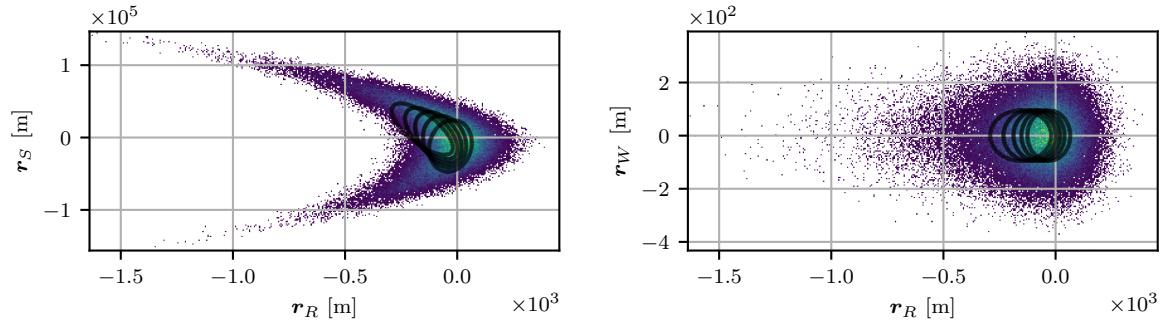
Figure 6.10: *NL_2 distance compared against MC samples, versus the computation time for different UP methods with various settings, for the Oneweb scenario after 2 and 10 orbital revolutions.*

The idea of the GMMs is that by splitting the initial Gaussian distribution into multiple smaller components, these components will remain Gaussian for a longer time and by combining the smaller Gaussian components at the final propagation time, the non-Gaussian shape of the PDF can be approximated. The method is expected to outperform LinCov, especially for long propagation times where the final distribution is non-Gaussian, but fails to approximate the shapes of these PDFs itself, shown in Figure 6.11b.

From Figure 6.11a it is clear that the components are initially spread out, and do approximate the initial distribution well in the R-S frame of the position. However, in Figure 6.11b it seems that the individual components all end up very close to each other in the final propagated uncertainty, resulting in a near Gaussian distribution again that does not include the tails of the banana-shaped distributions that arise from MC samples.



(a) *At t_0.*

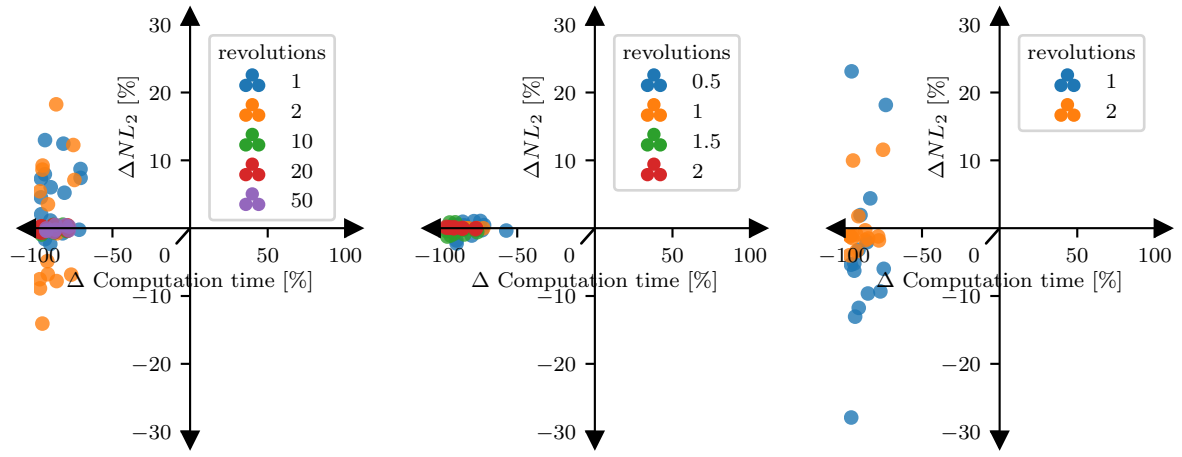


(b) *At 10 revolutions.*

Figure 6.11: *The components of the GMM created by splitting in the $\hat{\boldsymbol{a}}_{r,r}$ and $\hat{\boldsymbol{a}}_{r,v}$ directions with a split size of 5 and 3 respectively, compared against MC samples at different time epochs for the Oneweb scenario.*

This effect of ending up with a centred collection of GMM components can be due to the way that the individual components are propagated, using the UT (or in the case of GMM_MF, the MF method is applied to these same sample points generated by the UT). It is possible that by propagating the individual GMM components using a different method, such as LinCov, different results could be obtained. Similarly, using a larger splitting library could lead to better results in approximating the tails of the MC uncertainty. It should also be noted that the entire propagation, as well as the splitting of components, occurs in the Cartesian coordinate frame, in which the nonlinearity deforms the uncertainty into a non-Gaussian distribution in multiple directions, as shown in Figure 6.1 in Section 6.1. Thus, by splitting/propagating in a different coordinate frame, such as the modified equinoctial elements, better results can be expected [30].

In all of the scenarios, the uncertainties propagated using GMM_MF have much shorter computation times than those of GMM_UT, while for most scenarios the GMM_MF propagations with a low ϵ_{lf} achieve approximately the same levels of NL_2 distance as the GMM_UT method. Examples of this can be found in the previously shown Figures 6.2 and 6.10. To provide a more direct comparison between the methods, only the GMM_MF runs with the smaller value for ϵ_{lf} are taken and are compared one by one directly against the GMM_UT run with the same settings for split size and splitting direction. The results are presented as relative differences in percentage, shown in Figure 6.12. A clear pattern emerges where the propagations using GMM_MF decrease the computation time by ≈ 80% compared to GMM_UT, while the NL_2 distance only changes by less than 20% and does not favour any of the two methods.

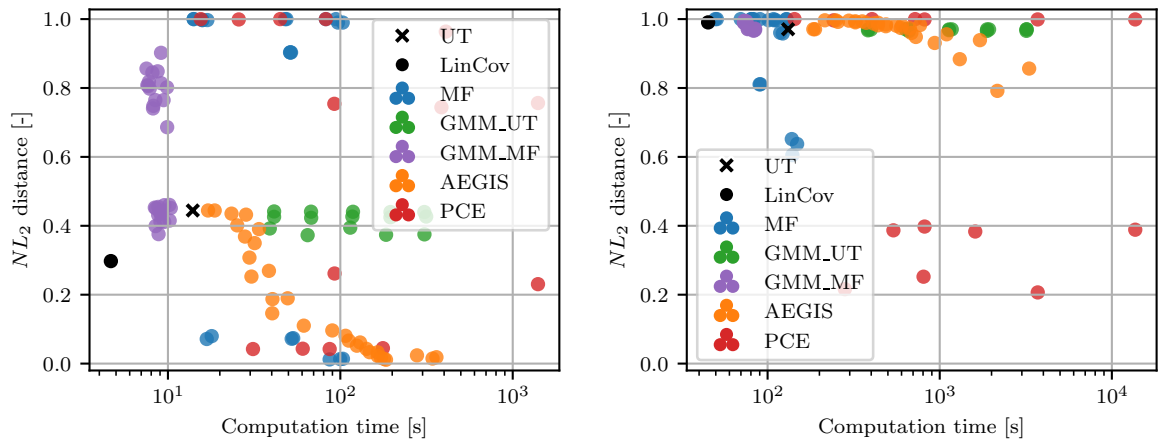


(a) *Oneweb.* **(b)** *Spirale.* **(c)** *Meteosat.*

Figure 6.12: *The relative difference of GMM_MF propagations compared to GMM_UT propagations in the NL_2 distance and computation time, for different scenarios and different number of orbital revolutions.*

6.6. Adaptive Entropy-based Gaussian Information Synthesis

The AEGIS method shows a clear trend between increasing computation times and smaller NL_2 distances in most scenarios, as seen in Figure 6.13. This is in line with the expectation that the final uncertainty is approximated better when the uncertainty is split more often, which increases the number of components to propagate and thus also the computation time. However, in all scenarios AEGIS falls short to the other methods in either the NL_2 distance, the computation time, or both. Unsurprisingly, since it combines the LinCov and UT at each propagation step, while also increasing the number of components after splitting, it has higher computation times than both of these methods. The smallest NL_2 values reached by the AEGIS method without causing out-of-memory errors are comparable to that of PCEs in only a few scenarios and number of revolutions. However, in these cases, the PCEs require less computation time and AEGIS does not provide any benefits. Compared to MF, for almost all propagations using the AEGIS method, there are propagations done with MF that provide equal or better results in both the NL_2 distance and computation time. The exceptions here can be attributed to a lack of variety in MF data points, but still show that the results from MF provide a more optimal Pareto front than those of AEGIS.



(a) *Velox at 2 revolutions.*

(b) *Velox at 20 revolutions.*

Figure 6.13: *NL_2 distance compared against MC samples, versus the computation time for different UP methods with various settings, for the Velox scenario after 2 and 20 orbital revolutions.*

The settings that cause AEGIS to propagate a single component, thus when the change in entropy during the full propagation time does not reach the selected tolerance level, essentially results in the UT, but with a small increase in computation time. This increase in computation time is shown in Figure 6.14, indicated with red arrows between the UT and a single component AEGIS propagation. The addition in computation time has two causing factors that are related to the differences between the UT and AEGIS.

The UT propagates all sigma points separately from t_0 to t_f, and can apply different time steps during the integration for each sigma point, as long as the state vectors at t_f are determined for all sigma points. Conversely, AEGIS requires the calculation of the change in differential entropy at each time step to determine whether the propagation should be stopped, to split the component. This calculation requires the propagation of the STM, and an extra number of computations performed at each time step. Additionally, it requires that all sigma points are propagated simultaneously, in order to combine them into a covariance matrix and determine the nonlinear entropy. When using a variable step integrator, this simultaneous integration of all sigma points leads to a minimum time step being applied uniformly.

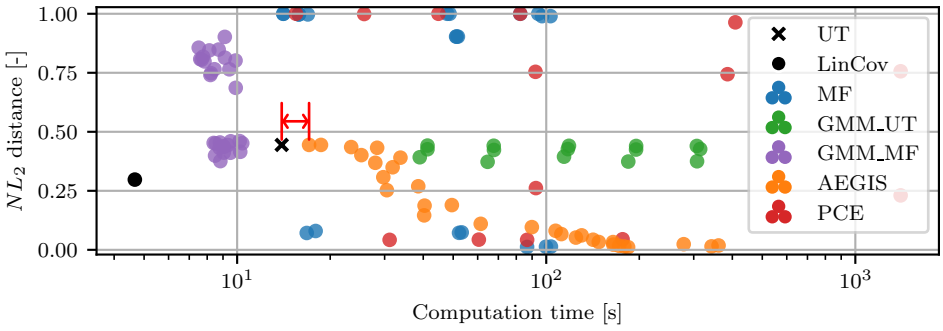


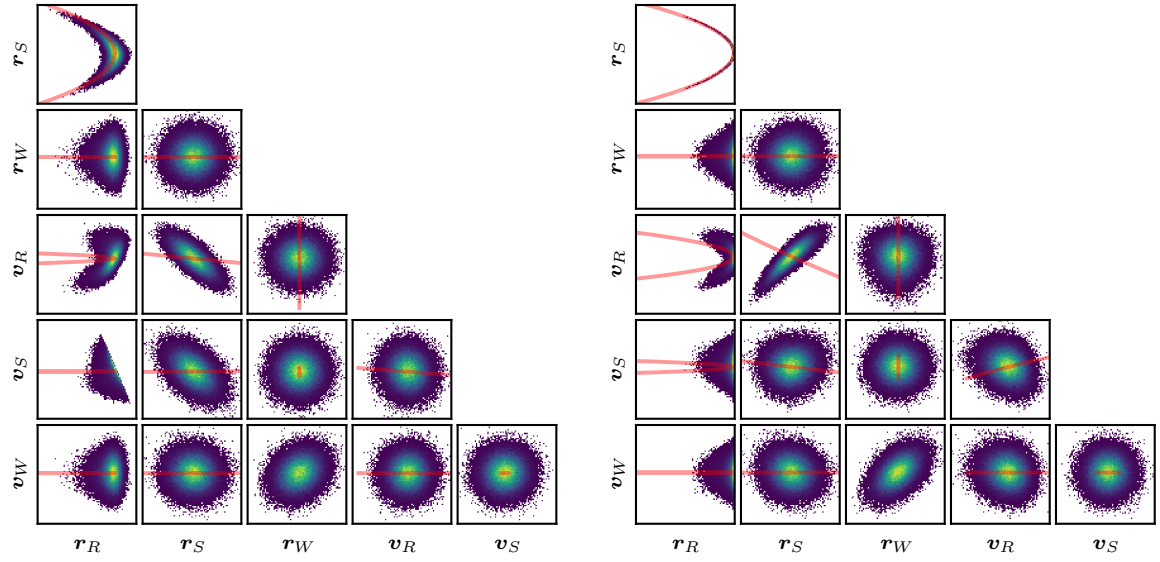
Figure 6.14: *NL_2 distance compared against MC samples, versus the computation time for different UP methods with various settings, for the Velox scenario after 2 revolutions. The difference in computation time between the UT and a single component AEGIS propagation is highlighted with red arrows.*

Figure 6.13 also shows the effect of long propagation times on the NL_2 distance after propagating with AEGIS. For 2 revolutions, the NL_2 distance decreases significantly when the computation time increases, and some settings even reach an NL_2 distance close to 0. However, for 20 revolutions, the NL_2 distance is close to 1 for all settings of AEGIS, and only goes to 0.8 for very long computation times. A possible explanation for this is related to the results of the GMMs. In Figure 6.11 it is seen that the components of a GMM at t_0 end up towards the centre of the distribution at t_f. For short propagation times, the AEGIS method can still approximate the final distribution well, since the difference between the AEGIS method and GMMs here is that AEGIS can keep splitting the components repeatedly, and thus also later on in the propagation, depending on the value for

ϵ_{ent}. However, for long propagation times, due to the exponential increase in the number of components due to splitting, the number of splits required to keep splitting each component at short intervals leads to unreachable computation times. A smaller number of splits here will increase the interval between splitting components, which increases the propagation time of each component and thus also increase the deviation between the Gaussian approximation and the true PDF (which becomes non-Gaussian over time) of each component. It is expected that for long propagation times, a large number of splits can still reach low NL_2 values along the trend of increasing the computation times for decreasing NL_2 values, however, this results in unreasonably long computation times, and would not be more efficient than a propagation using MC samples.

6.7. Elliptical Orbits

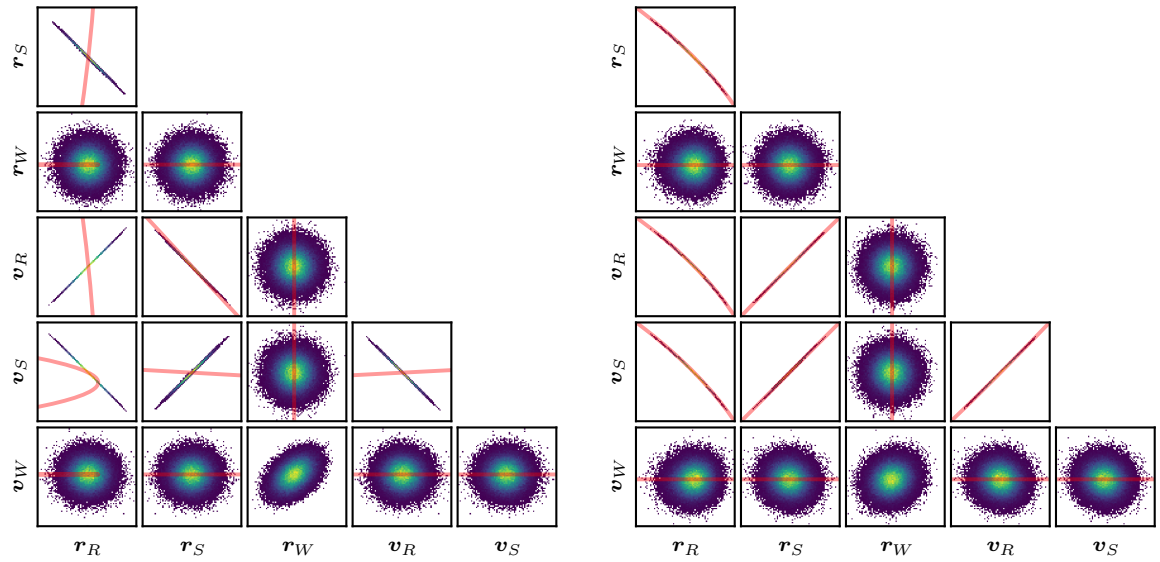
In Figure 6.15 the correlations between the 6 dimensions in the RSW frame are shown for the LEO orbits of the Oneweb scenario after 10 and 50 revolutions, with the state vector's components along the mean orbital trajectory shown on top of the graphs in red. Note that the scales of the axes are determined by the spread of the uncertainty, thus the orbits are shown with deformed proportions, for the sake of a direct comparison with the probability distribution. With this view, it can be seen that the banana-shaped probability distribution in the R-S frame is mostly formed by differences in the phase of the various sample points along similar orbits. This is in line with the phenomenon that when the uncertainty is propagated in Keplerian elements, only the true anomaly θ changes significantly, which also represents the phase along the orbit, while the other elements are relatively inert.



(a) *10 revolutions.* **(b)** *50 revolutions.*

Figure 6.15: *Correlations between each of the 6 dimensions in the RSW coordinate frame, shown with* 10^5 *propagated MC samples for the Oneweb scenario after 10 and 50 revolutions, with the components of the state vector in each dimension shown in red in the same frames.*

In Figure 6.16, the same type of figures are shown for one of the HEO scenarios, where it can be seen in Figure 6.16b that after a full number of revolutions, near periapsis again, the uncertainty is also distributed along the orbital trajectory, in a non-Gaussian shape. Note that the banana shape is present in Figure 6.16b, though more difficult to see due to the diagonal alignment along the R-S frame, which is likely caused by the mean state vector being slightly away from the periapsis, combined with the elliptical shape of the orbit. However, after 1.5 orbital revolutions shown in Figure 6.16a, thus being near apoapsis, the probability distribution deviates significantly from the path of the orbit trajectory.



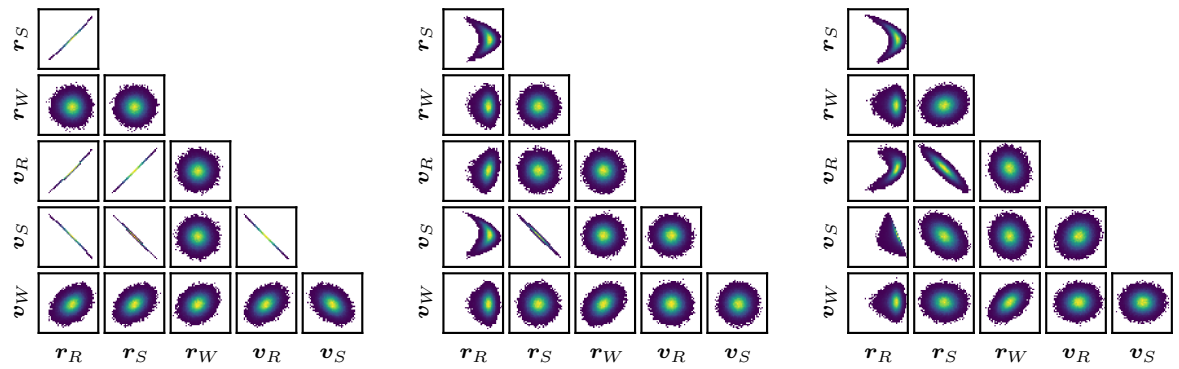
(a) *After 1.5 revolutions (at apoapsis).*

(b) *After 2 full revolutions (at periapsis).*

Figure 6.16: *Correlations between each of the 6 dimensions in the RSW coordinate frame, shown with* 10^5 *propagated MC samples for the Spirale scenario after 1.5 and 2 revolutions, with the components of the state vector in each dimension shown in red in the same frames.*

By measuring the uncertainty at a different point along the orbit, the difference in state vectors between MC samples is not only caused by a difference in phase, but also by the different orbit trajectories of each sample. After a full number of revolutions, samples with similar position vectors but different velocity vectors at t_0, have similar position vectors again at t_f (disregarding perturbations). However, this does not apply to non-integer number of revolutions, since different velocity vectors change the paths of the rest of the orbit. It is suspected that the highly elliptical shape of the orbit magnifies the effect that a different velocity vector at periapsis has on the state vector at apoapsis. This can cause the large spread of the uncertainty along a different direction than the orbit trajectory seen in Figure 6.16a.

In a similar way, the highly elliptical shape can increase the effect that a change in the state vector at periapsis has on the orbital period of a sample, which results in larger differences in the phase of samples. This can explain why the HEO scenarios show a non-Gaussian distribution even after a single orbital revolution, while for the other scenarios, this takes much longer, demonstrated in Figure 6.17. This is even the case for the GEO orbit of Meteosat, which has a larger orbital period than both HEO scenarios.



(a) *Cosmos at 1 revolution.*

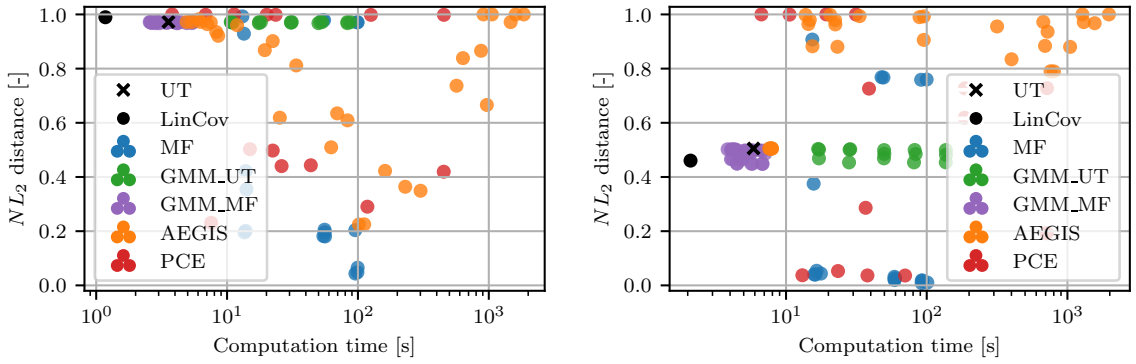
(b) *Meteosat at 2 revolutions.*

(c) *Velox at 10 revolutions.*

Figure 6.17: *Correlations between each of the 6 dimensions in the RSW coordinate frame, shown with* 10^5 *propagated MC samples for different scenarios, after specific numbers of revolutions.*

For both HEO scenarios, after 1 or 2 revolutions, the NL_2 distance of most results are near 1.0, except for some settings of MF, PCEs and AEGIS, shown in Figure 6.18a. This is due to the uncertainty being non-Gaussian al-

ready after a single orbital revolution, thus the methods and combinations that can only approximate Gaussian distributions do not score well on the NL_2 metric. A surprising result can be seen in Figure 6.18b, where the propagation time of 1.5 revolutions is longer than that of 1 revolution in Figure 6.18a, yet the NL_2 distances of most methods and settings are lower. Both LinCov and the UT approximate the MC samples relatively well after 1.5 revolutions with $NL_2 \approx 0.5$, suggesting that the spread of the uncertainty at apoapsis deviates less from a Gaussian probability distribution than the uncertainty at periapsis.



(a) *At 1 revolution (periapsis).*

(b) *At 1.5 revolutions (apoapsis).*

Figure 6.18: *NL_2 distance compared against MC samples, versus the computation time for different UP methods with various settings, for the Spirale scenario after 1 and 1.5 orbital revolutions.*

In Figure 6.18b it is also seen that the AEGIS method still performs poorly. Only the settings for AEGIS which result in the lowest computation times have the same NL_2 distance as the UT, and all other settings have higher computation times and higher NL_2 distances. This implies that splitting the uncertainty into multiple GMM components (while splitting in the direction corresponding to the largest eigenvalue) only results in worse approximations of the true uncertainty. Finally, for both integer and non integer numbers of revolutions for the HEO scenarios, the performances of MF and PCEs are the same as the results for the circular orbits discussed above, where both methods reach the lowest NL_2 distances, especially for non-Gaussian uncertainties. A small difference is that for the HEO scenarios, MF and PCEs have roughly equal computation times, whereas for the LEO and GEO scenarios, for most settings the MF method has smaller computation times than PCEs.

6.8. Optimal Uncertainty Propagation Methods

Looking at the trade-off between short computation times and small NL_2 distances, the methods that stand out are LinCov, MF and PCEs. These three methods provide the best options to use, depending on the desire for low computation times or high accuracy estimations, or a combination of both.

LinCov is the fastest method of all implemented methods, and scores well on the NL_2 distance when the final uncertainty is Gaussian. However, the method fails to approximate non-Gaussian uncertainties, and thus loses its effectiveness for long propagation times or highly elliptical orbits. While the method can not approximate the curvature of the non-Gaussian uncertainties in the R-S frame, it does capture the spread along the along-track (S) direction, as well as the thickness of the distribution shape in the R-S frame.

On the other hand, PCEs are able to approximate all uncertainties accurately, performing very well on the NL_2 distance metric, especially for non-Gaussian uncertainties. However, PCEs have significantly longer computation times than LinCov, typically 1-2 orders of magnitude larger, and also have longer computation times than most other methods, except MC.

Finally, MF provides an excellent alternative to the shortcomings of both methods. It scores well on the NL_2 distance, also for non-Gaussian uncertainties, and has shorter computation times than PCEs. The MF method also contains various setting parameters that influence both the computation time and the accuracy of the estimation. A lower LF tolerance level leads to the propagation of more important points, which increases the accuracy of the estimation while also increasing the computation times. This can be used to obtain an uncertainty propagation that satisfies the desired trade-off between a high accuracy and low computation times. The combination between low computation times and high accuracy estimations, with the potential to adjust setting parameters to increase the accuracy or shorten the computation time, makes MF a viable and potentially useful UP method for conjunction assessments.

7

Conjunction Analysis Test Setup

Two different satellite conjunction scenarios have been set up, to analyse the capabilities of the MF method when used to propagate the uncertainties for a conjunction analysis. The scenarios include a high relative velocity conjunction, as well as a low relative velocity conjunction, based on real conjunctions predicted in Privateer Wayfinder [115]. Due to the high density of space objects in LEO, and the goal to simulate realistic conjunctions, both scenarios contain two LEO satellites.

In Wayfinder, the mean states and covariance matrices of both satellites are given at the TCA. To simulate a proper conjunction analysis scenario, the mean states and covariance matrices of both objects at a start time t_0 are required, where t_0 is at a chosen distance of Δt_{prop} seconds before t_{TCA}. This Δt_{prop} determines the length of the uncertainty propagations, and represents how far ahead a conjunction event would be analysed. In this study, a Δt_{prop} of both $48h$ and $96h$ is chosen, further referenced as the 'short' and 'long' versions, to demonstrate the effects of a long uncertainty propagation on the methods used.

The mean states at t_0 are obtained by back propagating from the mean states at t_{TCA} using Tudat, where the required accuracy of the propagation is determined by comparing the backwards and forwards propagation of the mean state, described in Appendix C. To obtain a realistic covariance matrix, the ephemerides from each of the objects in the conjunction are taken from Wayfinder, and a covariance matrix is retrieved from the reported epoch $t_{0,cov}$ that is closest to $t_{TCA} - \Delta t_{prop}$. The true t_0 is then taken to be the midpoint between $t_{0,cov}$ of both objects, and used to back-propagate the mean states. Note that due to the limited availability of the reported covariance matrices of the space objects, this final t_0 differs from $t_{TCA} - \Delta t_{prop}$.

Since Wayfinder uses an unknown method to propagate the covariance, in which the size of the uncertainty does not seem to grow in time, taking a covariance matrix from Wayfinder at t_0 and propagating it forward with the implemented UP methods from Chapter 3 will result in a larger uncertainty at t_{TCA} than intended with the conjunction scenario. This effect is demonstrated in Figure 7.1, where the covariance matrix taken from Wayfinder at t_0 is propagated forward using the UT, and compared against the reported covariance matrices at later epochs. The volume of the covariance matrix is calculated as the product of the eigenvalues. To mitigate this effect, the covariance matrices found for t_0 are scaled by a factor of 10^{-4} (resulting in a reduction of the standard deviations of 10^{-2}), for both objects in each scenario. The full covariance matrices used for each case are documented in the scenario descriptions, in Sections 7.1 and 7.2 below.

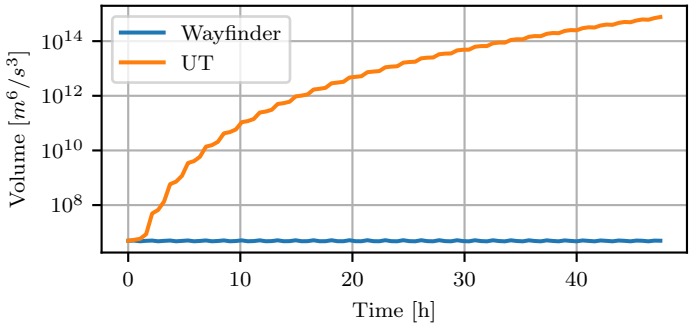


Figure 7.1: *Comparison of the volume of the covariance matrix over time of the UT with the reported covariance matrices by Wayfinder*

Additionally, to properly evaluate the accuracies of the conjunction analyses, a scenario with a significant probability of collision is desired. Thus for all scenarios, to increase the probability of collisions, the mean separation between the objects at t_{TCA} is decreased to $2.5m$ by shifting the second object's position vector towards

the first object, after which the new mean states are back propagated to t_0. The new mean states at t_{TCA} are calculated according to:

$$\overline{\boldsymbol{r}}_{S,new} = \overline{\boldsymbol{r}}_{S,old} + \left(\overline{\boldsymbol{r}}_{P,old} - \overline{\boldsymbol{r}}_{S,old}\right)\left(1 - \frac{\Delta r_{new}}{\left|\overline{\boldsymbol{r}}_{P,old} - \overline{\boldsymbol{r}}_{S,old}\right|}\right) \tag{7.1}$$

$$\overline{\boldsymbol{v}}_{S,new} = \overline{\boldsymbol{v}}_{S,old} \tag{7.2}$$

$$\overline{\boldsymbol{x}}_{P,new} = \overline{\boldsymbol{x}}_{P,old} \tag{7.3}$$

Where $\overline{\boldsymbol{x}}_P = \left[\overline{\boldsymbol{r}}_P, \overline{\boldsymbol{v}}_P\right]$ and $\overline{\boldsymbol{x}}_P = \left[\overline{\boldsymbol{r}}_P, \overline{\boldsymbol{v}}_P\right]$ are the mean state of the primary and secondary object, $_{old}$ and $_{new}$ indicate the vectors before and after the transformation respectively, and Δr_{new} is the desired mean separation at t_{TCA}.

Finally, to obtain a low relative velocity scenario with a relative velocity of approximately $100 m/s$, the velocity components in the mean vectors at t_{TCA} are artificially adjusted, starting from two vectors with a slightly higher relative velocity. The new mean states are calculated according to:

$$\overline{\boldsymbol{v}}_{S,new} = v_{ratio}\overline{\boldsymbol{v}}_S + (1 - v_{ratio})\,\overline{\boldsymbol{v}}_P \tag{7.4}$$

$$\overline{\boldsymbol{r}}_{S,new} = \overline{\boldsymbol{r}}_{S,old} \tag{7.5}$$

$$\overline{\boldsymbol{x}}_{P,new} = \overline{\boldsymbol{x}}_{P,old} \tag{7.6}$$

Where $v_{ratio} = \Delta v_{new}/\Delta v_{old}$ is the ratio between the desired relative velocity and the old relative velocity.

7.1. High Relative Velocity Scenario

The first scenario is a conjunction between the UniSat B and PNE04 satellite. An illustration of the orbits of both objects around t_{TCA} is shown in Figure 7.2. The chosen time windows of the short and long version for this scenario start at 2025-04-28 04:51:31.814503 UTC and 2025-04-26 09:55:31.814484 UTC respectively.

Figure 7.2: *Illustration of the conjunction between the UniSat B and the PNE04 satellite at 2025-04-30 04:57:13.256749 UTC via Wayfinder [115].*

The Gaussian distributions of the objects' uncertainties at t_0 for the short version of this scenario are given by:

$$\bar{\boldsymbol{x}}_P = \begin{bmatrix} -4.857E+5 & 6.918E+6 & -1.136E+6 & 1.088E+3 & -1.132E+3 & -7.349E+3 \end{bmatrix} \begin{bmatrix} m & m/s \end{bmatrix}$$
$$\bar{\boldsymbol{x}}_S = \begin{bmatrix} -4.857E+5 & 6.918E+6 & -1.136E+6 & 1.088E+3 & -1.132E+3 & -7.349E+3 \end{bmatrix} \begin{bmatrix} m & m/s \end{bmatrix}$$

$$\boldsymbol{P}_P = \begin{bmatrix} 8.816E-1 & -1.209E-2 & -1.326E-2 & 2.138E-4 & -1.486E-4 & -1.238E-5 \\ -1.209E-2 & 9.190E-1 & 6.274E-2 & -9.724E-5 & 1.131E-3 & 8.406E-5 \\ -1.326E-2 & 6.274E-2 & 9.348E-1 & -4.793E-5 & 4.692E-4 & 2.323E-4 \\ 2.138E-4 & -9.724E-5 & -4.793E-5 & 5.489E-6 & 4.456E-7 & 3.721E-7 \\ -1.486E-4 & 1.131E-3 & 4.692E-4 & 4.456E-7 & 2.634E-6 & -1.311E-6 \\ -1.238E-5 & 8.406E-5 & 2.323E-4 & 3.721E-7 & -1.311E-6 & 3.710E-6 \end{bmatrix} \begin{bmatrix} m^2 & \frac{m^2}{s} \\ \frac{m^2}{s} & \frac{m^2}{2^2} \end{bmatrix}$$
$$\boldsymbol{P}_S = \begin{bmatrix} 9.389E-1 & 2.321E-2 & 5.871E-2 & 1.161E-3 & 2.163E-4 & -1.115E-4 \\ 2.321E-2 & 8.856E-1 & 1.877E-2 & 2.697E-4 & 2.629E-4 & -3.322E-5 \\ 5.871E-2 & 1.877E-2 & 9.095E-1 & 2.667E-4 & 5.838E-5 & 1.575E-4 \\ 1.161E-3 & 2.697E-4 & 2.667E-4 & 2.360E-6 & -9.179E-7 & -9.519E-7 \\ 2.163E-4 & 2.629E-4 & 5.838E-5 & -9.179E-7 & 5.316E-6 & -4.287E-7 \\ -1.115E-4 & -3.322E-5 & 1.575E-4 & -9.519E-7 & -4.287E-7 & 4.212E-6 \end{bmatrix} \begin{bmatrix} m^2 & \frac{m^2}{s} \\ \frac{m^2}{s} & \frac{m^2}{2^2} \end{bmatrix}$$

Similarly for the long version of this scenario:

$$\bar{\boldsymbol{x}}_P = \begin{bmatrix} 5.615E+5 & -6.863E+6 & -1.031E+6 & -9.198E+2 & -1.179E+3 & 7.436E+3 \end{bmatrix} \begin{bmatrix} m & m/s \end{bmatrix}$$
$$\bar{\boldsymbol{x}}_S = \begin{bmatrix} 5.615E+5 & -6.863E+6 & -1.031E+6 & -9.198E+2 & -1.179E+3 & 7.436E+3 \end{bmatrix} \begin{bmatrix} m & m/s \end{bmatrix}$$

$$\boldsymbol{P}_P = \begin{bmatrix} 9.999E-1 & -9.133E-6 & -1.876E-5 & -5.440E-4 & -1.806E-4 & -1.637E-4 \\ -9.133E-6 & 9.999E-1 & 2.457E-5 & -1.806E-4 & 8.763E-4 & 6.776E-4 \\ -1.876E-5 & 2.457E-5 & 1.000E+0 & -1.637E-4 & 6.778E-4 & 3.339E-4 \\ -5.440E-4 & -1.806E-4 & -1.637E-4 & 1.057E-5 & 6.179E-7 & 9.604E-7 \\ -1.806E-4 & 8.763E-4 & 6.778E-4 & 6.179E-7 & 3.578E-6 & -1.306E-6 \\ -1.637E-4 & 6.776E-4 & 3.339E-4 & 9.604E-7 & -1.306E-6 & 4.158E-6 \end{bmatrix} \begin{bmatrix} m^2 & \frac{m^2}{s} \\ \frac{m^2}{s} & \frac{m^2}{2^2} \end{bmatrix}$$
$$\boldsymbol{P}_S = \begin{bmatrix} 1.000E+0 & 2.944E-5 & 2.156E-5 & 1.288E-3 & 3.820E-4 & -6.868E-5 \\ 2.944E-5 & 9.999E-1 & 1.759E-5 & 3.820E-4 & -4.977E-4 & 4.983E-5 \\ 2.156E-5 & 1.759E-5 & 9.999E-1 & -6.851E-5 & 4.987E-5 & -1.314E-4 \\ 1.288E-3 & 3.820E-4 & -6.851E-5 & 2.768E-6 & -1.687E-6 & 4.610E-9 \\ 3.820E-4 & -4.977E-4 & 4.987E-5 & -1.687E-6 & 1.037E-5 & -7.826E-7 \\ -6.868E-5 & 4.983E-5 & -1.314E-4 & 4.610E-9 & -7.826E-7 & 5.301E-6 \end{bmatrix} \begin{bmatrix} m^2 & \frac{m^2}{s} \\ \frac{m^2}{s} & \frac{m^2}{2^2} \end{bmatrix}$$

The settings and acceleration models used for the propagations with high-fidelity dynamics have been analysed for the long version of this scenario, such that both the integration error and the model error (compared to the highest fidelity model available in Tudat) induced by the propagation are kept below $1m$. The full analysis is documented in Appendix C, and the final settings are repeated here in Table 7.1.

Acceleration type	
Central body gravity (Earth)	100 x 100 (GOCO05c)
3rd body perturbations (point mass)	Sun, Moon
Solar radiation pressure	C_R = 1.5 (Cannonball)
Atmospheric drag	C_D = 2.0 (US76)
Environment models	
Central body rotation model	Constant rotation rate determined at t_0
3rd body position model (Moon)	Keplerian 2-body orbit determined at t_0
position model (Sun	Retrieved every epoch from Spice
Propagator settings	
Integrator	variable rkdp87 (tolerance: 2E-6)
Propagator	Cowell

Table 7.1: *Propagation settings used for the high-velocity scenario*

The specifications of the satellites are shown in Table 7.2. The mass and cross sectional areas are taken from ESA DISCOS (Database and Information System Characterising Objects in Space)[1]. The HBR is calculated using $HBR = \sqrt{A_P/\pi} + \sqrt{A_S/\pi}$, where A_P and A_S are the average cross sectional area of the primary and secondary object respectively.

Primary object	
Satellite mass	26 kg
Average cross sectional area	0.238 m^2
Secondary object	
Satellite mass	922 kg
Average cross sectional area	12.145 m^2
Joint properties	
HBR	2.24 m

Table 7.2: *Specifications of the satellites in the high-velocity scenario.*

7.2. Low Relative Velocity Scenario

The second scenario is a conjunction between the Starlink-2046 and Starlink-3671 satellite. An illustration of the orbits of both objects around t_{TCA} is shown in Figure 7.3. The chosen time windows of the short and long version for this scenario start at 2025-05-05 07:16:10.814592 UTC and 2025-05-03 08:52:10.814565 UTC respectively.

Figure 7.3: *Illustration of the conjunction between the Starlink-2046 and the Starlink-3671 satellite at 2025-05-07 07:31:00.105278 UTC via Wayfinder [115].*

[1] https://discosweb.esoc.esa.int/

The Gaussian distributions of the objects' uncertainties at t_0 for the short version of this scenario are given by:

$$\bar{\boldsymbol{x}}_P = \begin{bmatrix} 5.158E+6 & -4.295E+6 & -1.714E+6 & 4.142E+3 & 2.684E+3 & 5.763E+3 \end{bmatrix} \begin{bmatrix} m & m/s \end{bmatrix}$$
$$\bar{\boldsymbol{x}}_S = \begin{bmatrix} 5.158E+6 & -4.295E+6 & -1.714E+6 & 4.142E+3 & 2.684E+3 & 5.763E+3 \end{bmatrix} \begin{bmatrix} m & m/s \end{bmatrix}$$

$$\boldsymbol{P}_P = \begin{bmatrix} 8.655E-1 & 1.107E-2 & 5.601E-3 & 3.467E-4 & -1.968E-4 & -1.280E-4 \\ 1.107E-2 & 9.200E-1 & 5.707E-2 & -4.425E-4 & 7.194E-4 & 3.401E-4 \\ 5.601E-3 & 5.707E-2 & 9.473E-1 & -4.144E-4 & 4.968E-4 & 5.009E-4 \\ 3.467E-4 & -4.425E-4 & -4.144E-4 & 4.844E-6 & 5.930E-7 & 4.468E-8 \\ -1.968E-4 & 7.194E-4 & 4.968E-4 & 5.930E-7 & 3.866E-6 & -1.962E-6 \\ -1.280E-4 & 3.401E-4 & 5.009E-4 & 4.468E-8 & -1.962E-6 & 3.614E-6 \end{bmatrix} \begin{bmatrix} m^2 & \frac{m^2}{s} \\ \frac{m^2}{s} & \frac{m^2}{2^2} \end{bmatrix}$$
$$\boldsymbol{P}_S = \begin{bmatrix} 8.643E-1 & 7.488E-3 & 1.936E-3 & 3.892E-4 & -2.179E-4 & -1.497E-4 \\ 7.488E-3 & 9.211E-1 & 5.729E-2 & -4.635E-4 & 6.763E-4 & 3.314E-4 \\ 1.936E-3 & 5.729E-2 & 9.474E-1 & -4.457E-4 & 4.691E-4 & 5.014E-4 \\ 3.892E-4 & -4.635E-4 & -4.457E-4 & 4.763E-6 & 6.500E-7 & 1.711E-7 \\ -2.179E-4 & 6.763E-4 & 4.691E-4 & 6.500E-7 & 3.947E-6 & -1.955E-6 \\ -1.497E-4 & 3.314E-4 & 5.014E-4 & 1.711E-7 & -1.955E-6 & 3.611E-6 \end{bmatrix} \begin{bmatrix} m^2 & \frac{m^2}{s} \\ \frac{m^2}{s} & \frac{m^2}{2^2} \end{bmatrix}$$

Similarly for the long version of this scenario:

$$\bar{\boldsymbol{x}}_P = \begin{bmatrix} 1.022E+6 & -4.511E+6 & -5.160E+6 & 7.204E+3 & -8.683E+2 & 2.188E+3 \end{bmatrix} \begin{bmatrix} m & m/s \end{bmatrix}$$
$$\bar{\boldsymbol{x}}_S = \begin{bmatrix} 1.022E+6 & -4.511E+6 & -5.160E+6 & 7.204E+3 & -8.683E+2 & 2.188E+3 \end{bmatrix} \begin{bmatrix} m & m/s \end{bmatrix}$$

$$\boldsymbol{P}_P = \begin{bmatrix} 1.000E+0 & -6.079E-6 & 4.468E-5 & -5.983E-5 & 1.263E-4 & 4.213E-4 \\ -6.079E-6 & 9.999E-1 & 5.371E-5 & 1.262E-4 & 1.570E-5 & 8.754E-4 \\ 4.468E-5 & 5.371E-5 & 9.999E-1 & 4.211E-4 & 8.755E-4 & 6.478E-4 \\ -5.983E-5 & 1.262E-4 & 4.211E-4 & 5.436E-6 & 7.654E-7 & -1.854E-6 \\ 1.263E-4 & 1.570E-5 & 8.755E-4 & 7.654E-7 & 8.177E-6 & -3.948E-6 \\ 4.213E-4 & 8.754E-4 & 6.478E-4 & -1.854E-6 & -3.948E-6 & 5.834E-6 \end{bmatrix} \begin{bmatrix} m^2 & \frac{m^2}{s} \\ \frac{m^2}{s} & \frac{m^2}{2^2} \end{bmatrix}$$
$$\boldsymbol{P}_S = \begin{bmatrix} 1.000E+0 & -1.490E-6 & 4.106E-5 & -7.747E-5 & 1.187E-4 & 3.608E-4 \\ -1.490E-6 & 9.999E-1 & 5.655E-5 & 1.186E-4 & 3.286E-5 & 9.018E-4 \\ 4.106E-5 & 5.655E-5 & 9.999E-1 & 3.607E-4 & 9.018E-4 & 6.485E-4 \\ -7.747E-5 & 1.186E-4 & 3.607E-4 & 5.352E-6 & 5.834E-7 & -1.589E-6 \\ 1.187E-4 & 3.286E-5 & 9.018E-4 & 5.834E-7 & 8.261E-6 & -4.061E-6 \\ 3.608E-4 & 9.018E-4 & 6.485E-4 & -1.589E-6 & -4.061E-6 & 5.830E-6 \end{bmatrix} \begin{bmatrix} m^2 & \frac{m^2}{s} \\ \frac{m^2}{s} & \frac{m^2}{2^2} \end{bmatrix}$$

The settings and acceleration models used for the propagations with high-fidelity dynamics have been analysed for the long version of this scenario, such that both the integration error and the model error (compared to the highest fidelity model available in Tudat) induced by the propagation are kept below $1\,m$. The full analysis is documented in Appendix C, and the final settings are repeated here in Table 7.3.

Acceleration type	
Central body gravity (Earth)	100 x 100 (GOCO05c)
3rd body perturbations (point mass)	Sun, Moon
Solar radiation pressure	C_R = 1.5 (Cannonball)
Atmospheric drag	C_D = 2.0 (US76)
Environment models	
Central body rotation model	Retrieved every epoch from Spice
3rd body position model (Moon)	Keplerian 2-body orbit determined at t_0
3rd body position model (Sun)	Retrieved every epoch from Spice
Propagator settings	
Integrator	variable rkdp87 (tolerance: 2E-6)
Propagator	Cowell

Table 7.3: *Propagation settings used for the low-velocity scenario*

The specifications of the satellites are shown in Table 7.4. The mass and cross sectional areas are taken from ESA DISCOS (Database and Information System Characterising Objects in Space)[2]. The HBR is calculated using $HBR = \sqrt{A_P/\pi} + \sqrt{A_S/\pi}$, where A_P and A_S are the average cross sectional area of the primary and secondary object respectively.

Primary object	
Satellite mass	260 kg
Average cross sectional area	13.562 m^2
Secondary object	
Satellite mass	260 kg
Average cross sectional area	13.562 m^2
Joint properties	
HBR	4.16 m

Table 7.4: *Specifications of the satellites in the low-velocity scenario.*

7.3. Sampling at Time of Closest Approach

To determine the collision probability with the propagated uncertainties of both objects at t_{TCA}, a large number of samples are needed, from which only a small number of pairs will result in a collision. For a lower estimated P_C, a higher number of sample pairs is needed to keep the same error on the value of P_C. The minimum number of samples needed can be determined by [63]:

$$N_{samples} > \frac{4(\mathrm{e}-2)\,(1-P_C)}{P_C \cdot e_{pc}^2} \log\left(\frac{2}{1-\beta}\right) \tag{7.7}$$

where β is the confidence level and e_{pc} is the relative error of the collision probability. Both the high-velocity and low-velocity scenarios have been adjusted such that the final P_C values are in the order of 10^{-4} or higher. However, even though these values are relatively high for a conjunction assessment, they still lead to a high number of $N_{samples}$ needed. Using a confidence level of β =95% and a P_C value of 1E-4, getting a relative error of 0.1 requires approximately 10^7 sample pairs.

With the limited computational power available in this research, propagating all 10^7 samples for both objects from t_0 to t_{TCA} is not feasible, at most 10^5 samples can be propagated for the full propagation time. Therefore, an extra step is introduced at t_{TCA} to generate the required 10^7 sample pairs from the propagated uncertainties, using a conversion of the uncertainty to a Gaussian distribution in MEE, based on the method used in the Brute Force Monte Carlo from-TCA mode in [112].

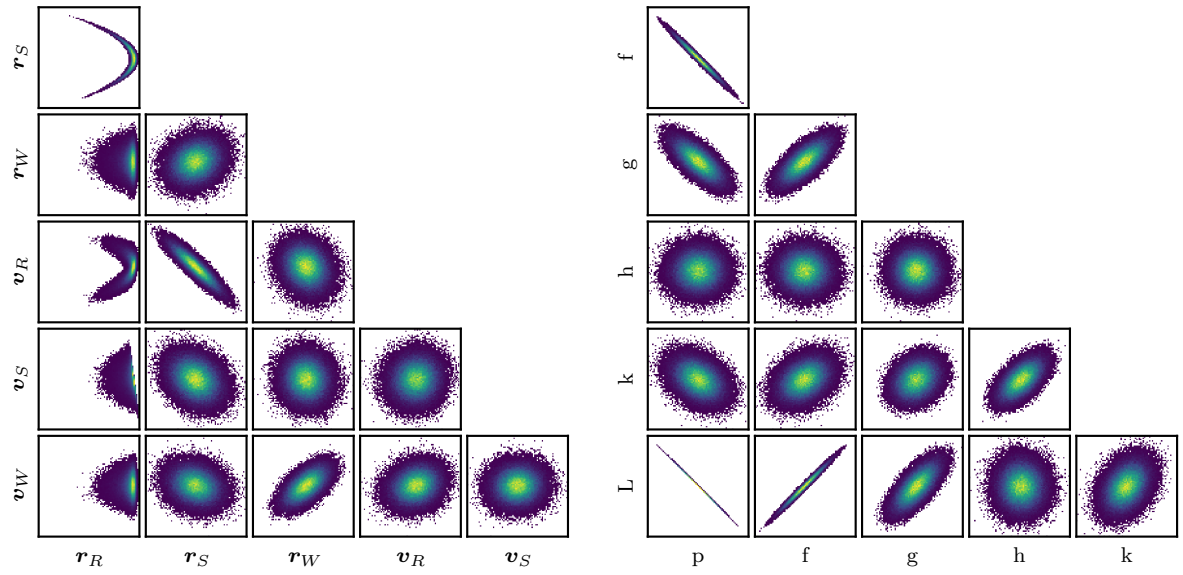
The uncertainties of both objects are first propagated in Cartesian coordinates from t_0 to t_{TCA}, for instance using 10^5 MC samples, or using MF. These samples at t_{TCA} are then converted from Cartesian coordinates to

[2]https://discosweb.esoc.esa.int/

MEE, using the conversions described in Appendix B. From the state vectors in MEE, a Gaussian distribution is created by calculating the mean and covariance, using Equations 4.8 and 4.9 described in Section 4.2. Using the mean vector and covariance matrix of this Gaussian distribution, all 10^7 required samples can be generated in MEE, using Equation 4.10. Finally, these 10^7 generated samples for both objects' uncertainties are converted to Cartesian coordinates using the conversions described in Appendix B, and can be used to calculate P_C.

The conversion to a Gaussian distribution in MEE enables the generation of a large number of samples, that follow the same non-Gaussian distribution in Cartesian coordinates as the smaller number of propagated samples. This is possible when the propagated non-Gaussian uncertainty in Cartesian coordinates follows a (near) Gaussian distribution in MEE. This is the case for long propagation times, since the uncertainty distributes itself along the orbital trajectory, as described in Section 6.7, which corresponds to a Gaussian distribution in the 6th element in MEE: the true longitude L.

The final uncertainty of the Velox scenario after 20 orbital revolutions, is shown in Figure 7.4 with 10^5 propagated MC samples, in Cartesian coordinates as well as in MEE. This shows that even for the highly non-Gaussian distribution in Cartesian coordinates, the uncertainty becomes a Gaussian distribution when all samples are converted to MEE. Additionally, Figure 7.5 shows the original Cartesian MC samples, together with contour lines of the Gaussian distribution created in MEE and converted to Cartesian coordinates. The contour lines of the Gaussian distribution in MEE show no deviation from the original MC samples in Cartesian coordinates.



(a) *In the RSW frame in Cartesian coordinates.*

(b) *In modified equinoctial elements.*

Figure 7.4: *Correlations between each of the 6 dimensions in two different coordinate systems, shown with* 10^5 *propagated MC samples for the Velox scenario after 20 revolutions.*

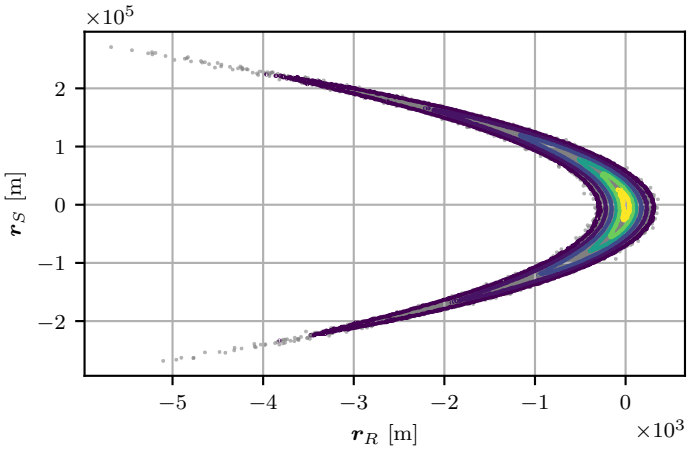


Figure 7.5: *The position uncertainty of a Gaussian distribution created in MEE from 10^5 propagated MC samples, shown in Cartesian coordinates in the Radial and Along-track frame, compared against the original MC samples in Cartesian coordinates, after 20 revolutions of the Velox scenario.*

Finally, it is noted that this approach is only used with uncertainty propagation methods that propagate a set of samples points from t_0 to t_{TCA}, such as MC and MF. For uncertainty propagation methods that result in a final Gaussian distribution, or a GMM, this approach is not needed, since all samples required at t_{TCA} can be directly sampled from this Gaussian distribution or GMM in Cartesian coordinates.

8

Conjunction Analysis Results

For each of the four scenarios described in Chapter 7, the P_C is calculated with the MC method, by generating 10^7 sampled state vectors for the primary and secondary object at t_{TCA}, and calculating the DCA_{true} of each sample pair in a one-on-one analysis. Using Equation 4.6, the P_C is calculated as the fraction of sample pairs for which the DCA_{true} is smaller than the chosen HBR for that scenario. To generate the sampled state vectors at t_{TCA}, the uncertainties propagated with MC samples or MF are transformed into Gaussian distributions in MEE, from which 10^7 samples are generated and converted back to Cartesian coordinates, using the approach described in Section 7.3. The uncertainties propagated using LinCov are sampled directly from the Gaussian distribution in Cartesian coordinates at t_{TCA}.

The uncertainties are propagated from t_0 to t_{TCA} with 10^5 MC samples as a baseline, as well as with 10^4 and 10^3 samples, with MF using various different settings, and with LinCov, to determine the effects of using a different UP method on the final value of P_C. The methods and settings used to propagate the uncertainties from t_0 to t_f are the same for each scenario, and are documented in Table 8.1. The settings for the MF method are cross-matched, resulting in 45 different simulations using MF.

Method	Setting parameter	Values used
MC	$N_{samples}$	{10^3, 10^4, 10^5}
LinCov	N.A.	N.A.
MF	$N_{samples}$	{10^3, 10^4, 10^5}
	ϵ_{lf}	{0.5, 1, 4, 10, 100}
	LF dynamics	{Kepler 2-body, low, medium}

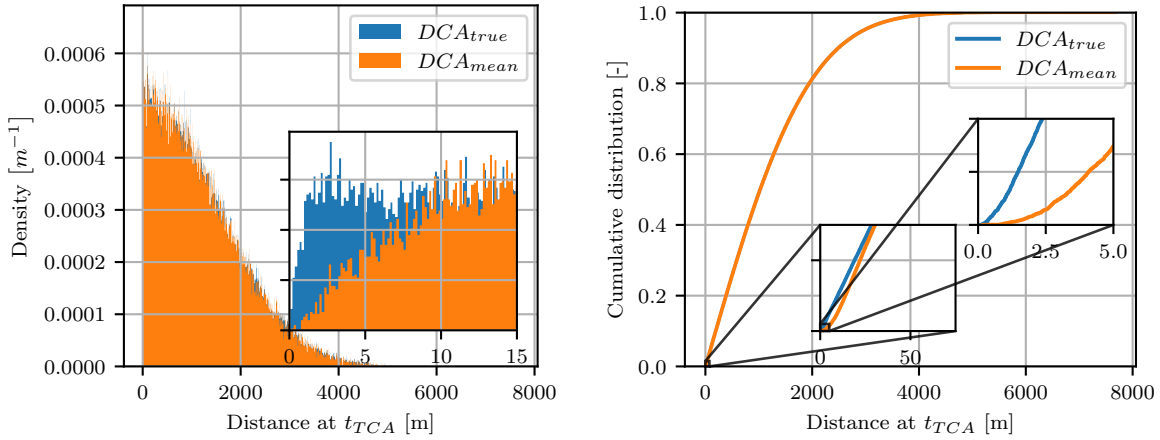
Table 8.1: *The various values used for the different settings of the UP methods, used for the propagation of the uncertainty from t_0 to t_{TCA}.*

The Kepler 2-body dynamics indicates that the LF samples are propagated using only 2-body dynamics between the state vector and the Earth, using the procedure described in Algorithm 3. With the 'low' setting for LF dynamics, the sample points are propagated using Tudat, where the Earths spherical harmonics of degree and order (2,2) are included, as well as the 3rd body point mass perturbation of the Sun and the Moon. With the 'medium' setting, the Earth's spherical harmonics of degree and order (2,2), the 3rd body point mass perturbation of the Sun and Moon, as well as the atmospheric drag and solar radiation are included as acceleration models using Tudat. It is also noted that when using 10^5 samples for MF, this is done using 10 batches, in each of which 10^4 samples are propagated using MF, with their own separate calculation and propagation of important samples. This is done to avoid out-of-memory errors in the process of computing the pivoted Cholesky decomposition, where a $10^5 \times 10^5$ matrix becomes too large to handle in Python.

8.1. True Time of Closest Approach

To determine the effect of calculating the distance at the true time of closest approach for each individual sample pair rather than using the distance at the same mean time of closest approach for all samples, the two methods are compared to each other using 10^6 MC sample pairs. The state vectors are sampled at $t_{TCA,mean}$ using Gaussian distributions in MEE, based on the uncertainty propagated from t_0 to $t_{TCA,mean}$ using 10^5 MC samples. Figure 8.1 shows the differences in the closest distances between the sample pairs. While the distributions seem almost similar for the sample pairs with large distances ($> 50m$), the distributions differ significantly for distances near $0m$. The DCA_{true} captures a lot more occurrences of distances between 0 and $10m$, which is expected, since for a given pair of samples, the $t_{TCA,mean}$ is most likely not the actual TCA for that pair, and the miss distance can be equal or smaller than the distance at $t_{TCA,mean}$. The difference in the distribution for small values of DCA is especially important for the calculation of P_C using a small HBR, since it directly changes the values for N_c. A comparison between different UP methods can still be done using the P_C values

determined with the DCA_{mean}, as long as the comparison is consistent. However, as seen in Figure 8.3, the N_t will be much smaller, and thus many more samples are required to generate accurate results. Therefore, the use of the DCA_{true} is essential for this study.



(a) *Histogram.*

(b) *Cumulative distribution.*

Figure 8.1: *The distribution of the Euclidean distance between sampled pairs at $t_{TCA,true}$ and at $t_{TCA,mean}$ for 10^6 sample pairs, for the short version of the low-velocity scenario, with zoomed-in views for smaller distances.*

To determine the $t_{TCA,true}$ and subsequently DCA_{true}, an iterative algorithm is used to repeatedly shrink the time interval around the $t_{TCA,true}$ until the time interval is small enough that the midpoint can be used as the value for $t_{TCA,true}$, which is described in Algorithm 2 in Appendix A. To use this algorithm, an initial time window of t_{window} around the $t_{TCA,mean}$ is needed to start with, which should contain the $t_{TCA,true}$ of all samples. A smaller time window leads to a faster convergence, and thus less computation time. An initial guess for t_{window} is taken based on the spread of the position uncertainty of 8.5σ (suggested by Chan [81]) divided by the relative velocity of the mean state vectors. The resulting differences between $t_{TCA,true}$ and $t_{TCA,mean}$ are visualised for one of the scenarios in Figure 8.2. For each scenario, it is deduced that for the number of sample pairs used in this study (10^7), the $t_{TCA,true}$ can be found within ± 2 seconds of the $t_{TCA,mean}$ for all sample pairs, leading to a use of $t_{window} = 2$s.

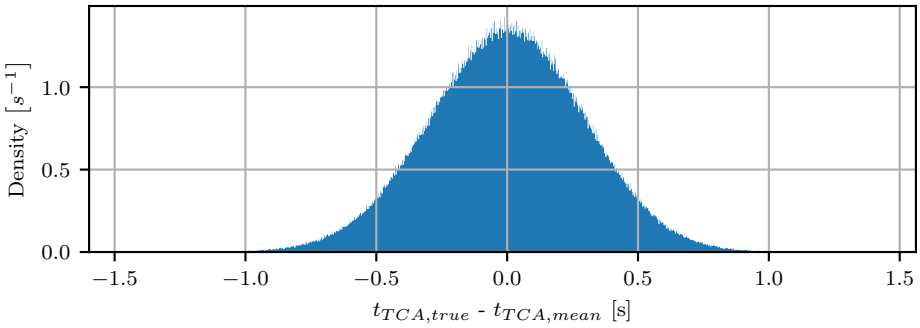
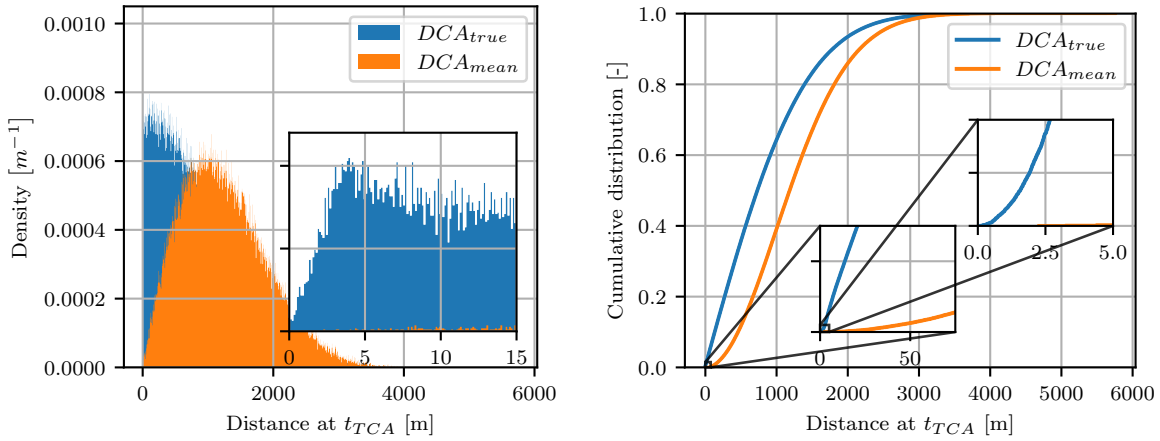


Figure 8.2: *The distribution of the time difference between the mean time of closest approach and the true time of closest approach of sample pairs, for the short version of the low-velocity scenario*

8.2. Influence of Relative Velocity

In Figure 8.3 it can be seen that the differences between the DCA_{true} and DCA_{mean} are much more visible for the high-velocity scenario than those for the low-velocity scenario shown in Figure 8.1. The different distances no longer follow the same distribution for sample pairs with high distances, and the difference between the counts of pairs with a small DCA_{true} or DCA_{mean} differs even more. For both scenarios, a sample pair that reaches a small DCA_{true}, can have a higher DCA_{mean} when the $t_{TCA,true}$ differs from the $t_{TCA,mean}$. The DCA_{mean} depends on the distributions of both uncertainties at $t_{TCA,mean}$, which follow the orbital trajectories

and cross each other at a higher angle for the high-velocity scenario than for the low-velocity scenario, due to the differences in the magnitude of the relative velocity between both objects. Therefore, in the high-velocity scenario, the samples have higher distances at $t_{TCA,mean}$ compared to $t_{TCA,true}$ than for the low-velocity scenario, for the sample pairs that result in small distances at $t_{TCA,true}$, thus leading to the increased difference in distributions for the high-velocity scenario.



(a) *Histogram.*

(b) *Cumulative distribution.*

Figure 8.3: *The distribution of the Euclidean distance between sampled pairs at $t_{TCA,true}$ and at $t_{TCA,mean}$ for 10^6 sample pairs, for the short version of the high-velocity scenario, with zoomed-in views for smaller distances.*

A second difference in the effects of calculating the DCA_{true} can be seen in Figure 8.4, where the differences between the $t_{TCA,true}$ and $t_{TCA,mean}$ are much smaller than those for the low-velocity scenario, shown in Figure 8.2. This can be explained by the higher relative velocity between the objects in each sample pair. The high relative velocity means that variations in the velocity vectors between sampled states and the mean states have smaller effects on the relative velocity between the states of the sample pair. For lower relative velocities, a similar change in the velocity vectors of the objects results in a larger change in the time of the closest approach. While this means that for the high-velocity scenario, a smaller t_{window} can be taken to calculate DCA_{true}, it does not mean that using DCA_{true} instead of DCA_{mean} is less important for the high-velocity scenario. Instead, the opposite is true, as indicated by the bigger difference in distributions in Figure 8.3.

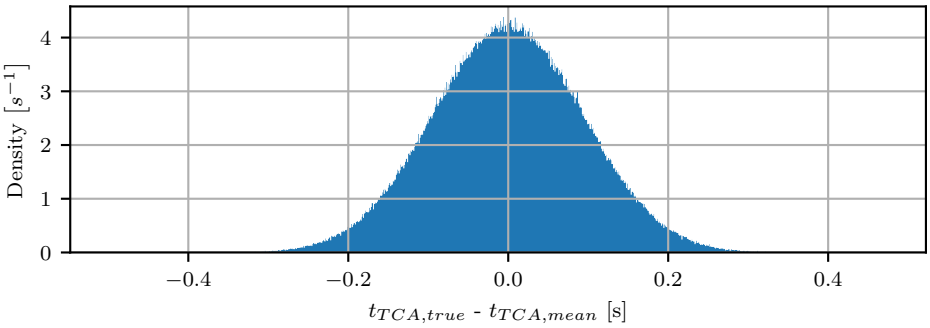
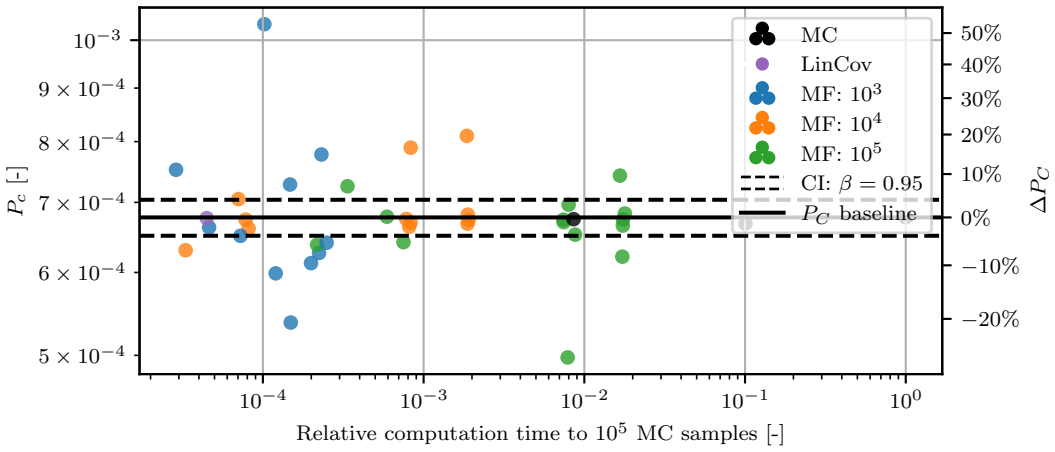


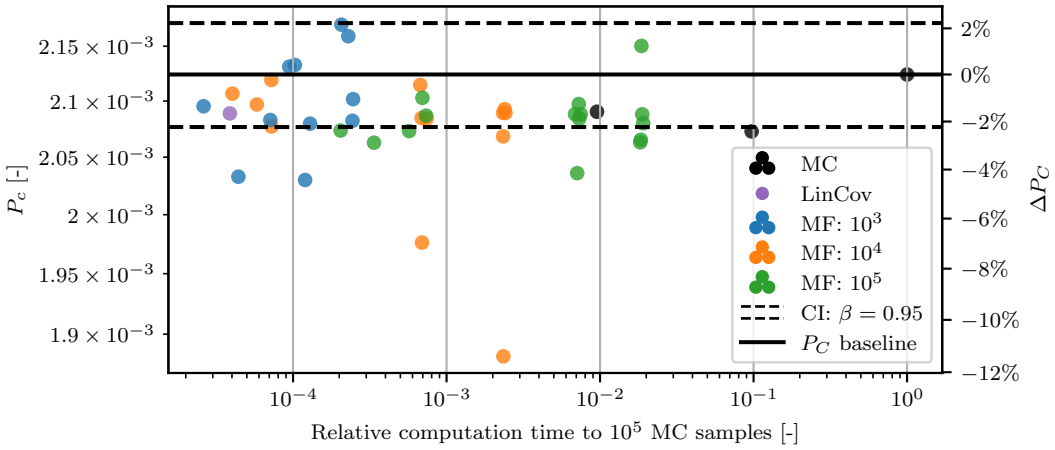
Figure 8.4: *The distribution of the time difference between the mean time of closest approach and the true time of closest approach of sample pairs, for the short version of the high-velocity scenario*

Figure 8.5a and Figure 8.5b show the collision probability P_C calculated using the uncertainties propagated by the different methods and settings listed in Table 8.1, for the short version of the high-velocity and low-velocity scenario respectively. In these figures, the baseline P_C, which is calculated using the uncertainties that have been propagated with 10^5 MC samples, is shown together with the 95% confidence interval, which is based on the number of samples used in the P_C calculation. The coloured results show the P_C values and relative computation times when the uncertainties are propagated to t_{TCA} using MC with smaller sample sizes, MF with various settings, or LinCov.

For both scenarios, shown in Figure 8.5, the P_C is within the same order of magnitude as the baseline P_C for all settings used for the MF method, as well as when using LinCov or MC with smaller sample sizes. Both scenarios also have a similar number of outliers and when the results are grouped for the $N_{samples}$ setting of MF, the only visible pattern is found in the computation time per group, indicating that the propagation of all the samples with LF dynamics makes up most of the computation time. However, there are no clear differences visible between the results of the high-velocity scenario and the results of the low-velocity scenario.



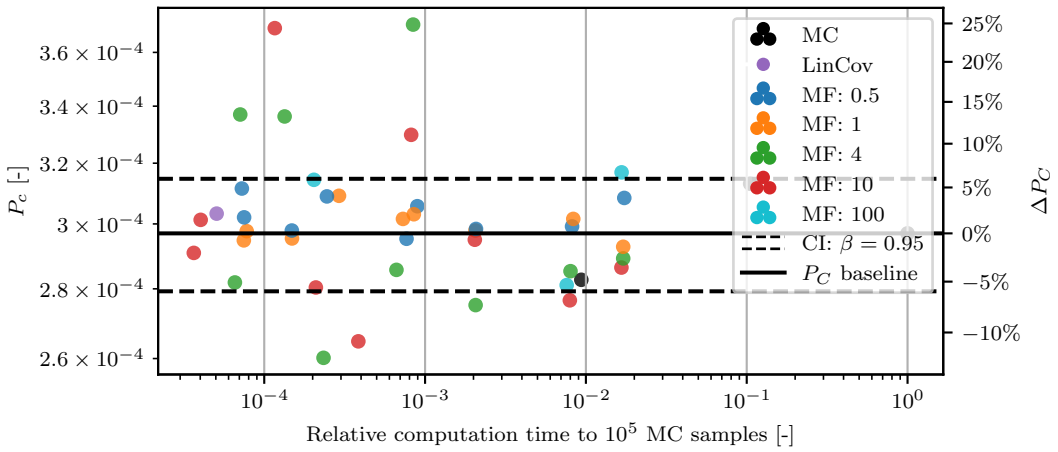
(a) *High-velocity scenario.*



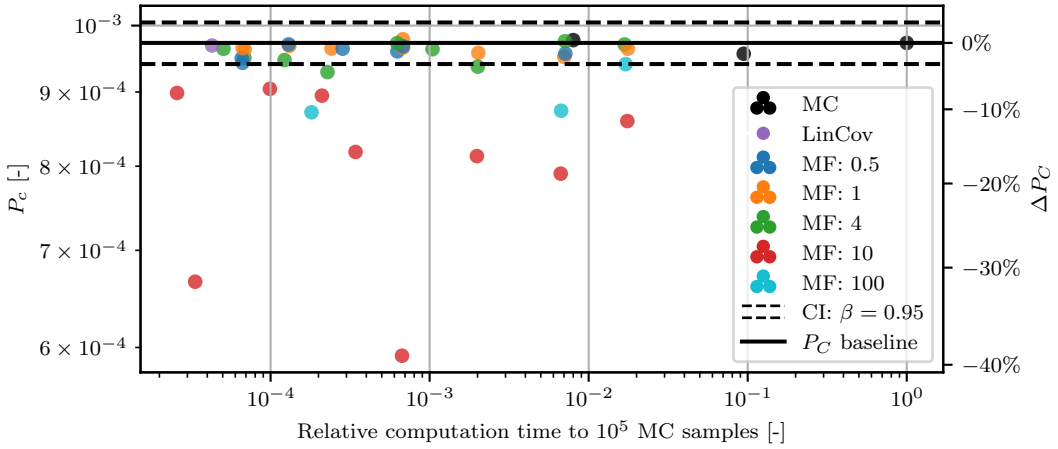
(b) *Low-velocity scenario.*

Figure 8.5: *The probability of collision between the uncertainties propagated using various methods and settings plotted against the computation time, and compared to the baseline P_C of the uncertainties propagated with 10^5 MC samples, for the short versions of the high-velocity and low-velocity scenario, with the MF settings grouped by $N_{samples}$.*

In Figure 8.6, the P_C values are again shown for the different methods and settings, but this time the MF settings are grouped and coloured by the ϵ_{lf} setting, and the results are shown for the long versions of both scenarios. It can be seen that for the low-velocity scenario, the outliers are clearly characterised by the two highest values for ϵ_{lf}, thus the MF propagations with the least accurate requirements lead to the largest deviations in the P_C. For the low-velocity scenario, MF settings with ϵ_{lf} of $0.5m$, $1.0m$ or $4.0m$ all result in a very accurate P_C value, with deviations within the 95% confidence interval. For the high-velocity scenario, the outliers also include the ϵ_{lf} value of $4.0m$, indicating that the resulting P_C value in the high-velocity scenario depends more on the accuracy of the expansions used in the MF propagation from t_0 to t_{TCA}, than in the low-velocity scenario. This can have several explanations, a likely reason being the difference in orbital trajectories and accelerations experienced between the two different scenarios. It can also be that the high relative velocity and short duration of the close encounter places more importance on the accuracy of the uncertainty propagation.



(a) *High-velocity scenario.*

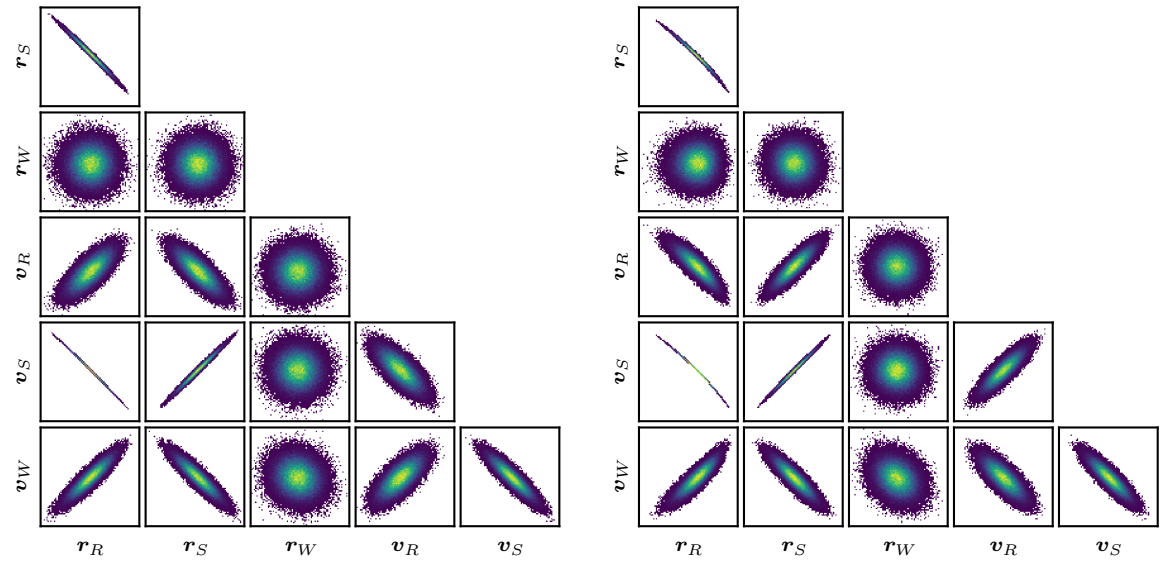


(b) *Low-velocity scenario.*

Figure 8.6: *The probability of collision between the uncertainties propagated using various methods and settings plotted against the computation time, and compared to the baseline P_C based on the uncertainties propagated with 10^5 MC samples, for the long versions of the high-velocity and low-velocity scenario, with the MF settings grouped by ϵ_{lf}.*

8.3. Influence of Propagation Time

The short versions of both scenarios have a propagation time of $\approx 48h$ while the long versions have a propagation time of $\approx 96h$. In Chapter 6 it has been shown that for longer propagation times, the uncertainty becomes increasingly non-Gaussian. An exception to this is a highly elliptical orbit, where the phase along the orbit matters, however, both the high-velocity and the low-velocity scenarios contain near-circular orbits. Figure 8.7 shows the distribution of the uncertainty of the primary object for the low-velocity case, both for the short and long version. While a sharp eye can notice a slight curvature in the R-S frame for the long version in Figure 8.7b, the distribution for both the short and long version still look Gaussian.



(a) *Short version* $\Delta t \approx 48h$ **(b)** *Long version* $\Delta t \approx 96h$

Figure 8.7: *Correlations between each of the 6 dimensions in the RSW coordinate frame, of the uncertainty of the primary object at* t_{TCA}*, shown with* 10^5 *propagated MC samples for the short and long version of the low-velocity scenario.*

It should be noted that the shape of the distribution in the R-S frame are aligned along the diagonals for both cases, which prevents the axes from being scaled such that the curvature is as clearly visible as in the figures shown in Figure 6.1 in Chapter 6. The propagated uncertainty of the short version is shown in Figure 8.8 again, where the 6 directions are the 6 eigenvectors of a covariance matrix fitted on the MC sample states. This ensures that the directions with the largest spread are taken as separate axes, which can be scaled independently such that curvatures in the shapes of the uncertainty are more easily visible. While the eigenvectors themselves are 6-dimensional and can not be used to make clear deductions about the distribution of the uncertainty in position or velocity respectively, it does clearly show that the propagated uncertainty is non-Gaussian already, even for the short version.

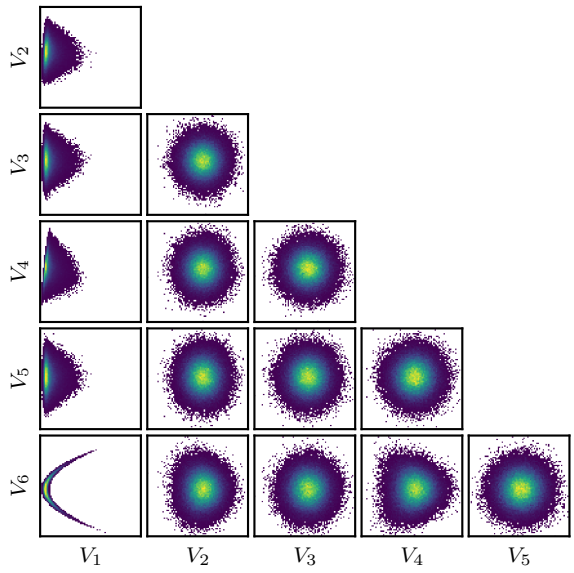
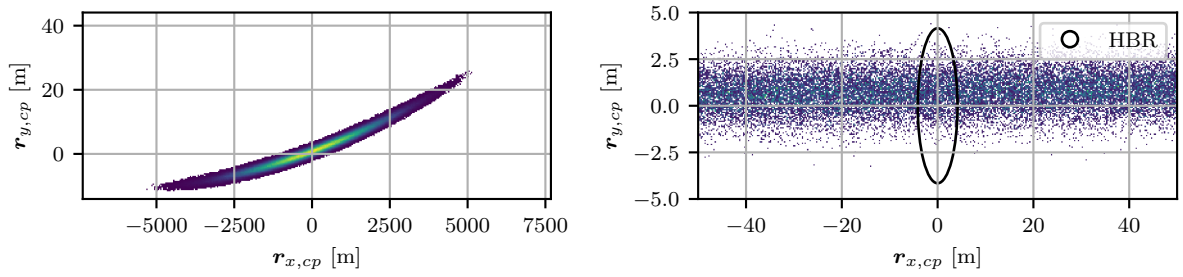


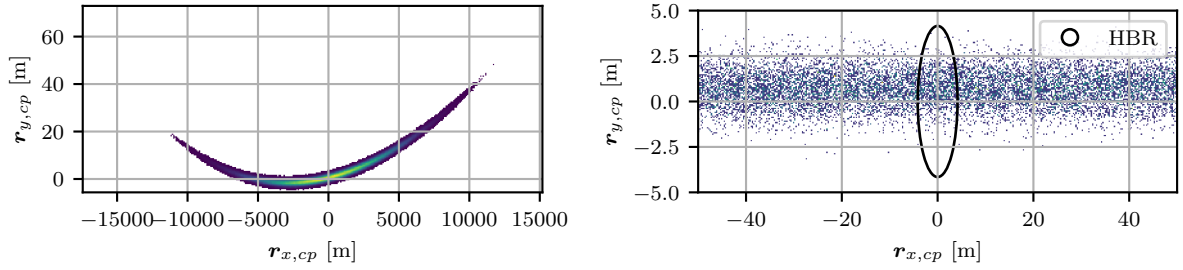
Figure 8.8: *Correlations between the projections of the state vector along each of the 6 eigenvectors of a fitted covariance matrix, of the uncertainty of the primary object at* t_{TCA}*, shown with* 10^5 *propagated MC samples for the short version of the low-velocity scenario.*

In Figure 8.9 the relative position vectors are shown in the conjunction plane for both the short and long version of the low-velocity scenario. From Figure 8.9 it can be seen that the distribution of sample pairs along the

conjunction plane becomes more spread out for the long scenario, where the shape follows a strong curvature that is already partially visible in the short version. In the zoomed versions, it can be seen that due to the larger spread of the uncertainty in the long version, the density of sample pairs with relative position vectors near $[0,0,0]$, the area where collisions are counted, becomes more sparse. Thus, the P_C values become smaller, since the total number of sample pairs are the same in both versions, and thus the confidence interval of 95% becomes larger too.



(a) *Short version* $\Delta t \approx 48h$



(b) *Long version* $\Delta t \approx 96h$

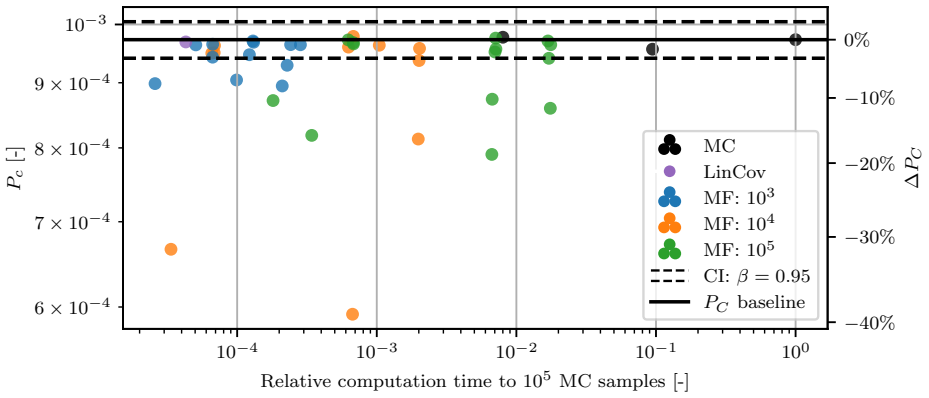
Figure 8.9: *The relative position in the conjunction plane for* 10^6 *sample pairs, including a zoomed version with the HBR added for scale, for the short and long version of the low-velocity scenario.*

For a direct comparison between the short and long versions of both scenarios, the reader is referred to Figures F.1 and F.2 in Appendix F. Despite the difference in the spread of samples between the versions noted above, no clear differences are seen between the short and long versions when comparing the P_C values of the propagations with different settings of MF to the baseline P_C using MC samples. While for the low-velocity scenario the spread of the relative error of P_C using MF seems to be slightly larger for the long version, the opposite is true for the high-velocity scenario. A possible cause can be that the initial uncertainty at t_0 has been scaled down too much, such that the distribution of the uncertainty does not spread out as much as expected, based on the effects of the propagation times seen in the comparison study, described in Section 6.1.

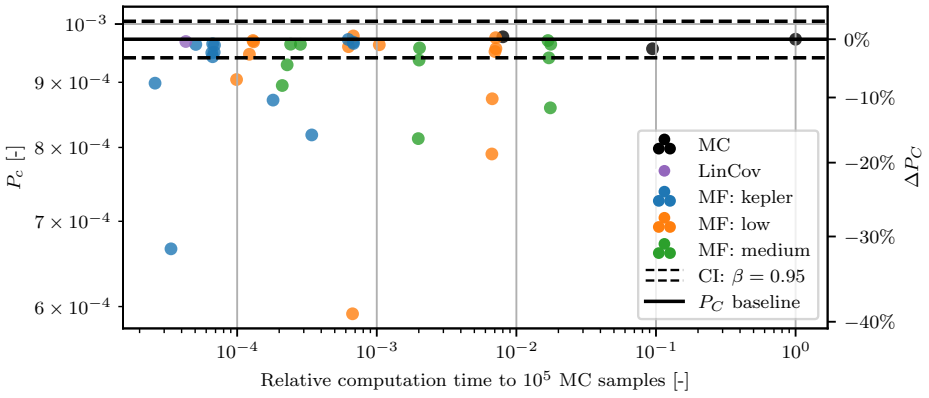
8.4. Multi-Fidelity Settings

In Figure 8.10 the P_C values are shown for the long version of the low-velocity scenario, where the colour groupings of the MF results are based on the three different settings changed: ϵ_{lf}, $N_{samples}$ and the LF dynamics. Figure 8.10a and Figure 8.10b demonstrate clear correlations between the $N_{samples}$ and the computation time as well as between the LF dynamics and the computation time. For a higher number of points, the computation time increases. Similarly, for a higher level of accuracy of the LF dynamics (medium > low > Kepler), the computation time also increases significantly. Both of these settings are closely related to the computation time required to propagate the LF sample points, which evidently makes up most of the total computation time of the MF propagation. Surprisingly, the accuracy of the LF dynamics does not seem to have any clear influence on the P_C values, suggesting that the mapping using the important samples can adequately correct the LF samples, even when using Kepler dynamics.

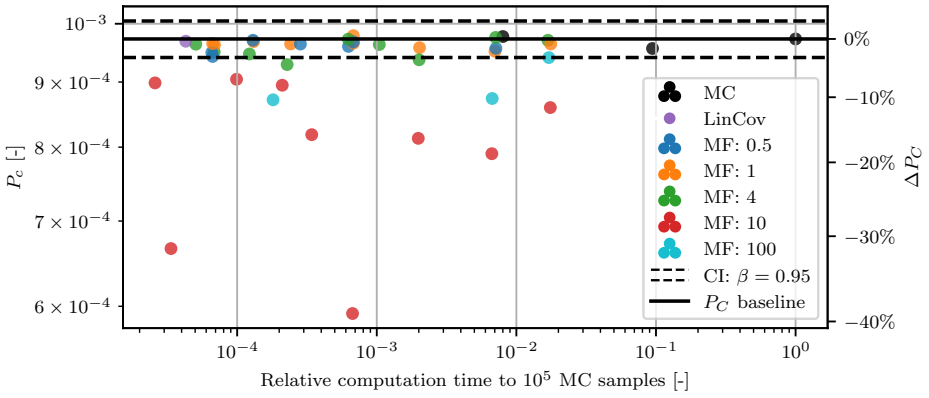
Figure 8.10c shows that the ϵ_{lf} setting is the most important influence on the final P_C values. The largest values of $100m$ and $10m$ are the only settings for which the P_C falls outside of the 95% confidence interval. On the contrary, the ϵ_{lf} does not show any significant influence on the computation time.



(a) *Grouped by $N_{samples}$ settings.*



(b) *Grouped by LF dynamics.*



(c) *Grouped by ϵ_{lf} settings.*

Figure 8.10: *The probability of collision between the uncertainties propagated using various methods and settings plotted against the computation time, and compared to the baseline P_C, for the long version of the low-velocity scenario, with different colour groupings of the MF settings.*

In Figure 8.11 the effect of the ϵ_{lf} setting on the P_C values is investigated further, by plotting the P_C values against the number of important samples used in the MF propagation, which is directly influenced by the ϵ_{lf} setting. From Figure 8.11 it is clear that for a low number of important samples, $r_{mf} < 6$, the P_C value deviates significantly from the true P_C (defined here by the baseline P_C calculated using a MC propagation with 10^5 samples). Whereas for $r_{mf} \approx$ 7-8, the P_C value is within the 95% confidence interval, which is a relative error of smaller than 4% from the actual value in this case. Figure 8.11 also shows that with 7-8 important samples, the P_C values are close to the baseline irrespective of the $N_{samples}$ or LF dynamics. This suggests that when using a low value for ϵ_{lf} to ensure enough important samples, the $N_{samples}$ and accuracy of LF dynamics can be taken as low values, to obtain a P_C value close to the baseline using MC, while having a very large speed-up in computation times compared to using MC (more than 4 orders of magnitude).

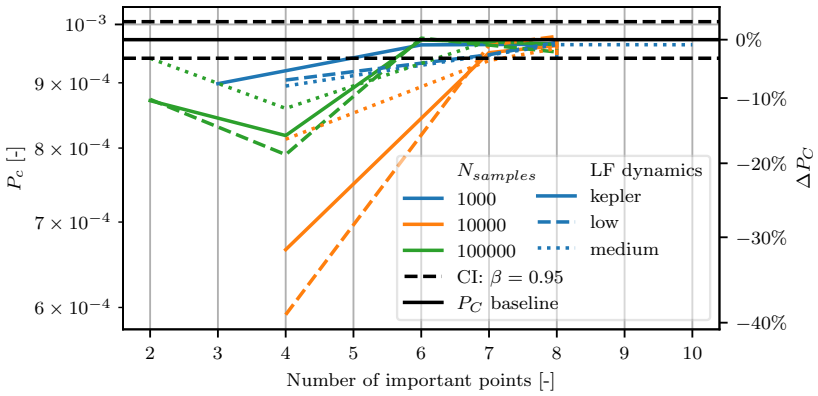


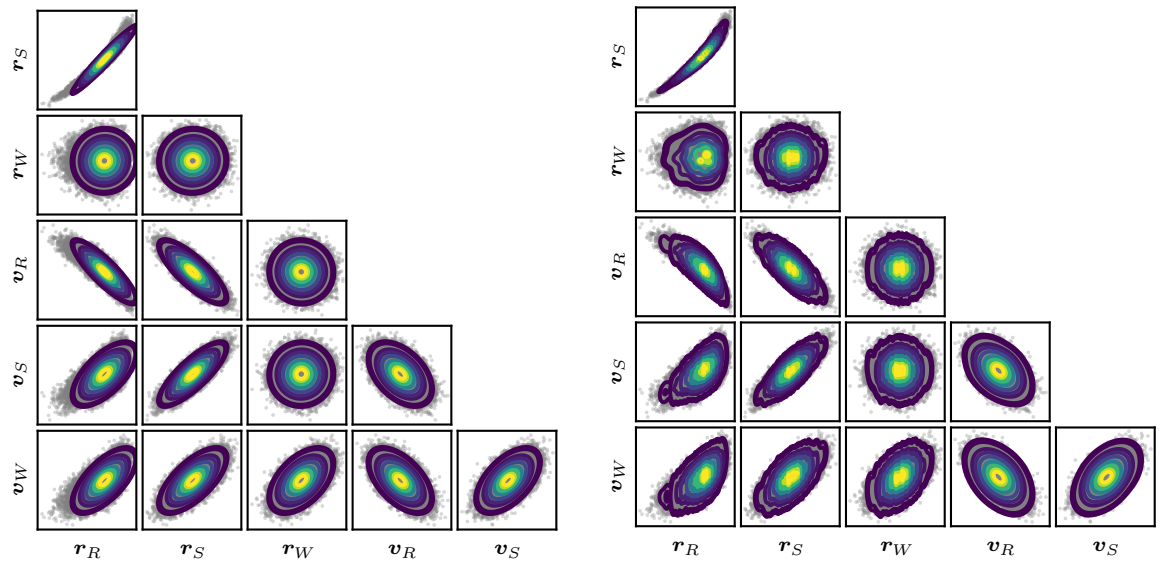
Figure 8.11: *The probability of collision between the uncertainties propagated using MF plotted against the number of important samples used in the MF propagation and compared to the baseline P_C , for the long version of the low-velocity scenario.*

The results shown in Figure 8.11 are also documented for the other scenario and versions in Appendix F. While the patterns between the number of important samples and the P_C values are less clear for the other scenarios/versions, it is still the case that for $r_{mf} = 8$, the P_C values generated with all settings of $N_{samples}$ and LF dynamics are within the 95% confidence interval.

8.5. Linearised Covariance

A surprising result that has not been mentioned yet, is the performance of using LinCov as the UP method from t_0 to t_{TCA}. This method has been included in the comparison to show the need for an accurate UP that can propagate the uncertainty even without assumptions of Gaussianity. It is shown in Chapter 6 that when the uncertainty becomes non-Gaussian, such as after 50 orbital revolutions of the Velox scenario (which corresponds to $\Delta t = 78.83h$), that LinCov can no longer approximate the shape of the distribution generated using MC samples. Yet, in the long version of both scenarios (which also contain LEO orbits, for a longer Δt than the $78h$ of the Velox scenario after 50 revolutions), the P_C values generated using the uncertainties propagated with LinCov are nearly identical to that when using MC.

A possible explanation could lie in the way that LinCov approximates the non-Gaussian uncertainty, as shown in Figure 8.12a. Comparing LinCov to the MC samples, the mean state vectors are identical, as well as the spread of uncertainty in each of the directions except for the radial (R) direction. This becomes clear when comparing the results to using the MF method shown in Figure 8.12b, where the non-Gaussian distribution in the radial direction is approximated better. In the R-S frame of LinCov, where the curvature of the distribution is not approximated well, the 'length' and 'width' of the shapes are the same as for the MC samples. This result, combined with the results of the P_C values, suggests that an accurate approximation of the curvature in the R-S frame is not necessary to accurately predict the collision probability of two space objects. However, it is possible that this result arises only when both object have similar orbits (same altitude and eccentricity), their uncertainties at t_0 are of the same size, and/or the uncertainties of both objects are propagated with the same propagation time.



(a) *LinCov*

(b) *MF with Kepler LF dynamics,* $N_{samples} = 10^4$ *and* $\epsilon_{lf} = 1m$.

Figure 8.12: *Correlations between each of the 6 dimensions in the RSW coordinate frame, of the propagated uncertainty of the primary object at* t_{TCA} *using different UP methods, shown as contour lines placed on top of* 10^5 *propagated MC samples in gray, for the long version of the high-velocity scenario.*

Besides the specifics of the scenarios chosen for the simulations, the effect that using a Gaussian distribution in MEE to generate all the sample pairs at t_{TCA} has on the differences between the P_C values using different UP methods is also unknown. Similarly, the calculation of the true time of closest approach for each individual sample pair, after propagating the full uncertainty to the mean time of closest approach, can also influence the way that inaccuracies in the propagated uncertainties affect the final collision probability. To quantify these effects requires more research, which is outside the scope of this study. However, the insignificant differences between P_C values when using MC as the UP method with 10^3 or 10^4 samples, does suggest that the sampling method used at t_{TCA} plays a role in the relation between the UP method used and the final P_C value.

9

Conclusions

In the introduction of this report, the main objective of the research has been stated as:

Reduce the safety risks of space debris by improving the efficiency of computationally expensive tasks in conjunction analysis.

To achieve this, this study aimed to answer the following research question:

How can novel orbit uncertainty propagation methods contribute to the improvement upon the calculation of the collision probability between two space objects?

9.1. Sub-questions

To build up an answer to this main research question, a number of sub-questions are answered individually, from which a final conclusion to the main research question is developed afterwards.

How do different orbit uncertainty methods compare to each other in terms of accuracy and computational efficiency?

To answer this question, 7 different orbit uncertainty propagation methods have been analysed and compared to results generated with Monte Carlo samples. The uncertainty propagation methods are used with various different settings to propagate the uncertainty of a space object in 5 unique test case scenarios, for various numbers of orbital revolutions for each test case. The orbits of the scenarios differ in altitude, eccentricity and inclination, and include two low Earth orbits, two highly elliptical orbits and one geostationary orbit. The accuracy of each propagated uncertainty is determined using a normalised distance metric between the propagated uncertainty and 10^5 propagated MC samples. The methods analysed are the Linearised Covariance method (LinCov), the Unscented Transform (UT), the Multi-Fidelity method (MF), Gaussian Mixture Models (GMMs) combined with the UT and MF, Adaptive Entropy-based Gaussian-mixture Information Synthesis (AEGIS), and Polynomial Chaos Expansions (PCEs).

It is found that for all scenarios and propagation lengths, LinCov is the most computationally efficient method. Although it has the lowest computation times, it is only able to approximate the MC samples well when the final uncertainty is still Gaussian, with a significant drop in accuracy for long propagation times or highly elliptical orbits. The UT has a similar performance in accuracy as LinCov, but has a longer propagation time. While both methods assume a Gaussian final uncertainty, and perform similarly on the distance metric with respect to the MC samples, the approximations are slightly different. Where LinCov ignores the curvature of the uncertainty and captures the size and location of the uncertainty well near the propagated mean state vector, the UT spreads out more, and captures more of the tails of the uncertainty. For the implemented splitting directions in Cartesian coordinates, the GMMs produce poor results. While it has a higher computation time than LinCov, the various settings result in similar or worse results in terms of accuracy. The implementation combining GMMs with MF produces similar accuracies to the combination of GMMs with UT for all scenarios, and provides a significant speed-up of 1-2 orders of magnitude. The PCEs have shown the highest accuracies when compared to MC samples, and excel for non-Gaussian uncertainties compared to most other methods tested. When using an expansion order of 2 and a number of samples equal or larger than the number of coefficients in the expansion, the PCE is shown to converge to the MC samples, even for non-Gaussian distributions. The MF method has shown to be an excellent alternative between LinCov and PCEs. Most settings used for MF require less computation time than PCEs, while also being able to approximate the final uncertainty well, even for non-Gaussian uncertainties where LinCov fails. Using a higher number of samples only has a small effect on the accuracy of MF, while influencing the computation time significantly. The low-fidelity tolerance level, which determines the number of important samples used, has very small effects on the computation time, but majorly influences the accuracy of the final uncertainty. While the optimal values for the low-fidelity toler-

ance level varies per scenario, it is recommended to use a value within one order of magnitude as the smallest tolerance level that can be reached by the low-fidelity approximation.

Is Multi-Fidelity a feasible orbit uncertainty propagation method to use for conjunction analysis?

The MF method has been shown to be a feasible method to propagate the uncertainties of two space objects from a time of observation to the time of closest approach in a conjunction assessment. The MF method has been used to propagate the uncertainty of both objects for $48h$ and $96h$ for two realistic conjunction assessment scenarios, one with a high relative velocity between both objects, and one with a low relative velocity, leading to 4 test cases. In each test case, the collision probability P_C is calculated using the uncertainties propagated by various different settings of MF, as well as using the uncertainty propagated with 10^5 MC samples to generate a P_C value that can be used as a baseline to compare to. For almost all of the settings used of MF, the P_C value is within the same order of magnitude as the baseline P_C, where for most settings the deviation falls within the 95% confidence interval, which has a relative error of $\approx$0.05. It is found that when using small values for the low-fidelity tolerance level, such that enough important samples are propagated, the propagation model used for the low-fidelity propagation can be as simple as two-body Kepler dynamics while still producing P_C values near-identical to the baseline P_C. This leads to a significant speed-up over the computation time of MC samples, of up to 4 orders of magnitude.

How do challenging scenarios such as long time horizons, highly elliptical orbits and low relative velocities, affect the performance of High-Fidelity orbit uncertainty propagation in conjunction assessment?

The 4 test cases for which the MF method has been used to propagate the uncertainties of the two space objects before calculating the P_C, consist of two short propagation times ($\Delta t = 48h$) and two long propagation times ($\Delta t = 96h$), as well as two scenarios with a high relative velocity and two scenarios with a low relative velocity. The long versions have demonstrated that even for long time horizons, the MF method can be used to generate accurate results for the P_C value. While a longer time horizon results in lower P_C values overall, due to the growing size of the uncertainty during the propagation resulting in more dilution, the differences between the P_C values when using MF and the baseline P_C value do not show different behaviour for the longer propagation times compared to the short propagation times. For the low relative velocity scenarios, the deviation of the P_C value from the baseline has been shown to be a clear result of the number of important samples used in the MF method for the uncertainty propagation, while for the high relative velocity scenarios this result is less visible. Compared to those of the high relative velocity scenario, the results for the low relative velocity scenario show no decrease in performance of the MF method when used in a conjunction assessment.

9.2. Main Research Question

Based on the answers to the sub-questions discussed above, a final conclusion can be presented towards the main research question of this study:

How can novel orbit uncertainty propagation methods contribute to the improvement upon the calculation of the collision probability between two space objects?

In this study, the Multi-Fidelity method has been proven to be an adequate alternative to the currently used methods for uncertainty propagation. It combines computationally efficient low-fidelity propagations with a limited use of high-fidelity propagations to obtain an optimal combination of low computation times and high accuracy estimations. The MF method performs especially well for propagating non-Gaussian uncertainties, for which it is more computationally efficient than the available alternatives: using PCEs or Monte Carlo samples. The other methods implemented in this study, LinCov, the UT, AEGIS and GMMs, show declining performance in accuracy for test case scenarios with long propagation times.

Furthermore, the Multi-Fidelity method has been used as a successful substitute for using Monte Carlo samples to propagate the uncertainty of two space objects in several conjunction assessments. Using Multi-Fidelity as the method to propagate uncertainties from an initial epoch up to the time of closest approach, as opposed to using Monte Carlo samples, caused only small and converging changes in the calculated collision probabilities, while providing a significant speed up in the computation time. When using a small tolerance level in the MF method, such that at least 8 important samples are used in the propagation, the collision probability can be calculated with a relative error of approximately 5% compared to using MC samples. Using the MF method in these cases results in a smaller computation time of up to 4 orders of magnitude. These results have been

found for a conjunction assessment with a high relative velocity as well as with a low relative velocity between the objects, and for time windows of up to 4 days.

Additionally, the LinCov method has been included in the conjunction assessment study, which has shown surprising results that an uncertainty propagation method that assumes final Gaussian uncertainties, still performs well in the conjunction assessment where both objects have non-Gaussian uncertainties. This suggests that poor performing methods in the uncertainty propagation comparison study still have the potential to accurately predict the collision probability in conjunction assessments. Finally, it is concluded that the Multi-Fidelity method tested in this study, can significantly improve upon the computation times of calculating the collision probability between two space objects.

9.3. Discussion

This study has been limited both in time and resources, and thus the research that has been performed can be improved upon in specific areas, or benefit from additional analyses. In this section, the limitations of this research are discussed in more detail, as well as the effects thereof on the results and conclusions presented.

In the comparison study of the uncertainty propagation methods, 8 different approaches have been implemented, to attempt to provide a broad discussion of the different methods available in literature. However, while these 8 methods are deemed promising prospects or commonly used methods, they do not encompass all available uncertainty propagation methods, and some alternatives are left out. Examples of uncertainty propagation methods that have not been included are the use of intrusive polynomial chaos expansions and differential algebra techniques. Similarly, newly developed techniques, such as artificial intelligence (e.g. neural networks [120–122]) could provide better alternatives to the uncertainty propagation methods implemented in this study.

On the contrary, the large number of complex methods implemented in this study has resulted in a limited depth of the investigation into each uncertainty propagation method. Since only a single comparison study is done, albeit with various settings used and different test cases, the full potential of some methods are not reached. The GMMs have been shown to perform better when the nonlinearity is restricted to a small number of dimensions, such as when using sets of osculating elements rather than a Cartesian coordinate system. Similarly, the UT and AEGIS could provide better results when combined with sets of osculating elements. PCEs provide many options to use, such as intrusive, non-intrusive, choice of integration schemes, coefficient solving techniques and expansion orders, which have not all been utilised. Finally, for the MF method, the level of accuracy of both the high-fidelity and low-fidelity dynamics used has not been altered in the comparison study, which could have shown more potential in the computation times of MF compared to other methods. While many methods may not have been used in the most optimal way for the test case scenarios, a wide variety of settings is used where possible, at a similar level of complexity for each method, such that a generic widespread comparison between the various methods can still be made.

In the conjunction assessment study, the research has been limited by the total computation time of the uncertainty propagations and collision probability calculations. Since critical conjunction events often have collision probabilities as low as 10^{-4}, a number of samples of 10^7, or even more, is desired to provide statistically accurate information. In this study, 10^7 is the upper limit on the number of samples that can realistically be used in the simulations due to a limit of computing power. This restricted the possibilities of test cases with low collisions probabilities, such as when using longer time horizons, larger miss distances or larger uncertainties.

The finite computing power also means that not all 10^7 samples can be propagated from the start of the propagation, neither for the MC method nor for the MF method, which has led to the use of a Gaussian distribution in modified equinoctial elements (MEE) to sample from at the time of closest approach. The effect of using this Gaussian distribution, which has been fitted on the 10^3-10^5 samples propagated using MF and MC, has not been studied in this research. While it has been verified that the shape of the uncertainty propagated with MC and MF can be approximated well by a Gaussian distribution in MEE, it is well possible that the inclusion of this intermediate step at the time of closest approach mitigates the effect that a less accurate uncertainty propagation method has on the final collision probability.

A surprising result that could be a consequence of this effect, is that the use of the LinCov method to propagate the uncertainties to the time of closest approach does not lead to significant deviations in the collision probability either. Since the LinCov method fails to approximate non-Gaussian uncertainties, the high performance of this method in conjunction assessment test cases that include non-Gaussian uncertainties can indicate that the test cases are not harsh enough to truly test the performance of the Multi-Fidelity method either. However,

using the diverse test cases of this study, the results shown for the Multi-Fidelity method are still promising and indicate high potential for the use of the method in conjunction assessments.

9.4. Recommendations

Following the limitations of this research outlined above, a number of recommendations for further research can be made, to provide further insights in the problems investigated in this study, or to complement it with more attention to areas that are not fully covered in this research.

It is recommended that for further comparison studies between various uncertainty propagation methods, the representation of the uncertainty in osculating elements is considered. The comparison study performed in this research has been limited to the propagation of uncertainties in Cartesian coordinates, whereas combining methods such as GMMs, the UT and AEGIS with modified equinoctial elements could provide much better results, possible shifting the relative advantages and disadvantages of the various methods.

While the second part of this research has been focused on the MF method, the comparison study also highlights the capabilities of the PCEs to accurately approximate non-Gaussian uncertainties. Studying the use of PCEs in conjunction assessment is not new, but can certainly benefit from more attention, possibly even by applying PCEs to the same test case scenarios and methods used to calculate the collision probability in this research.

The different settings used for MF in the conjunction assessment show that using Keplerian 2-body orbits for the low-fidelity dynamics with small values for the low-fidelity tolerance provides very low computation times, while still resulting in similar collision probabilities compared to propagating with MC samples. For further research, it is recommended to use these settings for the MF method, and to perform more extensive testing on the collision probabilities generated, to determine the accuracy of this method in different circumstances, with more reliable results. Test cases can include highly elliptical orbits, longer time windows, different sizes of initial uncertainties, and varying sizes of the hard-body radius. These different test cases will result in different values for the calculated P_C, which will likely be smaller values, since the test cases used in this research result in relatively high values of P_C to allow accurate estimations with a limited number of samples. For further research, larger sample sizes are recommended, to extend the range of P_C values that can be estimated with MF.

Besides different test case scenarios, the optimal settings of MF can also be investigated further using different methods to calculate the collision probability. This research focused on a single method to determine the collision probability with a large number of MC sample pairs generated at the time of closest approach, while in reality there are many different methods used to determine the collision probability with varying computation times and significances of their results. The use of MF as an uncertainty propagation method can be investigated in combination with analytical methods, such as Foster's, Alfano's or Hall's method. Similarly, with small modifications to reduce the size of the required matrix multiplications (such as using batches to determine the important samples), the MF method can be used to propagate all required samples from the initial epoch to the time of closest approach, thereby eliminating the need for the Gaussian distributions in MEE as discussed above. With this direct generation and propagation of the samples using MF, the effectiveness of the method can be assessed further.

Finally, the surprising result of LinCov used in the conjunction assessment study sparks the question whether LinCov can also be used to propagate the uncertainties of the objects to the time of closest approach, before calculating the collision probability with the Monte Carlo method. Further research can investigate the performance of LinCov in conjunction assessment when combined with Monte Carlo collision probability methods, to determine its potential utility. Additional research can also focus on the effects of the accuracy of the propagated uncertainty on the final collision probability, when the true time of closest approach is determined individually per generated sample pair after sampling at the mean time of closest approach. This can potentially lead to an explanation of these surprising results in this study.

Bibliography

[1] ESA Space Debris Office, "Esa's annual space environment report," 07 2024. [Online]. Available: https://www.sdo.esoc.esa.int/environment_report/Space_Environment_Report_latest.pdf

[2] F. Alby, E. Lansard, and T. Michal, "Collision of cerise with space debris," in *Proceedings of the Second European Conference on Space Debris*, vol. 393, 01 1997.

[3] T. Kelso, "Analysis of the iridium 33-cosmos 2251 collision," in *Proceedings of the 19th AIAA/AAS Astrodynamics Specialist Conference*, vol. 135, 09 2009.

[4] R. Rovetto and T. Kelso, "Preliminaries of a space situational awareness ontology," in *Proceedings of the AAS/AAIA Spaceflight Mechanics Meeting*, 02 2016.

[5] Y. Luo and Z. Yang, "A review of uncertainty propagation in orbital mechanics," *Progress in Aerospace Sciences*, vol. 89, pp. 23–39, 12 2016.

[6] K. DeMars, R. Bishop, and M. Jah, "A splitting gaussian mixture method for the propagation of uncertainty in orbital mechanics," in *Proceedings of the AAS Space Flight Mechanics Meeting*, vol. 140, 02 2011, pp. 1419–1438.

[7] B. Jones, N. Parrish, and A. Doostan, "Postmaneuver collision probability estimation using sparse polynomial chaos expansions," *Journal of Guidance, Control, and Dynamics*, vol. 38, no. 8, pp. 1425–1437, 2015.

[8] M. Lohr, R. Joseph, R. Fry, and J. Nelson, "A fokker-planck model for a two-body problem," in *Proceedings of the AIP Conference*, vol. 617, 05 2002, pp. 340–371.

[9] A. Poore, J. Aristoff, and J. Horwood, "Covariance and uncertainty realism in space surveillance and tracking," Numerica Corporation Fort Collins United States, Tech. Rep., 06 2016.

[10] J. Handschin and M. David, "Monte carlo techniques to estimate the conditional expectation in multi-stage non-linear filtering†," *International Journal of Control*, vol. 9, pp. 547–559, 1969.

[11] H. Akashi and H. Kumamoto, "Random sampling approach to state estimation in switching environments," *Automatica*, vol. 13, no. 4, pp. 429–434, 1977.

[12] G. Kitagawa, "Monte carlo filter and smoother for non-gaussian nonlinear state space models," *Journal of computational and graphical statistics*, vol. 5, no. 1, pp. 1–25, 1996.

[13] A. Doucet, S. Godsill, and C. Andrieu, *On sequential simulation-based methods for Bayesian filtering*. Department of Engineering, University of Cambridge Cambridge, UK, 1998.

[14] N. Gordon, D. Salmond, and A. Smith, "Novel approach to nonlinear/non-gaussian bayesian state estimation," *IEE proceedings F (radar and signal processing)*, vol. 140, no. 2, pp. 107–113, 1993.

[15] M. Isard and A. Blake, "Contour tracking by stochastic propagation of conditional density," in *Proceedings of the 4th European Conference on Computer Vision*. Springer, 1996, pp. 343–356.

[16] M. West, "Mixture models, monte carlo, bayesian updating, and dynamic models," *Computing Science and Statistics*, pp. 325–325, 1993.

[17] S. Brooks, A. Gelman, G. Jones, and X. Meng, *Handbook of markov chain monte carlo*. CRC press, 2011.

[18] B. Schutz, B. Tapley, and G. Born, *Statistical orbit determination*. Elsevier, 2004.

[19] J. Crassidis and J. Junkins, *Optimal estimation of dynamic systems*. Chapman and Hall/CRC, 2004.

[20] O. Montenbruck and E. Gill, *Satellite Orbits*. Springer, 01 2000.

[21] J. Junkins, M. Akella, and K. Alfriend, "Non-gaussian error propagation in orbital mechanics," *Journal of The Astronautical Sciences*, vol. 44, pp. 541–563, 1996.

[22] S. Julier, J. Uhlmann, and H. Durrant-Whyte, "A new approach for filtering nonlinear systems," in *Proceedings of 1995 American Control Conference-ACC'95*, vol. 3. IEEE, 1995, pp. 1628–1632.

[23] ——, "A new method for the nonlinear transformation of means and covariances in filters and estimators," *IEEE Transactions on Automatic Control*, vol. 45, no. 3, pp. 477–482, 2000.

[24] S. Julier and J. Uhlmann, "Reduced sigma point filters for the propagation of means and covariances through nonlinear transformations," in *Proceedings of the 2002 American control conference (IEEE Cat. No. CH37301)*, vol. 2. IEEE, 2002, pp. 887–892.

[25] ——, "Unscented filtering and nonlinear estimation," *Proceedings of the IEEE*, vol. 92, no. 3, pp. 401–422, 2004.

[26] G. Terejanu, P. Singla, T. Singh, and P. Scott, "Uncertainty propagation for nonlinear dynamic systems using gaussian mixture models," *Journal of guidance, control, and dynamics*, vol. 31, no. 6, pp. 1623–1633, 2008.

[27] J. Horwood, N. Aragon, and A. Poore, "Gaussian sum filters for space surveillance: theory and simulations," *Journal of Guidance, Control, and Dynamics*, vol. 34, no. 6, pp. 1839–1851, 2011.

[28] K. DeMars, R. Bishop, and M. Jah, "Entropy-based approach for uncertainty propagation of nonlinear dynamical systems," *Journal of Guidance, Control, and Dynamics*, vol. 36, no. 4, pp. 1047–1057, 2013.

[29] K. Vishwajeet, P. Singla, and M. Jah, "Nonlinear uncertainty propagation for perturbed two-body orbits," *Journal of Guidance, Control, and Dynamics*, vol. 37, no. 5, pp. 1415–1425, 2014.

[30] V. Vittaldev and R. Russell, "Space object collision probability using multidirectional gaussian mixture models," *Journal of Guidance, Control, and Dynamics*, vol. 39, no. 9, pp. 2163–2169, 2016.

[31] J. Horwood and A. Poore, "Adaptive gaussian sum filters for space surveillance," *IEEE transactions on automatic control*, vol. 56, no. 8, pp. 1777–1790, 2011.

[32] M. Psiaki, J. Schoenberg, and I. Miller, "Gaussian sum reapproximation for use in a nonlinear filter," *Journal of Guidance, Control, and Dynamics*, vol. 38, no. 2, pp. 292–303, 2015.

[33] G. Terejanu, P. Singla, T. Singh, and P. Scott, "Adaptive gaussian sum filter for nonlinear bayesian estimation," *IEEE Transactions on Automatic Control*, vol. 56, no. 9, pp. 2151–2156, 2011.

[34] R. Park and D. Scheeres, "Nonlinear mapping of gaussian statistics: theory and applications to spacecraft trajectory design," *Journal of guidance, Control, and Dynamics*, vol. 29, no. 6, pp. 1367–1375, 2006.

[35] I. Park, K. Fujimoto, and D. Scheeres, "Effect of dynamical accuracy for uncertainty propagation of perturbed keplerian motion," *Journal of Guidance, Control, and Dynamics*, vol. 38, no. 12, pp. 2287–2300, 2015.

[36] I. Park, "Dynamical realism and uncertainty propagation," Ph.D. dissertation, University of Colorado Boulder, 2016.

[37] M. Majji, J. Junkins, and J. Turner, "A high order method for estimation of dynamic systems," *The Journal of the Astronautical Sciences*, vol. 56, no. 3, pp. 401–440, 2008.

[38] S. Lee, H. Lyu, and I. Hwang, "Analytical uncertainty propagation for satellite relative motion along elliptic orbits," *Journal of Guidance, Control, and Dynamics*, vol. 39, no. 7, pp. 1593–1601, 2016.

[39] P. Sengupta, S. Vadali, and K. Alfriend, "Second-order state transition for relative motion near perturbed, elliptic orbits," *Celestial Mechanics and Dynamical Astronomy*, vol. 97, pp. 101–129, 2007.

[40] P. Hawkes, *Modern map methods in particle beam physics*. Academic Press, 1999.

[41] R. Armellin, P. Di Lizia, F. Bernelli-Zazzera, and M. Berz, "Asteroid close encounters characterization using differential algebra: the case of apophis," *Celestial Mechanics and Dynamical Astronomy*, vol. 107, pp. 451–470, 2010.

[42] M. Valli, R. Armellin, P. Di Lizia, and M. Lavagna, "Nonlinear mapping of uncertainties in celestial mechanics," *Journal of Guidance, Control, and Dynamics*, vol. 36, no. 1, pp. 48–63, 2013.

[43] N. Wiener, "The homogeneous chaos," *American Journal of Mathematics*, vol. 60, no. 4, pp. 897–936, 1938.

[44] B. Jones, A. Doostan, and G. Born, "Nonlinear propagation of orbit uncertainty using non-intrusive polynomial chaos," *Journal of Guidance, Control, and Dynamics*, vol. 36, no. 2, pp. 430–444, 2013.

[45] B. Jones and A. Doostan, "Satellite collision probability estimation using polynomial chaos expansions," *Advances in Space Research*, vol. 52, no. 11, pp. 1860–1875, 2013.

[46] J. Feldhacker, B. Jones, A. Doostan, and J. Hampton, "Reduced cost mission design using surrogate models," *Advances in Space Research*, vol. 57, no. 2, pp. 588–603, 2016.

[47] A. Prabhakar, J. Fisher, and R. Bhattacharya, "Polynomial chaos-based analysis of probabilistic uncertainty in hypersonic flight dynamics," *Journal of guidance, control, and dynamics*, vol. 33, no. 1, pp. 222–234, 2010.

[48] P. Dutta and R. Bhattacharya, "Nonlinear estimation of hypersonic state trajectories in bayesian framework with polynomial chaos," *Journal of guidance, control, and dynamics*, vol. 33, no. 6, pp. 1765–1778, 2010.

[49] F. Xiong, S. Chen, and Y. Xiong, "Dynamic system uncertainty propagation using polynomial chaos," *Chinese Journal of Aeronautics*, vol. 27, no. 5, pp. 1156–1170, 2014.

[50] A. Doostan, A. Validi, and G. Iaccarino, "Non-intrusive low-rank separated approximation of high-dimensional stochastic models," *Computer Methods in Applied Mechanics and Engineering*, vol. 263, pp. 42–55, 2013.

[51] A. Narayan, C. Gittelson, and D. Xiu, "A stochastic collocation algorithm with multifidelity models," *SIAM Journal on Scientific Computing*, vol. 36, no. 2, pp. A495–A521, 2014.

[52] B. Jones and R. Weisman, "Multi-fidelity orbit uncertainty propagation," *Acta Astronautica*, vol. 155, pp. 406–417, 2019.

[53] B. Jones, E. Delande, E. Zucchelli, and M. Jah, "Multi-fidelity orbit uncertainty propagation with systematic errors," in *Proceedings of the 2019 Advanced Maui Optical and Space Surveillance Technologies Conference*, 2019.

[54] B. Peherstorfer, K. Willcox, and M. Gunzburger, "Survey of multifidelity methods in uncertainty propagation, inference, and optimization," *Siam Review*, vol. 60, no. 3, pp. 550–591, 2018.

[55] X. Zhu, A. Narayan, and D. Xiu, "Computational aspects of stochastic collocation with multifidelity models," *SIAM/ASA Journal on Uncertainty Quantification*, vol. 2, no. 1, pp. 444–463, 2014.

[56] D. Vallado, P. Crawford, R. Hujsak, and T. Kelso, "Revisiting spacetrack report# 3," in *Proceedings of the AIAA/AAS Astrodynamics Specialist Conference and Exhibit*, 2006.

[57] E. Zucchelli, E. Delande, B. Jones, and M. Jah, "Multi-fidelity orbit determination with systematic errors," *The Journal of the Astronautical Sciences*, vol. 68, pp. 695–727, 2021.

[58] A. Fossà, R. Armellin, E. Delande, M. Losacco, and F. Sanfedino, "Multifidelity orbit uncertainty propagation using taylor polynomials," in *Proceedings of the AIAA SciTech 2022 Forum*, 2022.

[59] B. Jia and M. Xin, "Incremental uncertainty propagation with multi-fidelity gaussian process," in *Proceedings of the AIAA SCITECH 2022 Forum*, 2022.

[60] T. Wolf, E. Zucchelli, and B. Jones, "Multi-fidelity uncertainty propagation for objects in cislunar space," in *Proceedings of the AIAA SCITECH 2022 Forum*, 2022.

[61] A. Fossà, R. Armellin, E. Delande, and F. Sanfedino, "Multifidelity orbit uncertainty propagation in the presence of process noise," in *Proceedings of the 2023 AAS/AIAA Astrodynamics Specialist Conference*, 08 2023.

[62] E. Kerr and N. Sanchez-Ortiz, "State of the art and future needs in conjunction analysis methods, processes and software," in *Proceedings of the 8th European Conference on Space Debris*, 2021, pp. 20–23.

[63] J. Li, Z. Yang, and Y. Luo, "A review of space-object collision probability computation methods," *Astrodynamics*, vol. 6, no. 2, pp. 95–120, 2022.

[64] W. De Vries and D. Phillion, "Monte carlo method for collision probability calculations using 3d satellite models," in *Proceedings of the Advanced Maui Optical and Space Surveillance Technologies Conference*, 2010.

[65] C. Sabol, C. Binz, A. Segerman, K. Roe, and P. Schumacher Jr, "Probability of collision with special perturbation dynamics using the monte carlo method," *Advances in the Astronautical Sciences*, vol. 142, pp. 1081–1094, 2011.

[66] I. Grande-Olalla, N. Sanchez-Ortiz, J. Antonio Pulido, and K. Merz, "Collision risk assesment and avoidance manoeuvres. new tool coram for esa," in *Proceedings of the 6th European Conference on Space Debris*, 2013.

[67] C. Yang and M. Kumar, "An adaptive monte carlo method for uncertainty forecasting in perturbed two-body dynamics," *Acta Astronautica*, vol. 155, pp. 369–378, 2019.

[68] J. Dolado, P. Legendre, R. Garmier, B. Reveling, and X. Pena, "Satellite collision probability computation for long term encounters," *Advances in the Astronautical Sciences*, vol. 142, 2011.

[69] R. Pastel, "Estimating satellite versus debris collision probabilities via the adaptive splitting technique," in *Proceedings of the International Conference on Computer modeling and simulation*, 2011.

[70] V. Vittaldev and R. Russell, "Space object collision probability via monte carlo on the graphics processing unit," *The Journal of the Astronautical Sciences*, vol. 64, pp. 285–309, 2017.

[71] B. Jones, A. Doostan, and G. Born, "Conjunction assessment using polynomial chaos expansions," in *Proceedings of the 23rd International Symposium and Space Flight Dynamics*. Jet Propulsion Laboratory Pasadena, 2012.

[72] R. Armellin, A. Morselli, P. Di Lizia, and M. Lavagna, "Rigorous computation of orbital conjunctions," *Advances in Space Research*, vol. 50, no. 5, pp. 527–538, 2012.

[73] A. Morselli, R. Armellin, P. Di Lizia, and F. Zazzera, "A high order method for orbital conjunctions analysis: Monte carlo collision probability computation," *Advances in Space Research*, vol. 55, no. 1, pp. 311–333, 2015.

[74] ——, "A high order method for orbital conjunctions analysis: sensitivity to initial uncertainties," *Advances in Space Research*, vol. 53, no. 3, pp. 490–508, 2014.

[75] F. Chan, *Spacecraft collision probability*. American Institute of Aeronautics and Astronautics, 2008.

[76] N. Adurthi and P. Singla, "Conjugate unscented transformation-based approach for accurate conjunction analysis," *Journal of Guidance, Control, and Dynamics*, vol. 38, no. 9, pp. 1642–1658, 2015.

[77] S. Zhang, T. Fu, D. Chen, and H. Cao, "Satellite instantaneous collision probability computation using equivalent volume cuboids," *Journal of Guidance, Control, and Dynamics*, vol. 43, no. 9, pp. 1757–1763, 2020.

[78] J. Foster and H. Estes, *A parametric analysis of orbital debris collision probability and maneuver rate for space vehicles*. NASA, National Aeronautics and Space Administration, Lyndon B. Johnson Space Center, 1992.

[79] S. Alfano, "A numerical implementation of spherical object collision probability," *The Journal of the Astronautical Sciences*, vol. 53, pp. 103–109, 2005.

[80] R. Patera, "General method for calculating satellite collision probability," *Journal of Guidance, Control, and Dynamics*, vol. 24, no. 4, pp. 716–722, 2001.

[81] K. Chan, "Short-term vs. long-term spacecraft encounters," in *AIAA/AAS astrodynamics specialist conference and exhibit*, 2004.

[82] X. Xu and Y. Xiong, "A method for calculating probability of collision between space objects," *Research in Astronomy and Astrophysics*, vol. 14, no. 5, pp. 601–609, 2014.

[83] R. Serra, D. Arzelier, M. Joldes, J. Lasserre, A. Rondepierre, and B. Salvy, "Fast and accurate computation of orbital collision probability for short-term encounters," *Journal of Guidance, Control, and Dynamics*, vol. 39, no. 5, pp. 1009–1021, 2016.

[84] R. García-Pelayo and J. Hernando-Ayuso, "Series for collision probability in short-encounter model," *Journal of Guidance, Control, and Dynamics*, vol. 39, no. 8, pp. 1904–1912, 2016.

[85] S. Alfano and D. Oltrogge, "Probability of collision: Valuation, variability, visualization, and validity," *Acta Astronautica*, vol. 148, pp. 301–316, 2018.

[86] R. Patera, "Satellite collision probability for nonlinear relative motion," *Journal of Guidance, Control, and Dynamics*, vol. 26, no. 5, pp. 728–733, 2003.

[87] V. Coppola *et al.*, "Including velocity uncertainty in the probability of collision between space objects," in *Proceedings of the 22nd AAS/AIAA Spaceflight Mechanics Meeting*, vol. 143. American Astronautical Soc, 2012, pp. 2159–2178.

[88] S. Alfano, "Addressing nonlinear relative motion for spacecraft collision probability," in *Proceedings of the AIAA/AAS astrodynamics specialist conference and exhibit*, 2006.

[89] L. Chen, X. Bai, Y. Liang, and K. Li, "Calculation of collision probability," *Orbital Data Applications for Space Objects*, pp. 135–183, 2017.

[90] V. Vittaldev and R. Russell, "Collision probability for space objects using gaussian mixture models," in *Proceedings of the 23rd AAS/AIAA Space Flight Mechanics Meeting*, vol. 148. Citeseer, 2013, pp. 2339–2358.

[91] K. DeMars, Y. Cheng, and M. Jah, "Collision probability with gaussian mixture orbit uncertainty," *Journal of Guidance, Control, and Dynamics*, vol. 37, no. 3, pp. 979–985, 2014.

[92] C. Shelton and J. Junkins, "Probability of collision between space objects including model uncertainty," *Acta Astronautica*, vol. 155, pp. 462–471, 2019.

[93] K. Fujimoto and D. Scheeres, "Tractable expressions for nonlinearly propagated uncertainties," *Journal of Guidance, Control, and Dynamics*, vol. 38, no. 6, pp. 1146–1151, 2015.

[94] Y. Khatri and D. Scheeres, "Nonlinear semi-analytical uncertainty propagation for conjunction analysis," *Acta Astronautica*, vol. 203, pp. 568–576, 2023.

[95] ——, "Hybrid method of uncertainty propagation for long-term conjunction analysis," in *Proceedings of the 33rd AAS/AIAA Space Flight Mechanics Meeting*, 2023, pp. 23–376.

[96] ——, "Hybrid method of uncertainty propagation for near-earth conjunction analysis," *Journal of Guidance, Control, and Dynamics*, pp. 1–14, 2024.

[97] Y. Khatri, "Semi-analytical uncertainty propagation and conjunction assessment," Ph.D. dissertation, University of Colorado at Boulder, 2023.

[98] M. Balducci and B. Jones, "Probability of collision estimation and optimization under uncertainty utilizing separated representations," *The Journal of the Astronautical Sciences*, vol. 67, no. 4, pp. 1648–1677, 2020.

[99] P. Maybeck, *Stochastic models, estimation, and control*. Academic press, 1982.

[100] S. Julier and J. Uhlmann, "New extension of the kalman filter to nonlinear systems," in *Proceedings of the SPIE*, vol. 3068, 1997, pp. 182–193.

[101] H. Sorenson and D. Alspach, "Recursive bayesian estimation using gaussian sums," *Automatica*, vol. 7, no. 4, pp. 465–479, 1971.

[102] M. Isabel Ribeiro, "Gaussian probability density functions: Properties and error characterization," *Institute for Systems and Robotics*, 2004.

[103] V. Vittaldev and R. Russell, "Multidirectional gaussian mixture models for nonlinear uncertainty propagation," *Computer Modeling in Engineering and Sciences*, vol. 111, pp. 83–117, 01 2016.

[104] K. Tuggle and R. Zanetti, "Automated splitting gaussian mixture nonlinear measurement update," *Journal of Guidance, Control, and Dynamics*, vol. 41, no. 3, pp. 725–734, 2018.

[105] H. Xie, T. Xue, W. Xu, G. Liu, H. Sun, and S. Sun, "Orbital uncertainty propagation based on adaptive gaussian mixture model under generalized equinoctial orbital elements," *Remote Sensing*, vol. 15, no. 19, 2023.

[106] M. Eldred, "Recent advances in non-intrusive polynomial chaos and stochastic collocation methods for uncertainty analysis and design," in *Proceedings of the 50th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference*, 2009.

[107] J. Feinberg and H. Langtangen, "Chaospy: An open source tool for designing methods of uncertainty quantification," *Journal of Computational Science*, vol. 11, pp. 46–57, 2015.

[108] K. DeMars, "Nonlinear orbit uncertainty prediction and rectification for space situational awareness," Ph.D. dissertation, University of Texas at Austin, 2010.

[109] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, "Scikit-learn: Machine learning in Python," *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.

[110] M. Hejduk, D. Snow, and L. Newman, "Satellite conjunction assessment risk analysis for "dilution region" events: issues and operational approaches," in *Proceedings of the Space Traffic Management Conference*, 2019.

[111] K. Binder and D. Heermann, *Monte Carlo Simulation in Statistical Physics*. Springer-Verlag, 2010.

[112] D. Hall, S. Casali, L. Johnson, B. Skrehart, and L. Baars, "High fidelity collision probabilities estimated using brute force monte carlo simulations," in *Proceedings of the 2018 AAS/AIAA Astrodynamics Specialist Conference*, 2018.

[113] L. Baars, D. Hall, and S. Casali, "Assessing geo and leo repeating conjunctions using high fidelity brute force monte carlo simulations," in *Proceedings of the 2019 AAS/AIAA Astrodynamics Specialist Conference*, 2019.

[114] E. Denenberg, "Satellite closest approach calculation through chebyshev proxy polynomials," *Acta Astronautica*, vol. 170, pp. 55–65, 2020.

[115] A. Fielding, S. Wozniak, and M. Jah, "Privateer wayfinder," 2022. [Online]. Available: https://wayfinder.privateer.com/

[116] C. Acton, "Ancillary data services of nasa's navigation and ancillary information facility," *Planetary and Space Science*, vol. 44, no. 1, pp. 65–70, 1996.

[117] C. Acton, N. Bachman, B. Semenov, and E. Wright, "A look towards the future in the handling of space science mission geometry," *Planetary and Space Science*, vol. 150, pp. 9–12, 2018.

[118] Delft High Performance Computing Centre, "DelftBlue Supercomputer (Phase 2)," 2024. [Online]. Available: https://www.tudelft.nl/dhpc/ark:/44463/DelftBluePhase2

[119] V. Vittaldev, "Uncertainty propagation and conjunction assessment for resident space objects," Ph.D. dissertation, University of Texas at Austin, 2015.

[120] K. Vanslette, A. Lay, D. Kusterer, and K. Schroeder, "Rapid and uncertainty quantified orbital propagation using uncertainty-aware ai," in *Proceedings of the Advanced Maui Optical and Space Surveillance Technologies Conference*, 2024.

[121] Z. Zhang, Y. Shi, and H. Han, "Multivariate attention-based orbit uncertainty propagation and orbit determination method for earth–jupiter transfer," *Applied Sciences*, vol. 14, no. 10, 2024.

[122] X. Zhou, D. Qiao, and X. Li, "Neural network-based method for orbit uncertainty propagation and estimation," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 60, no. 1, pp. 1176–1193, 2023.

[123] C. Acton, N. Bachman, A. Bailey, M. Barnes, B. Semenov, and E. Wright, "An overview of reference frames and coordinate systems in the spice context." 4 2023. [Online]. Available: https://naif.jpl.nasa.gov/pub/naif/toolkit_docs/Tutorials/pdf/individual_docs/17_frames_and_coordinate_systems.pdf

[124] K. Wakker, *Fundamentals of astrodynamics*. Institutional Repository Library Delft University of Technology, 2015.

[125] D. Vallado, *Fundamentals of astrodynamics and applications*. Springer Science & Business Media, 2001, vol. 12.

[126] G. Hintz, "Survey of orbit element sets," *Journal of guidance, control, and dynamics*, vol. 31, no. 3, pp. 785–790, 2008.

[127] L. Iess, W. Folkner, D. Durante, M. Parisi, Y. Kaspi, E. Galanti, T. Guillot, W. Hubbard, D. Stevenson, J. Anderson, D. Buccino, L. Casajus, A. Milani, R. Park, P. Racioppa, D. Serra, P. Tortora, M. Zannoni, H. Cao, R. Helled, J. Lunine, Y. Miguel, B. Militzer, S. Wahl, J. Connerney, S. Levin, and S. Bolton, "Measurement of jupiter's asymmetric gravity field," *Nature*, vol. 555, no. 7695, pp. 220–222, 2018.

A

Algorithms

Algorithm 1: Identification of r_{mf} important samples [52].

Data: $\boldsymbol{X}^L(\Xi)$ and r_{mf}
Result: ordered list of indices $\underline{\mathscr{I}}$ corresponding to important samples and decomposition $\boldsymbol{C}_H$
$\boldsymbol{V} = \boldsymbol{X}^L$;
$[\boldsymbol{w}_c]_\ell = \boldsymbol{V}_{:,\ell} \cdot \boldsymbol{V}_{:,\ell}$ for $\ell = 1, \ldots, n_{state}$;
$\underline{\mathscr{I}} = \emptyset$;
$\boldsymbol{C}_H = \boldsymbol{0} \in \mathbb{R}^{n_{state} \times r_{mf}}$;
for $n = 1, ..., r_{mf}$ **do**
  $e_c, i = \arg\max_{\ell \in \{n, ..., n_{state}\}} \left([\boldsymbol{w}_c]_\ell\right)$;
  if $e_c < \epsilon_{num}$ **then**
    $n = n + 1$;
    stop
  end
  $\underline{\mathscr{I}} = \underline{\mathscr{I}} \cup \{i\}$;
  Swap the n'th and i'th columns in $\boldsymbol{V}$;
  Swap the n'th and i'th values in $\boldsymbol{w}_c$;
  $[\boldsymbol{C}_H]_{n,n} = \sqrt{[\boldsymbol{w}_c]_n}$;
  $[C_H]_{t,n} = \left(\boldsymbol{V}_{:,t}^T \boldsymbol{V}_{:,n} - \sum_{j=1}^{n-1} [\boldsymbol{C}_H]_{t,j} [\boldsymbol{C}_H]_{n,j}\right) / [C_H]_{n,n}$ for $t = n+1, \ldots, n_{state}$;
  $[\boldsymbol{w}_c]_t = [\boldsymbol{w}_c]_t - [\boldsymbol{C}_H]_{t,n}^2$, for $t = n+1, \ldots, n_{state}$;
end
$\boldsymbol{C}_H = [\boldsymbol{C}_H]_{1:n,1:n}$;

Algorithm 2: Determination of true time of closest approach, using bisection.

Data: $\boldsymbol{x}_P\left(t_{TCA,mean}\right)$, $\boldsymbol{x}_S\left(t_{TCA,mean}\right)$, t_{window}, N_{bisect}, ϵ_{TCA}, ϵ_{DCA}
Result: Time $t_{TCA,true}$ and distance DCA_{true} at the actual point of closest approach.
$\Delta\boldsymbol{t} = [-t_{window}, +t_{window}]$;
$\mathscr{I} = [0, 1]$;
while $\max\left(\left|[\boldsymbol{d}_{rel}]_{2:n} - [\boldsymbol{d}_{rel}]_{1:n-1}\right|\right) < \epsilon_{DCA}$ *and* $\max\left(\left|[\Delta\boldsymbol{t}]_{2:n} - [\Delta\boldsymbol{t}]_{1:n-1}\right|\right) < \epsilon_{DCA}$ **do**
  Replace $\Delta\boldsymbol{t}$ with the combined set of $[\Delta\boldsymbol{t}]_{j-1:j+1}$ for each j in $\mathscr{I}$;
  Expand $\Delta\boldsymbol{t}$ with N_{bisect} evenly spaced points in $\langle t_j, t_{j+1}\rangle$ for each t_j in $[\Delta\boldsymbol{t}]_{1:n-1}$;
  Determine $\boldsymbol{X}_P^k(\Delta\boldsymbol{t})$ using Algorithm 3 with $\boldsymbol{x}_{t_0} = \boldsymbol{x}_P\left(t_{TCA,mean}\right)$ and $\Delta\boldsymbol{t} = \Delta\boldsymbol{t}$;
  Determine $\boldsymbol{X}_S^k(\Delta\boldsymbol{t})$ using Algorithm 3 with $\boldsymbol{x}_{t_0} = \boldsymbol{x}_S\left(t_{TCA,mean}\right)$ and $\Delta\boldsymbol{t} = \Delta\boldsymbol{t}$;
  $\boldsymbol{d}_{rel}(\Delta\boldsymbol{t}) = |\boldsymbol{r}_P(\Delta\boldsymbol{t}) - \boldsymbol{r}_S(\Delta\boldsymbol{t})|$;
  $\mathscr{I} = \text{indmin}(\boldsymbol{d}_{rel})$;
end
$t_{TCA,true} = \Delta\boldsymbol{t}[\mathscr{I}[0]]$;
$DCA_{true} = \boldsymbol{d}_{rel}[\mathscr{I}[0]]$;

Algorithm 3: Parallel propagation of states with Kepler 2-body dynamics.

Data: $\boldsymbol{x}_{t_0}$, $\Delta\boldsymbol{t}$, and ϵ_θ
Result: Propagated Cartesian state vectors at the relative time epochs $\boldsymbol{X}^k(\Delta\boldsymbol{t})$
Calculate $\left[a_{t_0}, e_{t_0}, i_{t_0}, \omega_{t_0}, \Omega_{t_0}, \theta_{t_0}\right]$ using Equations B.2 to B.7 with $\boldsymbol{x}_{t_0}$;
$E_{t_0} = 2\arctan\left(\sqrt{\frac{1-e_{t_0}}{1+e_{t_0}}}\tan\left(\frac{\theta_{t_0}}{2}\right)\right)$;
$M_{t_0} = E_{t_0} - e_{t_0}\sin\left(E_{t_0}\right)$; `/* M and E must be expressed in [rad] */`
$M(\Delta\boldsymbol{t}) = M_{t_0} + \sqrt{\frac{\mu}{a_{t_0}^3}}(\Delta\boldsymbol{t} - t_0)$;
$E(\Delta\boldsymbol{t}) = M(\Delta\boldsymbol{t})$;
while $\max\left(\left|E(\Delta\boldsymbol{t}) - M(\Delta\boldsymbol{t}) - e_{t_0}\sin\left(E(\Delta\boldsymbol{t})\right)\right|\right) > \epsilon_\theta$ **do**
  $E(\Delta\boldsymbol{t}) = M(\Delta\boldsymbol{t}) + e_{t_0}\sin\left(E(\Delta\boldsymbol{t})\right)$;
end
$\theta(\Delta\boldsymbol{t}) = 2\arctan\left(\sqrt{\frac{1+e_{t_0}}{1-e_{t_0}}}\tan\left(\frac{E(\Delta\boldsymbol{t})}{2}\right)\right)$;
Calculate R_{ref}^{cart} using Equation B.8 with i_{t_0}, ω_{t_0} and Ω_{t_0};
Calculate $\boldsymbol{X}^k(\Delta\boldsymbol{t})$ using Equations B.9 and B.10 with a_{t_0}, e_{t_0} and $\theta(\Delta\boldsymbol{t})$;

B

Coordinate transformations

The various different coordinate systems and frames used to represent states in this study are clarified in the following sections, as well as the methods used to convert between them. The main coordinate system used in this study is that of the Earth Centred Inertial (ECI) J2000 frame with Cartesian coordinates. Thus, all state representations used in different frames and coordinate systems are obtained by converting to and from the ECI J2000 frame / Cartesian coordinates.

A state vector of a satellite in Cartesian coordinates is represented by:

$$\boldsymbol{x} = \begin{bmatrix} \boldsymbol{r} \\ \boldsymbol{v} \end{bmatrix} = \begin{bmatrix} \boldsymbol{r}_x \\ \boldsymbol{r}_y \\ \boldsymbol{r}_z \\ \boldsymbol{v}_x \\ \boldsymbol{v}_y \\ \boldsymbol{v}_z \end{bmatrix} \tag{B.1}$$

The ECI J2000 coordinate frame is defined by the equatorial and ecliptic plane of the Earth at J2000 (2000-01-01 12:00:00 TDB) [123], as shown in Figure B.1. The x vector points towards the intersection of both planes, the z vector is the normal to the equatorial plane, and the y vector completes the right handed coordinate system with $y = z \times x$.

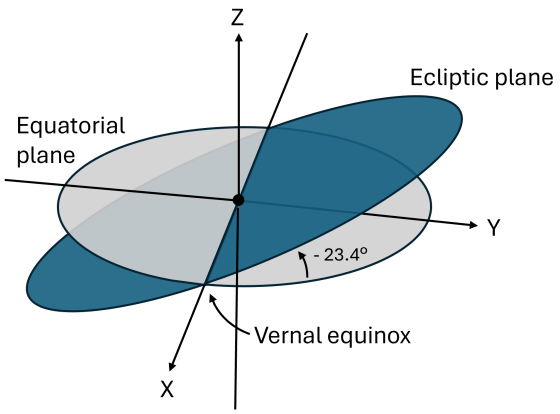


Figure B.1: *The ECI J2000 coordinate frame, based on [123].*

B.1. Keplerian elements

In this study, the Keplerian elements used are defined by $[a, e, i, \omega, \Omega, \theta]$. Each element is defined in Table B.1, of which the angles are shown in Figure B.2.

Symbol	Keplerian element
a	Semi major axis
e	Eccentricity
i	Inclination
ω	Argument of periapsis
Ω	Longitude of ascending node
θ	True anomaly

Table B.1: *The Keplerian elements.*

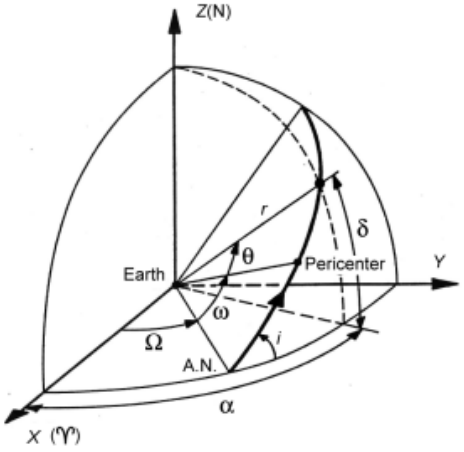


Figure B.2: *Keplerian elements shown in the ECI J2000 frame [124].*

To convert from Cartesian coordinates to Keplerian elements, the following equations can be used [125]:

$$r = |\boldsymbol{r}| \qquad v = |\boldsymbol{v}| \qquad \boldsymbol{h} = \boldsymbol{r} \times \boldsymbol{v} \qquad \boldsymbol{\eta} = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \times \boldsymbol{h} \tag{B.2}$$

$$a = \frac{1}{\frac{2}{r} - \frac{v^2}{\mu}} \qquad \boldsymbol{e} = \frac{\left(v^2 - \frac{\mu}{r}\right)\boldsymbol{r} - (\boldsymbol{r} \cdot \boldsymbol{v})\,\boldsymbol{v}}{\mu} \tag{B.3}$$

$$\cos(i) = \frac{\boldsymbol{h}_z}{|\boldsymbol{h}|} \tag{B.4}$$

$$\cos(\Omega) = \frac{\boldsymbol{\eta}_x}{|\boldsymbol{\eta}|} \qquad \text{if}\left(\boldsymbol{\eta}_y < 0\right) : \Omega = 360° - \Omega \tag{B.5}$$

$$\cos(\omega) = \frac{\boldsymbol{\eta} \cdot \boldsymbol{e}}{|\boldsymbol{\eta}||\boldsymbol{e}|} \qquad \text{if}\,(\boldsymbol{e}_z < 0) : \omega = 360° - \omega \tag{B.6}$$

$$\cos(\theta) = \frac{\boldsymbol{e} \cdot \boldsymbol{r}}{|\boldsymbol{e}||\boldsymbol{r}|} \qquad \text{if}\,(\boldsymbol{r} \cdot \boldsymbol{v} < 0) : \theta = 360° - \theta \tag{B.7}$$

To convert back from Keplerian elements to Cartesian coordinates, a reference frame is introduced, by aligning the x-axis and y-axis in the orbital plane, with the x-axis pointing towards perigee, as shown in Figure B.3.

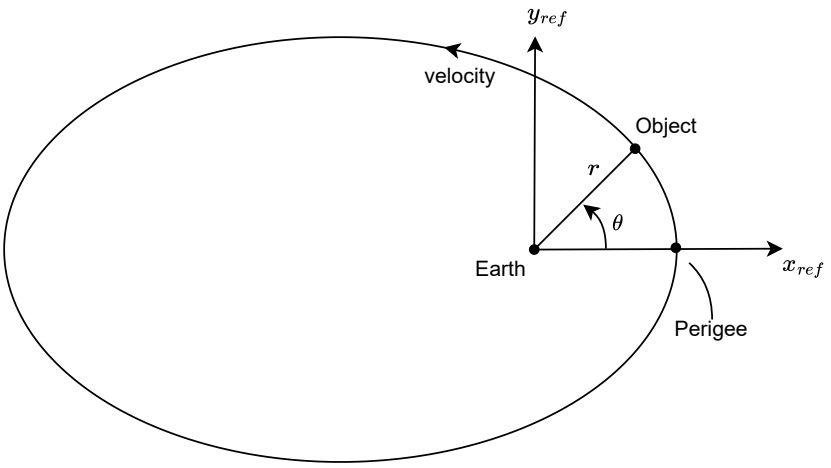


Figure B.3: *Illustration of the reference frame aligned along the orbital plane, used to transform from Keplerian elements to Cartesian coordinates, based on [124]*

Using the radius and the angular momentum, the Keplerian elements can be expressed in the coordinates of the reference frame, after which a composed rotation matrix is used to obtain the position and velocity vector in Cartesian coordinates in the ECI J2000 frame. By combining both these processes, the following equations can be used to transform Keplerian elements to Cartesian coordinates [124]:

$$R_{ref}^{cart} = \begin{bmatrix} \cos(\Omega) & -\sin(\Omega) & 0 \\ \sin(\Omega) & \cos(\Omega) & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(i) & -\sin(i) \\ 0 & \sin(i) & \cos(i) \end{bmatrix} \begin{bmatrix} \cos(\omega) & -\sin(\omega) & 0 \\ \sin(\omega) & \cos(\omega) & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{B.8}$$

$$r = \frac{a\left(1-e^2\right)}{1+e\cos(\theta)} \qquad H = \sqrt{\mu a\left(1-e^2\right)} \tag{B.9}$$

$$\begin{bmatrix} \boldsymbol{r}_x \\ \boldsymbol{r}_y \\ \boldsymbol{r}_z \end{bmatrix} = R_{ref}^{cart} \begin{bmatrix} r\cos(\theta) \\ r\sin(\theta) \\ 0 \end{bmatrix} \qquad \begin{bmatrix} \boldsymbol{v}_x \\ \boldsymbol{v}_y \\ \boldsymbol{v}_z \end{bmatrix} = \frac{\mu}{H} R_{ref}^{cart} \begin{bmatrix} -\sin(\theta) \\ e+\cos(\theta) \\ 0 \end{bmatrix} \tag{B.10}$$

B.2. Modified Equinoctial Elements

The set of Modified Equinoctial Elements (MEE) used in this study is set k in [126], where the state vector is represented by: $[p, f, g, h, k, L]$. The MEE can be obtained from the Keplerian elements and vice versa, using the following equations [126]:

$$p = a\left(1-e^2\right) \tag{B.11}$$
$$f = e * \cos(\omega + I\Omega) \tag{B.12}$$
$$g = e * \sin(\omega + I\Omega) \tag{B.13}$$
$$h = \tan^I(i/2) * \cos(\Omega) \tag{B.14}$$
$$k = \tan^I(i/2) * \sin(\Omega) \tag{B.15}$$
$$L = \omega + I\Omega + \theta \tag{B.16}$$

Where I is the retrograde factor, which is set to $I = +1$ and $I = -1$ for prograde and retrograde orbits respectively. To convert from a Cartesian state to MEE and vice versa, the conversions between a Cartesian state and Keplerian elements can be used together with the conversions between Keplerian elements and MEE.

B.3. RSW frame

The Radial (R), Along-track (S) and Cross-track(W) coordinate frame, also known as the RIC frame or RTN frame, is centred on a satellite's position and oriented with the R-S frame in its orbital plane. It can be used to represent relative coordinates between multiple satellites, or between sampled state vectors and the mean state vector in the case of probability distributions. An illustration of the coordinate frame is shown in Figure B.4.

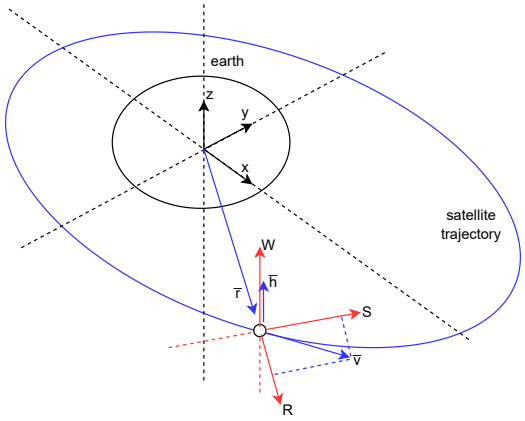


Figure B.4: *The RSW coordinate frame shown together with the ECI J2000 coordinate frame.*

To convert between the ECI and the RSW frame requires a translational and a rotational transformation. The coordinates of the basis vectors of the RSW frame can be expressed in the ECI frame by:

$$\hat{\boldsymbol{R}} = \overline{\boldsymbol{r}} \qquad \hat{\boldsymbol{S}} = \overline{\boldsymbol{h}} \times \overline{\boldsymbol{r}} \qquad \hat{\boldsymbol{W}} = \overline{\boldsymbol{h}} \tag{B.17}$$

Where $\overline{\boldsymbol{h}} = \overline{\boldsymbol{r}} \times \overline{\boldsymbol{v}}$ and $\overline{\boldsymbol{r}}$ and $\overline{\boldsymbol{v}}$ are the position and velocity vectors of the mean state, in the ECI frame.

The conversions of the position and velocity vectors between the frames are then given by:

$$\overline{\boldsymbol{\omega}} = \begin{bmatrix} 0 \\ 0 \\ \frac{|\overline{\boldsymbol{h}}|}{|\overline{\boldsymbol{r}}|^2} \end{bmatrix} \qquad R_{ECI}^{RSW} = \begin{bmatrix} \hat{R}_x & \hat{R}_y & \hat{R}_z \\ \hat{S}_x & \hat{S}_y & \hat{S}_z \\ \hat{W}_x & \hat{W}_y & \hat{W}_z \end{bmatrix} \qquad R_{RSW}^{ECI} = \begin{bmatrix} \hat{R}_x & \hat{S}_x & \hat{W}_x \\ \hat{R}_y & \hat{S}_y & \hat{W}_y \\ \hat{R}_z & \hat{S}_z & \hat{W}_z \end{bmatrix} \tag{B.18}$$

$$\boldsymbol{r}_{RSW} = R_{ECI}^{RSW}\left(\boldsymbol{r}_{ECI} - \overline{\boldsymbol{r}}_{ECI}\right) \qquad \boldsymbol{r}_{ECI} = R_{RSW}^{ECI}\,\boldsymbol{r}_{ECI} + \overline{\boldsymbol{r}}_{ECI} \tag{B.19}$$

$$\boldsymbol{v}_{RSW} = R_{ECI}^{RSW}\left(\boldsymbol{v}_{ECI} - \overline{\boldsymbol{v}}_{ECI}\right) - \overline{\boldsymbol{\omega}} \times \overline{\boldsymbol{r}}_{RSW} \qquad \boldsymbol{v}_{ECI} = R_{RSW}^{ECI}\left(\boldsymbol{v}_{RSW} + \overline{\boldsymbol{\omega}} \times \boldsymbol{r}_{RSW}\right) + \overline{\boldsymbol{v}}_{ECI} \tag{B.20}$$

Where $\boldsymbol{r}_{RSW}$ and $\boldsymbol{v}_{RSW}$ represent the position and velocity vector in the RSW frame, $\boldsymbol{r}_{ECI}$ and $\boldsymbol{v}_{ECI}$ the position and velocity in the ECI frame, and $\overline{\boldsymbol{r}}_{ECI}$ and $\overline{\boldsymbol{v}}_{ECI}$ the position and velocity vector of the mean state around which the RSW frame is centred.

B.4. Conjunction Plane

The conjunction-plane coordinate frame is defined in Cartesian coordinates, centred on the primary object, and aligned such that the z-axis lies along the relative velocity vector. The choice of x- and y-axis of the conjunction plane remain free, but are chosen in this study such that the x-axis of the conjunction plane aligns as close as possible to the z-axis of the ECI frame in Cartesian coordinates [112].

To determine the relative position of the secondary object with respect to the primary object in the conjunction plane, the following transformation can be used:

$$\boldsymbol{r}_{rel,cp} = \begin{bmatrix} \hat{\boldsymbol{X}}_x & \hat{\boldsymbol{X}}_y & \hat{\boldsymbol{X}}_z \\ \hat{\boldsymbol{Y}}_x & \hat{\boldsymbol{Y}}_y & \hat{\boldsymbol{Y}}_z \\ \hat{\boldsymbol{Z}}_x & \hat{\boldsymbol{Z}}_y & \hat{\boldsymbol{Z}}_z \end{bmatrix} \times \boldsymbol{r}_{rel,ECI} \tag{B.21}$$

Where $\hat{\boldsymbol{X}}$, $\hat{\boldsymbol{Y}}$ and $\hat{\boldsymbol{Z}}$ are the unit vectors of the axes of the conjunction plane expressed in Cartesian coordinates of the ECI j2000 frame, which can be found by [112]:

$$\hat{\boldsymbol{Z}} = \frac{\boldsymbol{v}_S - \boldsymbol{v}_P}{|\boldsymbol{v}_S - \boldsymbol{v}_P|} \qquad \hat{\boldsymbol{X}} = \frac{\hat{\boldsymbol{Z}} \times \hat{\boldsymbol{K}}}{\left|\hat{\boldsymbol{Z}} \times \hat{\boldsymbol{K}}\right|} \qquad \hat{\boldsymbol{Y}} = \hat{\boldsymbol{Z}} \times \hat{\boldsymbol{X}} \tag{B.22}$$

C

Integrator Analysis

This appendix describes the process of selecting the integrator and propagator settings in Tudat, to be used for each test case scenario. This process is done in several steps, which follow the same structure for each test case scenario.

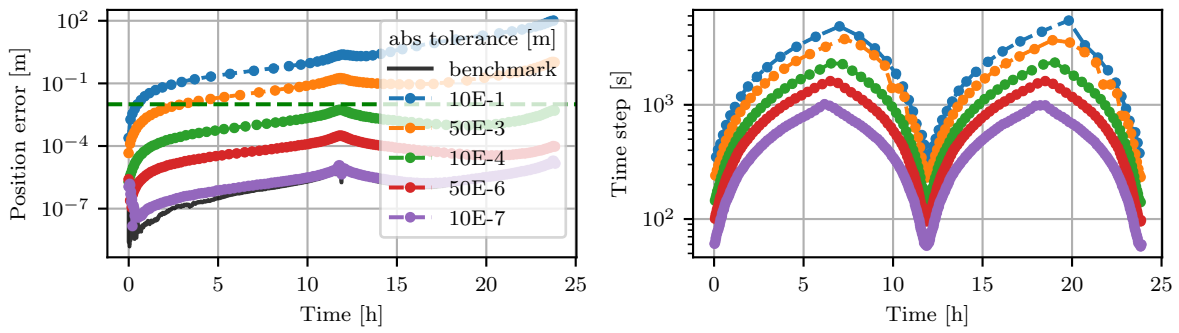
First, a benchmark propagation is created, with a fixed step integrator and a spherical harmonic gravitational acceleration from the Earth with degree and order 2. The error of this benchmark is analysed by comparing a benchmark propagation with a second propagation that uses a time step twice as small. Under the assumption that the numerical error is the dominant source in the integration error, which grows when the time step is increased, the second propagation can be considered to have a smaller integration error and can be used as a truth to compare the benchmark propagation to. When the time step becomes small, rounding errors become a larger source of the integration error, which show erratic behaviour. By choosing a time step from a range for which the integration error increases linearly with increasing time steps, it is ensured that the numerical error is the dominant source, such that the maximum position error with respect to the smaller time step can be used as an error estimate for the chosen time step. In this way, several time steps are analysed, and a benchmark time step is chosen for each scenario.

With the benchmark propagation, various integrators can be compared and analysed for their integration errors. A number of multistage integrators, both with fixed step and variable step, are analysed with various settings, by ranging the fixed time step from large to small, or ranging the absolute tolerance levels ϵ_a from large to small with a relative tolerance of 10^{-13}. The integration error is deduced by comparing the propagation to the earlier obtained benchmark, and the number of function evaluations is recorded, to use as a direct estimation of the relative propagation time compared to other integrators. From this, an integrator and its settings are selected.

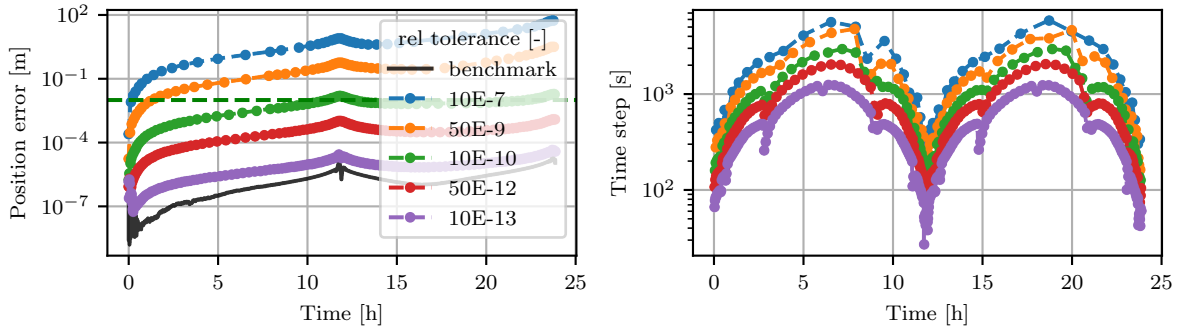
In Tudat, the absolute and relative tolerances are combined at each step by:

$$\epsilon_{int} = \epsilon_a + \boldsymbol{x}_t \cdot \epsilon_r \tag{C.1}$$

where ϵ_{int} is the combined tolerance applied in the integration step, ϵ_a and ϵ_r are the absolute and relative tolerances, and $\boldsymbol{x}_t$ is the state vector at the current epoch. To allow for an analysis of tolerance levels with a single changing variable, it is decided to keep one of the tolerances fixed at a negligibly small value, while varying the other tolerance. To simplify the analysis further, only the absolute tolerance is varied, while keeping the relative tolerance at 10^{-13} for all cases. This choice is made based on the smoother and more predictable behaviour of the time step over time, when using the absolute tolerances rather than relative tolerances. An example of this is demonstrated for one of the elliptical cases, shown in Figure C.1. It can be seen that for tolerance settings that result in similar levels of position error over time, varying the absolute tolerance produces more predictable time steps. Thus, in all further analyses, the tolerance used is the absolute tolerance, where the relative tolerance is always set at 10^{-13}, such that it does not influence the final tolerance.



(a) *Varying the absolute tolerance, with a relative tolerance of* 10^{-13}.



(b) *Varying the relative tolerance, with a relative tolerance of* 10^{-13}.

Figure C.1: *The position error (left) and the time step (right) over time, for various values of absolute or relative tolerance, using the variable step rkdp87 integrator and Cowell propagator for the Cosmos scenario.*

Additionally, during the analysis of the integrators, it is found that the variable step integrators induce much higher values of rounding errors in the integrator error, compared to the fixed step integrators or the benchmark propagation, such that the variable step integrators are unable to reach the desired error level for the various test case scenarios. The rounding errors are found to be caused by an accumulation of floating point errors of the time steps on to the determined time epochs. While the floating point errors are small, approximately $10^{-8}s$, the accumulation of these after 10^3 time steps reaches $10^{-5}s$, which can be translated into a position error of approximately $1m$, since the orbital velocity reaches $10^4 m/s$ for LEO objects. To solve this problem, the pre-developed propagation software of Tudat is adjusted for the variable step integrators, such that at every time epoch, the new time step is converted to the nearest integer, before being used to calculate the next time epoch. Since the time steps used in the variable step integrators are larger than 10^2 for all scenarios, this change has negligible effects on the time steps, which means that neither the computation times nor the numerical errors are affected by these changes, while the high accumulation of rounding errors is removed.

Finally, with the integrator and propagator selected, the various acceleration and environment models are analysed. A nominal model is created with all available accelerations in Tudat included, shown in Table C.1, and compared to various propagations in which a single acceleration is removed or reduced in complexity, one at a time. The effect of removing/reducing these accelerations is then compared to a position error threshold, and all accelerations that have an effect small enough are excluded from the final model. In this way, the final model represents an accuracy set by the final error threshold, often set to $1m$ in this study. The following changes are made to the accelerations: The solar radiation pressure, atmospheric drag and relativistic correction, are removed. The central body gravity is reduced to degree and order 2x2, and to a point mass gravity (both with the 100x100 model as a baseline). The 3^{rd} body spherical harmonics are reduced to 3^{rd} body point mass perturbations, and the 3^{rd} body point mass perturbations are removed. To reduce the complexity of the environment models, the rotation models (of the central body as well as of 3^{rd} bodies with spherical harmonics included in the accelerations) are changed to a linear rotation model, using the orientation and rotation rate of the body at t_0, which is retrieved from SPICE once per propagation. The 3^{rd} body position models are changed to Keplerian 2-body orbits around their respective central bodies, based on their state vectors at t_0, retrieved from SPICE once at the start of the propagation. These models replace the complex models, where the orientations/positions are retrieved from SPICE at each time epoch for which the accelerations are required.

Acceleration type	
Central body gravity (Earth)	100 x 100 (GOCO05c)
3rd body spherical harmonics (100 x 100)	Moon (gggrx1200), Mars (jgmro120d), Venus (shgj180u), Mercury (jgmess160a)
3rd body spherical harmonics (Zonal coefficients of degree 8)	Jupiter (Model described in [127])
3rd body perturbations (point mass)	Sun, Saturn, Neptune, Uranus, Io, Europa, Callisto, Ganymede, Amalthea, Phobos, Deimos, Mimas, Enceladus, Tethys, Dione, Rhea, Titan, Hyperion, Iapetus
Solar radiation pressure	C_R = 1.5 (Cannonball)
Relativistic Correction	Sun (Schwarzschild)
Atmospheric drag	C_D = 2.0 (US76)
Environment models	
Central body rotation model	Retrieved every epoch from Spice
3rd body rotation model (Moon, Mars, Venus, Mercury, Jupiter)	Retrieved every epoch from Spice
3rd body position model	Retrieved every epoch from Spice

Table C.1: *Propagation settings used for the full model in Tudat.*

In the following sections, the results and selections of these settings and models are shown per test case scenario of the uncertainty propagation comparison analysis, and for the long versions of the high- and low-velocity scenario of the conjunction assessment study, following the structure outlined above. To ensure the benchmark error and integrator error do not influence the results of the comparisons further on, the integrator error threshold is set to 2 orders of magnitude smaller than the final model error threshold, and similarly the benchmark error is set to two orders of magnitude below the integrator error threshold. The scenarios of the conjunction assessment study include two space objects for each scenario. The effects of the integrator settings and propagation models are shown for both objects, where the final chosen settings are the same for both objects, and chosen based on a worst case scenario. The settings of the short versions of both scenarios are taken to be identical to those of the long versions. The low-fidelity settings for the conjunction assessment scenarios are determined once for all scenarios simultaneously and are described in Chapter 7.

C.1. Velox Scenario

For the Velox scenario, the final error is chosen to be below $1m$, resulting in a maximum integrator error of $10^{-2}m$ and a benchmark error of $10^{-4}m$.

The benchmark analysis is shown in Figure C.2, from which a benchmark time step of $20s$ is selected.

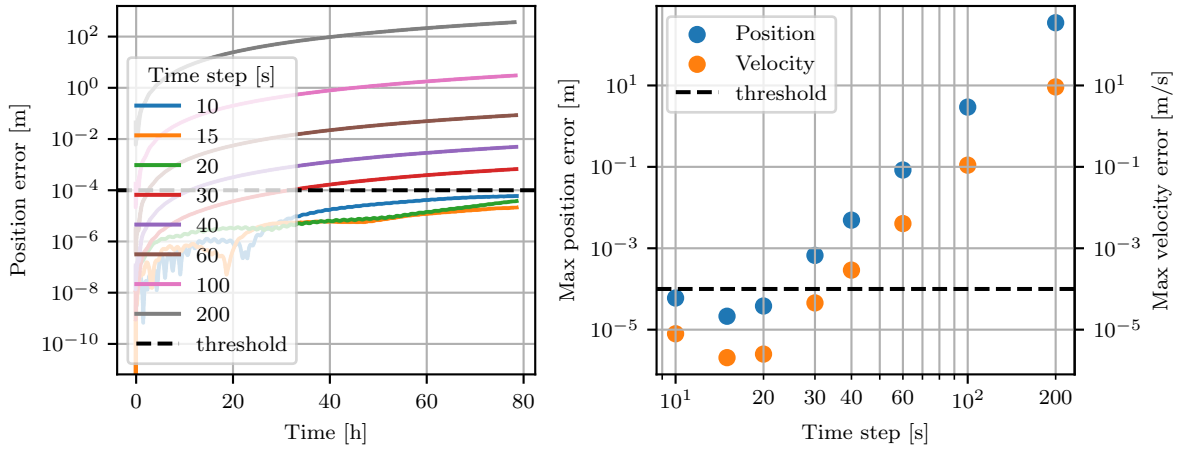


Figure C.2: *Benchmark error analysis for the Velox scenario.*

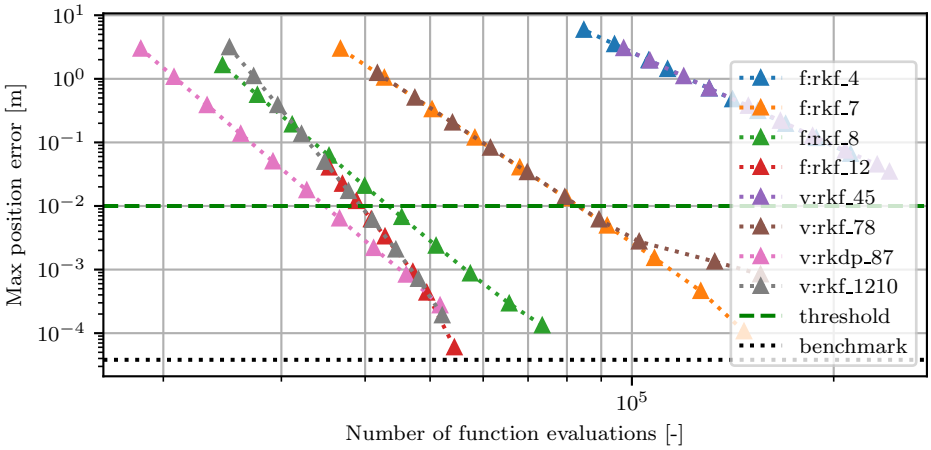


Figure C.3: *Integrator analysis for the Cowell propagator for the Velox scenario.*

The integrator analysis for the Cowell propagator is shown in Figure C.3. The chosen integrator is the variable step rkdp87 integrator with an absolute tolerance of 2.0E-6. The position error over time for the variable step rkdp87 integrator is shown in Figure C.4

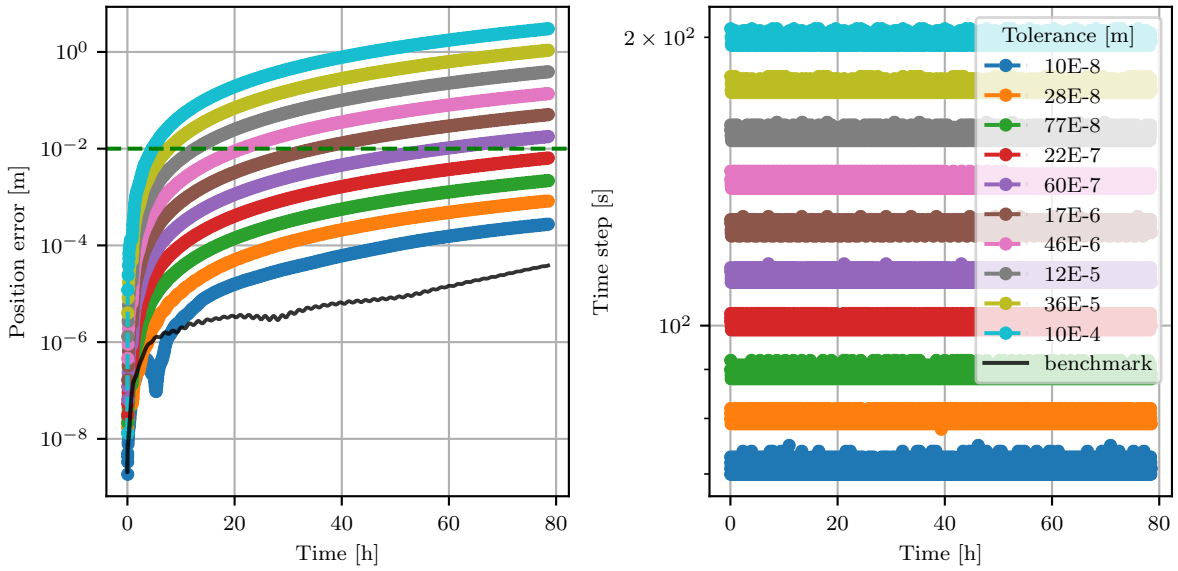


Figure C.4: *The position error (left) and the time step (right) over time for various absolute tolerances using the variable step rkdp87 integrator and Cowell propagator for the Velox scenario.*

The effects of removing/reducing accelerations are shown in Figure C.5.

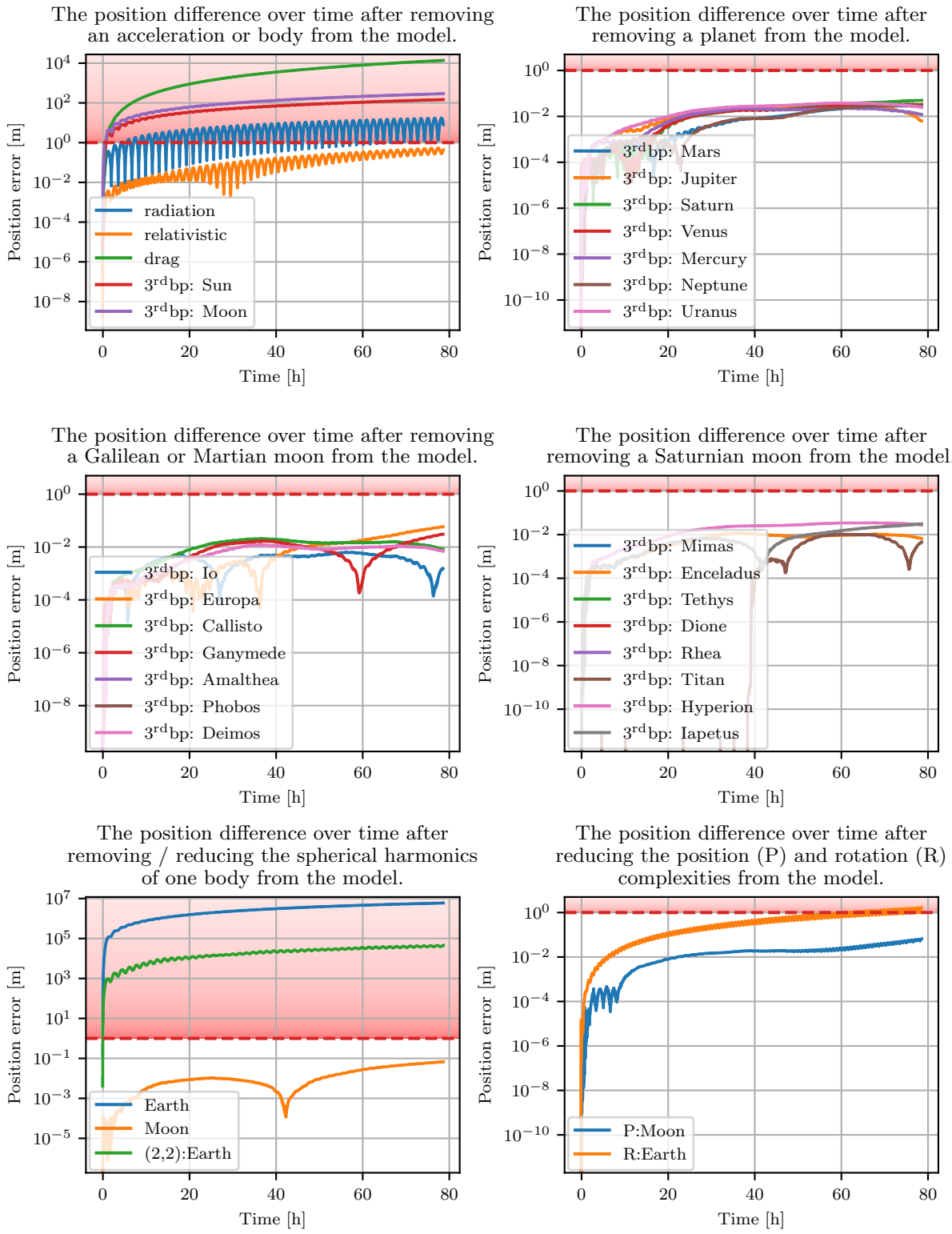


Figure C.5: *Acceleration model analysis of the Velox scenario.*

For this scenario, all 3rd body perturbations can be removed, except for the Sun and Moon, where for the Moon, the spherical harmonics can be reduced to a point mass. Both the atmospheric drag and solar radiation are kept, but the relativistic correction is removed from the final model. Furthermore, since the change from Earths 100x100 spherical harmonics to 2x2 or to a point mass gravity both result in high position errors, the final model still contains the 100x100 spherical harmonics. Finally, both the position model of the Moon, as well as the rotation model of the Earth are reduced in complexity.

All changes for which the effects remain below $1m$ have been included in the final model, which is shown in Table C.2. Based on the effects shown in Figure C.5, a more lenient selection of accelerations and models is made for the low-fidelity model (resulting in higher propagation errors and lower computation times), which is shown in Table C.3.

Acceleration type	
Central body gravity (Earth)	100 x 100 (GOCO05c)
3rd body perturbations (point mass)	Sun, Moon
Solar radiation pressure	C_R = 1.5 (Cannonball)
Atmospheric drag	C_D = 2.0 (US76)
Environment models	
Central body rotation model	Constant rotation rate determined at t_0
3rd body position model (Moon)	Keplerian 2-body orbit determined at t_0
3rd body position model (Sun)	Retrieved every epoch from Spice
Propagator settings	
Integrator	variable rkdp87 (tolerance: 2E-6)
Propagator	Cowell

Table C.2: *Propagation settings used for the high-fidelity dynamics of the Velox scenario*

Acceleration type	
Central body gravity (Earth)	2 x 2 (GOCO05c)
Atmospheric drag	C_D = 2.0 (US76)
Environment models	
Central body rotation model	Constant rotation rate determined at t_0
Propagator settings	
Integrator	variable rkdp87 (tolerance: 2E-6)
Propagator	Cowell

Table C.3: *Propagation settings used for the low-fidelity dynamics of the Velox scenario*

C.2. Oneweb Scenario

For the Oneweb scenario, the final error is chosen to be below $1m$, resulting in a maximum integrator error of $10^{-2}m$ and a benchmark error of $10^{-4}m$.

The benchmark analysis is shown in Figure C.6, from which a benchmark time step of $25s$ is selected.

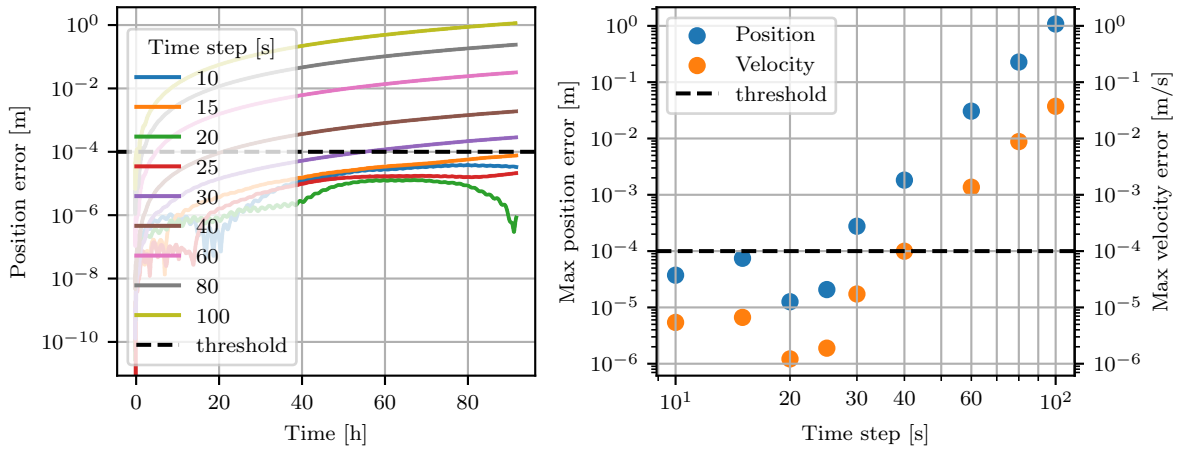


Figure C.6: *Benchmark error analysis for the Oneweb scenario.*

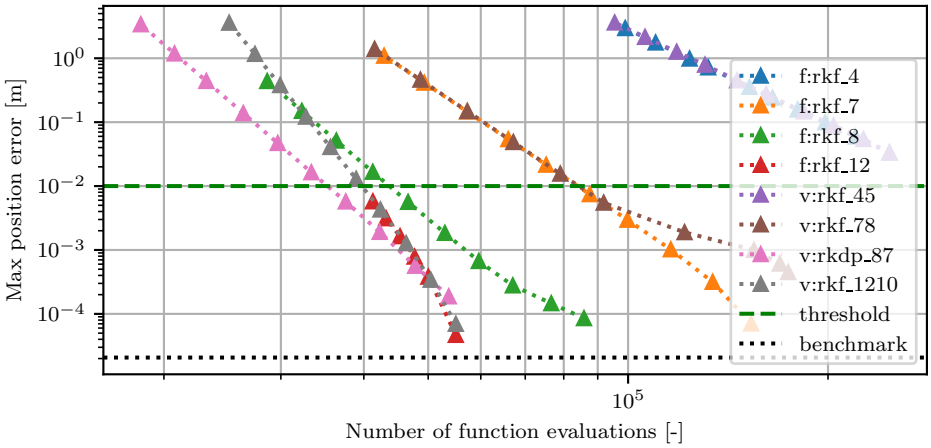


Figure C.7: *Integrator analysis for the Cowell propagator for the Oneweb scenario.*

The integrator analysis is for the Cowell propagator is shown in Figure C.7. The chosen integrator is the variable step rkdp87 integrator with an absolute tolerance of 2.0E-6. The position error over time for the variable step rkdp87 integrator is shown in Figure C.8

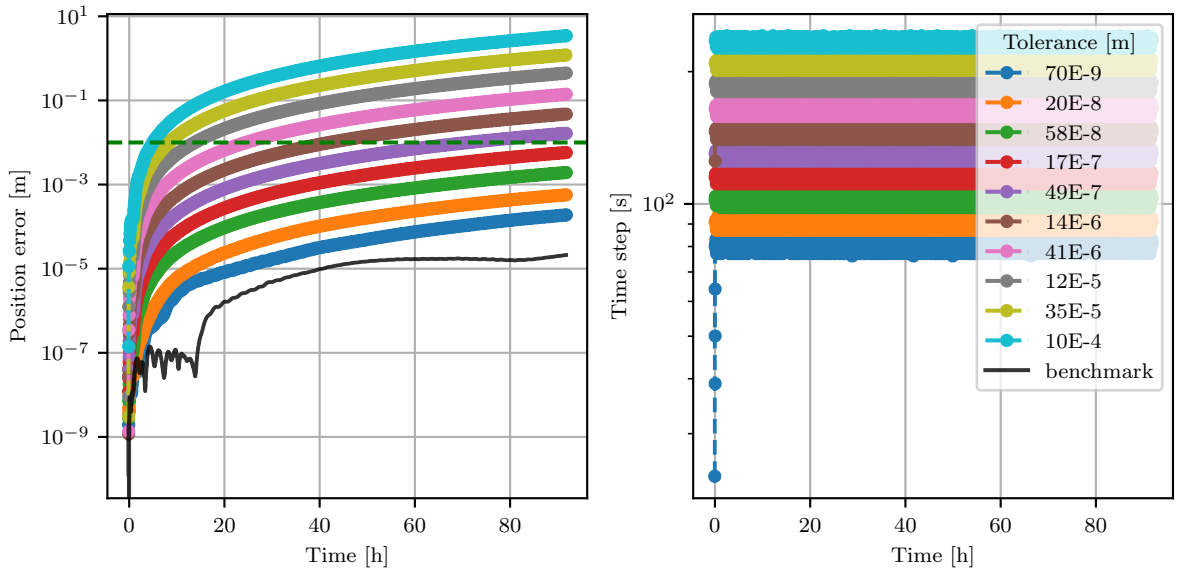


Figure C.8: *the position error (left) and the time step (right) over time for various absolute tolerances using the variable step rkdp87 integrator and Cowell propagator for the Oneweb scenario.*

The effects of removing/reducing accelerations are shown in Figure C.9.

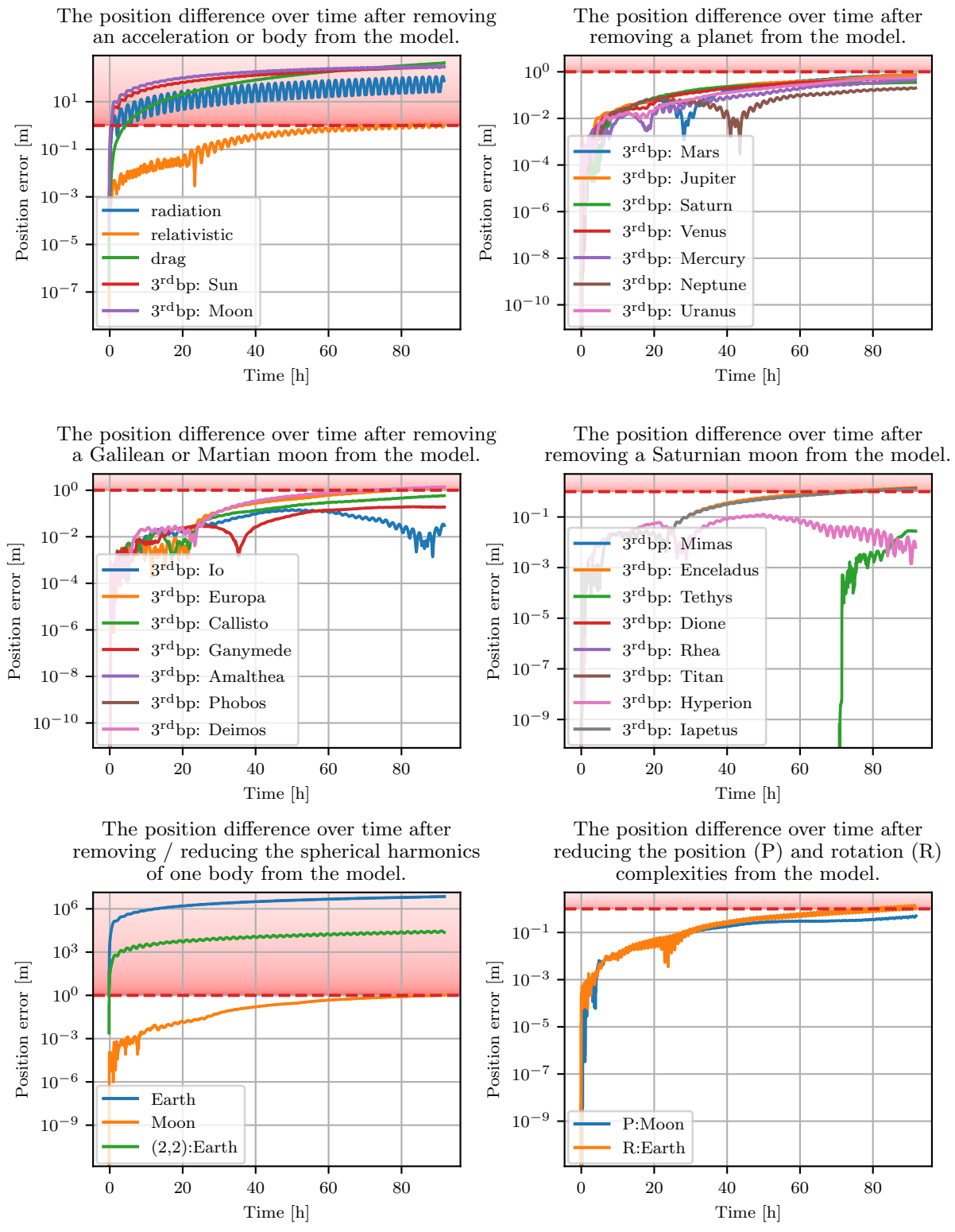


Figure C.9: *Acceleration model analysis of the Oneweb scenario.*

For this scenario, all 3rd body perturbations can be removed, except for the Sun and Moon, where for the Moon, the spherical harmonics can be reduced to a point mass. Both the atmospheric drag and solar radiation are kept, but the relativistic correction is removed from the final model. Furthermore, since the change from Earths 100x100 spherical harmonics to 2x2 or to a point mass gravity both result in high position errors, the final model still contains the 100x100 spherical harmonics. Finally, both the position model of the Moon, as well as the rotation model of the Earth are reduced in complexity.

All changes for which the effects remain below $1m$ have been included in the final model, which is shown in Table C.4. Based on the effects shown in Figure C.9, a more lenient selection of accelerations and models is made for the low-fidelity model (resulting in higher propagation errors and lower computation times), which is shown in Table C.5.

Acceleration type	
Central body gravity (Earth)	100 x 100 (GOCO05c)
3rd body perturbations (point mass)	Sun, Moon
Solar radiation pressure	C_R = 1.5 (Cannonball)
Atmospheric drag	C_D = 2.0 (US76)
Environment models	
Central body rotation model	Constant rotation rate determined at t_0
3rd body position model (Moon)	Keplerian 2-body orbit determined at t_0
3rd body position model (Sun)	Retrieved every epoch from Spice
Propagator settings	
Integrator	variable rkdp87 (tolerance: 2E-6)
Propagator	Cowell

Table C.4: *Propagation settings used for the high-fidelity dynamics of the Oneweb scenario*

Acceleration type	
Central body gravity (Earth)	2 x 2 (GOCO05c)
Environment models	
Central body rotation model	Constant rotation rate determined at t_0
Propagator settings	
Integrator	variable rkdp87 (tolerance: 2E-6)
Propagator	Cowell

Table C.5: *Propagation settings used for the low-fidelity dynamics of the Oneweb scenario*

C.3. Cosmos Scenario

For the Cosmos scenario, the final error is chosen to be below $1m$, resulting in a maximum integrator error of $10^{-2}m$ and a benchmark error of $10^{-4}m$.

The benchmark analysis is shown in Figure C.10, from which a benchmark time step of $30s$ is selected.

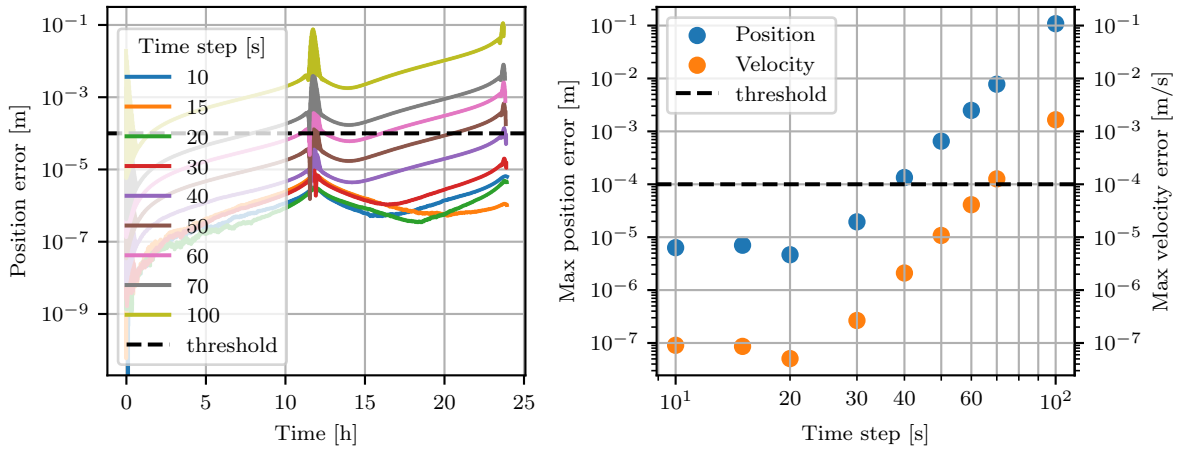


Figure C.10: *Benchmark error analysis for the Cosmos scenario.*

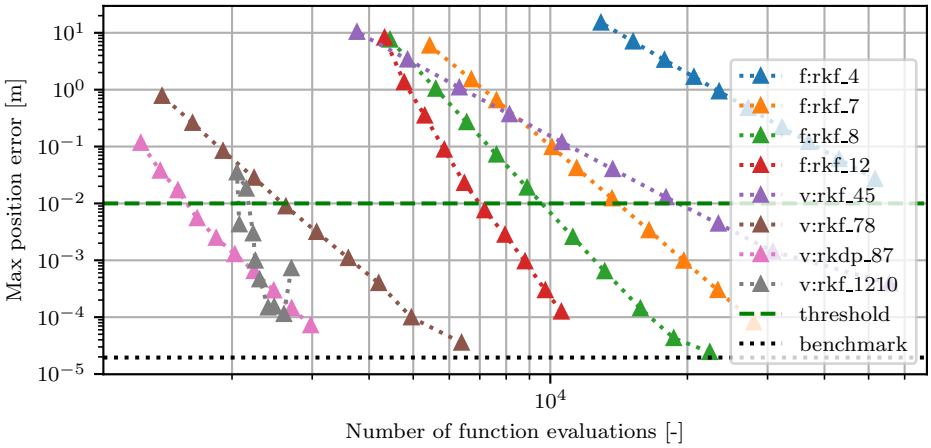


Figure C.11: *Integrator analysis for the Cowell propagator for the Cosmos scenario.*

The integrator analysis for the Cowell propagator is shown in Figure C.11. The chosen integrator is the variable step rkdp87 integrator with an absolute tolerance of 1.0E-3. The position error over time for the variable step rkdp87 integrator is shown in Figure C.12

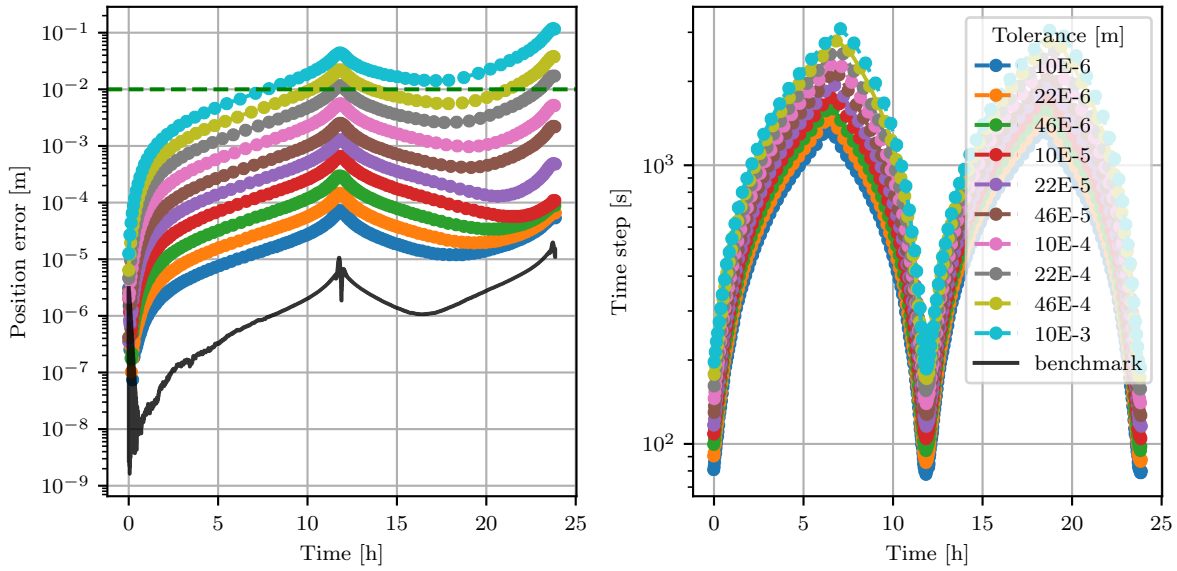


Figure C.12: *the position error (left) and the time step (right) over time for various absolute tolerances using the variable step rkdp87 integrator and Cowell propagator for the Cosmos scenario.*

The effects of removing/reducing accelerations are shown in Figure C.13.

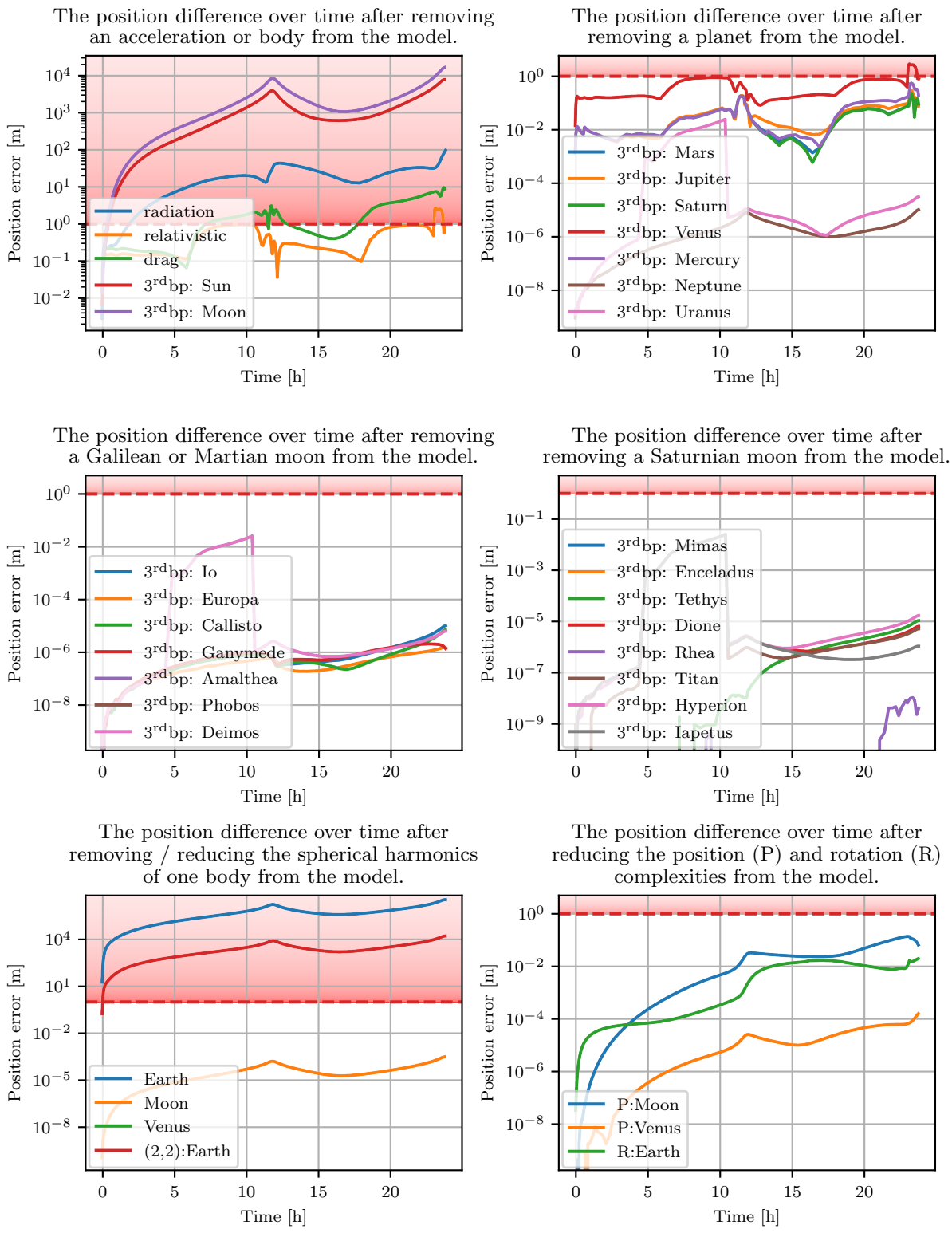


Figure C.13: *Acceleration model analysis of the Cosmos scenario.*

For this scenario, all 3rd body perturbations can be removed, except for the Sun, Moon and Venus, where for the Moon and Venus, the spherical harmonics can be reduced to a point mass. The atmospheric drag, solar radiation and relativistic correction are all kept in the final model. Furthermore, since the change from Earths 100x100 spherical harmonics to 2x2 or to a point mass gravity both result in high position errors, the final model still contains the 100x100 spherical harmonics. Finally, both the position model of the Moon and Venus, as well as the rotation model of the Earth are reduced in complexity.

All changes for which the effects remain below $1m$ have been included in the final model, which is shown in Table C.6. Based on the effects shown in Figure C.13, a more lenient selection of accelerations and models is made for the low-fidelity model (resulting in higher propagation errors and lower computation times), which is shown in Table C.7.

Acceleration type	
Central body gravity (Earth)	100 x 100 (GOCO05c)
3rd body perturbations (point mass)	Sun, Moon, Venus
Solar radiation pressure	C_R = 1.5 (Cannonball)
Atmospheric drag	C_D = 2.0 (US76)
Relativistic Correction	Sun (Schwarzschild)
Environment models	
Central body rotation model	Constant rotation rate determined at t_0
3rd body position model (Moon, Venus)	Keplerian 2-body orbit determined at t_0
3rd body position model (Sun)	Retrieved every epoch from Spice
Propagator settings	
Integrator	variable rkdp87 (tolerance: 1E-3)
Propagator	Cowell

Table C.6: *Propagation settings used for the high-fidelity dynamics of the Cosmos scenario*

Acceleration type	
Central body gravity (Earth)	2 x 2 (GOCO05c)
3rd body perturbations (point mass)	Sun, Moon
Environment models	
Central body rotation model	Constant rotation rate determined at t_0
3rd body position model (Moon)	Keplerian 2-body orbit determined at t_0
3rd body position model (Sun)	Retrieved every epoch from Spice
Propagator settings	
Integrator	variable rkdp87 (tolerance: 1E-3)
Propagator	Cowell

Table C.7: *Propagation settings used for the low-fidelity dynamics of the Cosmos scenario*

C.4. Spirale Scenario

For the Spirale scenario, the final error is chosen to be below $1m$, resulting in a maximum integrator error of $10^{-2}m$ and a benchmark error of $10^{-4}m$.

The benchmark analysis is shown in Figure C.14, from which a benchmark time step of $30s$ is selected.

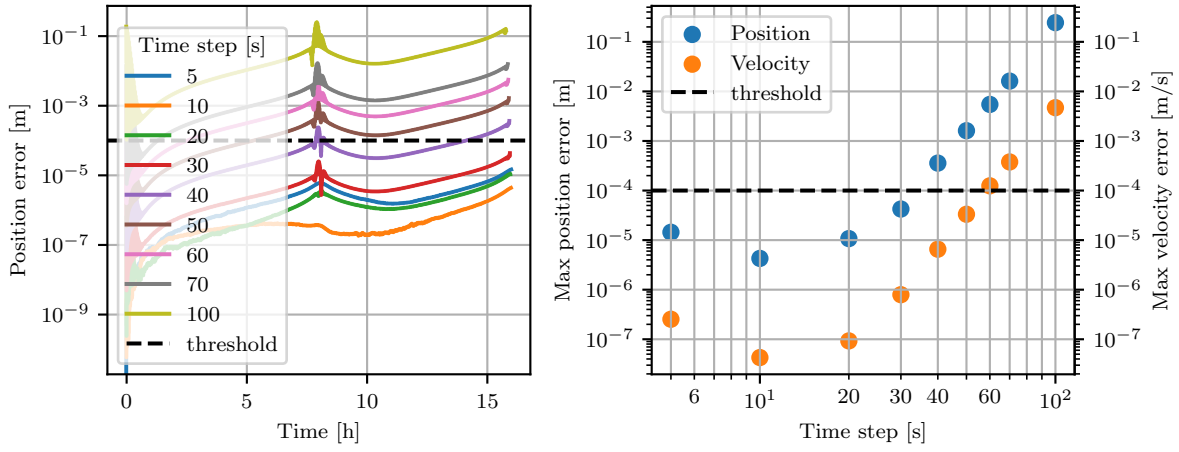


Figure C.14: *Benchmark error analysis for the Spirale scenario.*

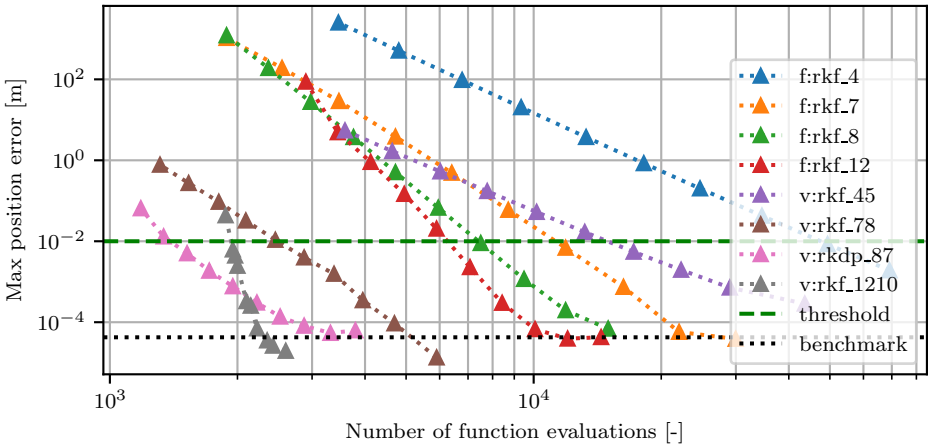


Figure C.15: *Integrator analysis for the Cowell propagator for the Spirale scenario.*

The integrator analysis for the Cowell propagator is shown in Figure C.15. The chosen integrator is the variable step rkdp87 integrator with an absolute tolerance of 2.0E-3. The position error over time for the variable step rkdp87 integrator is shown in Figure C.16

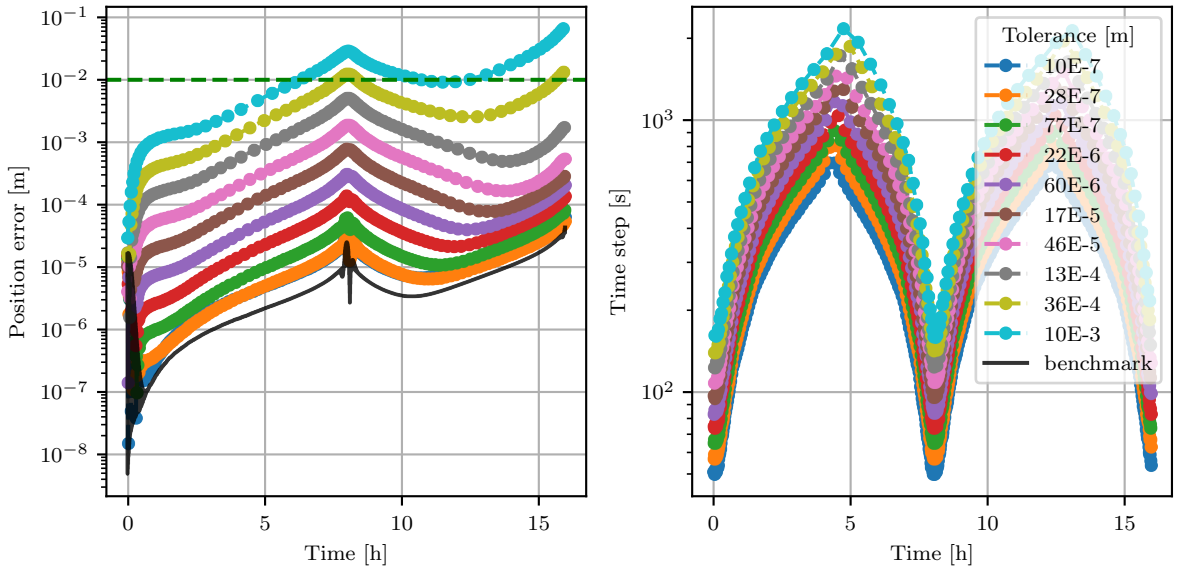


Figure C.16: *the position error (left) and the time step (right) over time for various absolute tolerances using the variable step rkdp87 integrator and Cowell propagator for the Spirale scenario.*

The effects of removing/reducing accelerations are shown in Figure C.17.

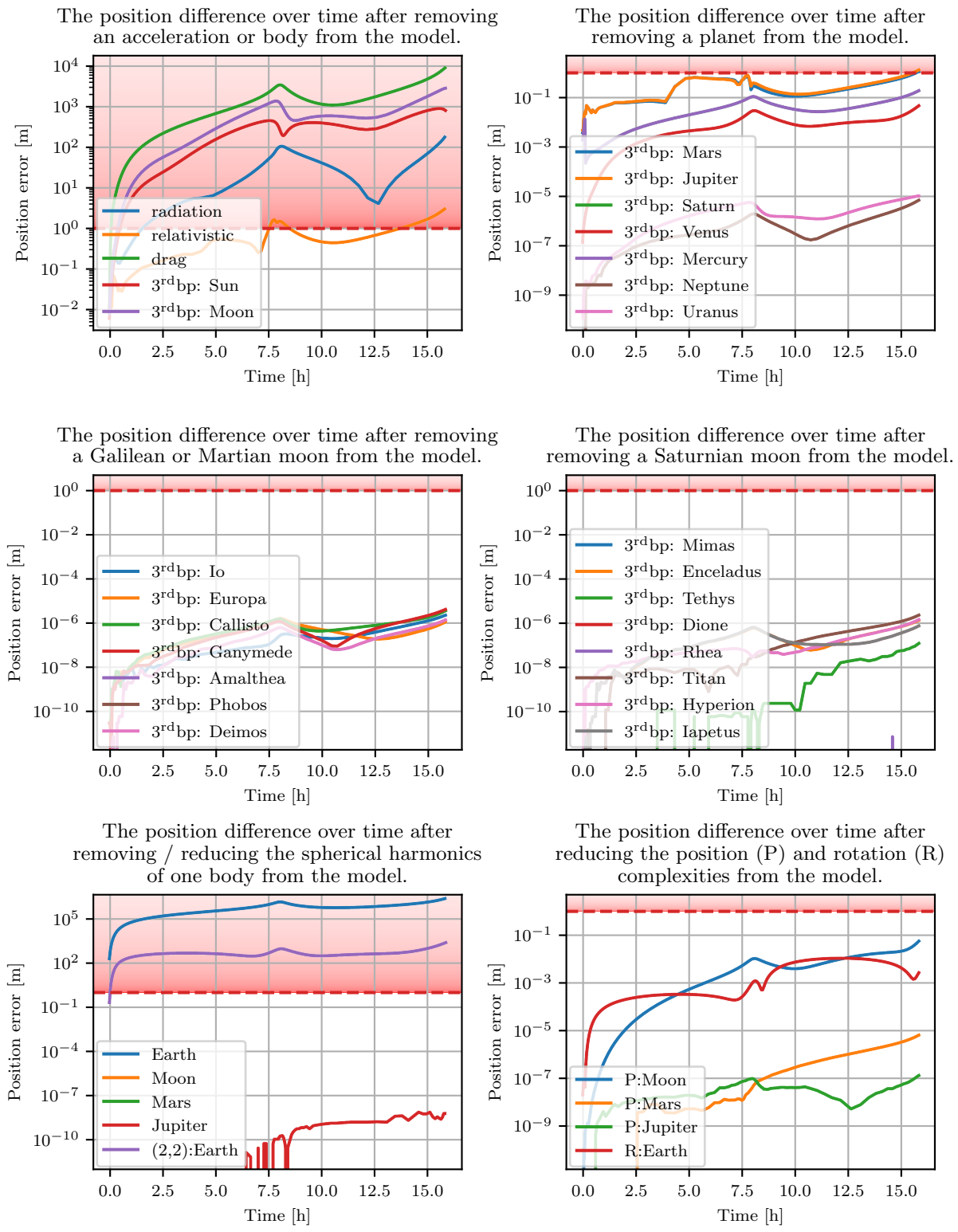


Figure C.17: *Acceleration model analysis of the Spirale scenario.*

For this scenario, all 3rd body perturbations can be removed, except for the Sun, Moon, Mars and Jupiter, where for the Moon, Mars and Jupiter, the spherical harmonics can be reduced to a point mass. The atmospheric drag, solar radiation and relativistic correction are all kept in the final model. Furthermore, since the change from Earths 100x100 spherical harmonics to 2x2 or to a point mass gravity both result in high position errors, the final model still contains the 100x100 spherical harmonics. Finally, both the position models of the Moon, Mars and Jupiter, as well as the rotation model of the Earth are reduced in complexity.

All changes for which the effects remain below $1m$ have been included in the final model, which is shown in Table C.8. Based on the effects shown in Figure C.17, a more lenient selection of accelerations and models is made for the low-fidelity model (resulting in higher propagation errors and lower computation times), which is shown in Table C.9.

Acceleration type	
Central body gravity (Earth)	100 x 100 (GOCO05c)
3rd body perturbations (point mass)	Sun, Moon, Mars, Jupiter
Solar radiation pressure	C_R = 1.5 (Cannonball)
Atmospheric drag	C_D = 2.0 (US76)
Relativistic Correction	Sun (Schwarzschild)
Environment models	
Central body rotation model	Constant rotation rate determined at t_0
3rd body position model (Moon, Mars, Jupiter)	Keplerian 2-body orbit determined at t_0
3rd body position model (Sun)	Retrieved every epoch from Spice
Propagator settings	
Integrator	variable rkdp87 (tolerance: 2E-3)
Propagator	Cowell

Table C.8: *Propagation settings used for the high-fidelity dynamics of the Spirale scenario*

Acceleration type	
Central body gravity (Earth)	2 x 2 (GOCO05c)
3rd body perturbations (point mass)	Sun, Moon
Atmospheric drag	C_D = 2.0 (US76)
Environment models	
Central body rotation model	Constant rotation rate determined at t_0
3rd body position model (Moon)	Keplerian 2-body orbit determined at t_0
3rd body position model (Sun)	Retrieved every epoch from Spice
Propagator settings	
Integrator	variable rkdp87 (tolerance: 2E-3)
Propagator	Cowell

Table C.9: *Propagation settings used for the low-fidelity dynamics of the Spirale scenario*

C.5. Meteosat Scenario

For the Meteosat scenario, the final error is chosen to be below $1m$, resulting in a maximum integrator error of $10^{-2}m$ and a benchmark error of $10^{-4}m$.

The benchmark analysis is shown in Figure C.18, from which a benchmark time step of $600s$ is selected.

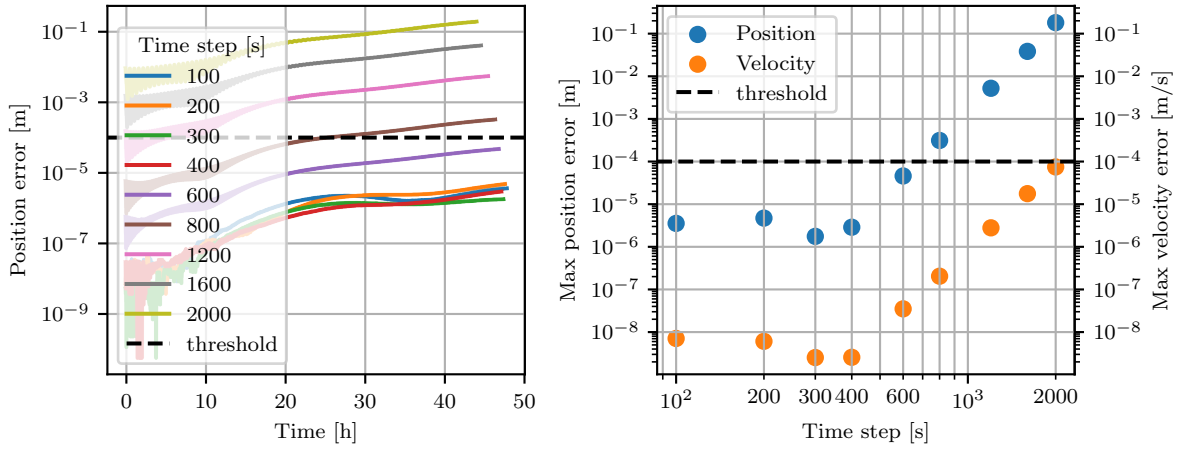


Figure C.18: *Benchmark error analysis for the Meteosat scenario.*

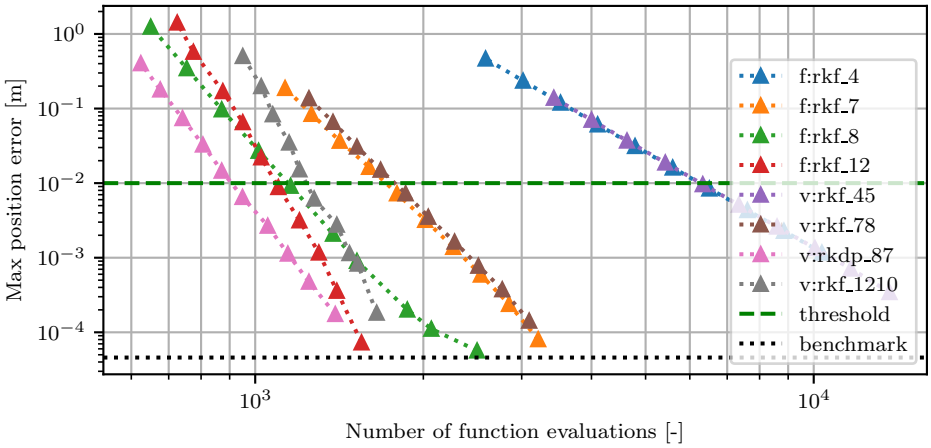


Figure C.19: *Integrator analysis for the Cowell propagator for the Meteosat scenario.*

The integrator analysis for the Cowell propagator is shown in Figure C.19. The chosen integrator is the fixed step rkf8 integrator with a time step of 1800s. The position error over time for the fixed step rkf8 integrator is shown in Figure C.20. It is noted that a variable step rkdp87 integrator and a fixed step rkf12 integrator can provide a faster propagation in this case. However, due to the small difference in performance, a fixed step integrator is chosen for this scenario, to include at least one scenario with a fixed step integrator. The rkf12 integrator is not chosen due to the optimal time step being significantly larger than that for the rkf8. Here, a smaller time step is preferred, to decrease interpolation errors.

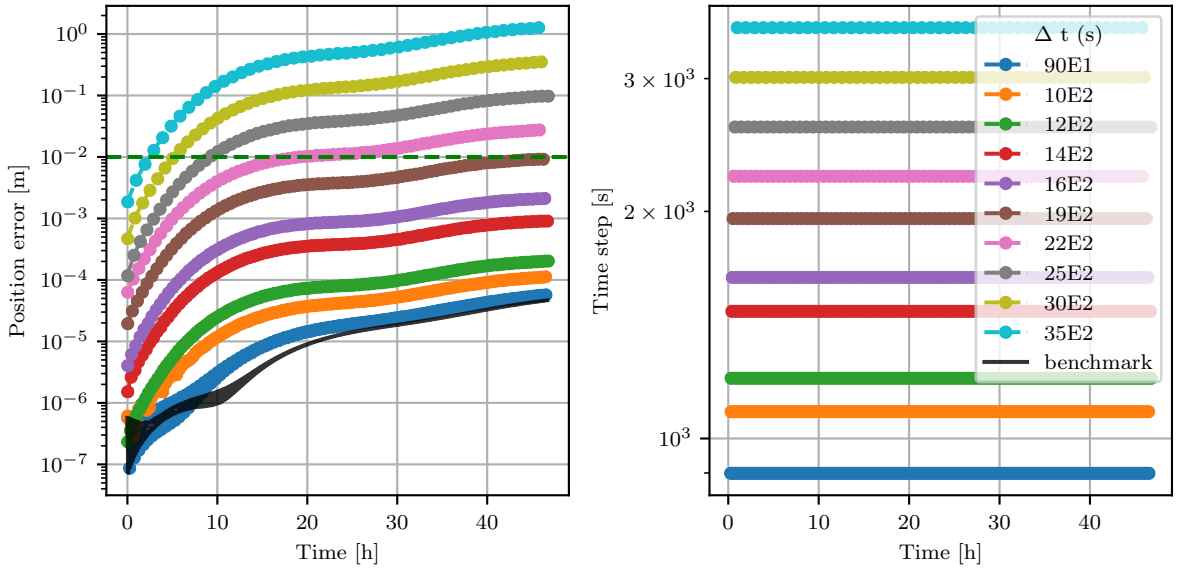


Figure C.20: *The position error (left) and the time step (right) over time for various time steps using the fixed step rkf8 integrator and Cowell propagator for the Meteosat scenario.*

The effects of removing/reducing accelerations are shown in Figure C.21.

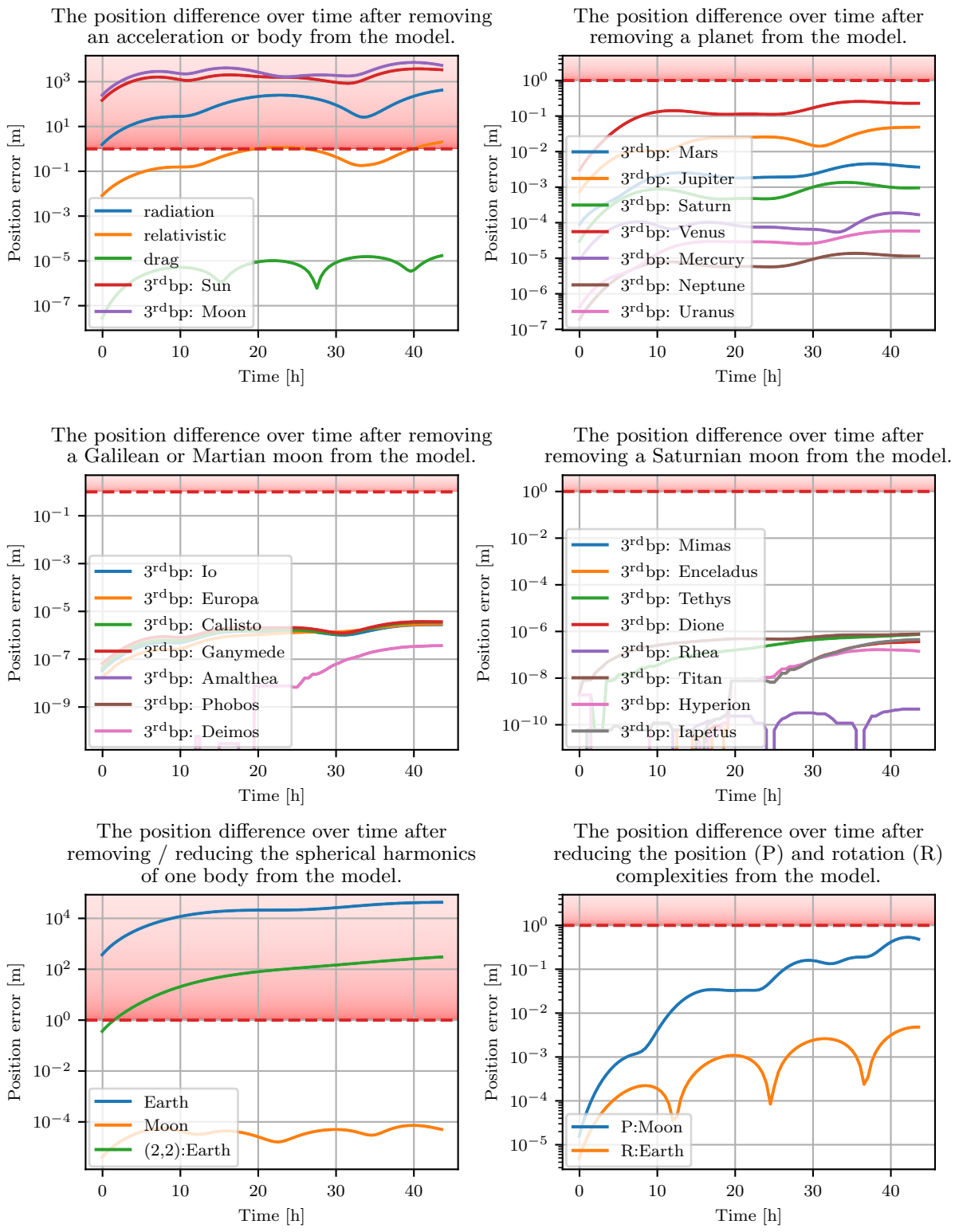


Figure C.21: *Acceleration model analysis of the Meteosat scenario.*

For this scenario, all 3rd body perturbations can be removed, except for the Sun and Moon, where for the Moon the spherical harmonics can be reduced to a point mass. Both the solar radiation and relativistic correction are kept, but the atmospheric drag is removed from the final model. Furthermore, since the change from Earths 100x100 spherical harmonics to 2x2 or to a point mass gravity both result in high position errors, the final model still contains the 100x100 spherical harmonics. Finally, both the position models of the Moon, Mars and Jupiter, as well as the rotation model of the Earth are reduced in complexity.

All changes for which the effects remain below $1m$ have been included in the final model, which is shown in Table C.10. Based on the effects shown in Figure C.21, a more lenient selection of accelerations and models is made for the low-fidelity model (resulting in higher propagation errors and lower computation times), which is shown in Table C.11.

Acceleration type	
Central body gravity (Earth)	100 x 100 (GOCO05c)
3rd body perturbations (point mass)	Sun, Moon
Solar radiation pressure	C_R = 1.5 (Cannonball)
Relativistic Correction	Sun (Schwarzschild)
Environment models	
Central body rotation model	Constant rotation rate determined at t_0
3rd body position model (Moon)	Keplerian 2-body orbit determined at t_0
3rd body position model (Sun)	Retrieved every epoch from Spice
Propagator settings	
Integrator	fixed rkf8 (time step: 1800s)
Propagator	Cowell

Table C.10: *Propagation settings used for the high-fidelity dynamics of the Meteosat scenario*

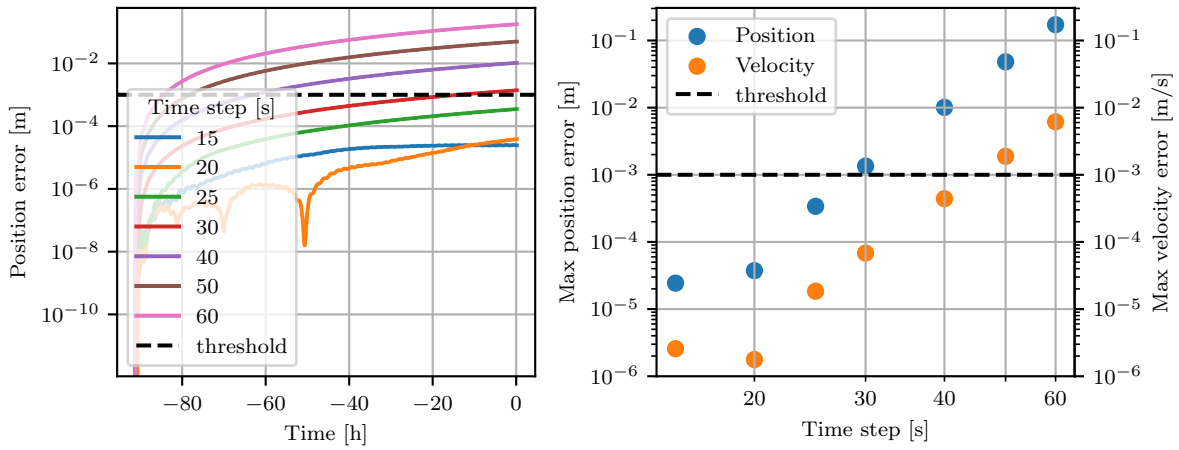
Acceleration type	
Central body gravity (Earth)	2 x 2 (GOCO05c)
Environment models	
Central body rotation model	Constant rotation rate determined at t_0
Propagator settings	
Integrator	fixed rkf8 (time step: 1800s)
Propagator	Cowell

Table C.11: *Propagation settings used for the low-fidelity dynamics of the Meteosat scenario*

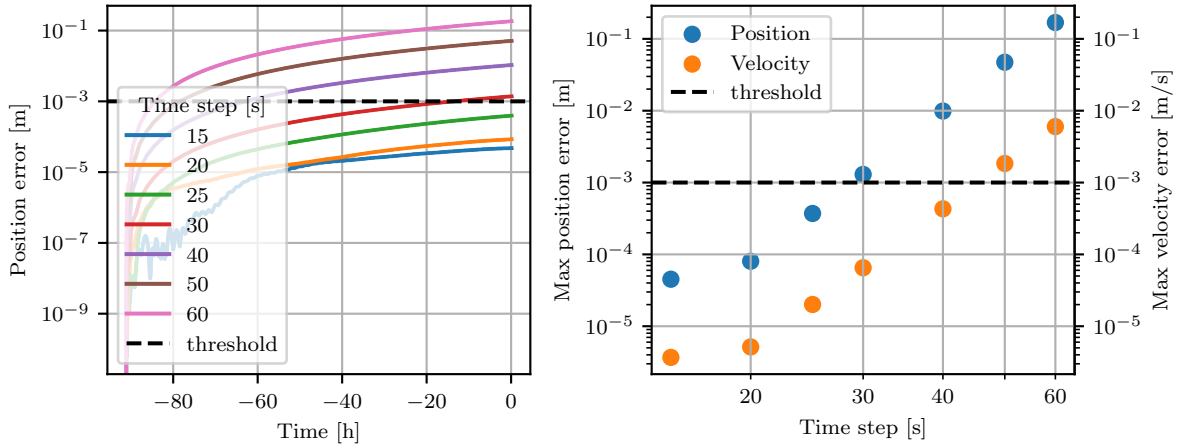
C.6. High-velocity Scenario

For the high-velocity scenario, the final error is chosen to be below $1m$, resulting in a maximum integrator error of $10^{-2}m$ and a benchmark error of $10^{-4}m$. The analysis is based on the long version of the scenario, for which the results are shown below.

The benchmark analysis is shown for both objects in Figure C.22, from which a benchmark time step of $25s$ is selected.



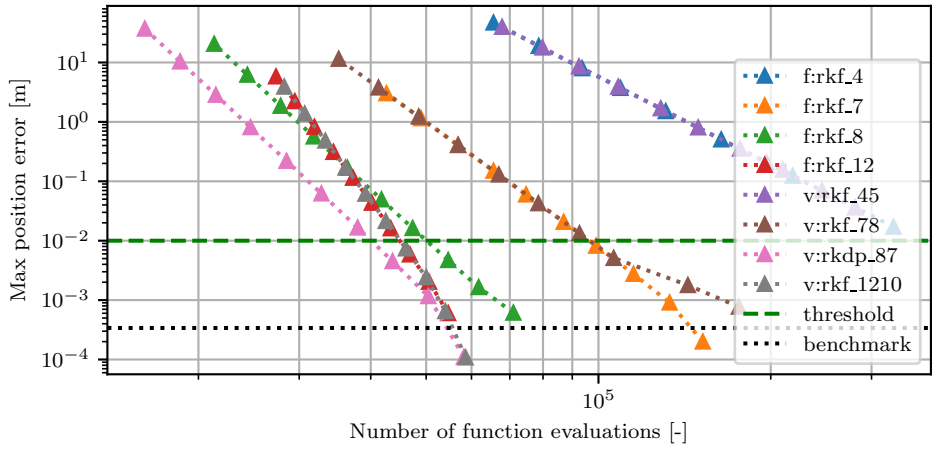
(a) *Object 1.*



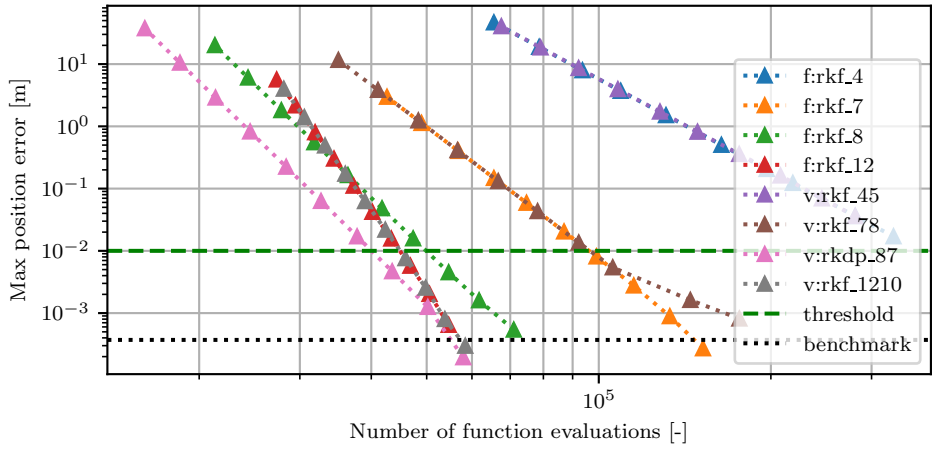
(b) *Object 2.*

Figure C.22: *Benchmark error analysis for the high-velocity scenario.*

The integrator analysis for the Cowell propagator is shown in Figure C.23. The chosen integrator is the variable step rkdp87 integrator with an absolute tolerance of 2.0E-6. The position error over time for the variable step rkdp87 integrator is shown in Figure C.24.

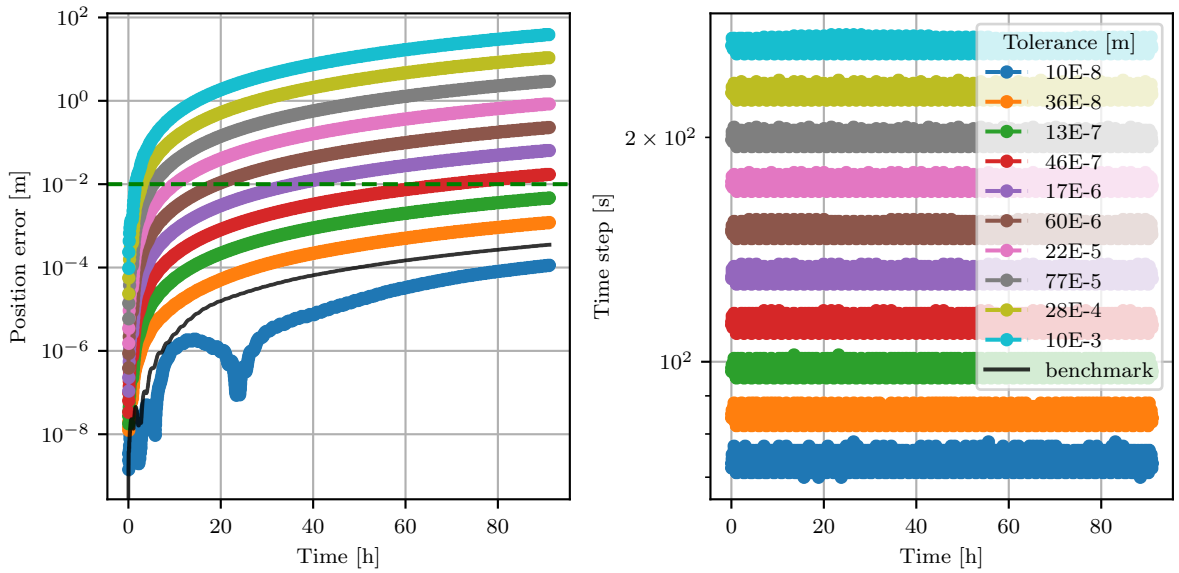


(a) *Object 1.*

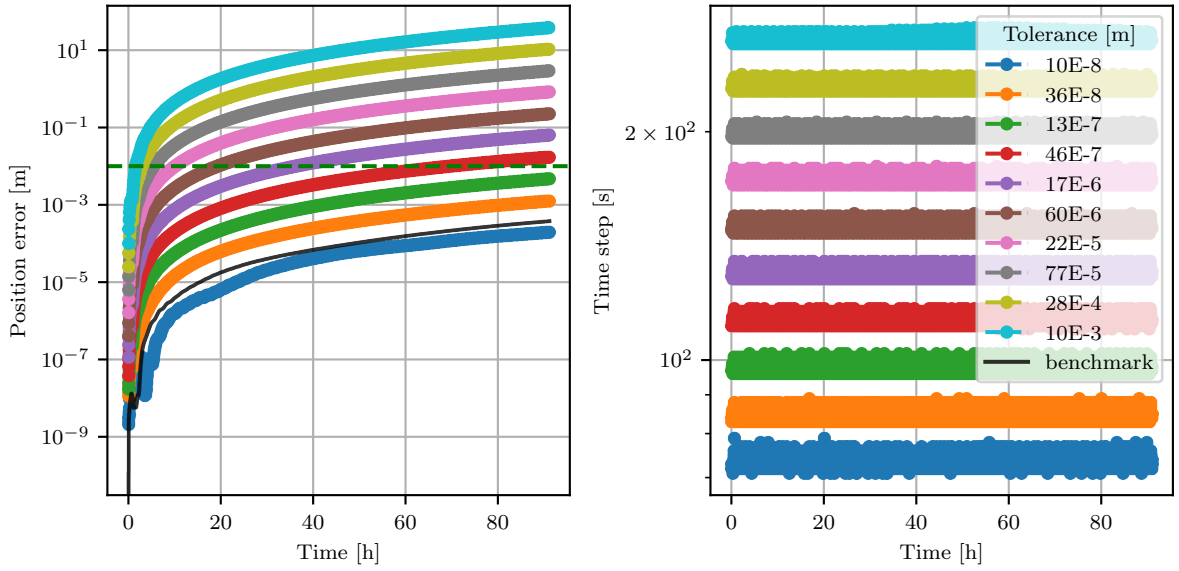


(b) *Object 2.*

Figure C.23: *Integrator analysis for the Cowell propagator for the high-velocity scenario.*



(a) *Object 1.*



(b) *Object 2.*

Figure C.24: *The position error (left) and the time step (right) over time for various absolute tolerances using the variable step rkdp87 integrator and Cowell propagator for the high-velocity scenario.*

The effects of removing/reducing accelerations are shown in Figures C.25 and C.26.

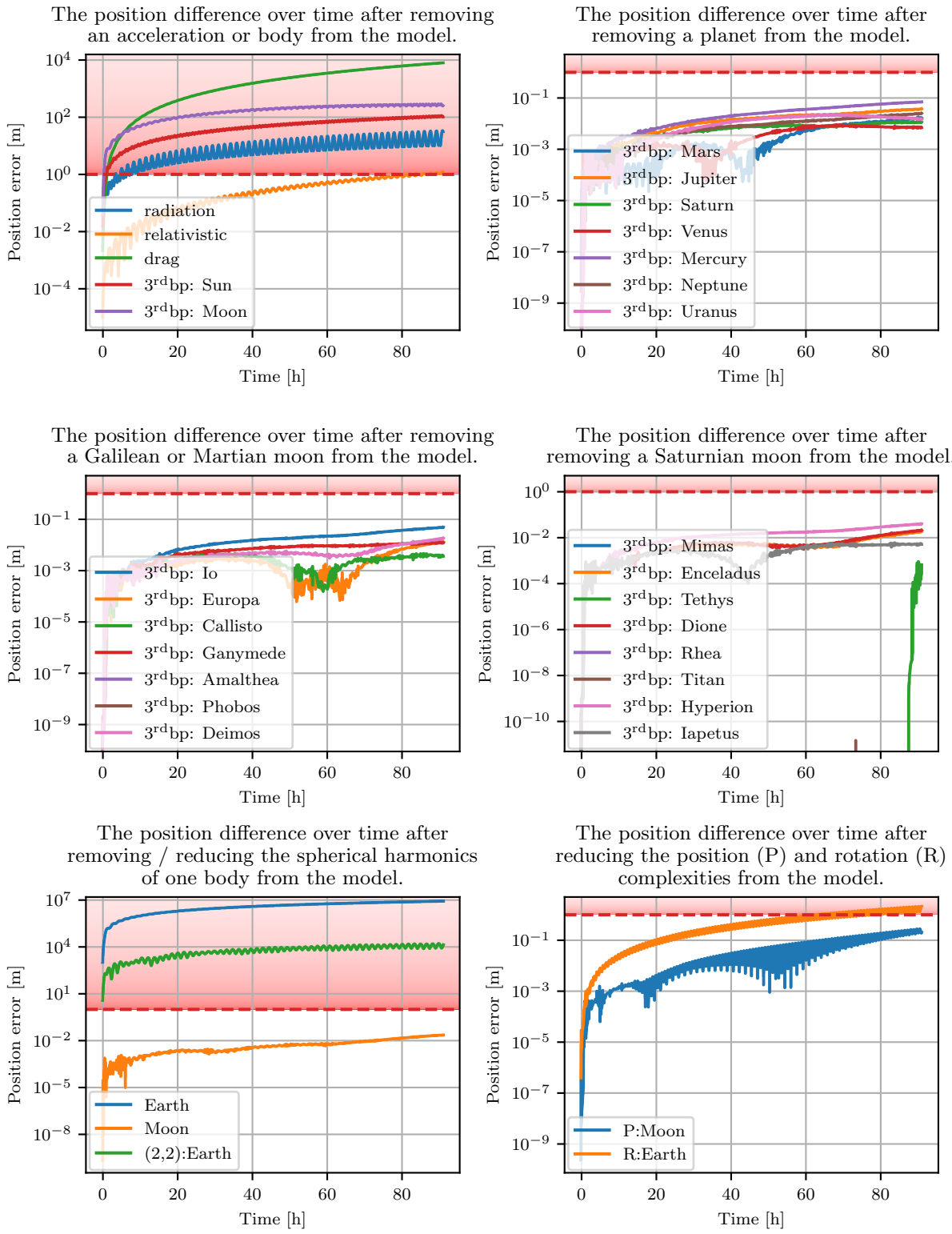


Figure C.25: *Acceleration model analysis of the high-velocity scenario for object 1.*

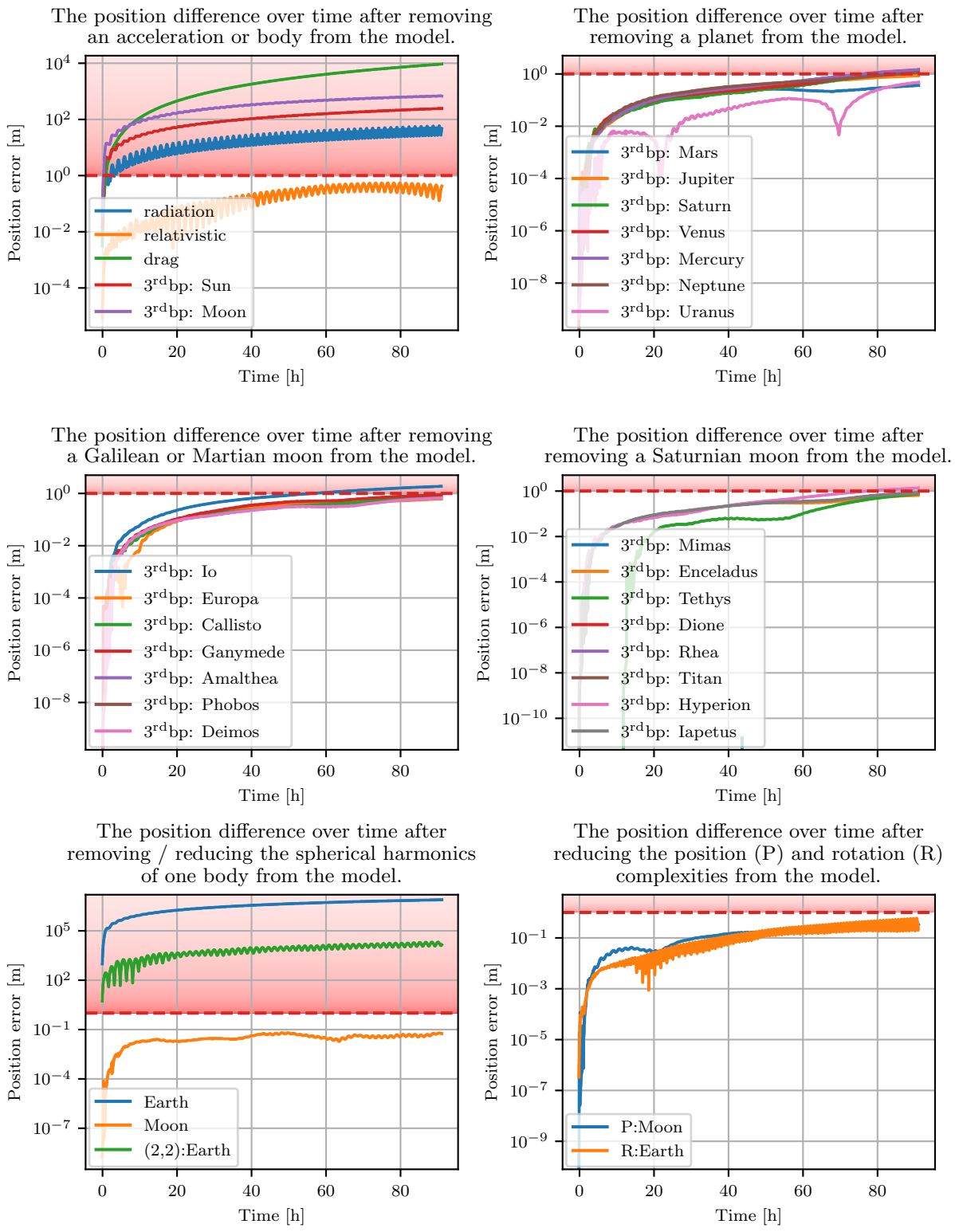


Figure C.26: *Acceleration model analysis of the high-velocity scenario for object 2.*

For this scenario, all 3rd body perturbations are removed, except for the Sun and Moon, where for the Moon, the spherical harmonics can be reduced to a point mass. Both the atmospheric drag and solar radiation are kept, but the relativistic correction is removed from the final model. Furthermore, since the change from Earths 100x100 spherical harmonics to 2x2 or to a point mass gravity both result in high position errors for both objects, the final model still contains the 100x100 spherical harmonics. Finally, both the position model of the Moon, as well as the rotation model of the Earth are reduced in complexity. All changes for which the effects

remain below $1m$ for both objects have been included in the final model, which is shown in Table C.12.

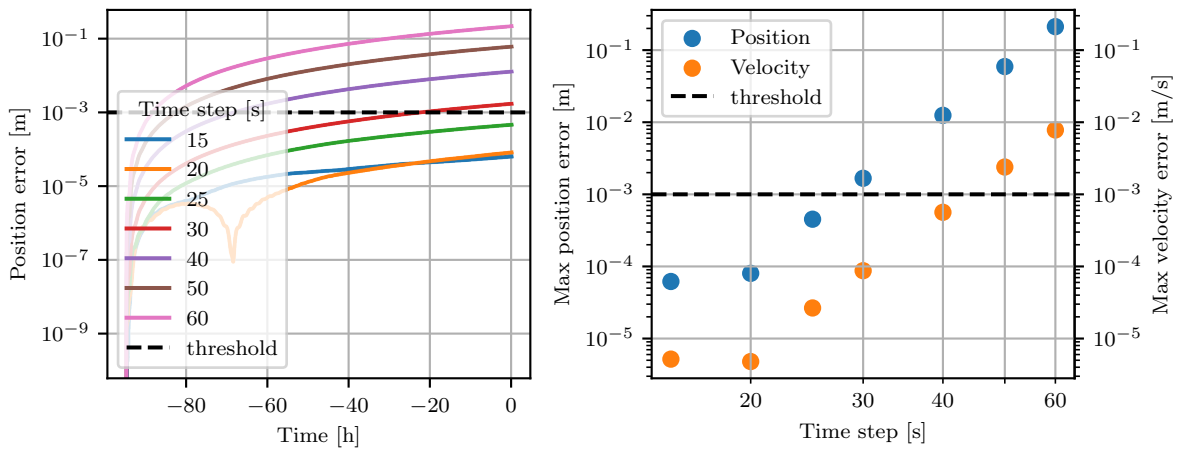
Acceleration type	
Central body gravity (Earth)	100 x 100 (GOCO05c)
3rd body perturbations (point mass)	Sun, Moon
Solar radiation pressure	C_R = 1.5 (Cannonball)
Atmospheric drag	C_D = 2.0 (US76)
Environment models	
Central body rotation model	Constant rotation rate determined at t_0
3rd body position model (Moon)	Keplerian 2-body orbit determined at t_0
position model (Sun	Retrieved every epoch from Spice
Propagator settings	
Integrator	variable rkdp87 (tolerance: 2E-6)
Propagator	Cowell

Table C.12: *Propagation settings used for the high-velocity scenario*

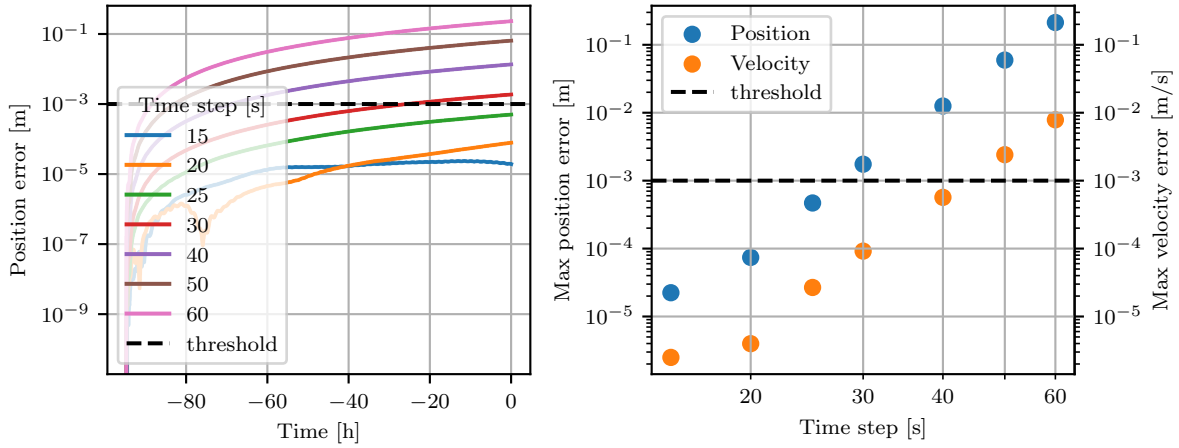
C.7. Low-velocity Scenario

For the low-velocity scenario, the final error is chosen to be below $1m$, resulting in a maximum integrator error of $10^{-2}m$ and a benchmark error of $10^{-4}m$. The analysis is based on the long version of the scenario, for which the results are shown below.

The benchmark analysis is shown for both objects in Figure C.27, from which a benchmark time step of $25s$ is selected.



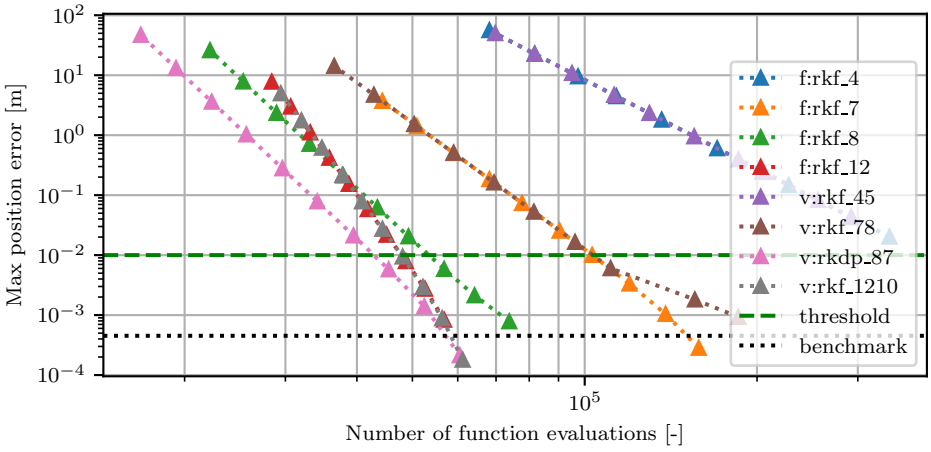
(a) *Object 1.*



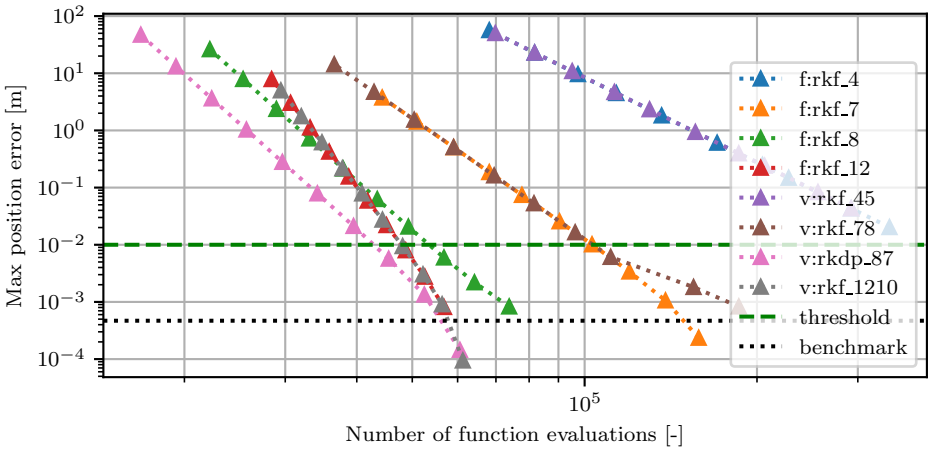
(b) *Object 2.*

Figure C.27: *Benchmark error analysis for the low-velocity scenario.*

The integrator analysis for the Cowell propagator is shown in Figure C.28. The chosen integrator is the variable step rkdp87 integrator with an absolute tolerance of 2.0E-6. The position error over time for the variable step rkdp87 integrator is shown in Figure C.29.

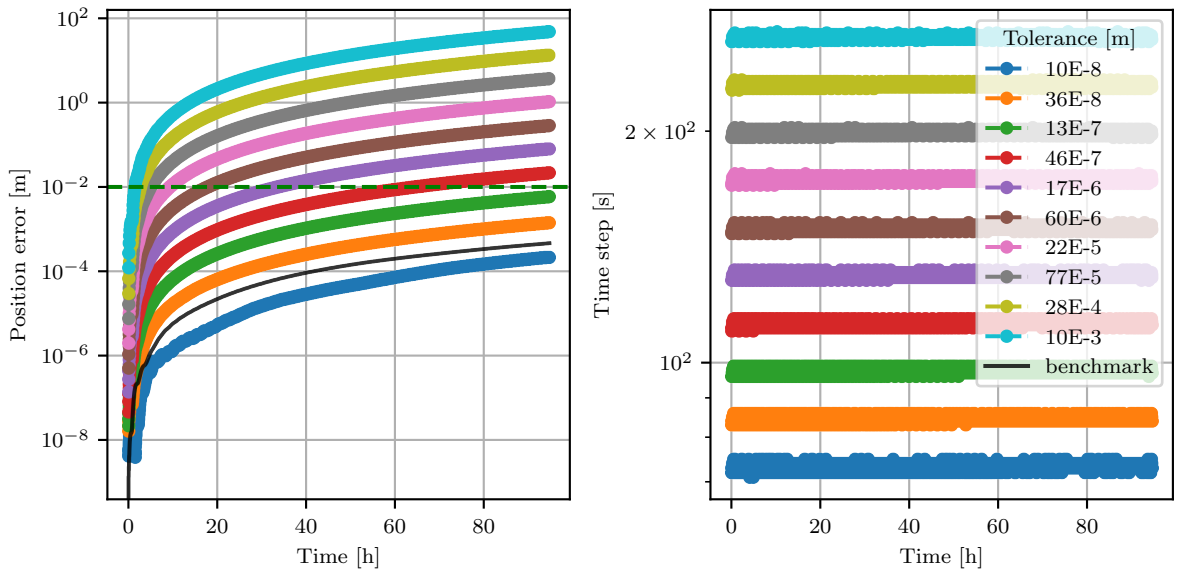


(a) *Object 1.*

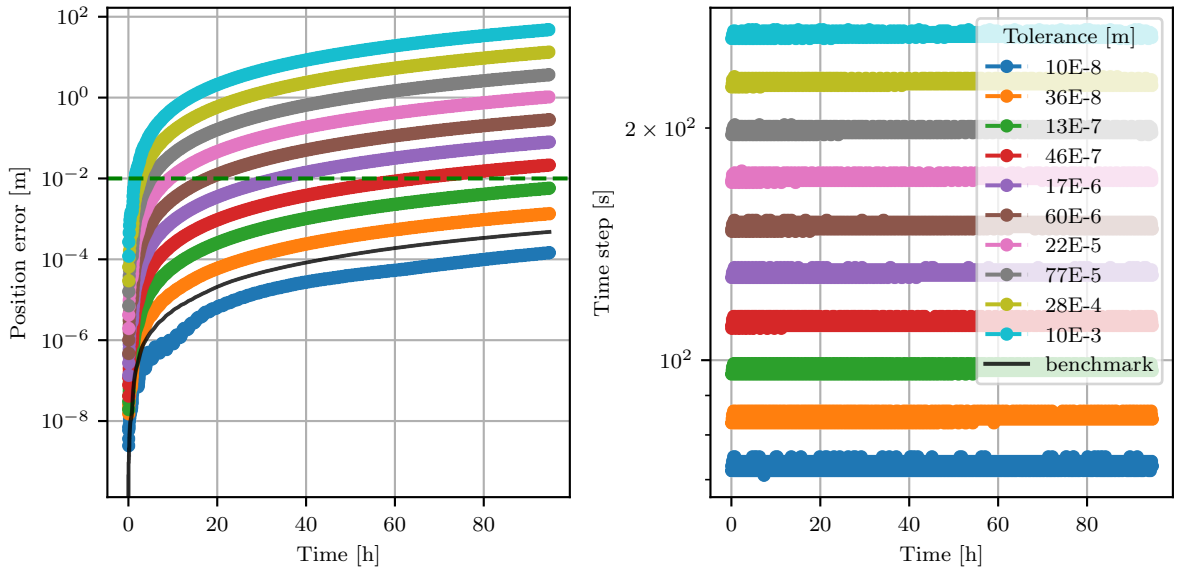


(b) *Object 2.*

Figure C.28: *Integrator analysis for the Cowell propagator for the low-velocity scenario.*



(a) *Object 1*



(b) *Object 2*

Figure C.29: *The position error (left) and the time step (right) over time for various absolute tolerances using the variable step rkdp87 integrator and Cowell propagator for the low-velocity scenario.*

The effects of removing/reducing accelerations are shown in Figures C.30 and C.31.

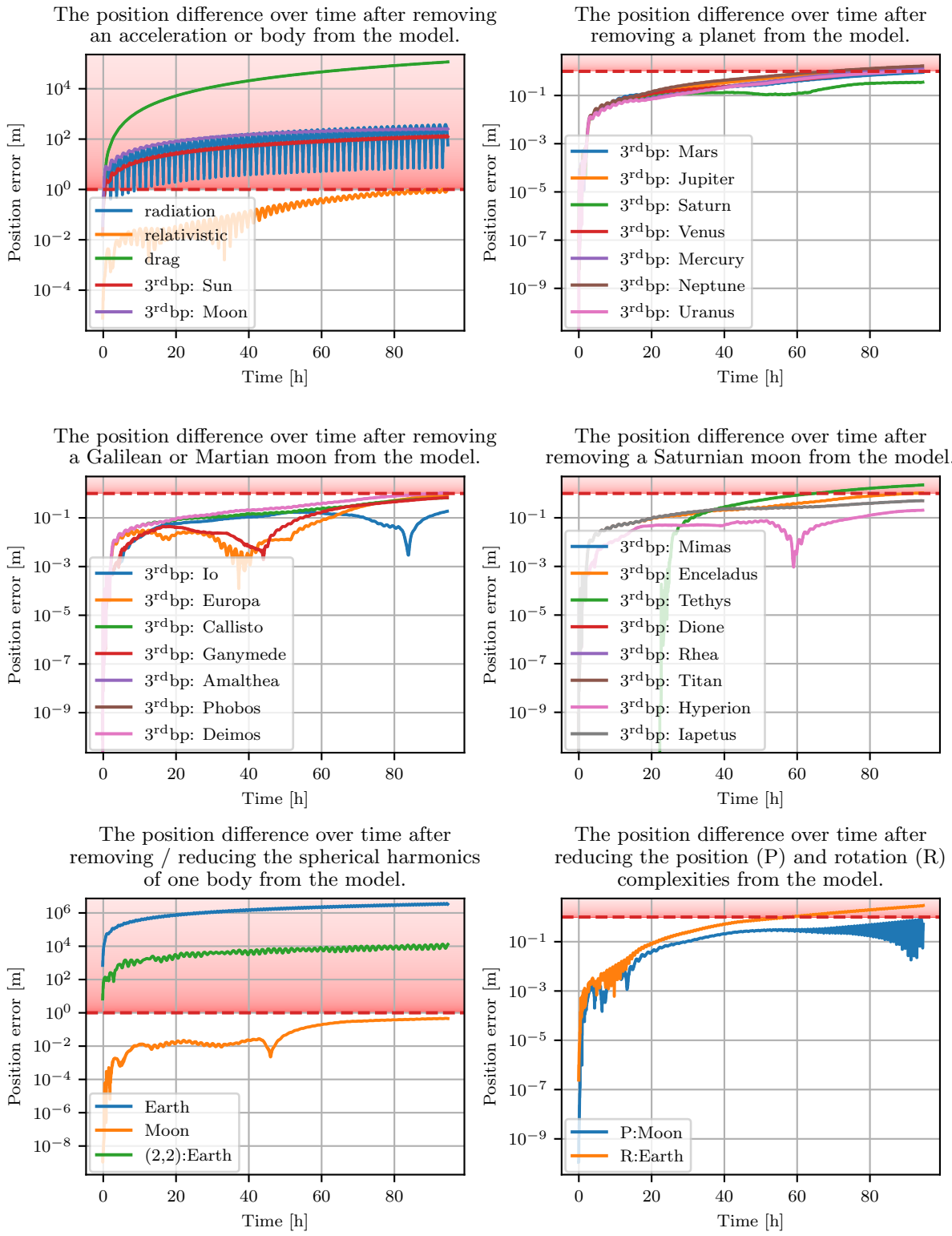


Figure C.30: *Acceleration model analysis of the low-velocity scenario for object 1.*

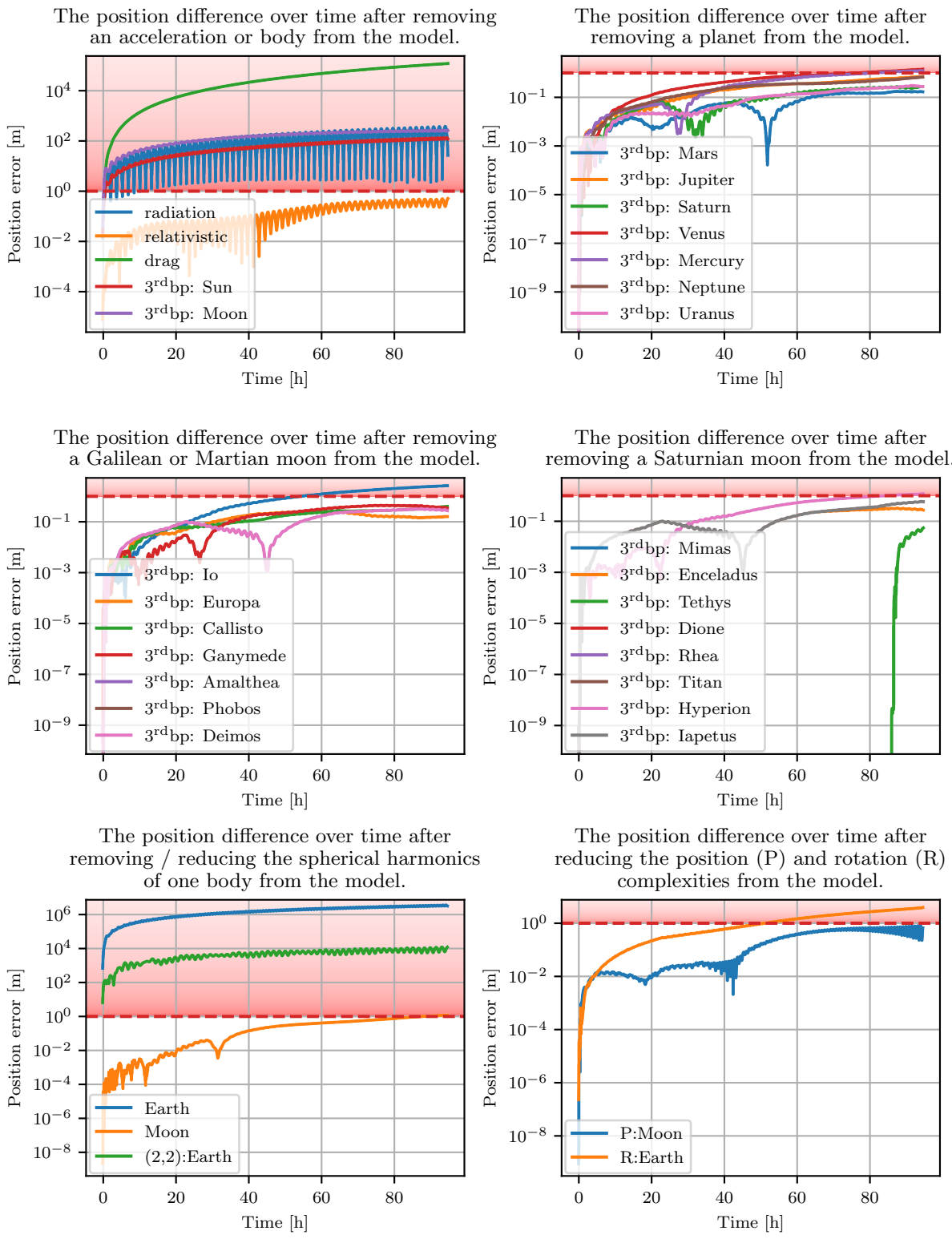


Figure C.31: *Acceleration model analysis of the lowvelocity scenario for object 2.*

For this scenario, all 3rd body perturbations are removed, except for the Sun and Moon, where for the Moon, the spherical harmonics can be reduced to a point mass. Both the atmospheric drag and solar radiation are kept, but the relativistic correction is removed from the final model. Furthermore, since the change from Earths 100x100 spherical harmonics to 2x2 or to a point mass gravity both result in high position errors for both objects, the final model still contains the 100x100 spherical harmonics. Finally, only the position model of the Moon is reduced in complexity.

All changes for which the effects remain below $1m$ for both objects have been included in the final model, which is shown in Table C.13.

Acceleration type	
Central body gravity (Earth)	100 x 100 (GOCO05c)
3rd body perturbations (point mass)	Sun, Moon
Solar radiation pressure	C_R = 1.5 (Cannonball)
Atmospheric drag	C_D = 2.0 (US76)
Environment models	
Central body rotation model	Retrieved every epoch from Spice
3rd body position model (Moon)	Keplerian 2-body orbit determined at t_0
3rd body position model (Sun)	Retrieved every epoch from Spice
Propagator settings	
Integrator	variable rkdp87 (tolerance: 2E-6)
Propagator	Cowell

Table C.13: *Propagation settings used for the low-velocity scenario*

D

Model Verification

This appendix contains the results of the verifications performed to ensure that the implemented uncertainty propagation methods perform as expected, by comparing results produced with the implemented methods to results found in literature, or to results generated by MC samples, for the same test case scenarios. The implemented and verified methods that are shown below are: AEGIS, UT, LinCov, MF and PCEs. For each method it is shown that there is a good agreement between the generated results and the baseline.

D.1. AEGIS verification

The implemented AEGIS method has been tested and compared against known results to verify that it functions correctly and produces the expected results. In [108] the Splitting Gaussian Mixture Unscented Kalman Filter (SGMUKF) is described. This is essentially the filter version of the Adaptive Entropy-based Gaussian-mixture information Synthesis (AEGIS) described in [28], which is implemented in this research. When the method is used to propagate an uncertainty forwards in time under the absence of observations, both methods function identically. The test case scenario of an elliptical orbit, described in [108] is replicated and used to produce results for, using the implemented AEGIS method. The mean vector of the initial uncertainty is transformed to Cartesian coordinates from the following Keplerian elements:

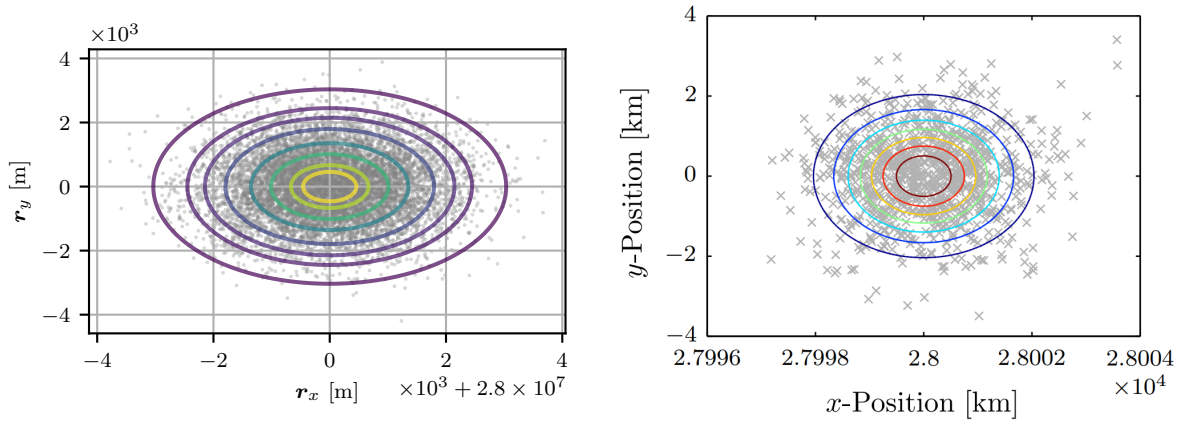
a	$35 \cdot 10^6$ m
e	0.2
i	0 deg
Ω	0 deg
ω	0 deg
θ	0 deg

Table D.1: *The Keplerian elements of the mean vector at the initial epoch.*

With the following covariance matrix:

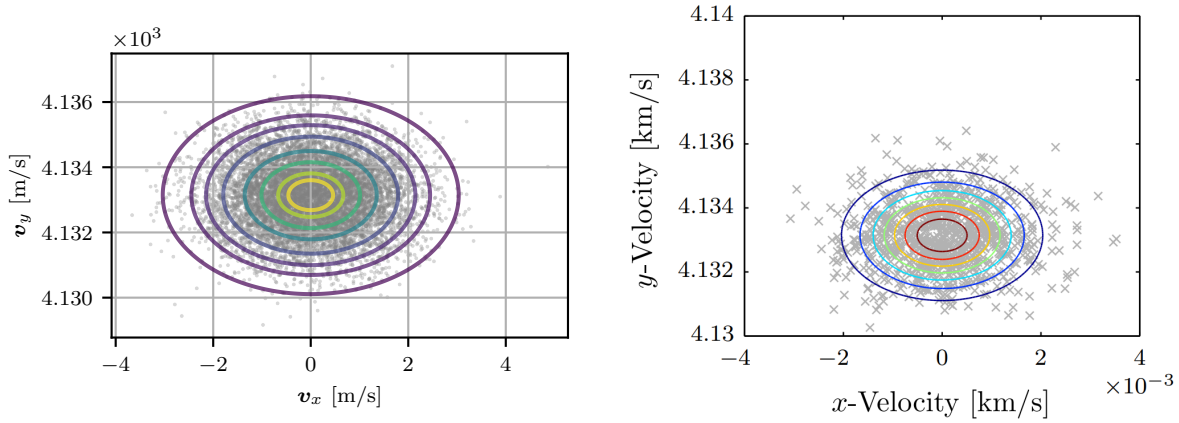
$$\boldsymbol{P} = \begin{bmatrix} 10^6 & 0 & 0 & 0 & 0 & 0 \\ 0 & 10^6 & 0 & 0 & 0 & 0 \\ 0 & 0 & 10^6 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} m^2 & \frac{m^2}{s} \\ \frac{m^2}{s} & \frac{m^2}{2^2} \end{bmatrix} \tag{D.1}$$

In Figures D.1 to D.8 the results are compared between the implemented AEGIS method and the SGMUKF method documented in [108] for both the position and velocity elements in the projected x-y plane for several propagation times. The results clearly demonstrate that the AEGIS method provides accurate and correct results for this test case scenario, and functions as expected.



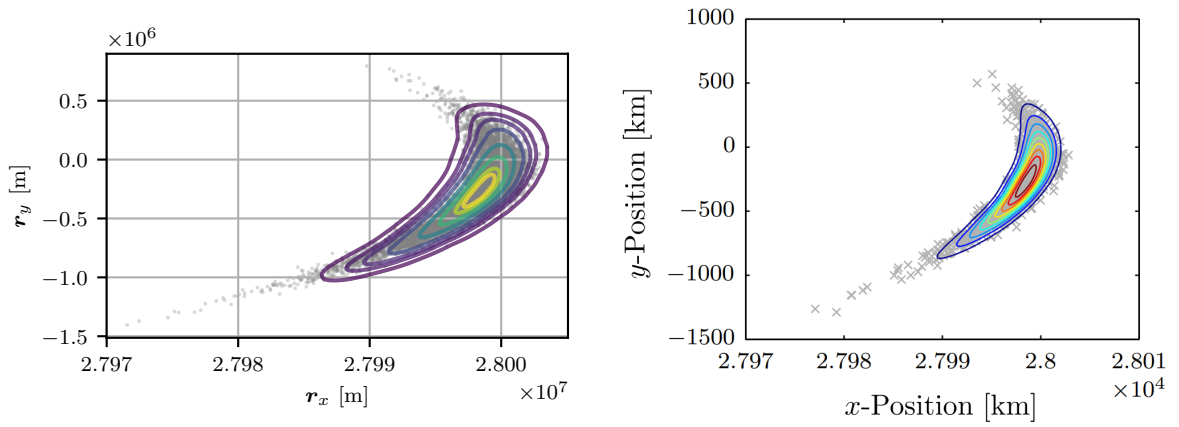
(a) *Own AEGIS implementation* **(b)** *SGMUKF from [108]*

Figure D.1: *Position (x-y projection) PDF contours with Monte Carlo samples at the initial epoch.*



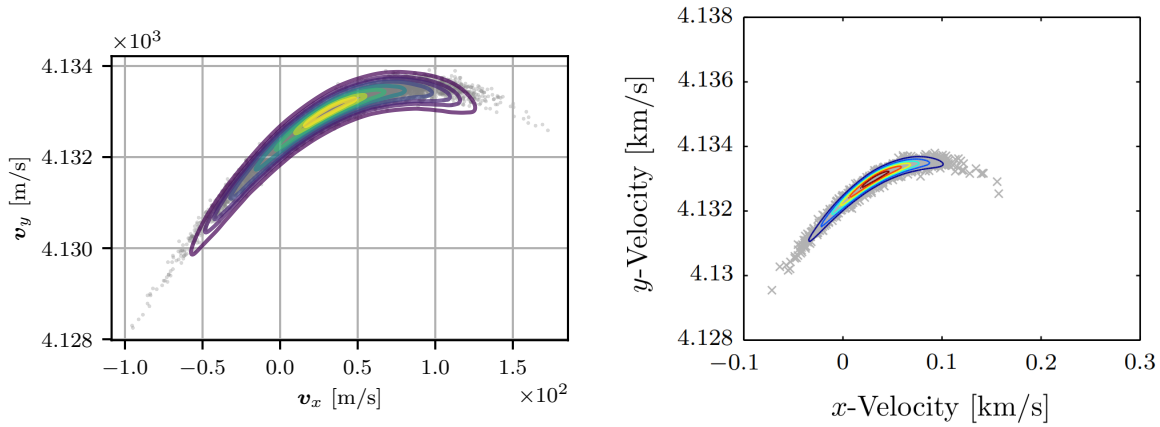
(a) *Own AEGIS implementation* **(b)** *SGMUKF from [108]*

Figure D.2: *Velocity (x-y projection) PDF contours with Monte Carlo samples at the initial epoch.*



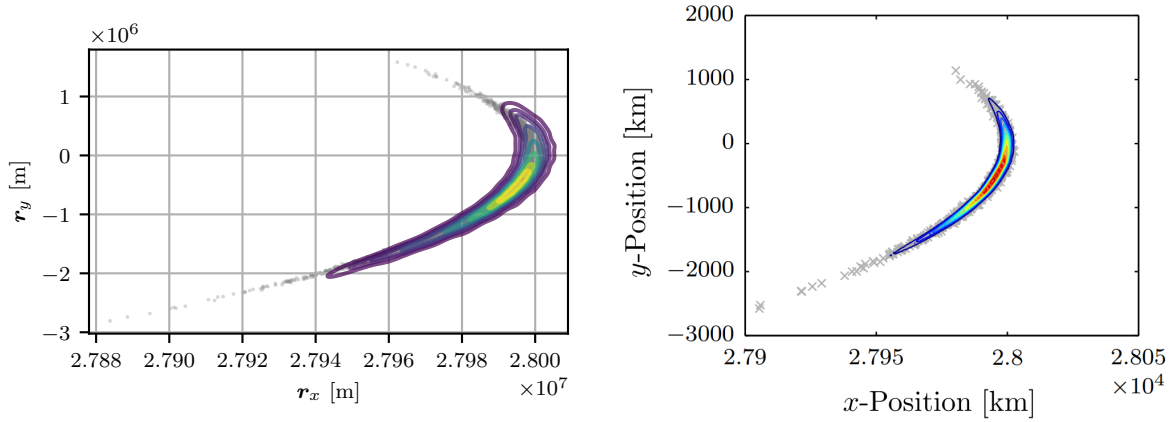
(a) *Own AEGIS implementation* **(b)** *SGMUKF from [108]*

Figure D.3: *Position (x-y projection) PDF contours with Monte Carlo samples after 1 orbit.*



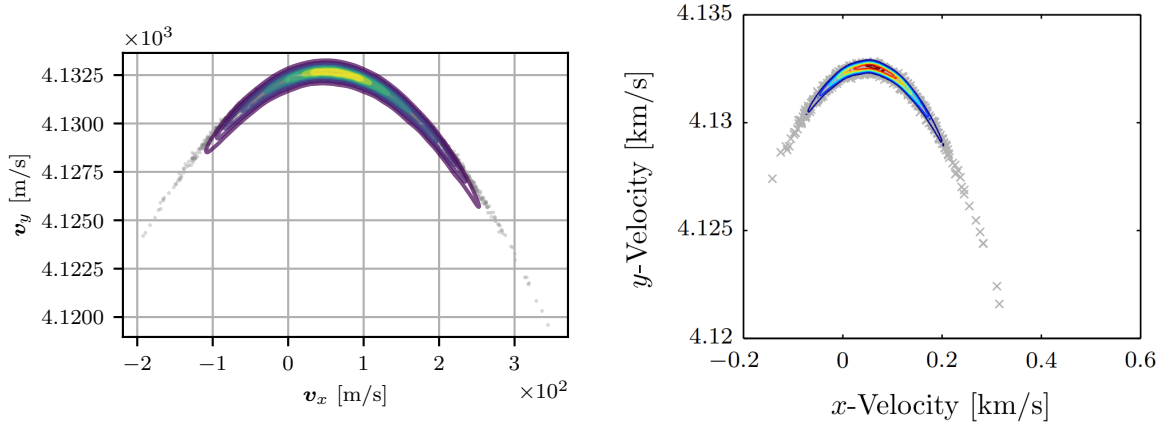
(a) *Own AEGIS implementation* **(b)** *SGMUKF from [108]*

Figure D.4: *Velocity (x-y projection) PDF contours with Monte Carlo samples after 1 orbit.*



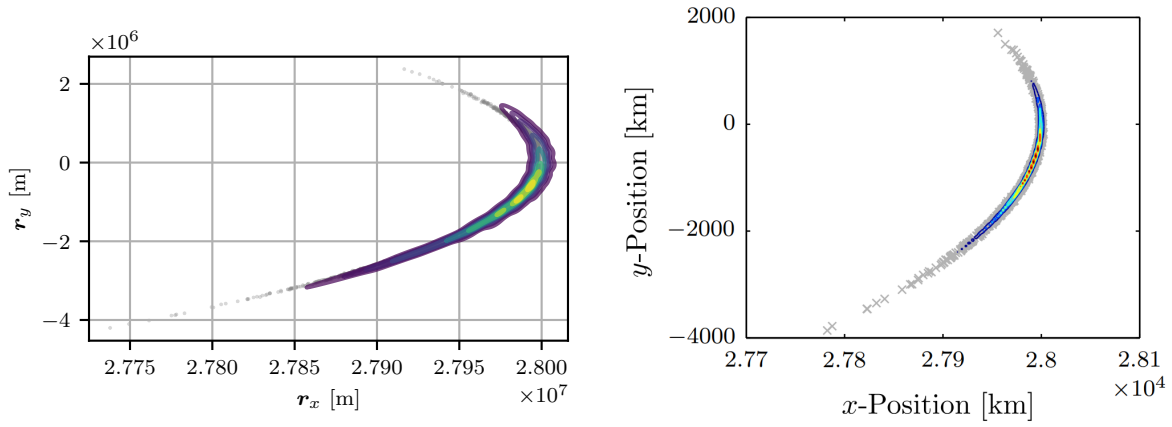
(a) *Own AEGIS implementation* **(b)** *SGMUKF from [108]*

Figure D.5: *Position (x-y projection) PDF contours with Monte Carlo samples after 2 orbits.*



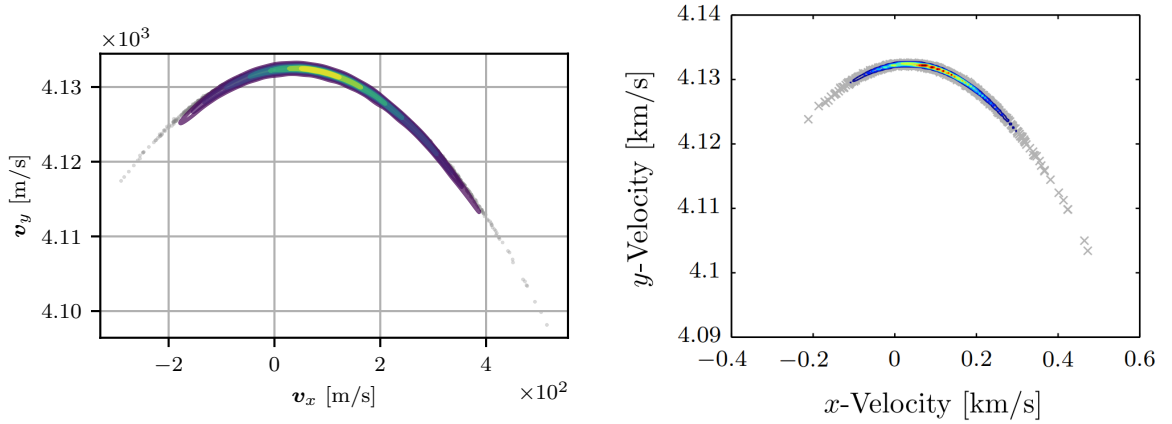
(a) *Own AEGIS implementation* **(b)** *SGMUKF from [108]*

Figure D.6: *Velocity (x-y projection) PDF contours with Monte Carlo samples after 2 orbits.*



(a) *Own AEGIS implementation* **(b)** *SGMUKF from [108]*

Figure D.7: *Position (x-y projection) PDF contours with Monte Carlo samples after 3 orbits.*

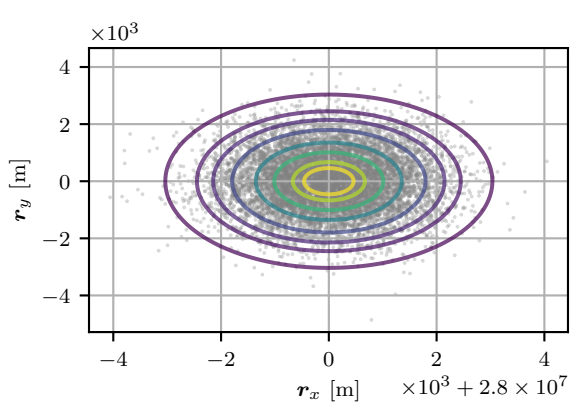


(a) *Own AEGIS implementation* **(b)** *SGMUKF from [108]*

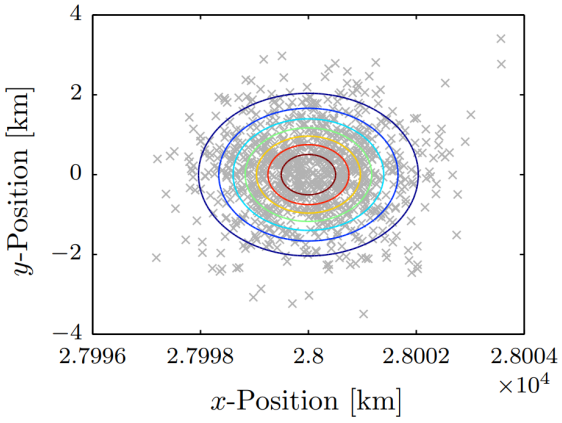
Figure D.8: *Velocity (x-y projection) PDF contours with Monte Carlo samples after 3 orbits.*

D.2. Unscented Transform verification

The implemented UT method has been tested and compared against known results to verify that it functions correctly and produces the expected results. In [108] the Unscented Kalman Filter (UKF) is described. This is essentially the filter version of the UT implemented in this research, described in Section 3.3. When the method is used to propagate an uncertainty forwards in time under the absence of observations, both methods function identically. The same test case scenario of an elliptical orbit from [108] described in Section D.1 is used to produce results using the implemented UT method. In Figures D.9 to D.16 the results are compared between the implemented UT method and the UKF method documented in [108] for both the position and velocity elements in the projected x-y plane for several propagation times. The results clearly demonstrate that the UT method provides accurate and correct results for this test case scenario, and functions as expected.

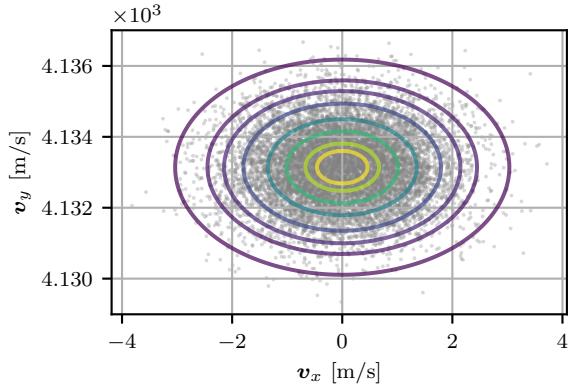


(a) *Own UT implementation*

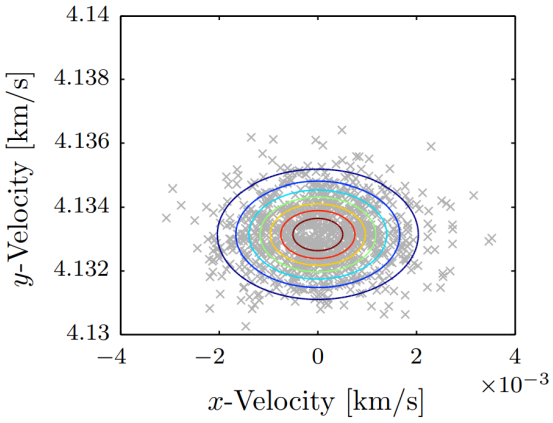


(b) *UKF from [108]*

Figure D.9: *Position (x-y projection) PDF contours with Monte Carlo samples at the initial epoch.*

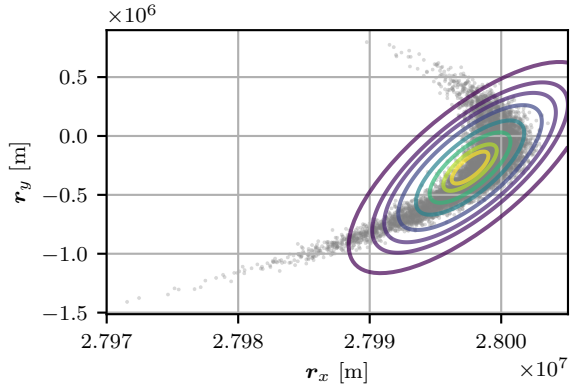


(a) *Own UT implementation*

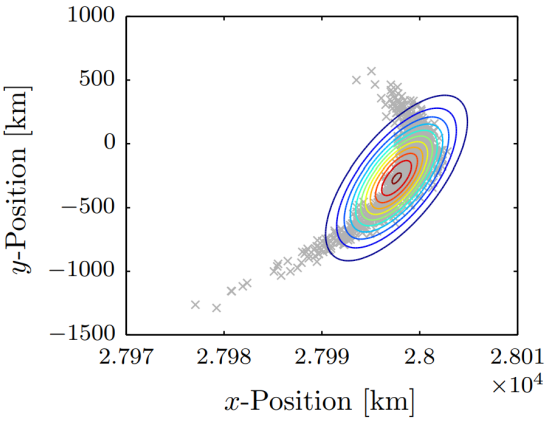


(b) *UKF from [108]*

Figure D.10: *Velocity (x-y projection) PDF contours with Monte Carlo samples at the initial epoch.*

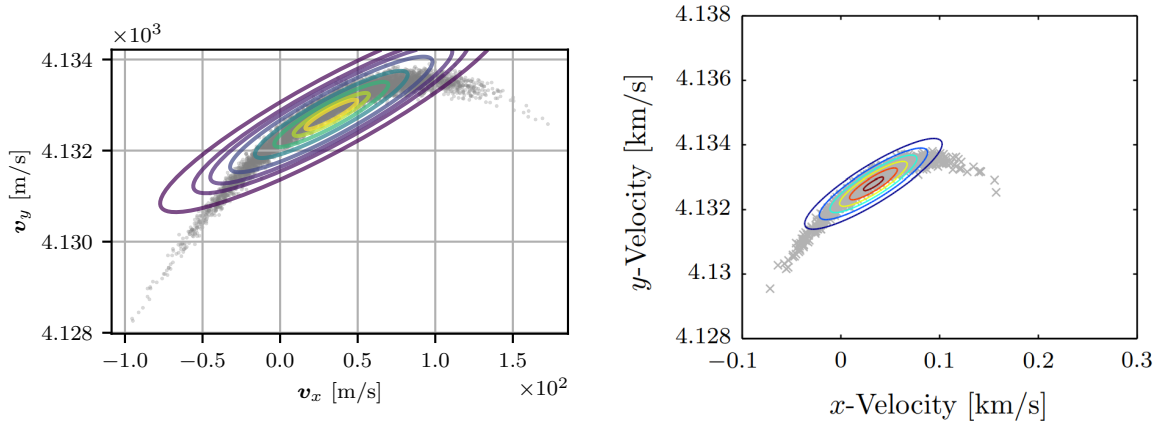


(a) *Own UT implementation*



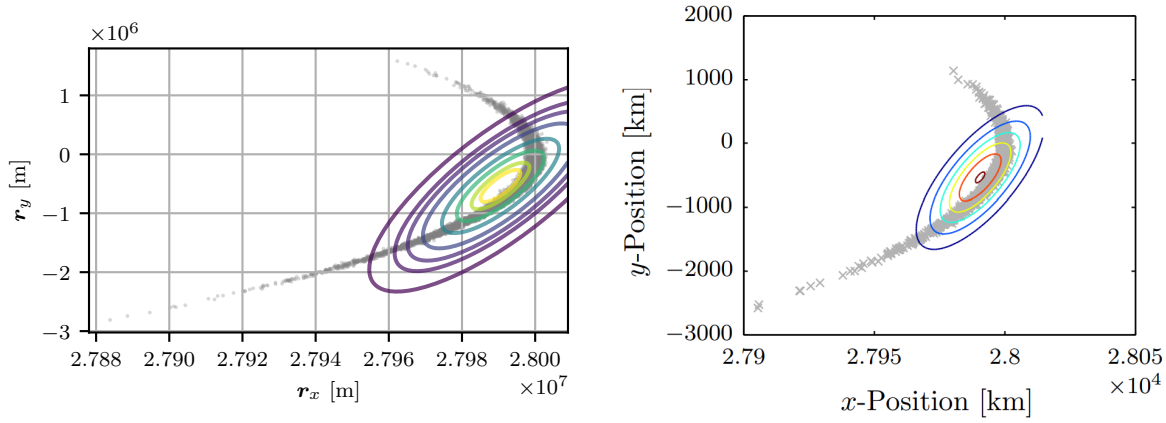
(b) *UKF from [108]*

Figure D.11: *Position (x-y projection) PDF contours with Monte Carlo samples after 1 orbit.*



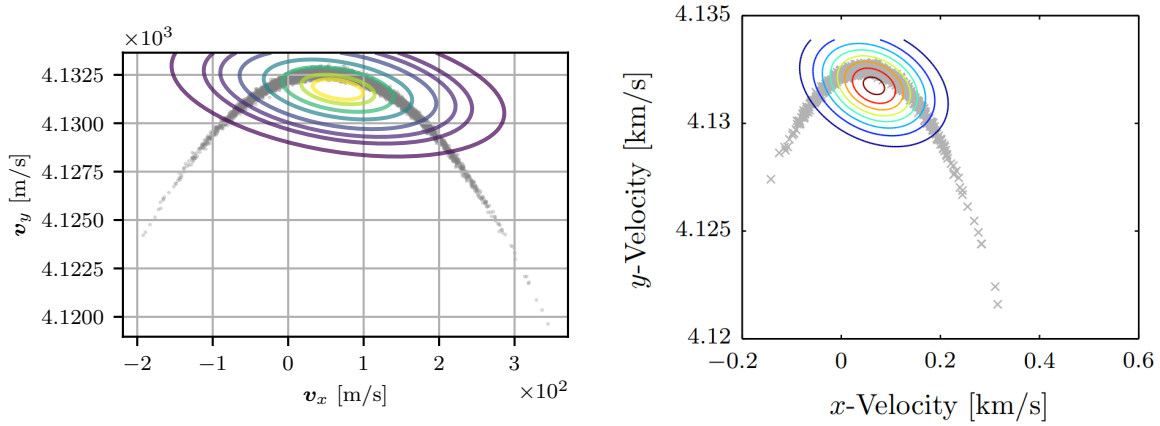
(a) *Own UT implementation* **(b)** *UKF from [108]*

Figure D.12: *Velocity (x-y projection) PDF contours with Monte Carlo samples after 1 orbit.*



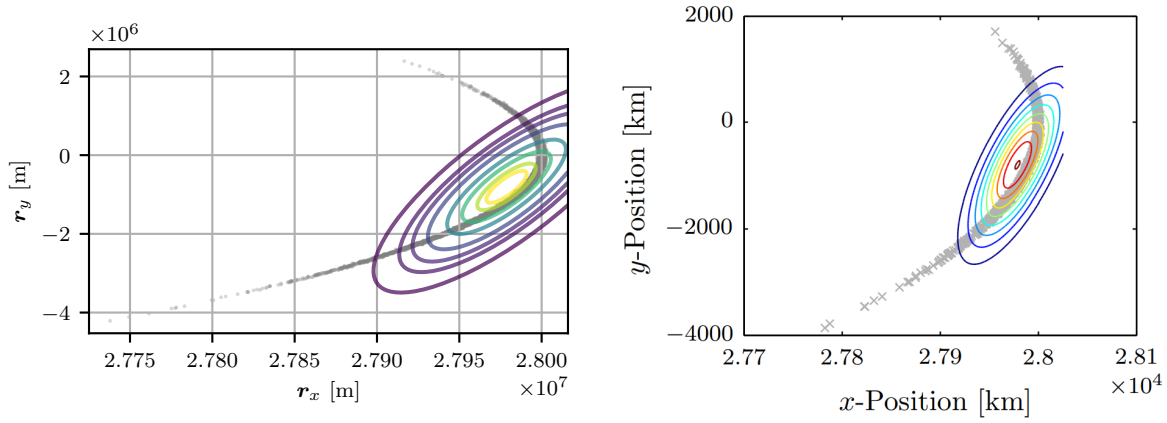
(a) *Own UT implementation* **(b)** *UKF from [108]*

Figure D.13: *Position (x-y projection) PDF contours with Monte Carlo samples after 2 orbits.*



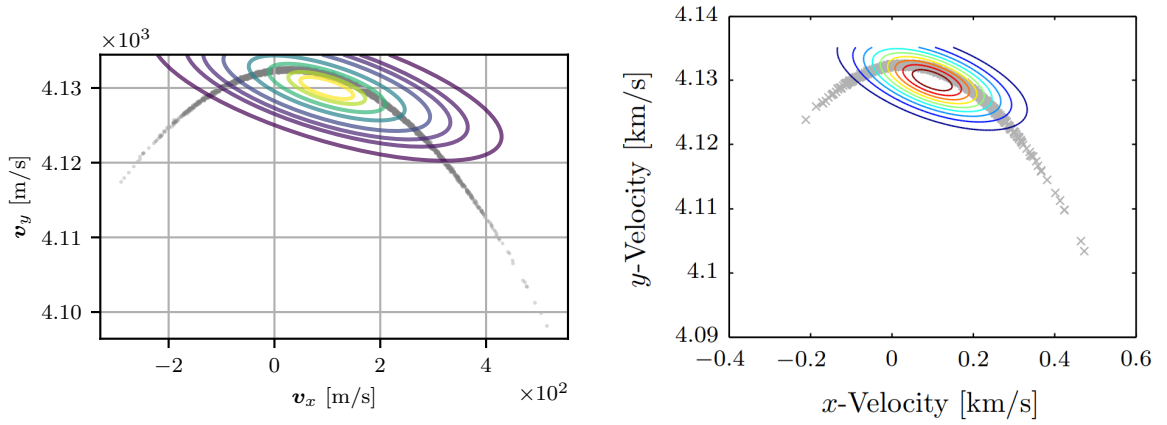
(a) *Own UT implementation* **(b)** *UKF from [108]*

Figure D.14: *Velocity (x-y projection) PDF contours with Monte Carlo samples after 2 orbits.*



(a) *Own UT implementation* **(b)** *UKF from [108]*

Figure D.15: *Position (x-y projection) PDF contours with Monte Carlo samples after 3 orbits.*

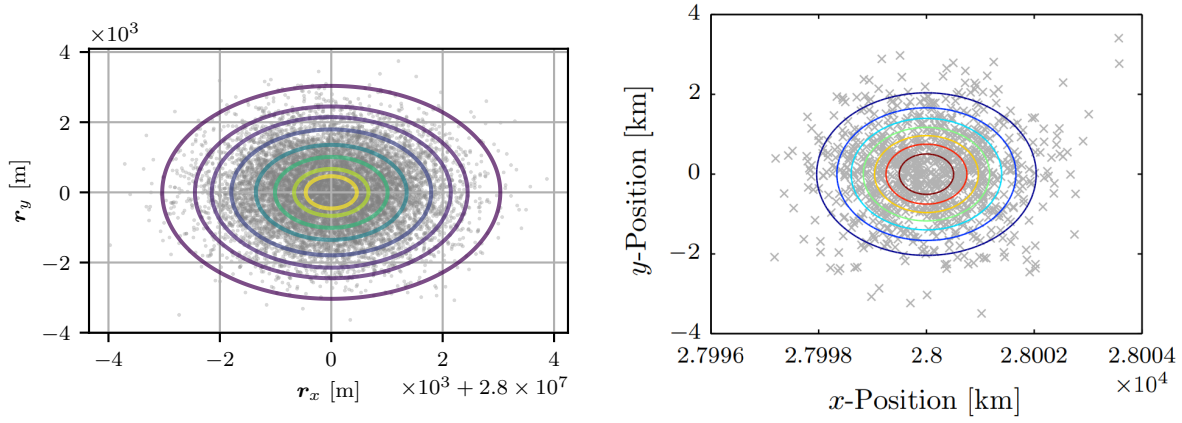


(a) *Own UT implementation* **(b)** *UKF from [108]*

Figure D.16: *Velocity (x-y projection) PDF contours with Monte Carlo samples after 3 orbits.*

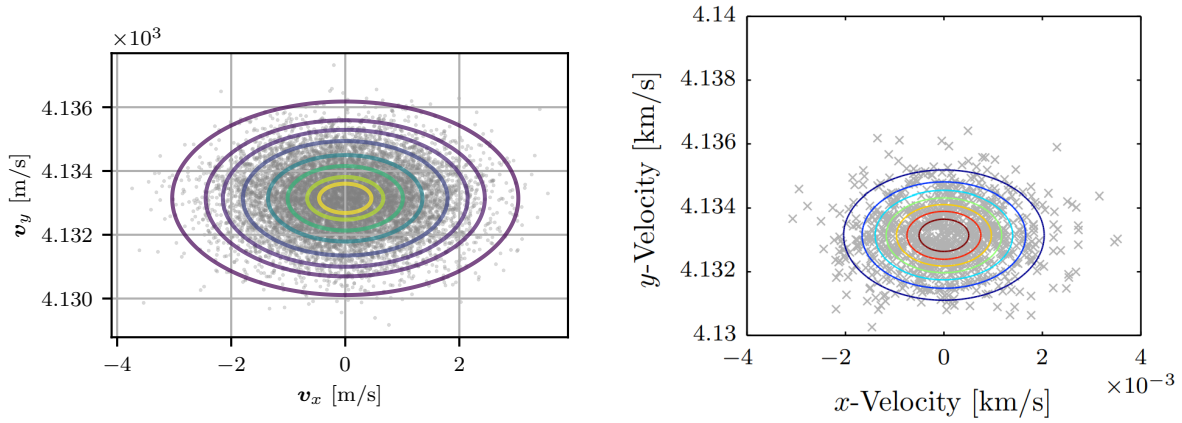
D.3. Linearised Covariance Method

The implemented LinCov method has been tested and compared against known results to verify that it functions correctly and produces the expected results. In [108] the Extended Kalman Filter (EKF) is described. This is essentially the filter version of the LinCov implemented in this research, described in Section 3.2. When the method is used to propagate an uncertainty forwards in time under the absence of observations, both methods function identically. The same test case scenario of an elliptical orbit from [108] described in Section D.1 is used to produce results for using the implemented LinCov method. In Figures D.17 to D.24 the results are compared between the implemented LinCov method and the EKF method documented in [108] for both the position and velocity elements in the projected x-y plane for several propagation times. The results clearly demonstrate that the LinCov method provides accurate and correct results for this test case scenario, and functions as expected.



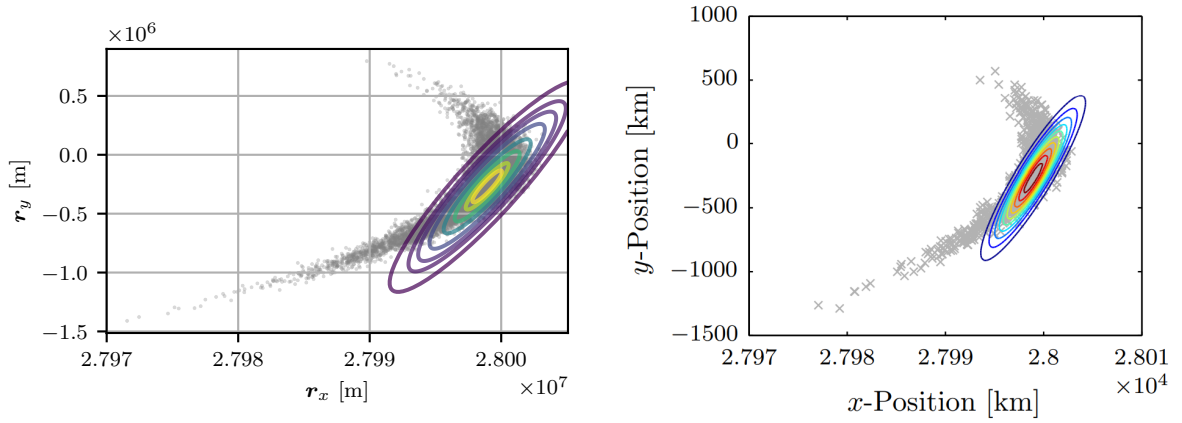
(a) *Own LinCov implementation* **(b)** *EKF from [108]*

Figure D.17: *Position (x-y projection) PDF contours with Monte Carlo samples at the initial epoch.*



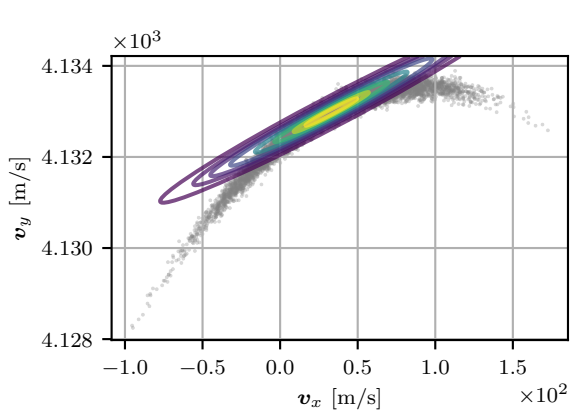
(a) *Own LinCov implementation* **(b)** *EKF from [108]*

Figure D.18: *Velocity (x-y projection) PDF contours with Monte Carlo samples at the initial epoch.*

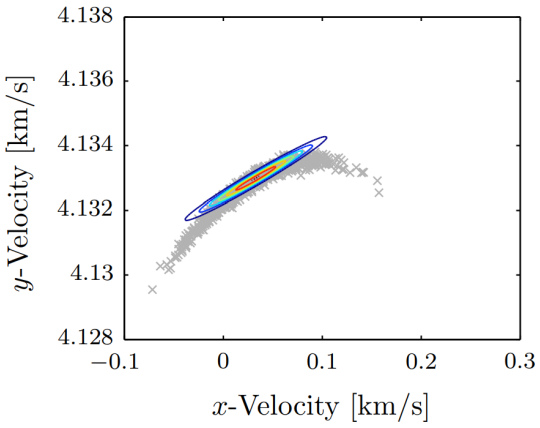


(a) *Own LinCov implementation* **(b)** *EKF from [108]*

Figure D.19: *Position (x-y projection) PDF contours with Monte Carlo samples after 1 orbit.*

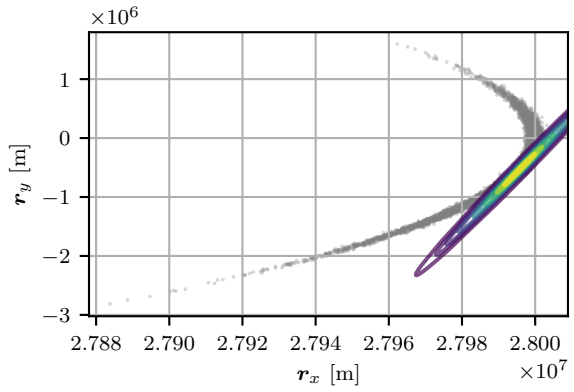


(a) *Own LinCov implementation*

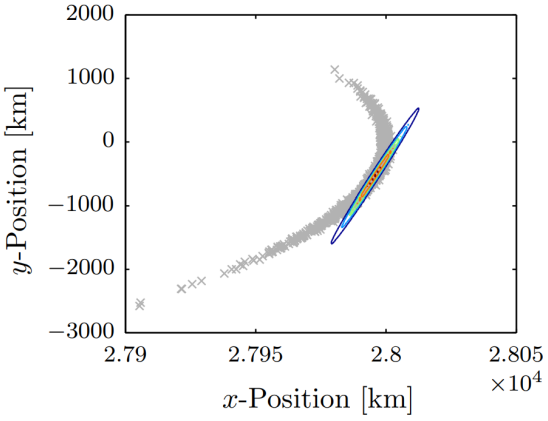


(b) *EKF from [108]*

Figure D.20: *Velocity (x-y projection) PDF contours with Monte Carlo samples after 1 orbit.*

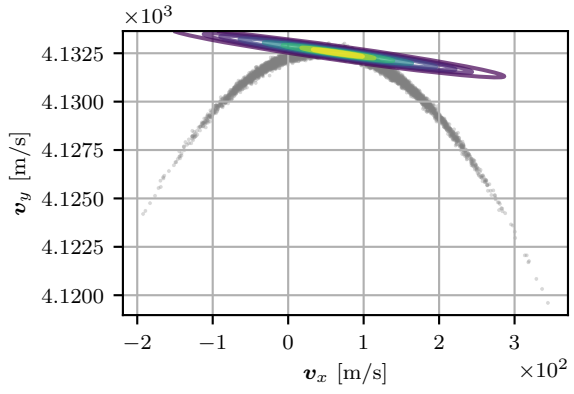


(a) *Own LinCov implementation*

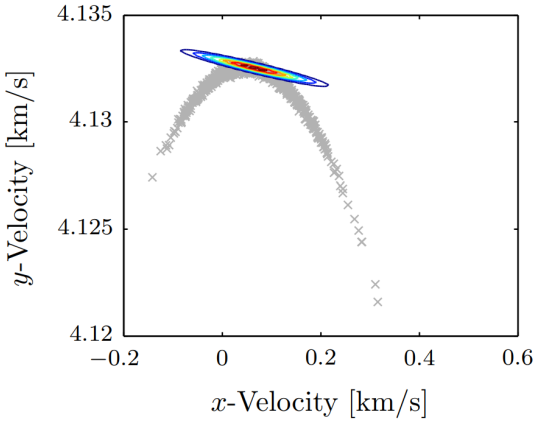


(b) *EKF from [108]*

Figure D.21: *Position (x-y projection) PDF contours with Monte Carlo samples after 2 orbits.*

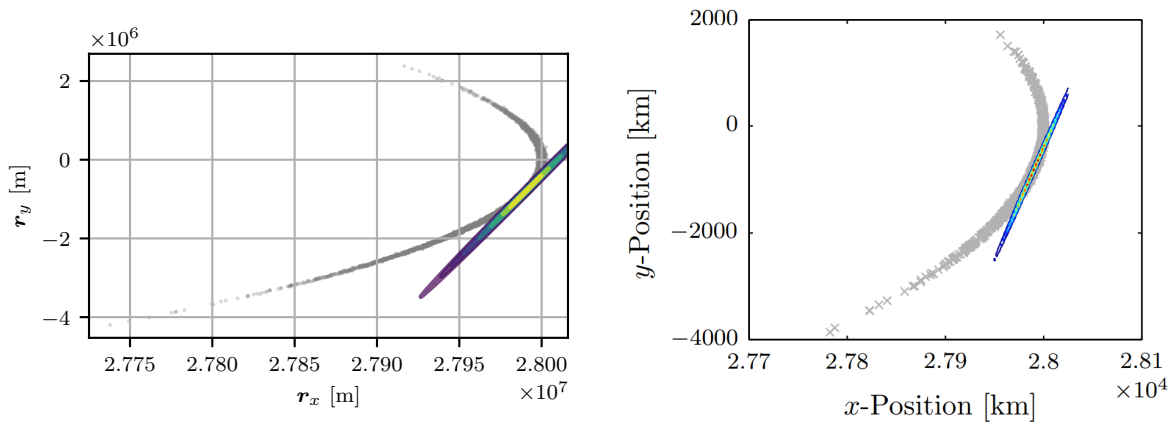


(a) *Own LinCov implementation*



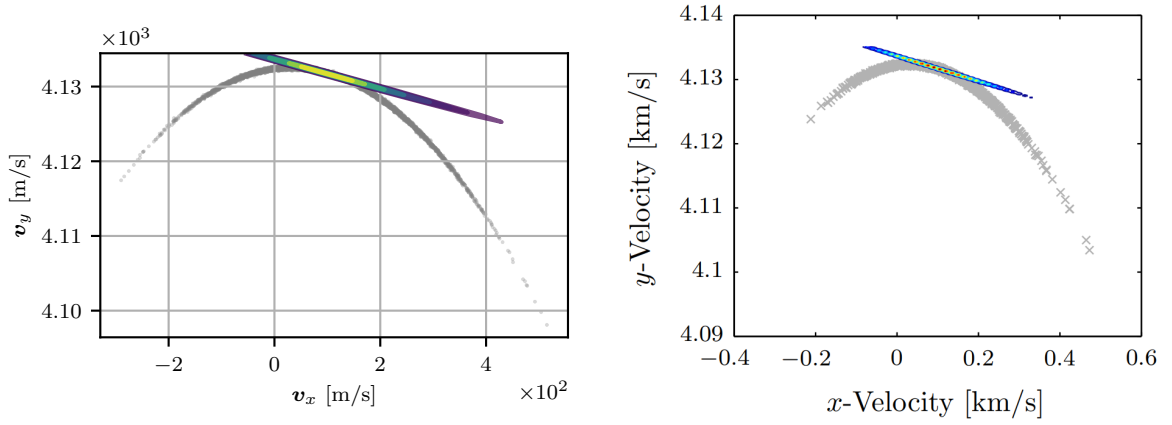
(b) *EKF from [108]*

Figure D.22: *Velocity (x-y projection) PDF contours with Monte Carlo samples after 2 orbits.*



(a) *Own LinCov implementation* **(b)** *EKF from [108]*

Figure D.23: *Position (x-y projection) PDF contours with Monte Carlo samples after 3 orbits.*



(a) *Own LinCov implementation* **(b)** *EKF from [108]*

Figure D.24: *Velocity (x-y projection) PDF contours with Monte Carlo samples after 3 orbits.*

D.4. Multi-Fidelity verification

The implementation of the MF method closely follows the description by Jones et al. [71]. In [71] paper, results for the MF method are shown for three test cases, with their initial distributions described by the following sets of Kepler elements:

	$a\ [m]$	$e\ [-]$	$i\ [deg]$	$\Omega\ [deg]$	$\omega\ [deg]$	$\theta\ [deg]$
Test Case 1	7078000	0.01	30	0	0	0
Test Case 2	26562000	0.74	63.4	0	0	0
Test Case 3	6878000	7.7E-4	45	0	90	0
σ	10000	0.001	$\frac{1}{36}$	$\frac{1}{36}$	$\frac{1}{36}$	$\frac{1}{36}$

Table D.2: *Keplerian elements of the initial distributions (mean vector and 1 standard deviation σ) of the three test cases in [71]*

The integrator analysis for these test case scenarios is combined with a set of acceleration models provided in [71], which are used in the verification to reach a better comparison. The accelerations are repeated here in Table D.3.

Model	Low-Fidelity	High-Fidelity
Central Body Gravity	Two-Body and J2	70×70
Third-Body Perturbations	None	Sun and Moon
Solar Radiation Pressure	None	Cannonball
Atmospheric Drag	None	Cannonball

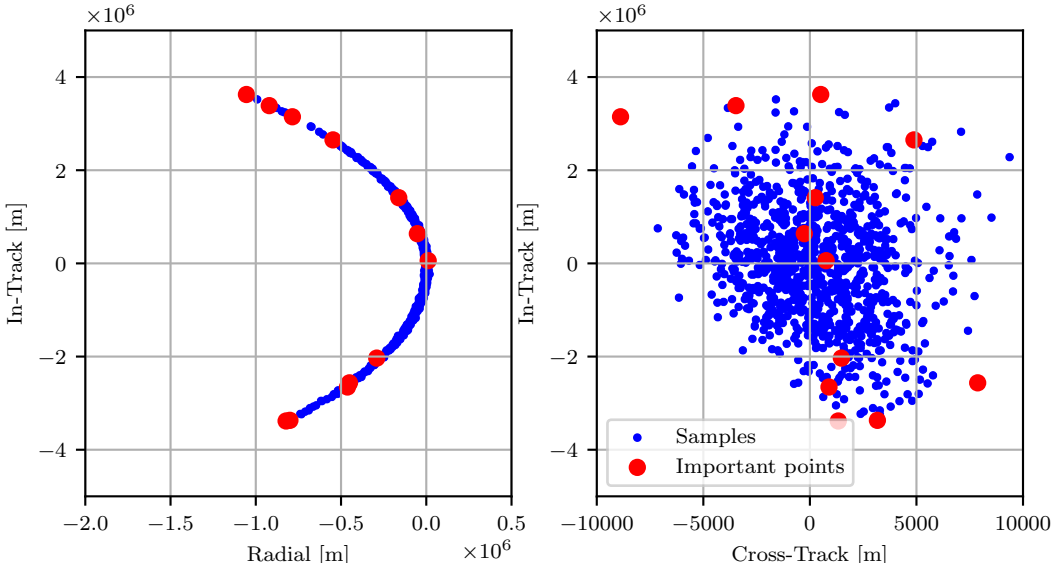
Table D.3: *Acceleration models used for low- and high-fidelity dynamics [71].*

Finally, the satellite parameters used are:

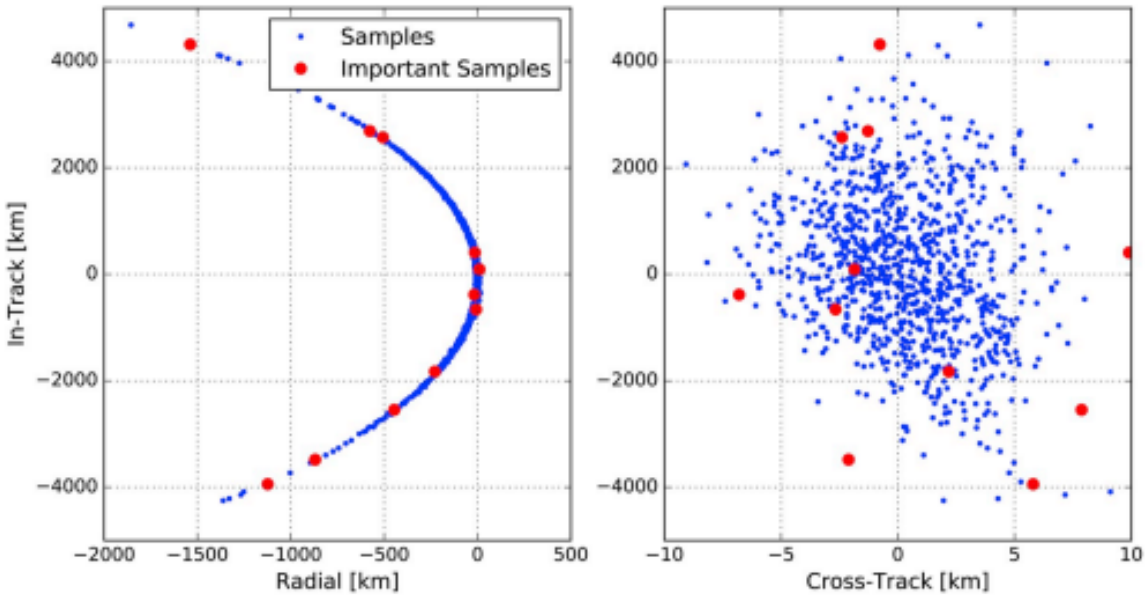
Satellite Mass	500 kg
Satellite Drag and SRP Area	1 m^2
C_D	2.0
C_R	1.5
t_0	2000-01-29 22:00:00.500 UTC

Table D.4: *Satellite parameters used for the test case scenarios [71].*

In Figure D.25 the results are compared between the implemented MF method and the results documented in [71] for all three test case scenarios. The results are presented in the RSW coordinate frame, based around the propagated mean vector. The good agreement between the results demonstrates that the implemented MF method provides accurate and correct results for these test case scenarios, and functions as expected.

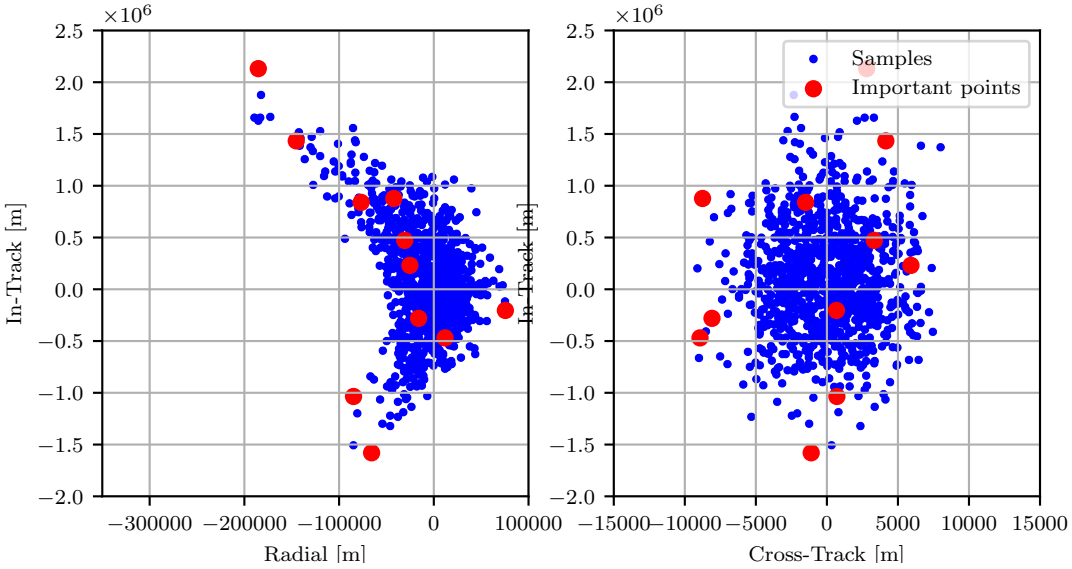


(a) *Implemented MF.*

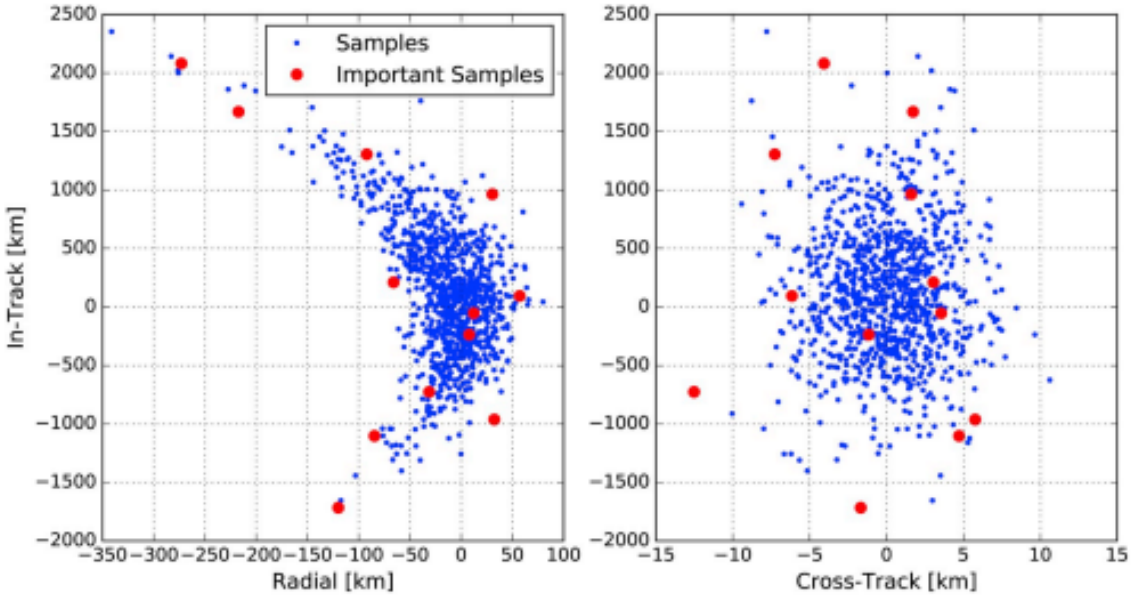


(b) *MF comparison [71]*

Figure D.25: *Scatter of propagated samples in the RSW frame around the propagated mean vector, for test case 1, at t = 24 hours.*

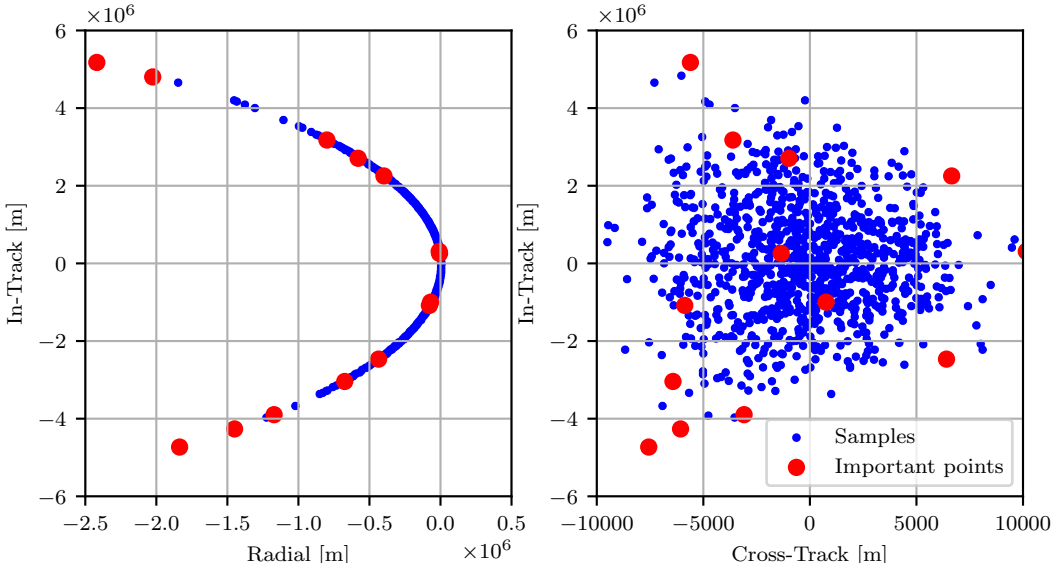


(a) *Implemented MF.*

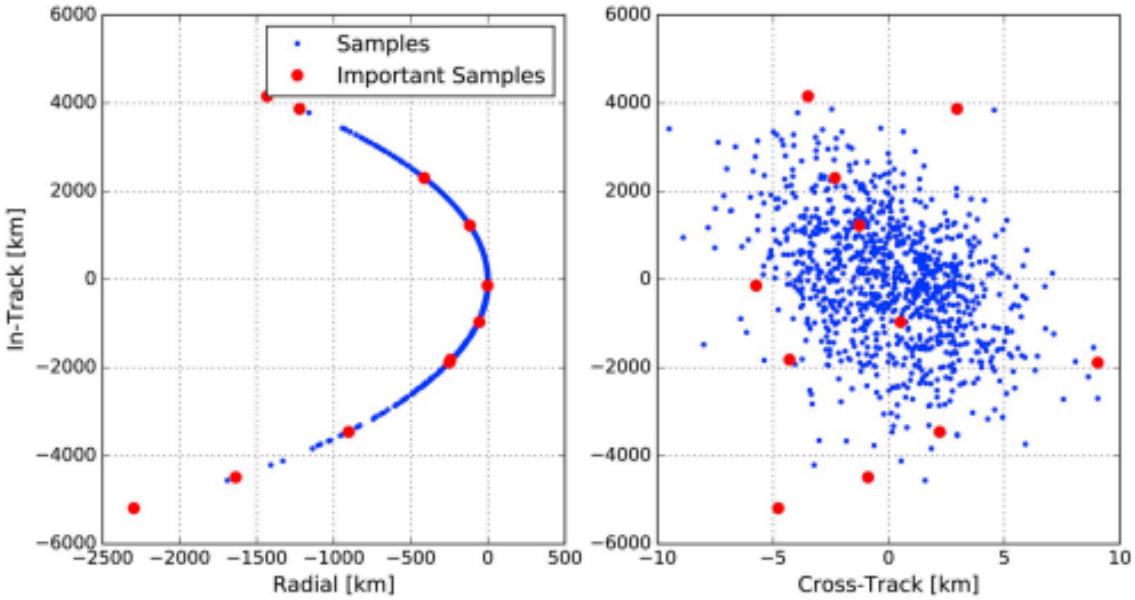


(b) *MF comparison [71]*

Figure D.26: *Scatter of propagated samples in the RSW frame around the propagated mean vector, for test case 2, at $t \approx 0.992$ days.*



(a) *Implemented MF.*



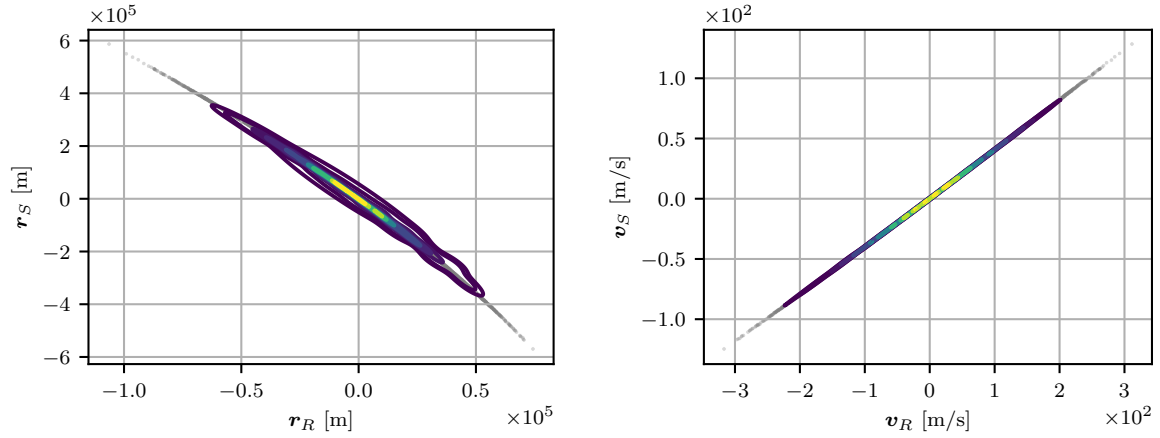
(b) *MF comparison [71]*

Figure D.27: *Scatter of propagated samples in the RSW frame around the propagated mean vector, for test case 3 at t = 24 hours.*

D.5. Polynomial Chaos Expansion verification

To verify that the implementation of the PCEs produces the correct results, it is compared directly against Monte Carlo samples for various different settings. The test case scenario chosen for this verification is the highly elliptical orbit of the Spirale scenario, after 2 orbital revolutions. The results generated with PCEs, shown with contour plots, are compared against 10^5 MC samples, shown in gray. The results are shown in the Radial (R) and Along-track (S) frame, for both the positions and velocities.

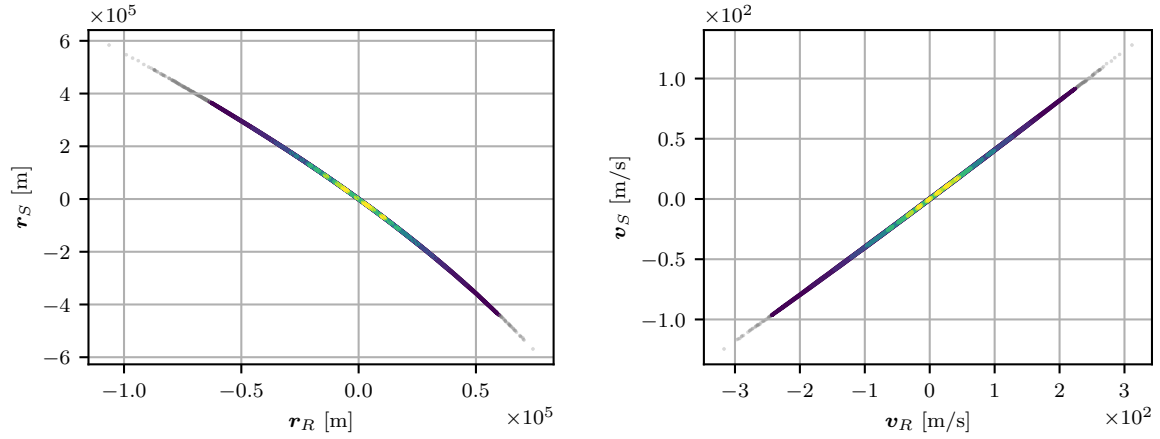
From Figures D.28 and D.29 it is seen that, for the regression method, the final expansion can not fully approximate the right shape of the MC samples yet for an N/P ratio below 1, whereas an N/P ratio of 1 or higher does generate the same distribution as when samples are propagated using MC. From Figures D.30 and D.31 it is seen that, for the spectral projection method, an expansion of order 1 can not approximate the nonlinear shape of the final uncertainty as shown by the MC samples, whereas an expansion order of 2 can generate a similar distribution as when using MC samples. The clear convergence to the MC samples and the good agreement between PCEs with sufficient values for N/P and P_e and MC samples demonstrates that the implementation of the PCEs functions as expected.



(a) *2D position in the RSW frame.*

(b) *2D velocity in the RSW frame.*

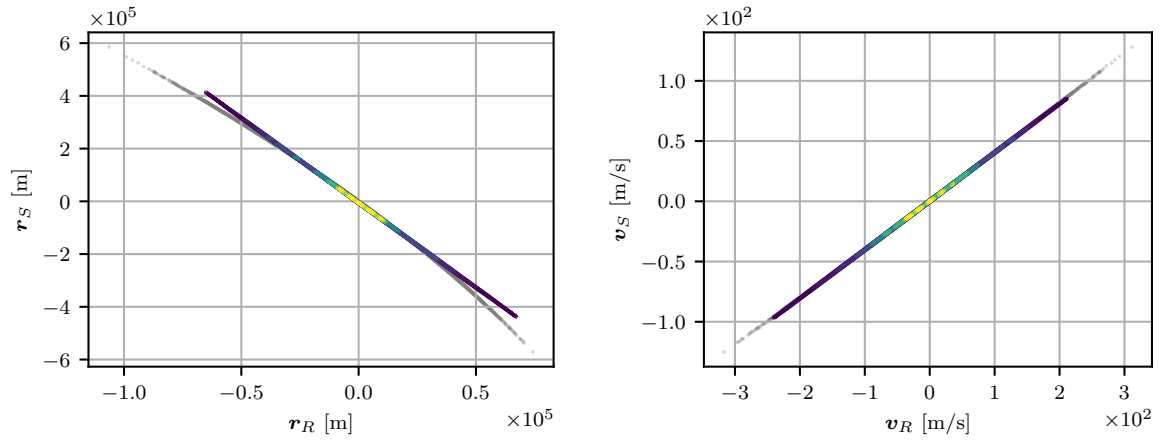
Figure D.28: *Contour plots of a PCE using regression with $P_e = 3$ and $N/P = 0.9$ compared against 10^5 MC samples in the RSW frame.*



(a) *2D position in the RSW frame.*

(b) *2D velocity in the RSW frame.*

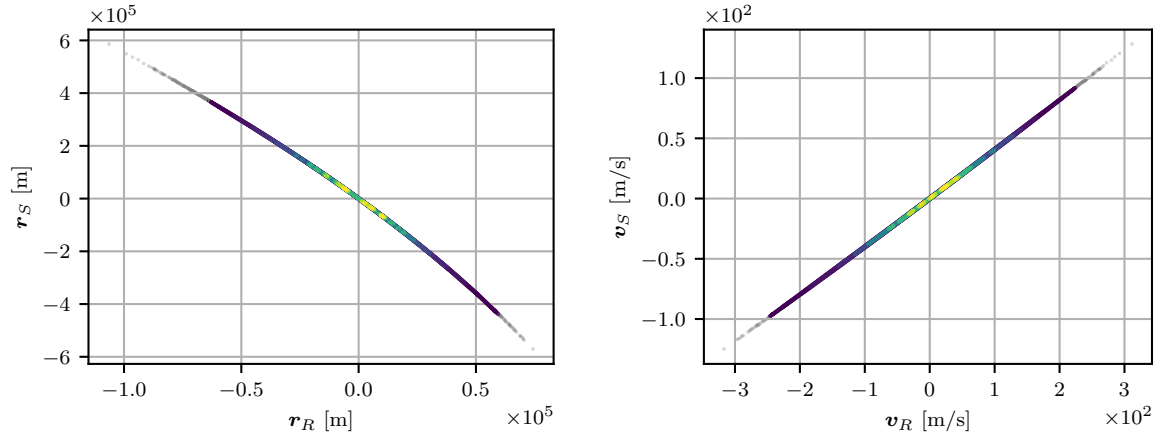
Figure D.29: *Contour plots of a PCE using regression with $P_e = 3$ and $N/P = 1$ compared against 10^5 MC samples in the RSW frame.*



(a) *2D position in the RSW frame.*

(b) *2D velocity in the RSW frame.*

Figure D.30: *Contour plots of a PCE using spectral projection with $P_e = 1$ and $Q = 3$ compared against 10^5 MC samples in the RSW frame.*



(a) *2D position in the RSW frame.*

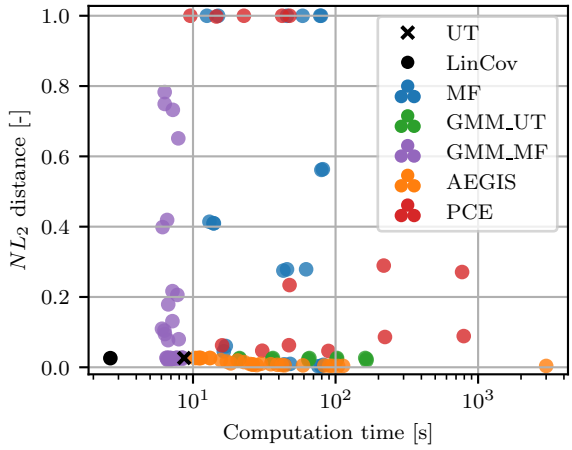
(b) *2D velocity in the RSW frame.*

Figure D.31: *Contour plots of a PCE using spectral projection with $P_e = 2$ and $Q = 3$ compared against 10^5 MC samples in the RSW frame.*

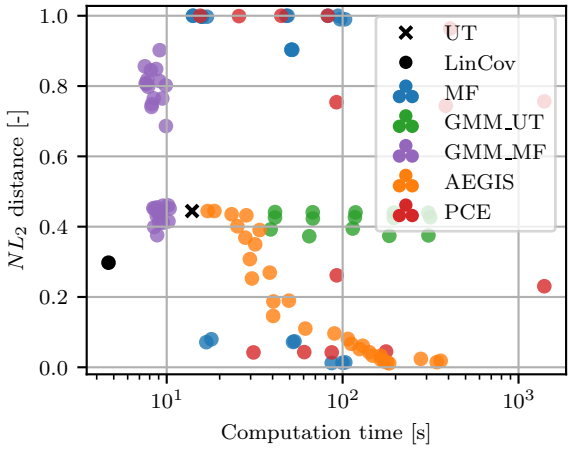
E

Uncertainty Propagation Results

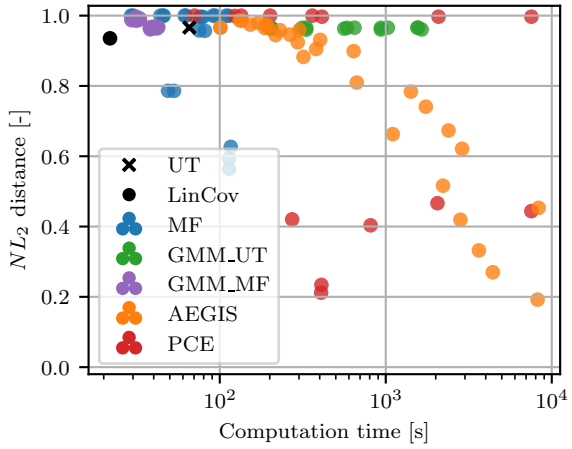
The following pages of this appendix contain the results to the uncertainty propagation comparison study. For each scenario and number of orbital revolutions, the NL_2 distance of the propagated uncertainties are shown against the computation time, for all the uncertainty propagations included in the comparison study. The NL_2 distance is a normalised metric that represents the distance of the propagated uncertainty to that propagated with MC samples, where a value of 0 represents identical distributions, and a value of 1 indicates no agreement at all.



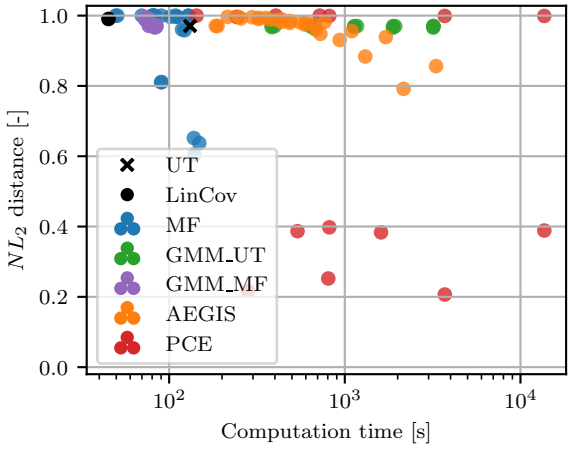
(a) *Velox at 1 revolution.*



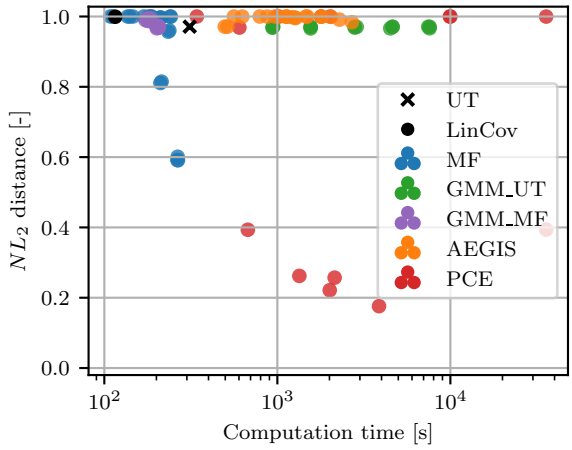
(b) *Velox at 2 revolutions.*



(c) *Velox at 10 revolutions.*

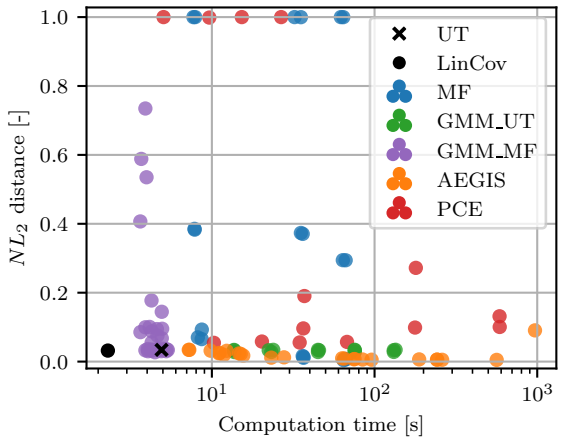


(d) *Velox at 20 revolutions.*

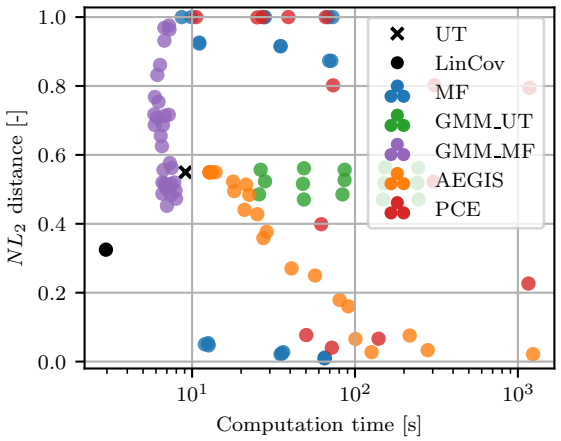


(e) *Velox at 50 revolutions.*

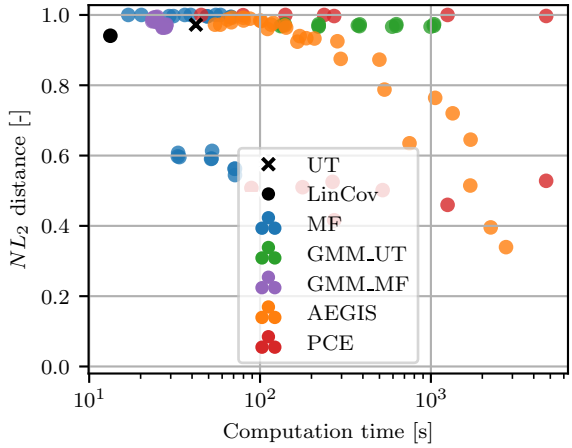
Figure E.1: *NL_2 norm compared against MC samples, versus the propagation time for different UP methods with various settings, for 1, 2, 10, 20 and 50 orbital revolutions of the Velox scenario.*



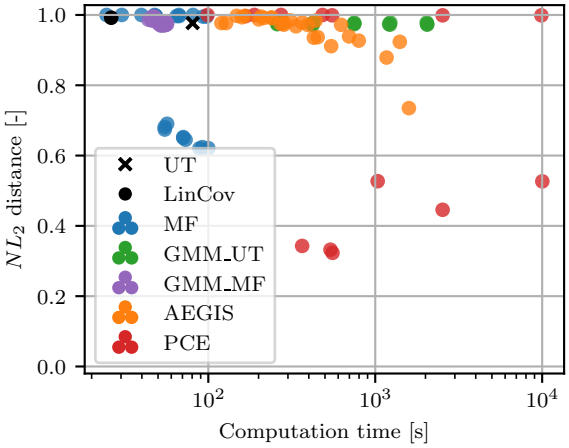
(a) *Oneweb at 1 revolution.*



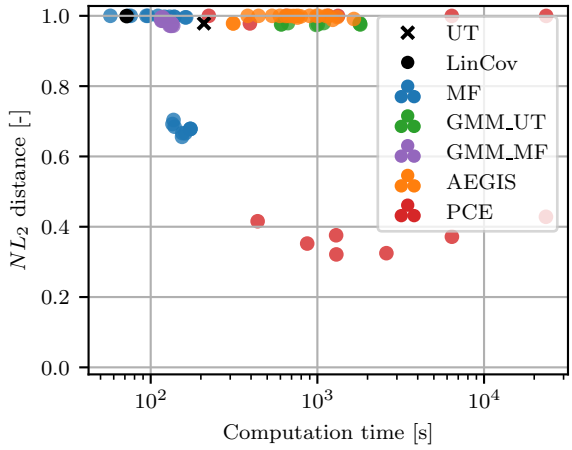
(b) *Oneweb at 2 revolutions.*



(c) *Oneweb at 10 revolutions.*

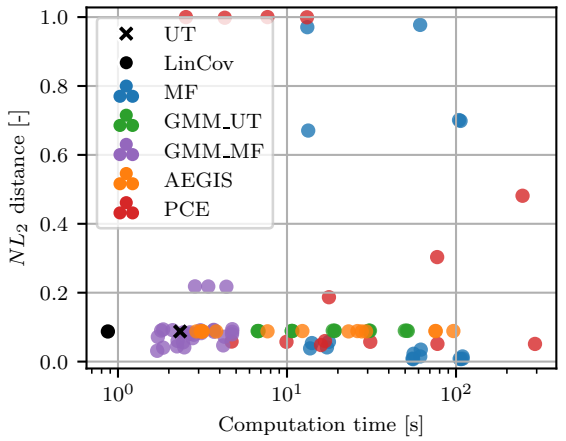


(d) *Oneweb at 20 revolutions.*

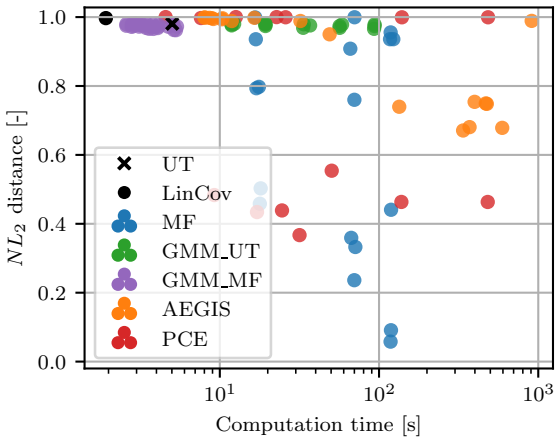


(e) *Oneweb at 50 revolutions.*

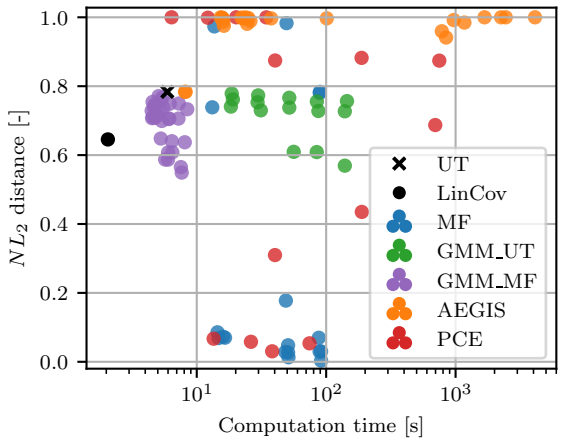
Figure E.2: *NL_2 norm compared against MC samples, versus the propagation time for different UP methods with various settings, for 1, 2, 10, 20 and 50 orbital revolutions of the Oneweb scenario.*



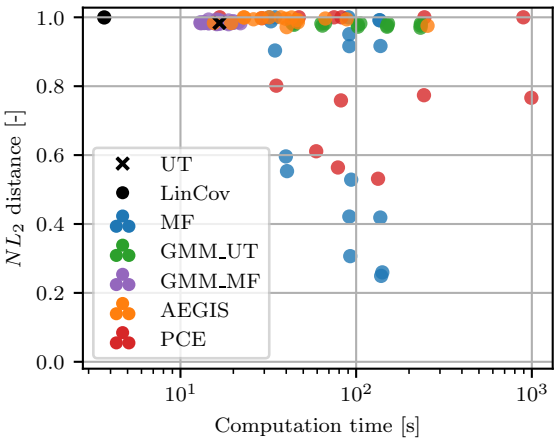
(a) *Cosmos at 0.5 revolutions.*



(b) *Cosmos at 1 revolution.*

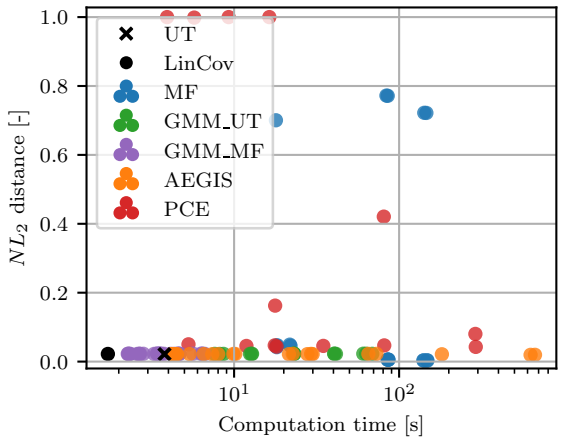


(c) *Cosmos at 1.5 revolutions.*

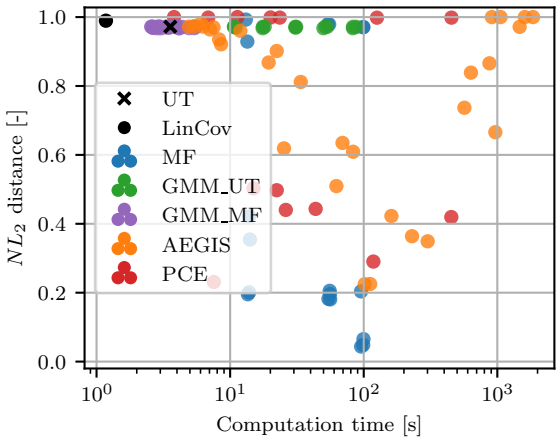


(d) *Cosmos at 2 revolutions.*

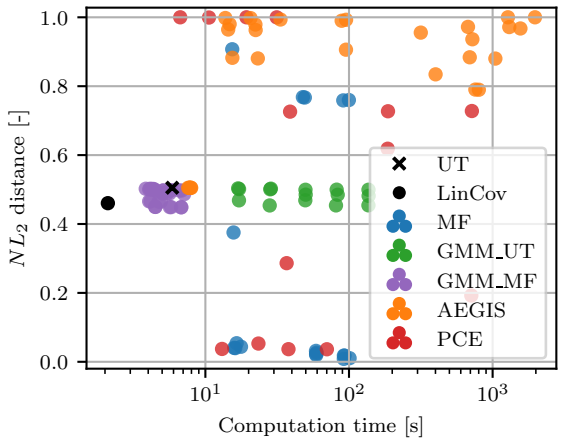
Figure E.3: *NL_2 norm compared against MC samples, versus the propagation time for different UP methods with various settings, for 0,5, 1, 1.5 and 2 orbital revolutions of the Cosmos scenario.*



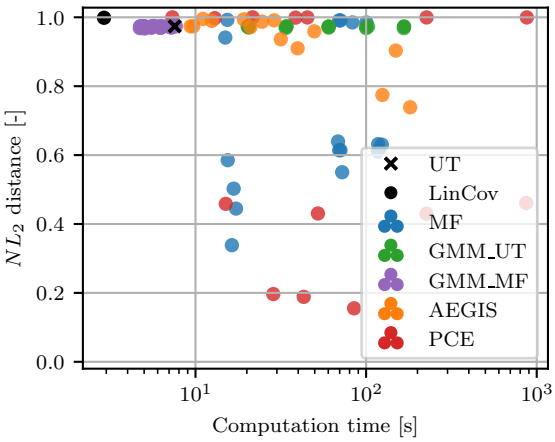
(a) *Spirale at 0.5 revolutions.*



(b) *Spirale at 1 revolution.*

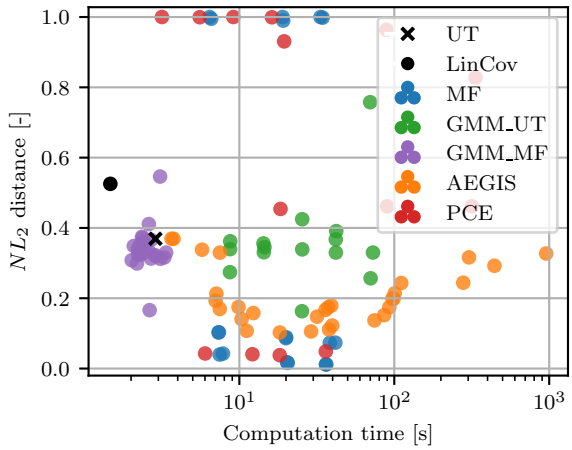


(c) *Spirale at 1.5 revolutions.*

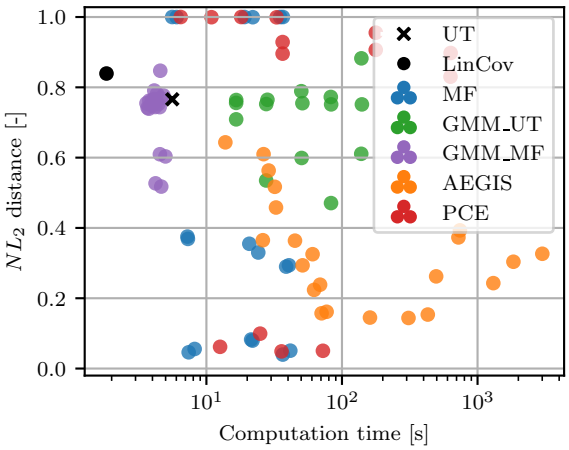


(d) *Spirale at 2 revolutions.*

Figure E.4: *NL_2 norm compared against MC samples, versus the propagation time for different UP methods with various settings, for 0,5, 1, 1.5 and 2 orbital revolutions of the Spirale scenario.*



(a) *Meteosat at 1 revolution.*



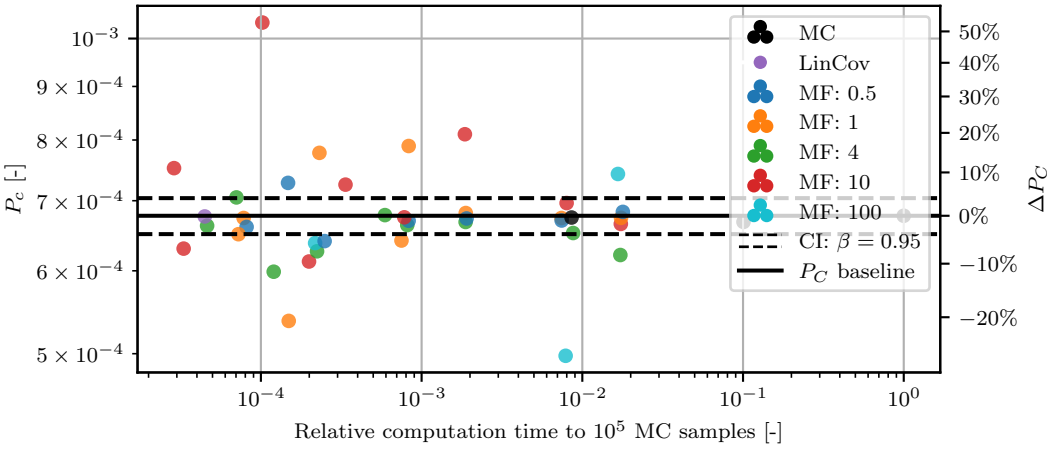
(b) *Meteosat at 2 revolutions.*

Figure E.5: *NL_2 norm compared against MC samples, versus the propagation time for different UP methods with various settings, for 1 and 2 orbital revolutions of the Meteosat scenario.*

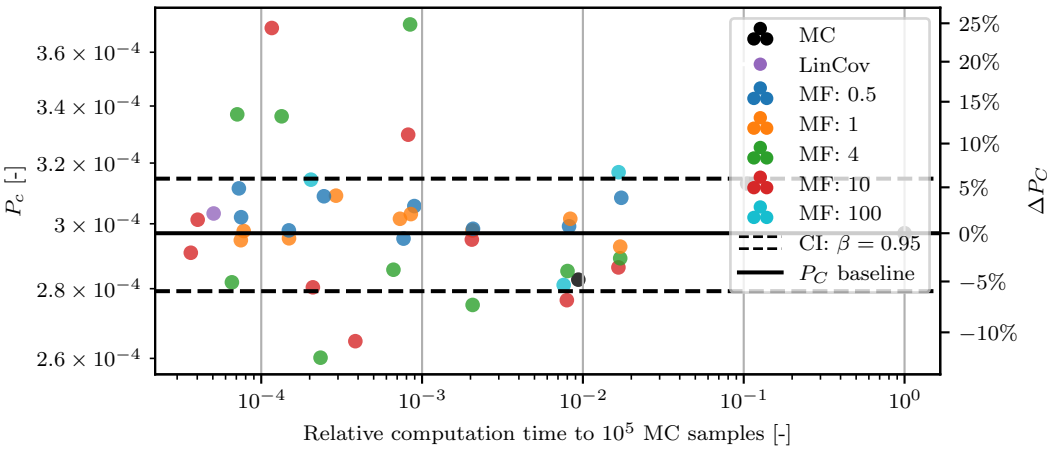
F

Conjunction Assessment Results

In this appendix, the results of the conjunction assessment study are shown for all 4 scenarios: the short and long versions of the high- and low-velocity scenarios. The results include the collision probability after various uncertainty propagations of the two objects shown against the relative computation time compared to the baseline, as well as the collision probability after MF propagations shown against the number of important samples used in the propagation.

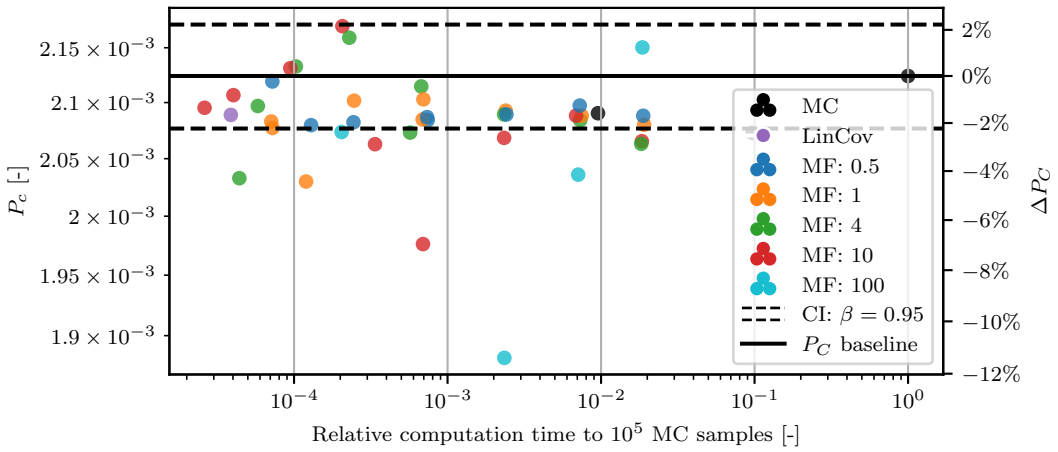


(a) *Short version* $\Delta t \approx 48h$..

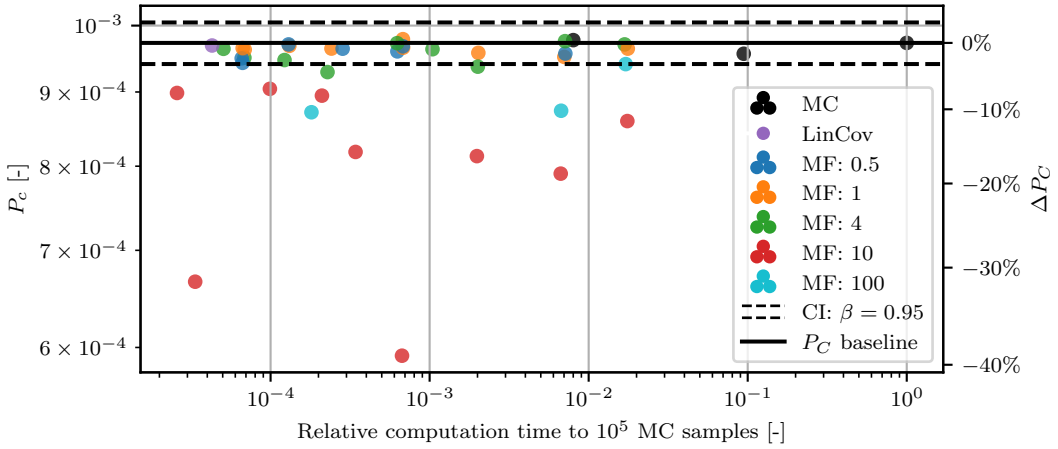


(b) *Long version* $\Delta t \approx 96h$.

Figure F.1: *The probability of collision between the uncertainties propagated using various methods and settings compared to the baseline P_C based on the uncertainties propagated with 10^5 MC samples, for the short and long version of the high-velocity scenario.*

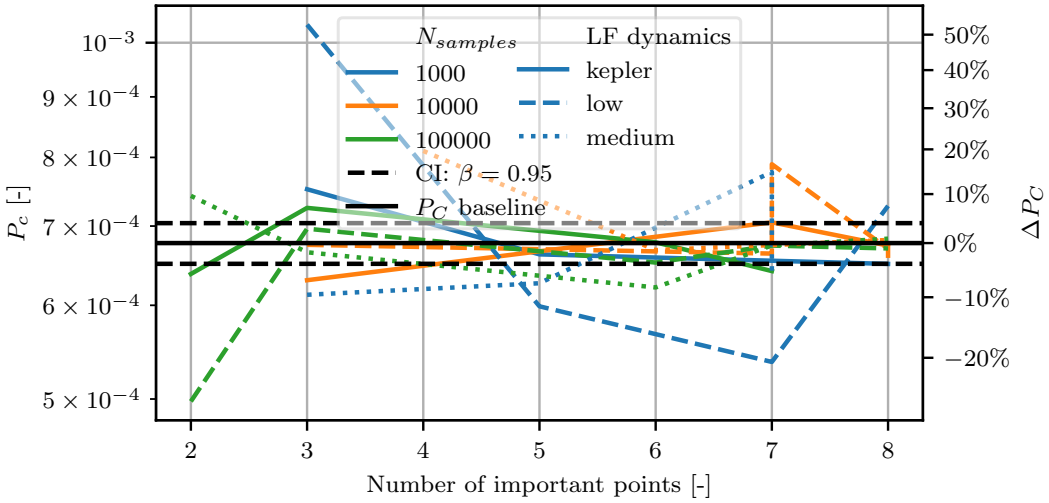


(a) *Short version* $\Delta t \approx 48h$..

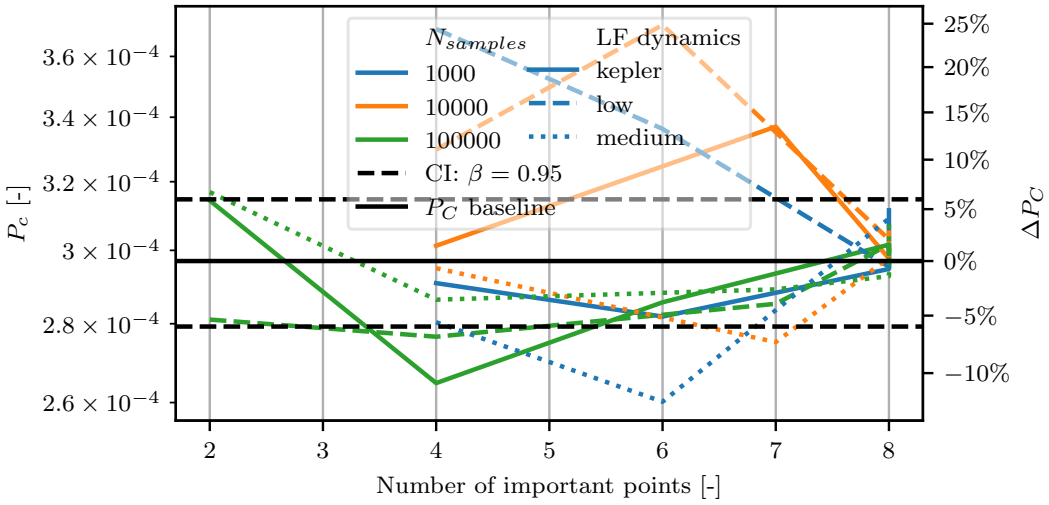


(b) *Long version* $\Delta t \approx 96h$.

Figure F.2: *The probability of collision between the uncertainties propagated using various methods and settings compared to the baseline* P_C *based on the uncertainties propagated with* 10^5 *MC samples, for the short and long version of the low-velocity scenario.*

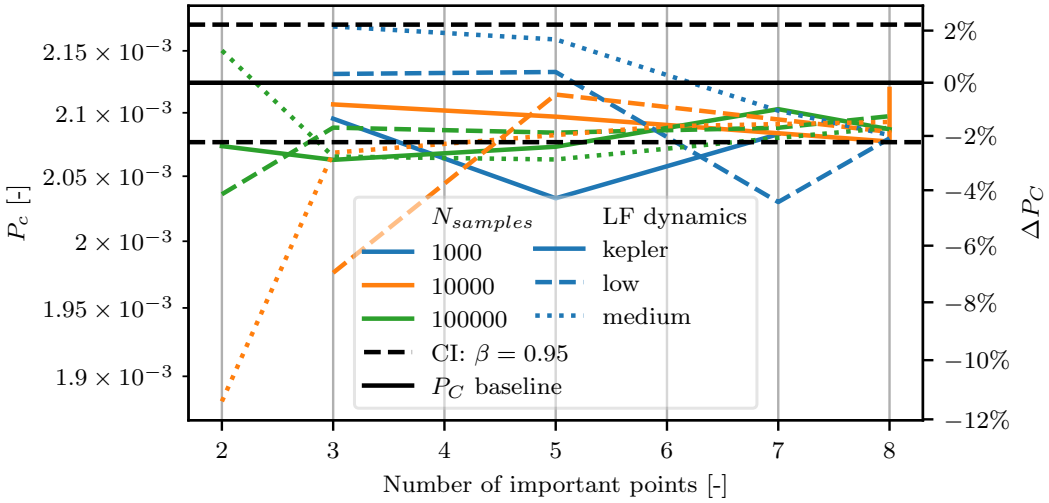


(a) *Short version* $\Delta t \approx 48h$..

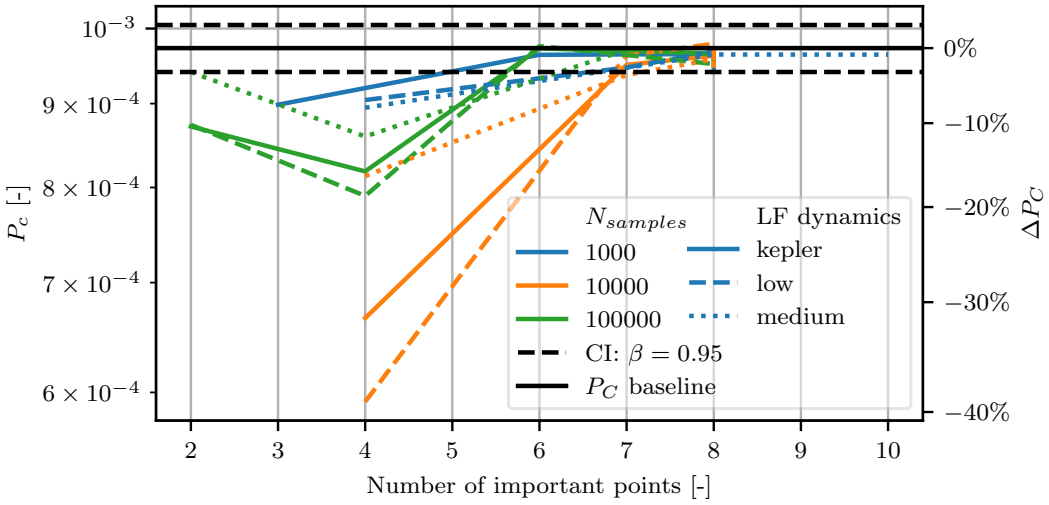


(b) *Long version* $\Delta t \approx 96h$.

Figure F.3: *The probability of collision between the uncertainties propagated using MF plotted against the number of important samples used in the MF propagation and compared to the baseline* P_C *, for the short and long version of the high-velocity scenario.*



(a) *Short version* $\Delta t \approx 48h$..



(b) *Long version* $\Delta t \approx 96h$.

Figure F.4: *The probability of collision between the uncertainties propagated using MF plotted against the number of important samples used in the MF propagation and compared to the baseline* P_C *, for the short and long version of the low-velocity scenario.*

G

Study planning

G.1. Work-Packages

To perform the research described in Chapter 2, a number of work-packages are created which (in the order presented) collectively perform a research that aims to answer the research questions:

- **investigate relevant literature**
  Research into literature surrounding the topic of uncertainty propagation and create a literature study of all the scientific papers on this topic relevant for this research.
- **propose and define research**
  Create a set of research questions and sub-questions to define the research to be performed, subdivide this into work-packages and create a preliminary planning.
- **implement uncertainty propagation methods**
  Implement a number of uncertainty propagation methods into code functions such that they can be used later on. These include MC, the linearised method, UT, MF, GMMs, PCEs and DA.
- **validate methods**
  Validate the implemented uncertainty propagation methods by comparing the results of the methods to those described in literature, by applying the methods to test cases described in these papers.
- **identify test scenarios**
  Choose and motivate a number of test scenarios to use to evaluate the efficiency and performance of the different uncertainty propagation methods. Define all the parameters required to implement these test scenarios.
- **compare uncertainty propagation methods**
  Perform the comparison of the various uncertainty propagation methods using the described test scenarios by running each test scenario with each uncertainty propagation method and documenting the results.
- **analyse results**
  Analyse the documented results for the different methods and test scenarios and make a comparison between the methods based on efficiency and accuracy.
- **choose collision probability methods**
  Choose and motivate methods to be used to calculate the collision probability. Multiple methods can be taken depending on the test case (short-term, long-term, etc.) or the method that it will be combined with (e.g. methods assuming Gaussianity or not).
- **implement first collision probability methods**
  Implement the first methods to calculate the collision probability into code. These first methods include the methods to use MC and to use the Multi-Fidelity.
- **validate methods**
  Validate the implemented collision probability methods by comparing the results of the methods to those described in literature, by applying the methods to test cases described in these papers.
- **identify test scenarios**
  Choose and motivate a number of test scenarios to apply the combined uncertainty propagation and collision probability methods to to determine their efficiency and accuracy.

- **analyse multi-fidelity with collision probability**
  Apply the Multi-Fidelity method and the collision probability method to these test cases and determine the performance of this combination compared to a MC simulation.
- **implement more collision probability methods**
  Implement the other chosen collision probability methods into code. These are the methods required to use the combined MF-GMM method as well as that of MF-GMM with Automatic Domain Splitting.
- **analyse MF-GMM with collision probability**
  Calculate the collision probabilities in the test scenarios using the combined method of Multi-fidelity and Gaussian Mixture Models for the uncertainty propagation, and compare these results to the previous results obtained using only Multi-Fidelity and using MC.
- **analyse MF-GMM with ADS**
  Calculate the collision probabilities in the test scenarios using the combined method of Multi-fidelity and Gaussian Mixture Models equipped with a form of Automatic Domain Splitting for the uncertainty propagation, and compare these results to the previous results obtained.
- **analyse results**
  Analyse the agreements and differences in results between these various methods and determine the levels of efficiency and accuracy of the different methods applied to collision probability.
- **form conclusions**
  Form the proper conclusions on each of the research questions.
- **finalise thesis documentation**
  Finalise all the documentation required for the thesis, such that all results, conclusions, methodologies, and anything else that is important to this research, is documented in the proper formats, and no more writing is needed.
- **prepare thesis defence**
  Prepare for the final thesis defence presentation.

G.2. Provisional Planning

The performance of such a research requires a good planning to ensure a smooth execution. Since a planning can not predict all future events, it can be said with almost certainty that there will be deviations from this planning along the way when new information is presented. However, a provisional planning is still required to provide guidance, clear objectives and deadlines, and a way to manage and achieve expectations.

The work-packages described in Section G.1 provide a good way to divide the timeline of the thesis. Due to the uncertainty involved in planning a research beforehand, two possible scenarios are planned. The first scenario is called the optimistic scenario, which represents the planning in the case that all work-packages are finished according to or ahead of the planning, and includes all of the research objectives. The second scenario is called the realistic scenario, which represents the case that delays arise or when the execution of work-packages takes longer. This scenario does not include extra objectives and only schedules the minimal objectives. Figure G.1 and Figure G.2 show a Gantt Chart for both of these scenarios, in which the work-packages from Section G.1, as well as additional miscellaneous tasks, are planned into a time window with a start and end date. The Gantt charts are made at the level of weeks, to provide a detailed overview and division of tasks over time, while not falling subject to the dangers of micromanaging. An important detail to note is that the Gantt charts do not take into account the holiday weeks.

For more guidance on the structure of the thesis, a division of hours is made and shown in Figure G.3. Similarly to the Gantt chart in Figure G.1 and Figure G.2, this division is made per week to contain the appropriate level of detail. A significant amount of hours are scheduled to be spent on 'coding', which includes the entire setup of the numerical simulations, arguably the biggest part of the thesis. Such a large amount of time also provides margins for the well-known concept of fixing bugs in code. Besides that, writing completes the larger part of the time spent on the thesis, to ensure proper documentation being written along the way, and to avoid time management issues during a sprint before the final deadline. Lastly, around specific deliverables, time is devoted to miscellaneous tasks to account for administration work, review sessions, presentations and other related duties. Especially near the end of the timeline this category becomes dominant, due to the preparation of the thesis defence, where the research itself (related to coding and writing) should already be finished.

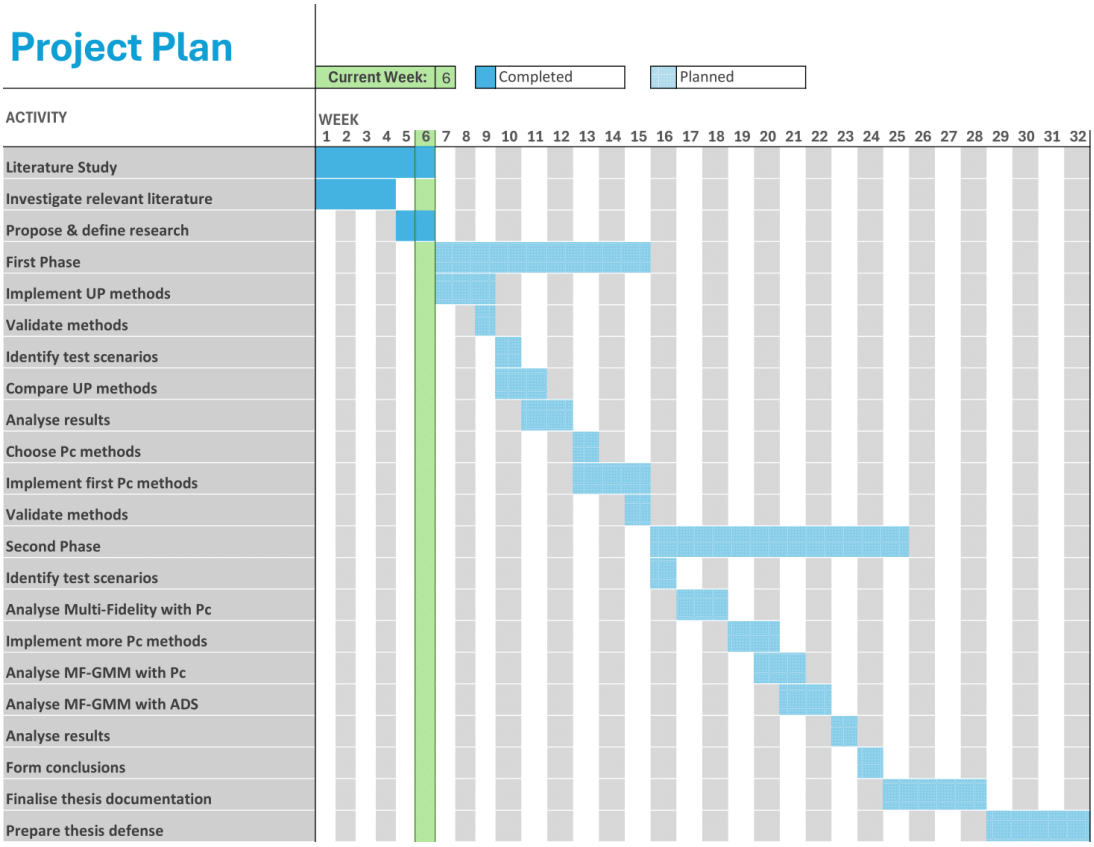


Figure G.1: *Gantt chart with the planning for the optimistic scenario, with week numbers based on the start date of the thesis.*

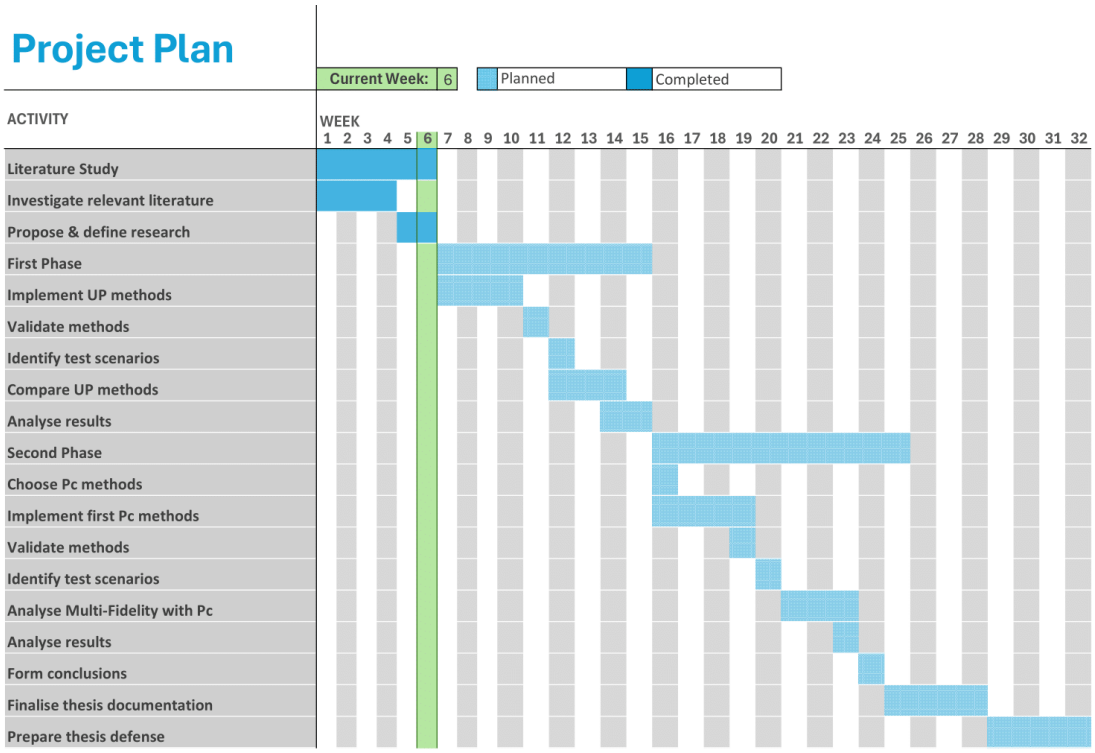


Figure G.2: *Gantt chart with the planning for the realistic scenario, with week numbers based on the start date of the thesis.*

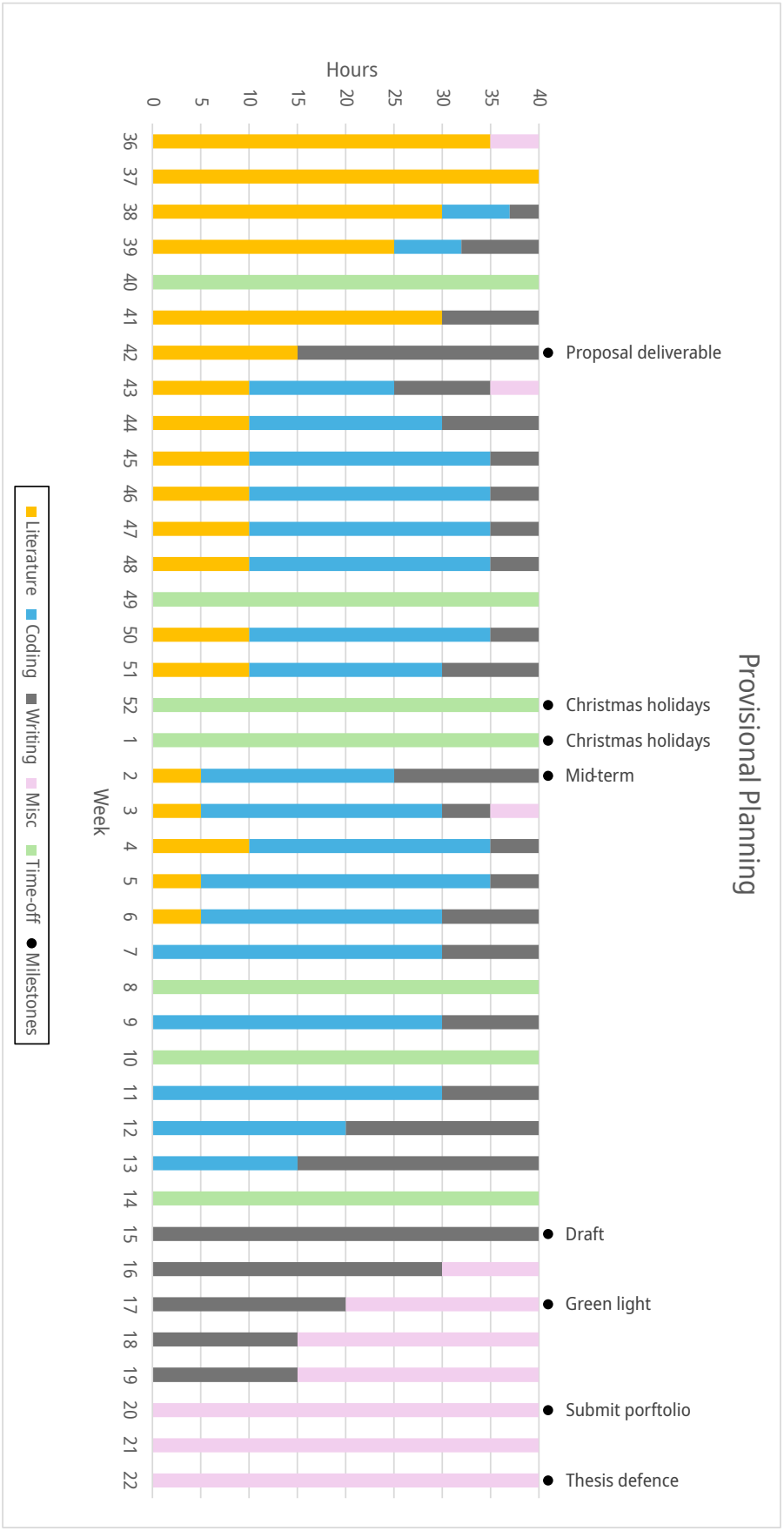


Figure G.3: *Provisional division of the hours spent per week, with holiday weeks taken into account, and week numbering done following the calendar year.*