Regularized Optimal Control Based Estimator

MSc Thesis

Augustas Barysas

Delft University of Technology

TU Delft

This page is intentionally left blank.

Regularized Optimal Control Based Estimator

MSc Thesis

by

Augustas Barysas

Student Number:	5048621
Project duration:	February 7, 2024 – February 24, 2025
Supervisor:	Steve Gehly, PhD
Cover image by:	NASA's Goddard Space Flight Center https://svs.gsfc.nasa.gov/12997/

Delft University of Technology

Acknowledgements

This thesis was the product of almost of a year of work, during which I received support in multiple disciples, from a variety of people. I would like to thank all of those that were involved and assisted me in the completion of this thesis, even in minor ways. I'd also like to highlight some of those whose contributions aided the progress and pushed my understanding the furthest. Firstly, I would like to thank Dominic Dirkx for his support in the field of numerical astrodynamics, particularly throughout the implementation of a custom propagator. I would also like to thank Dr. Giulio Baù for his support in understanding and troubleshooting the regularization methods discussed in this thesis. The implementation of the optimal control based estimator, particularly the unscented variant would likely not have been possible without the contributions of Carlo Pelt, who assisted and shared variants of his implementation during the project. But most of all, I would like to express my deepest gratitude to Dr. Steve Gehly for his supervision throughout the duration of the project. Without the plentiful feedback and numerous suggestions in the face of difficulties, this thesis would not have been possible.

Contents

Glossary

Abbreviations

A-OCBE	Adaptive Optimal Control Based Estimator.
BL-OCBE	Ballistic Linear Optimal Control Based Estimator.
DMC	Dynamic Model Compensation.
GEO	Geostationary Earth Orbit.
GL-OCBE	Generalized Linear Optimal Control Based Estimator.
KS	Kustaanheimo-Stiefel.
MEO	Medium Earth Orbit.
OCBE	Optimal Control Based Estimator.
RK	Runge-Kutta.
R-OCBE	Regularized Optimal Control Based Estimator.
SNC	State Noise Compensation.
SRP	Solar Radiation Pressure.
SSA	Space Situational Awareness.
STM	State Transition Matrix.
Tudat	TU Delft Astrodynamics Toolbox.
U-OCBE	Unscented Optimal Control Based Estimator.

OCBE Symbols

t_k	Time of measurement with index k.
$\vec{x}$	State vector.
$\hat{x}_{i\|j}$	Estimate of state at time t_i, using information at time t_j.
$\delta\hat{x}_{i\|j}$	Estimate of state deviation from reference (propagated) trajectory at time t_i, using information at time t_j.
$\bar{x}_{i\|j}$	Predicted (propagated) state at time t_i, using information at time t_j.
$\hat{P}_{i\|j}$	Estimate of covariance at time t_i, using information at time t_j.
$\bar{P}_{i\|j}$	Predicted (propagated) covariance at time t_i, using information at time t_j.
R_k	Covariance of measurement with index k.
$\vec{p}$	State adjoint vector.
$\hat{p}_{i\|j}$	Estimate of state adjoint at time t_i, using information at time t_j.
$\delta\hat{p}_{i\|j}$	Estimate of state adjoint deviation from reference (propagated) trajectory at time t_i, using information at time t_j.
$\vec{f}_n$	State derivative vector due to natural (non-control) dynamics.
Φ	State transition matrix.
B	Control input scaling matrix.
$\tilde{Q}$	Dynamic Uncertainty Matrix.
σ_Q	Dynamic Uncertainty.
$\vec{Y}_k$	System measurement vector with index k.
h	System measurement function (mapping state to measurement vector).
$\tilde{H}_k$	Linearized, matrix representation of the system measurement function, evaluated at time corresponding to measurement with index k.
u	Control input vector.
$\delta\hat{u}_{i\|j}$	Estimate of control input deviation from reference (propagated) trajectory at time t_i, using information at time t_j.
D_z	Mis-modelled dynamics detection control distance metric.
μ_z	Mean of mis-modelled dynamics detection control distance metric.
$\bar{\chi}_{i\|j}$	Sigma points corresponding to state estimate at time t_i, using information at time t_j.

$\bar{\gamma}_{i\|j}$	Measurements of sigma points corresponding to state estimate at time t_i, using information at time t_j.
$x_{i\|j}^{(k)}$	k-th sigma point corresponding to state estimate at time t_i, using information at time t_j.
W_k	Weight corresponding to k-th sigma point.
P_{Chol}	Cholesky decomposition of covariance.
P_{xy}	Cross covariance.
P_{yy}	Innovation covariance.

EDromo Regularization Symbols

t	Dimensionless Time.
$\vec{\lambda}$	Regularized state vector.
φ	Angle-like independent variable.
$\vec{h}$	Angular momentum vector.
$\vec{k}$	Normalized angular momentum vector.
$\vec{r}$	Position vector.
$\vec{i}$	Normalized Position vector.
U	Dimensionless (central body) perturbing potential.
ε	Dimensionless total energy.
θ	Generalized true anomaly.
ν	Orbital longitude.
$\vec{g}$	Generalized eccentricity vector.
G	Generalized eccentric anomaly.
F	Total (dimensionless) perturbing acceleration.
P	Total non-potential (dimensionless) perturbing acceleration.
R	Total (dimensionless) perturbing radial acceleration.
R_p	Total non-potential (dimensionless) perturbing radial acceleration.
N	Total (dimensionless) perturbing cross-track acceleration.
T_p	Total non-potential (dimensionless) perturbing along-track acceleration.
DU	Distance factor for non-dimensionalization.
TU	Frequency factor for non-dimensionalization.
R_0	Initial orbital radius magnitude.
μ	Central body gravitational parameter.

Other Symbols

U	Spherical Harmonic Gravity Potential.
ϕ	Body-fixed latitude.
θ	Body-fixed longitude.
R	Central body reference radius.
$\bar{P}_{lm}$	Normalized Legendre polynomial of degree l, order m.
$\bar{C}_{lm}$	Normalized cosine harmonic coefficient of degree l, order m.
$\bar{S}_{lm}$	Normalized sine harmonic coefficient of degree l, order m.
$\vec{a}$	Acceleration vector.
m	Mass.
Φ	Solar Flux.
A	(Reference) surface area.
c	Speed of Light.
C_r	Reflectivity coefficient.

α	Surface absorption coefficient.
δ	Diffuse reflectivity coefficient.
ρ	Specular reflectivity coefficient.
$\hat{e}_{\odot}$	Unit vector in direction of solar radiation source.
$\hat{e}_n$	Surface normal unit vector.
$\vec{u}$	Control input acceleration.
P_x	Covariance matrix corresponding to Cartesian state representation.
P_λ	Covariance matrix corresponding to EDromo state representation.
P_{rem}	Remediated covariance matrix.
V	Covariance eigenvector matrix.
Λ	Covariance eigenvalue matrix (diagonal).
Γ	Clipped covariance eigenvalue matrix (diagonal).
dt	Integration time step.
ε_{abs}	Absolute Tolerance.
ε_{rel}	Relative Tolerance.
a	Measurement error scaling factor.
σ_R	Radial measurement standard deviation.
σ_{Az}	Azimuth measurement standard deviation.
σ_{El}	Elevation measurement standard deviation.

1 Introduction

As of 2023 there were over 40,000 trackable objects in orbit around Earth, in form of both spacecraft and space debris[1]. As of late, many projects, including satellite constellations such as Starlink are contributing to a further increase in tracked objects around our home planet. With the International Space Station approaching the end of its lifespan[2], commercial replacements are also expected. It does not stop at low orbits, however. Missions the likes of Artemis from NASA are reaching for the Moon and beyond. Investigations into cis-lunar orbits for staging systems such as the Gateway concept have found NRO (Near Rectilinear Orbits) to be a particularly attractive family[3],[4].

Tracking as many of these objects as possible is crucial to successful operations and safety of both astronauts and active spacecraft and is part of SSA (Space Situational Awareness) initiatives led by the likes of NASA and ESA. Systems such as altimetry satellites make use of precisely estimated spacecraft states to yield meaningful scientific data. The importance of tracking is not limited to cooperative spacecraft, however. Autonomous spacecraft perform stationkeeping maneuvers that are likely unknown to ground systems prior to execution [5]. Malfunctions may result in unexpected changes in the behaviour of a spacecraft. Debris or out of service spacecraft create risk of collision as well. Knowledge of any such object is crucial to conjunction assessment and any following collision avoidance [5].

The variety of object sizes and trajectories results in varying tracking difficulties. In addition to complications due to severe distances, MEO (Medium Earth Orbit) and GEO (Geostationary Earth Orbit) regions often contain large spacecraft or debris with high area-to-mass ratios. Due to the significant effect of solar radiation pressure on these objects, their orbits periodically vary in both inclination and eccentricity [6]. Given that solar radiation pressure is generally difficult to model accurately, this introduces significant dynamics mis-modelling in the tracking algorithms, hindering their performance. These algorithms must output state estimates sufficiently accurate to ensure ground station sensors may locate their target given a time gap between measurements, hence techniques to account for the effect of mis-modelling in such orbits are crucial.

While these considerations are relevant for the already populated geostationary and graveyard orbits, another region of interest to SSA is the family of Tundra orbits. The reduced stability [7] and notable eccentricity of a Tundra orbit, compared to graveyard orbits, allows for re-entry of spacecraft using natural mechanisms within 25–100 years [8]. Should these properties, beneficial to disposal of spacecraft be exploited, the need for tracking and collision avoidance in the Tundra region will increase, despite the currently low traffic.

With more and more objects to be observed in various orbits and crowded observation infrastructure, the importance of computationally efficient state estimation algorithms, capable of working with sparse observations will continue to grow. Given long gaps between measurements, effects of mis-modelled dynamics (such as those due to solar radiation pressure) are significant and methods to compensate for these effects in an estimator are necessary. The first examples of such methods are the SNC (State Noise Compensation) and DMC (Dynamic Model Compensation) which are commonly applied in sequential estimators. SNC is a method that introduces a stochastic process noise term in the dynamics to prevent filter saturation and divergence [9]. A similar method is the DMC, which introduces a deterministic component to the process noise. This component of noise may be estimated to produce a prediction of mis-modelled accelerations which can in turn be used for maneuver detection. A notable drawback of these methods is the need for tuning of the introduced process noise. While the DMC method is less sensitive, it may still provide limitations when applying these methods in an automated manner, especially in cases where truth data is not available [9].

When it comes to the more complex estimation techniques, among the most widely applied are the Multiple Model Estimators [10]. The principle revolves around applying multiple filters with different underlying models. These are used to estimate states concurrently and to determine which model is best suited for the available measurements. While this makes the method well suited to low observability problems, the tendency to converge to a model may limit the performance should the characteristics of the problem change. To capture more varied conditions, the Interacting Multiple Model Estimator may be employed. This method extends the functionality of Multiple Model Estimator to ensure the ability to transition between models is retained. This may result in slightly worse performance in cases of fixed characteristics, though the method remains among the best compromises of complexity and performance [11] and lends itself to varying conditions in the presence of maneuvers [12].

Another such family of algorithms is the OCBE (Optimal Control Based Estimator), first introduced by Daniel P. Lubey[13] based on the work of Marcus J. Holzinger and Daniel J. Scheeres [14]. The OCBE allows for estimation of mis-modeled dynamics and maneuver detection using a set of observations and an assumed dynamics model. Unlike the previously discussed methods, the OCBE employs a user-defined cost function which is minimized using optimal control theory to obtain state estimates at both the a priori and measurement epochs. The original work in [13] introduces multiple variations of the OCBE, the simplest being the BL-OCBE (Ballistic Linear Optimal Control Based Estimator), a more computationally intensive Generalized Linear OCBE which can be used iteratively to solve non-linear problems as well as an automated adaptive OCBE. Similar to methods employing DMC, the OCBE outputs control estimates that represent the severity of the mis-modelled dynamics in addition to the estimated states and corresponding covariances. This is particularly convenient for maneuver detection, reconstruction and even environment model updates.

Other authors have since further developed the OCBE, including methods to reduce numerical instability in cis-lunar problems [15], or adjustments to the control distance metrics to reduce computational complexity and algorithm consistency for anomaly detection [16]. Greaves and Scheeres [17], [18] explored autonomous navigation of spacecraft using relative measurements and the OCBE, having updated it using the Unscented Transformation to account for strongly non-linear dynamics. However, regardless of the improvements made, the benefits of the estimator come at the cost of computational efficiency which suffers compared to other methods. This is of particular interest, especially when considering applications such as onboard autonomous navigation systems.

Computationally efficient algorithms for numerical integration are already of interest in the field of orbit propagation. A common approach that has been explored in the field of astrodynamics since the 1970s involves averaging perturbing terms [19], [20], [21]. The general approach to averaging starts with variation of parameters applied to a known solution of an unperturbed problem. Upon substitution, this is followed by a Fourier expansion of the full problem. Given that the perturbation is small, the chosen parameters vary slowly in time. It is thus stated that all terms of the Fourier expansion bar the average have a net-zero contribution to the solution over a period 2π. This yields a simplified set of equations 'averaged' over a period that can be solved to determine an approximation of the perturbed problem [22]. The main advantage of these methods is the computational performance over large time scales, with bounded error. However, due to the closed-form nature of the methods, the applicable environment models and accuracy are limited. In more recent studies, the semi-analytical averaging methods have been expanded to incorporate a larger variety of environments such as non-conservative drag forces[23]. Averaging methods have even been applied to optimal control problems, though in the context of optimal transfers[24] rather than the OCBE.

The Poincaré-Lindstedt method is another well known semi-analytical procedure to construct asymptotic approximations of solutions for non-linear systems. Sometimes also referred to as the continuation method[22], it requires the solution to be periodic and a forcing term to be in order ε. The method revolves around use of a time-like variable that scales the problem to be 2π periodic. Then, a series expansion is used to construct the approximation, the coefficients of which are determined by separating terms of the transformed system based on the powers of ε. The applications of the method are limited due to the need for a periodic solution and the limited timescale over which it remains accurate[22].

A more general version of the Poincaré-Lindstedt method is referred to as the method of multiple (time) scales. Unlike the Poincaré-Lindstedt method, this allows the construction of aperiodic solutions. The result is achieved by introducing a number of additional time scales as independent variables, related by factors ε^n and constructing approximations based on the resulting partial differential equation [25]. Unfortunately, this method may be cumbersome and depending on application, it is not guaranteed to yield a closed form solution. Furthermore, when using a large number of time scales, consistency constraints on the structure of the solution are introduced. This is easily violated should the free terms be chosen arbitrarily, and may result in failure of the analysis [26].

An alternative to traditional semi-analytical methods is regularization of the equations of motion. A number of regularization methods have been developed to improve performance of numerical integrators[27]. The equations of motion of the perturbed two-body problem have two issues. First of all, they contain a singularity as the orbital radius approaches zero. Secondly, they are unstable in the sense of Lyapunov. This results growth of truncation error during numerical integration processes. These issues are also present in the N-body problem. Thus, the primary objectives of regularization involve removing singularities and reducing the effect of the instability of the equations. With the process complete, the resulting system is also reduced to a linear form, at least for unperturbed trajectories. This may also provide benefits when applying a linear estimation method, such as the BL-OCBE.

Generally, to tackle the aforementioned problems, regularization methods involve two distinct aspects. The first, involves a change of independent variable to a fictitious time by means of a Sundman transform. This reduces the order of the singularity and introduces a form of analytical control of the integrator timestep [27]. In order to complete the regularization, a second step to further stabilize the equations of motion is necessary. This is done in one of two ways. An option is to embed the energy equation into the system and to linearize the equations to the form of a linear oscillator. A common alternative revolves around integrating sets of elements rather than the coordinates themselves.

A number of regularization methods with different Sundman transforms and formulations of the equations of motion are available in literature, many of which have been thoroughly tested for numerical performance [28] and compared to averaging methods [29]. It is generally found that regularization is more computationally intensive than averaging methods, however it is not naturally limited in its accuracy or environment models.

An example of a well-performing regularization for the two body problem are variants of the Dromo formulation [27], [28]. This formulation relies on propagation of elements including projections of the eccentricity vector on an intermediate frame, orientation of said frame, the total energy, as well as an element used to reconstruct the physical time.

While the previously discussed findings stand for the perturbed two-body problem, for cis-lunar missions, the dynamics can be modelled as the CR3B (circular restricted three body) problem. In this case, many of the previously discussed methods fail, especially so when considering averaging. As an alternative, [30] has shown that closed form solutions can still be obtained for the restricted three body problem, though in this case the problem is also confined to a single plane.

When it comes to regularization, variants of the KS (Kustaanheimo-Stiefel) formulation [27] can act as suitable alternatives, as also shown in [29]. Unlike the Dromo formulation which integrates elements, KS defines the state of the particle as a four-dimensional position, and its derivative with respect to the fictitious time. In order to reconstruct the Cartesian state, similar to Dromo, the total energy as well as an element to reconstruct the physical time are necessary. This results in a 10-dimensional system of differential equations representing the equations of motion. It should be noted that as a result of the dimensionality, the KS formulation requires a choice of a free parameter. While in general it can be fixed arbitrarily, strong perturbations may destabilize the system and lead to different solutions for different choices, hence it should be treated with sufficient caution [27].

State estimation in terms of a regularized representation of the state has also already been tackled to a degree. J. H. Ayuso [31] [32] implemented a least-squares batch estimator and a conventional Kalman filter in terms of 'Dromo' [33] regularized elements. This approach allowed for taking advantage of the linear form of the dynamics equations with linear filters, at the cost of additional non-linear transformations to, and from the estimated elements. For the KS regularization authors have also developed variational equations [34] and analytical covariance transformations [35] to and from Cartesian coordinates. Both of which are necessary pieces for an estimator implementation. However, none of these developments have been applied in the context of the OCBE. Given the potential benefits of the regularization methods and the generally computationally intensive OCBE it seems desirable to pursue performance gains by integrating the two methods. The research objective of this Master's thesis is thus defined to: investigate performance of the OCBE with incorporated regularized dynamics.

Considering the weakness of the traditional OCBE lies in the computational effort, any changes to computational efficiency and floating point operations are of great interest. Naturally, knowledge of performance over a range of applications rather than a particular case is preferred. As previously discussed, growth of spacecraft in various non-low Earth orbits, such as the Tundra orbit, is expected. These cases are sensitive to disturbing potential, where the regularization is expected to perform well. Evaluation of the flexibility of the modified OCBE for such cases may provide valuable insight in both direct application and prospect of gains in conditions less favourable for the algorithm. As a result, performance dependencies on severity of dynamics mis-modelling are of interest. Finally, given that treated orbits are relatively difficult to observe, effects of variation in the observation frequency and uncertainty are also considered crucial to the problem.

The research questions corresponding to the objective are thus summarised as follows:

- How does dynamics regularization affect the computational efficiency of the OCBE?
- How does the performance of the regularized OCBE depend on the severity of dynamics mismodelling?
- How does the performance of the regularized OCBE depend on measurement frequency and uncertainty?

This report contains the findings of the thesis and is structured as follows. Chapter 2 elaborates on the background of the methods used in the thesis directly. This includes the equations of the BL-OCBE and its variants, selection and discussion of regularization method, and key aspects of the dynamics models. Then, Chapter 3 provides an overview of the methodology used to answer the research questions, as well as the architecture of the algorithms implemented in pursuit of numerical results. This is followed by Chapter 4, which discusses the main developments of this thesis in, form of the changes made to the OCBE in order to facilitate estimation of regularized elements. Chapters 5 and 6 start the discussion of numerical results of the thesis, covering the selected test cases and corresponding implications on numerical integration accuracy respectively. Finally, the performance of all implemented algorithms is discussed in Chapter 7, and Chapter 8 provides the conclusions as well as summarises the recommendations for future work to build on the results of this thesis.

2
Background Methods

While contextual information and some literature findings are already presented in the introduction, there are a number of works that provide the basis for this Master's thesis. This chapter focuses on the literature findings that are directly implemented or otherwise used throughout the thesis. Section 2.1 summarizes the equations of the implemented OCBE algorithm. Then, Section 2.2 discusses the choice of the semi-analytical method to be integrated with the OCBE. Finally, a few important aspects of dynamics modelling are presented in Section 2.3.

2.1. OCBE

The first key algorithm to be discussed is the Optimal Control Based Estimator [13], [36]. The Optimal Control Based Estimator in general, is an algorithm performing state estimation by means of optimal control on an arbitrary system of the form shown in Equation 2.1, in which $\vec{f}_n$ is the state derivative purely due to natural dynamics, $\vec{x}$ is the state vector and t is the time. $\vec{u}$ is a fictitious control input (to be found by the estimator), while B is a control input scaling matrix, mapping the control input to a contribution to the state derivative.

$$\dot{\vec{x}} = \vec{f}\left(\vec{x}(t), t, \vec{u}(t)\right) = \vec{f}_n\left(\vec{x}(t), t\right) + B(t)\vec{u}(t) \tag{2.1}$$

In order to obtain a solution by optimal control theory, a Bolza-type cost function may be defined which takes the form in Equation 2.2 [13], [36].

$$\begin{aligned} J = & K_{k-1}\left(\vec{x}_{k-1}, \hat{x}_{k-1|k-1}, \hat{P}_{k-1|k-1}\right) \\ & + K_k\left(\vec{x}_k, \vec{Y}_k, R_k, t_k\right) \\ & + \int_{t_{k-1}}^{t_k} L\left(\vec{u}(\tau), \hat{u}(\tau), \tilde{Q}(\tau)\right) d\tau \end{aligned} \tag{2.2}$$

The first and second terms, K_{k-1} (Equation 2.3) and K_k (Equation 2.4) correspond to a least squares boundary costs for the a priori state and measurement information respectively [13]. These terms encourage the selection of state estimates close to the corresponding available information, given an uncertainty. The final term, (in which L is given by Equation 2.5) represents the mis-modelling of the system by means of a fictitious control input, which is used to connect the state estimates at the a priori and measurement epochs.

$$K_{k-1} = \frac{1}{2}\left(\hat{x}_{k-1|k-1} - \vec{x}_{k-1}\right)^T \hat{P}_{k-1|k-1}^{-1}\left(\hat{x}_{k-1|k-1} - \vec{x}_{k-1}\right) \tag{2.3}$$

$$K_k = \frac{1}{2}\left(\vec{Y}_k - h(t_k, \vec{x}_k)\right)^T R_k^{-1}\left(\vec{Y}_k - h(t_k, \vec{x}_k)\right) \tag{2.4}$$

$$L = \frac{1}{2}\left(\vec{u}(t) - \bar{u}(t)\right)^T \tilde{Q}^{-1}\left(\vec{u}(t) - \bar{u}(t)\right) \tag{2.5}$$

In these equations, $\vec{x}_{k-1}$ and $\vec{x}_k$ are the state vectors that the estimator aims to select by introducing an appropriate control policy $\vec{u}(t)$. $\hat{x}_{k-1|k-1}$ represents the state estimate at (a priori) time t_{k-1}, which was obtained using information available at time t_{k-1} while $\hat{P}_{k-1|k-1}$ is the corresponding covariance. $\vec{Y}_k$ is the measurement obtained at time t_k with covariance R_k and h is a function mapping a state to a corresponding measurement vector. The vector $\bar{u}(t)$ is the assumed control vector (generally set to zero given that dynamics are well-modelled) with the matrix $\tilde{Q}$ representing the dynamic uncertainty.

The priori state estimate and measurement covariances $\hat{P}_{k-1|k-1}$ and R_k as well as the dynamics uncertainty $\tilde{Q}$ in the cost function allow for weighing the importance of the deviations from initial state, measurement and expected control effort. The solution of the optimal control problem for the aforementioned cost function, thus leads to state estimates at both the a priori and measurement epochs as well as a fictitious control vector.

While the process of deriving a solution to the optimal control problem is not covered in this thesis, it is important to note that in order to obtain a control input minimizing the cost function, a state adjoint $\vec{p}$ (which is later related to the control input itself) defined in Equation 2.6 is necessary.

$$\dot{\vec{p}} = -\left(\frac{\partial L}{\partial \vec{x}}\right)^T - \left(\frac{\partial f}{\partial \vec{x}}\right)^T \vec{p}(t) \tag{2.6}$$

Lubey and Scheeres [13], [36], [37] used the results of the optimal control problem to develop multiple variants of the OCBE estimator. One of which is the BL-OCBE, which was implemented as part of this thesis based on the aforementioned literature. The BL-OCBE is a variant of the OCBE algorithm, which hinges on the assumption of linearized dynamics around a ballistic nominal trajectory. This leads to a significantly simplified implementation and a more computationally efficient process to obtain results. The equations relevant for it's implementation are thus summarized in this section.

2.1.1. BL-OCBE Estimation

The equations of the BL-OCBE are applied for sequential estimation, which due to the nature of the cost function results in solving a two-point boundary value problem for each measurement epoch. Unlike a Kalman filter, this results in a state estimate for both the prior and current states when processing a new measurement (effectively acting as a one step smoother). This section briefly elaborates on the estimation process and provides the notation used in the equations below.

Before moving on to the method in detail, a few comments on the notation should be made. First of all, a number of parameters with subscript notation of the form $k \mid k-1$ are present. This implies 'value corresponding to time t_k, using information from time t_{k-1}' (or equivalent given different indices). Furthermore, a number of symbols include a hat (i.e. $\hat{x}_{k|k}$) or bar (i.e. $\bar{x}_{k|k-1}$). The first, implies the variable is an estimate, which is an output of the BL-OCBE. The bar on the other hand, implies a predicted (propagated) value rather than an estimate. Finally, a number of steps in the estimation are performed in terms of linearized deviations evaluated with respect to the reference (propagated) trajectory [36]. Such deviations are indicated with δ, for example the OCBE may output an estimated state deviation $\delta\hat{x}_{k|k}$, which must be added to the corresponding reference trajectory state in order to obtain the estimate $\hat{x}_{k|k}$ itself.

To provide a top level overview of the approach, Figure 2.1 was created. Here the method is compared to a Kalman filter, in terms of the actions taken as part of the time and measurement update steps. While all steps performed in the Kalman filter are also present in the BL-OCBE (though with different mathematical formulations), the coloured blocks indicate steps part of the OCBE which are not present in the Kalman filter.

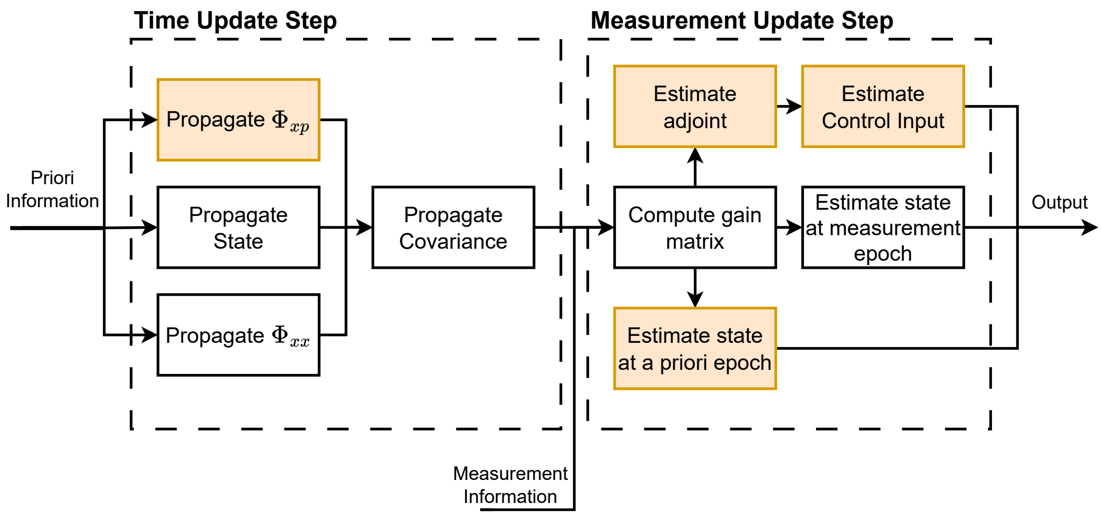


Figure 2.1: *Top-level overview of steps in a Kalman Filter and BL-OCBE. Coloured blocks indicate a step that is present in the BL-OCBE but absent in the Kalman filter. Remaining blocks indicate steps present in both algorithms.*

.

Similar to a Kalman filter, the approach as presented in [13] begins with a time update step. The state estimate $\hat{x}_{k-1|k-1}$ at time t_{k-1} is propagated to, $\bar{x}_{k|k-1}$. Then the propagated covariance is given by Equation 2.7.

$$\bar{P}_{k|k-1} = \bar{P}_k - \Phi_{xp}\Phi_{xx}^T \quad (2.7) \qquad \bar{P}_k = \Phi_{xx}\hat{P}_{k-1|k-1}\Phi_{xx}^T \quad (2.8)$$

Notice how the state transition matrix Φ is referenced with subscripts with xx and xp. This is due to the first major difference between the BL-OCBE and the Kalman filter. The OCBE solves an optimal control problem, and does so by obtaining the aforementioned adjoint of the state p. This adjoint may be propagated along the state with Equation 2.9, with $\vec{a}_n$ as the natural dynamics acceleration (last 3 rows of $\vec{f}_n$), $\vec{r}$ and $\vec{v}$ as the position and velocity respectively, while $\vec{p}_r$ and $\vec{p}_v$ are the vectors of the first and last 3 elements of the adjoint (in the case of Cartesian position/velocity state).

$$\dot{\vec{p}} = \begin{bmatrix} -\left(\frac{\partial \vec{a}_n}{\partial \vec{r}}\right)^T \vec{p}_v \\ -\vec{p}_r - \left(\frac{\partial \vec{a}_n}{\partial \vec{v}}\right)^T \vec{p}_v \end{bmatrix} \quad (2.9)$$

While propagation of the adjoint is not strictly necessary to perform state estimation, as was seen from Equation 2.7, the covariance propagation does require components of the state transition matrix corresponding to the adjoint (particularly Φ_{xp}). This, extended STM (State Transition Matrix) is given by Equations 2.10 - 2.12:

$$\dot{\Phi}(t,\tau) = A(t)\Phi(t,\tau) \quad (2.10) \qquad \Phi(t_k, t_{k-1}) = \begin{bmatrix} \Phi_{xx} & \Phi_{xp} \\ \Phi_{px} & \Phi_{pp} \end{bmatrix} \quad (2.11)$$

$$A(t) = \begin{bmatrix} \frac{\partial \dot{\vec{x}}}{\partial \vec{x}} & -B(t)\tilde{Q}(t)B(t)^T \\ -\frac{\partial^2}{\partial \vec{x}^2}(\dot{\vec{x}}^T \vec{p}) & -\frac{\partial \dot{\vec{x}}}{\partial \vec{x}}^T \end{bmatrix} \quad (2.12)$$

In which we once again observe the control gain matrix B and the assumed dynamic uncertainty matrix $\tilde{Q}$ (acting as a user input) from the original cost function. Fortunately in the simplified case of the BL-OCBE, Φ_{px} is reduced to the zero matrix and Φ_{pp} can be computed using Equation 2.13.

$$\Phi_{pp} = (\Phi_{xx}^{-1})^T \tag{2.13}$$

This reduces the necessary function evaluations and leaves only the Φ_{xx} and Φ_{xp} quadrants to be integrated numerically with Equations 2.14 and 2.15

$$\dot{\Phi}_{xx} = \frac{\partial \dot{\vec{x}}}{\partial \vec{x}} \Phi_{xx} \tag{2.14}$$

$$\dot{\Phi}_{xp} = \frac{\partial \dot{\vec{x}}}{\partial \vec{x}} \Phi_{xp} - B(t)\tilde{Q}(t)B(t)^T \Phi_{pp} \tag{2.15}$$

Once the state, covariance and the extended STM are propagated in the time update step, the measurement update may be performed which is also shown in Figure 2.1. The state estimate at the measurement epoch with the corresponding covariance is found using Equations 2.16 and 2.17 for the BL-OCBE. Here L_k is the BL-OCBE equivalent of a Kalman gain, given by Equation 2.18. With any auxiliary values necessary provided by Equations 2.19 to 2.21. Here, $\vec{Y}_k$ is the system measurement at time t_k, R_k is the corresponding covariance and h is the function mapping a state to a measurement.

$$\delta \hat{x}_{k|k} = \delta \bar{x}_{k|k-1} + L_k(\delta \vec{y}_k - \tilde{H}_k \delta \bar{x}_{k|k-1}) \tag{2.16}$$

$$\hat{P}_{k|k} = (I - L_k \tilde{H}_k)\bar{P}_{k|k-1}(I - L_k \tilde{H}_k)^T + L_k R_k L_k^T \tag{2.17}$$

$$L_k = \bar{P}_{k|k-1}\Psi_k \tag{2.18}$$

$$\Psi_k = \tilde{H}_k^T (R_k + \tilde{H}_k \bar{P}_{k|k-1} \tilde{H}_k^T) \tag{2.19}$$

$$\delta \vec{y}_k = \vec{Y}_k - h(t_k, \bar{x}(t_k)) \tag{2.20}$$

$$\tilde{H}_k = \frac{\partial h}{\partial \vec{x}}\Big|_{t_k} \tag{2.21}$$

As unlike the Kalman filter, the OCBE solves a two-point boundary value problem it may also be used to obtain an updated estimate of the state and covariance at the priori epoch:

$$\delta \hat{x}_{k-1|k} = \delta \hat{x}_{k-1|k-1} + \Phi_{pp}^T \bar{P}_k \Psi_k (\delta \vec{y}_k - \tilde{H}_k \delta \bar{x}_{k|k-1}) \tag{2.22}$$

$$\hat{P}_{k-1|k} = \hat{P}_{k-1|k-1} - \hat{P}_{k-1|k-1}\Phi_{xx}^T \Psi_k \tilde{H}_k \Phi_{xx} \hat{P}_{k-1|k-1} \tag{2.23}$$

The final benefit of the OCBE is that the solution of the optimal control problem can be used to reconstruct maneuvers or mis-modelled dynamics. Parallel to the state estimates, the system's optimal control policy can be obtained in terms of the adjoint estimates in Equations 2.24 and 2.25. While this process is not strictly necessary to perform the state estimation, the adjoint estimates may be applied to obtain the estimate of the fictitious control input itself (given by Equation 2.26), which can in turn be used in as part of distance metrics for maneuver detection and reconstruction as well as estimation of natural dynamics mis-modelling [36].

$$\delta \hat{p}_{k-1|k} = -\Phi_{xx}^T \Psi_k (\delta \vec{y}_k - \tilde{H}_k \delta \bar{x}_{k|k-1}) \tag{2.24}$$

$$\delta \hat{p}_{k|k} = -\Psi_k (\delta \vec{y}_k - \tilde{H}_k \delta \bar{x}_{k|k-1}) \tag{2.25}$$

$$\delta \hat{u}(t) = -\tilde{Q} \frac{\partial \dot{x}}{\partial u}^T \Phi_{pp}(t, t_{k-1}) \delta \hat{p}_{k-1|k} \tag{2.26}$$

As with a Kalman filter, this process may of course be repeated until the full series of measurements are used to obtain state estimates.

2.1.2. Smoothing

Following the evaluation of state estimates, Lubey and Scheeres[13] also provide equations to achieve smoothing for the OCBE state estimates. The goal of the smoothing algorithm is to update all estimates at times t_k with additional information obtained with later measurements. For each previously generated estimate, a new smoothed version is computed using Equations 2.27 and 2.28.

In these equations, l indicates the total number of measurements. Hence, for example, $\delta\hat{x}_{k|l}$ refers to the smoothed estimated state correction for time t_k using information at the end of the estimation, at time t_l. To apply these equations, a recursive algorithm is necessary, starting at the final measurement epoch where $k = l$. In this case the smoothed estimate is equal to the estimate computed using Equation 2.16. This provides all necessary components to compute $\delta\hat{x}_{k-1|l}$, which then can be used to compute $\delta\hat{x}_{k-2|l}$ and so on, until smoothing is achieved for all times.

$$\delta\hat{x}_{k-1|l} = \delta\hat{x}_{k-1|k-1} + S_{k-1}[\delta\hat{x}_{k|l} - (\Phi_{xx}\delta\hat{x}_{k-1|k-1})] \tag{2.27}$$

$$\delta\hat{P}_{k-1|l} = \hat{P}_{k-1|k-1} + S_{k-1}[\hat{P}_{k|l} - \bar{P}_{k|k-1}]S_{k-1}^T \tag{2.28}$$

Using:

$$S_{k-1} = \hat{P}_{k-1|k-1}\Phi_{xx}^T\bar{P}_{k|k-1}^{-1} \tag{2.29}$$

A direct expression of the smoothed control estimate can also be computed using Equations 2.30 and 2.31

$$\delta\hat{p}_{k-1|l} = -\hat{P}_{k-1|k-1}^{-1}S_{k-1}[\delta\hat{x}_{k|l} - (\Phi_{xx}\delta\hat{x}_{k-1|k-1})] \tag{2.30}$$

$$\delta\hat{u}_{t|t_l} = -\tilde{Q}(t)B(t)^T\Phi_{pp}\hat{P}_{k-1|k-1}^{-1}[\delta\hat{x}_{k-1|k-1} - \delta\hat{x}_{k-1|l}] \tag{2.31}$$

2.1.3. Adaptive OCBE

While the equations presented in the previous subsections provide the means to obtain and smooth state estimates, the effectiveness of the method still significantly depends on the user specified assumed dynamic uncertainty matrix $\tilde{Q}$. To alleviate the need for trial and error in choice of this parameter, and a means to update it throughout the estimation process Lubey[13] also introduced the A-OCBE (Adaptive Optimal Control Based Estimator).

A simplified overview of the steps to update the dynamic uncertainty in the A-OCBE is shown in Figure 2.2. First, a dynamic uncertainty matrix that is constant over a measurement gap is defined as shown in Equation 2.32, with an initial σ_Q equal to a low value representing a noise floor, below which dynamic mis-modelling can not be detected. This allows one to perform the state estimation process, based on the BL-OCBE equations discussed earlier in this chapter.

$$\tilde{Q} = (t_k - t_{k-1})\sigma_Q^2 I \tag{2.32}$$

Then a control distance metric D_z with mean μ_z is introduced. In the case of a BL-OCBE, a suitable distance metric was also developed by Lubey[13] and it is given by Equations 2.33 through 2.35.

$$D_z = (d\vec{y})^T M_J(d\vec{y}) \tag{2.33}$$

$$d\vec{y} = \delta\vec{y}_k - \tilde{H}_k\delta\bar{x}_{k|k-1} \tag{2.34}$$

$$M_J = \frac{1}{2}\left(R_k + \tilde{H}\bar{P}_{k|k-1}\tilde{H}^T\right)^{-1} - \left(\frac{\sigma_Q}{\sigma_{Q,NF}}\right)^2\left[\Phi_{xp}(\sigma_{Q,NF})\Phi_{xx}^T\right] \tag{2.35}$$

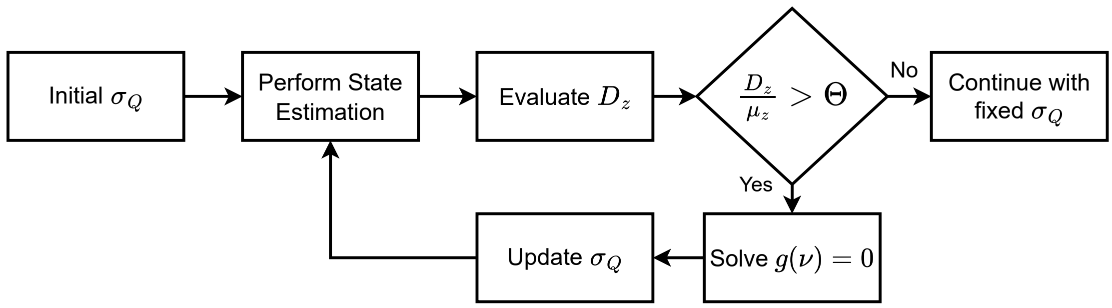


Figure 2.2: *Process used for adaptively modifying the assumed dynamic uncertainty during the state estimation process with the OCBE*

.

If the ratio of the control distance metric D_z and its mean μ_z is lower than a user specified threshold Θ, the dynamic uncertainty does not need to be adjusted and the estimation may proceed. If instead the ratio exceeds the threshold, it implies dynamics mis-modelling, above the expected magnitude has been detected and the dynamic uncertainty should be adjusted.

To do so, the σ_Q should be updated to enforce Equation 2.36. Effectively this implies that if the OCBE detects mis-modelled dynamics, it should adjust the dynamic uncertainty such that the instantaneous value of the distance metric is equal to its mean, and thus mis-modelling is no longer detected.

$$D_z(\sigma_Q) = \mu_z(\sigma_Q) \tag{2.36}$$

In practice, this is solved numerically by applying a root finding method, though it is convenient to first rewrite the problem in the logarithmic space and solve Equation 2.37 in terms of the independent variable υ, defined in Equation 2.38:

$$g(\upsilon) = \log_{10}\left[\frac{D_z(\upsilon)}{\mu_z(\upsilon)}\right] = 0 \tag{2.37}$$

$$\sigma_Q = 10^{\upsilon} \tag{2.38}$$

With a suitable value of υ obtained, it may be used to obtain an updated value of σ_Q from Equation 2.38. The estimation process may then be repeated with a more-appropriate dynamic uncertainty, which ought to lead to no detection of (larger than expected) mis-modelled dynamics.

It should be noted that in order to ensure the method does not significantly adjust the dynamic uncertainty if a measurement outlier results in a false positive dynamics mis-modelling detection, the correction to σ_Q should only be applied after a number of successive measurements resulting in distance metric ratios above the threshold. A more detailed discussion on the architecture of an implemented version of the A-OCBE algorithm, including this detail and more follows in Section 3.2.

2.1.4. Unscented Transformation

While intuitively it may seem that developing a modified OCBE algorithm would be easiest if starting from the most simple version in the BL-OCBE, there are a few aspects that may suggest otherwise. First of all, the BL-OCBE relies on linear propagation of states via the state transition matrix.

While for many applications this is not inherently a complication, there are consequences for a number of methods that should be considered. For example, a number of regularization methods discussed in the introduction boast an increased dimensionality of the state. In the case of the KS method, the state is represented as a 10 dimensional vector. This implies that an STM that also includes the quadrants for the adjoint is size 20×20, rather than 12×12 as it is in the case of state estimation in terms of Cartesian coordinates. This may introduce significant computational load, both when evaluating the matrix as part of the reference propagation and in cases where inverse of the STM is required. Additionally, other methods such as the EDromo method selected for this thesis (which is discussed in more detail in the next section), contain state elements subject to additional constraints such as quaternions. Linear propagation of such elements may violate their constraints and thus introduce error into the estimation process.

This issue in particular has been relevant in the discussion of attitude state estimation since the 1980s and solutions involving alternate definitions of error vectors have been found [38], though the approach may not be suitable to the BL-OCBE without significant modification. A more recent approach, also stemming from attitude estimation resolves this issue by applying the Unscented Transformation[39] on a filter, in which case the state is no longer propagated linearly using the state transition matrix. Instead a number of sigma points are obtained and each propagated through the full dynamics model, resolving a number of the aforementioned issues. As mentioned in the introduction, this approach has also been developed for the OCBE by Greaves and Scheeres[18], using the basis of the Unscented Kalman filter [40].

The differences of the unscented formulation of the OCBE compared to the traditional formulation begin with a new first step in the estimation process, the computation of $2n+1$ sigma points $\bar{\chi}_{k-1|k-1}$ using equations 2.39 - 2.44. Where $x^{(i)}$ is the i-th sigma point, n is the length of the state vector, $\bar{x}_{k-1|k-1}$ is the mean state and $(P_{Chol})_j$ is the j-th column of the Cholesky decomposition of the corresponding covariance $P_{k-1|k-1}$:

$$x^{(i)}_{k-1|k-1} = \bar{x}_{k-1|k-1} + \tilde{x}^{(i)} \tag{2.39}$$

$$x^{(0)}_{k-1|k-1} = \bar{x}_{k-1|k-1} \tag{2.40}$$

$$\tilde{x}^{(2j-1)} = \sqrt{\frac{n}{1-W_0}}\,(P_{Chol})_j \tag{2.41}$$

$$\tilde{x}^{(2j)} = -\sqrt{\frac{n}{1-W_0}}\,(P_{Chol})_j \tag{2.42}$$

$$W_0 = 1 - \frac{n}{3} \tag{2.43}$$

$$j = 1, 2, \ldots, n \tag{2.44}$$

Now, for the time update step instead of propagating the mean state with an STM, each of the sigma points may be propagated through the full dynamics individually to obtain $\bar{\chi}_{k|k-1}$ and a new mean and covariance may be approximated using Equation 2.45 and Equation 2.46 respectively (where $\otimes$ indicates the vector outer product).

$$\bar{x}_{k|k-1} = \sum_{i=0}^{2n} W_i x^{(i)}_{k|k-1} \qquad W_i = \begin{cases} 1 - \frac{n}{3} & i = 0 \\ \frac{1-W_0}{2n} & i = 1, 2 \ldots \end{cases} \tag{2.45}$$

$$P_{k|k-1} = \left(\sum_{i=0}^{2n} W_i \left[x^{(i)}_{k|k-1} - \bar{x}_{k|k-1} \right] \otimes \left[x^{(i)}_{k|k-1} - \bar{x}_{k|k-1} \right] \right) - \Phi_{xp} \Phi^T_{xx} \tag{2.46}$$

It should be noted that unfortunately for an application in the OCBE integration of the state transition matrix of the central sigma point is still necessary in order to obtain the control noise term needed for the covariance $P_{k|k-1}$. This does imply the costs of propagating the central sigma point will remain high, though the transform does still resolve the issue of violating state constraints via linear propagation.

Then, during the measurement update step, the sigma points are measured, the cross covariance P_{xy} and innovation covariance P_{yy} are found:

$$\bar{\gamma}_{k|k-1} = h(\bar{\chi}_{k|k-1}) \tag{2.47}$$

$$P_{xy} = \sum_{i=0}^{2n} W_i \left[x^{(i)}_{k|k-1} - \bar{x}_{k|k-1} \right] \otimes \left[y^{(i)}_{k|k-1} - \bar{y}_k \right] \tag{2.48}$$

$$P_{yy} = \sum_{i=0}^{2n} W_i \left[y^{(i)}_{k|k-1} - \bar{y}_k \right] \otimes \left[y^{(i)}_{k|k-1} - \bar{y}_k \right] \tag{2.49}$$

$$\bar{y}_k = \sum_{i=0}^{2n} W_i \bar{y}^{(i)}_{k|k-1} \tag{2.50}$$

Finally, the state update itself and its covariance for $t_{k|k}$ is given by Equation 2.51 and Equation 2.52 respectively, with $\vec{Y}_k$ as the measurement.

$$\hat{x}_{k|k} = \bar{x}_{k|k-1} + P_{xy} P_{yy}^{-1} (\vec{Y}_k - \bar{y}_k) \tag{2.51}$$

$$\hat{P}_{k|k} = P_{k|k-1} - (P_{xy} P_{yy}^{-1}) P_{yy} (P_{xy} P_{yy}^{-1})^T \tag{2.52}$$

The state estimate for $t_{k-1|k}$ is equivalent to that of the BL-OCBE and while Greaves and Scheeres[18] also provide adjoint and control estimate formulations, they are not reported here as they were not implemented during this thesis.

It is also important to note that while the unscented modification itself is limited to the updates summarized in this section, an algorithm implemented in this thesis and referred to as the U-OCBE (Unscented Optimal Control Based Estimator) includes the modifications in this section in addition to the adaptive dynamic uncertainty modification discussed in Subsection 2.1.3. Further details on the architecture and implementation of the full algorithms can be found in Section 3.2.

2.2. Regularization

As discussed in the introduction, a key aspect of this thesis is to investigate the effects of semi-analytical solutions or regularization on the OCBE. Unfortunately, covering multiple methods is not feasible considering time constraints. Instead, one regularization method was selected to be implemented and tested.

Introduced by Baù et. al.[28] is a regularization intended for the perturbed two body problem referred to as 'EDromo'. It has shown fantastic numerical performance [28] [29], even when compared to other regularization methods such as KS discussed in [27]. The primary drawback is that the performance is limited to the perturbed two-body problem.

That being said, if this regularization does not show significant improvement over traditional numerical solutions (when incorporated in the OCBE), it is likely that similar (if not worse) results could be expected from methods designed for broader applications. While such conclusions may be considered for various other regularization methods, the same can not be said about averaging methods. However, as discussed in the introduction, averaging remains limited in its versatility, and it has a fundamental limitations on accuracy. As a result, this thesis focuses on the regularization introduced in [28], the summary of which is presented in this section.

The regularization provided involves an 8 dimensional state $[\lambda_0, \lambda_1, ..., \lambda_7]$, with a Sundman transformation of time into angle φ as the independent variable. A conceptual overview of the orbital elements and the corresponding differential equations is presented here. However, the equations to reconstruct the inertial state from the orbital elements and vice versa are omitted from this report due to the relatively lengthy formulation. The full equations can still be found in the original work [28] using notation matching this thesis.

Returning to the meaning of the new state variables, the first corresponds to a choice of the time element. While the regularized dynamics equations can be applied with physical time as the independent variable, the best performance is achieved using an angle-like independent variable φ obtained by a Sundman transform. In order to reconstruct the physical time from the state variables (Equation 2.53), a 'linear' time element ($\lambda_{0,l}$) is necessary. It is possible to obtain this variable by integration directly, or instead to solve for a 'constant' time element $\lambda_{0,c}$ and then apply Equation 2.54. For the purposes of this thesis, $\lambda_{0,l}$ is used as the state variable throughout as it has shown to generally have a lower computational cost for the same accuracy compared to the alternatives [28]. λ_3 also appears in these equations and is the element representing the total energy, which must remain negative.

$$t = \lambda_{0,l} + \lambda_3^{3/2}(\lambda_2 \cos\varphi - \lambda_1 \sin\varphi) \tag{2.53}$$

$$\lambda_{0,l} = \lambda_{0,c} + \lambda_3^{3/2}\varphi \tag{2.54}$$

Defining the remaining elements requires a definition of an orbital frame and an 'intermediate' reference frame. The orbital frame is simply represented by the position $\vec{r}$ and angular momentum $\vec{h}$ vectors normalized (to $\vec{i}$ and $\vec{k}$) with the final axis $\vec{j}$ selected to complete the orthonormal frame. The intermediate frame is defined with respect to a generalized eccentricity vector given in Equation 2.55 as well as a generalized eccentric anomaly given in Equations 2.56 and 2.57 (where U is the dimensionless perturbing potential and ε is the dimensionless total energy). The orientation of the intermediate frame with respect to these parameters is shown in Figure 2.3.

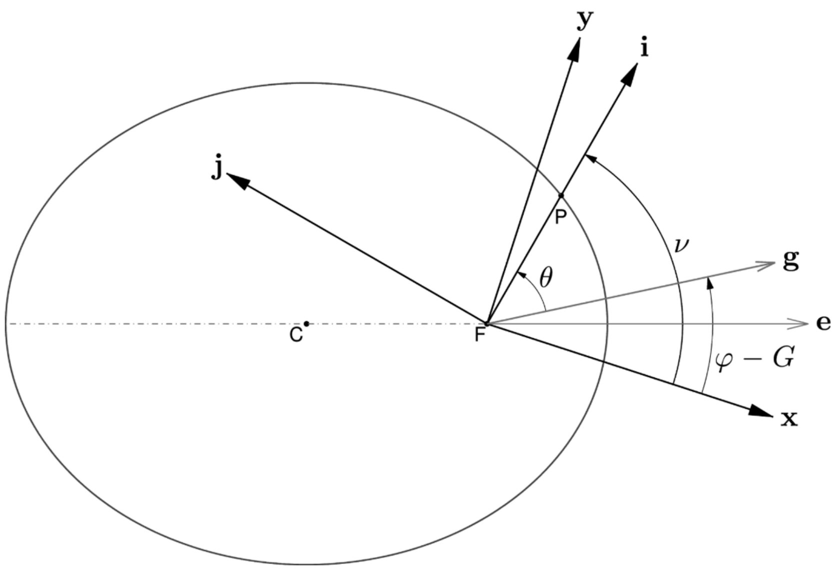


Figure 2.3: *The intermediate frame $\{F; \boldsymbol{x}, \boldsymbol{y}, \boldsymbol{k}\}$, as viewed from the $\boldsymbol{k}$ axis. The propagated object P occupies one point of the instantaneous osculating ellipse with a focus F and center C. $\boldsymbol{e}$ is the osculating eccentricity vector. Angles θ and ν are referred to as the generalized true anomaly and orbital longitude respectively, both of which are defined in terms of the geometry in this figure.[28]*

.

$$\vec{g} = -\vec{i} + \left(\frac{dr}{dt}\vec{i} + \frac{c}{r}\vec{j}\right) \times (c\vec{k}) \tag{2.55}$$

$$g \sin G = -2\varepsilon \frac{dr}{d\varphi} \tag{2.56}$$

$$g \cos G = 1 + 2\varepsilon r \tag{2.57}$$

with:

$$\vec{i} = \frac{\vec{r}}{r} \tag{2.58}$$

$$\vec{k} = \frac{\vec{h}}{h} \tag{2.59}$$

$$c = \sqrt{h^2 + 2r^2 U} \tag{2.60}$$

Based on this definition of the intermediate frame, elements λ_1, λ_2 are selected as first integrals of unperturbed motion, obtained using variation of parameters. They represent projections of the generalized eccentricity vector $\vec{g}$ onto two of the axes of the intermediate frame (**x** and **y**).
The remaining four state variables λ_4 - λ_7 are Euler parameters defining the attitude of the intermediate frame with respect to the orbital frame. This is sufficient to fully define the state of the propagated object, given the differential equations below.

The first of the differential equations of the regularization depends on the choice of time element. For traditional time, constant and linear time elements, the corresponding differential equations are given in 2.61, 2.62 and 2.63 respectively.

$$\frac{dt}{d\varphi} = \lambda_3^{3/2} \varrho \tag{2.61}$$

$$\frac{d\lambda_{0,c}}{d\varphi} = \lambda_3^{3/2} \left[(Rr - 2U) r + \frac{1}{\lambda_3} \left(\zeta - \frac{3}{2}\varphi \right) \frac{d\lambda_3}{d\varphi} \right] \tag{2.62}$$

$$\frac{d\lambda_{0,l}}{d\varphi} = \lambda_3^{3/2} [1 + (Rr - 2U) r + 2\Lambda_3 \zeta] \tag{2.63}$$

The remaining equations to be integrated that follow are:

$$\frac{d\lambda_1}{d\varphi} = (Rr - 2U) r \sin\varphi + \Lambda_3 [(1+\varrho)\cos\varphi - \lambda_1] \tag{2.64}$$

$$\frac{d\lambda_2}{d\varphi} = (2U - Rr) r \cos\varphi + \Lambda_3 [(1+\varrho)\sin\varphi - \lambda_2] \tag{2.65}$$

$$\frac{d\lambda_3}{d\varphi} = 2\lambda_3^3 \left(R_p \zeta + T_p n + \frac{\partial U}{\partial t} \sqrt{\lambda_3} \varrho \right) \tag{2.66}$$

$$\frac{d}{d\varphi} \begin{bmatrix} \lambda_4 \\ \lambda_5 \\ \lambda_6 \\ \lambda_7 \end{bmatrix} = N \frac{r^2}{2n} \begin{bmatrix} \lambda_7 c_v - \lambda_6 s_v \\ \lambda_6 c_v + \lambda_7 s_v \\ -\lambda_5 c_v + \lambda_4 s_v \\ -\lambda_4 c_v - \lambda_5 s_v \end{bmatrix} + \frac{\omega_z}{2} \begin{bmatrix} \lambda_5 \\ -\lambda_4 \\ \lambda_7 \\ -\lambda_6 \end{bmatrix} \tag{2.67}$$

Majority of the parameters given in these equations are functions of the independent variable φ or some combination of λ_1, λ_2 and λ_3. Notable exceptions are:

$$\begin{aligned} R &= \vec{F} \cdot \vec{i} \\ N &= \vec{F} \cdot \vec{k} \\ R_p &= \vec{P} \cdot \vec{i} \\ T_p &= \vec{P} \cdot \vec{j} \end{aligned} \tag{2.68}$$

Where $\vec{F}$ is the total dimensionless perturbing acceleration vector and $\vec{P}$ is the component of $\vec{F}$ not caused by a perturbing potential. The means to obtain the intermediate parameters found in previous equations are given in Equations 2.69 to 2.77:

$$\Lambda_3 = \frac{1}{2\lambda_3}\frac{d\lambda_3}{d\varphi} \quad (2.69) \qquad r = \lambda_3\varrho \quad (2.70)$$

$$n = \sqrt{m^2 - 2\lambda_3\varrho^2 U} \quad (2.71) \qquad m = \sqrt{1-\lambda_1^2-\lambda_2^2)} \quad (2.72)$$

$$\varrho = 1 - \lambda_1\cos\varphi - \lambda_2\sin\varphi \quad (2.73) \qquad \zeta = \lambda_1\sin\varphi - \lambda_2\cos\varphi \quad (2.74)$$

$$c_v\varrho = \cos\varphi - \lambda_1 + \frac{\zeta\lambda_2}{m+1} \quad (2.75) \qquad s_v\varrho = \sin\varphi - \lambda_2 - \frac{\zeta\lambda_1}{m+1} \quad (2.76)$$

$$\omega_z = \frac{n-m}{\varrho} + \frac{(2U - Rr)\left(2-\varrho+m\right)r + \Lambda_3\zeta\left(\varrho - m\right)}{m\,(1+m)} \tag{2.77}$$

One may notice that two perturbing force projections ($\vec{P}\cdot\vec{k}$ and $\vec{F}\cdot\vec{j}$) are not directly present in the dynamics. The cross-track contribution of non-potential forces ($\vec{P}\cdot\vec{k}$) is taken into account in Equation 2.67 as part of N, without separation from their counter part caused by a perturbing potential. On the other hand, the contributions of along-track potential forces are captured by the time derivative of the potential found in Equation 2.66, which then re-appears in multiple differential equations as part of the Λ_3 term.

It should also be noted that time, force and potential in these equations is non-dimensionalized. Converting to and from such a formulation requires distance and frequency factors DU (Equation 2.78) and TU (Equation 2.79) [29] obtained from the initial orbital radius R_0 and central body gravitational parameter μ.

$$DU = \sqrt{R_0 \cdot R_0} \tag{2.78}$$

$$TU = \sqrt{\mu / DU^3} \tag{2.79}$$

2.3. Acceleration Modelling

The final aspect of the problem yet to be covered, is the mathematical representations of the various physical elements in the environment. As mentioned in the introduction, the focus of the thesis lies on high Earth orbits such as GEO. As a result, some common aspects of dynamics modelling such as atmospheric drag are not relevant. Instead, the focus lies on modelling gravity and SRP (Solar Radiation Pressure), both of which are discussed in this section.

One of the objectives of this thesis is to evaluate performance of the OCBE given imperfect knowledge of the environment. As a result, the literature survey was carried out considering two separate environment models shall be necessary. The first and more complex - to represent the solution closer to reality, considered the 'true' environment. This environment model may be used for the purposes of simulating measurements and performing error calculations. The second and slightly simpler model - to simulate an imperfect, 'known' representation of the environment. This model is directly incorporated in the OCBE estimation procedure, used to propagate estimated states between measurement epochs.

Any numerical solution parameters or models not discussed in this section or Chapters 5 and 6 such as body ephemerides can be considered the default options in the tudat astrodynamics library [1]. In

[1] https://docs.tudat.space/en/latest/_src_user_guide/state_propagation/environment_setup/default_env_models.html

most cases, this simply implies NASA's SPICE toolkit [41].

2.3.1. Gravity

When it comes to celestial body gravity models, two were considered excluding the traditional point-mass representation, primarily due to their availability in the tudat astrodynamics library. This includes the GOCO05c spherical harmonic gravity model introduced by Fecher et. al. [42] as well as the model discussed by Werner and Scheeres representing a body as a homogeneous mass distribution polyhedral [43].

The former was deemed much more suitable to the problem at hand as it allows for easy tuning of fidelity and is based on readily available data and coefficients, whereas the polyhedron model is much more suited to modelling smaller bodies like asteroids which are not considered in this thesis. Hence, any non-point-mass gravity representation discussed in this report is computed using the gravitational potential given in Equation 2.80 as a function of co-rotating spherical coordinates $[\phi, \theta, r]$.

$$U(\phi,\theta,r) = \sum_{l=0}^{l_{max}} \sum_{m=0}^{l} \mu \frac{R^l}{r^{l+1}} \bar{P}_{lm}\left(\sin\phi\right) \cdot \left[\bar{C}_{lm}\cos(m\theta) + \bar{S}_{lm}\sin(m\theta)\right] \tag{2.80}$$

where l and m are the degree and order of the spherical harmonic term, $\bar{P}_{lm}$ is the corresponding normalized Legendre polynomial [44], $\bar{C}_{lm}$ and $\bar{S}_{lm}$ are the GOCO05c determined normalized harmonic coefficients, and μ and R are the gravitational parameter and reference radius of the central body.

2.3.2. Solar Radiation Pressure

When it comes to solar radiation pressure, two models for the target are also available in tudat. However, instead of selecting one of the two like was done in the case of gravity, both models are to be used. One for the true environment and one for the imperfect, known version. For the latter, the commonly known cannonball model given in Equation 2.81 shall be used.

$$\vec{a}_{SRP} = -\frac{\Phi C_r A}{mc}\hat{e}_{\odot} \tag{2.81}$$

On the other hand, for a representation closer to real conditions, the spacecraft can be modelled as a cuboid with attached rectangular solar panels. The specular-diffuse model that determines acceleration contributions of each individual surface element is given by Equation 2.82[45].

$$\vec{a}_{SRP} = -\frac{\Phi}{mc}A\cos\theta_{\odot n}\left[(\alpha+\delta)\left(\hat{e}_{\odot}+\frac{2}{3}\hat{e}_n\right)+2\rho\cos\theta_{\odot n}\hat{e}_n\right] \tag{2.82}$$

For both models Φ, m, A and c represent the solar flux, mass, (effective) surface area and speed of light respectively. C_r is the reflectivity coefficient. α, δ and ρ are the surface absorption coefficient, diffuse reflectivity and specular reflectivity respectively. $\hat{e}_{\odot}$ represents the unit vector in the direction of the radiation source. $\hat{e}_n$ is the surface normal unit vector. And finally, $\theta_{\odot n}$ is the angle between the two aforementioned vectors.

3

Methods

The primary objective of the thesis is to evaluate the benefits and drawbacks of the regularized formulation OCBE algorithm. As a result, it is crucial to define a method that will not only allow for implementation of the updates, but also a fair comparison with variants of the OCBE from literature. This chapter introduces the overarching method used to perform this comparison in Section 3.1 and describes the general architecture of the implemented algorithms in Section 3.2.

3.1. Thesis Approach

As previously mentioned, the main goal of the thesis is to quantify the effects of the regularization applied to the OCBE. Here a short summary of the approach to answer the research questions defined in Chapter 1 is provided.

First of all, in order to ensure the comparison of the algorithms is meaningful and easily reproducible, the results should be obtained in well documented, life-like test cases. Thus, the first step of the thesis primarily involves defining the orbit initial states and selecting the parameters of the observed object or spacecraft. Based on the findings in the Introduction, three distinct cases in the GEO region were chosen (and are described in more detail in Chapter 5, prior to the discussion of numerical results of the thesis).

With the test cases defined, the second step was to implement the numerical propagators necessary for the estimation algorithm. As per the research questions, the EDromo propagator discussed in Section 2.2 was implemented and tested, in addition to a Cowell formulation to act as a reference. Numerical propagation errors are generally rather sensitive to the choice of orbit and environment model. If not investigated and handled appropriately, these errors could affect the results of the OCBE and potentially even be wrongly attributed as a property of one of the estimator variants when drawing conclusions. To avoid this, in the third step, error sources such as truncation and floating point error were quantified to appropriately select numerical integration tolerances. Additionally, in order to further improve computational speed of the methods, the effects of various aspects of the environment (such as fidelity of the Earth's gravity model) were quantified, and sufficiently small contributions were neglected. This process is discussed in further detail in Chapter 6.

With preparatory work complete, the fourth step was implementing reference cases of the OCBE which perform state estimation in terms of Cartesian position and velocity, using dynamics in terms of a Cowell formulation. These were the A-OCBE and U-OCBE methods discussed in more detail in Subsection 3.2.2, both of which are based on the literature findings summarized earlier in Section 2.1. When applied to the selected test cases, they may act as a control group, providing a comparison point for the new (regularized) variant of the OCBE.

Then, the prototype of the regularized version of the algorithm was developed. This involved a few steps. First of all, as the R-OCBE is based around estimation of regularized elements rather than the Cartesian state, a number of modifications to the OCBE were made to facilitate these changes. This involves the derivation of a new control input scaling matrix, conversions of covariance matrices to the EDromo elements, et cetera. As the U-OCBE lends itself better to the necessary modifications than the A-OCBE, the implementation of this variant of the estimation algorithm was used as the baseline for the R-OCBE. An overview of this algorithm is provided in Subsection 3.2.3, while the individual modifications developed as part of this thesis are discussed in more detail in Chapter 4. Finally, with the algorithm changes applied and the method validated, its performance could be evaluated when applying it to the previously selected test cases while varying the input parameters.

With all the aforementioned steps completed, the results could be used to compare all three of the implemented versions of the OCBE based on a number of performance metrics to answer the research questions. Any potential benefits or drawbacks of the R-OCBE may thus be discussed, and all conclusions may be documented.

The thesis approach is thus summarized as follows:

1. Define test cases
2. Implement reference and regularized propagators
3. Quantify error sources
4. Implement 'Control Group' OCBE variants
5. Implement R-OCBE
6. Post-Process, Compare Algorithms, Analyze Results and Draw Conclusions

3.2. Algorithm Architecture

This section provides an overview of the algorithms implemented to answer the research questions. As all implemented algorithms were eventually incorporated in a single system, a full overview of the structure of the implementation is first provided in Subsection 3.2.1. Then, the most important components are discussed in more detail in the following subsections. This involves the dynamics propagation methods (Subsection 3.2.2), the estimation methods (Subsection 3.2.3) as well as the interface used to handle user inputs (Subsection 3.2.4).

3.2.1. Overview

As mentioned in the section introduction, all implemented algorithms are incorporated in a single system. This was done in order to facilitate bulk state estimation result generation, based on varying input parameters, in order to allow for convenient comparison of the implemented variants of the OCBE. An overview of this implementation is provided in form a block diagram in Figure 3.1, which is divided in four modules: The interface, the estimator and two dynamics solvers. The first of which being the 'Reference Dynamics Solver', used to generate truth data. The second being the 'Primary Dynamics Solver', which is used as part of the estimation process to propagate dynamics between measurement epochs.

As shown in Figure 3.1, the interface module acts as the bridge between the user input, primary dynamics solver and estimator. It is first used to store any input parameters ranging from measurement uncertainty and spacecraft initial state to choice of estimator and Earth's gravity model.

These inputs are then used to select propagator settings, which are then passed to the primary dynamics solver, such that relevant functions of the dynamics model can be defined. This includes methods to compute necessary perturbing accelerations, numerical integration with chosen tolerances, etc. These defined functions may then be combined to create a single propagation function, which is passed back to the interface, such that the estimator can request propagation of a trajectory between measurement epochs. Furthermore, the primary dynamics solver initializes storage of ad-

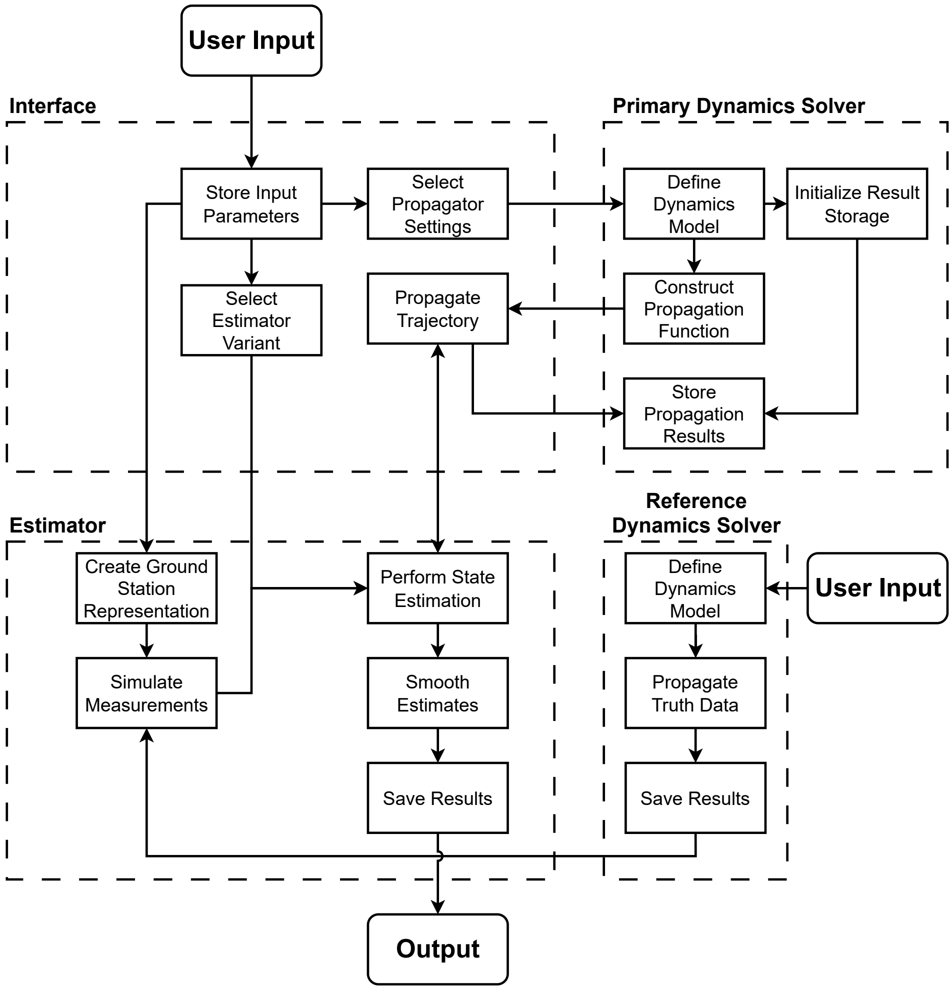


Figure 3.1: *Overview of the implemented estimation system*

ditional results that may be of interest for the investigation of performance for the full system. This includes values such as the total number of state derivative function evaluations over all requested propagations.

Another set of input parameters are used to configure the estimator module. This involves selection of the estimation method (between A-OCBE, U-OCBE and R-OCBE) as well as initialization of a ground station with appropriate location and measurement errors, which is in turn used to simulate a number of noisy measurements based on provided truth data. With the measurements obtained, the estimator selected and dynamics propagation functions available, the state estimation for the trajectory is performed. The results are then (usually) smoothed, and output with appropriate auxiliary information to identify the inputs used to generate the result.

It should be noted that the truth data is generated prior to the rest of the estimation procedure by the reference dynamics solver. This is done for two reasons. Firstly, this allows for a significantly sped up estimation process when testing different variants or settings of the OCBE on the same trajectory as the high fidelity truth data is not re-propagated multiple times.

In addition, state estimation with the OCBE enforces certain requirements on the propagator implementation such as propagating a 12×12 state transition matrix for the state and the adjoint. This information is not necessary for the truth data, where only the state as a function of time is necessary. Hence, using an alternate dynamics solver (completely decoupled from the rest of the system), for the truth data generation relaxes the implementation requirements and allows for higher fidelity models that may be too cumbersome to implement for an early version of the OCBE. For example, use of environment models and integration schemes from Tudat (TU Delft Astrodynamics Toolbox)[1] is accommodated by this implementation choice.

3.2.2. Dynamics Solver Modules

Primary Dynamics Solver

As previously discussed, the state estimation process with the OCBE requires the propagation of trajectories and corresponding state transition matrices. To obtain these results a dynamics solver (referred to as the "Primary" dynamics solver) was implemented. As it was deemed that the module will be used for a relatively limited number of short duration numerical propagations and thus computational speed will not be a particularly limiting factor (in the context of this thesis), the dynamics solver was implemented entirely in Python 3.10.

The goal of this module is to configure the functions necessary to perform state propagation, whereas the state propagation itself is called by the estimator module as required. In consequence, the steps discussed in this subsection are performed only once per set of measurements used in estimation and the initialization does not have to be repeated, regardless of number of propagations.

The architecture of the module is given in Figure 3.2. As the module was developed with the intent to incorporate it in the larger estimation process via an interface, it is initialized based on inputs received. This involves a few core aspects:

1. The definition of the propagated body. This includes various necessary parameters such as mass or reference area for solar radiation pressure.
2. The definition of the environment bodies. This involves setting gravitational parameters, ephemerides, et cetera.
3. Initialization of result storage. While not strictly necessary for a propagation, this process allows to track performance metrics that may be of interest over multiple propagations, such as the total number of state derivative evaluations.

Following the definitions of the bodies in the dynamics system, the perturbing acceleration functions of time and state are defined. As the EDromo propagator (used for the R-OCBE) requires primary body perturbing potential, the relevant equations are also defined if necessary.

The rest of the process depends on the selected propagator and the choice whether the state transition matrix should be propagated. For the Cowell propagator, the 6-dimensional state derivative function is defined by simply taking the sum of the perturbing acceleration functions in addition to the point mass of the central body. If the STM is also requested, a function to obtain the 12×12 matrix derivative is defined.

In order to ensure the output is compatible with the numerical integrator, regardless of the selection, an ‘extended’ state derivative function is defined. If the STM is not propagated, this function is identical to the previously defined state derivative. Otherwise, it handles reshaping of the STM matrix derivative into a vector combined with the state derivative.

[1] https://docs.tudat.space/en/latest/

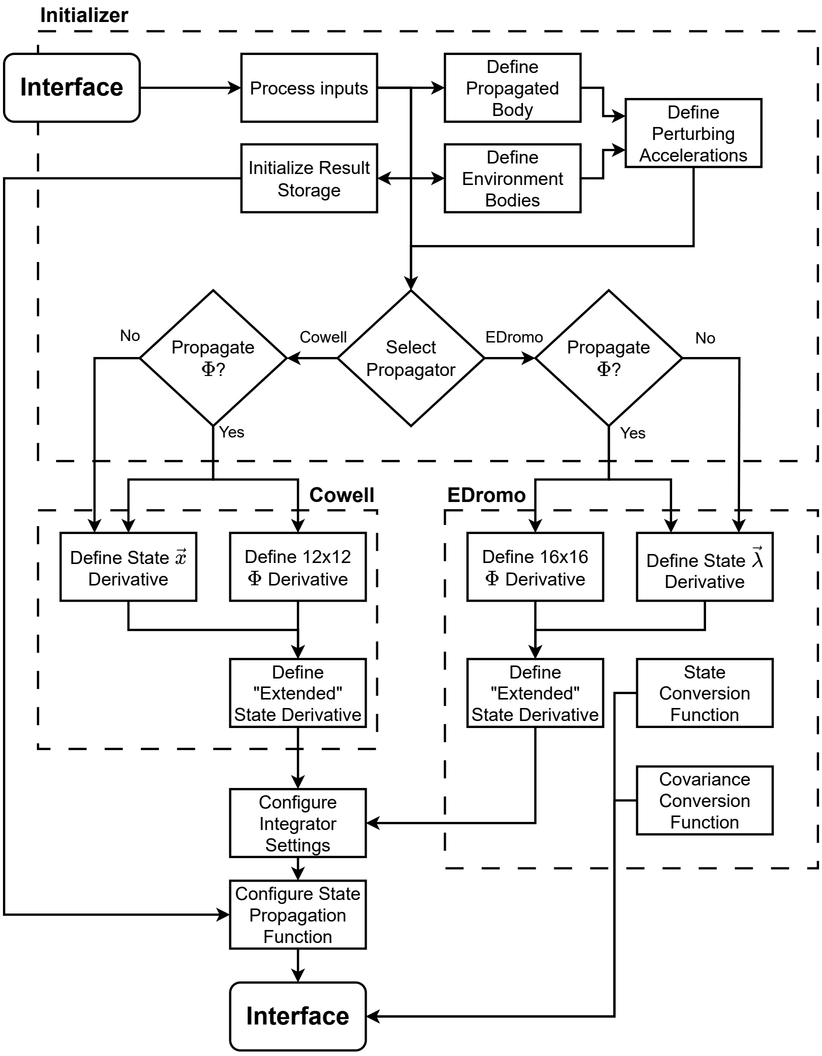


Figure 3.2: *Overview of the Dynamics Solver Module*

In the case of the EDromo propagator, the process is analogous, except the 8-dimensional state $\vec{\lambda}$ derivative is defined in terms of the equations given in Section 2.2. Similarly, a larger state transition matrix is necessary with a different formulation necessary to define Φ_{xp} (compared to Cowell), which is further discussed in Chapter 4. Regardless, a creation of an 'extended' state derivative function is also performed.

Finally, the integrator tolerances, initial and maximum step are set for the implemented RKF7(8) scheme [46], which is used to create the final output of the module, a state propagation function. However, in case of a regularized propagation, the independent variable is not time, and the value of the independent variable corresponding to a precise final time is not known before the start of the propagation. This provides a slight complication as for state estimation, a propagated state value precisely at the time of a measurement is necessary. To handle this, the following modification to the numerical integrator was made.

As part of the numerical integration process, to ensure the propagation terminates appropriately, the time is reconstructed from the independent variable φ after every integration step. If it exceeds the time of the measurement (t_m), a root finding algorithm is initialized solving Equation 3.1 for $d\varphi$:

$$0 = t_\mathrm{m} - t(d\varphi) = t_\mathrm{m} - \left(\lambda_{0,l} + \lambda_3^{3/2}(\lambda_2 \cos\varphi - \lambda_1 \sin\varphi)\right) \tag{3.1}$$

In which $t(d\varphi)$ is expanded using Equation 2.53. In this case, $\lambda_{0,l}$, λ_1, λ_2, λ_3 and φ are all functions of the step $d\varphi$, and for each iteration of the root finder, an integration step must be performed. This causes some performance losses, though the impact is treated in more detail in future chapters.

The solution is found using the classic Brent's method [47]. While other root-finding schemes were investigated, Brent's method was chosen as it is guaranteed to converge to a solution given a sign changing interval for the independent variable. As Equation 3.1 is by definition positive before the end of the propagation and negative if the integration step exceeds the final time, this condition is always satisfied. Other methods such as bisection [48] also provide such properties though Brent's method was found to converge in fewer iterations despite retaining the same accuracy. Being a 0th order method, it was also chosen over n-th order methods as they would require the evaluation of additional derivatives of the state vector, or a finite difference approach which is also computationally cumbersome.

With the integration method treated, the state propagation function can be fully defined and output to the interface module for use in state estimation. In the case of R-OCBE, the dynamics module also outputs functions mapping Cartesian state $\vec{x}$ to $\vec{\lambda}$ (and vice versa) as well as equivalent conversions for the Covariance matrix (which are discussed in Section 4.3). These methods are not strictly part of the propagation process, though including them in the dynamics solver module allows for minor computational performance gains due to re-occurring variables.

Reference Dynamics Solver
In addition to the dynamics solver discussed earlier in this section, a reference dynamics solver implemented using the Tudat library was also used for two purposes: generation of truth data (as seen earlier in Figure 3.1) and validation of the primary dynamics solver.

The structure of the reference dynamics solver is relatively similar to the one discussed in the previous section, though a few key differences are present. The reference solver:

- does not support propagation in regularized elements
- only supports propagation of 6×6 state transition matrices
- supports higher fidelity dynamics models
- supports additional numerical integration schemes

The lack of support for regularized elements and state transition matrices with adjoint components implied it was not easy to use in state estimation with the OCBE. However, those properties are not necessary for the truth data generation and the advantages provided by Tudat also implied it could be used to perform a dynamics model selection and error analysis, in addition to validation for the custom implemented primary dynamics solver, which is then incorporated into the OCBE estimation process. This selection procedure and error analysis follows in Chapter 6.

3.2.3. Estimator Module

Measurement Simulation
As discussed at the start of this section, the estimator module contains three methods: The A-OCBE, U-OCBE and R-OCBE. While each implementation is discussed below, all three methods share the same process for generation of measurements of a given trajectory. Each measurement is a vector

of the range, azimuth and elevation of the spacecraft with respect to the ground station. The approach to obtain a series of measurements over a trajectory is shown on the left side of Figure 3.3, and involves two key steps.

First, given inputs from the interface, a ground station is defined. This involves the location (defined in terms of latitude and longitude), the standard deviations of the measurement (in terms of range, azimuth and elevation) and the time gap dt between each measurement.

Then, the measurements themselves are simulated using the defined ground station and a 'true' trajectory provided by the Reference Dynamics Solver. In order to ensure a value of the true trajectory is available regardless of the integration step and/or measurement gap, it is interpolated using a 7th order Lagrange interpolation scheme [49]. Furthermore, a transparent Earth is assumed, hence the single ground station is sufficient to provide measurements regardless of the relative location of the spacecraft. This assumption was made considering that the goal of the system is to evaluate the estimator performance in various conditions. A transparent Earth allows to easily obtain dense measurements with only one ground station initialized, while sparse measurements resembling the effect of occlusion by the Earth can still be simulated by specifying large measurement gaps. The definition of the ground station thus allows for evaluation of the true range, azimuth and elevation of the spacecraft every dt seconds. Following which, error vectors sampled from Gaussian distributions, with the aforementioned standard deviations are added to simulate measurement noise.

Adaptive Ballistic-Linear OCBE

The first of the implemented variants of the OCBE is the adaptive ballistic linear optimal control based estimator. For purposes of brevity, it is also referred to as the adaptive OCBE or A-OCBE in this report. This method's integration in the estimator module is also shown in Figure 3.3. As discussed in Chapter 2, the method begins with the time update step of the BL-OCBE, given initial conditions and a dynamics propagation function. This allows for the second step, the evaluation of the control distance metric defined in Subsection 2.1.3, which, if larger than a specified threshold, indicates mis-modelled dynamics above the expected dynamic uncertainty have been detected.

Generally, the detection of such mis-modelled dynamics in the A-OCBE implies the dynamic uncertainty should be adjusted using the method in Subsection 2.1.3 to improve the accuracy of the estimator. However, in order to ensure a single measurement outlier does not result in a false positive, mis-modelled dynamics detection in three consecutive time steps is necessary to trigger the process. Should this be the case, it is beneficial to adjust the dynamic uncertainty before the first of the consecutive detections, hence the estimation process for the three most recent measurements is performed again, given an updated dynamic uncertainty.

Otherwise, the BL-OCBE measurement update step (from Section 2.1) is used to obtain new state estimates, until a state estimate is available for every measurement epoch. As a final step, before the output is produced, the BL-OCBE smoothing algorithm (also discussed in Section 2.1) is applied.

Unscented Adaptive Ballistic-Linear OCBE

The second implemented variant of the OCBE is the Unscented Adaptive Ballistic-Linear OCBE, also simply referred to as the Unscented OCBE or U-OCBE in this report. The Unscented version of the algorithm in terms of structure is quite similar to the Adaptive, and is shown on the right hand side of Figure 3.3, though it does contain some significant differences in the time and measurement update steps.

Unlike the A-OCBE, the Unscented method begins with the evaluation of sigma points as part of the unscented transform as discussed in Subsection 2.1.4. In the time update step, each of these sigma

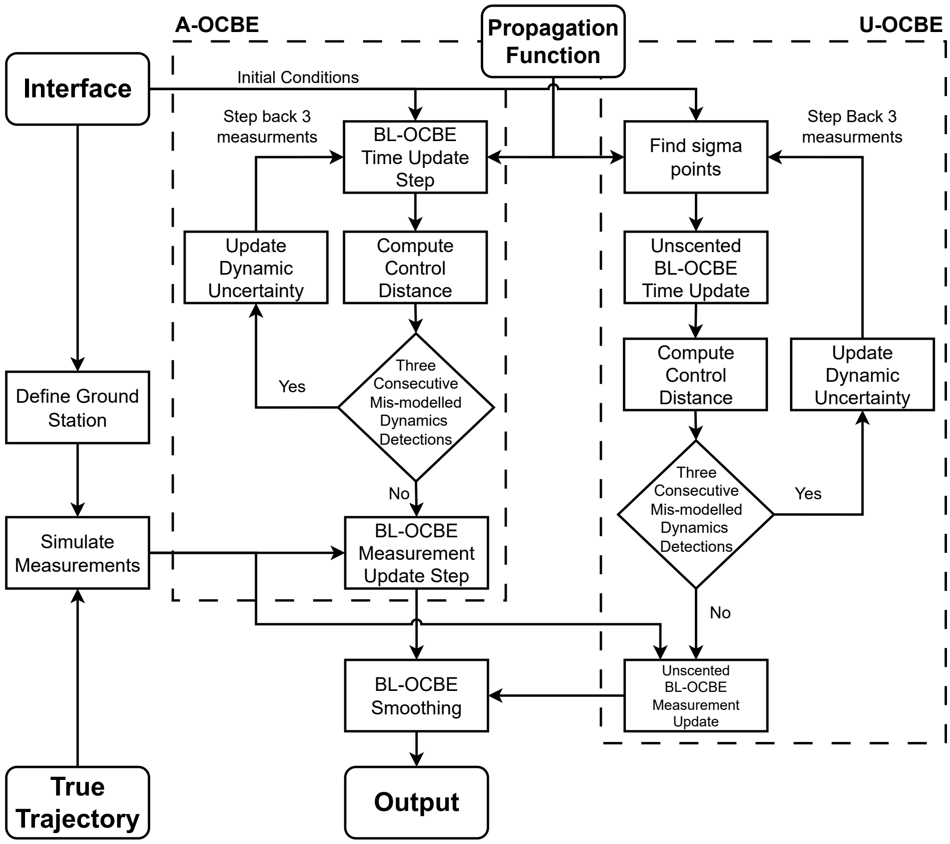


Figure 3.3: *Architecture of the implemented A-OCBE and U-OCBE algorithms in the Estimator Module*

points is propagated individually (hence reducing the speed of the algorithm), and a new predicted mean state and covariance at the time of the measurement is obtained.

In the same fashion as the Adaptive algorithm, at this stage, the control distance metric is computed and may be used to update the dynamic uncertainty given three consecutive mis-modelled dynamics detections. If this is not necessary, the Unscented version of the measurement update step discussed in Subsection 2.1.4 is applied until all measurements are treated and smoothing may be applied.

Regularized OCBE

Finally, the regularized optimal control estimator may be introduced. In order to pursue performance gains, two distinct aspects of the OCBE are adjusted. First of all, the dynamics are propagated using the EDromo elements introduced in Section 2.2 in hopes of reducing the costs of propagation between measurement epochs. Second, the time and measurement update equations are used to obtain state estimates in terms of the EDromo state vector $\vec{\lambda}$, rather than the Cartesian state $\vec{x}$. This modification allows for the application of a linear estimator, on underlying linear-perturbed dynamics (rather than the fully non-linear Cartesian equivalent). This was done with the hopes of improving the range of measurement gaps over which the linear estimator produces accurate results.

While there is no inherent reason as to why these modifications can not be made to the A-OCBE, the prototype R-OCBE implemented as part of this thesis is based on the time and measurement update steps of the unscented version of the algorithm. This was done primarily for the following reasons:

1. The U-OCBE is very computationally intensive, and incorporation of regularized dynamics may compensate for this weakness.
2. The U-OCBE is expected to perform better than A-OCBE (in terms of accuracy) under conditions of non-linear dynamics with long measurement gaps. These are conditions under which the EDromo propagator is expected to perform best.
3. The U-OCBE lends itself easier to modification, as terms such as $\tilde{H}_k$ (linearization of the measurement function), which are not readily available for regularized elements are not necessary for implementation.

Additionally, the smoothing and adaptive steps of the OCBE are omitted from this implementation due to limited time for the validation of the results given these modifications. While this is expected to diminish the performance to a degree, and generally it is recommended to investigate the incorporation of these steps in the future, their absence may allow for easier interpretation of the behaviour of the method. Regardless, Figure 3.4, summarizes the method, with the same inputs and an identical measurement generation process as the other implemented OCBE variants, on the left hand side of the figure.

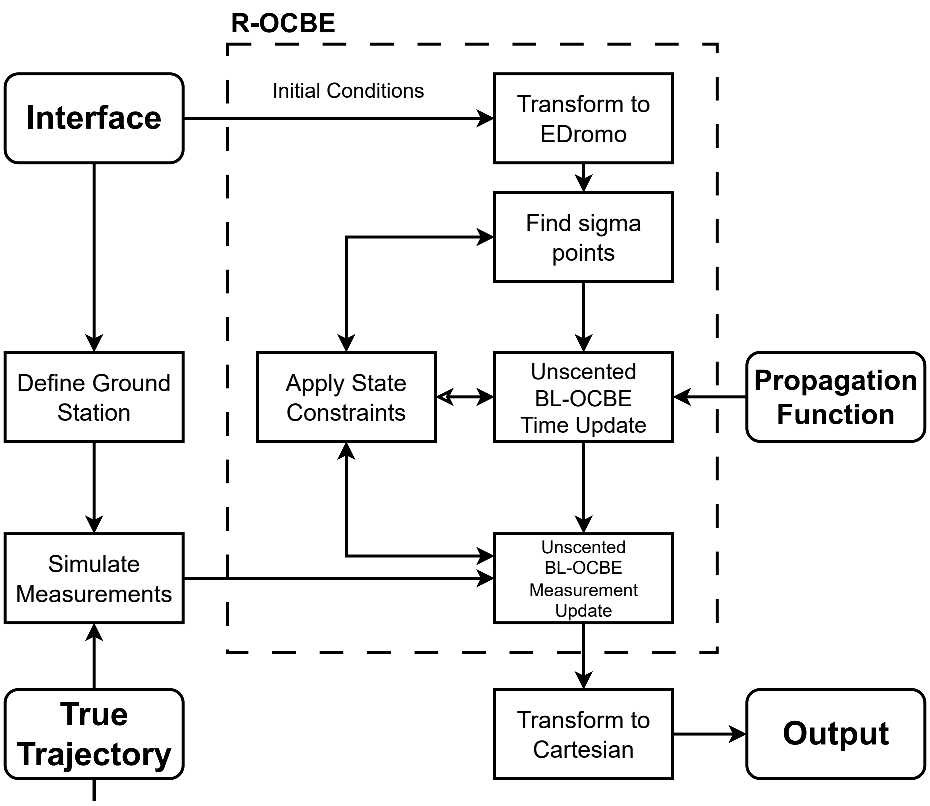


Figure 3.4: *Architecture of the implemented R-OCBE algorithms in the Estimator Module*

As the estimation in the R-OCBE is performed in terms of the regularized elements, the first step in the process is the transformation of the initial state. While this process for a state vector is discussed in [28], the covariance transformation is not treated. Hence, prior to the implementation of the R-OCBE, two methods for a covariance transformation were compared and the details of the selection process are discussed in Section 4.3.

With the regularized elements available, the unscented approach is followed in order obtain state estimates, starting with evaluation of sigma points, followed by the time and measurement update steps. However, unlike the Cartesian state, the regularized elements must adhere to constraints. This implies that any time when a state is modified outside propagation, it must be ensured it adheres to the constraints. This process is necessary:

1. Each time a sigma point is found using Equation 2.39.
2. When mean of the propagated sigma points is evaluated in the time update step using Equation 2.45.
3. When the state vector is updated using a correction obtained with Equation 2.51 for the measurement epoch or Equation 2.22 for the priori epoch.

The exact procedure that was developed and is performed at each of these steps is introduced in Section 4.1.

Finally, the OCBE equations require the propagation of an extended state transition matrix. The formulation for an appropriate control input scaling matrix B, used to obtain, the Φ_{xp} quadrant is not available from literature, hence it was developed and introduced in Section 4.2.
As with the other variants of the OCBE, the process is repeated until an estimate is available for every measurement epoch, at which point the results are saved for post-processing.

3.2.4. Interface Module

The final module yet to be discussed in detail is the interface, which handles the user input and incorporates appropriate dynamics into the OCBE. Fortunately, given the context of the previously discussed modules, its functionality is quite simple and the four blocks seen in Figure 3.1 are a largely sufficient representation.

The interface's functionality begins with storing of the input parameters, the format of which is defined for easy automated iteration. For the sake of completion, the list of possible input parameters is as follows:

- Run Identifier (for output documentation)
- OCBE variant to use
- Truth data source
- Orbit Initial State
- Known Dynamics Model Fidelity
- Initial State Covariance P_0
- Known satellite properties (i.e. mass)
- Time gap between measurements
- Location of ground station

With the inputs stored, the relevant parameters are passed to the dynamics module, which returns a state propagation function. In cases where the R-OCBE is used, the functions to convert to/from Cartesian state and EDromo elements and covariances (introduced in Section 4.3) are also obtained.

The remaining inputs in addition with these functions are used to run the selected OCBE variant (between A-OCBE, U-OCBE and R-OCBE). All results are written to files organized based on an identifier passed as one of the inputs to the interface module. Any post-processing of results can then be handled externally.

4
OCBE Regularization

As discussed in the previous chapter, a number of modifications to the U-OCBE were made in order to facilitate state estimation in terms of EDromo elements. This chapter summarizes the modifications made, starting with Section 4.1 which discusses the method used to update the state given an estimated correction. Then, Section 4.2 shows the derivation of the control input scaling matrix necessary for components of the STM. Finally, Section 4.3 covers the covariance transformation between a Cartesian representation and one corresponding to the regularized elements.

4.1. State Correction Method

Firstly, as the OCBE provides a state correction as part of the estimation process, a means to apply this to the state is necessary. For a state representation in terms of Cartesian coordinates, this is simply handled with a summation:

$$\hat{x} = \vec{x} + \delta\vec{x} \tag{4.1}$$

However, when treating the EDromo λ elements, complications arise if this approach is taken. First of all, the last four components of the state vector contain a quaternion. An addition of an arbitrary correction vector $\delta\vec{\lambda}$ may violate the unit norm constraint (Equation 4.2), which can in turn lead to complications when converting the state back to a Cartesian representation.

$$1 = \lambda_4^2 + \lambda_5^2 + \lambda_6^2 + \lambda_7^2 \tag{4.2}$$

This can be remedied by simply normalizing the relevant components using Equation 4.3.

$$\vec{\lambda}_{4-7,norm} = \frac{\vec{\lambda}_{4-7}}{\|\lambda_{4-7}\|} \tag{4.3}$$

Furthermore, as given in Equation 2.53, the physical time is reconstructed given not only the independent variable φ, but also $\lambda_{0,l}$, λ_1, λ_2 and λ_3, all of which may be adjusted in an estimation process. If this effect is not handled, a state obtained with a correction would correspond to a time different to that of the measurement used in the estimation process.

Fortunately, as the time of the measurement is known (and fixed) and the state elements are obtained from the estimator, the constraint may be applied in terms of the independent variable φ, an updated value of which can be solved for numerically. This is done by rewriting Equation 2.53 as follows and applying a numerical root finding method:

$$0 = -t + \lambda_{0,l} + \lambda_3^{3/2}(\lambda_2 \cos\varphi - \lambda_1 \sin\varphi) \tag{4.4}$$

Note that a similar process to obtain φ is necessary to terminate a propagation at a precisely selected final time, which was discussed earlier, in Subsection 3.2.2. In said discussion, the Brent's root finding method was selected to obtain a solution. However, this method requires specification of an interval of φ, which results in a sign changing right hand side of Equation 4.4. Such an interval is not possible to guarantee given an arbitrary update to the state vector $\vec{\lambda}$, hence a different root-finding method is necessary for this application. Instead, a first order method (Newton-Raphson) was chosen as it only requires an initial guess of φ instead of a full interval, while still resulting in quick convergence.

With an updated value of the independent variable obtained, it may be stored next to the state estimate, which can then be used to obtain a Cartesian representation of the state at the appropriate epoch for post-processing and result analysis.

Hence, the three step process shown in Figure 4.1 is applied in the R-OCBE when a state correction $\delta\vec{\lambda}$ is applied. However, it should be noted that Equation 4.4 is not guaranteed to converge for all state corrections. This is further discussed in Section 7.1.

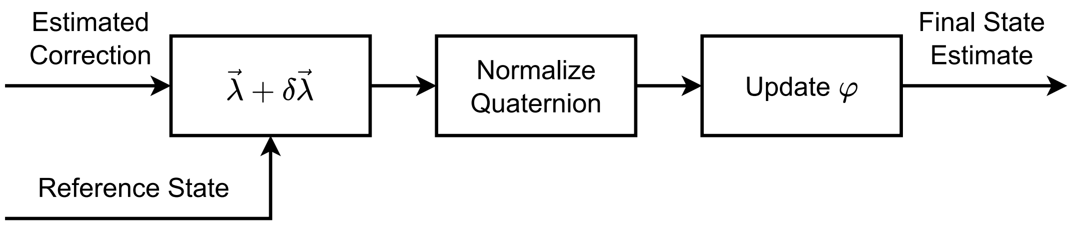


Figure 4.1: *Steps taken to update a reference state with an estimated correction*

4.2. Control Input Scaling

An important aspect of the regularized dynamics that differs from the Cartesian formulation is the handling of accelerations. While in a Cowell propagation any acceleration is a direct contribution to the velocity derivative, the effect on EDromo variables is more complicated.

This is key for the OCBE as if control inputs found in the estimation are defined as accelerations, they can be used to reconstruct the magnitude of maneuvers or environment mis-modelling. In order to ensure such a definition, an appropriate control input scaling matrix B is necessary when finding the state transition matrix quadrant relating the state to the adjoint (Φ_{xp} in Equation 2.15).

Hence, the control input scaling matrix B, mapping the control input in Cartesian coordinates, in the inertial reference frame to the derivative of the regularized elements is necessary. This is done in two steps, first a matrix B_1 is obtained, mapping the Cartesian control vector to a vector of projections used in the EDromo element derivative equations. Then a matrix B_2 is found by collecting relevant terms in the differential equations, to map the aforementioned vector to a state derivative contribution. The desired matrix is then given by $B = B_2 B_1$.

The introduction of the B matrix revolves around splitting the natural dynamics and the contribution of the control as shown in Equation 4.5, where $\frac{d\vec{\lambda}}{d\varphi}$ is given by Equation 2.63 through Equation 2.67

$$\frac{d\vec{\lambda}}{d\varphi} = f(\vec{\lambda}, \vec{u}) = \left(\frac{d\vec{\lambda}}{d\varphi}\right)_n + \left(\frac{d\vec{\lambda}}{d\varphi}\right)_u = \left(\frac{d\vec{\lambda}}{d\varphi}\right)_n + B \cdot \vec{u} \tag{4.5}$$

In the equations, the control input appears as part of the total dimensionless perturbing acceleration $\vec{F}$ as well as it's non-potential component $\vec{P}$. Both of these appear as projected components R, N, R_p, T_p which were also introduced in Equation 2.68. To obtain the the B matrix, the contributions of natural perturbations and the control are separated as follows: (taking advantage of the control input not being caused by a potential and thus $\vec{P}_u = \vec{F}_u$):

$$\vec{F} = \vec{F}_n + \vec{F}_u = -\frac{\partial U}{\partial \vec{r}} + \vec{P}_n + \vec{P}_u \tag{4.6}$$

$\vec{P}_u = \vec{F}_u$ also implies $R_u = R_{p,u}$, hence the control input can be represented by a vector as given by Equation 4.7. To ensure the control acceleration is dimensionless factors DU and TU given by Equations 2.78 and 2.79 are also included.

$$\vec{u}_{EDromo} = \begin{bmatrix} R_u \\ T_{p,u} \\ N_u \end{bmatrix} = B_1 \cdot \vec{u} = \frac{1}{DU \cdot TU^2} \begin{bmatrix} i_1 & i_2 & i_3 \\ j_1 & j_2 & j_3 \\ k_1 & k_2 & k_3 \end{bmatrix} \cdot \vec{u} \tag{4.7}$$

With $\vec{u}_{EDromo}$ obtained, if another matrix B_2 mapping this 3x1 vector to the 8x1 state derivative can be obtained, the full B matrix is determined. This is achieved by splitting the perturbing acceleration (R, N, R_p, T_p) terms in equations Equation 2.63 through Equation 2.67. into purely natural components, and ones dependent on control (i.e. $R = R_n + R_u$)

As the contribution of $\vec{u}$ to $\frac{d\lambda_3}{d\varphi}$ also appears in the remaining vector components, it is treated first. Equation 4.8 shows Equation 2.66 after having separated the natural dynamics (first row) from the control contribution (second row).

$$\begin{aligned} \frac{d\lambda_3}{d\varphi} &= 2\lambda_3^3 \left(R_{p,n}\zeta + T_{p,n} n + \frac{\partial U}{\partial t}\sqrt{\lambda_3}\varrho \right) \\ &\quad + 2\lambda_3^3 \left(R_{p,u}\zeta + T_{p,u} n \right) \\ &= \left(\frac{d\lambda}{d\varphi} \right)_n + \left(\frac{d\lambda}{d\varphi} \right)_u \end{aligned} \tag{4.8}$$

From this, one of the rows of the B_2 matrix can be constructed such that the definition $\left(\frac{d\lambda}{d\varphi}\right)_u = B_2 \vec{u}_{EDromo}$ is satisfied:

$$\left(\frac{d\lambda_3}{d\varphi} \right)_u = 2\lambda_3^3 \left(R_{p,u}\zeta + T_{p,u} n \right) = \left(2\lambda_3^3 \zeta \right) R_u + \left(2\lambda_3^3 n \right) T_{p,u} \tag{4.9}$$

$$\therefore B_{2,\lambda_3} = \begin{bmatrix} 2\lambda_3^3\zeta & 2\lambda_3^3 n & 0 \end{bmatrix} \tag{4.10}$$

The previous result reappears in future steps as part of Λ_3, hence it is also split:

$$\begin{aligned} \Lambda_3 = \frac{1}{2\lambda_3}\frac{d\lambda_3}{d\varphi} &= \frac{1}{2\lambda_3}\left[\left(\frac{d\lambda_3}{d\varphi} \right)_n + 2\lambda_3^3 \left(R_{p,u}\zeta + T_{p,u} n \right) \right] \\ &= \frac{1}{2\lambda_3}\left(\frac{d\lambda_3}{d\varphi} \right)_n + \lambda_3^2 \left(R_{p,u}\zeta + T_{p,u} n \right) \\ &= \Lambda_{3,n} + \Lambda_{3,u} \end{aligned} \tag{4.11}$$

With $\Lambda_{3,u}$ found, all components necessary to repeat the process for the remaining differential equations are available. Thus, the equation governing the time element (2.63) is treated next by setting $R = R_n + R_u$ and $\Lambda_3 = \Lambda_{3,n} + \Lambda_{3,u}$:

$$\begin{aligned} \frac{d\lambda_{0,l}}{d\varphi} &= \lambda_3^{3/2}[1 + ((R_n + R_u)r - 2U)r + 2(\Lambda_{3,n} + \Lambda_{3,u})\zeta] \\ &= \lambda_3^{3/2}\left(1 + (R_n r - 2U)r + 2\Lambda_{3,n}\zeta\right) + \left(\lambda_3^{3/2} R_u r^2 + 2\lambda_3^{3/2}\zeta\Lambda_{3,u}\right) \\ &= \left(\frac{d\lambda_{0,l}}{d\varphi} \right)_n + \left(\frac{d\lambda_{0,l}}{d\varphi} \right)_u \end{aligned} \tag{4.12}$$

Now, expanding $\Lambda_{3,u}$ and factoring out the relevant components leads to another row of the B_2 matrix:

$$\left(\frac{d\lambda_{0,l}}{d\varphi}\right)_u = \left(\lambda_3^{3/2} r^2 + 2\lambda_3^{7/2}\zeta^2\right) R_u + \left(2\lambda_3^{7/2}\zeta n\right) T_{p,u} \tag{4.13}$$

$$\therefore B_{2,\lambda_{0,l}} = \left[\left(\lambda_3^{3/2} r^2 + 2\lambda_3^{7/2}\zeta^2\right) \quad 2\lambda_3^{7/2}\zeta n \quad 0\right] \tag{4.14}$$

Continuing with an analogous process for Equation 2.64,

$$\begin{aligned}\frac{d\lambda_1}{d\varphi} &= ((R_n + R_u)r - 2U)r\sin\varphi + (\Lambda_{3,n} + \Lambda_{3,u})[(1+\varrho)\cos\varphi - \lambda_1] \\ &= \big((R_n r - 2U)r\sin\varphi + \Lambda_{3,n}[(1+\varrho)\cos\varphi - \lambda_1]\big) + \big(R_u r^2\sin\varphi + \Lambda_{3,u}[(1+\varrho)\cos\varphi - \lambda_1]\big) \\ &= \left(\frac{d\lambda_1}{d\varphi}\right)_n + \left(\frac{d\lambda_1}{d\varphi}\right)_u\end{aligned} \tag{4.15}$$

Which results in the following after expansion of $\Lambda_{3,u}$:

$$\left(\frac{d\lambda_1}{d\varphi}\right)_u = \left(r^2\sin\varphi + \lambda_3^2\zeta[(1+\varrho)\cos\varphi - \lambda_1]\right) R_u + \left(\lambda_3^2 n[(1+\varrho)\cos\varphi - \lambda_1]\right) T_{p,u} \tag{4.16}$$

$$\therefore B_{2,\lambda_1} = \left[\left(r^2\sin\varphi + \lambda_3^2\zeta[(1+\varrho)\cos\varphi - \lambda_1]\right) \quad \left(\lambda_3^2 n[(1+\varrho)\cos\varphi - \lambda_1]\right) \quad 0\right] \tag{4.17}$$

Repeating the process again for $\frac{d\lambda_2}{d\varphi}$ (starting from Equation 2.65):

$$\begin{aligned}\frac{d\lambda_2}{d\varphi} &= (2U - (R_n + R_u)r)r\cos\varphi + (\Lambda_{3,n} + \Lambda_{3,u})[(1+\varrho)\sin\varphi - \lambda_2] \\ &= \big((2U - R_n r)r\cos\varphi + \Lambda_{3,n}[(1+\varrho)\sin\varphi - \lambda_2]\big) + \big(-R_u r^2\cos\varphi + \Lambda_{3,u}[(1+\varrho)\sin\varphi - \lambda_2]\big) \\ &= \left(\frac{d\lambda_2}{d\varphi}\right)_n + \left(\frac{d\lambda_2}{d\varphi}\right)_u\end{aligned} \tag{4.18}$$

Expanding the terms and grouping based on R_u and $T_{p,u}$ yields:

$$\left(\frac{d\lambda_2}{d\varphi}\right)_u = \left(\lambda_3^2\zeta\left[(1+\varrho)\sin\varphi - \lambda_2\right] - r^2\cos\varphi\right) R_u + \left(\lambda_3^2 n\left[(1+\varrho)\sin\varphi - \lambda_2\right]\right) T_{p,u} \tag{4.19}$$

$$\therefore B_{2,\lambda_2} = \left[\left(\lambda_3^2\zeta\left[(1+\varrho)\sin\varphi - \lambda_2\right] - r^2\cos\varphi\right) \quad \left(\lambda_3^2 n\left[(1+\varrho)\sin\varphi - \lambda_2\right]\right) \quad 0\right] \tag{4.20}$$

Finally, equation 2.67 provides the derivatives of λ_4 through λ_7 and contains the perturbation projection N which is split as before:

$$\frac{d}{d\varphi}\begin{bmatrix}\lambda_4\\ \lambda_5\\ \lambda_6\\ \lambda_7\end{bmatrix} = N_n\frac{r^2}{2n}\begin{bmatrix}\lambda_7 c_v - \lambda_6 s_v\\ \lambda_6 c_v + \lambda_7 s_v\\ -\lambda_5 c_v + \lambda_4 s_v\\ -\lambda_4 c_v - \lambda_5 s_v\end{bmatrix} + N_u\frac{r^2}{2n}\begin{bmatrix}\lambda_7 c_v - \lambda_6 s_v\\ \lambda_6 c_v + \lambda_7 s_v\\ -\lambda_5 c_v + \lambda_4 s_v\\ -\lambda_4 c_v - \lambda_5 s_v\end{bmatrix} + \frac{\omega_z}{2}\begin{bmatrix}\lambda_5\\ -\lambda_4\\ \lambda_7\\ -\lambda_6\end{bmatrix} \tag{4.21}$$

However, ω_z defined in Equation 4.22 also contains perturbing accelerations R and the previously found Λ_3 term:

$$\begin{aligned}\omega_z &= \frac{n-m}{\varrho} + \frac{(2U - Rr)\left(2-\varrho+m\right)r + \Lambda_3\zeta\left(\varrho - m\right)}{m\,(1+m)} \\ &= \frac{n-m}{\varrho} + \frac{(2U - (R_n + R_u)\,r)\left(2-\varrho+m\right)r + \left(\Lambda_{3,n} + \Lambda_{3,u}\right)\zeta\left(\varrho - m\right)}{m\,(1+m)}\end{aligned} \tag{4.22}$$

Expanding the terms, collecting those containing R_u as well as $\Lambda_{3,u}$ and simplifying the result leads to (where again, the first row contains the natural contribution and the second corresponds to the control acceleration):

$$\begin{aligned}\omega_z &= \frac{n-m}{\varrho} + \frac{(2U - R_n r)\left(2-\varrho+m\right)r + \Lambda_{3,n}\zeta\left(\varrho - m\right)}{m(1+m)} \\ &+ \frac{-r^2 m + r^2\left(\varrho-2\right)}{m(1+m)}R_u + \frac{-m\zeta+\varrho\zeta}{m(1+m)}\Lambda_{3,u} \\ &= \omega_{z,n} + \omega_{z,u}\end{aligned} \tag{4.23}$$

Combining this result with Equation 4.11 in Equation 4.21, the equation that can be used to obtain the final four rows of B_2 is found:

$$\frac{d}{d\varphi}\begin{bmatrix}\lambda_4\\ \lambda_5\\ \lambda_6\\ \lambda_7\end{bmatrix} = N_n\frac{r^2}{2n}\begin{bmatrix}\lambda_7 c_v - \lambda_6 s_v\\ \lambda_6 c_v + \lambda_7 s_v\\ -\lambda_5 c_v + \lambda_4 s_v\\ -\lambda_4 c_v - \lambda_5 s_v\end{bmatrix} + \frac{\omega_{z,n}}{2}\begin{bmatrix}\lambda_5\\ -\lambda_4\\ \lambda_7\\ -\lambda_6\end{bmatrix} + N_u\frac{r^2}{2n}\begin{bmatrix}\lambda_7 c_v - \lambda_6 s_v\\ \lambda_6 c_v + \lambda_7 s_v\\ -\lambda_5 c_v + \lambda_4 s_v\\ -\lambda_4 c_v - \lambda_5 s_v\end{bmatrix} + \frac{\omega_{z,u}}{2}\begin{bmatrix}\lambda_5\\ -\lambda_4\\ \lambda_7\\ -\lambda_6\end{bmatrix} \tag{4.24}$$

$$\therefore B_{2,\lambda_4-\lambda_7} = \begin{bmatrix}
\left(\frac{-r^2m+r^2(\varrho-2)}{m(1+m)} + \frac{-m\zeta+\varrho\zeta}{m(1+m)}\lambda_3^2\zeta\right)\frac{\lambda_5}{2} & \left(\frac{-m\zeta+\varrho\zeta}{m(1+m)}\lambda_3^2 n\right)\frac{\lambda_5}{2} & \frac{r^2}{2n}(\lambda_7 c_v - \lambda_6 s_v)\\
\left(\frac{-r^2m+r^2(\varrho-2)}{m(1+m)} + \frac{-m\zeta+\varrho\zeta}{m(1+m)}\lambda_3^2\zeta\right)\frac{-\lambda_4}{2} & \left(\frac{-m\zeta+\varrho\zeta}{m(1+m)}\lambda_3^2 n\right)\frac{-\lambda_4}{2} & \frac{r^2}{2n}(\lambda_6 c_v + \lambda_7 s_v)\\
\left(\frac{-r^2m+r^2(\varrho-2)}{m(1+m)} + \frac{-m\zeta+\varrho\zeta}{m(1+m)}\lambda_3^2\zeta\right)\frac{\lambda_7}{2} & \left(\frac{-m\zeta+\varrho\zeta}{m(1+m)}\lambda_3^2 n\right)\frac{\lambda_7}{2} & \frac{r^2}{2n}(-\lambda_5 c_v + \lambda_4 s_v)\\
\left(\frac{-r^2m+r^2(\varrho-2)}{m(1+m)} + \frac{-m\zeta+\varrho\zeta}{m(1+m)}\lambda_3^2\zeta\right)\frac{-\lambda_6}{2} & \left(\frac{-m\zeta+\varrho\zeta}{m(1+m)}\lambda_3^2 n\right)\frac{-\lambda_6}{2} & \frac{r^2}{2n}(-\lambda_4 c_v - \lambda_5 s_v)
\end{bmatrix} \tag{4.25}$$

Compiling the results, the B_2 matrix given in Equation 4.26 is obtained, resulting in a fully defined control input scaling matrix $B = B_2 \cdot B_1$.

$$B_2 = \begin{bmatrix}
\lambda_3^{3/2} r^2 + 2\lambda_3^{7/2}\zeta^2 & 2\lambda_3^{7/2}\zeta n & 0\\
r^2\sin\varphi + \lambda_3^2\zeta[(1+\varrho)\cos\varphi - \lambda_1] & \lambda_3^2 n[(1+\varrho)\cos\varphi - \lambda_1] & 0\\
\lambda_3^2\zeta\left[(1+\varrho)\sin\varphi - \lambda_2\right] - r^2\cos\varphi & \lambda_3^2 n\left[(1+\varrho)\sin\varphi - \lambda_2\right] & 0\\
2\lambda_3^3\zeta & 2\lambda_3^3 n & 0\\
\left(\frac{-r^2m+r^2(\varrho-2)}{m(1+m)} + \frac{-m\zeta+\varrho\zeta}{m(1+m)}\lambda_3^2\zeta\right)\frac{\lambda_5}{2} & \left(\frac{-m\zeta+\varrho\zeta}{m(1+m)}\lambda_3^2 n\right)\frac{\lambda_5}{2} & \frac{r^2}{2n}(\lambda_7 c_v - \lambda_6 s_v)\\
\left(\frac{-r^2m+r^2(\varrho-2)}{m(1+m)} + \frac{-m\zeta+\varrho\zeta}{m(1+m)}\lambda_3^2\zeta\right)\frac{-\lambda_4}{2} & \left(\frac{-m\zeta+\varrho\zeta}{m(1+m)}\lambda_3^2 n\right)\frac{-\lambda_4}{2} & \frac{r^2}{2n}(\lambda_6 c_v + \lambda_7 s_v)\\
\left(\frac{-r^2m+r^2(\varrho-2)}{m(1+m)} + \frac{-m\zeta+\varrho\zeta}{m(1+m)}\lambda_3^2\zeta\right)\frac{\lambda_7}{2} & \left(\frac{-m\zeta+\varrho\zeta}{m(1+m)}\lambda_3^2 n\right)\frac{\lambda_7}{2} & \frac{r^2}{2n}(-\lambda_5 c_v + \lambda_4 s_v)\\
\left(\frac{-r^2m+r^2(\varrho-2)}{m(1+m)} + \frac{-m\zeta+\varrho\zeta}{m(1+m)}\lambda_3^2\zeta\right)\frac{-\lambda_6}{2} & \left(\frac{-m\zeta+\varrho\zeta}{m(1+m)}\lambda_3^2 n\right)\frac{-\lambda_6}{2} & \frac{r^2}{2n}(-\lambda_4 c_v - \lambda_5 s_v)
\end{bmatrix} \tag{4.26}$$

4.3. Covariance Transformations

Finally, as the estimation process requires an initial state covariance to kickstart the process, its representation in terms of the elements used in the estimation is required. In theory, it is possible to select a suitable value by trial and error. However, should a conversion between a Cartesian representation P_x and the elements P_λ be developed, more representative comparisons between the A-OCBE, U-OCBE and R-OCBE, using equivalent initial state covariances may be performed. Furthermore, a covariance in terms of the regularized elements may be more difficult to interpret considering the variety of the parameters and their non-dimensional nature. As a result, this section treats two methods that were developed to convert a covariance matrix between a Cartesian and regularized representation as well as covariance remediation which is necessary to compensate for the effects of introducing additional dimensions to the covariance.

4.3.1. Monte Carlo based transformation

The first approach to converting between P_x and P_λ is based around the Monte Carlo method. This involves sampling n points, converting each point individually to the desired elements and finding the covariance using the equation given in Equation 4.27, where $\vec{y}_i$ represents the individual samples and $\bar{y}$ is the mean thereof.

$$P = \frac{1}{n-1}\sum_{i=0}^{n}\left(\vec{y}_i - \bar{y}\right)\cdot\left(\vec{y}_i - \bar{y}\right)^T \tag{4.27}$$

Such an approach has been applied in literature in context of other regularization methods such as the KS method and it was found to be quite reliable despite the computational intensity [35]. Due to its simplicity when it comes to implementation it was also tested for the regularization discussed in this thesis. In the case of a conversion $P_x \to P_\lambda$ it is a three step approach:

1. Sample n points $\vec{x}_i$ from a multivariate Gaussian distribution with covariance P_x around mean $\vec{x}$.
2. Convert each sample $\vec{x}_i$ to regularized elements $\vec{\lambda}_i = f(\vec{x}_i, t)$.
3. Compute P_λ using Equation 4.27.

However, the inverse transform $P_\lambda \to P_x$ is more cumbersome. As discussed in Section 4.1, the conversion from EDromo elements to Cartesian coordinates involves the independent variable φ, which needs to be corrected given a change in regularized state λ. Additionally, given random sampling, the quaternion magnitude is not preserved and must also be corrected prior to the conversion of a sample. Hence, the following approach is applied:

1. Sample n points $\vec{\lambda}_i$ from a multivariate Gaussian distribution with covariance P_λ around mean $\vec{\lambda}$.
2. Normalize quaternion elements of each sample.
3. Find suitable φ_i using method described in Section 4.1.
4. Convert each sample $\vec{\lambda}_i$ to Cartesian coordinates $\vec{x}_i = g\left(\vec{\lambda}_i, \varphi_i\right)$.
5. Compute P_x using Equation 4.27.

4.3.2. Linear Covariance Mapping

The second method implemented revolves around using Jacobian matrices of one variable with respect to the other as shown in Equation 4.28 for the forward transform and Equation 4.29 for the inverse.

$$P_\lambda = \left[\frac{d\vec{\lambda}}{d\vec{x}}\right] P_x \left[\frac{d\vec{\lambda}}{d\vec{x}}\right]^T \tag{4.28}$$

$$P_x = \left[\frac{d\vec{x}}{d\vec{\lambda}}\right] P_\lambda \left[\frac{d\vec{x}}{d\vec{\lambda}}\right]^T \tag{4.29}$$

While the two necessary Jacobian matrices may be found analytically, the conversion between the elements and Cartesian coordinates is cumbersome. A much faster to implement alternative is simply using a finite-difference approach. Considering that the covariance conversion is only necessary for initialization of the estimator and post-processing of results, it is not expected that this will introduce error significant enough to affect the comparison between the different OCBE variants. As a result, a central difference method was implemented to find both $\frac{d\vec{x}}{d\vec{\lambda}}$ and $\frac{d\vec{\lambda}}{d\vec{x}}$ which can then be used to obtain the corresponding covariances.

4.3.3. Conversion method comparison

Upon implementation both methods discussed in the previous subsections were tested by performing conversions $P_{x0} \rightarrow P_{\lambda} \rightarrow P_{x1}$ and comparing the values of P_{x0} and P_{x1}. In broad terms, it was found that both approaches work, with limited error, though some particular cases result in complications.

Figure 4.2 shows the absolute element-by-element error after the conversion process for a diagonal covariance given by Equation 4.30. In this example, the Monte Carlo approach used 10000 samples (number chosen by trial and error, until an increase in number of samples no longer resulted in a significant change in error). This results in the largest errors in a few of the off-diagonal entries, primarily in the upper left quadrant. The errors reach up to 2% of the value of the largest entries of P_{x0}, which may be argued to be acceptable. The linear mapping method however, results in negligible errors in all elements.

$$P_{x0} = \begin{bmatrix} 100 I_{3\times3}[\mathrm{m}^2] & 0_{3\times3} \\ 0_{3\times3} & 10^{-4} I_{3\times3}[(\mathrm{m/s})^2] \end{bmatrix} \tag{4.30}$$

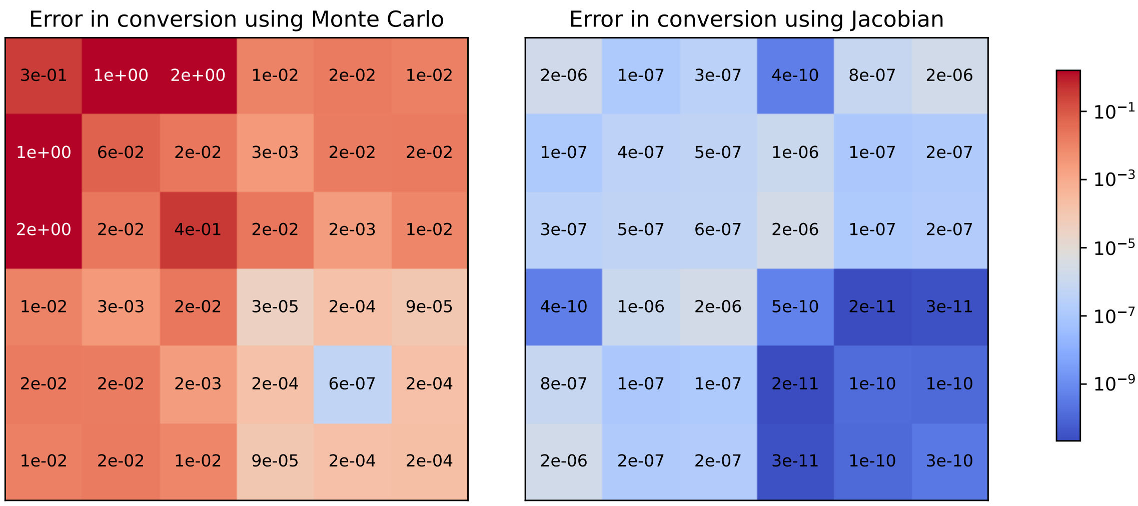


Figure 4.2: *Diagonal covariance conversion absolute error for both methods, given a mean in Tundra orbit (matching initial state specification in Table 5.1). Monte Carlo method using 10000 samples for both forward and backward conversion.*

This was also tested for more general, randomized matrices that are positive-definite and it was found that while the linear mapping approach retained it's accuracy, the Monte Carlo approach often resulted in multiple matrix elements diverging. An example of this is shown in Figure 4.3.

In this example, the original matrix P_{x0} is obtained from Equation 4.31, where A is a 6×6 matrix. In order to ensure P_{x0} is positive semi-definite, each entry of the A matrix is randomly sampled from a uniform distribution between 0 and 1. For this particular test, after the matrix product in Equation 4.31 this results in entry values for P_{x0} between 1 and 3. While the linear mapping approach also showcases some sensitivity in this case, as the errors related to the y position increase to around 1% of the covariance values, Monte Carlo results in errors multiple orders of magnitude above the original values of the matrix elements.

$$P_{x0} = AA^T \tag{4.31}$$

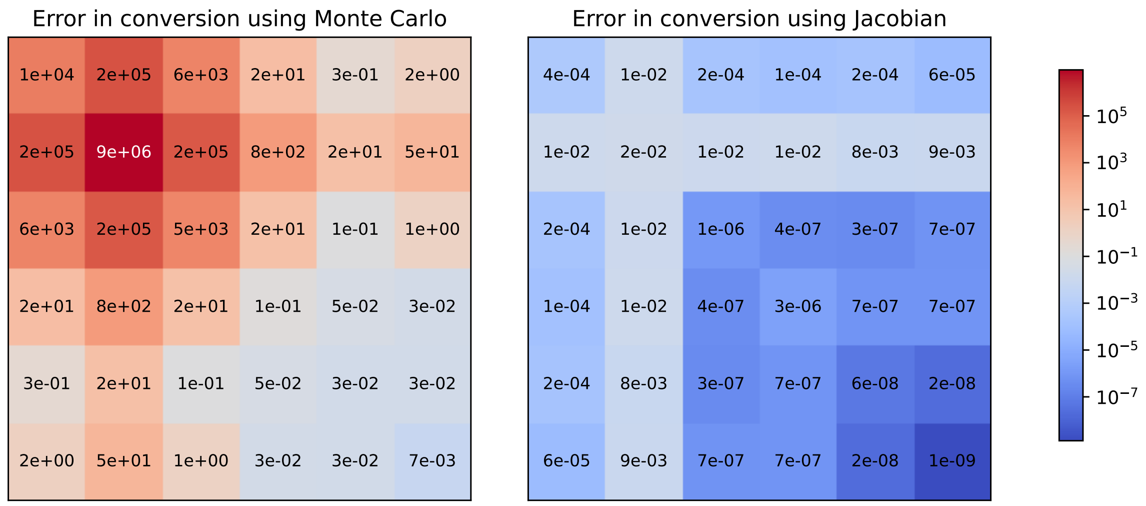


Figure 4.3: *Randomized positive semi-definite covariance conversion absolute error for both methods, given a mean in geostationary orbit (matching initial state specification in Table 5.1). Monte Carlo method using 10000 samples for both forward and backward conversion.*

This effect was particularly prominent when applying the conversion on a mean state with low (< 0.1) eccentricity. It was identified that the differences between the two methods in such conditions are already present in P_λ, particularly the quaternion components. As a result, this effect may be attributed to the definition of the quaternion and independent variable in the regularized elements. The quaternion describes an orientation of an intermediate frame, the definition of which is dependent on the true (or rather eccentric) anomaly which is in turn defined relative to an argument of periapsis. If the eccentricity of a mean state used in the conversion is zero, each sample obtained using Monte Carlo (with variations in position and velocity) introduces a different definition of the argument of periapsis. For some samples, (i.e. if only the radial position decreases w.r.t. the mean value) the spacecraft will be considered at or near the periapsis. Though if the radial position were to increase due to the random sample, the spacecraft is defined as near the apoapsis, causing significant variation in θ, which results in vastly different definitions of regularized elements and thus an inappropriate covariance conversion. As a result of these findings, in addition to the more computationally expensive nature of the method, it was deemed that the Monte Carlo approach is not appropriate for use in the R-OCBE.

On the other hand in the case of the method described in Subsection 4.3.2, only one case where the method failed was identified. This method also showcases sensitivity to low eccentricity values, though the ability to select the steps in the finite difference method allows for the issue to be avoided as long as eccentricity is not perfectly 0. However, even for values such as $e = 0.01$, it was found that for two points on the orbit ($\theta = 90°$ or $\theta = 270°$), given $\omega = 0°$, the solution to Equation 4.4 does not exist[1]. While this is not an issue for numerical propagation, when applying a finite difference approach to obtain the necessary Jacobian terms the method described in Section 4.1 fails to converge to a solution for φ. Thus, the conversion $P_\lambda \rightarrow P_{x1}$ results in a gross overestimation (or potentially negative values) of certain terms of the covariance (often those related to the x position). As this is a rather specific point of failure, that only appears in the conversion which is applied after the state estimation is complete (in the post-processing), this approach was selected for use in the R-OCBE with the knowledge that at particular points the Cartesian covariance may be significantly overestimated.

[1] For other values of argument of periapsis, the relationship is less clear, though the points do not vanish.

4.3.4. Covariance remediation

It should be noted that regardless of the conversion approach, the covariance is mapped from a 6×6 matrix to an 8×8. As the regularized elements contain a quaternion and two projections of the generalized eccentricity, they are not entirely independent of each other. This implies a covariance matrix in terms of the EDromo elements is rank 6 despite its shape. This is similar to the findings by Ayuso[31], though that discussion treats a covariance in terms of traditional Dromo elements rather than the modified EDromo which is the focus of this thesis. This aspect must be treated as it implies the matrix shall contain two eigenvalues of 0 and is in turn positive semi-definite rather than positive definite as assumed in the development of the OCBE. Furthermore, it violates the requirement for the Cholesky decomposition which is used in the Unscented OCBE approach introduced in Section 2.1, thus covariance remediation is a necessary step in the R-OCBE.

The approach of remediation by clipping eigenvalues is discussed in [50] and revolves around replacing the zero or negative eigenvalues of a given matrix with very small positive values to ensure the matrix is positive definite. In order to minimize the effect on the meaning of the matrix P, the first step of the process is to perform an eigen-decomposition:

$$P = V \Lambda V^T \tag{4.32}$$

Where V contains the eigenvectors of P and all eigenvalues are contained in the diagonal matrix Λ. If any of the elements of Λ are non-positive, they may be replaced with an appropriately chosen small positive value, yielding an updated diagonal matrix Γ. The remediated, positive-definite covariance matrix can then be found using Equation 4.33.

$$P_{rem} = V \Gamma V^T \tag{4.33}$$

Given that the covariance matrix is expected to be positive semi-definite and only eigenvalues of 0 must be replaced, the impact of remediation on the estimation process should be negligible while allowing the use of methods such as Cholesky decomposition.

5

Test Cases

As discussed in Section 3.1, the first step necessary to obtain numerical results is to define the test cases for which the performance of the R-OCBE shall be evaluated. Based on the findings of the literature survey, three distinct satellite orbits were selected. This chapter summarizes the tested orbits, dynamics models as well as the spacecraft itself in sections 5.1, 5.2 and 5.3 respectively.

5.1. Spacecraft Initial States

As mentioned in the chapter introduction, in order to gain insight on whether orbital parameters influence the performance of the R-OCBE three distinct initial states are used for the propagations. As the focus of thesis remains on the perturbed two-body problem and particularly on the GEO region, the initial states were selected to be representative of common cases:

1. A general satellite in geostationary orbit
2. An uncontrolled satellite in inclined graveyard orbit [51]
3. A heavily inclined and eccentric alternative disposal orbit - a tundra orbit [8]

The initial state parameters of each of these cases are summarized in Table 5.1.

Table 5.1: *Initial state of three selected test cases*

Test Case	Geostationary Orbit	Graveyard Orbit	Tundra Orbit
Initial Time [-]	J2000	J2000	J2000
Semi-Major Axis [km]	42164	42564	42164
Eccentricity [-]	0.01	0.01	0.3
Inclination [deg]	0	15	63.4
Argument of Periapsis [deg]	0	0	270
RAAN [deg]	0	0	0
True Anomaly [deg]	0	0	0

5.2. Dynamics Model

A key aspect of all astrodynamics problems that must be determined is, of course, the environment and the corresponding dynamics model. In order to compare the performance of estimation methods, two dynamics models must be defined. The first represents the true (or reference) solution and is used to obtain measurements as well as compute the final errors of the estimations. The second, represents the known dynamics and includes slight error in modelling. This model is a direct part of the OCBE variants.

As discussed in Section 3.1, the goal of the thesis is to determine if the R-OCBE can provide benefits to state estimation of spacecraft, in particular with low observation frequency. The work done in this thesis does not correspond to a specific spacecraft, instead it is focused on hypothetical, representative cases. As a result of this, one may argue that an extreme fidelity dynamics model, accurate to real life is not necessary, as long as the reference solution still captures the driving effects and provides sufficient detail to distinguish estimation methods. Hence, primarily to reduce runtimes, a lower fidelity dynamics model was used to compare the R-OCBE performance with the A-OCBE and U-OCBE.

While a detailed selection of the true dynamics model is presented in Chapter 6, the results are also provided here in order to summarize all parameters necessary for reproduction of results in one chapter. Hence, the 'true' solution of the trajectories are obtained with the following:

1. The Earth's gravity is modelled using spherical harmonic gravity, of degree 7 and order 7
2. The perturbing bodies (the Sun and the Moon) are modelled as point masses
3. The solar pressure is modelled using the panelled method discussed in Section 2.3
4. All other perturbations are neglected

When it comes to the known dynamics model, sources of error must be introduced relative to the reference model. Similar to the true model, the impact of simplifications of the known dynamics model is discussed in Chapter 6. The primary simplification present in all test cases, is the use of cannonball solar radiation pressure instead of the panelled model. Additionally, the impact of fewer spherical harmonic gravity terms is introduced for particular tests, where explicitly specified.

5.3. Spacecraft Parameters

The final aspect of the environment yet to be covered is the spacecraft itself. As discussed in the previous section, the only forces it is subject to are gravity and solar radiation pressure. Thus, only parameters used for these models are necessary to define the spacecraft. Table 5.2 summarizes the list of values used for all dynamics models. For the panelled body SRP model, the spacecraft is modelled as rectangular box, with fixed pointing towards the earth and rotating solar panels pointing towards the sun. In order to obtain representative parameters for a satellite in the GEO region, the GOES-R series datasheet was used [52].

It should be noted that the spacecraft mass was selected to be equal to that of a GOES-R series spacecraft, after approximately 5 years of operation. Additionally, in the case of cannonball SRP, the reference area was calculated such that it results in the same mean acceleration as the panelled model in a reference simulation (assuming the reflectivity coefficient is constant). This should result in a small yet noticeable error that is further discussed in Chapter 6.

Table 5.2: *Summary of defined spacecraft parameters*

Dynamics Model	Parameter	Value
Gravity	Mass [kg]	3547.9
Cannonball SRP	Reference Area [m^2]	74.598
	Reflectivity Coefficient [-]	1.3
Panelled SRP	Body Specular Reflectivity [-]	0.32
	Body Diffuse Reflectivity [-]	0.18
	Array Specular Reflectivity [-]	0.05
	Array Diffuse Reflectivity [-]	0.05
	Length [m]	5.6
	Width [m]	3.9
	Height [m]	6.1
	Array Area [m^2]	26.617

6

Dynamics Model Analysis

This chapter discusses the approach to select numerical integration methods and environment fidelity for both the 'true' (reference solver) and the 'known' (primary solver) solutions. Section 6.1 discusses the accuracy requirement for the numerical solution, as well as the selection of the reference fixed-step integration algorithm, used for the following environment selection. This is followed by Section 6.2, where the environment model for the true solutions is selected. Next, Section 6.3 briefly covers the effect of the simplified dynamics in the primary dynamics solver and the selection of the variable step integrator to be used in the OCBE algorithm. Finally, the numerical performance of the regularized formulation originally introduced in Section 2.2 is discussed in Section 6.4.

6.1. Benchmark Integration Error

When applying the OCBE, the primary goal is to evaluate each of the variant's capabilities at estimating the state of a spacecraft with measurement noise and dynamic mis-modelling. However, due to the nature of the perturbed two-body problem and the lack of analytic solution, numerical solution methods are necessary. These introduce additional sources of error, namely truncation and floating point errors. Generally, the exact values of these errors are unknown. To ensure the later found differences in performance can actually be attributed to the variations in the OCBE algorithm, it is important to quantify the impact of both of these error sources and ensure they are sufficiently small. In addition to ensuring accuracy of the algorithm itself, 'true' solutions must be defined for error calculations. It is also important to evaluate the extent to which these solutions can actually be considered true.

Thus, the first step of this process is to define a numerical accuracy requirement. For the purposes of comparison of the OCBE variants, the smallest expected standard deviation of the simulated measurements is in the order of 10 meters. To ensure the numerical propagation error does not eclipse the measurement uncertainty (with some margin) the requirement for numerical integration accuracy is set to be two orders of magnitude lower: $\mathcal{O}(0.1[\text{m}])$. While not necessary for measurement sampling, a velocity standard deviation is also used when setting an initial state covariance in the estimation process. This value was chosen two orders of magnitude below the position (0.1 meters per second), which could then be used to define a numerical accuracy requirement in terms of velocity to be $\mathcal{O}(0.001[\text{m/s}])$.

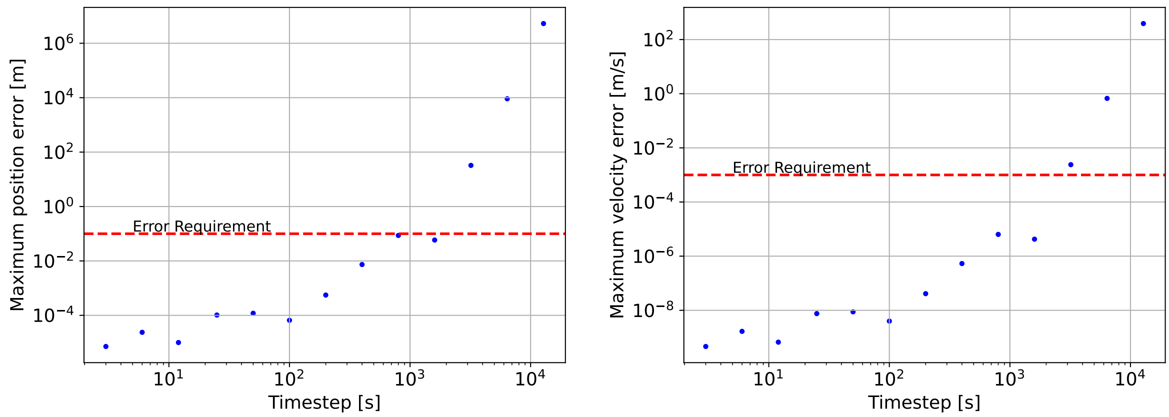
Note that the impact of environmental mis-modelling is not taken into account when setting this requirement. This is primarily due to the fact that the mis-modelling can easily be tuned to achieve a desired effect. For example, the spacecraft mass or effective area for cannonball solar pressure can intentionally be set incorrectly.

With the accuracy requirement set, a reference propagation must be performed as a starting point estimation of the error. To achieve this, for each of the test cases discussed in Chapter 5, two fixed-step propagations (for a set timestep) are performed using 7th and 8th order accurate integration schemes. When the truncation error is dominant, the difference between the 7th and 8th order propagation is a good representation of the overall error. The change in this error can also be predicted for a change in timestep. For a 7th order scheme, increasing the timestep by factor 2 is expected to increase the global truncation error by approximately factor 128, or more broadly speaking two orders of magnitude. If this trend is not observed, it is likely other effects are influencing the error of the integration (such as the floating point error). Using this information, a number of propagations with varying time-step were performed.

For the sake of completion, before discussing the results, the environment used for the initial benchmark generations contains the following:

- Spherical harmonic gravity of the Earth (up to degree and order 30)
- Point mass gravity of the Moon
- Point mass gravity of the Sun
- Panelled spacecraft SRP

The results of the benchmark error approximations are shown in Figure 6.1 for the Geostationary orbit, for a week long propagation (while the other orbit cases follow later in Figures 6.3 and 6.5). The first thing to notice is that in order to satisfy the error requirement set earlier, timesteps below 1000 [s] are necessary. However, while the trend of error increasing by 2 orders of magnitude for double the timestep was expected, this relationship only stands in the region violating the error requirement.



(a) *Position Error* **(b)** *Velocity Error*

Figure 6.1: *Geostationary Orbit test case integration error after one week. Each consecutive data point represents a factor 2 increase in timestep.*

To gain further insight into the issue, the error development over time was plotted for a number of the propagations, focusing on position error as the velocity displayed identical behaviour. For example, in the GEO case in Figure 6.2, the expected error scaling trend can be observed for large timesteps such as 3200 [s] and 6400 [s]. On the other hand, the effect of discrete floating point error is noticeable for a very small timestep such as 6 seconds. The region inbetween seems less well-behaved. From Figure 6.2, it is very clear that for dt=1600 [s] the error follows a very different, somewhat oscillatory trend to other timesteps. Additionally, dt = 800 [s] produces an error $\mathcal{O}(0.1\,[\mathrm{m}])$ while it was expected to produce $\mathcal{O}(10^{-3}\,[\mathrm{m}])$. As the precise reasoning for this behaviour was not determined, it is difficult to guarantee future propagations will deliver the same accuracy and a margin should be employed. Thus, an even smaller step (dt = 400 seconds) was chosen as the benchmark integration for the geostationary orbit test case.

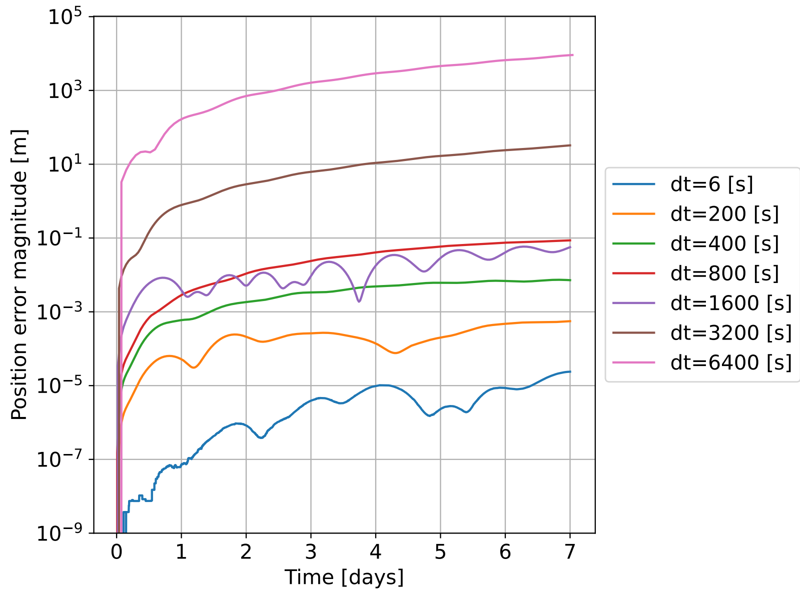
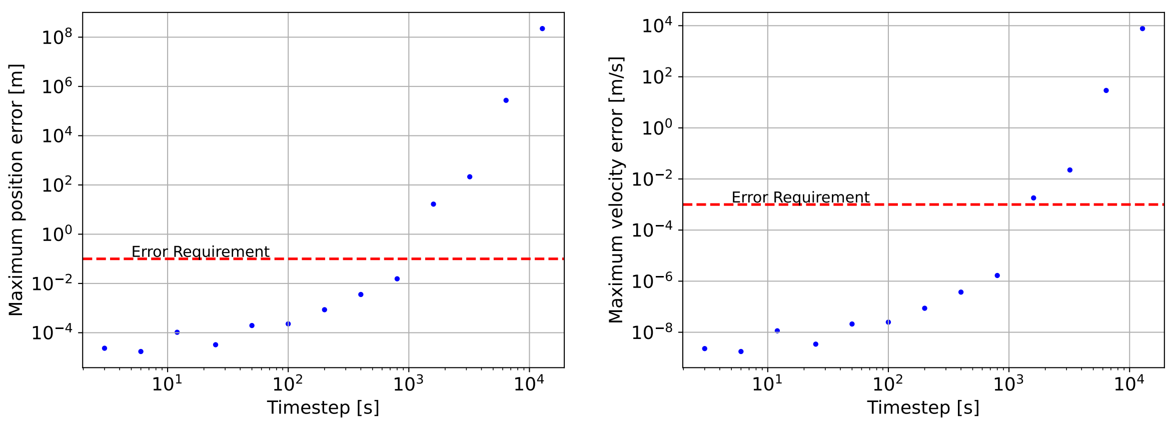


Figure 6.2: *Position Error Magnitude over duration of propagation for the GEO test case*

Similar to the Geostationary test case, from Figure 6.3 it can be observed that dt = 800 [s] would satisfy the accuracy requirements for the Tundra orbit as well. However, upon more detailed inspection of Figure 6.4, a slight dip in error before the first day of the propagation is observed, and only present for this timestep. In order to ensure that given different environment conditions this decrease in error does not disappear and result in a violation of the requirement, the dt = 400 [s] solution is used as the benchmark. This solution presents some slightly erratic error behaviour near the end of the seven-day propagation. However, considering the two order of magnitude margin to the requirement, this was not deemed an issue.



(a) *Position Error* **(b)** *Velocity Error*

Figure 6.3: *Tundra Orbit test case integration error after one week. Each consecutive data point represents a factor 2 increase in timestep.*

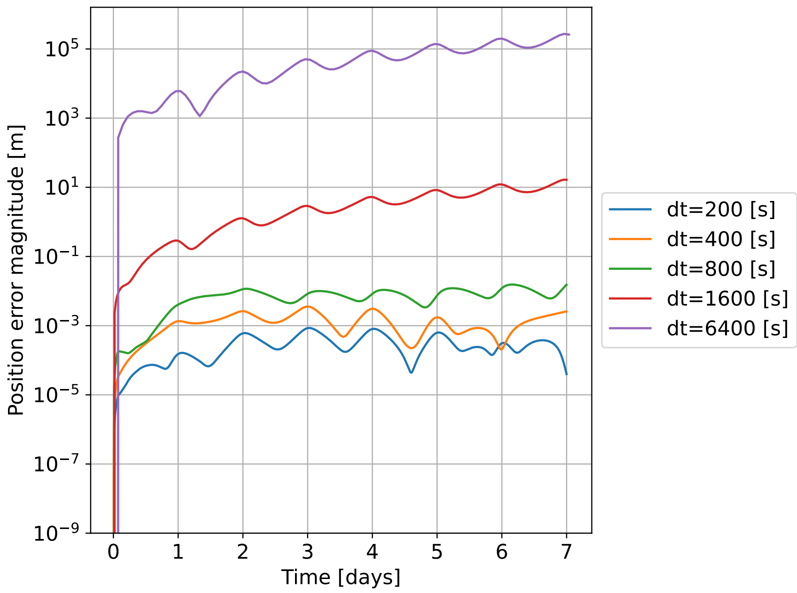
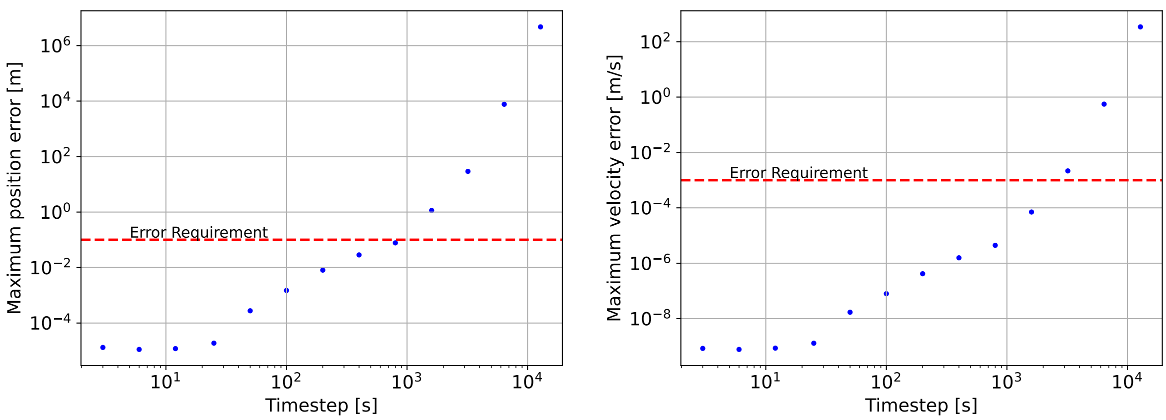


Figure 6.4: *Position Error Magnitude over duration of propagation for the Tundra test case*

Finally, for the Graveyard orbit the situation is a bit different. Figure 6.5 indicates that timesteps of 200, 400 and 800 seconds all satisfy the error requirement, but are extremely close together. Once again, the error behaviour over time is plotted in Figure 6.6 to gain additional insight before selecting the benchmark solution. Once again, large timesteps display steadily growing error separated by around two orders of magnitude as expected. However, the three propagations of interest all display error oscillations of at least one order of magnitude. Additionally, both the 400 and 800 second step solutions reach a maximum error just before the third day of the propagation. Considering the global error is generally expected to grow over time, this may be a reason to doubt the reliability of the maximum error metric in this particular case.



(a) *Position Error* **(b)** *Velocity Error*

Figure 6.5: *Graveyard Orbit test case integration error after one week. Each consecutive data point represents a factor 2 increase in timestep.*

On the other hand, while the dt = 200 seconds solution still displays the error decreasing at times, this decrease is periodic and much more consistent, displaying minima exactly one day apart. This likely implies an interaction between the truncation error and perturbations acting on the inclined orbit. While the behaviour is still not desired, at the very least it remains more predictable and consistently below the requirement limit. Furthermore, the maximum error is still observed at the end of the propagation and its mean is growing steadily in a shape similar to those observed for timesteps the likes of dt=3200 [s]. Resorting to lower timestep solutions was also considered, but the truly random influence of floating point error became more and more noticeable, without resulting in steadier error behaviour for the propagation. Considering that, the dt = 200 [s] solution was chosen as the most suitable benchmark to ensure $\mathcal{O}(0.1\,[\mathrm{m}])$ and $\mathcal{O}(0.001\,[\mathrm{m/s}])$ accuracy for the Graveyard orbit.

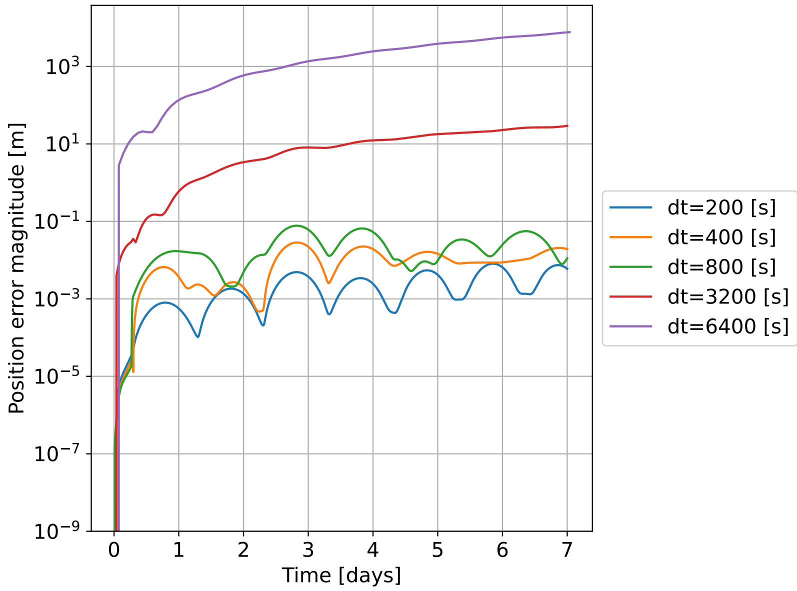


Figure 6.6: *Position Error Magnitude over duration of propagation for the Graveyard test case*

6.2. True Environment Selection

With benchmark solutions for each test case obtained, some tweaks to the environment model could be made. As discussed in Section 5.2, an extreme fidelity dynamics model was not deemed necessary for this application. Instead, the focus lies on including the effects on the order of magnitude of measurement and dynamics mis-modelling errors. In simpler terms, if the removal of a perturbation results in a change of state more than: $\mathcal{O}(0.1\,[\mathrm{m}])$ and $\mathcal{O}(0.001\,[\mathrm{m/s}])$, the perturbation should be included in the true solution of the dynamics problem. This requirement was chosen to coincide with the requirement set on the numerical integration precision, a few reasons for this may be highlighted:

1. The numerical accuracy requirement is set for a propagation of 1 week, whereas the measurement gaps in the testing of the OCBE are below a day, hence a margin is anticipated.
2. As discussed in Section 6.1, due to the behaviour of the integration error, the time steps selected resulted in error 1 order of magnitude below the requirement, providing additional margin in which the truncation error remains distinct from the modelling error.
3. In the OCBE implementation a further simplified model is employed in order to test the capabilities of the algorithms capability to compensate for dynamics mis-modelling. As long as this additional mis-modelling is at least one order of magnitude above the error requirement, the distinction between integrator truncation and modelling error in the truth solution shall remain of little consequence.

In any case, terms with a smaller effect than the requirement may be omitted as they increase computational time while their contributions would remain inconsequential due to the numerical integration error. In this section, only the selection process for the GEO test case is presented. This is simply due to other test cases leading to identical conclusions.

As a first step in picking the environment model, the magnitude of some accelerations acting on the spacecraft was plotted in Figure 6.7. This provides a general overview of how impactful the different perturbations are relative to each other. As expected, the point mass contributions of the sun and the moon are significant, followed closely by solar radiation pressure and the largest terms of the Earth's spherical harmonic gravity. All of these terms are expected to have a noticeable impact on the solution. The same can not be said about some higher order spherical harmonic terms, which result in average acceleration magnitudes as low as $\mathcal{O}(10^{-32}[\mathrm{m/s^2}])$.

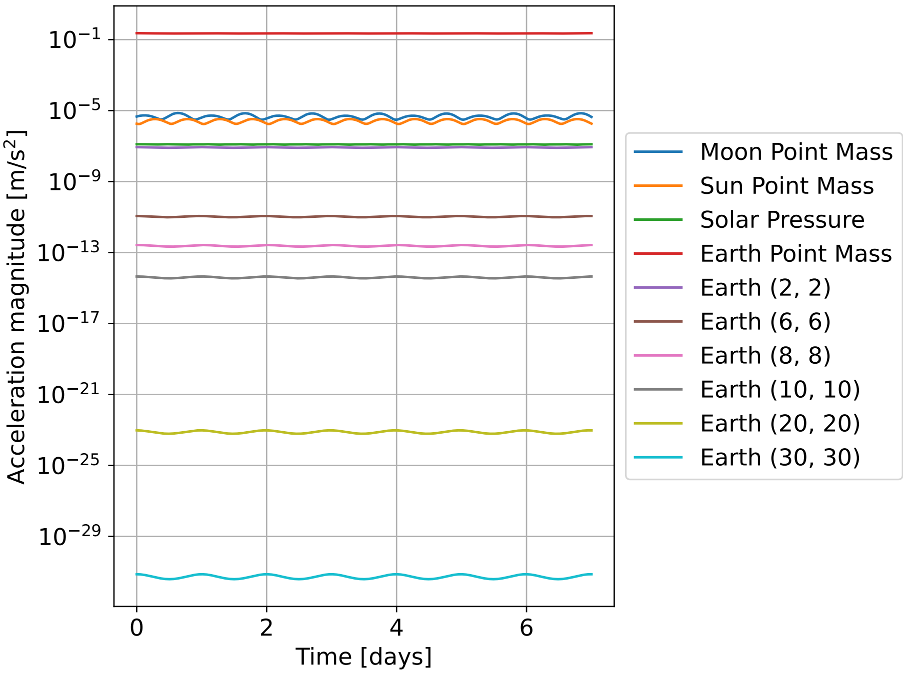
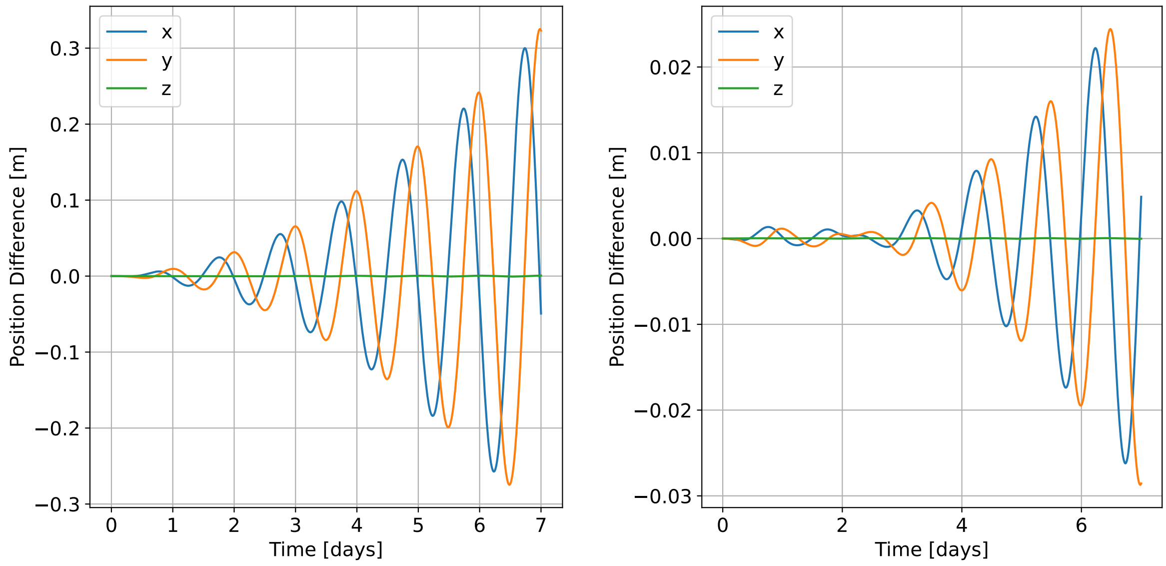


Figure 6.7: *Geostationary orbit acceleration magnitudes acting on spacecraft. Brackets indicating acceleration resulting from Spherical Harmonic term of (degree, order)*

In order to determine how many spherical harmonic terms may be omitted from the final environment model, a variety of propagations were performed. With each propagation, the number of spherical harmonic gravity terms was reduced. These results are then compared to the benchmark solutions defined in Section 6.1 by computing the state differences over time.

The position requirement was found to be the limiting one, hence the corresponding results are presented in Figure 6.8. As can be seen in Figure 6.8a, reducing the spherical harmonic terms to degree and order 6 results in a position error of over 0.1m in less than four days, violating the requirement. On the other hand, Figure 6.8b shows that the difference in position w.r.t. the reference solution remains well below the requirement even after a week of propagation. This implies degree and order 7 results in a suitable environment for the geostationary case.

For the Tundra and Graveyard orbits, the results remain extremely similar and are thus omitted from



(a) *Degree and order 6 spherical harmonic model* **(b)** *Degree and order 7 spherical harmonic model*

Figure 6.8: *Alternate Earth gravity model inertial position difference w.r.t benchmark solution*

this report. The core difference is that the state difference in the z component is comparable in magnitude to the x and y. This behaviour was expected considering the out of plane nature of both the orbits. Regardless, it does not result in a violation of the accuracy requirements when using the same 7, 7 gravity model.

The same procedure may be repeated for other perturbations such as the solar radiation pressure or point mass gravity contributions of the Sun and the Moon. However, from Figure 6.7 it can be seen that each of these accelerations is multiple orders of magnitude larger than the contribution of degree and order 6 spherical harmonic term, which is necessary to meet the requirements. It is thus easy to conclude that these perturbations will also be necessary.

Based on the previous analysis, the 'true' or 'reference' solutions for each of the test cases are generated with the following final choice of environment and integrator:

- Spherical harmonic gravity of the Earth (up to degree and order 7)
- Point mass gravity of the Moon
- Point mass gravity of the Sun
- Panelled spacecraft SRP
- 7th order accurate, fixed step RK (Runge-Kutta) integration scheme
- dt = 400 [s] for the GEO and Tundra test cases
- dt = 200 [s] for the Graveyard test case

With the aforementioned choices, after a propagation of one week, obtained solutions are considered to have state errors no larger than: $\mathcal{O}(0.1\,[\mathrm{m}])$ and $\mathcal{O}(0.001\,[\mathrm{m/s}])$ in inertial position and velocity respectively. Furthermore, as the numerical reference solution is obtained with fixed steps of 200 or 400 seconds, in order to facilitate OCBE testing on measurements at different observation gaps, the solution may be interpolated using a Lagrange 7th order interpolation [49].

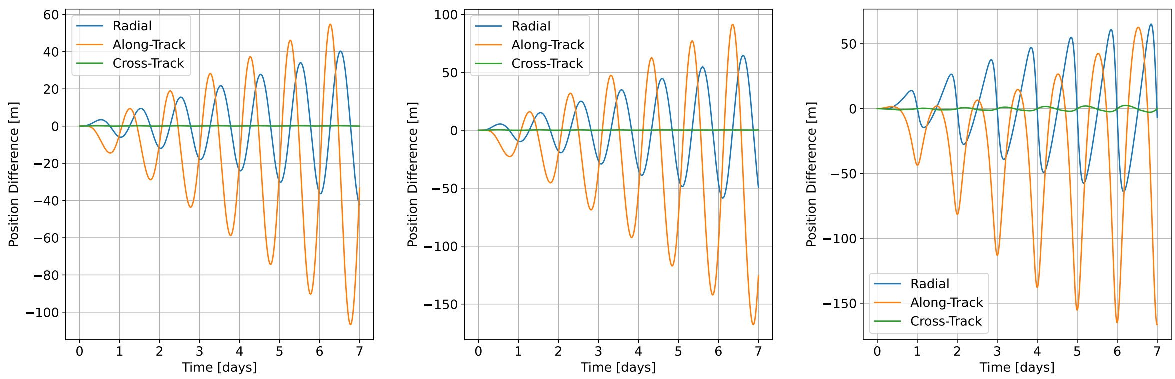
6.3. Known Dynamics Model

For the OCBE testing, it is desired to simulate imperfect knowledge of the environment and introduce some mis-modelling. One of the ways this can be done is by use of a simplified SRP model for the 'known' dynamics model.

The first step in setting up the simplified cannonball model, is choosing the parameters such that they provide similar results to the panelled specular diffuse model used for the true solution. Given the completed propagations, the mean acceleration due to the panelled specular diffuse model over the propagation can be computed. This can be used to obtain a suitable approximation of the reference area, by assuming a reflectivity coefficient of 1.3 and rewriting Equation 2.81 as follows:

$$A = \frac{\int_{t_0}^{t_f} |\vec{a}_{SRP}|}{t_f - t_0} \frac{mc}{\Phi C_r} \tag{6.1}$$

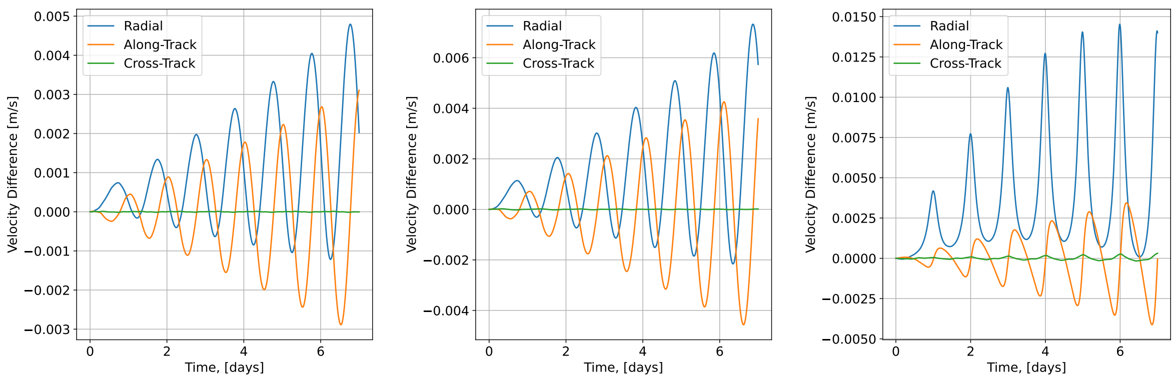
For the Geostationary Orbit, this results in $A = 74.598[\mathrm{m}^2]$. Considering how similar the results are for the other orbits ($72.006[\mathrm{m}^2]$ and $74.094[\mathrm{m}^2]$), the Geostationary value is used for all cases for the sake of simplicity. The next step is to determine the effect of such a simplification. The position difference with respect to the true solution in RSW coordinates is given in Figure 6.9. It can be seen that for all test cases, within half a day the position difference reaches the order of magnitude of the minimum standard deviation discussed in Section 6.1. This implies the mis-modelling is sufficiently significant to be detected by the OCBE, though it is not orders of magnitude larger than the measurement error, hence it is not expected to dominate the problem.



(a) *Geostationary Orbit* **(b)** *Graveyard Orbit* **(c)** *Tundra Orbit*

Figure 6.9: *RSW position difference resulting from use of cannonball solar pressure for each test case*

The impact on the velocity error was also quantified, which is shown in Figure 6.10. In can be observed that in Tundra conditions, error variation approximately factor 2 larger in the radial component of the velocity, over each orbital period is present. This can likely be attributed the eccentricity of the test case. Regardless, for all test cases the differences in state still exceed the guaranteed numerical precision (given a week-long error accumulation), hence for all test cases the SRP mis-modelling effects should remain identifiable in the OCBE estimation process and remain largely unaffected by numerical integration errors.



(a) *Geostationary Orbit* **(b)** *Graveyard Orbit* **(c)** *Tundra Orbit*

Figure 6.10: *RSW velocity difference resulting from use of cannonball solar pressure for each test case*

While the discussion above is based on results using the reference dynamics solver implemented using the Tudat library, as was discussed in Chapter 3, a custom 'Primary' dynamics solver was implemented for use in the OCBE. With the known environment selected, this solver was implemented. The validation of the implemented methods can be found in Appendix A.

With the primary dynamics solver implemented, a variable step integrator that maintains the error within specification was selected in order to improve performance of the estimation methods. It was found earlier that the introduction of the cannonball SRP model results in a modelling position error in $\mathcal{O}(100[\text{m}])$, while a numerical integration error (primarily truncation) of $\mathcal{O}(0.1[\text{m}])$ was expected. This is a very significant difference which implies the requirement for integration accuracy of the known dynamics model may be relaxed w.r.t. that of the true dynamics, without a significant effect on the difference between the two. To ensure this difference between models remains unaffected, the requirement was relaxed to two orders of magnitude below the modelling error introduced: $\mathcal{O}(1[\text{m}])$. However, as it was found that the velocity error due to the SRP model simplification is only one order of magnitude above the original requirement, it is not relaxed as it would in turn noticeably affect the difference between the dynamics models.

To perform the selection of appropriate tolerances, a number of variable step integrators were tested with the known environment model, comparing the obtained states to the previously found fixed step integration results. The final choice was made such that the necessary state derivative function evaluations are minimized while adhering to the updated error requirements: $\mathcal{O}(1[\text{m}])$, $\mathcal{O}(0.001[\text{m/s}])$. The only further constraint in this selection process was that only the RKF7(8) integration scheme was considered, as this is the only scheme fully implemented and tested as part of the primary dynamics solver.

As the process is rather straight forward and the behaviour of the Cowell propagator is generally well known, only the final selection of tolerances is discussed in this subsection. For all three of the orbit test cases, an identical absolute tolerance was chosen at a value of: $\varepsilon_{abs} = 10^{-10}$. The choice of relative tolerance varied, however. To achieve the requirements for the Geostationary case, $\varepsilon_{rel} = 10^{-12}$ was selected. This choice reduced the necessary function evaluations to 27.3% of the 400 [s] fixed step integration. For the Tundra case, $\varepsilon_{rel} = 10^{-12}$ violated the position accuracy requirement on the final day of the propagation by a small margin, hence the tolerance was reduced to: $\varepsilon_{rel} = 10^{-13}$. This results in a small loss in performance compared to the GEO case as 34.9% of the fixed-step function evaluations are now necessary to obtain the solution.

Finally, for the Graveyard case, a selection of relative tolerance of $\varepsilon_{rel} = 10^{-12}$ was made to satisfy the error requirements. While the value of the tolerance selected is the same as the GEO case, the performance gain relative to the fixed step solution is not. Only 13.45% of the function evaluations of the fixed-step propagation were necessary to meet the requirements. While at an initial glance such a major difference in performance may be surprising, it is explained by the fact that the reference fixed step solution employed a 200 [s] step rather than 400 [s] as in the other test cases. Indeed, it was found that the time steps of the variable step integrator for the graveyard orbit were comparable to that of the other solutions.

With the integrator selection for each test case completed and the impact of solar pressure mismodelling quantified, the known dynamics model is fully defined and error predictions are available. From this point onwards, in the analysis of the A-OCBE and U-OCBE, environment models used are the ones discussed in this section unless specifically stated otherwise.

6.4. EDromo formulation performance

For the R-OCBE, the regularized EDromo propagator was implemented and in order to select a set of suitable tolerances, a similar approach was applied as for the Cowell propagator. Though, considering the behaviour of this propagation scheme is less well known, the process is discussed in additional detail in this subsection. As the first step, for the known dynamics model, a set of GEO solutions with varying absolute and relative tolerance were obtained, with the necessary state derivative evaluations for all tested combinations displayed in Figure 6.11.

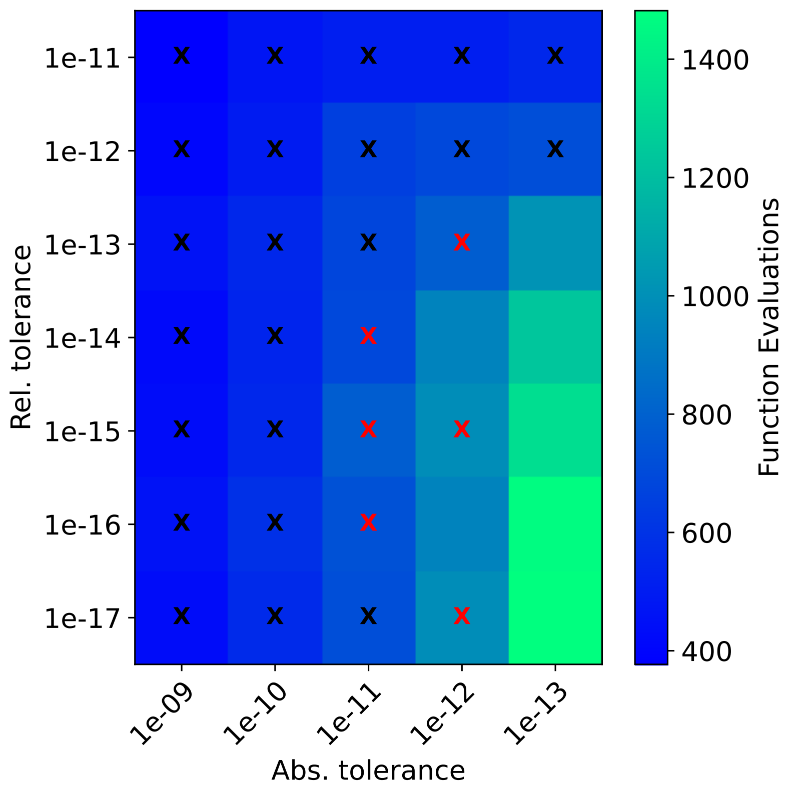


Figure 6.11: *Total state derivative function evaluations for propagated solutions with varying tolerances for the geostationary orbit. Red X indicates violation of position error requirement, while black X indicates violation of both error requirements.*

From Figure 6.11, it may be observed that the position error requirement is violated before that of the velocity in all cases (as indicated by the lack of 'x' corresponding to a violation of only the velocity error requirement). This implies the approach of relaxing the position requirement for the known dynamics model is most certainly resulting in a gain in performance of the propagator.

Regardless, it was found that for this particular case, the number of necessary function evaluations to obtain a solution satisfying the requirements depends substantially on the absolute tolerance. In fact, for absolute tolerance equal to or lower than $\varepsilon_{abs} = 10^{-14}$, it was found that the numerical integration may at times effectively get 'stuck' in taking steps smaller than 1 second (when converted from the φ independent variable). This can be explained considering the case when the solution error requirement resulting from the tolerances, is lower than the machine error. As with any integration, the integrator attempts to reduce the integration step to satisfy the error requirement. However, in this case, the state error is now significantly influenced by the random machine error, thus the step change does not guarantee the satisfaction of the error requirement, and the process may be repeated over and over again, leading to extremely small integration steps. While the issue could be resolved by making use of quadruple precision floating point format, as this was not implemented, the options encountering this issue were excluded from the selection.

At first glance tolerances $\varepsilon_{rel} = 10^{-14}$ as well as $\varepsilon_{rel} = 10^{-16}$ with $\varepsilon_{abs} = 10^{-12}$ both indicate good performance without violating the requirements. However, propagations with $\varepsilon_{rel} = 10^{-15}$ and $\varepsilon_{rel} = 10^{-17}$ violated the position requirement, while they are expected to be more accurate than the two aforementioned methods. This suggests that while relative tolerances $\varepsilon_{rel} = 10^{-14}$ as well as $\varepsilon_{rel} = 10^{-16}$ satisfy the requirements in this case, they may not do so reliably and should thus be investigated further.

Additionally, in order to further investigate the reasoning behind the strong effect of absolute tolerance, the two most efficient solutions corresponding with $\varepsilon_{abs} = 10^{-13}$ are also considered, despite the seemingly higher number of function evaluations. In order to make the final selection of tolerances, two aspects of the aforementioned selected configurations are discussed in further detail. First - the efficiency of the solution. Second - the error behaviour over time.

Given a few potentially suitable configurations have been selected, it is now possible to investigate the performance of each option in more detail. As discussed in earlier chapters, one of the concerns for the efficiency of the regularized propagator is that the state estimation procedure requires propagation of reference trajectories to a precise final time (corresponding to a measurement). As the implemented regularized propagator performs propagations using an independent variable φ which is different to time, and its final value is unknown prior the propagation, it is necessary to employ root finding to terminate at the appropriate time. To investigate the potential impact to the performance of the propagation, the breakdown of the function evaluations due to the integration steps and the root finding is shown in Figure 6.12, where the chosen candidates are also compared to the Cowell integration settings selected in the previous section.

It is rather clear that regardless of the final choice of tolerances, the regularized propagations require a fraction of the function evaluations of the Cowell propagation. However, the root finding contributes on average around 8% (~ 80) of the total necessary state derivative evaluations, which is likely necessary regardless of the length of the propagation. This implies, for propagations corresponding to short measurement gaps in the estimation, the contribution of the root finding may cause the relative performance of the regularized method to diminish. This is further discussed in Chapter 7.

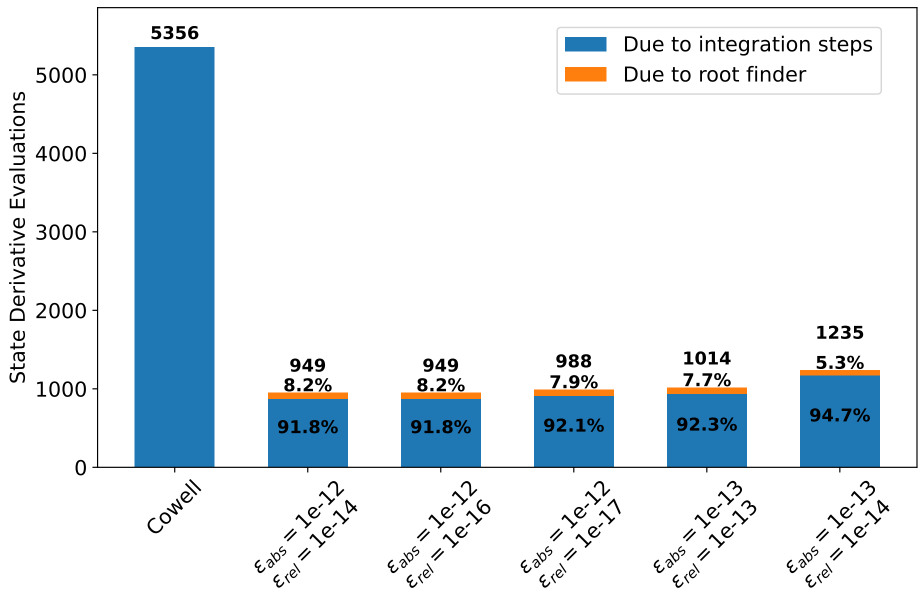
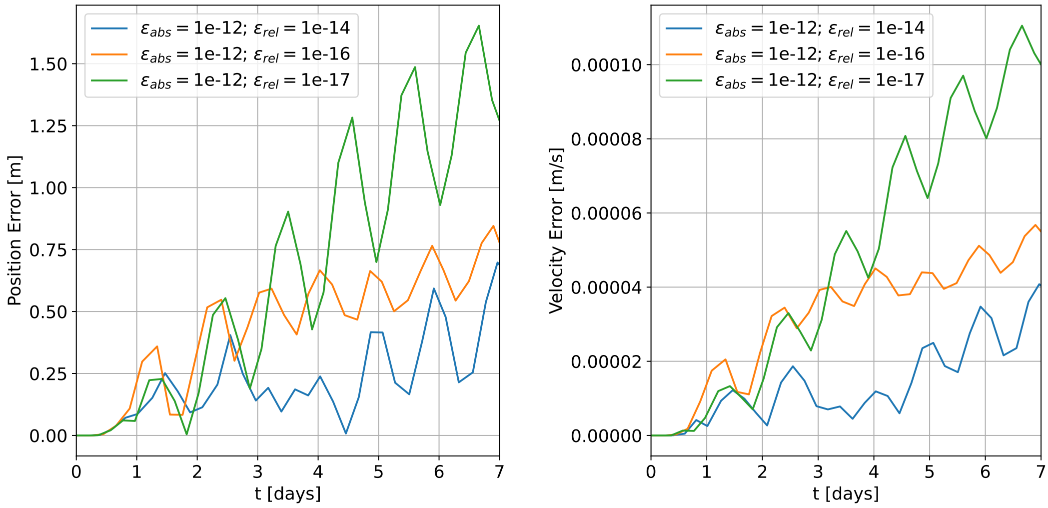


Figure 6.12: *Number of state derivative evaluations due to integration steps and root finding for selected regularized propagators, compared to a Cowell solution ($\varepsilon_{abs} = 10^{-10}$; $\varepsilon_{rel} = 10^{-12}$, as selected in Section 6.3). Solutions obtained for geostationary orbit.*

On the other hand, it can also be seen that the number of function evaluations for $\varepsilon_{abs} = 10^{-12}$ hardly depends on the relative tolerance. Tolerances $\varepsilon_{rel} = 10^{-14}$ and $\varepsilon_{rel} = 10^{-16}$ even result in the same total. It was confirmed that this was a slight coincidence, as the higher tolerance solution takes 3 fewer time steps, though it does iterate additional times in order to ensure the truncation error is within the specified bounds. Regardless, this behaviour implies the absolute tolerance's dominance. For the choice of $\varepsilon_{abs} = 10^{-13}$ in Figure 6.12, the dependence on relative tolerance is immediately more obvious, where, as expected, decreasing it results in additional function evaluations.

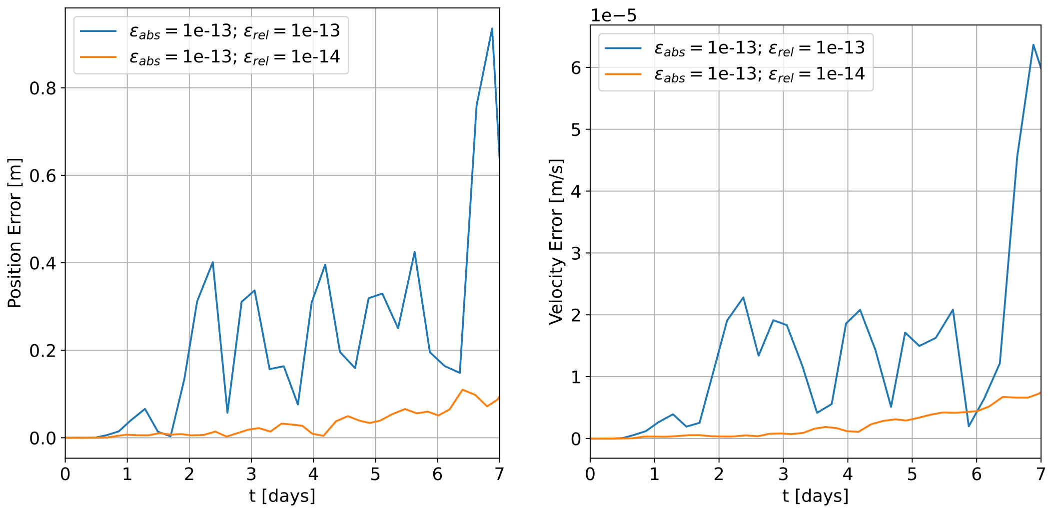
The second aspect that should be treated is the error behaviour over time, which is shown in Figure 6.13 for the solutions dominated by the absolute tolerance. It can be seen that all three of these solutions behave in a very similar manner for approximately the first two and a half days of the propagation, after which the integration with $\varepsilon_{rel} = 10^{-17}$ takes a slightly larger step in time and the errors grow apart. Considering this tolerance is expected to result in a more accurate solution, it seems plausible that in the context of an estimator, when propagating reference trajectories with $\varepsilon_{rel} = 10^{-16}$ and $\varepsilon_{rel} = 10^{-14}$, a similar 'mistake' may occur and result in a violation of the position error requirement. Especially so when considering the margin is rather small.

As the performance losses from reducing the absolute tolerance were not excessive (as was seen in Figure 6.12), the behaviour of the error for these settings was also investigated as shown in Figure 6.14. While both methods satisfy the requirements, $\varepsilon_{rel} = 10^{-13}$ results in an unexplained and very significant jump in error on the final day of the propagation. Considering the nature of this behaviour was not identified, the more well-behaved solution with $\varepsilon_{rel} = 10^{-14}$, which also provides additional margin, was selected for use in the R-OCBE, despite the slight loss in performance.



(a) *Position Error* **(b)** *Velocity Error*

Figure 6.13: *Geostationary Orbit test case integration error over the course of a week using different RKF7(8) tolerances, with respect to fixed step solution.*



(a) *Position Error* **(b)** *Velocity Error*

Figure 6.14: *Geostationary Orbit test case integration error over the course of a week using different RKF7(8) tolerances, with respect to fixed step solution.*

The same process was repeated for the other orbit test cases and the final selection of RKF7(8) tolerances satisfying the accuracy requirements of $\mathcal{O}(1\,[\mathrm{m}])$ and $\mathcal{O}(0.001\,[\mathrm{m/s}])$ for the regularized propagator are as follows:

- GEO - $\varepsilon_{abs} = 10^{-13}$; $\varepsilon_{rel} = 10^{-14}$
- Graveyard orbit - $\varepsilon_{abs} = 10^{-13}$; $\varepsilon_{rel} = 10^{-14}$
- Tundra orbit - $\varepsilon_{abs} = 10^{-13}$; $\varepsilon_{rel} = 10^{-15}$

7

OCBE Performance

This chapter discusses the numerical results obtained by applying the three implemented variants of the OCBE to the test cases discussed in Chapter 5, using the dynamics models defined in Chapter 6. First, Section 7.1 discusses the selection of the dynamic uncertainty parameter for the R-OCBE as well as an identified flaw. Then, the current iteration of the R-OCBE is compared to the other methods in Section 7.2, in terms of the sensitivity to measurement gaps and uncertainty. Finally, Section 7.3, discusses the ability of the methods to compensate for the presence of additional mismodelled dynamics, in form of reduced fidelity gravity models.

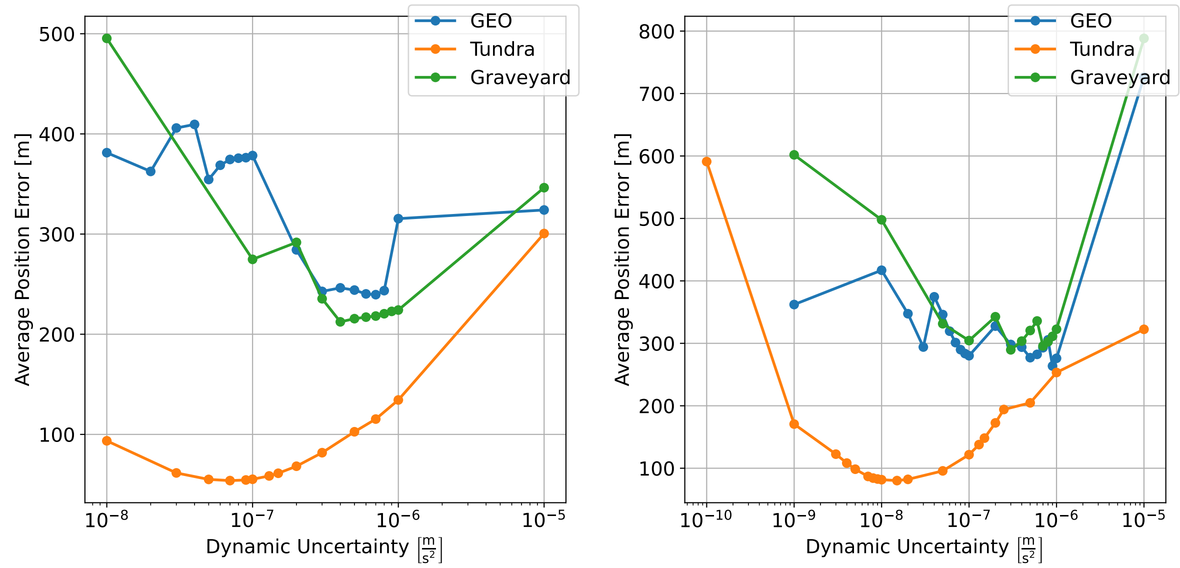
7.1. R-OCBE Dynamic Uncertainty Selection

As discussed in Chapter 3, the R-OCBE implementation does not contain the adaptive dynamic uncertainty modification. As a result, the first step necessary before comparisons between implemented variants of the OCBE are made, is a selection of an appropriate dynamic uncertainty matrix Q. For the BL-OCBE Lubey recommends a choice for Q such that $Q = \sigma_Q^2 I$, where σ_Q has the units of the control input ([m/s^2]). As a result, the approach to select appropriate values for σ_Q is rather simple: for each test case the estimation algorithm was applied for a wide range of values of dynamic uncertainty and a mean error is evaluated (excluding outlier values at the start of the estimation). For the purposes of this section, the measurements are simulated with standard deviations of 10 [m] for the range and 1 arcsecond for the angle measurements (Azimuth and Elevation). Though, choice of these values is not expected to significantly impact the choice of dynamic uncertainty.

In order to reduce the number of runs necessary, a more suitable range of σ_Q is identified by direct inspection of results, and a few additional iterations with a finer step in dynamic uncertainty are performed to find a good value. As in practice, perfect truth data is not available for tuning of the dynamic uncertainty, a true optimization of σ_Q may lead to over-optimistic results, hence a value near the minima of average error may be considered suitable to obtain a realistic representation of performance. This approach results in generation of Figure 7.1a to get an overview of the best performing option, though due to the compounding effect of dynamic mismodelling over time, it is generally suggested to scale the value depending on the measurement gap, hence the process should be repeated multiple times to find an appropriate scaling.

From the aforementioned figure an optimal dynamic uncertainty slightly below 10^{-7} [m/s^2] for the Tundra orbit is immediately obvious, with a generally smooth parabolic (given logarithmic x scale) relationship between the dynamic uncertainty and the average position error. For the other two test cases, the lowest achievable average error is approximately factor 5 higher, despite the same measurement and dynamics model errors, and they do not display the same parabolic behaviour, which is even more evident given an increased measurement gap as shown in Figure 7.1b.

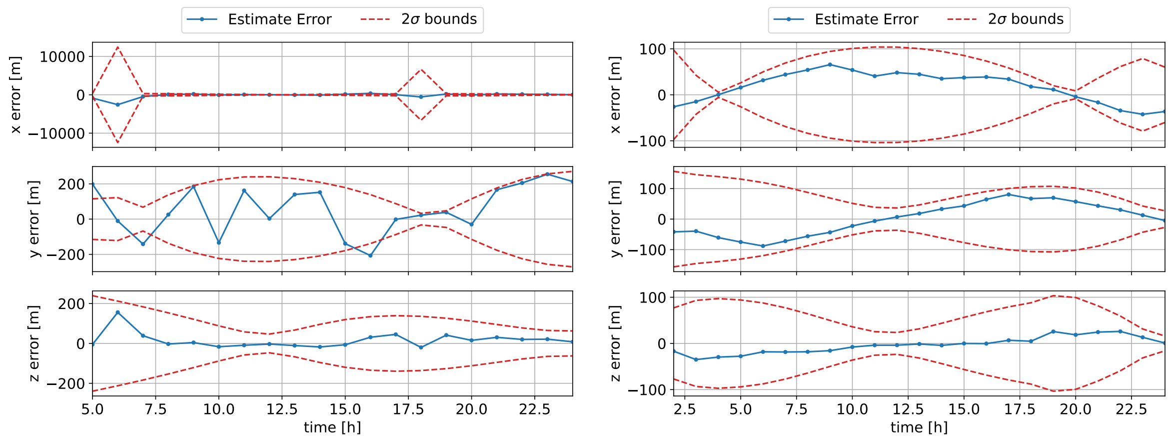
During the development of the Covariance conversion method in Section 4.3, a condition during which the method fails to obtain an updated independent variable φ and thus results in a poorly estimated Cartesian covariance was identified. Such cases were investigated in order to determine



(a) *Measurement Gap - 1.5 hours* **(b)** *Measurement Gap - 7 hours*

Figure 7.1: *Mean estimate position error with respect to the true solution as a function of dynamic uncertainty, for sequences of 20 measurements.*

if these conditions result in additional, unexpected impact on the estimator. An example of this is shown in Figure 7.2a, where a clearly divergent σ_x value is obtained precisely 12 hours apart in the estimation, despite the other dimensions remaining seemingly unaffected. That being said, a significant increase in the estimate error is also present at the same epochs, which was not originally expected. It appears that the same convergence issue (discussed in more detail in Section 4.3) is present when sigma points under these conditions are evaluated and corresponding values of φ are found.



(a) *Geostationary Orbit* **(b)** *Tundra Orbit*

Figure 7.2: *State estimate position error and 2σ bounds w.r.t. true solution, using measurements every hour.*

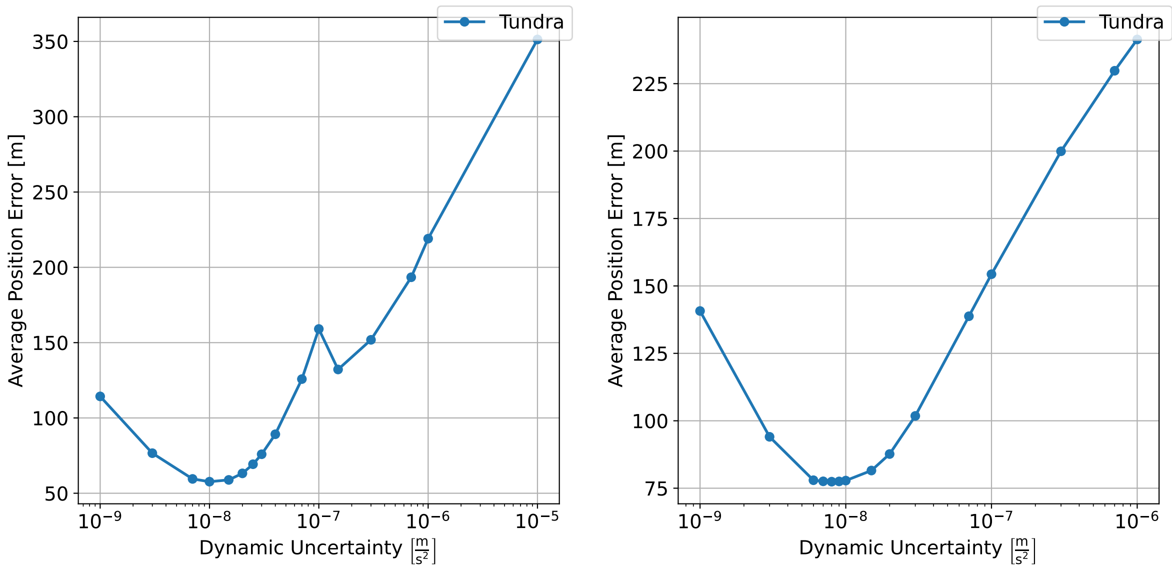
As this issue is entirely absent in the Tundra orbit (as shown in Figure 7.2b), which boasts a larger eccentricity, it is likely that the aforementioned conditions result in a failed update of φ in the estimation process, in addition to the covariance transformation. It is believed that this issue may potentially be resolved in a few ways, one of which involves estimating the independent variable

φ instead of $\lambda_{0,l}$ in order to ensure Equation 4.4 always has a solution to synchronize the estimate with the measurement. Unfortunately, due to a limited duration of the thesis, this change was not implemented and the rest of the performance evaluations in this chapter focus on the Tundra orbit, where the issue is avoided.

Returning to the topic of dynamic uncertainty selection, from Figure 7.1 it can be observed that from the tested values $\sigma_Q = 7 \cdot 10^{-8}$ [m/s^2] and $\sigma_Q = 1.5 \cdot 10^{-8}$ [m/s^2] for the 1.5 hour and 7 hour measurement gaps respectively give near-optimal solutions. Coincidentally, the ratio of dynamic uncertainties is precisely the inverse of the ratio of measurement gaps (which were selected completely arbitrarily). To test if this is a trend that can be taken advantage of, the process was repeated to identify a near-optimal value for a measurement gaps of 4.25 and 10 hours, for which an optimum are anticipated around $\sigma_Q = 2.5 \cdot 10^{-8}$ [m/s^2] and 10^{-8} [m/s^2] respectively, should Equation 7.1 be a reasonable method to obtain an approximation of a near-optimal dynamic uncertainty.

$$\sigma_Q(dt_h) = \frac{1.5}{dt_h} \cdot \left(7 \cdot 10^{-8}\right) \tag{7.1}$$

The corresponding results are given in Figure 7.3. As can be seen from the figure, the estimated Q for both measurements gaps is not quite at the optimal value, though it is near. For a measurement gap of 4.25 hours in Figure 7.3a, the optimum is around $\sigma_Q = 10^{-8}$, with a mean error $\approx$ 58 [m], whereas using the predicted $\sigma_Q = 2.5 \cdot 10^{-8}$ results in a mean error of $\approx$ 69 [m]. In case of the 10-hour gap, the optimum is around $\sigma_Q = 8 \cdot 10^{-9}$ with a mean error $\approx$ 77 [m], while the predicted $\sigma_Q = 10^{-8}$ results in a mean error only 1 [m] higher. In the context of measurement standard deviations in the order of 200 [m], this was deemed acceptable.



(a) *Measurement Gap - 4.25 hours* **(b)** *Measurement Gap - 10 hours*

Figure 7.3: *Mean estimate position error with respect to the true solution as a function of dynamic uncertainty, for sequences of 20 measurements.*

While it is likely possible to construct a more sophisticated empirical method to obtain $Q(dt)$, as it was earlier discussed, a detailed optimization of σ_Q would not be possible in a real world application. In turn Equation 7.1, provides a reasonable estimate for this particular application, while still capturing the effect of an imperfect choice of dynamic uncertainty.

7.2. Variations in Measurement Quality

With a method to select the dynamic uncertainty for the R-OCBE defined, it is finally possible to compare the three estimators. In order to answer the research questions stated in the introduction, the performance shall be evaluated on three primary metrics:

- The error of the estimates with respect to the true trajectory
- The number of state derivative function evaluation necessary to obtain the estimates
- The quality of the covariance estimate (is the state estimate error within 2σ)

In this section, the relationship of these metrics to the parameters defining measurements are discussed. Due to the expected benefits of the R-OCBE for long-term propagations, the first parameter of interest is the time gap between measurements. Furthermore, the sensitivity of the estimators to standard deviations of the simulated range, azimuth and elevation measurements is investigated. However, in order to limit the dimensionality of the problem, the three measurement standard deviations are coupled as shown in Equation 7.2 using an error scaling factor a, which may be increased in order to simulate worse quality measurements. This error scaling factor is used as an independent variable in a number of tests discussed in this chapter.

$$\vec{\sigma}_a = a \cdot \begin{bmatrix} \sigma_R \\ \sigma_{Az} \\ \sigma_{El} \end{bmatrix} = a \cdot \begin{bmatrix} 10\text{m} \\ 1\text{arcsec} \\ 1\text{arcsec} \end{bmatrix} \tag{7.2}$$

7.2.1. Mean Estimate Error

To get a first idea on the performance of the estimators, the mean error of each version was obtained for a sequence of 20 measurements with varying time gaps and the error scaling factor. For the purposes of conciseness, only error in terms of position is discussed as velocity errors display very similar behaviour. These results are shown in Figure 7.4.

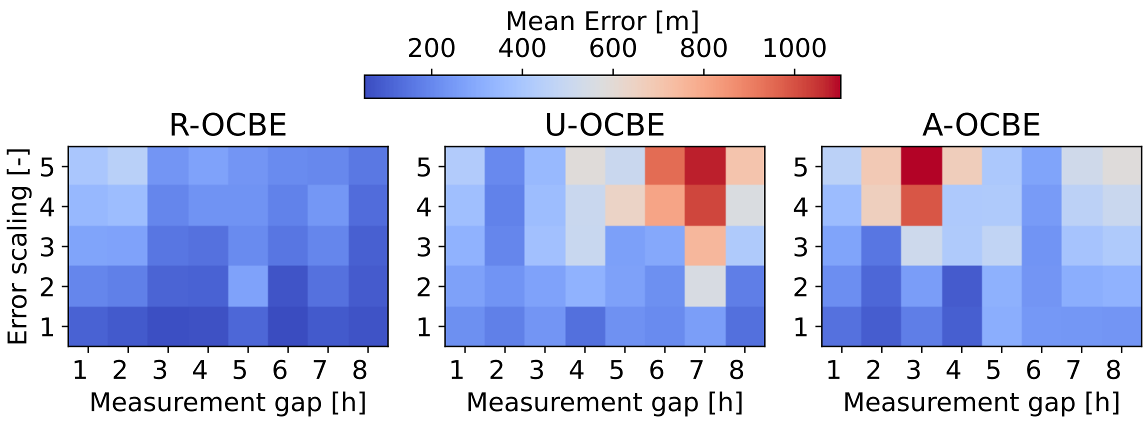


Figure 7.4: *Mean position error of the estimators as a function of the measurement error scaling and time gap. Each point of the grid is a sequence of 20 measurements in the Tundra orbit, with outliers prior to convergence excluded from the value of the mean error.*

It is rather clear that in terms of mean error the R-OCBE performs very well with respect to the other methods. The mean error is also rather unaffected by the measurement gap. The U-OCBE displays generally larger error, which also increases further with larger values of both error scaling and the measurement gap. On the other hand, while generally the A-OCBE displays a similar trend, there does appear to be a set of outliers for measurement gaps of 2 or 3 hours with high error scaling.

These examples were investigated further, and it was identified this effect was caused by a flaw in the post-processing of the results rather than the estimator. In order to ensure the state estimates prior to filter convergence do not affect the comparison of the OCBE methods, n datapoints with error magnitude above a threshold were deemed as outliers and hence not included in the mean error in Figure 7.4. However, as shown in Figure 7.5, in particular cases, a number of outliers prior to filter convergence were not identified (as the magnitude of the error was just under the threshold) and hence had a severe negative impact on the mean error.

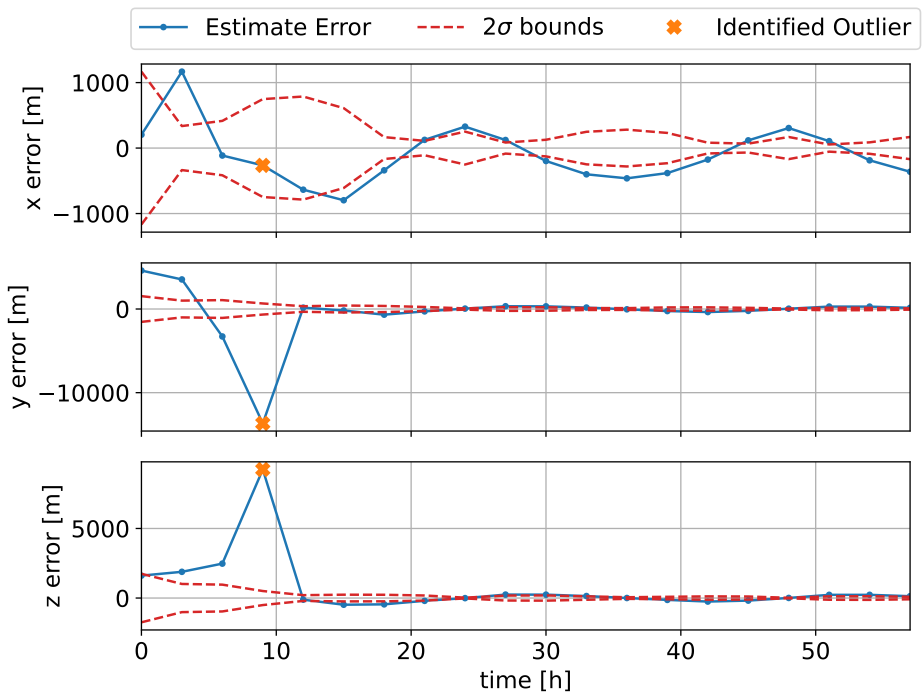


Figure 7.5: *A-OCBE Estimate position error w.r.t. to true solution, for: Measurement gap = 3 hours, Error scaling factor = 5*

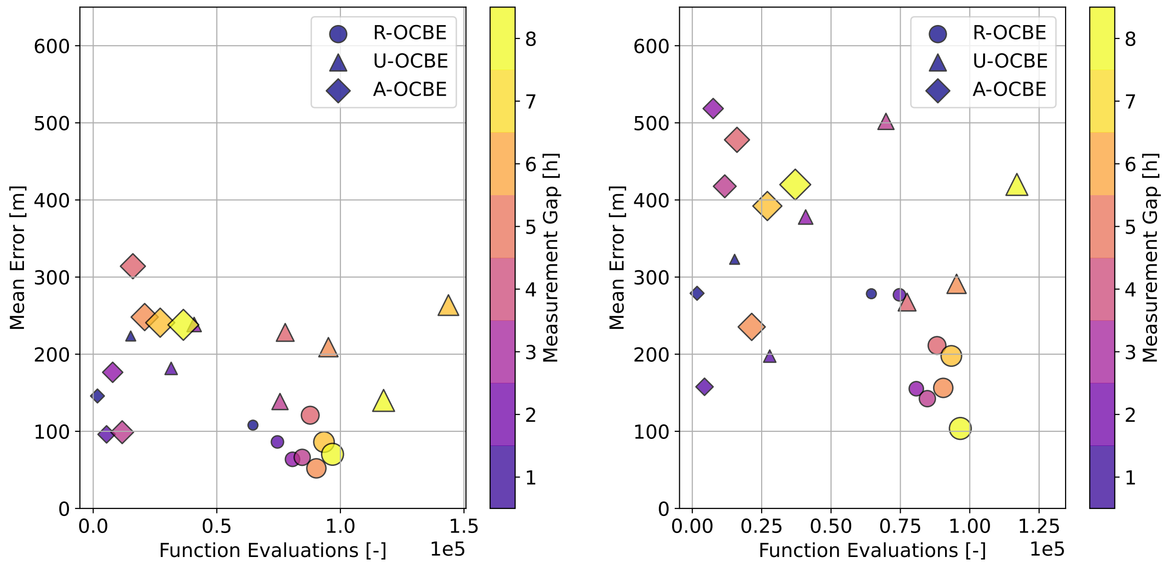
For example, the result shown in Figure 7.5 originally resulted in the highest mean error of all OCBE tests, at ≈ 1100 [m]. After revisiting and excluding three additional estimates at the start of the simulation, the mean error was reduced to a much more reasonable ≈ 490 [m], which is also more in line with the other tests. While this example was the most severe case, it was ensured that the outliers had not affected any results outside the 5 highest error configurations of the A-OCBE in Figure 7.4 and having made the appropriate adjustments, the method comparison was continued. Given that it was desired to highlight the presence of this post-processing artifact and its correction does not suggest any previously unidentified trends in Figure 7.4, an updated version of the Figure is omitted from this report for the sake of conciseness.

7.2.2. Estimation Costs

While Figure 7.4 provides an overview of the estimate error, of the methods, it does not provide any information when it comes to the costs of the estimation process, which is also of great interest.

Using the same data as in Figure 7.4, Figures 7.6a and 7.6b were generated, in which the theoretical best performing algorithms resulting in minimal error at the lowest cost are in the lower left corner.

Starting with the smaller measurement standard deviations in Figure 7.6a, it can indeed be seen that the output of the R-OCBE (indicated by the circular markers) results in a well grouped, very small mean error (50-100 [m]), with the necessary function evaluations ranging between $6 \cdot 10^4$ (for $dt = 1$ [h]) to $9 \cdot 10^4$ (for $dt = 8$ [h]), which is higher than quite a few of the other algorithms.



(a) *Measurement error scaling factor = 1*

(b) *Measurement error scaling factor = 3*

Figure 7.6: *The mean error and state derivative function evaluations of the implemented OCBE variants for varying measurement gap, which is indicated by the size of the markers in addition to the colour.*

Speaking of which, for small measurement gaps (particularly 2 - 3 hours), the A-OCBE (indicated by the diamonds) results in comparable error (≈ 100 [m]), at much fewer function evaluation (≈ 2000). This massive distinction in computational cost can be explained when considering the R-OCBE requires propagation of not only the estimated state itself, but also 16 sigma points between each measurement epoch. Furthermore, as it was already discussed in Chapter 6, the necessary root finding for the regularized propagator was expected to result in significant increase in costs, especially for low duration propagations. However, while the costs of the A-OCBE remain much lower than the R-OCBE even for higher measurement gaps, this comes at the cost of mean error growth to around 250 [m]. This increase in error was generally anticipated given the linear nature of the estimator, applied on a non-linear system, limiting the range over which the method is most accurate.

While these results seemingly already indicate the use cases for both of the two aforementioned algorithms, the U-OCBE (indicated by triangles) error seems generally higher than the other methods, while remaining more expensive than the A-OCBE. It is also seemingly by far the most sensitive method to measurment gap, at least in terms of function evaluations, as can be seen from the wide spread along the x-axis in Figure 7.6a. This can be explained considering the combined effect of the unscented transform and the adaptive dynamic uncertainty (Q) modification, which is also present for the U-OCBE. For the A-OCBE, the cost is limited as only one trajectory must be propagated between each measurement, with three additional trajectory propagations per Q adjustment necessary. On the other hand, to perform the unscented estimation, 12 additional sigma points require propagation between every measurement epoch, which must each also be propagated three additional times if the dynamic uncertainty is adjusted, causing significant growth in costs. That being said, for higher values of the measurement gap, the U-OCBE does result in lower error than the A-OCBE. This was also anticipated, given that the Unscented modification should aid in the handling of the non-linear dynamics effects that compound over time and damage the accuracy of the A-OCBE.

To inspect the effect of an increase in measurement standard deviation, Figure 7.6b was also created. Given a factor 3 increase in measurement uncertainty, the distribution of points changes quite significantly compared to Figure 7.6a. The first notable effect is that the U-OCBE results in generally lower mean error than the A-OCBE. This result implies that the U-OCBE benefits of better handling the non-linearity of the system are beneficial when exposed to poor quality measurements.

On the other hand, while the R-OCBE generally remains the most expensive in Figure 7.6b, it also results in fantastic error performance compared to the other methods. This error also seems relatively unaffected by the measurement gap, similar to the conditions for a measurement scaling factor 1 in Figure 7.6a. The original motivation for the approach of estimating regularized elements hinged on the idea that a linear estimator, applied to perturbed-linear (regularized) dynamics would perform better than if it were applied directly on the non-linear system (Cartesian equations of motion). While it is difficult to confirm that this is indeed the reasoning behind the consistently low mean error, which is generally less sensitive to measurement gap, it is a very plausible explanation. It should also be noted that the R-OCBE results are also the most tightly grouped in terms of function evaluations, which in turn implies the measurement gap (and thus propagation duration) has a much smaller effect on propagation costs for the regularized elements rather than a Cowell propagation, further confirming the findings of Section 6.4.

As a final note before moving on to testing the covariance estimate and the sensitivity of the estimators to the known dynamics model, an additional investigation to determine the reasoning behind the costs of the R-OCBE was performed. Part of the expenses can be attributed to the need to propagate 16 additional sigma points next to every state estimate trajectory (compared to only 12 for the U-OCBE). However, in Section 6.4 a concern was also raised, that the root finding necessary to terminate the propagation precisely at the measurement epoch would result in performance losses, particularly for low measurement gaps. To quantify the impact of this limitation Figure 7.7 was created. In the figure, the total state derivative function evaluations over the course of the estimation is shown as a function of measurement gap, with an indication as to how much of the costs can be attributed to the root finding.

As is rather obvious from the figure, the root-finder contribution to the costs is very significant, at around 40% for a measurement gap of one hour. Fortunately it remains relatively constant regardless of the measurement gap as the number of measurements handled for each datapoint is the same. Considering the earlier discussed small spread of points along the x-axis in Figure 7.6 compared to the other estimators, one can conclude that the propagation over time itself is much more efficient than in the cases of the Cowell propagator in A-OCBE and U-OCBE, and it can in theory become more efficient given extremely long measurement gaps. However, the costs of the R-OCBE for low measurement gaps can also likely be significantly reduced if:

- The R-OCBE modifications are applied to the A-OCBE instead of the U-OCBE, hence avoiding the propagation and root finding for each sigma point. Though, this may come at the cost of estimator error as the Unscented transform is expected to aid with the handling of non-linear perturbations, which are present even for the regularized elements.

- The propagation is performed in terms of time as the independent variable rather than φ. While the need for root finding is completely eliminated with this approach, it will make the cost of the estimator more sensitive to the measurement gap as the propagation itself will be less efficient.

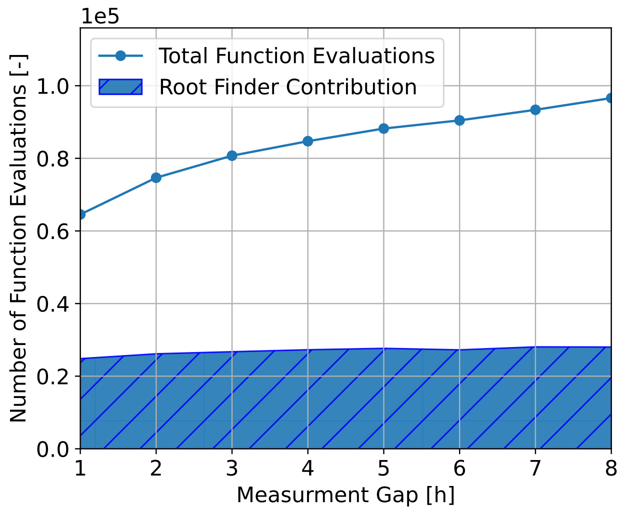


Figure 7.7: *State derivative function evaluations for the R-OCBE as a function of measurement gap. For each measurement gap, a series of 20 measurements was processed.*

7.2.3. Covariance Estimate Quality

While the earlier discussion shows the performance of the methods in terms of error with respect to the true solution, in practical applications this error is not available. Hence, a different metric to quantify the accuracy of the results is necessary. This is generally done using covariance estimates which are also obtained during the estimation process. The quality of this estimate, is the final performance metric with respect to which the OCBE variants are compared. In practice, it the covariance matrix may be used to get a measure of the confidence of the estimate by using the standard deviations σ^2 on the diagonal (though this does assume the errors are mostly decoupled). If the error with respect to the truth data is often outside the 2σ bounds, the estimator is effectively over-confident (the covariance is optimistic) and in practical applications may result in larger than expected error. Conversely, if the standard deviations are very high, the accuracy of the method is under-appreciated (the covariance is pessimistic), which may have adverse implications for high-precision applications.

In order to qualitatively investigate the covariance estimates, the error of the estimates over time is plotted, next to the 2σ bounds. As the trends are generally similar throughout the estimation sequences performed for the earlier results, two specific cases were selected for discussion in more detail in this section. In order to ensure a fair comparison of the different OCBE variants can be made, the first case discussed in Figure 7.8, treats conditions where the mean error of all three estimators is comparable (measurement error scaling factor 3 and a measurement gap of 6 hours).

Starting with the behaviour of the R-OCBE, (indicated with blue in Figure 7.8) it can be seen that the error is at all points contained within the 2σ bounds, though often the standard deviation is significantly higher than the actual error. It also displays rather odd triangular-wave behavior, with relatively constant amplitude, which may imply that the R-OCBE has a tendency to overestimate the

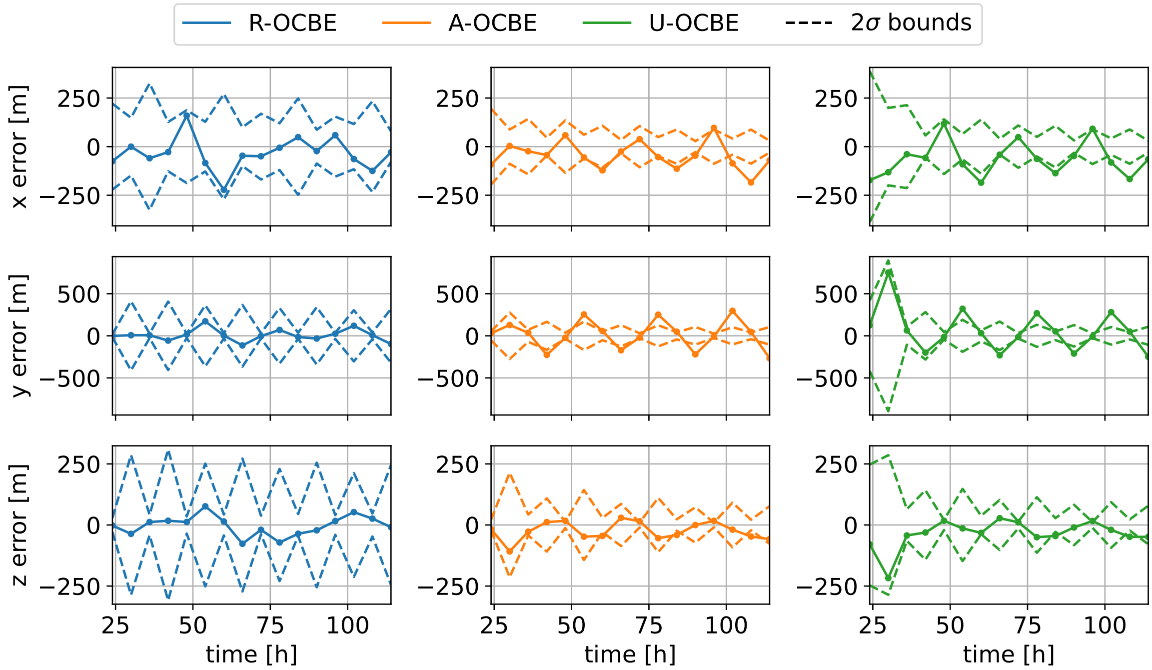


Figure 7.8: *Inertial position error of the estimators as a function of time over the course of the Tundra trajectory. Measurement error scaling = 3, measurement gap of 6 hours.*

covariance, though it is able to identify this and correct it for a subsequent measurement. While the precise underlying reason for this behaviour was not identified, it is likely that the the severity of the effect may be reduced should smoothing also be incorporated in the R-OCBE.

When it comes to the A-OCBE and U-OCBE in Figure 7.8, both of the methods indicate generally lower standard deviation values, which also result in the majority of the estimates within the 2σ bounds. However, there are a few data points where the bounds are violated, in particular in the latter half of the simulation where the covariance has further converged.

While the example above treats a case where all estimators are performing relatively equally in terms of mean error, the cases where the R-OCBE performs best are also of interest, as under these conditions the estimator is most likely to be used. Such conditions are shown in Figure 7.9, which corresponds to an 8-hour measurement gap rather than 6. In these conditions, the R-OCBE error is again always within the 2σ bounds, though after the initial ~ 75 hours of the simulation, the covariance remains constant despite the decrease in error and once again results in over-estimated uncertainty. It also maintains some oscillatory features originally observed in Figure 7.8, though it should be noted that for this measurement gap the standard deviation minima are obtained every orbital period, which potentially implies a relationship to the eccentricity of the orbit.

On the other hand, as previously discussed, the error for the A-OCBE and U-OCBE grows substantially for this measurement gap, and from Figure 7.9 it can be seen that it also results in the error exceeding the 2σ bounds in the second half of the trajectory. That being said, while in this particular case the two methods result in a very similar mean error, it does appear that the covariance obtained using the Unscented transform is larger and results in fewer state estimates outside the 2σ bounds, which does suggest the U-OCBE results in slightly better overall results for such conditions.

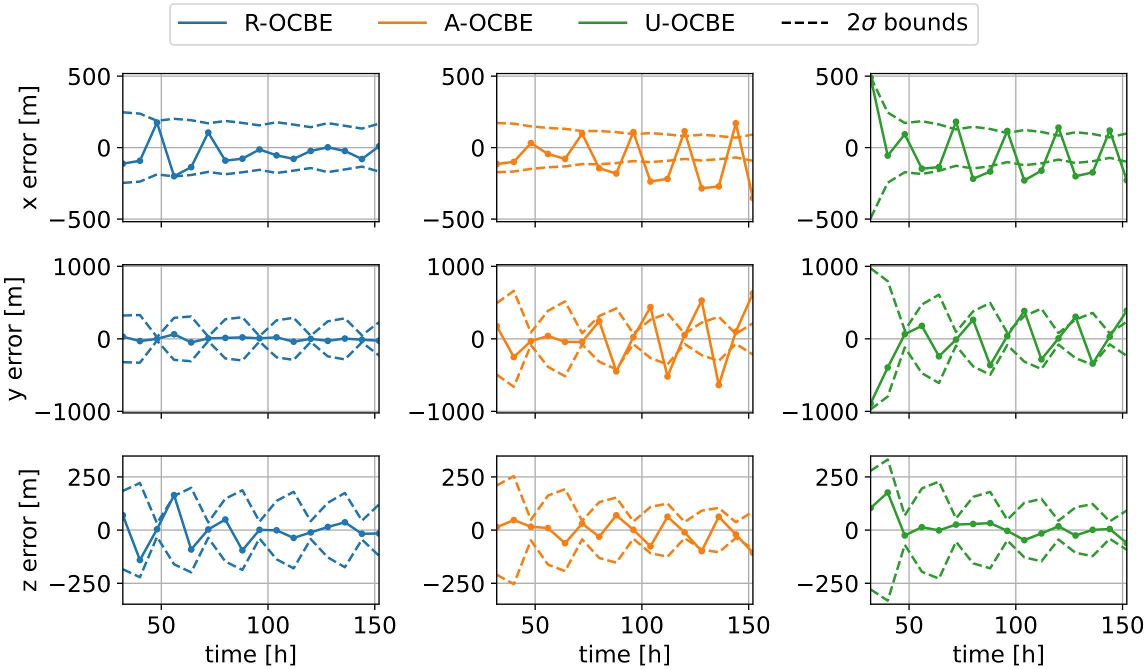


Figure 7.9: *Inertial position error of the estimators as a function of time over the course of the Tundra trajectory. Measurement error scaling = 3, measurement gap of 8 hours.*

Despite the fact that the estimates obtained via A-OCBE and U-OCBE occasionally do violate the bounds, it can be argued that the significant over-estimations of the R-OCBE may be more detrimental to some applications. As multiple other cases display similar behaviour, the reasoning behind the overestimation of covariance of the R-OCBE should be discussed. While part of the effect may be attributed to the lack of smoothing, another aspect should be considered. In the state estimation process, the method to update the state with a correction (discussed in Section 4.1), is not as straight forward as simply taking a sum of vectors, as in the case of estimation in terms of Cartesian state. During this process of applying a correction to the state, constraints are applied to the state vector $\vec{\lambda}$, though the effect of this process on the corresponding covariance is not treated. Indeed, in context of Kalman filtering with state constraints, particularly when a quaternion is present in the estimated vector, it has been found that applying a correction to the covariance is beneficial [53]. While such a modification was not made to the prototype R-OCBE implemented in this thesis, it should be considered for future versions of the method as it may aid in ensuring less pessimistic values of the covariance are obtained as part of the results.

7.3. Variations in Known Dynamics

The last research question that has yet to be treated focuses on the impact of dynamics mis-modelling on the OCBE. While the results obtained in the previous section use a dynamics model in the estimator with a simplified model for solar radiation pressure to simulate imperfect knowledge of the environment (cannonball instead of specular-diffuse panels), this provides little insight in terms of the impact of the dynamics model on the overall performance. As a result, in this section the estimation procedure is repeated, this time varying the gravity model fidelity instead of the measurement uncertainty.

Figure 7.10 shows the mean error of each of the estimators for a varying measurement gap, given a known dynamics model with a varying number of terms in the spherical harmonic gravity potential series. As the goal of this analysis is to determine the magnitude of the impact of simplification of the known dynamics model, rather than evaluate the particular contributions of specific zonal, sectoral or tesseral terms, the order of the spherical harmonic series is always selected to be equal to the degree (and both are thus indicated by the y-axis in Figure 7.10). Furthermore, the truth data remains matching that of the previous section for all cases.

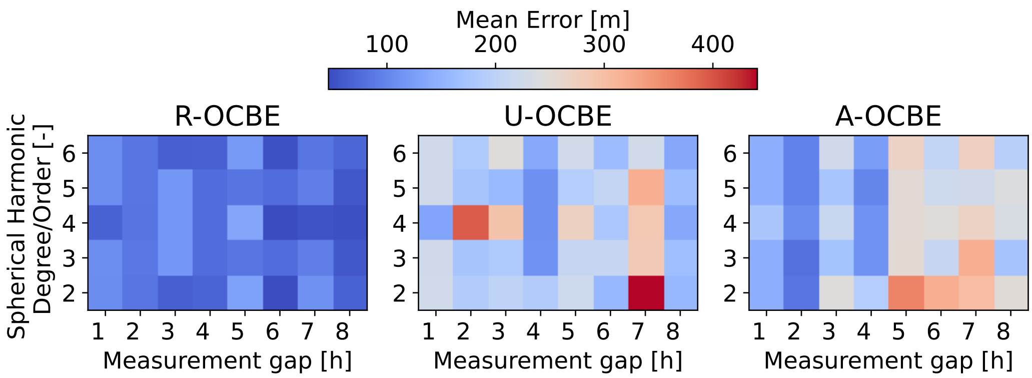


Figure 7.10: *Mean position error of the estimators as a function of the known dynamics spherical harmonic gravity model fidelity and measurement time gap. Each point of the grid is a sequence of 20 measurements (error scaling factor 1) in the Tundra orbit, with outliers prior to convergence excluded from the value of the mean error.*

Inspecting the results in Figure 7.10, one might observe that there is little in terms of an observable trend due to variation in number of spherical harmonic series terms, regardless of the measurement gap. Originally, it was expected that reducing the fidelity of the dynamics model would degrade the performance of the R-OCBE as a new choice of dynamic uncertainty would be required. On the other hand, it was predicted that the U-OCBE or A-OCBE, would remain relatively unaffected by the changes in known dynamics, given that the adaptive modification present is used to automatically update the dynamic uncertainty to a suitable value.

The reason these expectations were not met, may actually be determined by considering the magnitudes of the mis-modelled accelerations. Earlier, in Section 6.2, (or more precisely Figure 6.7) the magnitudes of a variety of perturbing accelerations over an orbit were presented. There it could be seen that the solar radiation pressure induces an acceleration in the order of magnitude of $\mathcal{O}(10^{-7})$ [m/s^2], which is larger than any of the spherical harmonic gravity terms with contributions lesser than that of the (2, 2) term. As in all tests a mis-modelling was introduced to the SRP by using a cannonball model instead of the panelled specular-diffuse method, the error in terms of acceleration could already be expected between $\mathcal{O}(10^{-8})$ and $\mathcal{O}(10^{-7})$ [m/s^2]. This error was taken into account in the dynamic uncertainty (σ_Q) selection in Section 7.1, which resulted in appropriate values in the range of $\mathcal{O}(1.3 \cdot 10^{-8})$ - $\mathcal{O}(10^{-7})$ [m/s^2] (depending on measurement gap). This implies that the R-OCBE is already 'prepared' to take into account mis-modelling larger than the contributions of spherical harmonic gravity terms, which in turns causes a simplification of the model not to have a significant impact on the error.

Similarly, from manual inspection of the dynamic uncertainties obtained by the A-OCBE and U-OCBE it was found that they also generally converge to a similar range of σ_Q, though it varies slightly between different tests. This further confirms that the selection process in Section 7.1 was suitable, while also suggesting that given a dominant mis-modelled acceleration, lower magnitude contributions may be excluded from the known dynamics model. This should improve runtimes without significant losses in accuracy, nor a need for adjustments to the dynamic uncertainty choice.

While this test failed to identify behaviour of the OCBE variants given changes in dynamics model, it is still possible to test the influence of dynamics by introducing mis-modelling noticeably larger than the SRP. This could be done by, for example, introducing a bias to the spacecraft mass or decoupling the spherical harmonic degree and order to test a gravity model taking into account only the J_2 term. Unfortunately, this would require some modification to the interface module and due to time constraints on the thesis it was not deemed possible and is left for future work.

8
Conclusions & Recommendations

As of late, a dramatic increase in space traffic has resulted in rapid growth of a variety of objects, from active spacecraft to debris, in orbit around Earth. Tracking these objects is crucial to Space Situational Awareness initiatives and collision avoidance. This includes orbits in the Geostationary region, where spacecraft with large area to mass ratios are common. This condition often results in rather significant dynamics mis-modelling (due to the significant impact of solar radiation pressure), which degrades the performance of many state estimation algorithms used for tracking.

A notable state estimation algorithm is the OCBE (Optimal Control Based Estimator), first introduced by Daniel P. Lubey [13]. The method is well suited to performing in the presence of mis-modelled dynamics and is thus a good fit for the aforementioned conditions, despite its generally rather high costs. While variants of the OCBE have been developed in literature including an unscented variant of the algorithm, in the context of orbital mechanics, the linear estimator is generally applied on the non-linear dynamics, which limits its range of applicability.

In the field of orbital mechanics propagation, the pursuit of efficiency has led to formulations of dynamics in terms of linear equations by means of regularization. One of such regularization method, referred to as EDromo [28], provides an 8-dimensional state representation in terms of elements $\lambda_{0,l} - \lambda_7$, and a new, angle-like independent variable φ. This formulation results in linear underlying dynamics (at least in the unperturbed case). Applying a linear estimator to the EDromo linear-perturbed state dynamics was thus considered a potential opportunity for performance gains in the context of state estimation. While a similar approach in literature has been done in the context of a Kalman filter [31], the same could not be said about the OCBE. Hence, as the primary product of this thesis, a prototype version of the Regularized Optimal Control Based Estimator was developed. This was achieved by modifying the Unscented Optimal Control Based Estimator in three key aspects.

First of all, in order to ensure the state estimation can be performed directly on the EDromo elements, a method to apply a correction vector to the state without violating the constraints was necessary. It was found that to satisfy the state constraints it is sufficient to simply normalize the quaternion after the vector summation of the original state and the correction vector. However, as this results in updates to the first four EDromo elements, using the updated vector to reconstruct a Cartesian state representation, results in state estimates at epochs different to the measurement epochs. This issue was resolved by adding an additional step to the state correction process. As the angle-like independent variable φ is necessary to convert the elements to a Cartesian state (and corresponding time), it may be updated together with the state estimate by means of numerical root-finding such that it enforces a fixed epoch.

Next, a matrix mapping a control input in terms of acceleration to the EDromo state derivatives was necessary to obtain a quadrant of the state transition matrix necessary for state estimation with the OCBE. This control input scaling matrix B was derived as a matrix product, of two components B_1 and B_2. The (3×3) B_1 matrix, maps a control input acceleration in the inertial frame, to the projections found in the regularized element dynamics. The (8×3) B_2, maps the aforementioned projections to their contribution to each of the regularized element derivatives.

Finally, two methods to convert covariance matrices between a Cartesian representation and an EDromo representation were developed and tested. While it was found that the Monte Carlo approach was not suitable for low eccentricity orbits, linear covariance mapping was found to work well in all conditions, bar two singularities for low eccentricity orbits. Additionally, due to the dimensionality of the EDromo elements, it was found that the corresponding covariance matrix becomes positive semi-definite, rather than positive-definite as assumed by the estimator. Hence, covariance remediation was also implemented as a necessary step in the estimation process.

With the methods implemented and numerical error of the dynamics models quantified, the research questions were answered by investigating the performance of the R-OCBE and comparing it to the A-OCBE and U-OCBE variants available in literature.

Based on the investigation to answer first research question '**How does the performance of the regularized OCBE depend on measurement frequency and uncertainty?**', it was found that the R-OCBE results in notably lower state estimate errors, regardless of the measurement uncertainty, compared to the A-OCBE and U-OCBE, especially for measurement gaps larger than 4 hours. However, a tendency to obtain a pessimistic covariance estimate was also identified.

During the investigation treating the second research question '**How does the performance of the regularized OCBE depend on the severity of dynamics mis-modelling?**', it was found that the method is capable of performing well, regardless of the presence of mis-modelled solar radiation pressure, given an appropriate selection of the dynamic uncertainty parameter. As further mis-modelling was introduced with a reduction in the fidelity of the spherical harmonics gravity model, the overall estimate quality was not impacted significantly. This was attributed to the fact that during the selection process of the dynamic uncertainty, a mis-modelling in the order of $\mathcal{O}(10^{-7})$ [m/s^2] due to SRP was already introduced. This mis-modelling was generally larger than any of the contributions of the spherical harmonic gravity terms. Hence, the selected dynamic uncertainty was already sufficiently high to allow the estimator to compensate for the mis-modelling introduced by truncation of the spherical harmonic gravity series. However, while it was not tested directly, it is still expected that a mis-modelling larger than what was used in the dynamic uncertainty selection will degrade the performance of the R-OCBE, given the lack of the adaptive modification.

The answer to final research question '**How does dynamics regularization affect the computational efficiency of the OCBE?**', was clear based on all previously obtained test results. It was immediately obvious that the aforementioned accuracy benefits of the R-OCBE come at significant computational costs. This was attributed primarily to the presence of the unscented transform and the inability of the regularized propagator to terminate at a measurement epoch without employing root finding for the angle-like independent variable φ. However, while the costs of the R-OCBE were generally higher than the A-OCBE or U-OCBE, the method was less sensitive to gaps between measurements, and it may thus still result in cheaper estimates given extremely long measurement gaps.

While some of strengths and weaknesses of the R-OCBE were identified, there are many additional steps that can be performed to both further evaluate the performance of the algorithm, and also to potentially improve it.

When it comes to steps to further evaluate the performance of the method, one aspect of the OCBE that has not been implemented as part of the thesis is the estimation of the mis-modelled acceleration as a control input. While the first step of this process (evaluation of adjoint estimates at measurement epochs) was implemented, the propagation of said adjoint, necessary to obtain the control input as a function of time was not. The quality of this reconstruction would be another suitable performance metric to evaluate the R-OCBE performance.

Furthermore, the truth data tested involves only trajectories with no maneuvers. Introducing discontinuities such as impulse maneuvers may also highlight the different OCBE variants performance when it comes to changing dynamic uncertainty. As the prototype R-OCBE does not include the 'adaptive' modification, its performance is expected to degrade in this case. However, the severity of this degradation is unclear and should be investigated.

Throughout the thesis a number of issues in the R-OCBE were identified, though not fully resolved. First of all, the linear mapping covariance transformation contains two singularities for low eccentricity orbits, caused by the application of a central difference method to determine the necessary Jacobian matrices. During this process, when a new regularized state vector is obtained (one including a finite step in one of the elements), an update of the independent variable φ is necessary to find partial derivatives corresponding to the appropriate time. Under particular conditions (twice per orbit) for very low eccentricity, the method to update φ has no solution, resulting in a very poor covariance conversion. Though, is likely that this issue can be eliminated entirely given analytical derivation of the Jacobian matrices.

The same issue resulting in failure to update φ is also encountered in the estimation process for low eccentricity orbits, which results in significant increase in estimate error. This issue is predominantly found when evaluating sigma points, as per the unscented transformation present in the R-OCBE. One approach to alleviate the issue is to omit the unscented modification, though this will require additional modifications to the BL-OCBE equations. Alternatively, the issue may be resolved by changing the vector that is modified in the state estimation. In the current implementation, state elements $\lambda_{0,l}$ - λ_7 are updated by the estimator, and φ is solved for such that it ensures the state estimate corresponds to the measurement time. Instead, if φ was updated with the estimator next to λ_1 - λ_7, and $\lambda_{0,l}$ was solved for to apply the measurement time constraint, a solution should always exist potentially avoiding the error.

As meantioned earlier, despite its accuracy, the R-OCBE was found to be more computationally expensive than initially expected. Two approaches have been identified that may compensate for this weakness. The most significant part of the costs is due the unscented transformation applied on an 8-dimensional state. It requires propagation of 16 additional sigma points, which results in significant additional propagation costs. Omitting the unscented transform would reduce the costs drastically, though it may result in slightly less accurate state estimates. The second reason for the high costs is the root-finding necessary to terminate an EDromo propagation precisely at a measurement epoch, given that the propagations are run in terms of an angle independent variable φ. This primarily affects short measurement gap performance, and it can be remedied by using time as the independent variable in the propagations instead. Such a modification would render the root-finding entirely unnecessary, though it has been found in literature that the propagator is less efficient when not using φ, hence the computational efficiency would degrade for long measurement gaps.

Finally, it should be noted that the R-OCBE prototype may further be improved by incorporating the adaptive modification and smoothing of the estimates. Both of these steps are expected to make the estimator more robust at insignificant additional costs.

All in all, the need of state estimation in the presence of mis-modelled dynamics is widespread and growing due to the uptick in constellations and other spacecraft in orbit around Earth. The implemented prototype of the R-OCBE already shows promise for high precision automatic tracking of objects with poorly modelled dynamics, particularly given significant gaps between measurements. Given the modifications proposed in this thesis, the efficiency of the algorithm may also be improved further, such that the method remains competitive for short gaps between measurements. The R-OCBE may then form the basis for the next generation of the SSA cataloging pipeline, meeting the ever-growing demands in both accuracy and efficiency.

Bibliography

[1] ESA. Space debris by the numbers, December 2023. Online; accessed 21-March-2024, https://www.esa.int/Space_Safety/Space_Debris/Space_debris_by_the_numbers.

[2] NASA. The international space station transition plan, September 2023. Online; accessed 16-March-2024, https://www.nasa.gov/faqs-the-international-space-station-transition-plan/.

[3] Ryan Whitley and Roland Martinez. Options for staging orbits in cis-lunar space. October 2015. https://ntrs.nasa.gov/api/citations/20150019648/downloads/20150019648.pdf.

[4] Emily Spreen, Kathleen Howell, and Diane Davis. Near rectilinear halo orbits and nearby higher-period dynamical structures: orbital stability and resonance properties. *Celestial Mechanics and Dynamical Astronomy*, 132, 06 2020. URL https://doi.org/10.1007/s10569-020-09968-2.

[5] Nasa spacecraft conjunction assessment and collision avoidance best practices handbook. February 2023.

[6] T. Schildknecht, R. Musci, and T. Flohrer. Properties of the high area-to-mass ratio space debris population at high altitudes. *Advances in Space Research*, 41:1039–1045, 2008. URL https://doi.org/10.1016/j.asr.2007.01.045.

[7] Ioannis Gkolias, Jerome Daquin, Fabien Gachet, and Aaron Rosengren. From order to chaos in earth satellite orbits. *The Astronomical Journal*, 152, November 2016. URL https://doi.org/10.3847/0004-6256/152/5/119.

[8] Alan Jenkin, John Mcvey, James Wilson, and Marlon Sorge. Tundra disposal orbit study. In *7th European Conference on Space Debris*, volume 7. ESA Space Debris Office, 2017. URL https://conference.sdo.esoc.esa.int/proceedings/sdc7/paper/328.

[9] Byron D. Tapley, Bob E. Schutz, and George H. Born. *Statistical Orbit Determination*. Elsevier Academic Press, 1st edition, 2004. URL https://doi.org/10.1016/B978-0-12-683630-1.X5019-X.

[10] X. Rong Li and Vesselin P. Jilkov. Survey of maneuvering target tracking. part v: Multiple-model methods. In *IEEE Transactions on Aerospace and Electronic Systems*, volume 41, October 2005.

[11] E. Mazor, A. Averbuch, Yaakov Bar-Shalom, and Joshua Dayan. Interacting multiple model methods in target tracking: a survey. *IEEE Transactions on Aerospace and Electronic Systems*, 34, January 1998. URL https://doi.org/10.1109/7.640267.

[12] Henk A. P. Blom and Yaakov Bar-Shalom. The interacting multiple model algorithm for systems with markovian switching coefficients. *IEEE Transactions on Automatic Control*, 33(8), 1988. URL https://doi.org/10.1109/9.1299.

[13] Daniel P. Lubey. *Maneuver Detection and Reconstruction in Data Sparse Systems with an Optimal Control Based Estimator*. PhD thesis, University of Colorado, 2015.

[14] Marcus Holzinger and Daniel Scheeres. Object correlation and maneuver detection using optimal control performance metrics. September 2010.

[15] Jesse Greaves and Daniel Scheeres. Observation and maneuver detection for cislunar vehicles: Using optical measurements and the optimal control based estimator. *The Journal of the Astronautical Sciences*, 68:826–854, October 2021. URL https://doi.org/10.1007/s40295-021-00283-y.

[16] Andris D. Jaunzemis, Midhun V. Mathew, and Marcus J. Holzinger. Control cost and mahalanobis distance binary hypothesis testing for spacecraft maneuver detection. *Journal of Guidance, Control, and Dynamics*, 39(9):2058–2072, 2016. URL https://doi.org/10.2514/1.G001616.

[17] Jesse A Greaves and Daniel J Scheeres. Relative estimation in the cislunar regime using optical sensors. September 2021.

[18] Jesse A Greaves and Daniel J Scheeres. *Autonomous Navigation for Distributed Space Systems via Spacecraft to Spacecraft Absolute Tracking*. PhD thesis, University of Colorado, 2023.

[19] C. E. Velez and A. J. Fuchs. Averaging techniques and their application to orbit determination systems. January 1974.

[20] W. D. Mcclain. A recursively formulated first-order semianalytic artificial satellite theory based on the generalized method of averaging. volume 1: The generalized method of averaging applied to the artificial satellite problem. Technical report, Computer Sciences Corp., November 1977.

[21] W. D. Mcclain. A recursively formulated first-order semianalytic artificial satellite theory based on the generalized method of averaging. volume 2: The explicit development of the first-order averaged equations of motion and for the nonspherical gravitational and nonresonant third-body perturbations. Technical report, Computer Sciences Corp., May 1978.

[22] Ferdinand Verhulst. *Nonlinear Differential Equations and Dynamical Systems*. Springer Berlin, Heidelberg, 2nd edition, 1996. URL https://doi.org/10.1007/978-3-642-61453-8.

[23] David Gondelach, Roberto Armellin, Hugh Lewis, Alexander Wittig, and Juan San Juan. Semi-analytical propagation with drag computation and flow expansion using differential algebra. 02 2017.

[24] Kirill Suslov, Maksim Shirobokov, and Anastasia Tseloustova. Minimum-energy transfer optimization between near-circular orbits using an approximate closed-form solution. *Aerospace*, 10(12), 2023. ISSN 2226-4310. URL https://doi.org/10.3390/aerospace10121002.

[25] J. Kevorkian and J. D. Cole. *Multiple Scale and Singular Perturbation Methods*, volume 114. Springer New York, 1996. URL https://doi.org/10.1007/978-1-4612-3968-0.

[26] Peter Kahn and Yair Zarmi. Limitations of the method of multiple-time-scales. July 2002. URL https://doi.org/10.48550/arXiv.nlin/0206045.

[27] Javier Roa. *Regularization in Orbital Mechanics: Theory and Practice*, volume 42. Walter de Gruyter GmbH, 1 edition, 2017.

[28] Giulio Baù, Claudio Bombardelli, Jesús Peláez, and Enrico Lorenzini. Non-singular orbital elements for special perturbations in the two-body problem. *Monthly Notices of the Royal Astronomical Society*, 454(3), 10 2015. URL https://doi.org/10.1093/mnras/stv2106.

[29] Davide Amato, Aaron Rosengren, and Claudio Bombardelli. Thalassa: a fast orbit propagator for near-earth and cislunar space. 01 2018. URL https://doi.org/10.2514/6.2018-1970.

[30] Irene Cavallari and Christos Efthymiopoulos. Closed-form perturbation theory in the restricted three-body problem without relegation. *Celestial Mechanics and Dynamical Astronomy*, 2022.

[31] Javier Hernando Ayuso. *Alternative State Representation for Orbit Uncertainty Propagation*. PhD thesis, The University of Tokyo, June 2018.

[32] Javier Hernando Ayuso. Uncertainty propagation in the n-body problem using dromo elements. *Acta Astronautica*, January 2018. URL https://doi.org/10.1016/j.actaastro.2017.12.030.

[33] Jesús Peláez, José Manuel Hedo, and Pedro Rodríguez de Andrés. A special perturbation method in orbital dynamics. *Celestial Mechanics and Dynamical Astronomy*, December 2006. URL https://doi.org/10.1007/s10569-006-9056-3.

[34] Vladimir A. Shefer. Variational equations in parametric variables and transformation of their solutions. *Cosmic Research*, 45, 2007. URL https://doi.org/10.1134/S0010952507040089.

[35] Alessandro Masat, Camilla Colombo, and Arnaud Boutonnet. Kustaanheimo–stiefel - cartesian orbital covariance transformation. *Nonlinear Dynamics*, 113:479–496, September 2024. URL https://doi.org/10.1007/s11071-024-10238-1.

[36] Daniel P. Lubey and Daniel J. Scheeres. Supplementing state and dynamics estimation with information from optimal control policies. In *17th International Conference on Information Fusion (FUSION)*, pages 1–7, 2014.

[37] Daniel P. Lubey, Daniel J. Scheeres, and Richard S. Erwin. Maneuver detection and reconstruction of stationkeeping spacecraft at geo using the optimal control-based estimator. *IFAC-PapersOnLine*, 48(9):216–221, 2015. URL https://doi.org/10.1016/j.ifacol.2015.08.086.

[38] Kalman Filtering for Spacecraft Attitude Estimation. E. j. lefferts and f. l. markley and m. d. shuster. *Journal of Guidance, Control and Dynamics*, 5(5), September 1982. URL https://doi.org/10.2514/6.1982-70.

[39] Mark L. Psiaki. Attitude-determination filtering via extended quaternion estimation. *Journal of Guidance, Control, and Dynamics*, 23, March 2000. URL https://doi.org/10.2514/2.4540.

[40] Dan Simon. *Optimal State Estimation: Kalman, H, and Nonlinear Approaches*. John Wiley Sons, Inc, 2006. URL https://doi.org/10.1002/0470045345.

[41] Charles Acton, Nathaniel Bachman, Boris Semenov, and Edward Wright. A look towards the future in the handling of space science mission geometry. *Planetary and Space Science*, 150, January 2018. URL https://doi.org/10.1016/j.pss.2017.02.013.

[42] T. Fecher, R. Pail, and T. Gruber. Goco05c: A new combined gravity field model based on full normal equations and regionally varying weighting. *Surveys in Geophysics*, January 2017. URL https://doi.org/10.1007/s10712-016-9406-y.

[43] Robert A. Werner and Daniel J. Scheeres. Exterior gravitation of a polyhedron derived and compared with harmonic and mascon gravitation representations of asteroid 4769 castalia. *Celestial Mechanics and Dynamical Astronomy*, 65:313–344, September 1996. URL https://doi.org/10.1007/BF00053511.

[44] S. A. Holmes and W. E. Featherstone. A unified approach to the clenshaw summation and the recursive computation of very high degree and order normalised associated legendre functions. *Journal of Geodesy*, 76:279–299, May 2002. URL https://doi.org/10.1007/s00190-002-0216-2.

[45] Francesco Darugna, Peter Steigenberger, Oliver Montenbruck, and Stefano Casotto. Ray-tracing solar radiation pressure modeling for qzs-1. *Advances in Space Research*, 62, 05 2018. URL https://doi.org/10.1016/j.asr.2018.05.036.

[46] Erwin Fehlberg. Classical fifth-, sixth-, seventh-, and eighth-order runge-kutta formulas with stepsize control. Technical report, NASA, NASA Marshall Space Flight Center Huntsville, AL, United States, October 1968. URL https://ntrs.nasa.gov/citations/19680027281.

[47] Richard P. Brent. *Algorithms for minimization without derivatives*, chapter 4, pages 47–61. Prentice Hall, Englewood Cliffs, N.J., 1973.

[48] Richard L. Burden and J. Douglas Faires. *Numerical Analysis*, chapter 2.1, pages 29–36. Prindle, Weber Schmidt, Boston, 3rd edition, 1985.

[49] Abdelwahab Kharab and Ronald B. Guenther. *An Introduction to Numerical Methods: A MATLAB Approach*. CRC Press, 2nd ed. edition, July 2005.

[50] Doyle T. Hall, Matthew D. Hejduk, and Lauren C. Johnson. Remediating non-positive definite state covariances for collision probability estimation. In *AAS/AIAA Astrodynamics Specialist Conference*. NASA, August 2017. URL https://ntrs.nasa.gov/citations/20170007917.

[51] Paul Anderson, Darren McKnight, Frank Di Pentino, and Hanspeter Schaub. Operational considerations of geo debris synchronization dynamics. In *66th International Astronautical Congress*. International Astronautical Federation, 2015. URL https://api.semanticscholar.org/CorpusID:89613560.

[52] NASA. Goes-r series data book. Technical report, Goddard Space Flight Center, May 2019. URL https://www.goes-r.gov/downloads/resources/documents/GOES-RSeriesDataBook.pdf.

[53] Simon J. Julier and Joseph J. LaViola Jr. On kalman filtering with nonlinear equality constraints. *IEEE Transactions on Signal Processing*, 55, June 2007. URL https://doi.org/10.1109/TSP.2007.893949.

A

Primary Dynamics Solver Validation

As discussed in Chapter 3, a dynamics solver was implemented as part of this thesis for use in the optimal control based estimator. As the Tudat library contains limitations making it unsuitable for use in the OCBE without significant modification, the primary dynamics solver had to be implemented from scratch, including the acceleration models, coordinate transformations, integration scheme, etc.

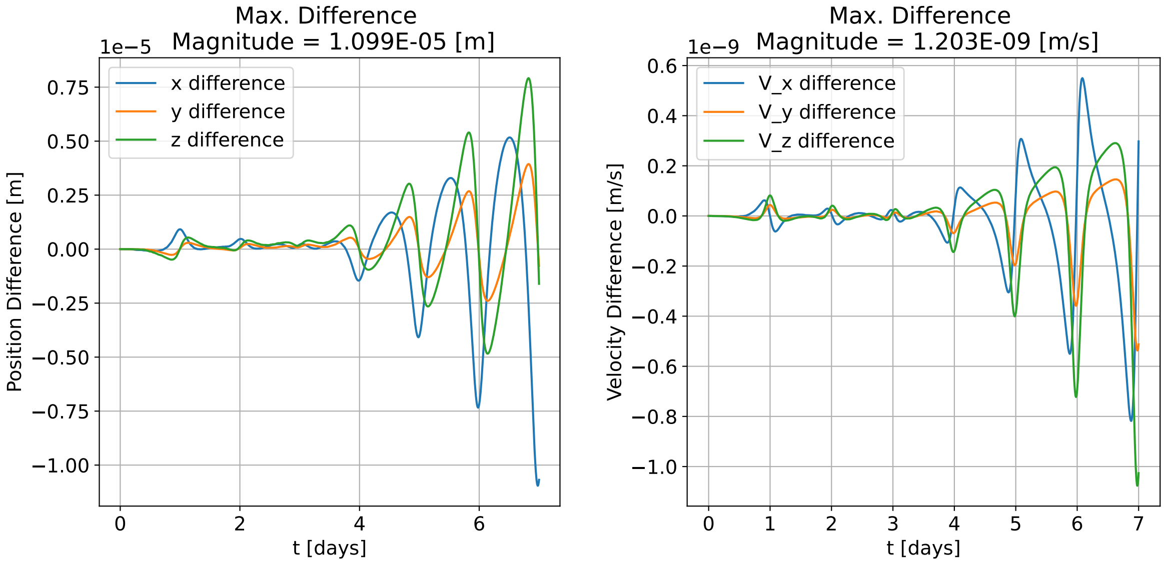
Thus, prior to use of the solver, a large number of tests were performed in order to ensure the solver is accurate to the desired degree. While a significant amount of unit tests for verification of functions were developed, this appendix focuses on key results of broader scale validation testing.

The validation was performed with a number of propagations with the primary dynamics solver, each including a different aspect of the simulated environment. For all validation tests, reference solutions were obtained using an equivalent environment in Tudat. In order to ensure any differences in the solution of the primary dynamics solver and the Tudat variant are not due to differences in the integration scheme, all tests were performed with a fixed-step 7th order RK scheme. This approach also implies the states obtained using both solvers are obtained at the same epochs, hence eliminating the need for interpolation when evaluating differences. It should be noted that while this appendix only presents the results for validation tests for a Tundra orbit, validation tests for all three test cases discussed in the thesis were performed, leading to equivalent conclusions. That being said, in order to validate the Cowell propagator, the following validation tests were performed, each of which is discussed in further detail below:

1. 'Point Mass Central Body' test (propagation with only point-mass gravity of the Earth acting on the spacecraft)
2. 'Perturbing Point Masses' test (propagation with point-mass gravity of the Earth, Moon and the Sun acting on the spacecraft)
3. 'Cannonball SRP' test (propagation with point-mass gravity of the Earth and SRP due to the Sun acting on the spacecraft)
4. 'Earth Centered Inertial to Earth Centered Fixed Spherical frame conversion' test (conversion of every state representation along a pre-propagated orbit)
5. 'Spherical Harmonic Gravity' test (propagation with only spherical harmonic gravity of the Earth acting on the spacecraft)

Tests, #1, #2, #3 were simply performed by propagating an orbit with the acceleration settings relevant to the test and finding a state difference between the implemented 'Primary dynamics solver' and the Tudat equivalent. The results for a Tundra orbit are shown in Figures A.1 through A.3.

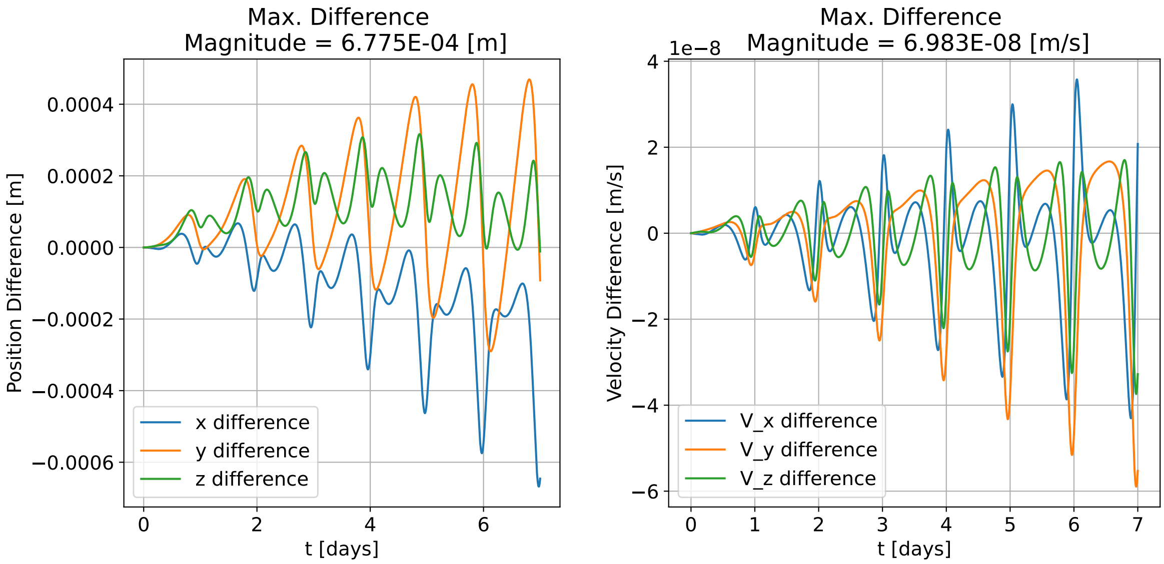
The maximum error w.r.t. a Tudat solution for the first three aforementioned tests is four orders of magnitude below the accuracy requirement for the known dynamics model set in Chapter 6, hence it is considered negligible. It should be noted that while negligible in magnitude, the error does not display discrete behaviour, which implies presence of a source of error different to machine error. While the precise reasoning behind this was not investigated in detail (due to the tiny effect), it may be caused by a number of steps necessary for the propagation, that are not inherently tied to the equations stemming from the physics models used (i.e. different methods to interpolate ephemerides.)



(a) *Position Error*

(b) *Velocity Error*

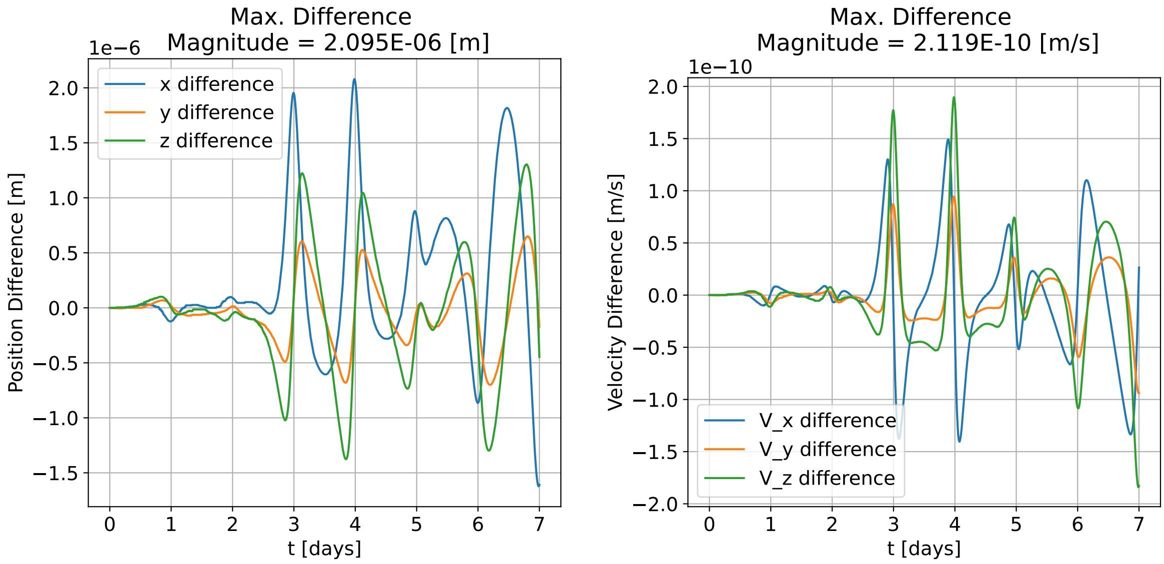
Figure A.1: *Tundra Orbit propagation state error, w.r.t Tudat reference solution, given only point mass of the Earth acting on the spacecraft*



(a) *Position Error*

(b) *Velocity Error*

Figure A.2: *Tundra Orbit propagation state error, w.r.t Tudat reference solution, given point masses of the Earth, Sun and Moon acting on the spacecraft*



(a) *Position Error*

(b) *Velocity Error*

Figure A.3: *Tundra Orbit propagation state error, w.r.t Tudat reference solution, given point mass of the Earth and cannonball solar radiation pressure acting on the spacecraft*

As the spherical harmonic gravity is a more complex perturbation to implement, two validation tests are dedicated to it, First test #4 covers the necessary conversion of a position vector in an 'Earth Centered Inertial' reference frame to a spherical 'Earth Centered Fixed' frame. As Tudat has built in functionality to output the relevant radius, latitude and longitude for a given propagation, a Tundra orbit was propagated again using Tudat, and the conversion results were stored. Then, the custom implemented frame conversion function was applied to the states obtained from the Tudat propagation to obtain new values of radius, latitude and longitude, and a difference was found and is shown in Figure A.4. The radial component obtained with both methods is found to be exactly the same, while the latitude and longitude show discrete differences. This can be attributed to the machine error introduced by inverse trigonometric functions necessary to obtain these components.

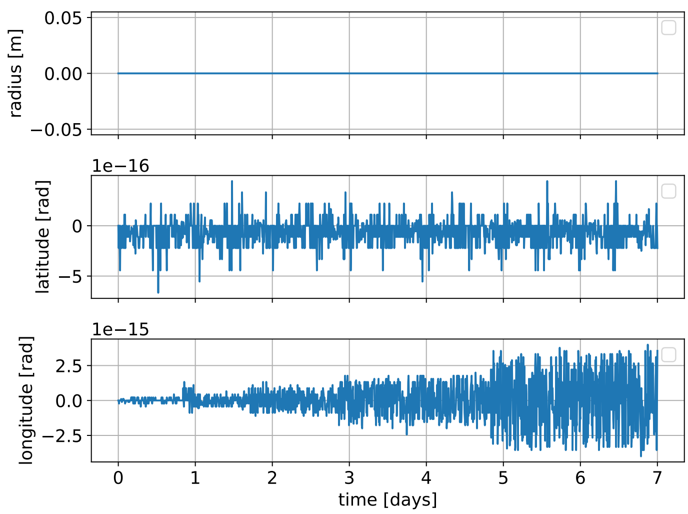
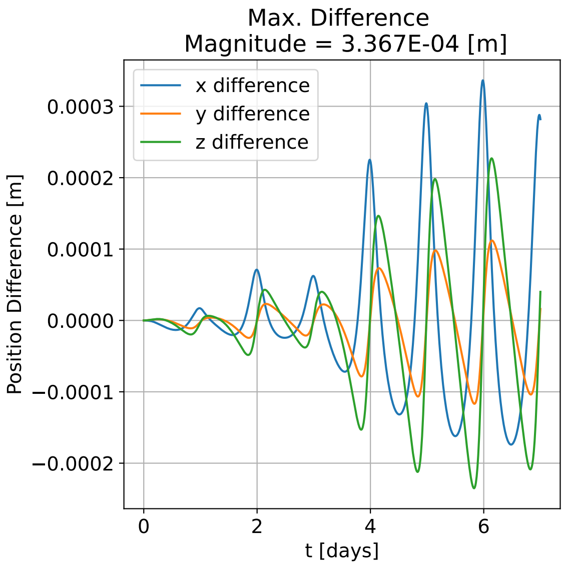


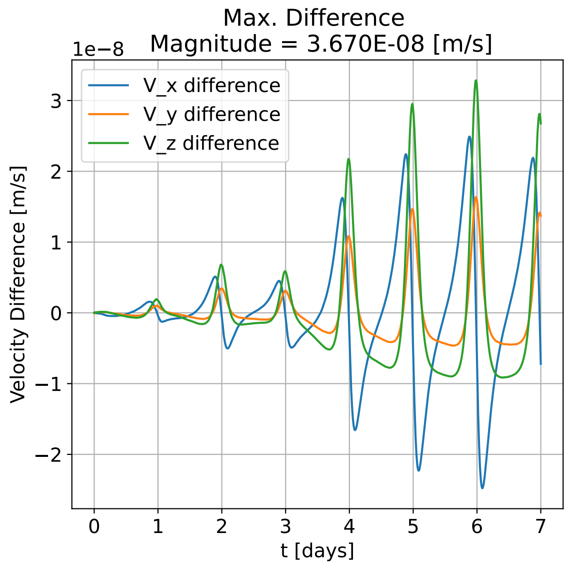
Figure A.4: *Difference in conversion method of 'Earth Centered Inertial' position to 'Earth Centered fixed, spherical' position between Tudat and the implemented primary dynamics solver.*

Then, similar to earlier propagation tests, an orbit is propagated using the full implementation of spherical harmonic gravity using degree and order 20 and 20 respectively, and a difference between the implemented dynamics solver and a Tudat equivalent is found. Similar to the earlier tests, this results in maximum difference in the order of $\mathcal{O}(10^{-4}\,[\mathrm{m}])$; $\mathcal{O}(10^{-8}\,[\mathrm{m/s}])$, as also shown in Figure A.5.

Similarly to the previous tests, the regularized propagator part of the 'Primary Dynamics Solver' was also compared to a Cowell propagation obtained using Tudat to validate its output. However, as fixing a step in terms of time in seconds for the regularized propagator is not trivial (given the propagation is handled in terms of independent variable φ), a different approach was necessary. Instead of using fixed-step solutions as before, variable-step solutions with low tolerances ($\varepsilon_{rel} = 10^{-17}$, $\varepsilon_{abs} = 10^{-13}$) were propagated, despite the knowledge that any possible modelling errors will now be coupled to the truncation error. As it was found that even with low tolerances the time-steps of the regularized propagator are large enough that they are generally not suitable for interpolation, the reference solution was instead interpolated to find differences between the two solvers. The results for a Tundra orbit propagation with aforementioned tolerances are shown in Figure A.6. While this results in larger error than the fixed-step solutions used in previous validation tests, this was anticipated given only 98 steps in time compared to the ~ 1500 used for the fixed-step solutions. Hence, the method was deemed functional and a more detailed tolerance selection follows in Chapter 6.

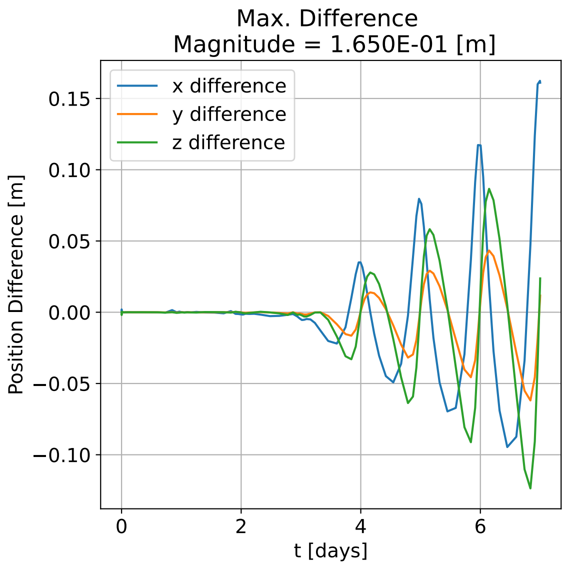


(a) *Position Error*

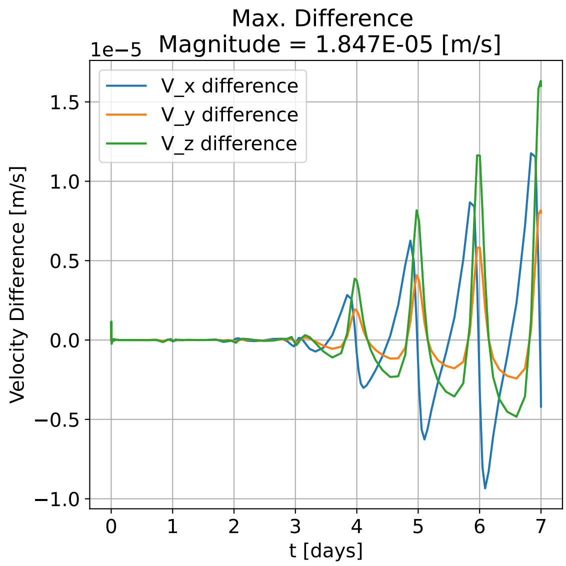


(b) *Velocity Error*

Figure A.5: *Tundra Orbit propagation state error, w.r.t Tudat reference solution, given spherical harmonic gravity of degree 20 and order 20 on the spacecraft*



(a) *Position Error*



(b) *Velocity Error*

Figure A.6: *Tundra Orbit propagation state error (of regularized propagator), w.r.t Tudat reference solution, given spherical harmonic gravity of degree 20 and order 20 on the spacecraft*

B

Thesis Planning Reflection

As part of the requirements for the thesis, a reflection on the planning is provided in this appendix. In the latter half of the literature study, two variants of a thesis planning were created. The first, representing an ambitious ‘best-case’ scenario and a back-up option that would still be sufficient for a thesis, though more limited in scope. A summary of both options is provided in form of a Gantt chart at the end of this appendix, with the ‘best-case’ scenario indicated by the purple and green objectives, while the back-up plan is indicated in yellow.

The main difference between the two planning options is that the best-case planning allowed for implementation of the of a regularized OCBE for the perturbed three body problem in cis-lunar orbits in additional to the geostationary orbit region that is treated as part of this thesis. Regardless of the planning method, an R-OCBE implementation was divided in 6 primary milestones (in addition to documentation of results):

1. Problem definition - Selection of necessary problem parameters (i.e. spacecraft parameters or initial state). It is separated as an independent milestone despite the low workload, as it is a prerequisite for all numerical estimation results.
2. Benchmark Generation - This milestone primarily involves propagation of truth solutions, with quantified error which are necessary for the estimation process later on. Similar to the problem definition milestone, this is a pre-requisite for numerical results.
3. Numerical OCBE implementation - This milestone covers the implementation (in addition to testing and first numerical results) of the A-OCBE and U-OCBE methods based on literature, which may later be used as control groups for the tested R-OCBE.
4. Regularized OCBE derivation - The derivation step involves obtaining all methods necessary to implement the R-OCBE, for example the control input scaling matrix B discussed in Chapter 4. Completion of this milestone implies a conceptually finished algorithm, which, if obtained, implies the method is possible to implement and hence the thesis research questions may be answered.
5. Regularized OCBE implementation - The penultimate milestone involves the implementation and testing of the R-OCBE.
6. Analysis of Results - Finally, analysis of the results covers all comparisons made between the OCBE variants and is separated as the final milestone. With it’s completion all thesis results may be documented.

This method of identifying large scale milestones allowed for easy tracking of pre-requisites for various steps of the thesis, while a bottom-up approach to allocating time to each milestone allowed for rather good approximations of the duration of each task. The majority of the steps were thus completed without significant complications, according to the durations allocated in the ‘alternate’ schedule section of the Gantt chart. However, one step was not originally foreseen in the planning.

Initially, it was desired to use the Tudat library for all dynamics propagation in all implemented variants of the OCBE. However, during the implementation of the A-OCBE, it was found that Tudat does not innately support propagation of state in terms of non-time independent variables, nor the extended STM. As a result, a decision between the following had to be made:

1. implementing these features in Tudat
2. implementing a custom propagator from scratch, allowing for the aforementioned features

Considering the lack of prior knowledge on the back-end of Tudat and lesser experience in C++ programming, it was deemed that implementing a propagator from scratch (with overall reduced features) is the less risky approach, despite a likely lengthier process with additional need for verification and validation. This resulted in a delay of one month and three (work) weeks in order to implement the desired environment models and perform validation of all methods.

Given that after this modification to the schedule there were no significant complications, and that the issue with Tudat was likely unavoidable, the scheduling is still overall deemed a success. That being said, the issue was identified relatively late. A major takeaway is that a schedule with more concurrent milestone progression may have allowed for an earlier schedule amendment.

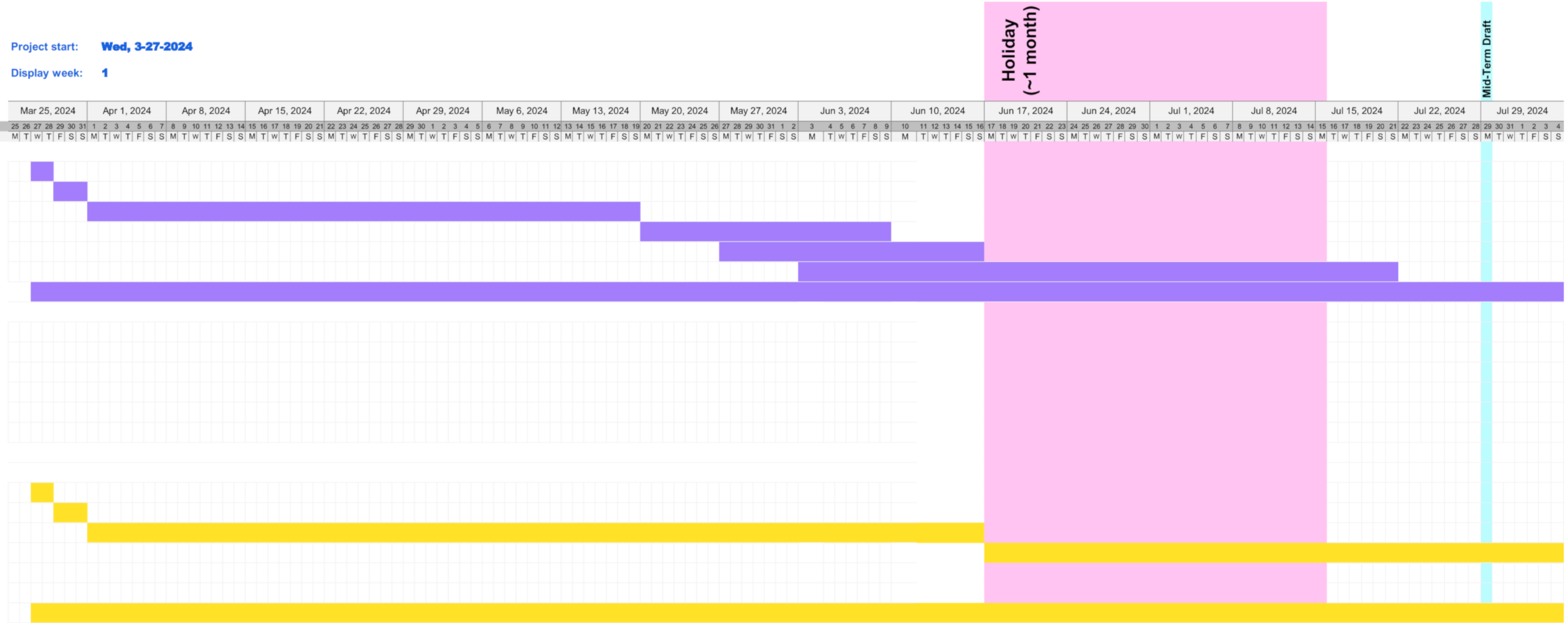
Master's Thesis Plan

Augustas Barysas

Project start: **Wed, 3-27-2024**

Display week: **1**

TASK	PROGRESS	START	END
Best Case Schedule - Geostationary Orbit Case			
Problem Definition	0%	3-27-24	3-28-24
Benchmark Generation	0%	3-29-24	3-31-24
Numerical OCBE Implementation	0%	4-1-24	5-19-24
Regularized OCBE Derivation	0%	5-20-24	6-9-24
Regularized OCBE Implementation	0%	5-27-24	6-16-24
Analysis of Results	0%	6-3-24	7-21-24
Documentation	0%	3-27-24	8-5-24
Best Case Schedule - Cis-lunar Orbit Extension			
Problem Definition	0%	8-6-24	8-7-24
Benchmark Generation	0%	8-8-24	8-10-24
Numerical OCBE Implementation	0%	8-11-24	8-27-24
Regularized OCBE Derivation	0%	8-28-24	9-25-24
Regularized OCBE Implementation	0%	9-11-24	10-9-24
Analysis of Results	0%	9-25-24	10-16-24
Documentation	0%	8-6-24	11-12-24
Alternative Schedule - Geostationary Orbit Case			
Problem Definition	0%	3-27-24	3-28-24
Benchmark Generation	0%	3-29-24	3-31-24
Numerical OCBE Implementation	0%	4-1-24	6-16-24
Regularized OCBE Derivation	0%	6-17-24	8-18-24
Regularized OCBE Implementation	0%	8-19-24	9-15-24
Analysis of Results	0%	9-16-24	10-16-24
Documentation	0%	3-27-24	11-12-24



Master's Thesis Plan

Augustas Barysas

TASK	PROGRESS	START	END
Best Case Schedule - Geostationary Orbit Case			
Problem Definition	0%	3-27-24	3-28-24
Benchmark Generation	0%	3-29-24	3-31-24
Numerical OCBE Implementation	0%	4-1-24	5-19-24
Regularized OCBE Derivation	0%	5-20-24	6-9-24
Regularized OCBE Implementation	0%	5-27-24	6-16-24
Analysis of Results	0%	6-3-24	7-21-24
Documentation	0%	3-27-24	8-5-24
Best Case Schedule - Cis-lunar Orbit Extension			
Problem Definition	0%	8-6-24	8-7-24
Benchmark Generation	0%	8-8-24	8-10-24
Numerical OCBE Implementation	0%	8-11-24	8-27-24
Regularized OCBE Derivation	0%	8-28-24	9-25-24
Regularized OCBE Implementation	0%	9-11-24	10-9-24
Analysis of Results	0%	9-25-24	10-16-24
Documentation	0%	8-6-24	11-12-24
Alternative Schedule - Geostationary Orbit Case			
Problem Definition	0%	3-27-24	3-28-24
Benchmark Generation	0%	3-29-24	3-31-24
Numerical OCBE Implementation	0%	4-1-24	6-16-24
Regularized OCBE Derivation	0%	6-17-24	8-18-24
Regularized OCBE Implementation	0%	8-19-24	9-15-24
Analysis of Results	0%	9-16-24	10-16-24
Documentation	0%	3-27-24	11-12-24