TLE-Based Manoeuvre Detection for Orbit Prediction

Development and Validation of a Satellite Manoeuvre Detection Algorithm Based on TLE Data Consistency Analysis for Orbit Prediction Applications

Guillermo Alejandro Orihuela Gómez

Delft University of Technology

TU Delft

DLR

TLE-Based Manoeuvre Detection for Orbit Prediction

Development and Validation of a Satellite Manoeuvre Detection Algorithm Based on TLE Data Consistency Analysis for Orbit Prediction Applications

by

Guillermo Alejandro Orihuela Gómez

to obtain the degree of Master of Science

at the Delft University of Technology,

to be defended publicly on August 14, 2026.

Student number:	6305482	
Project duration:	November 17, 2025 – August 14, 2026	
Thesis committee:	Dr. S. Gehly,	TU Delft, supervisor
	Prof. Dr. Ir. P.N.A.M. Visser,	TU Delft
	Ir. M.C. Naeije,	TU Delft
	M. Uteg,	DLR

Cover:	Space Shuttle Endeavour Arrival to the ISS during Expedition 22 by NASA (Modified)
Style:	TU Delft Report Style, with modifications by Guillermo Alejandro Orihuela Gómez

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

TU Delft

Dedication

To my parents and brother, for always supporting me and believing in me.

Preface

This thesis was carried out at the German Aerospace Center (DLR), within the Responsive Space Cluster Competence Centre (RSC³) in Trauen, as part of the Master of Science programme in Aerospace Engineering at TU Delft.

The completion of this work marks the end of a remarkable chapter in my academic journey and the beginning of a new stage in my professional career. Looking back, this thesis represents much more than a research project. It has been an opportunity to grow both professionally and personally, to work alongside exceptional people, and to confirm my passion for the field of astrodynamics and space operations.

First and foremost, I would like to express my sincere gratitude to my supervisors, Maurice and Marcus, for their guidance, trust, and continuous support throughout this project. Their expertise, feedback, and willingness to share their knowledge have been invaluable during the development of this work. I would also like to thank Mr. Steve Gehly for his academic supervision and valuable insights during the course of this thesis.

I am deeply grateful to the entire DLR RSC³ team in Trauen for welcoming me from the very first day and for creating such an inspiring and collaborative working environment. In particular, I would like to thank my fellow students and colleagues, who made the process of adapting to a new country, culture, and workplace considerably easier and far more enjoyable.

My sincere appreciation also goes to TU Delft for providing me with the opportunity to study in such an international environment. The experiences, challenges, and opportunities I encountered during my master's programme have opened countless doors and have helped me discover how fascinating and rewarding the space sector truly is.

Working at DLR has been my first real experience in the world of research. It has allowed me to apply the knowledge acquired throughout my studies to real-world problems, while developing new technical and analytical skills. More importantly, it has confirmed that this is the field to which I want to dedicate my professional career. DLR gave me an opportunity when no one else had done so, something I will always value immensely.

Finally, I would like to thank my family and friends. To my parents, my brother, and the rest of my family, thank you for always encouraging me to become a better version of myself and for teaching me never to give up, regardless of the challenges encountered along the way. Your support, confidence, and sacrifices have made all of this possible. To my friends, thank you for your encouragement throughout these years. A special thanks goes to Álvaro, whose friendship and support have been invaluable during my time in Germany.

I hope that the work presented in this thesis contributes, even in a small way, to the advancement of manoeuvre detection and orbit prediction capabilities, and serves as a foundation for future developments in Space Situational Awareness and satellite operations.

Guillermo Alejandro Orihuela Gómez
Hannover, July 2026

Summary

Satellite manoeuvres represent one of the main sources of uncertainty in orbit prediction. While modern orbit-propagation techniques are capable of accurately modelling the nominal evolution of spacecraft trajectories, even relatively small orbital corrections can introduce significant discontinuities that rapidly degrade prediction accuracy when not taken into account. This challenge is particularly relevant within the context of Space Situational Awareness (SSA), where reliable orbit prediction is essential for conjunction assessment, collision avoidance, mission planning, and space traffic management.

The objective of this thesis is to develop and validate a framework capable of detecting satellite manoeuvres, characterizing satellite manoeuvring behaviour, and incorporating manoeuvre awareness into orbit-prediction processes using only publicly available Two-Line Element (TLE) data. The work was conducted within the Improving Satellite Orbit Prediction Accuracy (ISOPA) framework developed at the German Aerospace Center (DLR).

The proposed methodology is based on the analysis of TLE consistency. Multiple historical TLEs are propagated to a common reference epoch and compared against the current orbital state. The resulting residuals are expressed in the Radial-Transverse-Normal (RTN) reference frame and used to construct a set of anomaly scores designed to highlight inconsistencies associated with manoeuvre events. Several score formulations are investigated, ranging from simple propagation-error metrics to innovation-based representations that explicitly exploit information from multiple historical TLEs. Additional refinements include normalization and whitening procedures, as well as machine-learning-based approaches designed to improve the discrimination between nominal and manoeuvre-affected orbital behaviour.

The results demonstrate that meaningful manoeuvre detection and satellite behavioural characterization can be achieved using only publicly available TLE data despite the inherent limitations, noise, and estimation uncertainties associated with these orbital products. The most significant performance improvements are obtained through innovation-based multi-backstep residual representations, which substantially outperform the original baseline formulation. Physically motivated whitening and normalization techniques further improve score stability and threshold transferability across satellites and time periods. Machine-learning refinements provide additional gains, although the results indicate that the majority of the exploitable information is already captured by the underlying physically based score formulations.

The final detection framework achieves high precision and recall in manoeuvre detection while maintaining consistent behaviour across the majority of the analysed satellites. The results further indicate that a significant fraction of the remaining detection errors originate from limitations in the underlying TLE data rather than from deficiencies in the detection methodology itself.

A second major contribution of this thesis is the development of a behavioural-classification framework that characterizes both long-term satellite behaviour and individual manoeuvre events. By analysing the temporal frequency and directional structure of detected manoeuvres, satellites can be grouped according to their operational behaviour and manoeuvring patterns. In addition, each detected manoeuvre can be analysed individually and classified according to its temporal regularity and directional characteristics. This enables the identification of manoeuvres that are consistent with station-keeping activities, while manoeuvres that do not exhibit a sufficiently structured pattern are classified as being of undetermined nature. Although the proposed methodology provides meaningful insight into the likely operational purpose of detected manoeuvres, the true intent behind a manoeuvre cannot be determined with certainty from orbital data alone and ultimately requires information from the satellite operator. Furthermore, the framework can be applied to previously unseen satellites using only their NORAD identifier and the desired analysis period, enabling operational behaviour assessment even in the absence of reference manoeuvre databases.

The final stage of the work focuses on the integration of manoeuvre awareness into the ISOPA pipeline. Detected manoeuvre epochs are used to identify and remove manoeuvre-contaminated samples from the orbit-prediction training process. This manoeuvre-aware strategy consistently improves prediction performance when compared with the original behaviour-agnostic implementation. For several satellites, reductions exceeding 35% in the Root Mean Square (RMS) fourteen-day position prediction error are achieved. These results demonstrate that the value of manoeuvre detection extends beyond event identification itself and can be translated into tangible improvements in operational orbit-prediction performance.

Overall, this thesis demonstrates that TLE consistency analysis provides a viable foundation for manoeuvre detection, behavioural characterization, and manoeuvre-aware orbit prediction. The developed framework successfully combines physically motivated modelling techniques with machine-learning refinements and establishes a practical pathway towards more robust and reliable orbit-prediction systems for future Space Situational Awareness applications.

Contents

List of Figures

List of Tables

Nomenclature

Abbreviations

Abbreviation	Definition
AE	Autoencoders
APC	Analytical Continuation Power Series
BLS	Batch Least Squares
CNES	French Space Agency
CNN	Convolutional Neural Network
CPF	Consolidated Prediction Format
CRYO2	Cryosat-2
DBSCAN	Density-Based Spatial Clustering of Applications with Noise
DLR	Deutsches Zentrum für Luft- und Raumfahrt
DNN	Deep Neural Network
DSST	Draper Semi-analytical Satellite Theory
EKF	Extended Kalman Filter
EME2000	Earth Mean Equator and Equinox of J2000.0
ENVI1	ENVISAT
ERM	Empirical Risk Minimization
FN	False Negative
FP	False Positive
FP	Finalization Phase
GAN	Generative Adversarial Network
GEO	Geostationary Orbit
GLP	Green Light Phase
GMM	Gaussian Mixture Model
GNSS	Global Navigation Satellite System
GP	Gaussian Process
GPR	Gaussian Process Regression
GRU	Gated Recurrent Unit
GSOC	German Space Operations Center
GT	Ground Truth
HAC	Hierarchical Agglomerative Clustering
HDBSCAN	Hierarchical Density-Based Spatial Clustering of Applications with Noise
IGS	International GNSS Service
iGMAS	International GNSS Monitoring and Assessment System
IGSO	Inclined Geostationary Orbit
ILRS	International Laser Ranging Service
IOD	Initial Orbit Determination
ISA	International Standard Atmosphere
ISOPA	Improving Satellite Orbit Prediction Accuracy
ITRF	International Terrestrial Reference Frame
JASO1	Jason-1
JASO2	Jason-2
JASO3	Jason-3
KF	Kalman Filter

Abbreviation	Definition
LEO	Low Earth Orbit
LRP	Literature Review Phase
LSTM	Long-Short Term Memory
MEO	Medium Earth Orbit
ML	Machine Learning
MLP	Multi-Layer Perceptron
MTP	Mid-term Phase
MWCF	Moving-window Curve-Fitting
NIS	Normalized Innovation Squared
OD	Orbit Determination
OP	Orbit Prediction
PDF	Probability Density Function
PNT	Position, Navigation and Timing
RF	Random Forest
RMS	Root Mean Square
RNN	Recurrent Neural Network
RSC³	Responsive Space Cluster Competence Center
RSO	Resident Space Object
RTN	Radial-Transverse-Normal
SAT	Semi-Analytical Satellite Theory
SEN3A	SENTINEL-3A
SEN3B	SENTINEL-3B
SEN6A	SENTINEL-6 Michael Freilich
Seq2Seq	Sequence-to-Sequence
SGP	Simplified General Perturbations
SLR	Satellite Laser Ranging
SNR	Signal-to-Noise Ratio
SSA	Space Situational Awareness
SRP	Solar Radiation Pressure
STF	Strong Tracking Filter
SVM	Support Vector Machine
SWOT1	SWOT
TCN	Temporal Convolutional Network
TCC	TLE Consistency Checking
TEME	True Equator, Mean Equinox
TLE	Two-Line Element
TP	True Positive
TTSA	TLE Time-Series Analysis
UKF	Unscented Kalman Filter
WP	Work Package
XGBoost	eXtreme Gradient Boosting
MSc	Master of Science
TU Delft	Delft University of Technology

Symbols

Symbol	Nomenclature
Δv	Manoeuvre impulse
r	Position vector
v	Velocity vector
TLE_i	Target TLE with index i
t_i	Epoch of TLE_i (target epoch)
k	Backstep index

Symbol	Nomenclature
K	Total number of backsteps ($K = 7$)
$\mathbf{x}_{i-k\to i}$	State obtained by propagating TLE_{i-k} to epoch t_i
$\mathbf{x}_i^{\mathrm{ref}}$	Reference state of the target TLE at its own epoch
$\Delta\mathbf{x}_{i-k}$	State residual for backstep k
$\Delta\mathbf{r}_{i-k}$	Position residual for backstep k
$\Delta\mathbf{v}_{i-k}$	Velocity residual for backstep k
$\Delta R,\ \Delta T,\ \Delta N$	Radial, transverse and normal position residual components
$\Delta V_R,\ \Delta V_T,\ \Delta V_N$	Radial, transverse and normal velocity residual components
$\|\Delta\mathbf{r}\|,\ \|\Delta\mathbf{v}\|$	Norms of the position and velocity residuals
$x,\ y,\ z$	Position components in EME2000
$\dot{x},\ \dot{y},\ \dot{z}$	Velocity components in EME2000
$\mathbf{h}$	Orbital angular momentum vector
$\hat{\mathbf{e}}_R,\ \hat{\mathbf{e}}_T,\ \hat{\mathbf{e}}_N$	Radial, transverse and normal unit vectors of the RTN triad
$\mathbf{C}_{\mathrm{EME2000}\to\mathrm{RTN}}$	Rotation matrix from EME2000 to RTN
$\mathbf{f}_i$	Feature vector at target epoch t_i
N_f	Dimension of the feature vector ($N_f = 63$)
y_i	Binary manoeuvre label at epoch i (evaluation only)
S_i	Scalar anomaly score at epoch i
$g(\cdot)$	Scoring function mapping features to a scalar score
q_s	Upper-envelope residual statistic of satellite s (passive test)
$Q_p(\cdot)$	p-th percentile operator
$\mathcal{V}_{\mathrm{val}}^{(s)}$	Set of validation residuals of satellite s
s	Satellite identifier
$r_{i,k}$	Scalar position-consistency feature at epoch t_i for backstep k
$S_i^{(1)}$	Baseline consistency score
$\hat{r}_i$	Median predictor over the older backsteps
u_i	Scalar back-innovation
MAD	Median absolute deviation
$\mathrm{MAD}_{\mathrm{fit}}$	MAD estimated on the fit segment
$S_i^{(2)}$	Median back-innovation score
$\mathbf{v}_i$	Innovation vector across older backsteps
$\tilde{\mathbf{v}}_i$	Robustly whitened innovation vector
$\tilde{v}_{i,k}$	k-th component of the whitened innovation vector
$\mathrm{med}_k,\ \mathrm{MAD}_k$	Component-wise median and MAD over the fit subset
$\mathrm{clip}(\cdot)$	Clipping operator
$z_{\max}$	Clipping level for normalized innovations
$S_i^{(3,\mathrm{raw})}$	Raw whitened vector innovation score
w_k	Weight assigned to older backstep k
p_w	Weighting exponent of the backstep weights
ϕ_i	Consensus fraction of backsteps above the significance threshold
$\mathbb{I}(\cdot)$	Indicator function
z_0	Significance threshold for the consensus factor
γ	Exponent of the consensus factor
$S_i^{(4,\mathrm{raw})}$	Raw tuned whitened vector innovation score
$S_i^{(4)}$	Tuned whitened vector innovation score

Symbol	Nomenclature
S_i^{phys}	Physical score used as baseline for the learning-based stage
s_i^{ML}	Probability-like output of the learning-based model
c_i^{ML}	Bounded multiplicative correction from the learning-based model
κ	Amplitude of the learning-based correction
span	Scale parameter of the learning-based correction
α_{ML}	Conservativeness parameter of the learning-based correction
$S_i^{(5)}$	ML-corrected whitened vector score
m_i	Strictly past rolling median of the physical score
W	Length of the backward (strictly past) window
σ_i	Robust scale estimate from the rolling MAD
z_i^{onset}	Standardized onset excursion of the score
p_i^{onset}	Onset prominence
β_{onset}	Strength of the onset correction
c_i^{onset}	Multiplicative onset correction factor
$S_i^{(6)}$	Onset-sensitive score
d_i	Binary detection variable at epoch i
τ_{high}	Upper (initiation) detection threshold
τ_{low}	Lower (maintenance) detection threshold
α	Hysteresis factor relating both thresholds
p	Threshold percentile used to derive τ_{high}
K_{enter}	Number of samples required to initiate an event
K_{confirm}	Number of samples required to confirm an event
$\Delta t_{\mathrm{merge}}$	Maximum separation for merging ground-truth events
t_{GT}^*	Representative epoch of a merged ground-truth event
$t_{\mathrm{pred}}^{\mathrm{start}},\ t_{\mathrm{pred}}^{\mathrm{end}}$	Start and end times of a predicted event
Δt_{snap}	Maximum forward shift allowed in the snap-forward rule
N_{GT}	Number of ground-truth events
N_{PRED}	Number of predicted events
N_{match}	Number of successful one-to-one matches
TP	True positives
FP	False positives
FN	False negatives
$\mathrm{Precision}$	Ratio of correct detections to total detections
Recall	Ratio of correct detections to total ground-truth events
F_β	Weighted harmonic mean of precision and recall
β	Weighting parameter of the F_β score
F_2	F_β score with $\beta = 2$, used for model selection
$\tau_{\mathrm{high}}^{(s)}$	Upper detection threshold of satellite s
$\tilde{x}$	Percentile-scaled (normalized) metric
$x^{(5)},\ x^{(95)}$	5th and 95th percentiles of a metric across configurations
$F_2^{\mathrm{high-perf}}$	F_2 score on the high-performing satellite subset
$F_2^{\mathrm{low-perf}}$	F_2 score on the low-performing satellite subset
$\mathcal{J}$	Weighted objective function maximized during optimization
C	Cluster of observations

Symbol	Nomenclature
n_C	Number of observations in cluster C
$\mathbf{o}_i$	Observation (feature vector) used for clustering
$\boldsymbol{\mu}_C$	Centroid of cluster C
$WCSS(C)$	Within-cluster sum of squares of cluster C
$\Delta(A, B)$	Increase in within-cluster variance when merging clusters A and B
Δt_i	Temporal spacing between consecutive manoeuvres
N	Number of manoeuvre events
$\mu_{\Delta t}$	Mean inter-manoeuvre interval
$\sigma_{\Delta t}$	Standard deviation of the inter-manoeuvre intervals
CV	Coefficient of variation of the inter-manoeuvre intervals
F	Frequency index
$\mathbf{p}_i$	Normalized directional vector of manoeuvre event i
$p_{R,i},\ p_{T,i},\ p_{N,i}$	Fractions of manoeuvre activity along the RTN axes
$a_{R,i},\ a_{T,i},\ a_{N,i}$	Per-axis activity magnitudes of manoeuvre event i
$\bar{\mathbf{p}}$	Mean manoeuvre direction
θ_i	Angle between $\mathbf{p}_i$ and the mean manoeuvre direction
D	Directionality index

1

Introduction

The rapid growth of satellites in Earth orbit has made the space environment increasingly crowded and operationally demanding. Both Low Earth Orbit (LEO) and Geostationary Orbit (GEO) host large numbers of active spacecraft, with LEO containing the vast majority of satellites and GEO hosting several hundred operational spacecraft [1]. This number continues to rise as commercial and institutional activity expands. Fragmentation events, particularly from explosions of derelict spacecraft and upper stages, have historically been a major source of debris, while accidental collisions, although less frequent, remain a critical contributor due to their high debris-generation potential. As a result, maintaining up-to-date and accurate catalogues has become essential for safe and sustainable operations. Because orbit prediction errors can accumulate quickly and compromise collision avoidance or planning activities, reliable orbit determination (OD) and propagation frameworks are critical.

Traditional orbit prediction relies on physical force models and numerical or semi-analytical propagators. These approaches model the main perturbations, such as atmospheric drag, solar radiation pressure (SRP), Earth's geopotential, and third-body effects, but face inherent limitations: atmospheric density uncertainty, SRP variability, measurement noise, unmodelled forces, and biases in data products such as Two-Line Elements (TLEs). While these models perform reasonably well over short time scales, their accuracy deteriorates rapidly, especially in LEO where environmental variability is greatest [2].

An alternative approach consists of complementing conventional orbit propagation models with data-driven correction techniques. Instead of explicitly modelling every perturbation source, Machine Learning (ML) methods learn systematic residual patterns between predicted and reference trajectories and use them to improve propagation accuracy. Following this philosophy, the Improving Satellite Orbit Prediction Accuracy (ISOPA) framework [3], developed by the Ground Segment of the Deutsches Zentrum für Luft- und Raumfahrt (DLR) Responsive Space Cluster Competence Center (RSC³), learns residuals between Simplified General Perturbations (SGP4)-propagated Two-Line Element (TLE) states and subsequently released TLE updates. These residuals are then used to correct future predictions, yielding significant improvements in orbit prediction accuracy.

However, current ML-based systems, including ISOPA, implicitly assume smooth residual behaviour driven only by model imperfections. In reality, many satellites perform station-keeping, attitude, orbit-maintenance or collision avoidance manoeuvres that introduce abrupt trajectory changes. These discontinuities severely degrade both classical and ML predictions if manoeuvres are not detected, as the algorithms cannot distinguish manoeuvring from nominal behaviour.

To address this, the present thesis develops a manoeuvre-detection and behaviour-classification module that identifies anomalous orbital changes using historical orbital data. The method classifies satellites according to their long-term manoeuvring behaviour and characterizes individual manoeuvres based on their operational nature. Manoeuvres performed to maintain a prescribed orbital configuration are distinguished from other types of orbital adjustments. The TLE epochs affected by the detected manoeuvres are subsequently removed from the ISOPA training database, preventing manoeuvre-contaminated observations from degrading orbit prediction performance.

The thesis is structured as follows. Chapter 2 presents the relevant information from the existing literature. Chapter 3 identifies the gap that the thesis aims to fill and presents the research questions. Chapter 4 describes the methodology used to achieve the objectives described. Chapter 5 presents the results after the evaluation of the pipeline. Chapter 6 presents the conclusions of the whole project and answers the research questions.

2

Literature Review

2.1. Orbit prediction in the context of Space Situational Awareness

2.1.1. Role of orbit prediction

Space Situational Awareness (SSA) refers to the capability to detect, track, characterize, and predict the behaviour of objects and events in the space environment in order to ensure the safe and sustainable use of outer space [4]. Two core capabilities of SSA are orbit determination, which estimates the current state of Resident Space Objects (RSOs) from observations, and orbit prediction, which propagates these estimated states into the future. Orbit prediction therefore enables catalogue maintenance, conjunction assessment, and collision avoidance. To support these operations, prediction frameworks must deliver reliable and frequently updated orbital states. However, widely used catalogue propagators (such as SGP4) often rely on simplified drag, gravity, and SRP models, causing prediction errors to grow rapidly with propagation time, especially in LEO [5].

As the number of tracked objects in Earth orbit continues to grow, operational centres must maintain increasingly large catalogues and generate a greater number of orbit predictions every day. Figure 2.1 illustrates the steady increase in the population of tracked space objects, which places a growing burden on orbit determination and catalogue maintenance systems. Since observations are not continuously available, the true state of many RSOs must be estimated by propagating their most recent known state between measurement updates. Therefore, maintaining the same observation cadence for every object becomes increasingly challenging. This results in longer intervals between measurements and a greater dependence on orbit propagation to estimate future states. Prediction errors accumulated during these intervals can significantly degrade state estimation accuracy, increasing the need for precise and computationally efficient orbit prediction methods [6, 7].

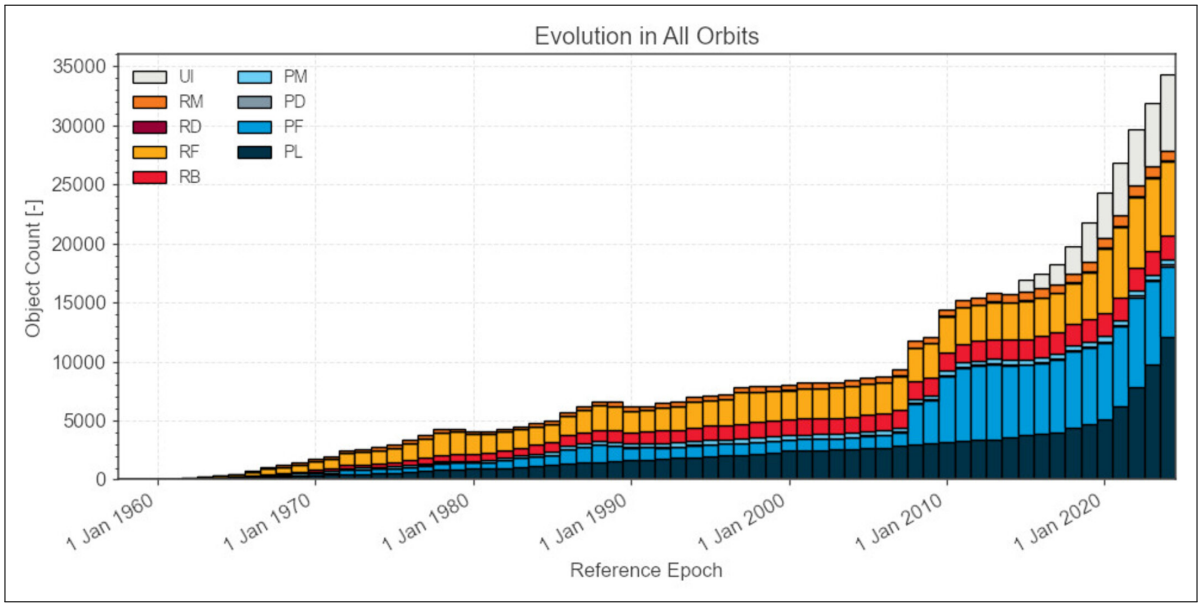

Figure 2.1: Evolution of the number of tracked space objects in Earth's orbit by object type [1].

Since the early days of spaceflight, multiple orbit prediction frameworks have been developed to address these challenges [8]. Early analytical methods captured only dominant perturbations, enabling low-resolution long-term estimates but performing poorly in strongly perturbed regimes such as LEO [9]. As understanding of orbital dynamics improved, more advanced numerical and semi-analytical techniques were introduced. Among these, Semi-Analytical Satellite Theory (SAT) combines analytical formulations of the dominant orbital perturbations with numerical integration of slowly varying orbital elements, providing a compromise between computational efficiency and prediction accuracy. SAT-based propagators have been widely adopted for catalogue maintenance, orbit determination, and large-scale space surveillance applications, where the propagation of thousands of objects must be performed efficiently [10]. However, their accuracy still depends strongly on force-model fidelity and the quality of tracking data.

To better understand the limitations and capabilities of these approaches, the following subsections review orbit prediction (OP), orbit determination (OD), and filtering methods in greater detail.

2.1.2. Classical orbit prediction frameworks

Classical orbit prediction methods aim to estimate the future state of a satellite by propagating a known initial state under a prescribed force model. Depending on how the equations of motion are represented and solved, orbit propagators are commonly classified into three categories: analytical, semi-analytical, and numerical methods [8]. Each approach offers different trade-offs between accuracy, computational cost, and robustness.

Analytical and semi-analytical propagators use explicit mathematical formulations of perturbing forces to model satellite motion. Analytical propagators, such as the SGP family (particularly SGP4) compute trajectories from closed-form expressions based on simplified dynamics. They are extremely fast and suitable for large-scale catalogue operations. Nevertheless, their truncated perturbation models limit long-term accuracy, especially in strongly perturbed regimes [9].

Semi-analytical methods bridge analytical and numerical approaches by expressing the equations of motion in orbital elements and averaging out short-period perturbations, while in some formulations also partially accounting for long-period effects. This allows the long-term orbital evolution to be propagated more efficiently [2]. Perturbations are incorporated through analytical or semi-empirical models, while the resulting mean-element equations are integrated numerically. These approaches typically rely on a truncated dynamical model, for example by retaining only a limited number of spherical harmonic terms in the geopotential expansion. Frameworks such as HEOSAT [11], a mean-element orbit propagator for

highly elliptical orbits developed by Lara, San-Juan, and Hautesserres, follow this philosophy, but their prediction accuracy remains strongly dependent on the fidelity of the force models used to represent atmospheric drag, solar radiation pressure, and other perturbations.

Numerical propagators solve Newton's equations of motion using full, high-fidelity force models, making them the most accurate class of classical orbit prediction techniques. Their performance depends on the integrator formulation over computed states and used steps. These choices lead to trade-offs between accuracy, robustness, and computational cost.

Atallah et al. compare several numerical integrators across different orbital regimes and show that accuracy is highly sensitive to the method used, drag, and model inaccuracies [12]. Additional studies reinforce these conclusions: Montenbruck finds that single- and multi-step integrators have comparable efficiency in many scenarios, while multi-step schemes excel when dense ephemerides are required [13]. Papanikolaou et al. show that performance varies substantially across LEO regimes, especially under drag [14]. Alternative formulations based on non-Cartesian element sets have also been proposed to improve the efficiency of numerical propagation. For example, Baù et al. introduced the EDromo propagator, which employs a regularized set of orbital elements and demonstrates superior accuracy-to-cost performance compared to several established propagation schemes, particularly for highly eccentric and perturbed orbits [15]. More advanced techniques, such as Sobolev polynomial-based integrators, have demonstrated substantial accuracy improvements over conventional numerical schemes, achieving up to 72 times lower propagation errors than adaptive Gaussian integration and more than two orders of magnitude improvements over commonly used Runge–Kutta methods [16]. However, these gains often come at the expense of increased computational cost, which has so far limited their widespread operational adoption.

2.1.3. Classical orbit determination and state estimation frameworks

Orbit determination is the process of estimating the current state of a spacecraft from tracking observations. Unlike orbit prediction, which propagates a known state into the future, orbit determination combines measurements with a dynamical model to reconstruct the most probable state and associated uncertainties. Since measurements are available only intermittently, all orbit determination methods rely on orbit propagation models to relate observations acquired at different epochs.

Estimation-based orbit determination frameworks combine a dynamical model with observational data to estimate a satellite's state and its uncertainties, effectively rectifying a nominal dynamical prediction through measurement updates. Two main estimator families dominate: Batch Least Squares (BLS) and Kalman Filter (KF).

Batch Least Squares processes measurements simultaneously to obtain a globally optimal solution, offering high accuracy and robustness to noise and data gaps. When combined with high-quality Global Navigation Satellite System (GNSS) data, it can achieve great precision [17]. However, its latency and computational cost limit its suitability for real-time or large-scale catalogue operations.

Kalman Filters, including Extended KF (EKF) and Unscented KF (UKF), update the state recursively as new observations arrive, making them well suited for real-time navigation. They can accommodate some nonlinearities and modelling errors [18], but their performance is highly sensitive to the quality of the initial conditions and to the assumed process and measurement noise statistics [19].

Hybrid estimators combine the strengths of both approaches. Batch-sequential filters integrate batch solutions into recursive updates to improve stability in nonlinear regimes [20], while unscented-transformation batch filters improve robustness in strongly perturbed environments [21]. Recent Bayesian-adaptive and second-order KF variants further improve resilience to modelling errors and measurement uncertainty, particularly during manoeuvres [22]. In all cases, these estimators improve performance by repeatedly using observations to rectify the evolving state. The resulting state estimates are subsequently incorporated into operational catalogues and are often distributed in the form of Two-Line Elements (TLEs). Since the manoeuvre detection algorithm developed in this work operates exclusively on TLE data, the quality and update frequency of these orbit determination solutions directly influence its performance.

2.1.4. Limitations of classical orbit prediction and determination frameworks

Despite decades of development, classical orbit prediction and determination frameworks still face fundamental limitations that constrain their reliability in modern environments. These limitations arise from imperfect environmental modelling [5, 8, 23], incomplete knowledge of RSO physical properties [24, 25], intermittent and noisy tracking data [6, 18, 26], and the inability of traditional propagators to accommodate unannounced manoeuvres or anomalous behaviour [7, 17]. As these uncertainties accumulate, prediction accuracy deteriorates rapidly, particularly in LEO [12, 27], where orbital dynamics are strongly influenced by atmospheric drag and highly variable thermospheric density driven by space weather conditions. These effects introduce significant time-varying perturbations that are difficult to model accurately and lead to faster divergence of propagated states compared to higher-altitude regimes. In operational settings, scalability constraints further require trade-offs between accuracy and computational cost [21, 28].

A concise overview of the main limitations of classical frameworks and their qualitative impact on orbit prediction is provided in Table 2.1. Together, these challenges motivate the development of data-driven approaches, such as ISOPA, that can learn object-specific residual patterns and improve prediction performance beyond what classical models can achieve. These methods will be further investigated in the following section.

Limitation	Impact
Incomplete environment modelling	Systematic propagation errors that grow with prediction horizon.
Unknown object properties	Persistent modelling biases affecting both state estimation and prediction.
Sparse or noisy observations	Reduced orbit determination accuracy, leading to poor initial state estimates.
Propagation of initial-state and model errors	Errors in all position and velocity components increase with propagation time, with along-track errors typically growing the fastest.
Unmodelled manoeuvres or anomalous behaviour	Large and often unpredictable deviations between predicted and actual trajectories.

Table 2.1: Main limitations of classical orbit determination and orbit prediction frameworks.

2.1.5. ML-based and hybrid orbit prediction

Classical orbit prediction frameworks are constrained by the limitations shown in Table 2.1. Several ML-based enhancement approaches have been shown to reduce these shortcomings by learning systematic residual errors of classical propagators and by modelling correction terms that capture effects that are difficult to represent explicitly [29, 30, 31, 32]. Such methods have also been demonstrated to improve prediction performance when tracking data are sparse, thereby extending the practical validity horizon compared to purely classical propagation [29, 30, 33]. More information about ML theory can be found in Appendix B.

Various data-driven architectures have been explored: recurrent neural networks, including Long-Short Term Memories (LSTMs), Gated Recurrent Units (GRUs), and Temporal Convolutional Networks (TCNs) variants, often outperform purely physics-based propagators [30, 33, 34, 35]. Classical ML approaches such as Support Vector Machines (SVMs) and Gaussian Processes (GPs) have also been used to correct TLE-based predictions, showing strong robustness under noisy or sparse tracking conditions [30, 31]. More recent studies investigate feature-based models such as Random Forests (RFs) [36].

In most of the cases, hybrid physics–ML approaches are used. They combine classical propagation with data-driven correction, addressing limitations that neither method can resolve on its own. In these frameworks, a traditional propagator (e.g., SGP4) provides a baseline trajectory, while an ML model learns and compensates for the resulting systematic residuals. This allows the ML component to capture unmodelled perturbations while preserving the physical consistency of the underlying model [29, 37]. Such hybrid error-correction schemes have shown good accuracy improvements across LEO and MEO [30, 33]. DLR RSC^3's ISOPA [3] system follows this approach by learning residuals from TLE-based predictions.

More advanced designs integrate ML directly into the estimation process, for example through hybrid neural-network/Kalman-Filter architectures that use learned corrections to improve state estimation and trajectory prediction [38]. Similar ideas have also been explored through deep-learning and ensemble-learning approaches for orbit-error prediction and correction [39, 40].

Despite their demonstrated accuracy gains, ML-based orbit prediction frameworks still face limitations in certain operational scenarios. Most challenges arise not from ML itself, but from the characteristics of SSA training data. Purely data-driven models depend on historical residual patterns, so sparse or noisy tracking can reduce generalisation and increase the risk of overfitting [24, 32]. ML predictors also extrapolate poorly to objects or orbital regimes not represented in the training set [41]. Furthermore, while prediction improvements can be quantified straightforwardly through aggregate performance metrics, identifying the root causes of degraded performance is often considerably more difficult. A model may achieve significant accuracy gains for most objects in a catalogue while performing poorly for a small subset, making it challenging to determine whether the degradation originates from data quality issues, unmodelled dynamical effects, insufficient training samples, or limitations of the learning architecture itself.

An overview of all orbit prediction approaches (classical and ML-based) is provided in Table 2.2.

Approach	Data required	Advantages	Disadvantages	Typical use
Analytical	Initial orbital elements	Very fast, catalogue-wide propagation	Limited force-model fidelity, lower long-term accuracy	Large catalogues
Semi-analytical	Initial state + simplified force models	Good balance between accuracy and efficiency	Dependent on force-model quality	SSA and catalogue maintenance
Numerical	Initial state + high-fidelity force models	Highest physical accuracy	Computationally expensive	High-fidelity orbit prediction
Hybrid physics–ML	Classical propagator + historical residuals	Improved accuracy through learned corrections	Requires representative training data and retraining	Modern SSA

Table 2.2: Comparison of classical and ML-enhanced orbit prediction approaches.

Another limitation emerges during uncooperative manoeuvres or abrupt dynamical changes. Existing ML propagators are trained on nominal behaviour and therefore cannot anticipate trajectory shifts, causing prediction errors to grow rapidly [33]. Incorporating a dedicated manoeuvre detection and event behaviour classification module addresses this gap by enabling ML propagators to re-initialise or adapt their strategy when the satellite transitions to a new dynamical state [42]. The creation of such a module will be investigated in this thesis.

The following figures illustrate the effect of an apogee-raising manoeuvre on orbital evolution. Prior to the manoeuvre, both trajectories are identical and share the same position and velocity state. The impulsive velocity increment modifies the orbital energy and geometry, resulting in a new orbital trajectory with different orbital elements. As the spacecraft progresses along its orbit, the effect of this perturbation accumulates, causing the separation between the nominal and manoeuvred trajectories to grow. Consequently, even a relatively small manoeuvre can lead to significant deviations after propagation. Figure 2.2 depicts the resulting change in orbital geometry, while Figure 2.3 shows the evolution of the position-vector difference between both trajectories as a function of true anomaly. The latter illustrates how the divergence increases as the spacecraft moves away from the manoeuvre location and the differences in orbital shape become more pronounced. If an orbit propagator is unaware that a manoeuvre has been performed, it will predict an orbit shape which matches the previous trajectory, diverging with respect to the actual one.

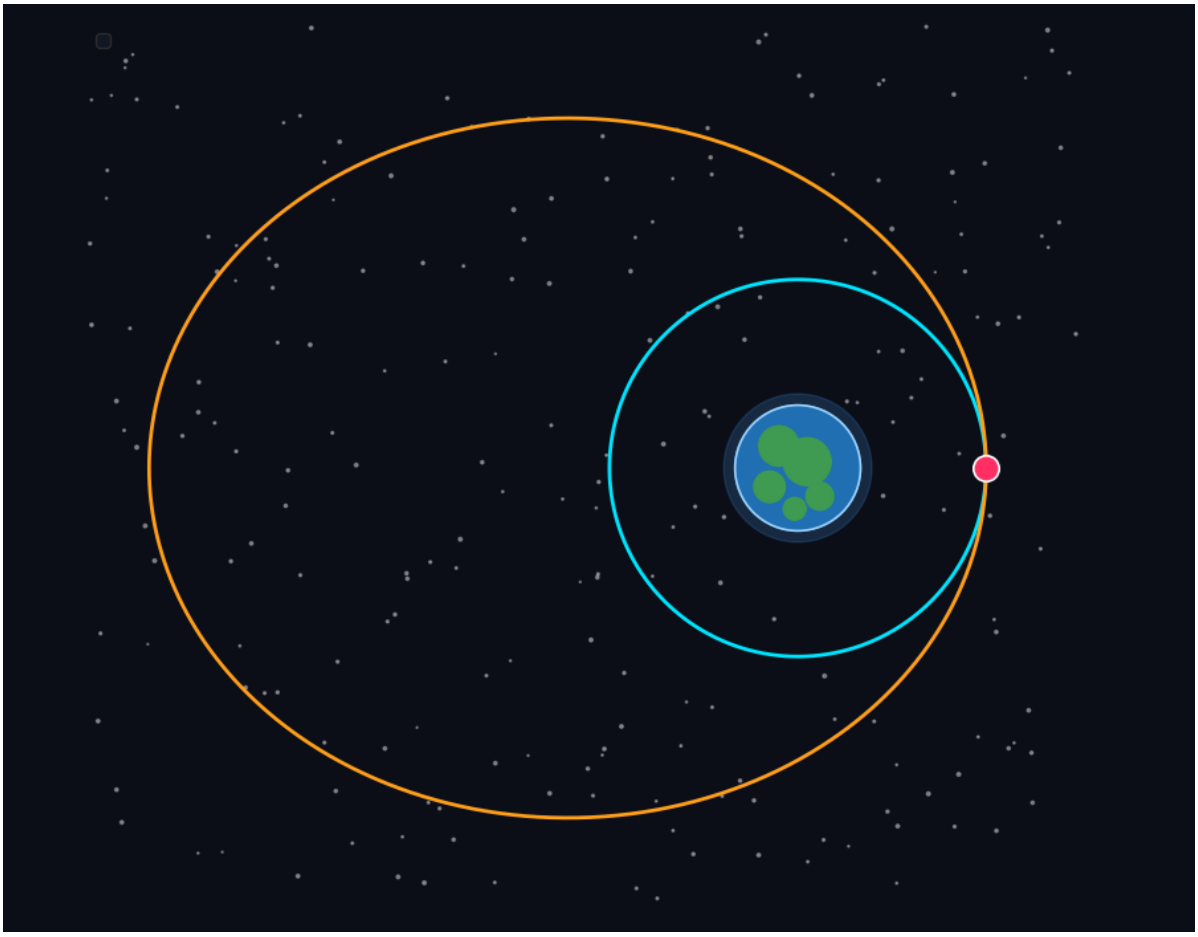

Figure 2.2: Orbit shape before and after a manoeuvre has been performed (manoeuvre point=red dot).

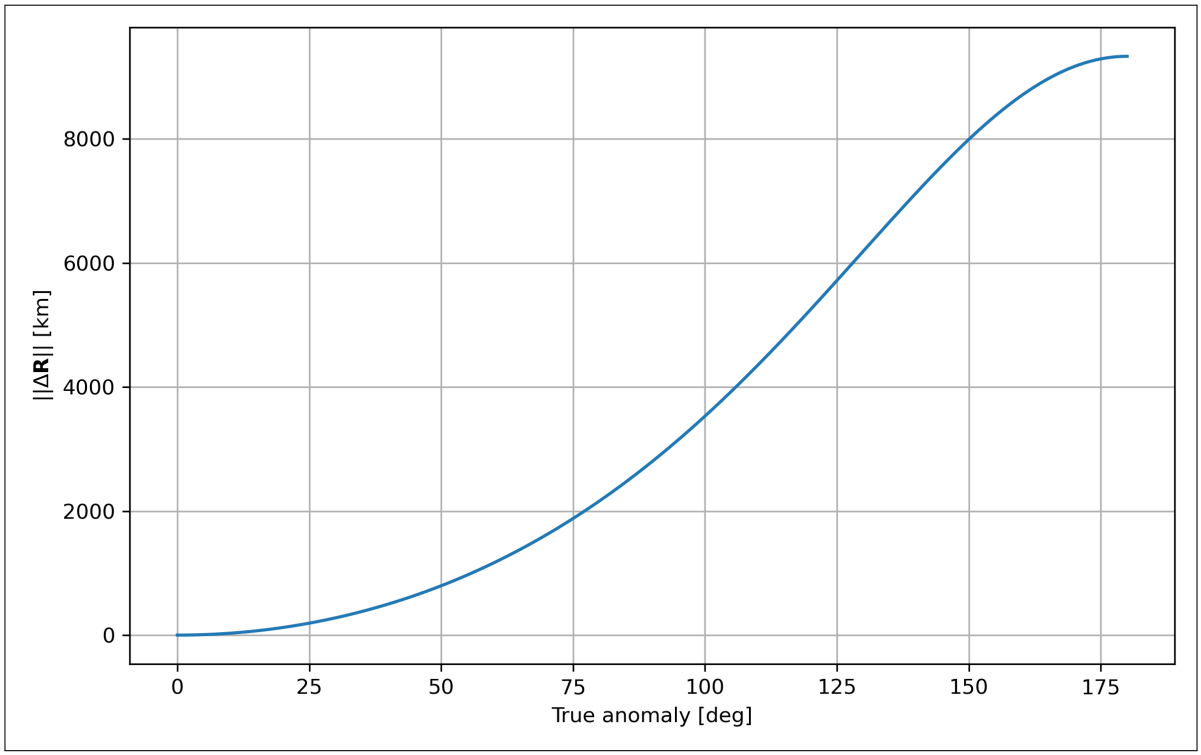

Figure 2.3: Evolution of difference in position vector magnitude between circular and elliptical orbits with respect to true anomaly. The values are provided for reference only and should not be interpreted as exact measurements.

2.2. Manoeuvre-detection frameworks

2.2.1. Motivation

Many modern SSA applications rely on continuous and accurate orbit prediction of RSOs. Many satellites are now actively manoeuvring (GNSS, large LEO constellations, etc.), so timely awareness of manoeuvres is of vital importance. RSOs in popular orbits such as GEO or Inclined Geostationary Orbit (IGSO) are required to perform frequent station-keeping manoeuvres because their missions depend on maintaining a very specific orbital geometry. GEO satellites, for example, are designed to remain fixed with respect to a point on the Earth's surface, enabling continuous broadcasting and communications services, while IGSO satellites must preserve their ground-track characteristics to guarantee navigation performance. Orbital perturbations gradually alter this geometry, causing deviations from the desired operational configuration unless corrective manoeuvres are performed [43, 44]. While cooperative systems, such as GNSS constellations, typically broadcast updated ephemerides following manoeuvres, these updates may be delayed or unavailable to all users. For satellites such as those in the BeiDou constellation that provide products to the International GNSS Service (IGS) or the International GNSS Monitoring and Assessment System (iGMAS), rapid recovery of accurate post-manoeuvre orbits remains essential to restore service availability [45].

Another aspect that makes manoeuvre detection important is catalog maintenance. Undetected manoeuvres can complicate updating satellite catalogues due to the creation of track association failures, with uncorrelated tracklets and/or duplicated objects. In some of the study cases, authors use TLE analysis to inspect satellite manoeuvre behaviour [46, 47] due to its public and near-real-time availability, at the expense of a lower accuracy that may hide small manoeuvres. In some other cases, authors use optical or radar measurements to detect orbital manoeuvres [48, 49], enjoying a higher accuracy that can even show small thrust events, at the expense of limited coverage and sensor availability.

When it comes to identifying orbital manoeuvres, some observable signatures are abrupt changes in orbital elements, residual spikes in TLE consistency checks [46, 47], filter innovation jumps (far outside the expected covariance) [50, 51, 52] or distributional anomalies in curve-fitting metrics [53].

Despite this, detecting manoeuvres in a reliable and accurate way poses several challenges. Some methods require large datasets to perform well, and observational data may suffer from long gaps or partial arcs due to tracking geometry limitations, sensor availability, or signal quality constraints [48, 49, 50]. Optical observations are further affected by weather and illumination conditions. Another key difficulty arises from environmental and model uncertainties (particularly atmospheric drag variability and solar radiation pressure changes) which can produce signatures similar to small thrust manoeuvres [46, 54].

2.2.2. Classical methods

2.2.2.1. Residual- and consistency-based methods

The core approach that this kind of classical manoeuvre detection methods follows is that predicted and observed states are compared. If there is a large difference between them, then a manoeuvre could be its cause.

TLE-based manoeuvre detection methods include TLE Consistency Checking (TCC), which is based on the idea that the spatial difference between a propagated state and the following published state should remain small in the absence of manoeuvres [46]. This approach is effective for detecting impulsive manoeuvres that produce abrupt state discontinuities, but it is less sensitive to continuous low-thrust activity, whose effects accumulate gradually and can be partially absorbed by the SGP4 fitting process. TLE Time-Series Analysis (TTSA), by contrast, compares consecutive TLEs to identify systematic changes in mean orbital elements or derived quantities. It relies on robust statistics and harmonic analysis to separate periodic variations (such as those induced by Earth's oblateness) from secular trends that indicate sustained perturbations [46]. As a result, TTSA is well suited to detect continuous thrust manoeuvres, which manifest as slow, persistent drifts in orbital elements, but its effectiveness degrades when TLE update intervals are long, limiting temporal resolution.

Closely related approaches follow the same residual-based logic, but check how orbital elements evolve over time rather than comparing individual state differences.

Moving-window curve-fitting (MWCF) methods rely on the assumption that orbital elements evolve smoothly in the absence of manoeuvres. A sliding time window of TLE data is used to fit low-order polynomials to the element evolution. As the window advances, successive fits are compared, and significant changes in fit parameters or residual statistics are interpreted as evidence of a manoeuvre.

Gui et al. employ two adjacent windows (one before and one after a candidate detection epoch) and compare curve-fit parameters such as slopes or constant offsets [53]. If the mismatch exceeds a predefined threshold, a manoeuvre is flagged.

In general, these methods are computationally efficient, robust to noise, and sensitive to both abrupt and gradual changes. However, they require careful feature and threshold tuning, and detection is inherently post-event, as it relies on windows that include observations after the manoeuvre epoch.

2.2.2.2. Gaussian uncertainty-based filtering methods

Kalman-filter-based methods provide sequential orbit estimation. The underlying principle is that a manoeuvre causes the predicted state and covariance to become inconsistent with the incoming measurements, resulting in increased residuals that can be used for manoeuvre detection.

One of the first works to show this approach is the one by Kelecy and Jah. They focus on the detection of low-thrust manoeuvres (very small, continuous accelerations) using an Extended Kalman Filter (EKF) applied to radar and optical tracking data. The EKF applies the standard Kalman filtering framework to a nonlinear dynamical system by linearising the equations of motion about a reference trajectory and updating this reference using the estimated state, thereby extending the validity of the linear approximation [50]. It performs well when the dynamics are only mildly nonlinear and the sampling rate is sufficiently high, but may fail in highly nonlinear regimes, such as high-eccentricity orbits. To improve performance under such conditions, Zhang et al. employ a Strong Tracking Filter (STF), a variant of the EKF that adaptively inflates the state covariance to maintain filter responsiveness during manoeuvres. While this approach mitigates filter divergence and improves manoeuvre tracking capability, it can lead to large post-manoeuvre error spikes and slow filter reconvergence, degrading orbit determination performance during the recovery phase [52].

Bergmann et al. use a nonlinear Kalman Filter, named Unscented Kalman Filter (UKF), over optical telescope observations [51]. The UKF approximates distributions, not functions, making it more suitable for the highly nonlinear dynamics of the cases above mentioned. Nevertheless, it is computationally more expensive and it may struggle with sudden large impulses.

In general, KF-based approaches are powerful, because they allow real-time detection and work with high-accuracy data (much more accurate than TLE), at the expense of being very sensitive to noise tuning (the selection of process and measurement noise covariance matrices) and requiring good OD initial conditions and high-cadence sensor data.

2.2.2.3. Uncertainty propagation/non-Gaussian methods

The methods discussed above rely on the assumption that state uncertainty can be adequately represented by a single Gaussian distribution. However, manoeuvres can lead to skewed or multimodal uncertainty structures that violate this assumption. To address this limitation, several approaches extend the same principle by propagating the full uncertainty distribution through the nonlinear orbital dynamics.

Yang et al. represent the initial uncertainty using a Gaussian Mixture Model (GMM), allowing multimodal distributions. Each Gaussian component is then propagated independently and the propagated Probability Density Function (PDF) is compared with incoming measurements. Significant deviations between the measurements and the predicted distribution indicate a likely manoeuvre [54].

Montilla et al. introduce two GMM-based detection metrics. The Mixture Standard Innovation Statistic extends the classical Normalized Innovation Squared (NIS) by evaluating innovation consistency over all Gaussian components rather than a single one [55]. The mixture-control metric follows the control-distance framework proposed by Holzinger et al., using the estimated minimum control effort required to reconcile the predicted and observed states as an indicator of manoeuvre likelihood [56]. This method performs well under intermittent observations and detects subtle deviations earlier than KF-based approaches.

In general, these approaches, by propagating the full probability distribution, detect manoeuvres earlier and with fewer false positives. However, the number of mixture components may need to be controlled through merging or pruning strategies to limit computational growth during uncertainty propagation.

2.2.2.4. Control-distance and optimal-control methods

The main idea behind this approach is that, given two orbital states, the minimum impulse required to "connect" them under known dynamics is computed. If this control effort is too large to be explained by natural perturbations or modelling errors, a manoeuvre is flagged.

Holzinger et al. define a control-distance metric, namely the minimum Δv required to transfer a spacecraft between two observed states. Besides manoeuvre detection, the method is also applied to object correlation and catalogue maintenance [56].

Lubey and Scheeres later extended this concept into the Optimal Control Based Estimator (OCBE) [57]. The OCBE operates similarly to a Kalman Filter, but augments the estimation process by computing, at each update step, the optimal control input required to connect the prior state estimate to the current measurement. These estimated control inputs represent unmodelled dynamics, and manoeuvres are detected whenever the inferred control effort exceeds a statistically defined threshold.

More recently, Jaunzemis et al. extended the control-distance concept through a binary hypothesis framework (manoeuvre/no manoeuvre) and introduced a combined detection metric based on both the control cost (minimum Δv) and the Mahalanobis distance [43]. By incorporating the Mahalanobis distance, which quantifies how many standard deviations the observed state deviates from the predicted state while accounting for uncertainty, the method provides a more robust discrimination of ambiguous cases.

2.2.3. ML-based methods

As shown in the previous subchapter, traditional detection methods rely on thresholds, filter innovations or model mismatches. Nevertheless, orbital behaviour is often nonlinear, multimodal and non-Gaussian.

ML provides flexible tools capable of capturing these elements. More information about the mentioned methods can be found under Appendix B.

The value of ML has been demonstrated across several SSA tasks. “Pattern-of-life” classification, i.e., the identification of typical long-term behavioural patterns of a satellite, has been successfully applied in GEO [58], supported by behavioural datasets and benchmarks [59]. Sequence-based models have been used for manoeuvre detection in catalog maintenance [60] and for continuous-thrust OD [61]. Feature-based models such as Convolutional Neural Networks (CNNs) have also shown good effectiveness when manoeuvres show certain spatial or temporal structure [62], while unsupervised methods such as anomaly detection based on Generative Adversarial Networks (GANs) capture deviations when known examples are scarce [63].

Despite these promising results, manoeuvre detection cannot yet be considered a solved problem. Most existing ML-based approaches are developed and validated on specific datasets, orbital regimes, or manoeuvre types, making their generalisation to large and heterogeneous satellite catalogues challenging. Furthermore, many methods rely on tracking data that are not consistently available in operational SSA environments. As a result, there remains a need for robust and scalable detection frameworks capable of operating directly on available catalogue data and providing information for orbit prediction systems.

2.2.3.1. Supervised learning methods

Supervised manoeuvre detection requires labelled examples, such as a manoeuvre list provided by the satellite operator, for the algorithm development.

Sequence-based models, such as LSTMs, GRUs or Recurrent Neural Networks (RNNs), treat manoeuvre detection as a time-series classification problem. They learn temporal signatures that correlate with manoeuvres based on that data. They perform manoeuvre detection by analysing the temporal evolution of the input data and identifying patterns associated with manoeuvring behaviour.

Some of the first works employing LSTM networks for manoeuvre detection were those by Cipollone et al. and Ying et al. [60, 64]. These approaches leverage sequential neural networks trained on orbital or observation-derived time series (e.g. TLE-based features or optical observation residuals), demonstrating that temporal models can effectively capture manoeuvre signatures even in the presence of noisy catalogue data. In particular, they highlight the capability of LSTM architectures to learn complex temporal patterns directly from data, reducing the need for manually tuned detection thresholds.

As briefly mentioned before, LSTM-based approaches have also been explored in orbit determination contexts for objects undergoing continuous thrust [61], showing that data-driven temporal models can complement, or even enhance, classical estimation filters.

The main strengths of this family of methods lie in their ability to model temporal dependencies and automatically learn manoeuvre-related features from raw sequential data. However, they typically require large amounts of labelled manoeuvre data for training, and their performance may degrade when applied to satellites with different dynamical regimes or limited historical data. Furthermore, their integration into operational pipelines remains non-trivial due to their data dependency and limited interpretability.

On the other hand, feature-based ML classifiers such as RFs, Deep Neural Networks (DNNs), and decision trees, operate on features derived from the available data. These features summarize relevant characteristics of the orbital dynamics and are used to identify patterns associated with manoeuvring and non-manoeuvring behaviour [65].

The main strength of this family of methods is that their training is generally simpler and their decision process is often more interpretable than that of deep sequential architectures. However, their performance is highly dependent on the quality and relevance of the engineered features, and they are generally less effective at capturing long-range temporal dependencies. In addition, their ability to generalize across different satellite populations, orbital regimes, and manoeuvre types can be limited, as the selected features are often tailored to the characteristics of the training dataset. As a result, significant feature redesign or retraining may be required when applying these methods to new operational scenarios.

2.2.3.2. Unsupervised learning methods

These methods operate without labeled manoeuvre events: they detect anomalies by identifying deviations from clusters, densities or norms.

Bai et al. used unsupervised clustering of TLE-derived residual features. They have shown good quality isolating manoeuvre-related behaviour by building clusters using different methods over changes in orbital elements obtained from the difference between predicted and observed states [47].

Some of the techniques used in clustering and density-based approaches are further explained in Appendix B, like k-means, GMM, Hierarchical Density-Based Spatial Clustering of Applications with Noise (HDBSCAN) or autoencoders.

With respect to generative and anomaly-detection approaches, GAN-based detectors like the one developed in [63] can learn how to model nominal (non-manoeuvring) behaviour such that anomalies will arise when the discriminator identifies significant deviations.

Some of the strong points of this family of approaches are that no labelled data are required and that outliers are naturally detected. On the other hand, it is often difficult to assign a physical meaning to the resulting clusters, many methods require manual interpretation of the discovered patterns, and they are not typically optimized for operational decision thresholds.

2.2.3.3. Hybrid physics-ML learning methods

As outlined in Appendix B, hybrid physics–ML approaches do not constitute a separate learning paradigm (such as supervised or unsupervised learning). Instead, they represent a new modelling dimension in which physical models (such as SGP4 or KF) and data-driven components are combined within a single framework, getting the best of both worlds by analyzing behaviour on top of dynamical models. Their relevance in this work stems from the nature of the available data: the datasets primarily consist of propagated states, residuals, and innovation sequences, which naturally lend themselves to physics-guided correction rather than purely data-driven prediction.

For instance, in [61] an example of physics-informed sequence models can be observed. An LSTM is embedded directly within the OD pipeline to compensate for unmodelled continuous thrust. Instead of predicting full trajectories, the network learns to infer manoeuvre signatures from residual patterns and then estimates representations of the unknown accelerations. These learned corrections are fused with a classical filter, so that the OD process benefits simultaneously from the physical dynamical model and the data-driven prediction of unmodelled forces.

Likewise, in [60] the LSTM manoeuvre-detector is built on top of propagation residuals, using sequences of orbital elements to identify temporal structures that deviate from the nominal propagated behavior.

Table 2.3 shows an overview of all revised manoeuvre detection frameworks (classical and ML-based), showing its working principle, required data, and strengths and limitations.

Approach	Principle	Data required	Advantages	Disadvantages
Residual or consistency (incl. MWCF)	Propagated vs. published states; detect jumps	Elements or states (e.g., TLEs)	Simple, fast	Low accuracy, poor continuous-thrust detection
Filtering-based (Gaussian & non-Gaussian)	Innovation jumps indicate inconsistencies	Measurements (e.g., radar/optical/GNSS)	Real-time, high accuracy	Sensitive to tuning and initialisation
Control-distance	Compute minimum Δv to connect states	States (precise estimates with uncertainty)	Physical interpretability	Expensive
Supervised ML	Learn labeled manoeuvre signatures	Labeled data (elements or states and/or measurements)	High accuracy	Needs labels, heavy training
Unsupervised ML	Learn nominal behaviour; flag anomalies	Unlabeled data (elements or states and/or measurements)	No labels, flexible	Harder to interpret
Hybrid physics–ML	Physics propagation + learned residuals or features	Elements or states + learning signal (residuals, OD outputs, or measurements)	Physically consistent, adaptive	Requires training and operational integration

Table 2.3: Comparison of manoeuvre detection approaches (classical and ML-based)

2.3. Synthesis and remaining gaps

The reviewed literature highlights that orbit prediction, manoeuvre detection and data-driven residual analysis have all evolved into substantial research areas. Classical propagators provide the backbone of operational SSA, while ML-based methods improve their performance. In parallel, a wide range of manoeuvre detection frameworks exist, relying on thresholds, filtering, or even ML.

Despite the clear conceptual link between orbit prediction and manoeuvre detection, relatively few works explicitly integrate both capabilities within a unified operational framework. ML-based predictors typically assume smooth dynamical evolution, whereas manoeuvre-detection methods focus on identifying deviations from expected behaviour without directly adapting the prediction process. Figure 2.4 illustrates the relationship between orbit prediction, residual generation, and manoeuvre detection. A propagation algorithm first estimates the future trajectory of a spacecraft. The differences between the predicted and observed states are then computed in the form of residuals, which contain information about potential orbital discontinuities. These residual time series can subsequently be analysed to identify manoeuvre events. While orbit prediction and manoeuvre detection are often treated as separate problems, the present work aims to connect both processes by integrating residual-based manoeuvre awareness into the ISOPA orbit-prediction framework.

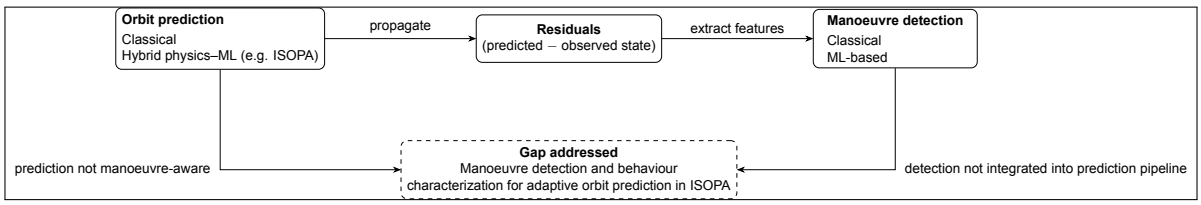

Figure 2.4: Conceptual relationship between orbit prediction and residual-based manoeuvre detection.

Several gaps have been identified in the existing literature about this topic:

- No study analyses how manoeuvre-awareness could influence the operation or configuration of ML-based orbit prediction pipelines.
- Existing manoeuvre detection methods operate at the event level and do not provide a structured characterization of satellite manoeuvring behaviour over longer time scales.
- No study analyses manoeuvres according to their underlying operational purpose, as inferred from their frequency and directional patterns.
- The literature provides no clear assessment of which residual-based features and time-series representations are most informative for manoeuvre detection.
- The amount of historical data required for reliable manoeuvre identification is not quantified.
- The robustness and consistency of detection performance across satellites with different dynamical regimes is not systematically analysed.

The next chapter formulates the research questions that arise naturally from these observations.

3

Research Objective

Building on the literature review presented in Chapter 2, it is clear that ML techniques are increasingly being adopted across SSA applications because of their demonstrated ability to improve orbit prediction accuracy and manoeuvre detection performance. However, these two research directions have largely progressed in parallel.

A common limitation across both strands of literature is the lack of manoeuvre awareness in hybrid orbit prediction. ML-based residual correctors, such as ISOPA, typically assume smooth residual evolution driven by modelling imperfections. Conversely, manoeuvre detection frameworks are primarily designed to flag individual events or estimate manoeuvre magnitude, rather than provide a characterization of said manoeuvres and integrate it in an orbit prediction pipeline.

To the best of the author's knowledge, no existing framework jointly detects sudden changes in orbital behaviour from residuals, classifies their manoeuvres according to behaviour categories, and adapts an ML-based orbit prediction pipeline accordingly. Furthermore, the literature does not provide a systematic assessment of which residual-based representations are most informative for both manoeuvre detection and their characterization, nor how many post-manoeuvre observations are needed to ensure reliable identification. This latter aspect is particularly critical, as it directly determines detection latency in near-real-time applications.

Therefore, the objective of this MSc thesis is to develop and evaluate a manoeuvre detection and classification module based on residual patterns between observed and predicted orbital states, and to integrate it into the ISOPA orbit prediction pipeline. This involves extracting informative residual features that capture manoeuvre signatures, and assessing the impact of the proposed extension on prediction accuracy.

To achieve this objective, the following research questions are formulated:

RQ1: Manoeuvre Detection Analysis

How can data-driven analysis be used to identify manoeuvre events and characterise satellite manoeuvring behaviour?

- Which TLE consistency representations provide the most informative features for manoeuvre detection?
- How do different residual-based score representations influence the performance and reliability of manoeuvre detection?
- How does the presence of outliers affect detection performance, and how can outliers be mitigated?
- How many subsequent observations (or TLE updates) are required after a manoeuvre to reliably detect its occurrence?
- How consistent is the detection performance across different satellites?

RQ2: Manoeuvre-Aware Orbit Prediction

How can manoeuvre awareness be incorporated into a hybrid orbit-prediction framework to improve operational performance?

- How should detected behaviour classes be used to adapt the prediction strategy within the ISOPA pipeline?
- To what extent does manoeuvre-aware adaptation improve prediction accuracy compared with a behaviour-agnostic orbit-prediction model?

4

Methodology

This chapter describes the methodological pipeline used to detect manoeuvres, classify them and interface with ISOPA. The chapter is structured as follows. Section 4.1 provides an end-to-end overview of the pipeline. Section 4.2 motivates the final methodological decisions. Section 4.3 details the residual generation process from orbit products, the dataset construction, and chronological splits. Section 4.4 defines the passive satellite exclusion, score capping and manoeuvre detection methodology. Section 4.5 shows the event detection logic. Section 4.6 describes satellite-level behaviour classification, both according to frequency and nature, and manoeuvre nature classification. Section 4.7 provides an explanation of how the behaviour of unknown satellites is evaluated. Section 4.8 shows the integration of the output of the algorithm into ISOPA.

4.1. End-to-end pipeline overview

The overall pipeline is designed to detect manoeuvre-related behaviour changes from orbit products (TLE) and to provide both satellite and manoeuvre behavioural classes suitable for downstream orbit prediction within an already created framework (ISOPA). At a high level, residuals are generated from input orbit products, organized according to their reference epoch, and used to generate scores according to the discrepancy with respect to a nominal (non-manoeuvre-affected) orbit. Manoeuvres are identified as temporally coherent deviations from this nominal baseline.

Figure 4.1 summarizes the main processing stages and outputs: historical TLEs are propagated to the epoch of a target TLE, transformed from TEME to EME2000 and RTN, and converted into TLE-consistency metrics. These time series are annotated with manoeuvre ground truth, split chronologically, converted into anomaly/consistency scores, and processed through event-detection logic to produce final manoeuvre detections and both satellite-level and manoeuvre-level behaviour indicators. Finally, the algorithm acts as a supervisory layer for ISOPA by identifying manoeuvre-contaminated epochs and excluding them from the training dataset, ensuring that the orbit-prediction framework is trained on nominal orbital evolution.

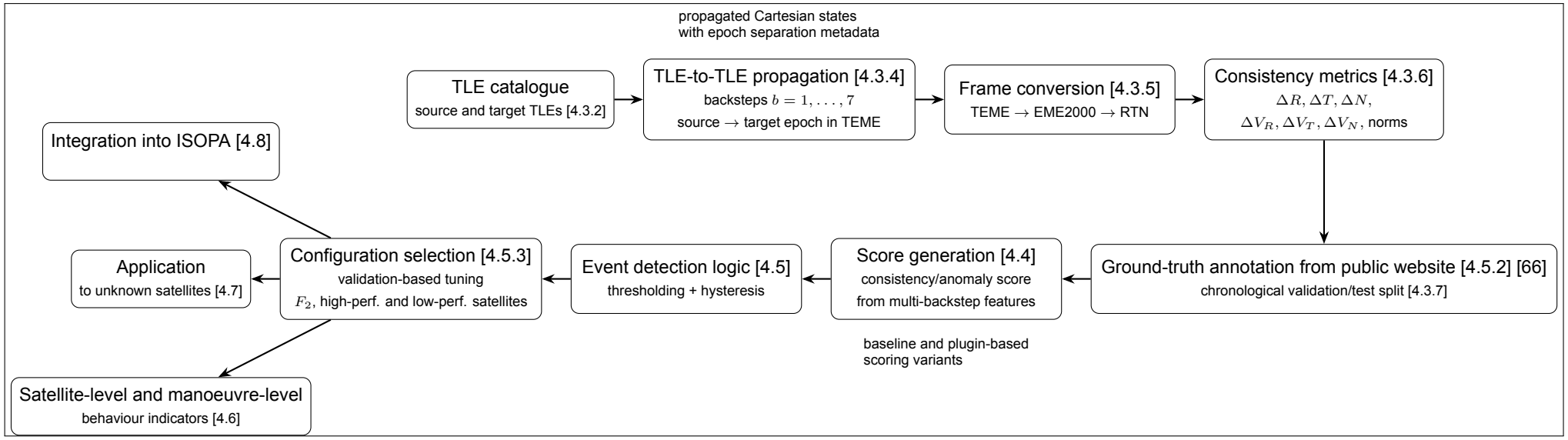

Figure 4.1: End-to-end methodological pipeline.

4.2. Problem framing and methodological decisions

4.2.1. Unsupervised vs supervised modelling

Manoeuvre events are sparse relative to the total number of available samples, and their effects extend over multiple subsequent epochs with an *a priori* unknown duration. This results in highly imbalanced datasets and blurred class boundaries, making it difficult to define consistent labels at the sample level.

An initial approach based on supervised learning using nominal (non-manoeuvre) windows was explored. While the models themselves were able to fit the training data, it became evident that the main limitation did not lie in model capacity, but in the quality and consistency of the underlying TLE data. In practice, TLEs exhibit irregular noise patterns that vary across satellites and time periods, leading to inconsistencies that are not related to manoeuvres. As a result, models trained on nominal data tend to overfit satellite-specific or epoch-specific artefacts, reducing their ability to generalize across different objects and operational conditions. Moreover, the lack of enough labelled data made it also difficult to follow this kind of approach.

This observation motivates a shift towards a behaviour-centric formulation, in which manoeuvres are not learned explicitly from labels, but are instead identified as deviations from internally consistent orbital behaviour. By focusing on relative inconsistencies between TLEs rather than absolute classification, the method becomes more robust to the intrinsic noise (as these effects are largely shared across propagated states), and lack of enough labelled manoeuvre information.

4.2.2. TLE-based vs CPF-based residuals

An alternative approach to residual generation would consist of comparing TLE-derived states against a high-precision external reference, such as Consolidated Prediction Format (CPF) ephemerides (which are used within ISOPA to assess the quality of the generated output), or even comparing high-accuracy ephemerides against each other. While this would provide physically meaningful residuals, it introduces strong dependencies on external data sources that are only available for a limited subset of satellites and time periods, and are therefore not suitable for large-scale or fully automated processing. In contrast, TLE data provide global coverage and continuous availability across a wide range of objects, at the cost of higher and less structured uncertainty.

In contrast, the adopted approach relies exclusively on TLE-based consistency metrics, computed from internal discrepancies between multiple TLEs propagated to a common epoch. This choice ensures that the method remains fully self-contained and directly applicable to publicly available data, without requiring access to high-precision orbit products.

Moreover, restricting the pipeline to TLE-based inputs is essential to meet the operational objective of near real-time applicability. Since TLEs are continuously updated and widely accessible, a method based solely on TLE data can be deployed in a scalable and timely manner across large satellite populations.

4.2.3. Real-time and generalization constraints

A central design requirement of the methodology is the ability to generalize across a wide range of satellites and time periods while maintaining near real-time performance. In contrast to approaches tailored to specific missions or relying on high-fidelity orbit determination, the objective is to develop a unified framework that can be applied consistently to any object for which TLE data is available.

This requirement introduces a trade-off. On one hand, the method must remain lightweight and computationally efficient to support large-scale processing and rapid updates. On the other hand, it must be robust to the heterogeneous and sometimes inconsistent nature of TLE data, which varies significantly across satellites and time periods.

The combination of these constraints makes generalization one of the main challenges of the problem.

4.2.4. Final methodological choice

Based on these considerations, manoeuvre detection is formulated as a consistency-based anomaly detection problem. The methodology can be summarized as follows:

1. Construct TLE-to-TLE consistency features by propagating historical TLEs to a common target epoch and measuring their discrepancies with respect to the most recent orbit solution.
2. Convert these discrepancies into scalar anomaly scores that quantify deviations from nominal behaviour.
3. Identify manoeuvres as temporally coherent excursions of the anomaly score through threshold-based event detection logic calibrated on validation data.

Figure 4.2 shows a representation of how the score is expected to behave when a manoeuvre is detected: after a threshold has been obtained through the optimization logic, scores above that value will be identified as manoeuvres.

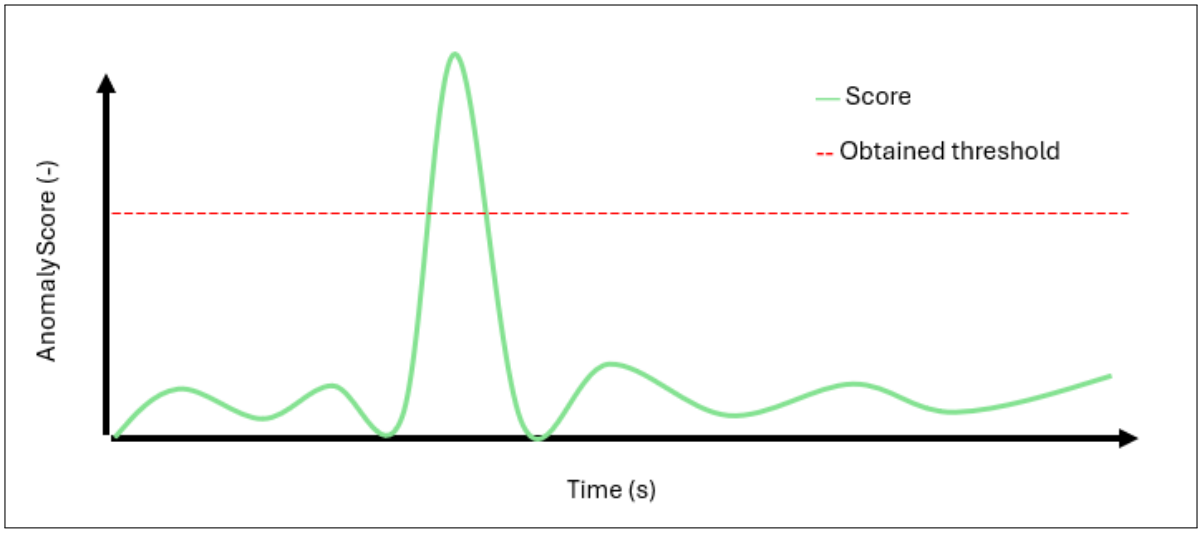

Figure 4.2: Representation of how a score anomaly would be identified as a manoeuvre.

4.2.5. Relation to other manoeuvre detection approaches

In the available literature, several works address satellite manoeuvre detection using strategies that are conceptually related to the one adopted in this thesis. Within the TLE-only category, Kelecy et al. [67] analyze the effect of thresholds and window lengths when constructing detection filters based on polynomial fits to orbital energy or inclination variations extracted from raw TLE data. Mukundan and Wang [68] similarly rely on previous-to-following TLE propagation and study how the choice of window size and threshold affects detection performance. Lemmens and Krag [46] propose two complementary TLE-based approaches, namely a consistency check based on the discrepancy between a propagated TLE and the subsequent published one, and a time-series analysis framework based on robust statistics and harmonic analysis of orbital-element histories. Bai et al. [47] also operate on historical TLE data, but formulate the problem from a data-mining perspective through unsupervised clustering of nominal and manoeuvre-related behaviour patterns.

Although these approaches share the use of TLE-derived information, the proposed methodology differs from them in several relevant aspects. First, most previous analyses are conducted on one or a few active satellites with relatively clear manoeuvre signatures, whereas the present framework is evaluated on a broader and more heterogeneous set of 17 satellites, including active and passive objects.

A second difference concerns the data assumptions underlying the detection problem. In addition to TLE-based methods, a substantial part of the literature relies on richer sources of information, such as radar or passive-optical measurements, reconstructed orbital states, or state uncertainty information [50, 51, 52, 54, 55, 56, 69]. These approaches benefit from physically consistent state estimates and, in many cases, explicit uncertainty characterization, but they require tracking infrastructures or orbit-determination pipelines that are not generally available. By contrast, the methodology proposed here is deliberately constrained to a TLE-only setting in order to preserve scalability, public-data applicability, and near-real-time usability.

A similar distinction applies to machine-learning-based methods. Some TLE-only learning approaches, such as those of Wang et al. [70] and Li et al. [62], formulate the problem as classification over temporal windows or dynamical regimes rather than as the detection of discrete manoeuvre events. Other learning-based works rely on richer or alternative data sources, including historical orbital information from catalog-maintenance pipelines [60], simulated optical residuals [64], measurement-based orbit-determination data [61], synthetic orbital trajectories [63, 65], or high-accuracy ephemerides and astrometric time series for pattern-of-life characterization [58, 59]. While these methods show the potential of machine learning for manoeuvre-related analysis, they are generally less aligned with a strictly causal, TLE-only, event-level detection setting.

Another important difference lies in parameter selection. In several previous TLE-based approaches, the analysis is centered precisely on how detection performance changes as thresholds, window lengths, filtering settings, or related design parameters are varied [67, 71]. Even in more systematic methods such as those of Lemmens and Krag [46] or Bai et al. [47], the final behaviour still depends on methodological choices that are fixed in advance and not always optimized in a strictly forward generalization setting. In contrast, the methodology proposed in this thesis selects its main parameters exclusively on a validation subset and evaluates them on a later chronological split, without using explicit knowledge of future manoeuvres. This makes the pipeline fully causal and more representative of the conditions under which it would operate in practice.

A further key distinction concerns detection latency. Existing TLE-based methods often rely on temporal windows or broader consistency patterns that require several TLEs after the manoeuvre epoch before an event can be robustly identified [46, 67, 68]. Learning-based methods based on temporal windows exhibit similar limitations, since they typically aggregate context over multiple observations. By contrast, the present method is designed to detect a manoeuvre as early as the first TLE immediately following the event, since each sample is constructed exclusively from past information and the score is evaluated directly at the current epoch.

Finally, the detection logic adopted in this thesis is intentionally strict. A true positive is declared only when the anomaly score exhibits a clear spike at the first TLE immediately after the manoeuvre epoch; otherwise, the event is counted either as a false positive or as a false negative, depending on the location of the response. However, a limited tolerance window is introduced to account for TLEs whose epochs lie very close to the manoeuvre epoch. In such cases, the orbit-determination process used to generate the TLE may not yet fully reflect the orbital discontinuity, causing the corresponding score increase to appear one or a few updates later. Even with this tolerance, the adopted criterion remains considerably stricter than those used in approaches that allow manoeuvre signatures to be detected over much broader post-manoeuvre intervals [46, 67].

From this perspective, the proposed method should be regarded not merely as a manoeuvre-detection algorithm, but as an operational framework for causal, near-real-time, event-level manoeuvre detection based exclusively on TLE data.

4.3. Residual generation and feature construction from orbit products

4.3.1. Purpose

This section summarizes how reference orbits are propagated and how residuals between observed and predicted states are computed. Residuals form the primary model input, encoding the cumulative effect of unmodeled dynamics (including manoeuvres). Compared to raw states dominated by nominal orbital motion, residuals isolate behavioural deviations and are therefore more informative for manoeuvre detection.

4.3.2. Satellites feeding the dataset

The satellites selected for the dataset comprise those for which manoeuvre episodes and corresponding descriptions are available, excluding satellites whose datasets were majorily based on 1990's data [66]. After removing them, there are 13 satellites together with 4 additional passive satellites (AJISAI, LAGEOS1, LAGEOS2, and STARLETTE).

Including as many satellites as possible, with different manoeuvre frequencies and structures in the input dataset is essential to ensure that the pipeline is exposed to a range of dynamical behaviours. This design enables the method not only to distinguish between different types of active satellites but also to correctly identify the absence of manoeuvres, thereby allowing the classification produced at the output to reflect the same structure present in the input data and ensuring a closed evaluation of the system.

4.3.3. Orbit products and native reference frames

The orbit products to feed the model are TLEs, a compact representation of satellite orbits distributed by NORAD [72], designed to be propagated with SGP4. Propagation yields Cartesian states in the True Equator, Mean Equinox (TEME) frame, which is associated with SGP4 but is not a formally defined IAU/IERS inertial frame.

4.3.4. Residual definition and preprocessing

The adopted residual family corresponds to *spread-only consistency metrics*, defined as internal inconsistencies between historical TLEs when propagated to a common target epoch.

For each satellite, let $\{\mathrm{TLE}_i\}$ denote the chronological sequence of available TLEs. For a given target TLE TLE_i with epoch t_i, a set of previous TLEs TLE_{i-k} with $k = 1, \ldots, 7$ is selected. Each of these historical TLEs is propagated forward to the common epoch t_i using SGP4, generating a set of predicted states

$$\mathbf{x}_{i-k\to i} = \mathrm{SGP4}(\mathrm{TLE}_{i-k}, t_i), \tag{4.1}$$

expressed initially in the TEME frame.

The most recent TLE TLE_i at its own epoch corresponds to the reference state:

$$\mathbf{x}_i^{\mathrm{ref}} = \mathbf{x}(\mathrm{TLE}_i, t_i). \tag{4.2}$$

where $\mathbf{x}_i^{\mathrm{ref}}$ denotes the reference state associated with the target TLE at its own epoch.

All states are then transformed from TEME to EME2000 and subsequently projected into the RTN frame, as described in the following subsection. Residuals are defined as the difference between each propagated historical state and the reference state at the same epoch:

$$\Delta\mathbf{x}_{i-k} = \mathbf{x}_{i-k\to i} - \mathbf{x}_i^{\mathrm{ref}}. \tag{4.3}$$

In RTN coordinates, this yields position and velocity residual components

$$\Delta\mathbf{r}_{i-k} = \begin{bmatrix}\Delta R\\ \Delta T\\ \Delta N\end{bmatrix}, \qquad \Delta\mathbf{v}_{i-k} = \begin{bmatrix}\Delta V_R\\ \Delta V_T\\ \Delta V_N\end{bmatrix}, \tag{4.4}$$

which quantify discrepancies in radial distance, along-track motion, and orbital plane orientation, as well as their corresponding velocity components. The generation of this coordinate frame will be further explained in the following subsection 4.3.5.

A total of seven historical TLEs are considered for each target epoch (resulting in eight states including the reference). This choice reflects a trade-off between information content and robustness. Older TLEs tend to diverge more strongly in the presence of manoeuvres, providing a clearer separation between nominal and manoeuvre-affected behaviour. However, excessively long propagation intervals introduce large model-driven errors (on the order of several kilometres after a few days) [71], which can dominate the residual signal and degrade its interpretability. Limiting the number of backsteps therefore ensures that residuals remain informative while avoiding contamination from long-term SGP4 propagation errors. The exact number of 7 is chosen because it is still a light algorithm to be run and older TLEs do not diverge much, but it contains enough information so that manoeuvre signatures are undoubtedly identifiable.

Finally, additional scalar consistency metrics such as $\|\Delta\mathbf{r}\|$ and $\|\Delta\mathbf{v}\|$ are derived from the RTN components to summarize the magnitude of the discrepancies.

4.3.5. Common inertial analysis frame and local orbital plane

To enable residual computation, TLEs are converted into a common inertial frame: Earth Mean Equator and Equinox of J2000.0 (EME2000), the standard Earth-Centered Inertial (ECI) frame provided by Orekit, an open-source space dynamics library developed by the French Space Agency (CNES) and widely used in both operational and research astrodynamics applications [73]. An ECI frame avoids artificial periodicities caused by Earth rotation.

For interpretability, residuals are also represented in a local orbital frame, chosen as the Radial-Transverse-Normal (RTN) frame. In the RTN frame, radial residuals primarily capture geometric position errors in the direction of the radius vector, often associated with altitude and eccentricity-phase deviations. Transverse residuals are mainly sensitive to along-track errors, which can accumulate over time due to mean motion mismodelling and therefore indirectly reflect semi-major axis or energy-related errors. Normal residuals are associated with out-of-plane dynamics and are linked to inclination and node variations.

Let the satellite position and velocity in EME2000 be denoted by

$$\mathbf{r} = \begin{bmatrix} x \\ y \\ z \end{bmatrix}, \qquad \mathbf{v} = \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \end{bmatrix}. \tag{4.5}$$

The local RTN frame is constructed from the instantaneous orbital geometry. The radial unit vector is defined as

$$\hat{\mathbf{e}}_R = \frac{\mathbf{r}}{\|\mathbf{r}\|}. \tag{4.6}$$

The orbital angular momentum vector is

$$\mathbf{h} = \mathbf{r} \times \mathbf{v}, \tag{4.7}$$

from which the normal unit vector is obtained as

$$\hat{\mathbf{e}}_N = \frac{\mathbf{h}}{\|\mathbf{h}\|}. \tag{4.8}$$

The transverse unit vector completes the right-handed orthonormal triad:

$$\hat{\mathbf{e}}_T = \hat{\mathbf{e}}_N \times \hat{\mathbf{e}}_R. \tag{4.9}$$

Using these basis vectors, the rotation matrix from EME2000 to RTN is written as

$$\mathbf{C}_{\mathrm{EME2000}\to\mathrm{RTN}} = \begin{bmatrix} \hat{\mathbf{e}}_R^T \\ \hat{\mathbf{e}}_T^T \\ \hat{\mathbf{e}}_N^T \end{bmatrix}. \tag{4.10}$$

In this work, the RTN basis is constructed from the reference state associated with the target TLE at its own epoch. The residuals are computed as the difference between the state obtained by propagating an earlier source TLE to the target epoch and the state obtained from the target TLE at that same epoch. Figure 4.3 shows the RTN basis.

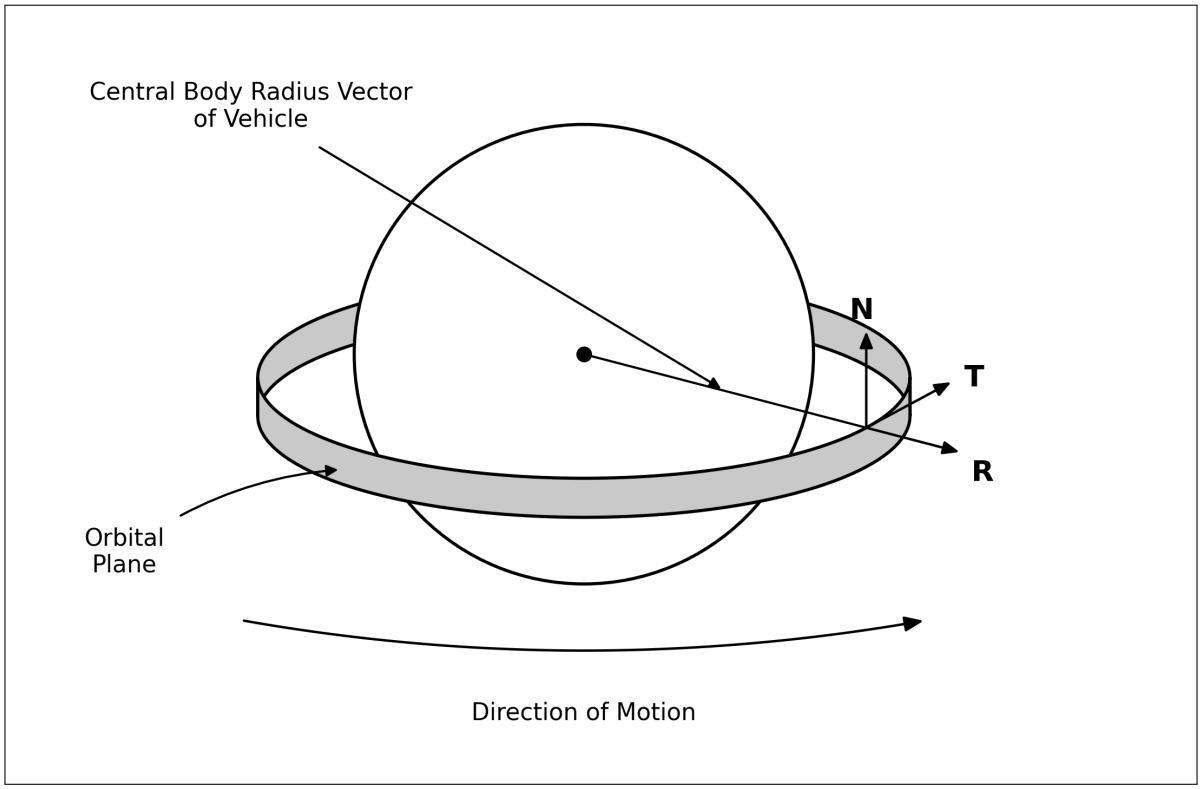

Figure 4.3: Representation of RTN basis.

These position and velocity residuals, initially expressed in EME2000, can then be projected into RTN as

$$\Delta\mathbf{r}_{\mathrm{RTN}} = \mathbf{C}_{\mathrm{EME2000}\to\mathrm{RTN}}\,\Delta\mathbf{r}_{\mathrm{EME2000}}, \tag{4.11}$$

$$\Delta\mathbf{v}_{\mathrm{RTN}} = \mathbf{C}_{\mathrm{EME2000}\to\mathrm{RTN}}\,\Delta\mathbf{v}_{\mathrm{EME2000}}. \tag{4.12}$$

In component form, this yields

$$\Delta\mathbf{r}_{\mathrm{RTN}} = \begin{bmatrix}\Delta R\\ \Delta T\\ \Delta N\end{bmatrix}, \qquad \Delta\mathbf{v}_{\mathrm{RTN}} = \begin{bmatrix}\Delta V_R\\ \Delta V_T\\ \Delta V_N\end{bmatrix}. \tag{4.13}$$

4.3.6. Feature representation and per-epoch dataset structure

The dataset is structured at the level of individual target epochs, each representing the discrepancy between a reference TLE and the 7 previous TLEs propagated to the same epoch.

For a given target index i, let $\{\Delta\mathbf{x}_{i-k}\}_{k=1}^{K}$ denote the residuals obtained from $K = 7$ previous TLEs propagated to t_i. From these residuals, a set of scalar and component-wise features is constructed. These include:

- **Component-wise residuals in RTN:** ΔR, ΔT, ΔN, ΔV_R, ΔV_T, ΔV_N for each backstep k,
- **Norm-based metrics:** $\|\Delta\mathbf{r}\|$ and $\|\Delta\mathbf{v}\|$,
- **Temporal separation features:** time difference between source and target TLE epochs.

These features form a vector representation

$$\mathbf{f}_i \in \mathbb{R}^{N_f}, \tag{4.14}$$

where $N_f = 63$ in the selected configuration.

Each dataset entry therefore corresponds to a single target epoch and contains:

- the feature vector $\mathbf{f}_i$,
- a binary label $y_i \in \{0, 1\}$ indicating whether the epoch is affected by a manoeuvre (based on ground-truth intervals), used exclusively for evaluation purposes and not as input to the algorithm,
- metadata such as satellite identifier and epoch timestamp.

4.3.7. Chronological split strategy

To ensure a realistic evaluation scenario, the dataset is partitioned chronologically for each satellite. All samples are first ordered by their epoch, and then divided into two disjoint subsets: a validation set and a test set. The validation subset corresponds to the first half of the time series, while the test subset contains the rest of the epochs, thus preserving the natural temporal ordering of the data.

This chronological split is essential to avoid information leakage from future observations into the calibration process. In particular, all model selection steps, including threshold calibration, passive-satellite identification, and configuration optimization, are performed exclusively on the validation subset. The test subset is kept completely unseen during this process and is used only for the final evaluation of the method.

Observations about this partition and the impact it has on the outcome of the algorithm will be discussed in the results section.

4.4. Manoeuvre detection via consistency-based anomaly scoring

4.4.1. Overview of the scoring framework

Given the feature representation described in the previous section, manoeuvre detection is formulated as a scalar scoring problem at the level of individual target epochs. For each epoch t_i, a feature vector $\mathbf{f}_i$ encodes the discrepancies between the current TLE and the 7 previous TLEs propagated to the same epoch.

The objective of the scoring function is to map $\mathbf{f}_i$ into a scalar anomaly score

$$S_i = g(\mathbf{f}_i), \tag{4.15}$$

such that nominal behaviour corresponds to low scores, while manoeuvre-affected epochs produce significantly higher values.

Rather than adopting a single fixed formulation, multiple scoring strategies have been developed and evaluated progressively, ranging from simple physically motivated baselines to more advanced data-driven approaches.

4.4.2. Passive satellite exclusion

The exclusion of passive satellites is based on the simplest propagation scheme available in the framework, namely a baseline previous-to-following TLE propagation approach, which will be described in Subsection 4.4.4.1. The underlying criterion is defined using the position discrepancy between consecutive TLEs propagated with a one-step backstep, quantified by the Euclidean norm

$$S_i = g(\mathbf{f}_i) = |\Delta \mathbf{r}_{i,1}|, \tag{4.16}$$

which provides a direct measure of the consistency of the orbit propagation between successive TLE updates. For each satellite, the validation subset is extracted using the same chronological split adopted throughout the pipeline. A robust upper-envelope statistic is then computed from this subset as a high percentile of the residual distribution:

$$q_s = Q_{99.6}\left(\mathcal{V}_{\text{val}}^{(s)}\right), \tag{4.17}$$

where $\mathcal{V}_{\text{val}}^{(s)}$ denotes the set of $\|\Delta \mathbf{r}_{i,1}\|$ values corresponding to the validation rows of satellite s. A satellite is classified as passive if

$$q_s < 1.2 \text{ km}. \tag{4.18}$$

This rule is intentionally robust: by relying on a high percentile rather than the absolute maximum, the classification is not affected by a small number of isolated spikes. As a result, satellites that are almost

always dynamically quiet are correctly identified as passive even if occasional anomalous TLE rows are present. Physically, this reflects the characteristic behaviour of passive satellites, whose propagation-consistency residuals remain close to zero over long time spans, with only sporadic outliers that do not alter the overall structure of the signal.

In the present dataset, this procedure yields a 100% classification accuracy, consistently separating AJISAI, LAGEOS1, LAGEOS2, and STARLETTE from the active subset. Figures 4.4 and 4.5 illustrate this difference: Figure 4.4 shows the residual evolution for a passive satellite (AJISAI), characterized by low values and some few spikes. This satellite has a q_s value of 0.648 km. Figure 4.5 shows the behaviour of an active satellite (SEN3B), where larger and more frequent spikes are observed over a comparable time span. SEN3B shows a q_s value of 22.696 km.

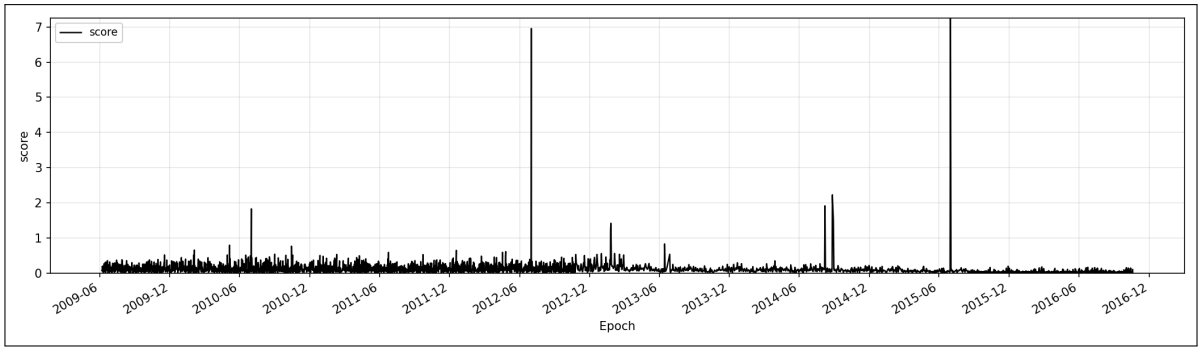

Figure 4.4: AJISAI score evolution on the passive satellites exclusion test for the validation set

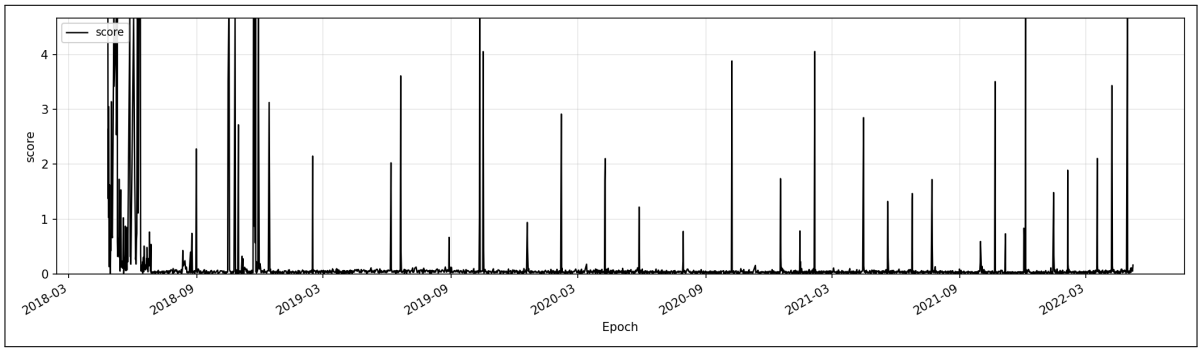

Figure 4.5: SEN3B score evolution on the passive satellites exclusion test for the validation set, after zooming in

The passive list obtained in this way is stored and excluded from the subsequent optimization and event-detection stages.

4.4.3. Adaptive score capping for cross-satellite comparability

Due to limitations in SGP4 propagation and TLE generation, certain epochs exhibit extremely large residual outliers. Since the detection logic relies on percentile-based thresholds computed from the validation set, the presence of such extreme values leads to an artificial inflation of the thresholds. As a result, the method becomes less generalizable across satellites and more difficult to interpret.

To mitigate this issue, a capping strategy is introduced to suppress extreme outliers. Specifically, the 99th percentile of the residual distribution within each validation split is used as an upper bound, effectively removing only the most extreme values. This ensures that the detection logic is driven by the nominal orbital behaviour rather than being dominated by spurious outliers. Consequently, threshold values are reduced and become more consistent across satellites, significantly improving the robustness and generalization capability of the method.

A clear illustration of this motivation is provided in Figures 4.6 and 4.7. The former shows the evolution

of the HY-2D validation-split residuals prior to outlier capping in the baseline case, while the latter depicts the same data after applying the capping strategy. These plots represent the score evolution available to the detection algorithm and therefore directly illustrate the effect of the capping strategy on the input used for threshold definition. The second figure not only offers a clearer and more interpretable representation, but also enables the use of more meaningful threshold values, as the thresholds are defined relative to the maximum score observed in the corresponding validation set for each satellite. Consequently, when extreme outliers are present, the thresholds are determined by exceptionally large values, whereas capping the scores reduces the effective maximum, as illustrated in the second figure, and allows thresholds to be set at more meaningful levels.

Importantly, this approach does not negatively impact satellites that do not exhibit extreme outliers. In such cases, the residual evolution is already smooth, and removing the highest values has a negligible effect. After incorporating this strategy, a noticeable improvement in the results was observed, further confirming its effectiveness.

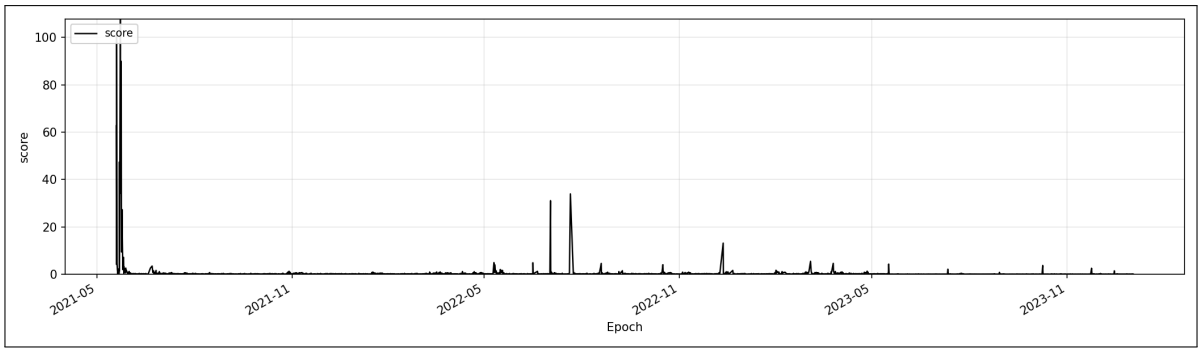

Figure 4.6: Evolution of HY-2D residuals for the validation split of the baseline configuration before capping

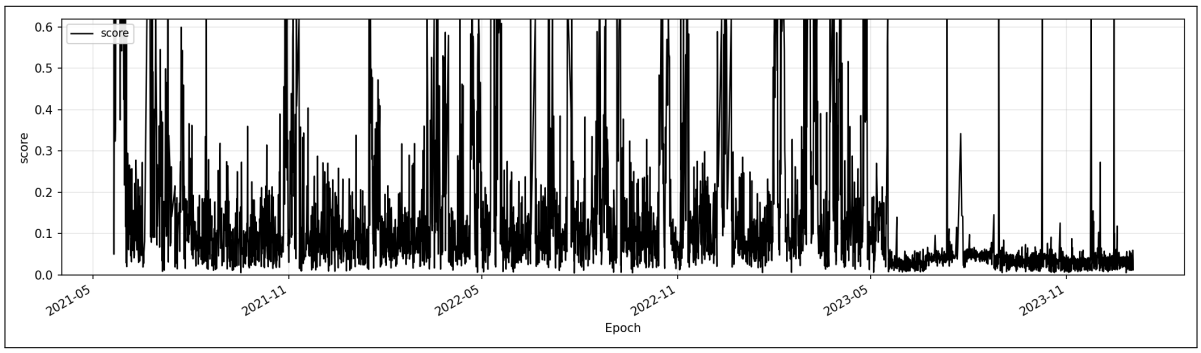

Figure 4.7: Evolution of HY-2D residuals for the validation split of the baseline configuration after capping

4.4.4. Progressive development of the scoring function

All subsequent scoring-function developments are carried out on the active-satellite subset (with passive objects excluded) and on capped residuals, thereby mitigating the influence of outliers. Rather than adopting existing formulations from the literature, the scoring functions proposed in this thesis are designed from first principles, guided by the expected behaviour of TLE-to-TLE propagation residuals before, during, and after manoeuvre events.

4.4.4.1. Baseline consistency score

The first scoring formulation, as briefly mentioned before, uses only the consistency magnitude associated with the shortest propagation gap, namely backstep $k = 1$. Following the definitions of Section 4.3.4, for a target epoch t_i let

$$r_{i,k}, \qquad k = 1, \ldots, K, \quad K = 7, \tag{4.19}$$

denote the scalar position-consistency feature at the target epoch t_i associated with backstep k, i.e. the magnitude $\|\Delta\mathbf{r}_{i-k}\|$ of the RTN position residual obtained by propagating TLE_{i-k} to t_i and comparing it with the reference state of TLE_i. Here the epoch index i always denotes the target epoch under evaluation: all features $r_{i,k}$ share the same target epoch t_i and the same reference TLE_i, and differ only in the backstep k, i.e. in how many TLEs back the propagated source element set is taken. The shortest gap corresponds to $k = 1$. The baseline score is therefore simply defined as

$$S_i^{(1)} = r_{i,1}. \tag{4.20}$$

This is the most direct formulation: manoeuvres are expected to produce large discrepancies already in the immediate TLE-to-next-TLE position vector magnitude comparison. However, this score is highly sensitive to local TLE noise, since it relies on a single backstep only and does not exploit the redundancy available in the historical TLE sequence.

4.4.4.2. Median back-innovation score

From this point onward, all score formulations are presented for the position residuals only. The velocity residuals are incorporated through an identical processing chain, and the final score is obtained as the equally weighted combination of the position- and velocity-based contributions.

The second stage introduces the idea of *back-innovation*. Instead of using only the shortest backstep $r_{i,1}$, its consistency is compared against the typical consistency observed across the $K - 1 = 6$ older backsteps $r_{i,k}$, $k = 2, \ldots, K$. A row-wise predictor is built as the median of these older backsteps:

$$\hat{r}_i = \mathrm{median}\{r_{i,2}, \ldots, r_{i,K}\}. \tag{4.21}$$

The innovation is then defined as

$$u_i = |r_{i,1} - \hat{r}_i|\,. \tag{4.22}$$

Finally, the innovation is normalized per satellite using a robust scale estimated from the fit segment of the shortest-backstep consistency values:

$$S_i^{(2)} = \frac{u_i}{\mathrm{MAD}_{\mathrm{fit}}\,(r_{i,1})}, \tag{4.23}$$

where MAD denotes the median absolute deviation, a measure of statistical dispersion based on the median of the absolute deviations from the median. This stage is more robust than the raw baseline because it compares the shortest backstep against the local historical pattern rather than evaluating it in isolation. As a result, the score becomes sensitive to sudden changes in consistency behaviour, which are more indicative of manoeuvres than persistently large residuals.

4.4.4.3. Whitened vector innovation score

The third stage replaces the scalar innovation by a vector-valued innovation across the older backsteps. This vector is associated with the target epoch t_i and, by construction, has exactly $K - 1 = 6$ components: one per older backstep, each contrasting the shortest backstep $r_{i,1}$ against one older backstep $r_{i,k}$, $k = 2, \ldots, K$:

$$\mathbf{v}_i = \begin{bmatrix} r_{i,1} - r_{i,2} \\ r_{i,1} - r_{i,3} \\ \vdots \\ r_{i,1} - r_{i,K} \end{bmatrix} \in \mathbb{R}^{K-1}. \tag{4.24}$$

This vector is robustly whitened using the median and MAD estimated from the fit subset:

$$\tilde{\mathbf{v}}_i = \frac{\mathbf{v}_i - \mathrm{med}_k(\mathbf{v}_{\mathrm{fit}})}{\mathrm{MAD}_k(\mathbf{v}_{\mathrm{fit}})}, \tag{4.25}$$

with the division understood component-wise. Extreme values are clipped to a fixed range to avoid domination by isolated outliers:

$$\tilde{\mathbf{v}}_i \leftarrow \mathrm{clip}(\tilde{\mathbf{v}}_i,\, -z_{\max},\, z_{\max}), \qquad z_{\max} = 20. \tag{4.26}$$

The score is then obtained from the norm of the whitened innovation vector. In the default formulation, this is an unweighted L_2-type aggregation over the $K-1$ components:

$$S_i^{(3,\mathrm{raw})} = \sqrt{\frac{1}{K-1}\sum_{k=2}^{K} \tilde{v}_{i,k}^2}, \tag{4.27}$$

where $\tilde{v}_{i,k}$ denotes the component of $\tilde{\mathbf{v}}_i$ associated with older backstep k. As before, this raw score is normalized per satellite using the MAD of the fit segment. This stage is more informative because it captures the full innovation pattern across multiple backsteps, allowing the method to distinguish systematic deviations from isolated noise fluctuations.

4.4.4.4. Tuned whitened vector innovation score

The fourth stage keeps the same whitened-vector structure but introduces three important refinements that were found to improve performance.

First, older backsteps are not treated uniformly. A decreasing weight is assigned according to their recency, so that more recent older backsteps count more:

$$w_k \propto \frac{1}{(k-1)^{p_w}}, \qquad k = 2, \ldots, K, \tag{4.28}$$

with $p_w = 0.1$ in the selected configuration, so that the most recent older backstep ($k = 2$) receives the largest relative weight. This weighting scheme was introduced empirically during the validation campaign. The underlying rationale is that more recent backsteps are expected to be slightly more representative of the current orbital behaviour than older ones, while still preserving information from the full propagation history. The relatively small value of p_w ensures that the weighting remains mild, avoiding excessive reliance on any single backstep. In addition, the weights are normalized such that

$$\sum_{k=2}^{K} w_k = 1, \tag{4.29}$$

ensuring that the resulting score remains on a comparable scale to the unweighted formulation and can be interpreted as a weighted average contribution from all older backsteps. The specific chosen value was the result of an iterating process in which all values from $p_w = 0.1$ to $p_w = 0.5$ were tried in intervals of 0.1, yielding $p_w = 0.1$ the best outcome.

Second, the clipping level is decreased to $z_{\max} = 15$, instead of the default value of $z_{\max} = 20$ used by the initial whitened-vector innovation score. This still allows strong manoeuvre signatures to remain visible while applying a slightly more restrictive cap to extreme normalized innovations.

Third, a consensus factor is introduced to reward epochs for which several backsteps simultaneously exhibit large normalized innovations. Let

$$\phi_i = \frac{1}{K-1}\sum_{k=2}^{K} \mathbb{I}\left(|\tilde{v}_{i,k}| > z_0\right), \tag{4.30}$$

where $\mathbb{I}(\cdot)$ denotes the indicator function, equal to 1 when the condition inside the brackets is satisfied and 0 otherwise, and where $z_0 = 3$ in the selected configuration after an optimization process. The consensus fraction $\phi_i \in [0, 1]$ therefore represents the fraction of older backsteps whose normalized innovations exceed the selected significance threshold.

The raw score becomes

$$S_i^{(4,\mathrm{raw})} = \left(\sqrt{\sum_{k=2}^{K} w_k\, \tilde{v}_{i,k}^2}\right)(0.5 + \phi_i), \tag{4.31}$$

The first term corresponds to a weighted norm of the whitened innovation vector, while the second term increases the score when multiple backsteps simultaneously indicate anomalous behaviour. This

consensus mechanism was introduced to favour coherent multi-backstep signatures over isolated large innovations.

After per-satellite MAD normalization, this becomes the final tuned score:

$$S_i^{(4)} = \frac{S_i^{(4,\mathrm{raw})}}{\mathrm{MAD}_{\mathrm{fit}}\left(S^{(4,\mathrm{raw})}\right)}. \tag{4.32}$$

This formulation constitutes the best purely physical score found during development, where "purely physical" refers to a score based exclusively on algebraic transformations of the historical residual evolution, without the use of machine-learning models, additional transformations specifically designed to enhance sharp rises (reflecting the behaviour expected for most manoeuvres) or satellite-specific tuning. The present formulation subsequently serves as the baseline on top of which the learning-based ranker is built.

4.4.4.5. ML-corrected whitened vector ranker

The fifth stage starts from the tuned whitened vector score described above and applies a conservative learning-based correction. In the implementation, this is done by a plugin whose physical core is explicitly fixed to the best tuned whitened-vector configuration described in the previous subsection.

The first step is therefore unchanged:

$$S_i^{\mathrm{phys}} = S_i^{(4)}. \tag{4.33}$$

The objective of the learning-based stage is not to replace the physical detector, but to exploit information that is difficult to capture through a single handcrafted score. While the tuned whitened-vector score provides a robust summary of the innovation magnitude, different innovation patterns may still lead to similar physical scores. The learning-based component therefore attempts to identify combinations of innovation features that are statistically associated with manoeuvring and non-manoeuvring behaviour, allowing candidate epochs to be ranked more effectively.

Importantly, machine learning is not used as the primary detection mechanism. Instead, it is deliberately introduced as a second-stage refinement after the physical scores have been optimized. The reasoning is that physically interpretable metrics should explain as much of the manoeuvre signature as possible before moving on to data-driven methods. The learning-based model is therefore tasked with capturing residual patterns that are not adequately represented by the physical scores, combining the interpretability of physics-based detection with the capabilities of machine learning.

Additional descriptive features are then extracted from the absolute whitened innovations $|\tilde{\mathbf{v}}_i|$, including:

- the full absolute innovation vector,
- the largest three components,
- the mean and standard deviation across backsteps,
- the fractions of components above fixed thresholds.

These features are robustly scaled and passed to a small satellite-aware multilayer perceptron, which outputs a probability-like correction term $s_i^{\mathrm{ML}} \in [0, 1]$, based on the first 30% of nominal epochs. The output s_i^{ML} can be interpreted as a confidence measure indicating how strongly the observed innovation pattern resembles manoeuvring behaviour according to the examples seen during training. Values above 0.5 increase the physical score, whereas values below 0.5 reduce it. Rather than replacing the physical score, the model only applies a bounded multiplicative correction:

$$c_i^{\mathrm{ML}} = 1 + \kappa\,(2s_i^{\mathrm{ML}} - 1), \tag{4.34}$$

where

$$\kappa = (1 - \alpha_{\mathrm{ML}}). \tag{4.35}$$

In the selected configuration, $\alpha_{\mathrm{ML}} = 0.9$, so the learning-based branch only perturbs the physical baseline mildly. The final score at this stage is

$$S_i^{(5)} = S_i^{\mathrm{phys}}\, c_i^{\mathrm{ML}}. \tag{4.36}$$

The purpose of this stage is not to let the neural model dominate the decision, but to learn small corrections that improve the ordering of candidate epochs according to their likelihood of corresponding to a manoeuvre. In practice, the physical score already captures most manoeuvre signatures, but different innovation patterns may still produce similar score values. The learning-based correction exploits additional information contained in the innovation features to slightly increase the scores of epochs whose patterns resemble previously observed manoeuvres and decrease those that are more consistent with nominal behaviour. As a result, manoeuvring epochs tend to be ranked ahead of nominal epochs while preserving the robustness and interpretability of the physical score. Once again, the specific numerical values were obtained through validation experiments, keeping the configuration that yielded the best overall performance.

4.4.4.6. Onset-sensitive ranker

The sixth stage augments the previous score with a causal onset correction designed to increase sensitivity to sharp score rises. This stage was introduced after observing that some satellites did not perform well not because the adopted scoring strategy was incorrect, but rather because their scores were inherently high. For each satellite, the physical score S_i^{phys} is compared against the strictly past rolling median over a backward window of length W:

$$m_i = \mathrm{median}\{S_{i-W}, \ldots, S_{i-1}\}, \tag{4.37}$$

where m_i is the local baseline level of the score (its recent past median) and W is the length of the strictly-past window.

A robust scale estimate σ_i is obtained from the rolling median absolute deviation (MAD) over the same past window, scaled by a factor of 1.4826 to make it comparable to the standard deviation for normally distributed data.

The onset statistic is then the standardized excursion of the current score above its recent past level,

$$z_i^{\mathrm{onset}} = \frac{S_i^{\mathrm{phys}} - m_i}{\sigma_i}, \tag{4.38}$$

clipped to a maximum magnitude $z_{\max}$. Only positive excursions are rewarded, yielding the *onset prominence*:

$$p_i^{\mathrm{onset}} = \frac{\max(0,\, \mathrm{clip}(z_i^{\mathrm{onset}}, 0, z_{\max}))}{z_{\max}}. \tag{4.39}$$

The onset prominence $p_i^{\mathrm{onset}} \in [0, 1]$ is thus zero whenever the current score does not exceed its recent past median, grows with the size of the rise, and saturates at 1 for excursions of $z_{\max}$ robust standard deviations or more. It produces a multiplicative correction factor

$$c_i^{\mathrm{onset}} = 1 + \beta_{\mathrm{onset}} \left(2p_i^{\mathrm{onset}} - 1\right), \tag{4.40}$$

where $\beta_{\mathrm{onset}} \in [0, 1]$ is the correction strength, so that the factor lies in the interval $[1 - \beta_{\mathrm{onset}},\, 1 + \beta_{\mathrm{onset}}]$. With the selected parameters, $W = 41$, $\beta_{\mathrm{onset}} = 0.1$, and $z_{\max} = 3$. The score becomes

$$S_i^{(6)} = S_i^{\mathrm{phys}}\, c_i^{\mathrm{ML}}\, c_i^{\mathrm{onset}}. \tag{4.41}$$

Specific numerical values were again obtained through an optimization process. This stage improves responsiveness at manoeuvre onset while remaining causal and therefore compatible with near real-time operation.

4.5. Event detection from anomaly scores

4.5.1. Main logic

Once the anomaly score sequence $\{S_i\}$ has been computed, manoeuvre detection is performed directly at the level of individual target epochs.

A binary detection variable is first defined as

$$d_i = \mathbb{I}(S_i \geq \tau_{\mathrm{high}}), \tag{4.42}$$

where τ_{high} is the main decision threshold. In order to introduce hysteresis and avoid unstable switching once an event has started, a lower maintenance threshold is also defined as

$$\tau_{\text{low}} = \alpha\, \tau_{\text{high}}, \text{ where } \qquad 0 < \alpha < 1. \tag{4.43}$$

The event extraction logic is therefore based on two thresholds only:

- τ_{high}, used to initiate a detection,
- τ_{low}, used to maintain the detection while the score remains elevated.

A candidate event is initiated when the score exceeds the upper threshold τ_{high}. Once initiated, the event remains active as long as the score stays above the lower threshold τ_{low}.

In the final implementation, the persistence parameters for event initiation and confirmation are fixed to one sample:

$$K_{\text{enter}} = 1, \qquad K_{\text{confirm}} = 1. \tag{4.44}$$

such that event initiation and termination are effectively determined by a single-sample hysteresis criterion. This choice is motivated by the fact that each score already aggregates multi-backstep temporal information, making additional temporal persistence unnecessary and potentially detrimental for short-lived manoeuvre signatures. Consecutive samples within the same hysteresis region are grouped into a single detected event defining the manoeuvre interval.

4.5.2. Ground-truth event construction and one-to-one matching

In the evaluation stage, ground-truth (GT) manoeuvres are treated as representative point events derived from the manoeuvre ground truth data [66]. Each manoeuvre record is first parsed from the corresponding satellite text file, and the manoeuvre start epoch is retained as the reference timestamp.

Because some manoeuvres occur in closely spaced sequences, GT events that are separated by less than

$$\Delta t_{\text{merge}} = 2000 \text{ min} \tag{4.45}$$

are merged into a single GT event. In the final implementation, the representative epoch of the merged GT event is chosen as the *last* event of the merged group. Thus, if a merged group contains start times

$$t_1 < t_2 < \cdots < t_n \tag{4.46}$$

such that

$$t_{k+1} - t_k \leq \Delta t_{\text{merge}}, \qquad k = 1, \ldots, n-1, \tag{4.47}$$

the resulting GT representative time is

$$t^*_{\text{GT}} = t_n. \tag{4.48}$$

This choice is consistent with the final detection objective, since the score is expected to react most clearly once the full manoeuvre sequence has been completed.

Predicted events are built from the binary detection mask obtained from the thresholding and hysteresis logic described above. Since the final implementation uses just one sample for starting and confirming the event, each predicted event corresponds to one or more consecutive detected epochs. Each epoch is associated with the real interval

$$[t_{i-1}, t_i], \tag{4.49}$$

where t_{i-1} and t_i are the epochs of two consecutive TLEs. Therefore, a predicted event is represented by its start and end times in Unix format:

$$[t^{\text{start}}_{\text{pred}},\, t^{\text{end}}_{\text{pred}}]. \tag{4.50}$$

In the most common case of a single detected epoch, this interval is simply the TLE-to-TLE interval immediately associated with that score.

The correspondence between GT and detected events is established through one-to-one matching. For each GT event with representative time t^*_{GT}, a direct match is first attempted: the event is considered detected if there exists a predicted interval such that

$$t^{\mathrm{start}}_{\mathrm{pred}} \leq t^*_{\mathrm{GT}} \leq t^{\mathrm{end}}_{\mathrm{pred}}. \tag{4.51}$$

Since each predicted row represents the interval between two consecutive TLEs, this criterion effectively checks whether the manoeuvre is captured in the first TLE interval immediately following the representative GT epoch. In that sense, the method does not require waiting for several future TLEs: the intended detection is the first post-manoeuvre interval.

If no direct overlap exists, a secondary *snap-forward* rule is applied. The GT timestamp is shifted to the start time of the next predicted event, provided that this forward shift does not exceed

$$\Delta t_{\mathrm{snap}} = 60 \text{ h} = 3600 \text{ min}. \tag{4.52}$$

After this shift, overlap is tested again. Formally, if

$$t^{\mathrm{start}}_{\mathrm{pred},j} > t^*_{\mathrm{GT}} \tag{4.53}$$

is the nearest predicted-event start after the GT timestamp, the snapped time is accepted only if

$$t^{\mathrm{start}}_{\mathrm{pred},j} - t^*_{\mathrm{GT}} \leq \Delta t_{\mathrm{snap}}. \tag{4.54}$$

This rule makes the matching slightly more tolerant to small timing mismatches while still preserving the near-real-time interpretation of the detector. In practice, this accounts for cases in which the first post-manoeuvre TLE interval is too short for the score to develop a sufficiently strong anomaly, despite the manoeuvre already having occurred. Results show that this small fix is necessary to avoid missing detected manoeuvres on satellites with long TLE update cadence, but it was not a decisive increase in the performance of the algorithm.

Once the possible GT-to-predicted correspondences have been built, the final assignment is solved through a one-to-one bipartite matching. This ensures that each predicted event can explain at most one GT event, and each GT event can be matched to at most one predicted event. If N_{GT} denotes the number of GT events, N_{PRED} the number of predicted events, TP a true positive, FP a false positive, and FN a false negative, the event-level metrics are then computed as

$$\mathrm{TP} = N_{\mathrm{match}}, \qquad \mathrm{FP} = N_{\mathrm{PRED}} - N_{\mathrm{match}}, \qquad \mathrm{FN} = N_{\mathrm{GT}} - N_{\mathrm{match}}, \tag{4.55}$$

where N_{match} is the number of successful one-to-one matches.

4.5.3. Configuration selection and evaluation metrics

Event-level performance is quantified through precision, recall, and the F_β score:

$$\mathrm{Precision} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FP}}, \tag{4.56}$$

$$\mathrm{Recall} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FN}}, \tag{4.57}$$

$$F_\beta = (1 + \beta^2)\frac{\mathrm{Precision} \cdot \mathrm{Recall}}{\beta^2\,\mathrm{Precision} + \mathrm{Recall}}. \tag{4.58}$$

The score used throughout model selection is F_2, which assigns greater importance to recall and therefore penalizes missed manoeuvres more strongly than false alarms. This choice reflects the operational requirements of ISOPA, where retaining manoeuvre-contaminated data is considered more harmful than unnecessarily eliminating a small number of clean observations. Consequently, the detector is intentionally optimized to favour manoeuvre detection over the reduction of false positives.

The optimization is performed in two stages. In the first stage, a coarse grid of threshold percentiles and hysteresis factors is evaluated on the validation set [30]. The high threshold is derived from the validation score distribution as

$$\tau^{(s)}_{\mathrm{high}} = Q_p\left(\{S^{(s)}_i\}_{i \in \mathrm{VAL}}\right), \tag{4.59}$$

where p is the candidate threshold percentile. For each candidate pair (p, α), the validation-set performance is evaluated and the best configuration is chosen according to the global F_2 score.

Once the best first-stage configuration has been identified, per-satellite validation metrics are used to divide the active satellites into two groups:

- **high-performing satellites**, for which recall and precision are both acceptable,
- **low-performing satellites**, which systematically exhibit poorer behaviour under the same scoring logic.

In the implementation, a satellite is assigned to the high-performing group if

$$\mathrm{Recall} \geq 0.50 \qquad \text{and} \qquad \mathrm{Precision} \geq 0.30, \tag{4.60}$$

and to the low-performing group otherwise. These threshold values were selected heuristically based on engineering judgement, with a deliberate bias towards recall over precision, reflecting the operational priority of avoiding missed manoeuvre detections over reducing false alarms.

This separation is introduced because a small set of difficult satellites which do not perform well due to intrinsic TLE problems can dominate the global metric and bias the optimization towards overly conservative settings that do not maximize performance on the majority of the dataset. The second optimization stage therefore uses a weighted objective that evaluates both groups separately. The following Figures 4.8 and 4.9 illustrate this effect. In the case of the problematic satellite, the optimization process tends to increase the detection threshold in order to miss false manoeuvres that would otherwise be detected, despite the fact that the underlying scoring function already exhibits a large number of false positives. However, this threshold increase may simultaneously suppress weaker but real manoeuvre signatures in the high-performing satellite, causing additional false negatives. As a result, the global optimum becomes biased by the behaviour of a small number of problematic satellites. By introducing satellite-performance labels and, as discussed later, performance-dependent weighting factors, the influence of these satellites on the optimization process can be reduced, leading to a more balanced threshold selection across the entire dataset.

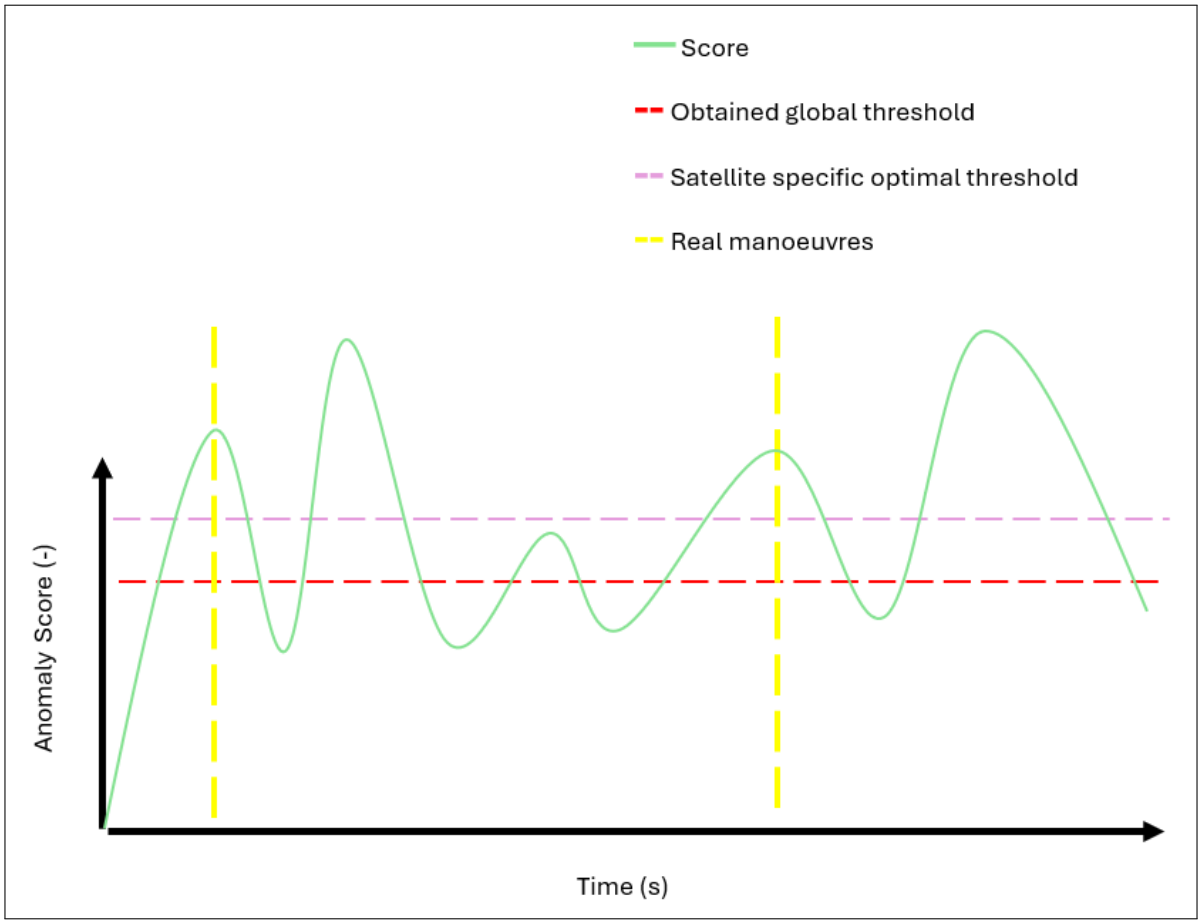

Figure 4.8: Example of scoring function for a problematic, low-performing satellite.

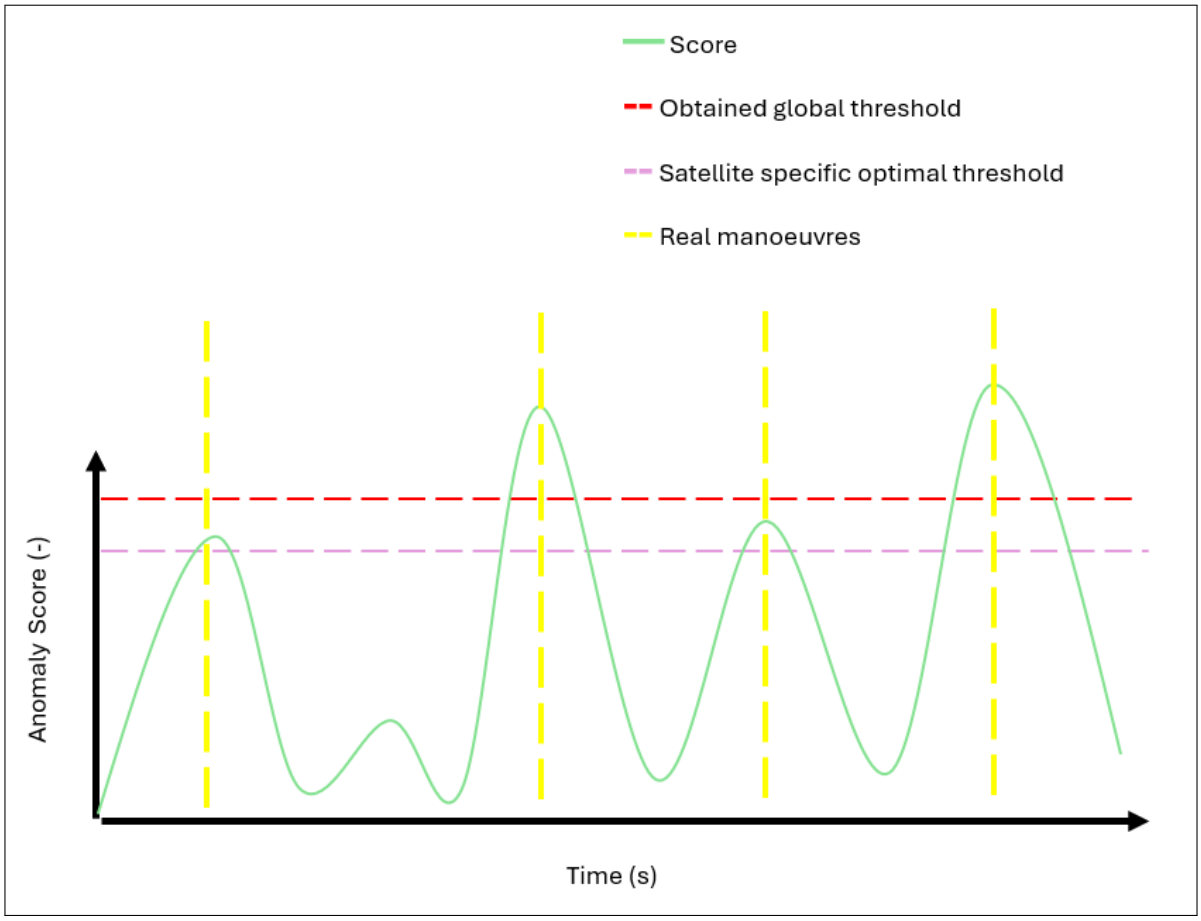

Figure 4.9: Example of scoring function for a nominal, high-performing satellite.

Let $F_2^{\text{high-perf}}$ and $F_2^{\text{low-perf}}$ denote the validation-set F_2 scores obtained on the corresponding subsets. To make both terms comparable, each metric is normalized over the full second-stage validation grid using percentile-based scaling:

$$\tilde{x} = \text{clip}\left(\frac{x - x^{(5)}}{x^{(95)} - x^{(5)}},\, 0,\, 1\right), \tag{4.61}$$

where $x^{(5)}$ and $x^{(95)}$ are the 5th and 95th percentiles of the metric across all second-stage configurations.

The final objective function maximized during the refined search is

$$\mathcal{J} = 0.6\,\tilde{F}_2^{\text{high-perf}} + 0.4\,\tilde{F}_2^{\text{low-perf}}. \tag{4.62}$$

The parameter grid used during the refined search was defined around the optimal values identified during the initial search, allowing a finer exploration of the most promising region of the parameter space. The two factors multiplying the F_2^{X} values were established arbitrarily, so that larger weight was given to high-performing satellites without making low-performing satellites completely irrelevant.

The only free detection parameters that remain to be calibrated are the threshold percentile p and the hysteresis factor α. The percentile p determines the primary detection threshold τ_{high} from the score distribution, while α defines the lower maintenance threshold according to

$$\tau_{\text{low}} = \alpha\,\tau_{\text{high}}. \tag{4.63}$$

Consequently, each candidate configuration is fully specified by the parameter pair

$$(p, \alpha), \tag{4.64}$$

which uniquely determines the corresponding hysteresis thresholds

$$(\tau_{\text{high}}, \tau_{\text{low}}). \tag{4.65}$$

4.6. Behaviour representation and satellite and manoeuvre classification

4.6.1. Purpose

Beyond detecting manoeuvre-like intervals, the objective is to characterize satellite behaviour over extended time horizons by analysing both the frequency and the operational nature of the detected manoeuvres. To achieve this, individual manoeuvre and non-manoeuvre episodes are first identified and subsequently aggregated into satellite-level descriptors. These descriptors are then used to classify satellites into three behavioural categories based on their manoeuvring frequency: “non-manoeuvring”, “occasional-manoeuvring”, and “frequent-manoeuvring” satellites.

A second classification level focuses on the nature of the manoeuvres themselves. Additional features are extracted to distinguish satellites that primarily perform regular station-keeping activities from those that predominantly execute larger orbital adjustments. Finally, each detected manoeuvre episode is analysed individually and assigned to one of these two categories, enabling both satellite-level and manoeuvre-level behavioural characterization.

4.6.2. Behavioural clustering using hierarchical agglomerative clustering based on Ward's linkage

To perform the satellite-level behavioural classification described in the previous section, a Hierarchical Agglomerative Clustering (HAC) approach based on Ward’s linkage criterion was employed [74]. This unsupervised statistical method groups observations according to their similarity without requiring predefined class labels. Unlike partitioning algorithms such as K-means, hierarchical clustering constructs a nested tree-like structure, known as a dendrogram, which provides a visual representation of how clusters are progressively merged throughout the clustering process.

The algorithm begins by considering each observation as an individual cluster. At every iteration, the two clusters that result in the smallest increase in within-cluster variance are merged. This procedure is repeated until all observations belong to a single cluster, producing a hierarchy of cluster relationships at different levels of granularity.

Ward’s linkage criterion determines which clusters should be merged by minimizing the increase in the total within-cluster sum of squares, also referred to as the error sum of squares. Given a cluster C containing n_C observations $\mathbf{o}_i$, the within-cluster variance can be expressed as

$$WCSS(C) = \sum_{\mathbf{o}_i \in C} \|\mathbf{o}_i - \boldsymbol{\mu}_C\|^2, \tag{4.66}$$

where $\boldsymbol{\mu}_C$ represents the centroid of the cluster. When two clusters A and B are considered for merging, Ward’s method evaluates the increase in variance produced by the union of both clusters:

$$\Delta(A, B) = WCSS(A \cup B) - WCSS(A) - WCSS(B). \tag{4.67}$$

The pair of clusters yielding the smallest value of $\Delta(A, B)$ is merged at each step. Consequently, Ward’s linkage tends to create compact and internally homogeneous clusters while avoiding the formation of elongated or poorly separated groups. The parameters included in $\mathbf{o}_i$ depend on the specific clustering task and are defined in the following two subsections (Subsections 4.6.2.1 and 4.6.2.2).

4.6.2.1. Clustering according to frequency of manoeuvres

In the first case, namely the analysis of manoeuvre frequency, the only feature provided to the clustering algorithm is the average number of manoeuvres per year. Using the total number of detected manoeuvres is deemed inappropriate, as the available observation period varies significantly among

satellites: some have more than fifteen years of data, whereas others have less than ten. To assess the impact of detection errors on the resulting classification, the clustering procedure is applied both to the reference manoeuvre catalogue and to the manoeuvres identified by the proposed detection algorithm, allowing the differences between the two classifications to be analysed.

4.6.2.2. Clustering according to nature of manoeuvres

Another way of clustering the satellites and their manoeuvres is according to their operational nature. Satellites typically perform manoeuvres following two distinct patterns. On the one hand, station-keeping activities are usually executed at relatively regular intervals and with a consistent direction in the orbital reference frame in order to maintain a desired orbital geometry. On the other hand, satellites may also perform manoeuvres for mission-specific purposes, collision avoidance, orbit transfers, or other operational requirements, which generally exhibit less temporal and directional structure. To distinguish between these behaviours, two complementary indicators were introduced: the *frequency index* and the *directionality index*.

The frequency index quantifies the regularity with which manoeuvres occur over time. Let t_i denote the starting epoch of manoeuvre event i, ordered chronologically. The temporal spacing between consecutive manoeuvres is defined as

$$\Delta t_i = t_{i+1} - t_i. \tag{4.68}$$

For a set of N manoeuvre events, the mean and standard deviation of the inter-manoeuvre intervals can be computed as

$$\mu_{\Delta t} = \frac{1}{N-1} \sum_{i=1}^{N-1} \Delta t_i, \tag{4.69}$$

$$\sigma_{\Delta t} = \sqrt{\frac{1}{N-1} \sum_{i=1}^{N-1} (\Delta t_i - \mu_{\Delta t})^2}. \tag{4.70}$$

From these quantities, the coefficient of variation is defined as

$$CV = \frac{\sigma_{\Delta t}}{\mu_{\Delta t}}, \tag{4.71}$$

which provides a normalized measure of the dispersion of the manoeuvre intervals. The frequency index is then computed as

$$F = \frac{1}{1 + CV}. \tag{4.72}$$

This formulation constrains the metric to the interval $[0, 1]$. Values close to one indicate highly periodic manoeuvre patterns with nearly constant temporal spacing, whereas values close to zero correspond to irregularly distributed manoeuvres. Consequently, satellites performing regular station-keeping activities are expected to exhibit significantly higher frequency indices than satellites executing sporadic operational manoeuvres.

The second metric, referred to as the directionality index, characterizes the consistency of the manoeuvre directions throughout the observation period. For each detected manoeuvre event, a directional vector is constructed in the RTN reference frame.The relative contribution of each RTN axis is represented by a non-negative, normalized vector

$$\mathbf{p}_i = \begin{bmatrix} p_{R,i} \\ p_{T,i} \\ p_{N,i} \end{bmatrix}, \qquad p_{R,i} + p_{T,i} + p_{N,i} = 1, \qquad p_{R,i},\, p_{T,i},\, p_{N,i} \geq 0, \tag{4.73}$$

where $p_{R,i}$, $p_{T,i}$ and $p_{N,i}$ describe the fraction of the manoeuvre activity attributed to the radial, transverse and normal directions, respectively.

This directional vector is an axis attribution derived from the same TLE-differencing residuals used for detection. For each epoch inside a detected event, and separately for each RTN axis, the back-innovations of both the position residual components ($\Delta R, \Delta T, \Delta N$) and the velocity residual components ($\Delta V_R, \Delta V_T, \Delta V_N$) are robustly standardized (median/MAD from the fit segment, with clipping) and aggregated across backsteps through a weighted root-mean-square. The position and velocity contributions are then blended into a single per-axis activity magnitude $a_{R,i}, a_{T,i}, a_{N,i} \geq 0$. Because these magnitudes are built from squared (RMS) quantities, they quantify *how much* residual energy is associated with each axis. Aggregating over the event and normalizing by the total activity yields the vector $\mathbf{p}_i$ above.

To evaluate the overall directional consistency, all event vectors $\mathbf{p}_i$ are first normalized to unit length, and their mean direction $\bar{\mathbf{p}}$ is computed (and likewise normalized). The similarity between each event vector and this mean direction is then evaluated through the cosine similarity, and the directionality index is defined as the mean cosine similarity between all event vectors and the group mean direction,

$$D = \frac{1}{N} \sum_{i=1}^{N} \cos(\theta_i), \tag{4.74}$$

where θ_i is the angle between manoeuvre vector $\mathbf{p}_i$ and the average manoeuvre direction $\bar{\mathbf{p}}$.

Because every $\mathbf{p}_i$ has non-negative entries and lies on the probability simplex, all event vectors reside in the non-negative octant of $\mathbb{R}^3$. Consequently, the angle between any two of them is at most $90°$, so that $\cos(\theta_i) \in [0, 1]$ and θ_i can never approach $180°$; values close to unity indicate that all manoeuvres repeatedly excite nearly the same RTN axis (or axis combination), while lower values indicate a larger directional dispersion.

Together, these two indicators provide a compact representation of manoeuvre behaviour. Satellites exhibiting both high frequency and high directionality indices are strong candidates for station-keeping behaviour, while satellites with low values in one or both metrics are more likely to perform manoeuvres driven by operational or mission-specific requirements. The results of this clustering procedure can be found under 5.6.1.2.

4.6.3. Individual classification of manoeuvres

As discussed previously, these two long-term satellite classifications provide a useful high-level characterization of satellite behaviour. However, from the perspective of an orbit-prediction framework, they offer limited operational information. For ISOPA, it is particularly important to determine whether a detected manoeuvre belongs to a larger, structured sequence of planned manoeuvres exhibiting consistent frequency and directional patterns, or whether it corresponds to an isolated orbital adjustment performed for purposes such as collision avoidance, orbit transfer, or other mission-specific objectives.

This distinction is relevant because the appropriate prediction strategy may differ between both cases. If a manoeuvre forms part of a recurring station-keeping pattern, future orbital evolution may be partially predictable once the underlying manoeuvring behaviour has been identified. Conversely, isolated manoeuvres typically introduce a discontinuity that is more appropriately handled by reinitializing the prediction process using post-manoeuvre data.

To enable this functionality, an additional classification layer was incorporated into the proposed framework. For each detected manoeuvre, a frequency index and a directionality index are computed and compared with those of neighbouring manoeuvres. A station-keeping manoeuvre sequence is declared when at least six consecutive manoeuvre episodes exhibit a frequency index within a factor of 0.5 to 2 of the median value of the group, and a directionality index within ±12° of the corresponding median directionality. These criteria are intended to identify groups of manoeuvres that display both temporal regularity and directional consistency, which are characteristic features of station-keeping operations.

It should be noted that the three parameters defining this classification procedure (the minimum sequence length, the allowable frequency-index deviation, and the allowable directionality-index devia-

tion) were selected empirically and may be refined in future work if additional validation data become available.

4.7. Evaluation of unknown satellites

Once trained and validated on the reference satellite set, the algorithm can be applied to previously unseen satellites. In this operational mode, the user specifies the satellite name, its NORAD identifier, and the time interval to be analysed. The algorithm then performs manoeuvre detection and behavioural characterization using the methodology described in the previous sections.

The outputs generated by the framework are:

- An anomaly-score time series covering the requested analysis period.
- The detection threshold associated with that score.
- The number of detected manoeuvre events.
- A classification of the detected manoeuvres according to their operational nature.
- A long-term behavioural label for the satellite, based on both the frequency and nature of its manoeuvres.

To improve robustness, the detection threshold is not computed exclusively from the requested analysis interval. Instead, it is estimated using a substantially longer historical period, extending up to ten years before the end of the selected time span whenever data are available. This approach reduces the sensitivity of the threshold estimation to short-term anomalies, isolated score spikes, or unusually quiet periods that could otherwise bias the detection process.

4.8. Output representation and integration into ISOPA

The proposed manoeuvre-detection pipeline is designed to operate alongside the TLE-based ISOPA framework. Its primary role is to provide complementary information that enables the prediction process to account for dynamical discontinuities and satellite-dependent behaviour.

The underlying motivation is that standard orbit-prediction approaches assume a continuous dynamical evolution of the spacecraft state. This assumption is violated whenever a manoeuvre occurs, since the post-manoeuvre trajectory is no longer fully represented by information contained in pre-manoeuvre observations. Consequently, training data affected by manoeuvres may degrade prediction performance. The proposed method therefore acts as a supervisory layer: once a manoeuvre has been detected, the TLE epochs affected by its influence are identified and excluded from the ISOPA training dataset, ensuring that the prediction model is trained primarily on nominal orbital behaviour. The effectiveness of this strategy can then be assessed by comparing the prediction accuracy of manoeuvre-aware and behaviour-agnostic versions of ISOPA.

To support this integration, the pipeline provides three types of outputs.

First, a binary manoeuvre-detection signal is generated at the TLE epoch level, indicating whether manoeuvre-like events have been identified within a given time interval. This information is subsequently used to determine which TLE epochs should be considered manoeuvre-contaminated and therefore excluded from the training process.

Second, all detected manoeuvres within the analysed time span are classified according to their operational nature. In particular, the method distinguishes between manoeuvres associated with routine station-keeping activities and larger orbital-adjustment manoeuvres. Although such information could be used to adapt the prediction strategy to different manoeuvre types, this functionality is beyond the scope of the present work. Consequently, all detected manoeuvres are treated identically during the integration with ISOPA.

Third, the algorithm produces a long-term behavioural characterization of the satellite. Based on the frequency and nature of its manoeuvres, the satellite is assigned to the corresponding behavioural classes derived from the training population, providing a high-level description of its operational behaviour.

5

Results and discussion

5.1. Experimental protocol

This chapter presents the results obtained with the manoeuvre detection pipeline described in Chapter 4. All experiments were conducted on the same set of 17 satellites, including 13 active satellites and 4 passive satellites. Passive satellites were automatically identified and excluded from the detection optimization stage, so that the reported event-detection results correspond to the active subset only. The passive set remained constant across all configurations and consisted of AJISAI, LAGEOS1, LAGEOS2, and STARLETTE.

The results are organized according to the seven successive score formulations introduced in Section 4.4:

1. baseline consistency score,
2. median back-innovation score,
3. whitened vector innovation score,
4. tuned whitened vector innovation score,
5. ML-corrected whitened vector ranker,
6. onset-sensitive ranker,
7. onset-sensitive ranker with a satellite-specific (SARAL) temporal refinement.

The final configuration includes an additional satellite-specific refinement motivated by the analysis of SARAL, which exhibited a localized validation failure despite showing significantly better behaviour in the test period. Unlike the previous score formulations, this refinement does not introduce a general modification of the detection strategy, but rather addresses a clearly identifiable period of degraded TLE consistency affecting this specific satellite. The details of this analysis and the motivation for the refinement are discussed in Section 5.3.6.

The main evaluation metric is the event-level F_2 score, which prioritizes recall over precision. The code reports four main aggregate views:

- validation performance on all active satellites,
- test performance on all active satellites,
- validation performance on the high-performing subset,
- test performance on the high-performing subset.

The last configuration should be interpreted separately from the previous ones, as it incorporates a targeted refinement derived from the failure-case analysis of SARAL rather than a global modification of the scoring formulation.

Table 5.1 summarizes the global evolution of the method across the seven configurations.

Configuration	VAL F_2	TEST F_2	VAL HIGH-PERF F_2	TEST HIGH-PERF F_2	High-perf./Low-perf.
Baseline	0.5909	0.6699	0.7120	0.7845	6 / 7
Median innovation	0.6613	0.7249	0.7725	0.8092	9 / 4
Whitened innovation	0.7045	0.7596	0.7415	0.7942	11 / 2
Tuned whitened innovation	0.7070	0.7691	0.7434	0.8000	11 / 2
ML-corrected WVI	0.7094	0.7666	0.7484	0.8022	11 / 2
Onset-sensitive ranker	0.7143	0.7794	0.7527	0.8070	11 / 2
Onset + SARAL refinement	0.7359	0.7892	0.7510	0.8134	12 / 1

Table 5.1: Global evolution of the manoeuvre detection performance across the seven score formulations.

A first general trend can already be observed from Table 5.1. The transition from the baseline representation to innovation-based representations provides the largest performance improvement, while subsequent refinements yield smaller but still meaningful gains.

A second important observation, briefly introduced earlier, is that the test performance is consistently higher than the validation performance across all configurations. This behaviour supports the robustness of the optimization procedure and suggests that the selected thresholds do not overfit specific characteristics of the validation interval. Instead, they generalize well to unseen future data, which is a desirable property for operational deployment.

It is worth noting that the primary objective throughout the optimization process was to maximize the validation (F_2) score. As shown in Table 5.1, the final configuration achieves the highest (F_2) values on both the validation and test sets.

However, the same trend is not observed for the (F_2) scores of the high-performing satellites. In some cases, intermediate configurations achieve slightly better results on this subset than the final configuration. This behaviour is expected. As the detection framework evolves, an increasing number of satellites are reassigned from the low-performing group to the high-performing group. Consequently, the high-performing subset gradually incorporates more challenging satellites, which can reduce the aggregate (F_2) score of that group despite the overall improvement in global detection performance. In other words, the apparent degradation is primarily a consequence of the changing composition of the subset rather than a deterioration of the detector itself.

5.2. Baseline detection performance

The baseline experiment uses the simplest possible score, namely the raw consistency magnitude associated with backstep 1. Its purpose is to establish a physically interpretable reference against which all later refinements can be compared.

The baseline achieves

$$F_2^{\mathrm{VAL,ALL}} = 0.5909, \qquad F_2^{\mathrm{TEST,ALL}} = 0.6699, \tag{5.1}$$

with corresponding high-performing-subset values

$$F_2^{\mathrm{VAL,HIGH-PERF}} = 0.7120, \qquad F_2^{\mathrm{TEST,HIGH-PERF}} = 0.7845. \tag{5.2}$$

These numbers already indicate that even the simplest consistency score is able to capture a substantial fraction of manoeuvre events. However, its main weakness is precision. On the validation set, the baseline produces 1342 predicted events for 572 GT events, with 913 false positives. This large number of false alarms reflects the fact that the raw score is highly sensitive to isolated TLE noise and does not exploit the redundancy available in the backstep structure.

At the satellite level, the baseline produces a strongly heterogeneous behaviour. Some satellites already perform reasonably well: CRYO2, SARAL, SEN3A and SEN3B all exceed $F_2 = 0.63$ on validation, while HY-2A and SWOT1 remain particularly challenging. On the test set, several satellites improve markedly, including JASO3 ($F_2 = 0.931$), SARAL ($F_2 = 0.921$) and SEN6A ($F_2 = 0.957$). This already illustrates one of the key properties of the dataset: detection difficulty is strongly satellite-dependent.

Figures 5.1 and 5.2 illustrate two examples of the baseline configuration. From this point onward, the black curve represents the anomaly score evolution, dashed vertical orange lines indicate ground-truth manoeuvres that were not detected (False Negatives, FNs), whereas dashed vertical green lines indicate correctly detected ground-truth manoeuvres (True Positives, TPs). The dashed horizontal red line represents the optimized high detection threshold (τ_{high}) corresponding to the configuration under study, with score values exceeding this threshold classified as manoeuvres according to the logic described previously.

Figure 5.1 illustrates the behaviour of a well-performing satellite, SEN3B. Although the score exhibits a consistent noise floor during nominal periods, most ground-truth manoeuvres coincide with clear score excursions, with peaks exceeding the detection threshold and therefore being successfully identified by the detector. This good temporal agreement between score increases and known manoeuvre events explains the high validation performance achieved by this satellite.

In contrast, Figure 5.2 shows the behaviour of a challenging case, SWOT1, for which the validation F_2 score is only 0.254. During the initial part of the validation interval, the score presents a persistently elevated and irregular behaviour, making it difficult to distinguish genuine manoeuvre signatures from nominal fluctuations. Later in the sequence, the score decreases significantly and remains most of the time below the detection threshold. As a consequence, many manoeuvres indicated by the ground-truth annotations do not produce sufficiently large score increases and are therefore missed by the detector.

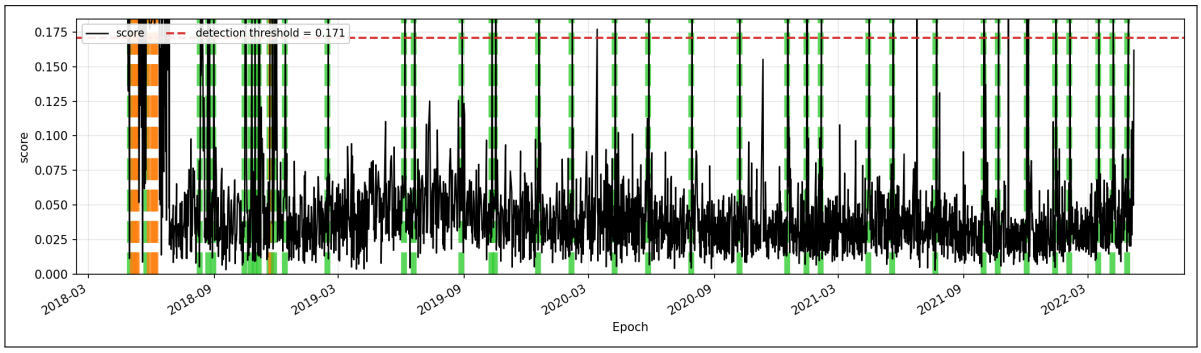

Figure 5.1: SEN3B score evolution for the validation set on the baseline configuration

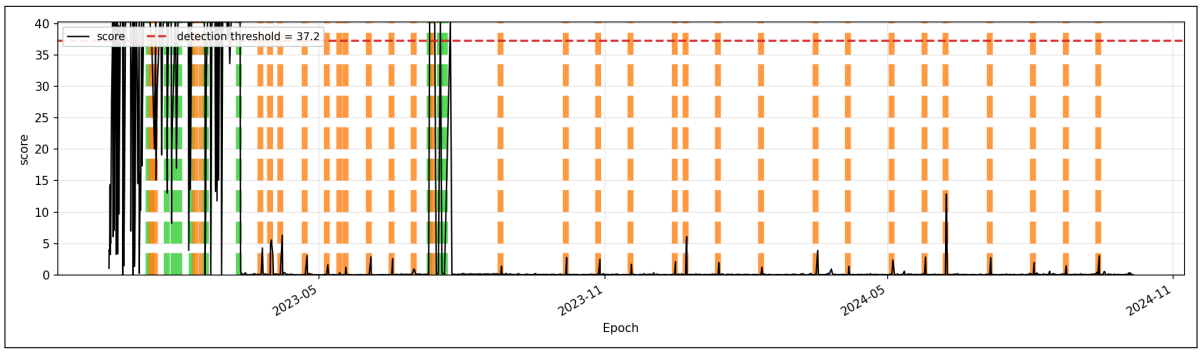

Figure 5.2: SWOT1 score evolution for the validation set on the baseline configuration

5.3. Evolution of the scoring methodology

5.3.1. From the baseline to median back-innovation

The first major improvement is obtained by replacing the raw backstep-1 magnitude by the median back-innovation score. This raises the global validation performance from

$$0.5909 \rightarrow 0.6613 \tag{5.3}$$

and the test performance from

$$0.6699 \rightarrow 0.7249. \tag{5.4}$$

On the high-performing subset, the score improves from 0.7120 to 0.7725 in validation and from 0.7845 to 0.8092 in test.

This confirms that comparing the current behaviour to the recent historical pattern is already much more robust than using a single raw consistency magnitude. The number of high-performing satellites increases from 6 to 9. The gain is therefore not limited to the aggregate metrics: it also indicates a more consistent detection performance across different satellites.

5.3.2. Whitened vector innovation

The introduction of the whitened vector innovation score produces the next major jump in performance. The global F_2 increases to

$$F_2^{\mathrm{VAL,ALL}} = 0.7045, \qquad F_2^{\mathrm{TEST,ALL}} = 0.7596, \tag{5.5}$$

while the high-performing-subset values become

$$F_2^{\mathrm{VAL,HIGH-PERF}} = 0.7415, \qquad F_2^{\mathrm{TEST,HIGH-PERF}} = 0.7942. \tag{5.6}$$

The number of high-performing satellites rises again, reaching 11, while only HY-2A and SARAL remain in the low-performing subset.

By jointly considering the innovation pattern across multiple backsteps and normalizing each component according to its nominal variability, the score becomes less sensitive to raw scale differences and more responsive to genuinely anomalous multi-backstep behaviour. The improvement is therefore both quantitative and methodological.

The validation performance of several previously difficult satellites improves substantially. For example, HY-2A reaches $F_2 = 0.541$, a clear increase with respect to the baseline (where it showed $F_2 = 0.199$), and SWOT1 reaches $F_2 = 0.714$, becoming one of the best-behaved satellites in validation. On the test set, the method achieves very strong results for several satellites, including CRYO2, SARAL, SEN3A, SEN3B and SEN6A. Figures 5.4 and 5.6 illustrate the improvement on the test set of SWOT1 satellite when transitioning from the median back-innovation score to the whitened vector innovation formulation.

As shown in Figures 5.3 and 5.4, the score distribution in the validation set for the SWOT1 satellite in the median back-innovation approach differed significantly in scale from that in the test set. Since detection thresholds are calibrated exclusively on validation and directly transferred to test, this mismatch leads to a complete failure in detection for the test set, resulting in $F_2 = 0$.

In contrast, Figures 5.5 and 5.6 show that the whitened vector innovation produces a much more consistent score distribution across validation and test. As a result, the thresholds learned on validation remain meaningful when applied to the test set, yielding a substantially improved performance of $F_2 = 0.895$.

This example highlights that the main advantage of the whitened formulation is not only an improvement in separability, but also a significantly enhanced robustness to distributional shifts between validation and test.

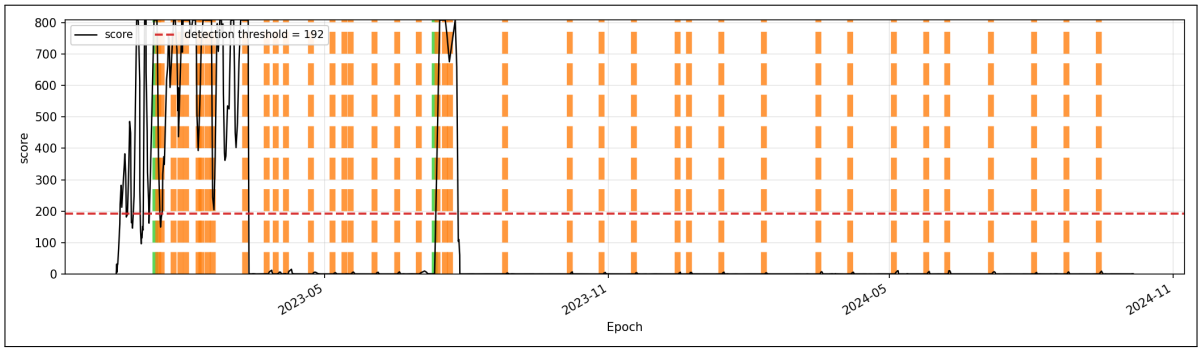

Figure 5.3: SWOT1 score evolution for the validation set for the median back-innovation approach

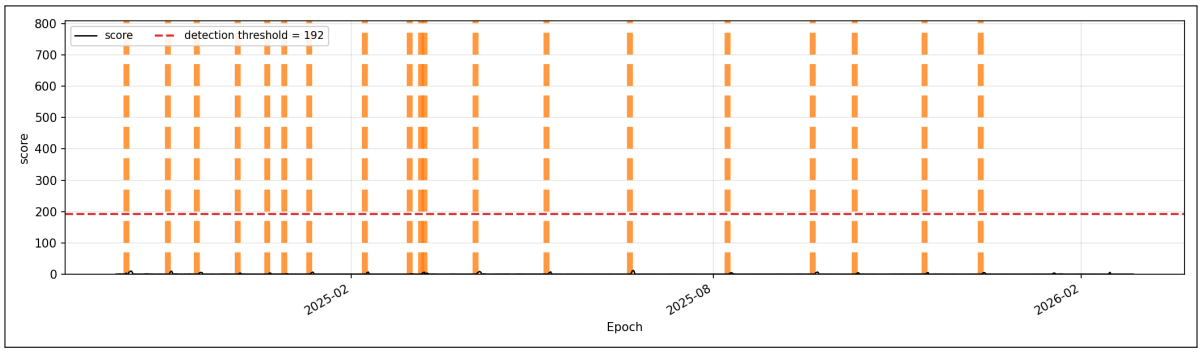

Figure 5.4: SWOT1 score evolution for the test set for the median back-innovation approach

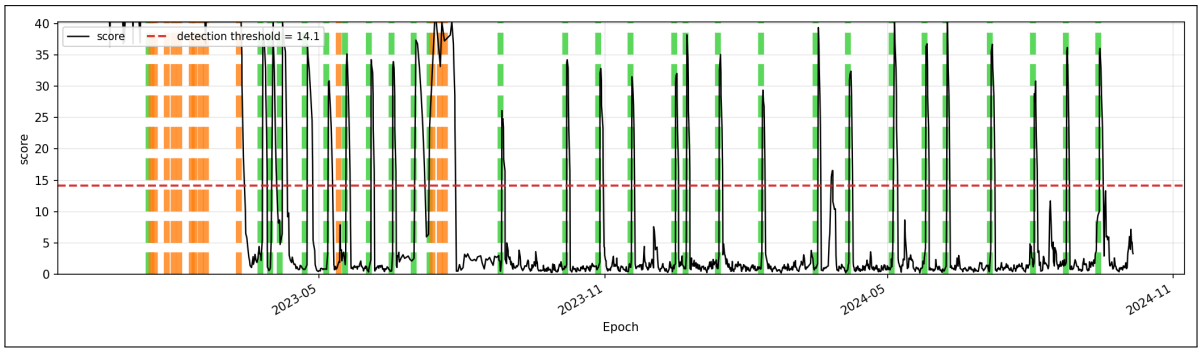

Figure 5.5: SWOT1 score evolution for the validation set for the whitened vector innovation approach

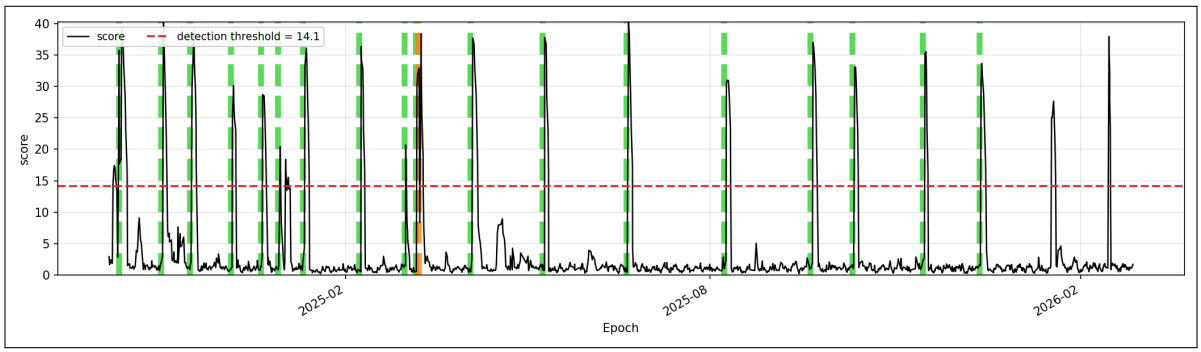

Figure 5.6: SWOT1 score evolution for the test set for the whitened vector innovation approach

5.3.3. Physically tuned whitened innovation

The tuned whitened innovation introduces weighted backsteps, stricter clipping bounds and a consensus factor. This produces a modest but consistent further improvement:

$$F_2^{\mathrm{VAL,ALL}} = 0.7070, \qquad F_2^{\mathrm{TEST,ALL}} = 0.7691. \tag{5.7}$$

The corresponding high-performing-subset scores are

$$F_2^{\mathrm{VAL,HIGH-PERF}} = 0.7434, \qquad F_2^{\mathrm{TEST,HIGH-PERF}} = 0.8000. \tag{5.8}$$

This stage produces the best-performing score formulation based solely on the physically motivated residual information, before introducing machine-learning corrections or additional refinements based on observed temporal behaviour. While the improvement on the validation set is relatively small, the larger gain observed on the test set suggests that this formulation enhances generalization to previously unseen data, rather than simply yielding a better fit to the validation set.

5.3.4. Hybrid physical--ML correction

The ML-corrected whitened score introduces a conservative neural correction on top of the tuned physical score. The resulting aggregate metrics are

$$F_2^{\mathrm{VAL,ALL}} = 0.7094, \qquad F_2^{\mathrm{TEST,ALL}} = 0.7666, \tag{5.9}$$

with

$$F_2^{\mathrm{VAL,HIGH-PERF}} = 0.7484, \qquad F_2^{\mathrm{TEST,HIGH-PERF}} = 0.8022. \tag{5.10}$$

Compared with the tuned physical score, the hybrid score slightly improves validation performance but slightly degrades test performance:

$$0.7691 \rightarrow 0.7666 \quad \text{on TEST (ALL).} \tag{5.11}$$

This finding suggests that most of the useful structure is already captured by the physical score, and that the data-driven correction is only able to provide marginal refinements. In other words, the ML branch remains helpful, but only as a bounded correction rather than a dominant component, confirming that most of the exploitable structure is already captured by the physical formulation. Since model selection is driven by validation (F_2), this configuration remains preferable despite the small reduction in test-set performance, as it provides the highest score according to the optimization criterion adopted in this work.

5.3.5. Onset-sensitive refinement

The onset-sensitive ranker produces the best general score formulation prior to the SARAL-specific refinement. Its performance becomes

$$F_2^{\mathrm{VAL,ALL}} = 0.7143, \qquad F_2^{\mathrm{TEST,ALL}} = 0.7794, \tag{5.12}$$

with

$$F_2^{\mathrm{VAL,HIGH-PERF}} = 0.7527, \qquad F_2^{\mathrm{TEST,HIGH-PERF}} = 0.8070. \tag{5.13}$$

This stage slightly improves both validation and test performance compared to the plain ML-corrected score. The gain is not dramatic, but it is consistent across all four aggregate views. This suggests that explicitly favouring sharp score rises helps align the detections more closely with manoeuvre onset without damaging the stability of the overall score.

5.3.6. Failure-case analysis and SARAL refinement

Among all satellites, SARAL represents the clearest example of a time-localized failure mode. In the general configurations, SARAL performs very poorly in validation despite behaving much better in test. In the plain whitened and hybrid formulations, it remains one of the two low-performing satellites, together with HY-2A. For instance, in the best, final innovation configuration, SARAL reaches only

$$F_2^{\mathrm{VAL}} = 0.427, \tag{5.14}$$

with 21 true positives and 34 false negatives in validation out of 55 GT events, even though its test performance is already much stronger in the later interval.

The observed behaviour on the validation set motivated a dedicated failure-case analysis. The following interpretation is introduced here, after the global methodology has already converged, because it is not part of the general score design but rather a refinement driven by a specific satellite-dependent anomaly.

Therefore, the resulting score can still be expressed in the same generic form as

$$S_i^{(7)} = S_i^{\text{phys}}\, c_i^{\text{ML}}\, c_i^{\text{onset}}. \tag{5.15}$$

More specifically, for the SARAL satellite, a portion of the dataset spanning from February 2013 to October 2015 was excluded from the analysis. During this period, the TLE-derived signals exhibit unusually large noise spikes and manoeuvres that are significantly harder to detect using the proposed method.

This behaviour is consistent with the operational history of the mission. SARAL operated in a 35-day repeat orbit with active station-keeping during its nominal phase, and from March 2015 onward, the satellite experienced reaction wheel degradation, leading to non-nominal orbit control and increased ground track deviations [75, 76].

After this period, the satellite transitioned towards a regime with reduced orbit control, eventually entering a drifting phase in 2016, where routine station-keeping manoeuvres were no longer performed [77].

From late 2015 onwards, the TLE behaviour becomes significantly more regular, and manoeuvres are more clearly detectable, resulting in improved performance of the proposed detection method.

The effect of the refinement is very evident at the satellite level. After applying the SARAL-specific filtering strategy, the satellite achieves perfect recall on the validation interval, detecting all ground-truth manoeuvres:

$$\text{TP} = 9, \qquad \text{FP} = 3, \qquad \text{FN} = 0, \tag{5.16}$$

which corresponds to

$$\text{Precision} = 0.750, \qquad \text{Recall} = 1.000, \qquad F_2 = 0.937. \tag{5.17}$$

With respect to the global effect of the change, the SARAL refinement produces the best result of all seven configurations:

$$F_2^{\text{VAL,ALL}} = 0.7359, \qquad F_2^{\text{TEST,ALL}} = 0.7892. \tag{5.18}$$

The validation gain is especially pronounced with respect to the onset-sensitive configuration without SARAL filtering:

$$0.7143 \rightarrow 0.7359, \tag{5.19}$$

while the test gain is smaller but still positive:

$$0.7794 \rightarrow 0.7892. \tag{5.20}$$

A particularly important effect is that the number of high-performing satellites increases from 11 to 12, leaving only HY-2A in the low-performing subset. This confirms that a substantial portion of the residual error was driven by a localized and identifiable data-quality issue rather than by a fundamental limitation of the score itself. Most satellites exhibit different behavioural regimes over their time spans which, if carefully accounted for, could potentially improve the overall detection performance. Nevertheless, the SARAL case is the only one in which such a refinement is considered justified. This is because its score structure clearly reveals a specific temporal interval that systematically degrades performance, rather than a general limitation of the method. In contrast, for the remaining satellites, the observed variability is either less structured or more uniformly distributed over time, making it difficult to isolate a well-defined segment whose exclusion would be both meaningful and consistent.

Figures 5.7 and 5.8 show the change in the score evolution of the validation set of SARAL satellite before and after the refinement.

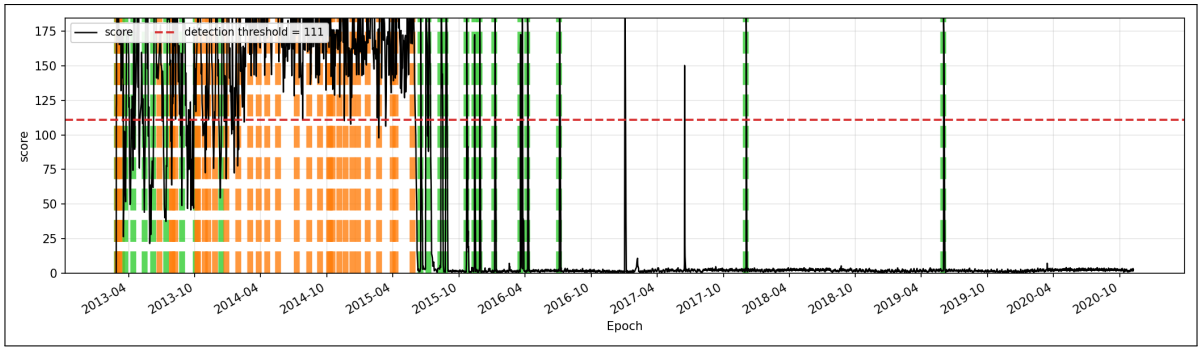

Figure 5.7: SARAL score evolution for the validation set for the pre-refinement approach

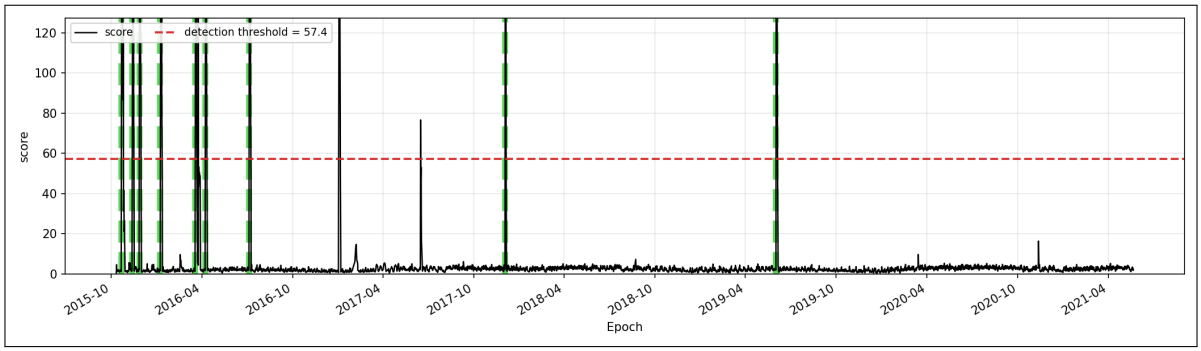

Figure 5.8: SARAL score evolution for the validation set for the post-refinement approach

The last satellite that remains classified as low-performing is HY-2A. It has been observed that its behaviour is not driven by a single localized anomaly, but rather by broader changes in the statistical properties of its TLE time series, as it will be shown in a dedicated subsection 5.5.3.

In particular, a clear regime shift is identified around March 2014. Before this date, the TLE sequence is relatively stable, whereas afterwards it becomes significantly noisier. The typical absolute mean-motion variation between consecutive TLEs increases by approximately one order of magnitude (from $\sim 10^{-6}$ to $\sim 10^{-5}$ rev/day), and the fraction of TLEs with large drag-related coefficients (e.g. $|BSTAR| > 5 \times 10^{-4}\ R_E^{-1}$) rises from below 3% to more than 40% in certain periods. Additionally, the temporal spacing between TLE updates also changes, increasing from roughly 7 h to about 8.7 h during the degraded phase.

This behaviour suggests that the issue is a loss of short-term consistency in the TLE sequence itself. From the perspective of a propagation-consistency-based detector, this results in persistently elevated residuals that are difficult to distinguish from genuine manoeuvre signatures.

Unlike the SARAL case, however, this degradation cannot be attributed to a single well-defined temporal interval, but instead spans multiple periods (e.g. 2014–2018 and 2019), interleaved with more stable phases. For this reason, applying a similar temporal filtering strategy would not be straightforward or well justified.

As a result, HY-2A is retained as a difficult satellite within the general evaluation, and its behaviour is interpreted as a limitation of the underlying TLE data quality rather than of the detection methodology itself.

5.4. Event detection performance

All the attempted configurations use a percentile-based thresholding strategy applied separately to each satellite, such that the same threshold percentile is used across the dataset while the numerical threshold value is computed from the score distribution of each individual object. In the best case, the optimized event-detection performance was obtained with a threshold percentile of 46.25 and a hysteresis coefficient $\alpha = 0.48$.

Under this configuration, the event-level performance on the validation set reaches a precision of 0.5609, a recall of 0.7981, and an F_2 score of 0.7359, corresponding to 419 correctly detected manoeuvre events out of 525 ground-truth events, with 328 false positives and 106 false negatives. When the same configuration is transferred to the test split, performance improves further, reaching a precision of 0.6347, a recall of 0.8404, and an F_2 score of 0.7892, with 516 true positives, 297 false positives, and 98 false negatives over a total of 614 test events.

Figure 5.9 shows the per-satellite precision–recall distribution on the validation set for the final selected configuration. Several satellites cluster in a region of both high recall and moderate-to-high precision, whereas others exhibit a more asymmetric behaviour.

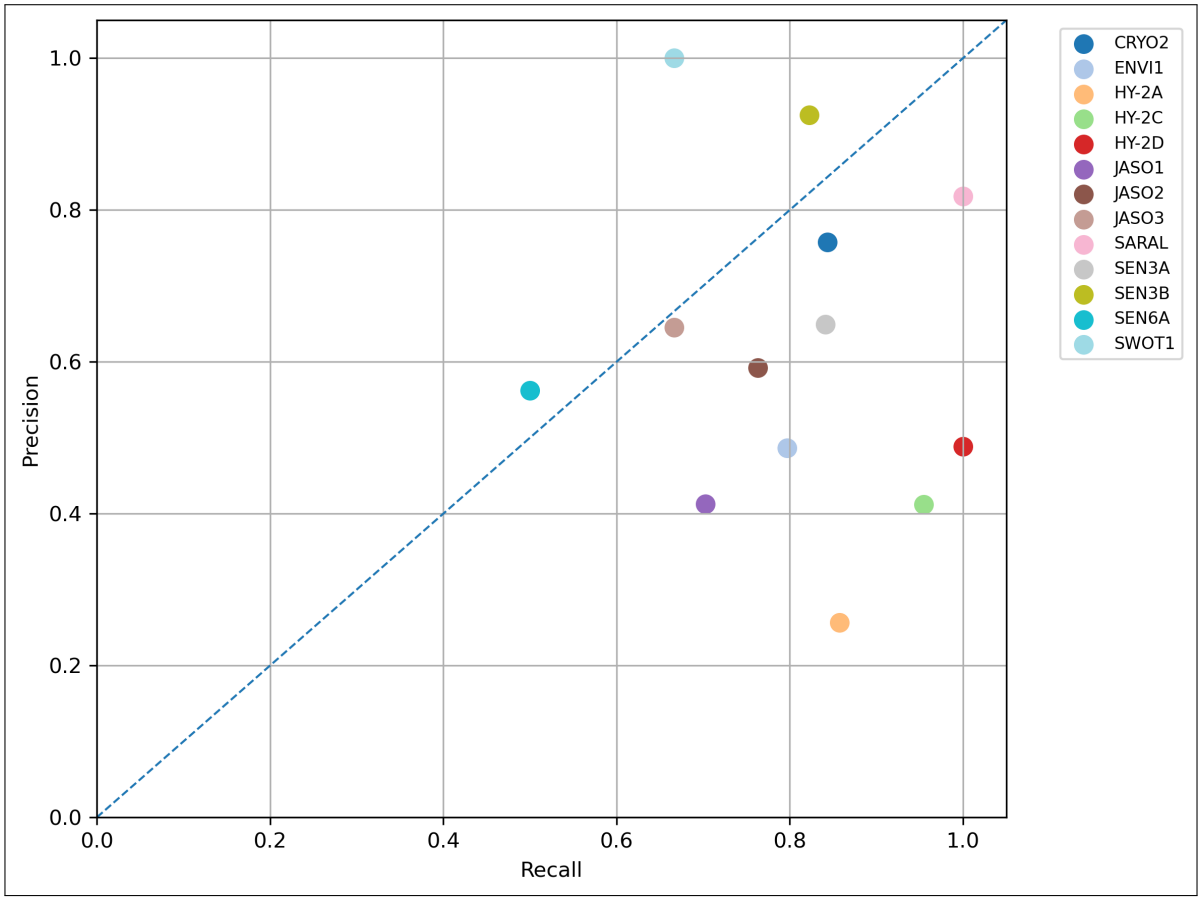

Figure 5.9: Per-satellite precision–recall scatter plot on the validation set for the final selected event-detection configuration.

The same picture for the test split is shown in Figure 5.10. The overall distribution shifts favourably with respect to validation, with several satellites moving closer to the upper-right region of the plot. The figure confirms that a small subset of satellites remain responsible for most of the residual degradation in precision, which again points to the presence of object-specific dynamical or data-quality effects that cannot be completely removed by a single global configuration.

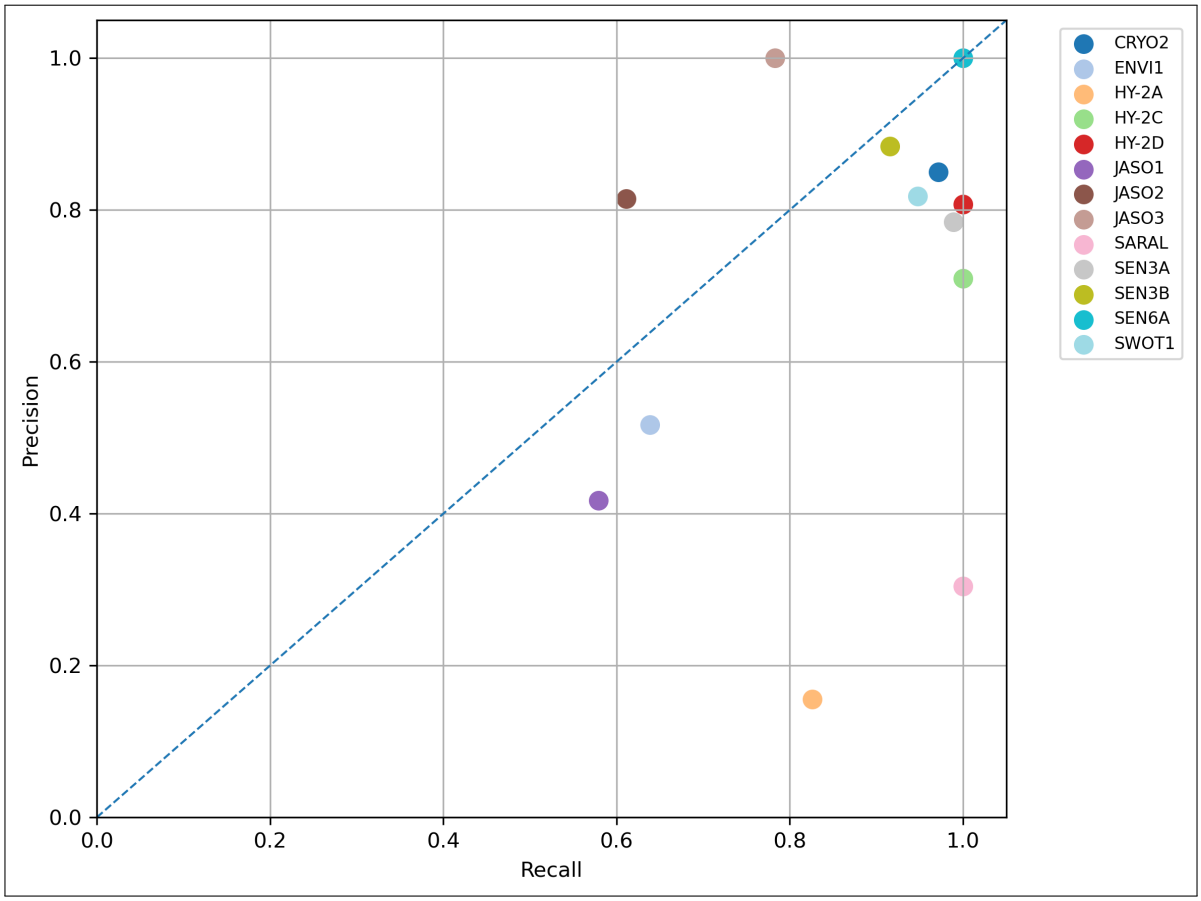

Figure 5.10: Per-satellite precision–recall scatter plot on the test set for the final selected event-detection configuration.

Figure 5.11 compares the final F_2 performance of each satellite between the validation and test intervals. Most satellites lie above the diagonal line, indicating that their performance improves when the optimized configuration is applied to the unseen test period. This behaviour suggests that the selected thresholds do not overfit the validation interval, but instead capture generalizable properties of the score distributions. The absence of a systematic degradation in test performance is particularly relevant for an operational detection framework, where future unseen observations must be processed without additional tuning.

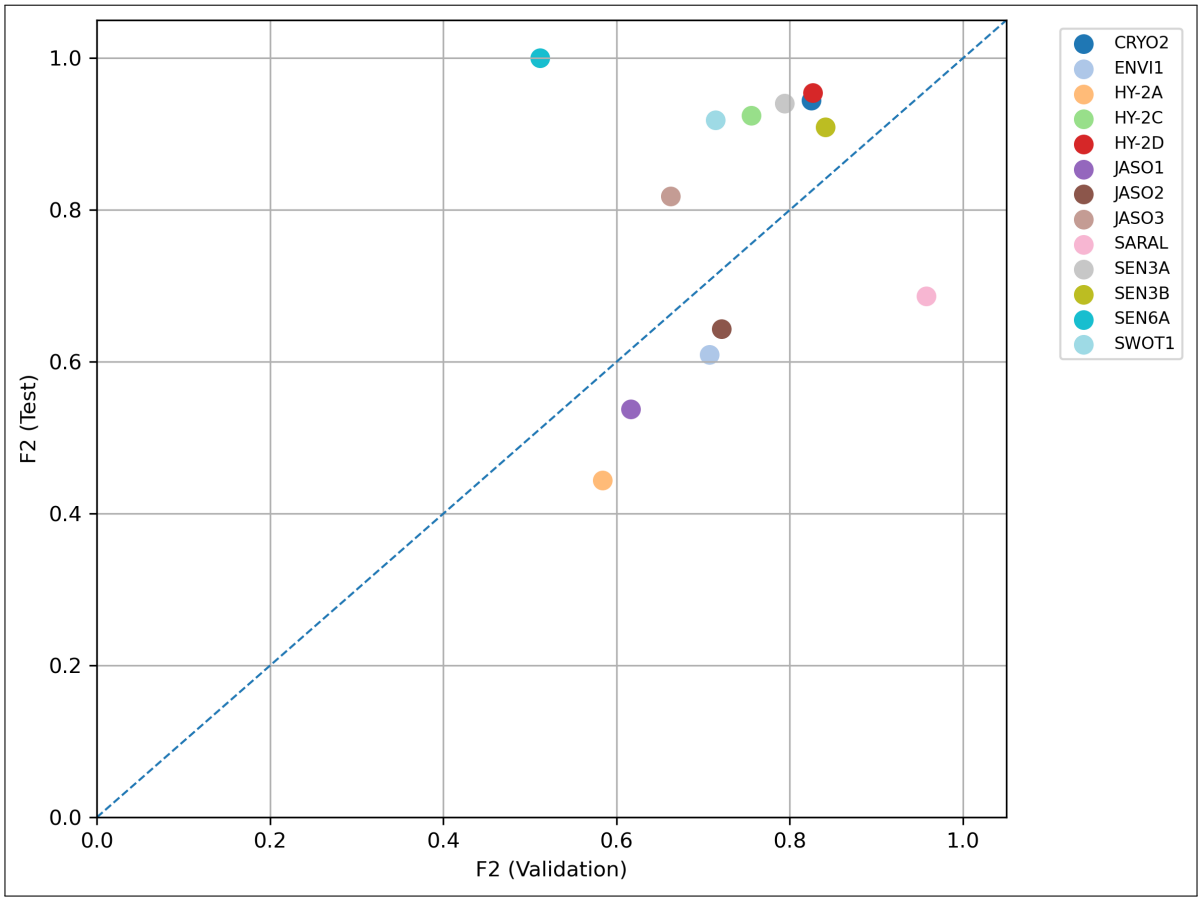

Figure 5.11: F_2 values for validation and test splits for all active satellites

Figure 5.12 provides a direct comparison of the final F_2 performance across all active satellites. All analysed objects achieve $F_2 > 0.5$, indicating that the detector consistently performs above a random decision baseline for every satellite considered. Furthermore, only five satellites remain below $F_2 = 0.7$, while several objects achieve values close to $F_2 = 0.9$. Considering that the same percentile-based thresholding strategy is applied across satellites with different missions, orbital characteristics, and data-quality conditions, these results demonstrate the general applicability of the proposed approach. The remaining performance differences are therefore attributed mainly to satellite-dependent TLE characteristics rather than to a lack of robustness of the detection framework itself.

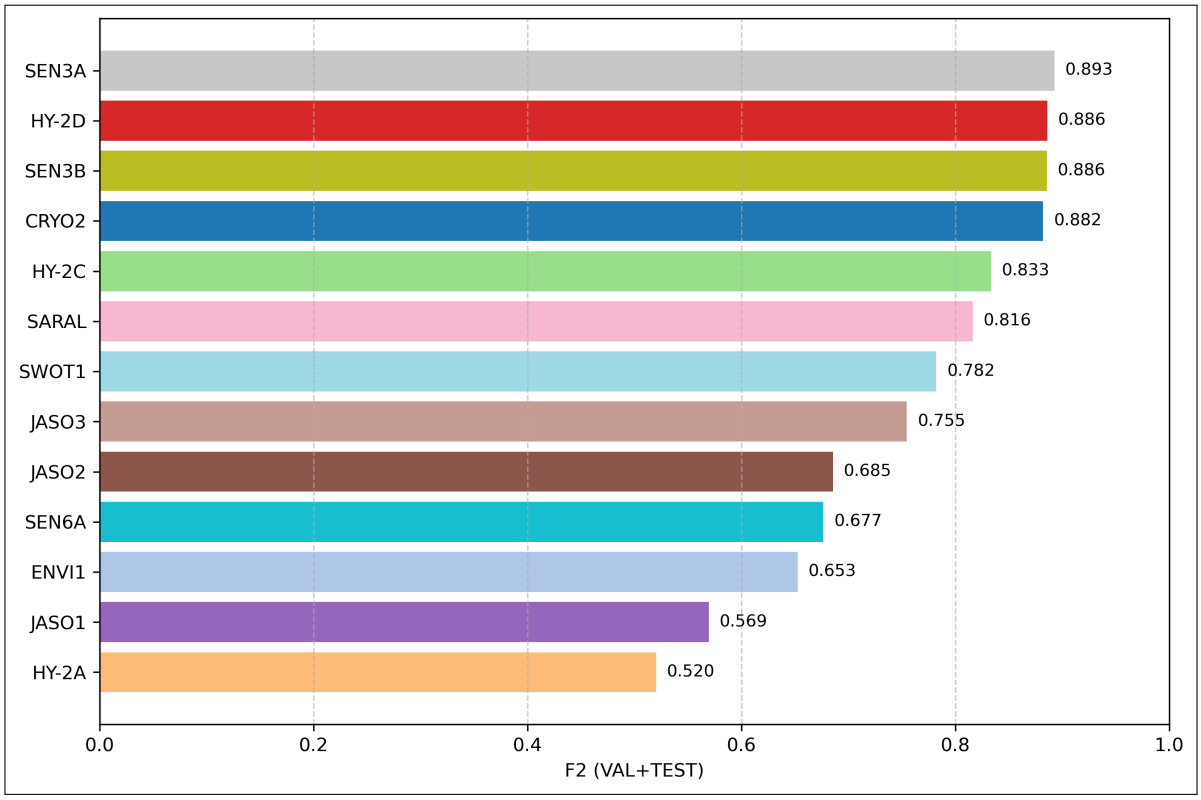

Figure 5.12: Final F_2 values for for all active satellites, on validation and test splits combined

Specific metrics, such as the number of TP, FP, FN, as well as final values for precision, recall, and F_2 can be observed in Appendix C.

5.5. Satellite-level qualitative analysis

The aggregate metrics show a clear progression, but the per-satellite plots provide essential insight into why the different configurations behave as they do. Three broad classes of behaviour are observed.

5.5.1. High-performing satellites

Satellites such as CRYO2, SEN3A, SEN3B and, in later stages, JASO3 exhibit a regular residual structure, clear score spikes around manoeuvre epochs, and relatively few false alarms. These objects tend to enter the high-performing subset already in early configurations and remain there throughout the progression. Their plots typically show that the main limitation of the early methods is not recall but excessive background variability, which is gradually reduced by the innovation-based and whitened scores. Figures 5.13 and 5.14 show the evolution of the score of the test set of CRYO2 on the baseline and final approaches, and Figures 5.15 and 5.16 show the evolution of the score on the validation set of SEN3B for the baseline and final approaches. As it can be observed, the behaviour is already good enough to yield a good result on the baseline approach, but the evolution on the pipeline keeps most of the detected manoeuvres while reducing substantially background noise.

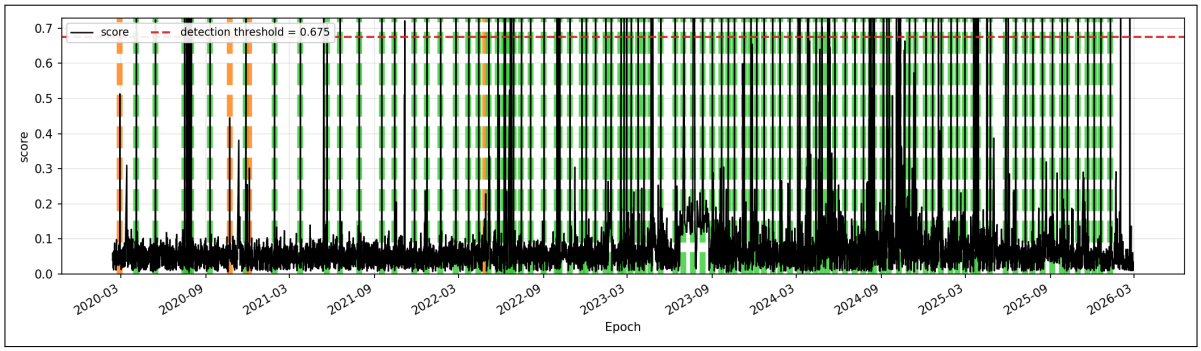

Figure 5.13: CRYO2 score evolution for the test set for the baseline approach

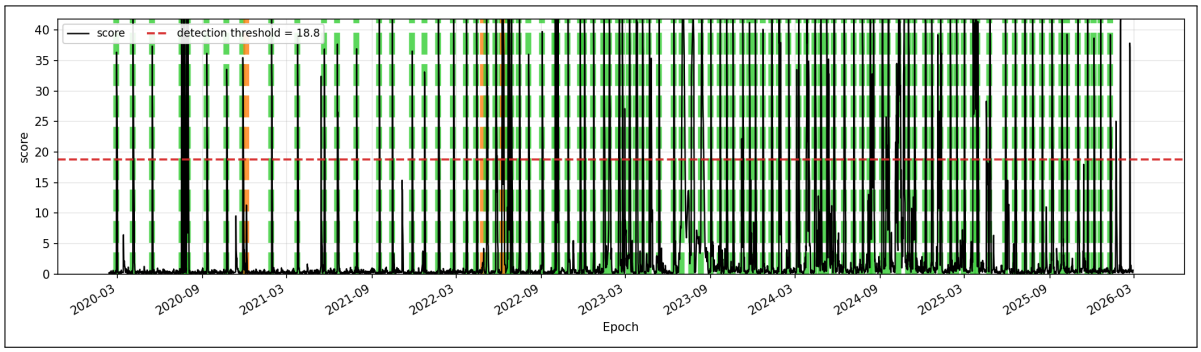

Figure 5.14: CRYO2 score evolution for the test set for the final approach

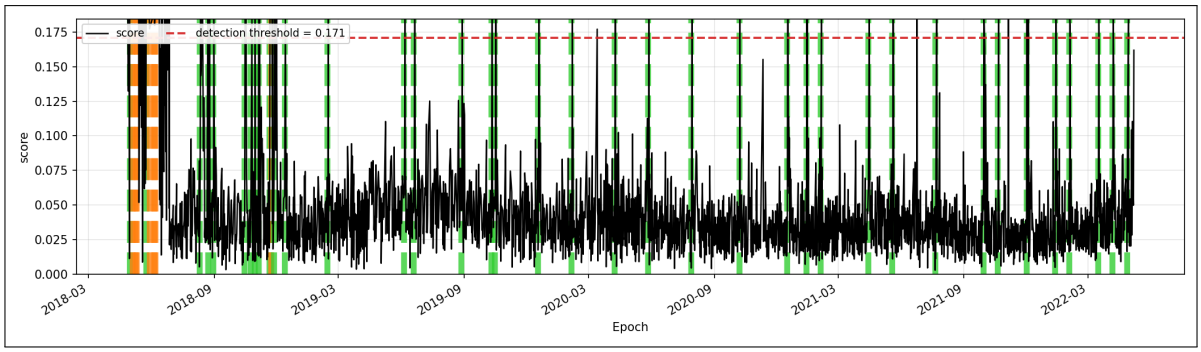

Figure 5.15: SEN3B score evolution for the validation set for the baseline approach

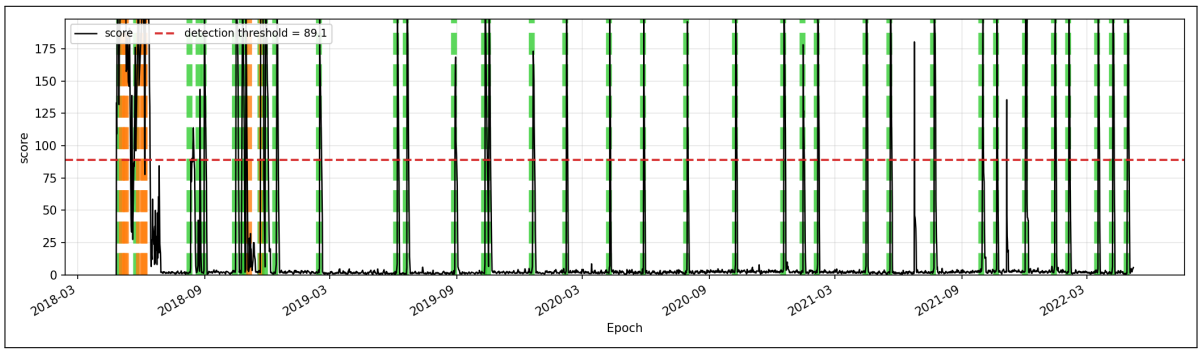

Figure 5.16: SEN3B score evolution for the validation set for the final approach

5.5.2. Intermediate-performing satellites

Satellites such as ENVI1, JASO1 and JASO2 occupy an intermediate regime. Their manoeuvres are detectable, but the background score is noisier and the balance between precision and recall is more sensitive to threshold choice. These satellites benefit substantially from the move from the baseline to the innovation-based scores, which suppress a large number of false positives while preserving most true detections, but still struggle to perform as well as the well-behaved satellites, even for the best-working approach. The improvement is especially visible in ENVI1 and JASO2, as it can be observed in Figures 5.17 and 5.18, showing the improvement between the baseline and the best-working approach for the validation set of ENVI1 satellite, and between 5.19 and 5.20, which show a similar improvement for the validation split of JASO2 satellite.

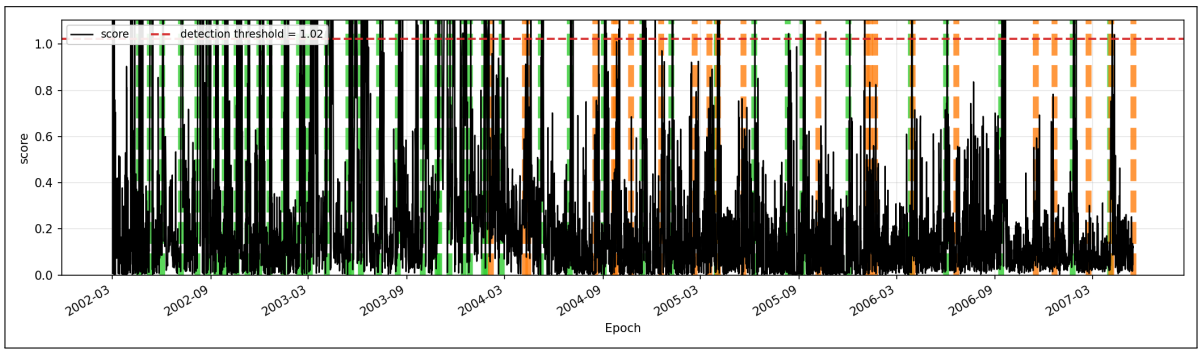

Figure 5.17: ENVI1 score evolution for the validation set for the baseline approach

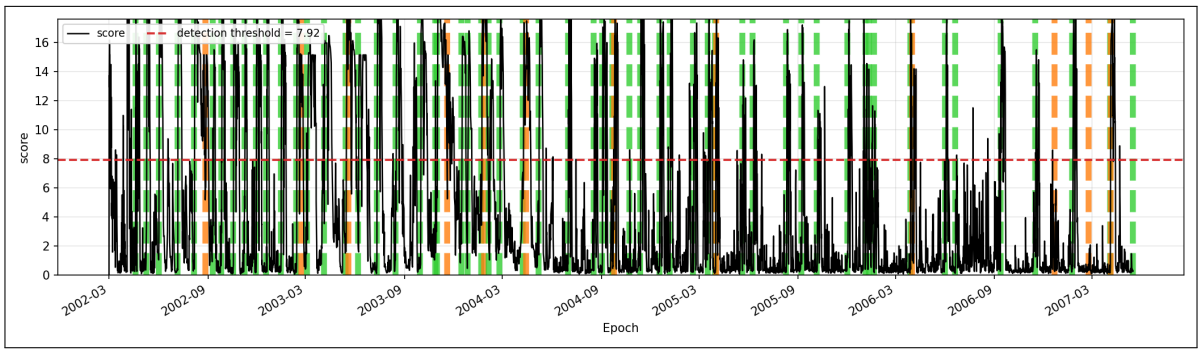

Figure 5.18: ENVI1 score evolution for the validation set for the final approach

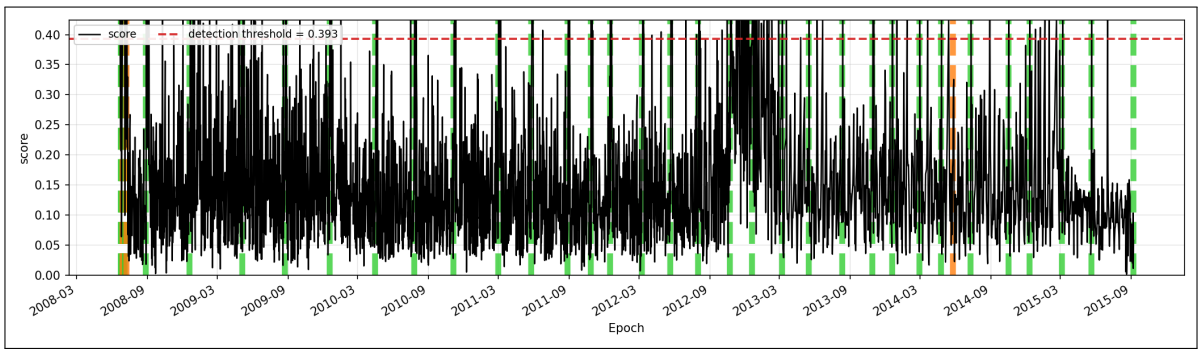

Figure 5.19: JASO2 score evolution for the validation set for the baseline approach

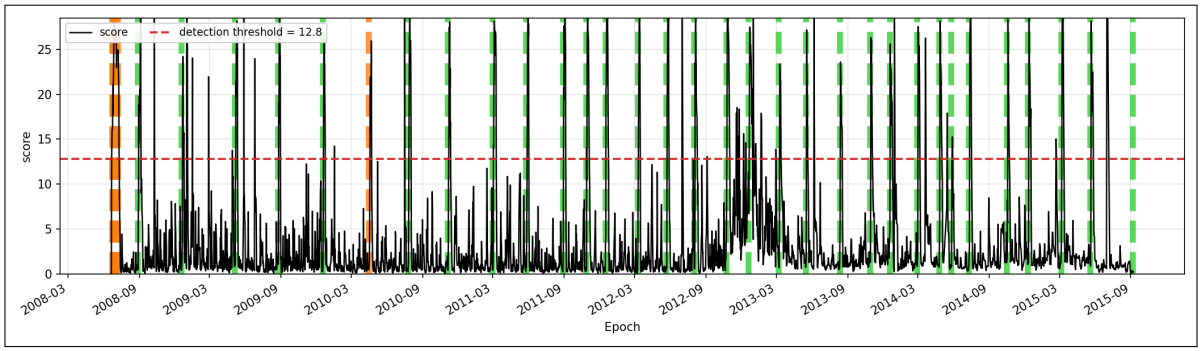

Figure 5.20: JASO2 score evolution for the validation set for the final approach

5.5.3. Low-performing satellites

HY-2A and SWOT1 are the clearest difficult cases. HY-2A remains problematic throughout the full progression and is the only low-performing satellite left after the final SARAL refinement. Its scores are strongly affected by background variability, which produces many false positives even in the later formulations. As it can be observed in Figures 5.21 and 5.22, the evolution from the baseline to the final approach doesn't make a great performance improvement. SWOT1 behaves differently: it is poor in the baseline but becomes much more tractable once the innovation-based scores are introduced, as it was shown before. This contrast illustrates that not all difficult satellites fail for the same reason.

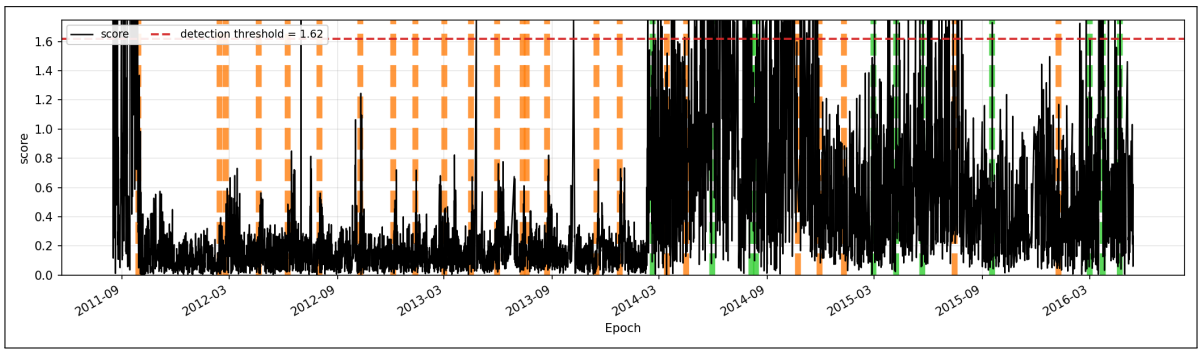

Figure 5.21: HY-2A score evolution for the validation set for the baseline approach

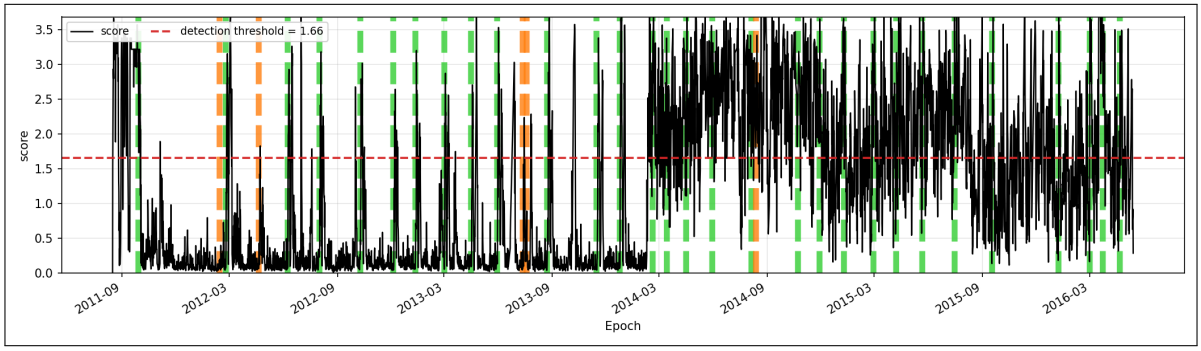

Figure 5.22: HY-2A score evolution for the validation set for the final approach

5.6. Behaviour modelling results

5.6.1. Hierarchical agglomerative clustering based on Ward's linkage

The behavioural characterization framework introduced in Section 4.6 is applied to the detected manoeuvre dataset. The objective is twofold: to classify satellites according to their manoeuvring fre-

quency and to distinguish between station-keeping activities and larger orbital adjustments. The resulting satellite-level and manoeuvre-level classifications are presented and discussed in the following sections.

5.6.1.1. Clustering according to frequency of manoeuvres

The first analysis focuses on manoeuvre frequency. Figure 5.23 presents the dendrogram obtained from the reference manoeuvre dataset, which provides the baseline classification against which the clustering results obtained from the detected manoeuvres will later be compared.

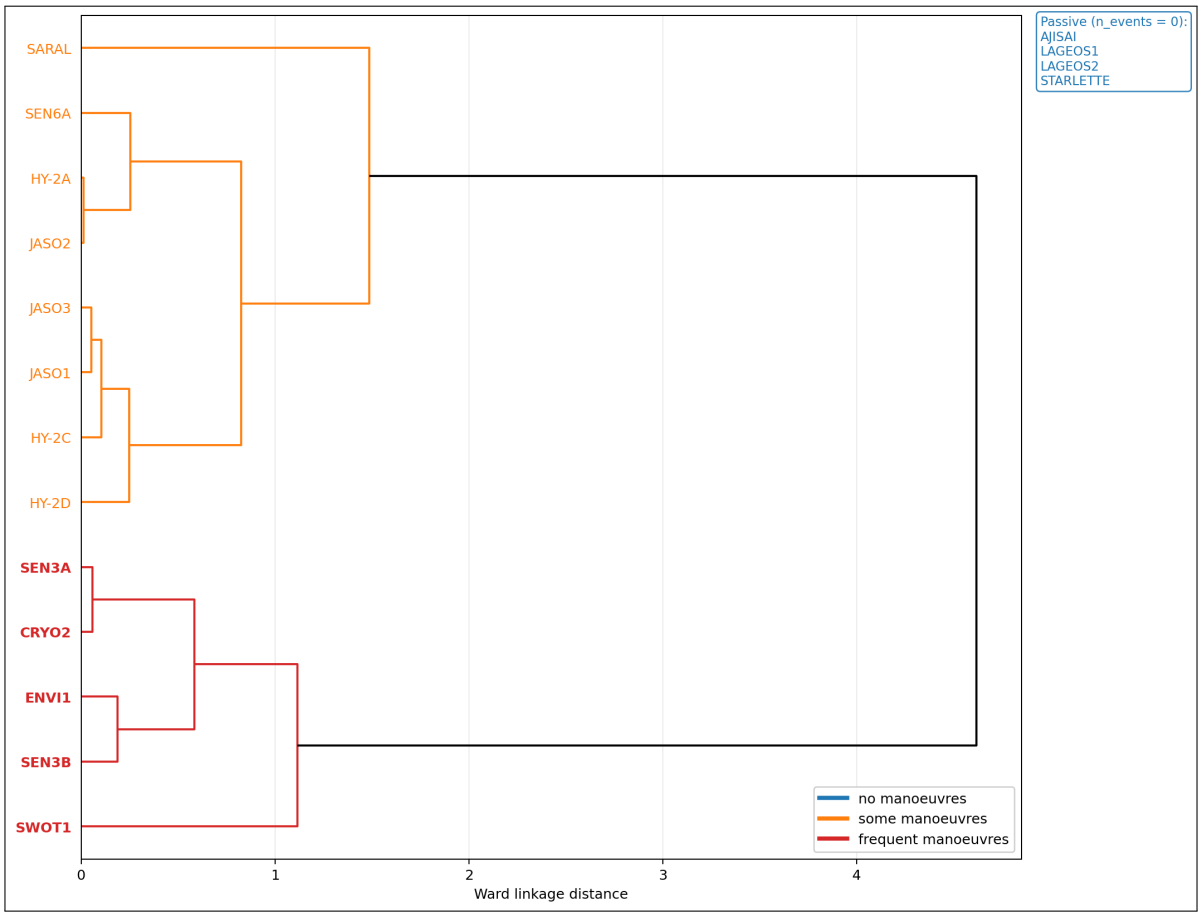

Figure 5.23: Dendogram for the real manoeuvres

Moreover, in the detected manoeuvre case, the following dendogram is obtained (see Figure 5.24):

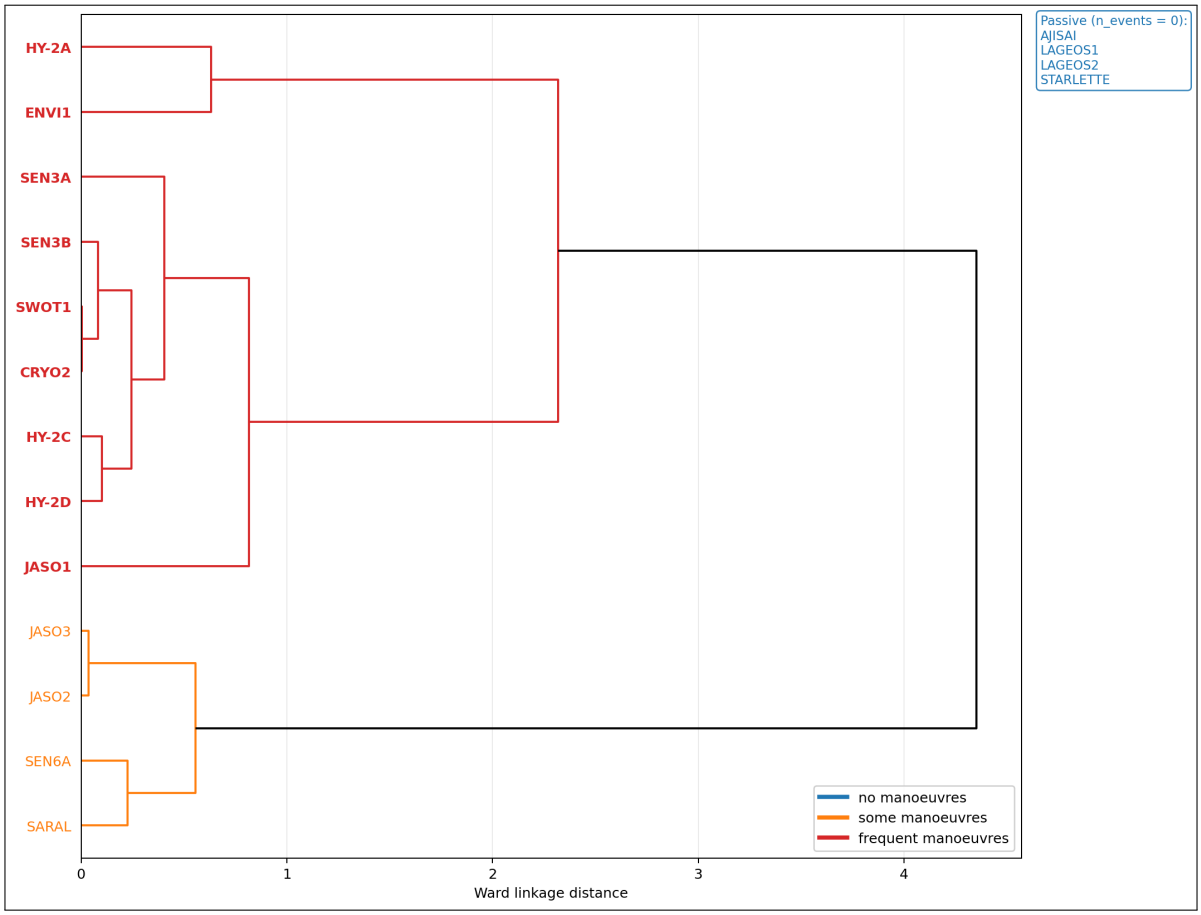

Figure 5.24: Dendogram for the detected manoeuvres

As it can be observed in the previous two figures, there is a mismatch in 4 of the active satellites: HY-2A, HY-2C, HY-2D and JASO1 satellites are labelled as frequently manoeuvring in the detected case and as sparsely manoeuvring in the real case. This is understandable, since the number of real and detected events per year differ very much in these 4 satellites, as it is shown in the following plot (see Figure 5.25):

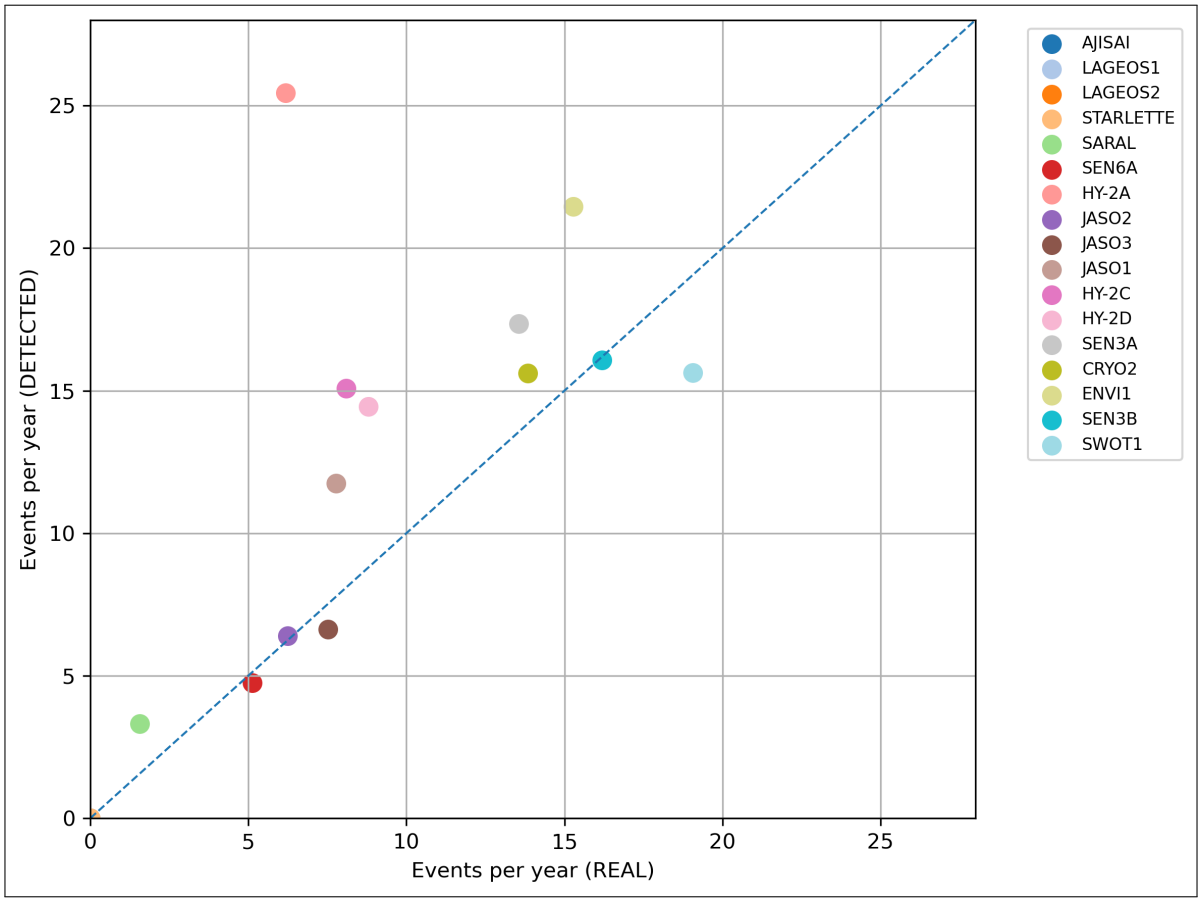

Figure 5.25: Real and detected events per year

The clustering obtained from the detected manoeuvres agrees with the reference classification for 9 of the 13 active satellites. Of the four mismatches, three correspond to satellites whose estimated manoeuvre frequencies lie close to the boundary separating the sparse- and frequent-manoeuvring groups. In these cases, relatively small differences in the estimated number of manoeuvres per year are sufficient to alter the cluster assignment. The remaining mismatch corresponds to HY-2A, whose manoeuvre activity is substantially overestimated due to the same TLE-consistency issues discussed in Section 5.5.3.

Furthermore, the observed misclassifications are largely driven by an overestimation of manoeuvre activity rather than by missed manoeuvres. From the perspective of orbit prediction, this is generally the more acceptable type of error, since falsely identifying additional manoeuvres may lead to the exclusion of some nominal data, whereas failing to detect actual manoeuvres can leave manoeuvre-contaminated observations in the training dataset and potentially degrade prediction performance.

5.6.1.2. Clustering according to nature of manoeuvres

The frequency and directionality indices are subsequently combined to characterize the operational nature of each satellite's manoeuvring activity. Figure 5.26 presents the resulting dendrogram, which reveals the natural separation between satellites dominated by regular station-keeping operations and those primarily performing larger operational manoeuvres.

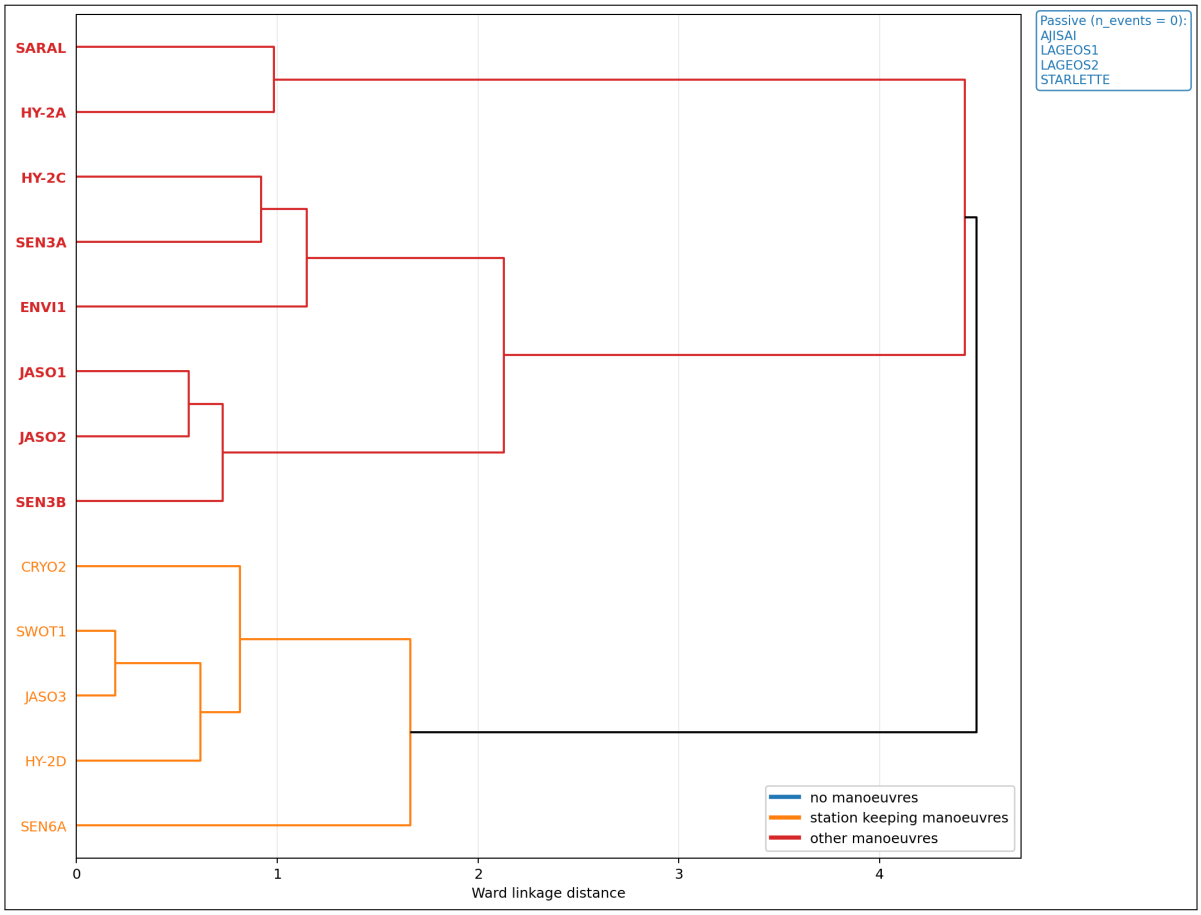

Figure 5.26: Dendogram of nature of detected manoeuvres

Moreover, the following Figure 5.27 shows the frequency and directionality indices for each of the satellites:

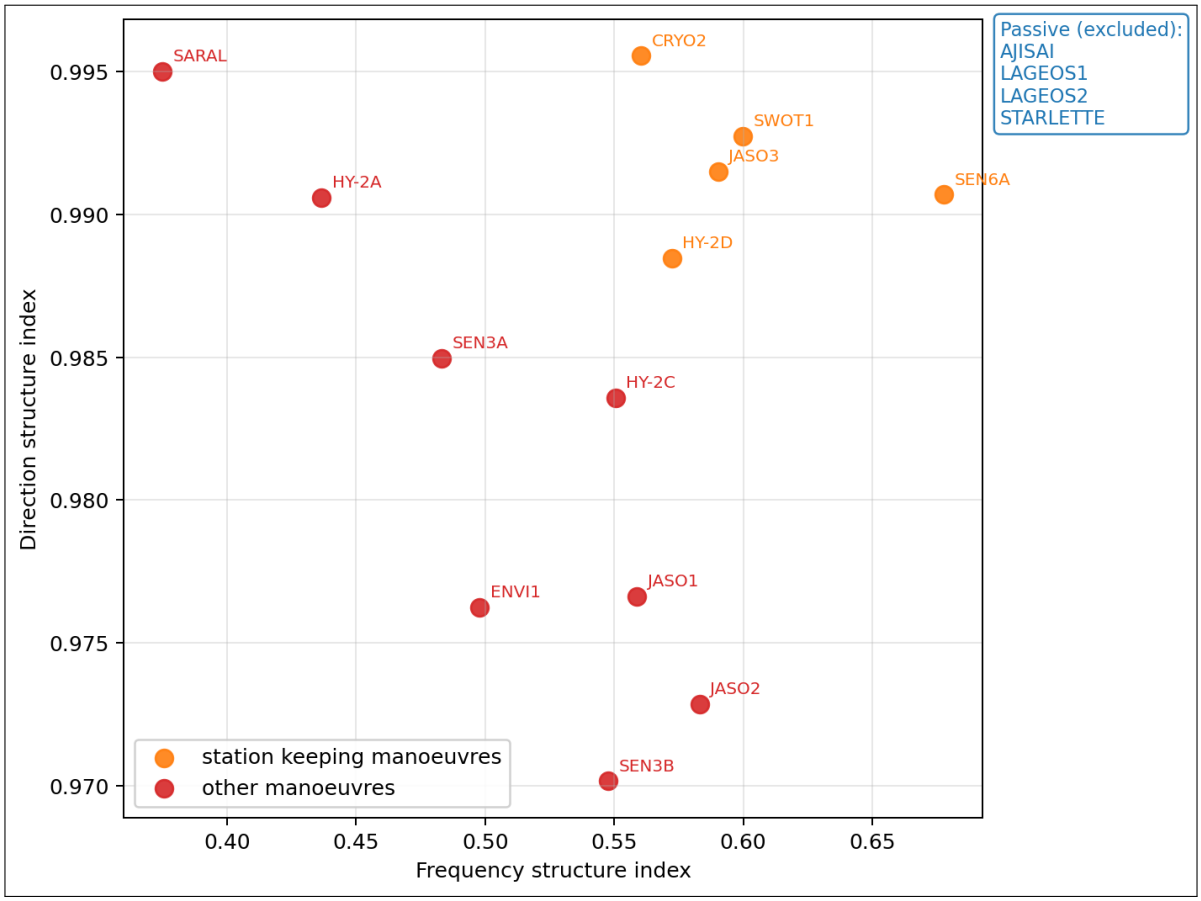

Figure 5.27: Frequency and directionality indices for each satellite

Although the hierarchical clustering identifies a group of satellites exhibiting more regular and frequent manoeuvring behaviour, the separation is noticeably less pronounced than in the frequency-based classification presented previously. Several satellites lie close to the boundary between both groups, indicating that the frequency and directionality indices alone do not always provide sufficient discrimination to unambiguously classify the long-term operational nature of a satellite.

This behaviour is not entirely unexpected. Both indices compress more than a decade of manoeuvring activity into only two scalar quantities, inevitably discarding temporal information about changes in operational behaviour throughout the mission lifetime. As a result, satellites that alternate between different operating regimes may appear similar to satellites exhibiting a single, consistent manoeuvring strategy.

For this reason, the satellite-level classification should be interpreted primarily as a high-level behavioural indicator rather than as a definitive categorization. From an operational perspective, the manoeuvre-level classification developed in the following section is considerably more informative, since it characterizes each detected manoeuvre individually and can therefore support propagation strategies that adapt to the nature of each event rather than assigning a single label to an entire multi-year mission.

5.6.2. Individual classification of manoeuvres

The previous satellite-level classification provides a general characterization of the dominant manoeuvring behaviour over the complete observation period. However, assigning a single operational label to an entire satellite lifetime can be limiting, as spacecraft may undergo different operational phases and execute manoeuvres with different purposes throughout their mission. Therefore, a second analysis is performed at the individual manoeuvre level, where each detected event is characterized according to its own temporal spacing and directional behaviour.

The following examples (Figures 5.28–5.33) illustrate groups of consecutive manoeuvres that exhibit consistent temporal spacing and similar RTN-directional signatures, indicating that they likely correspond to recurring station-keeping strategies. These cases are not intended as a replacement for the satellite-level classification, but rather as evidence that individual manoeuvre sequences can reveal operational patterns that may be hidden when analysing the complete mission history with a single global descriptor.

The selected examples are chosen because they provide clear visual evidence of this behaviour across different satellites. In all cases, the identified manoeuvres appear as distinct score peaks emerging from relatively low and stable noise floors, with comparable magnitudes and consistent directional characteristics within each group. The fact that similar patterns are observed for several independent satellites suggests that the proposed event-level classification captures physically meaningful behaviour rather than satellite-specific artefacts.

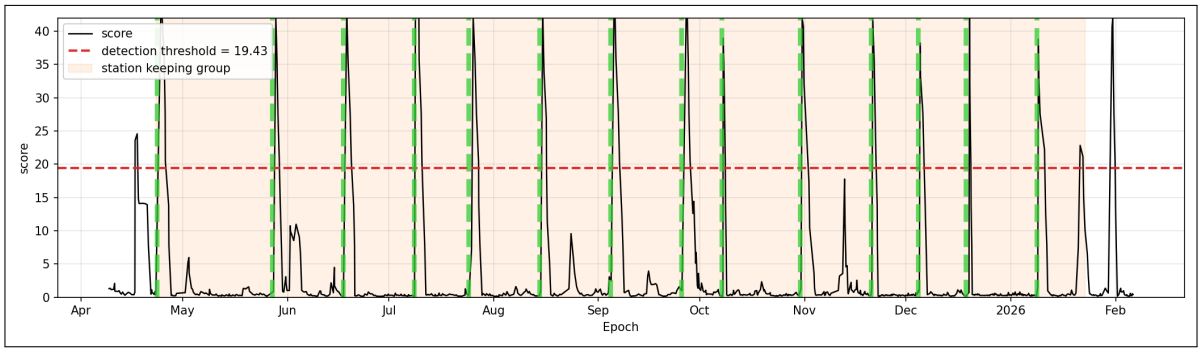

Figure 5.28: Station-keeping group of manoeuvres for CRYO2 satellite, with a median frequency of 21 days in the T direction

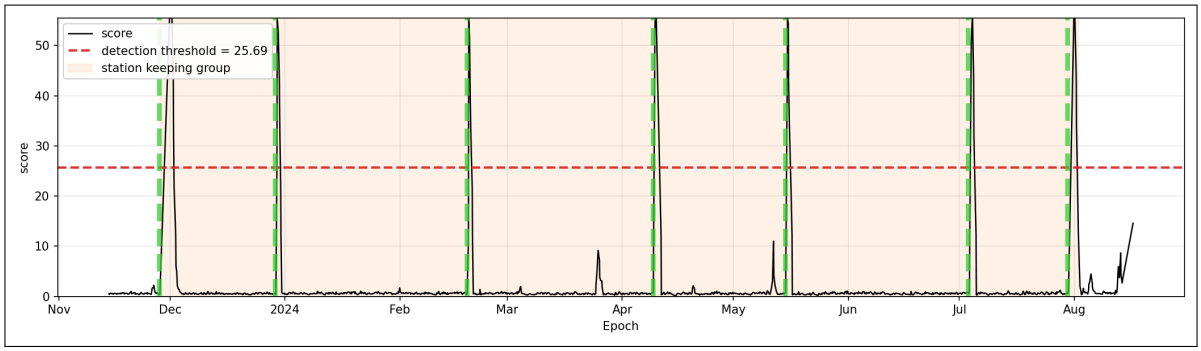

Figure 5.29: Station-keeping group of manoeuvres for HY-2C satellite, with a median frequency of 43 days in the R+T direction

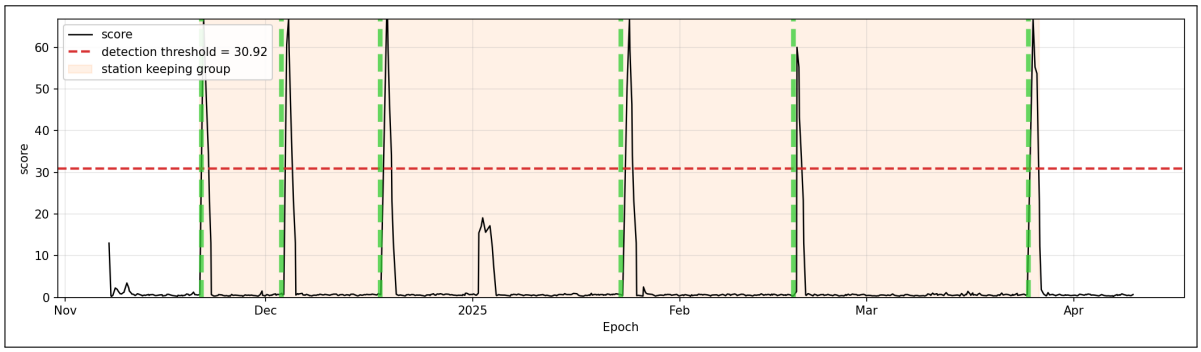

Figure 5.30: Station-keeping group of manoeuvres for HY-2D satellite, with a median frequency of 26 days in the T direction

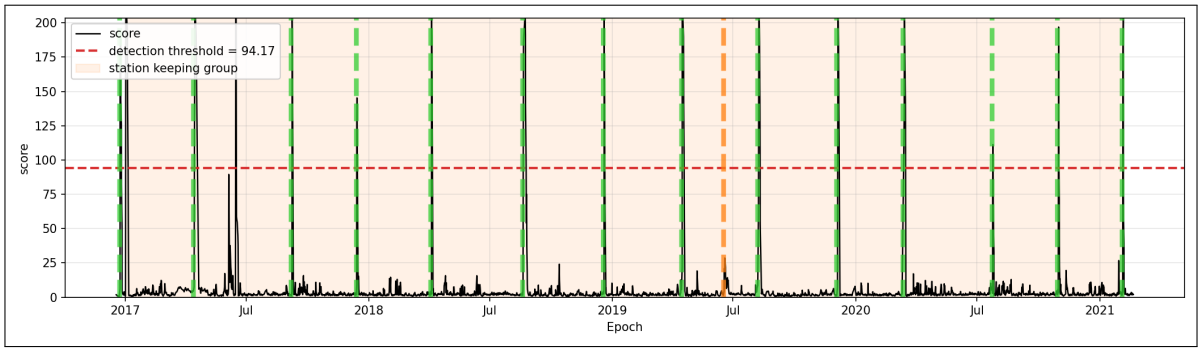

Figure 5.31: Station-keeping group of manoeuvres for JASO3 satellite, with a median frequency of 108 days in the T direction

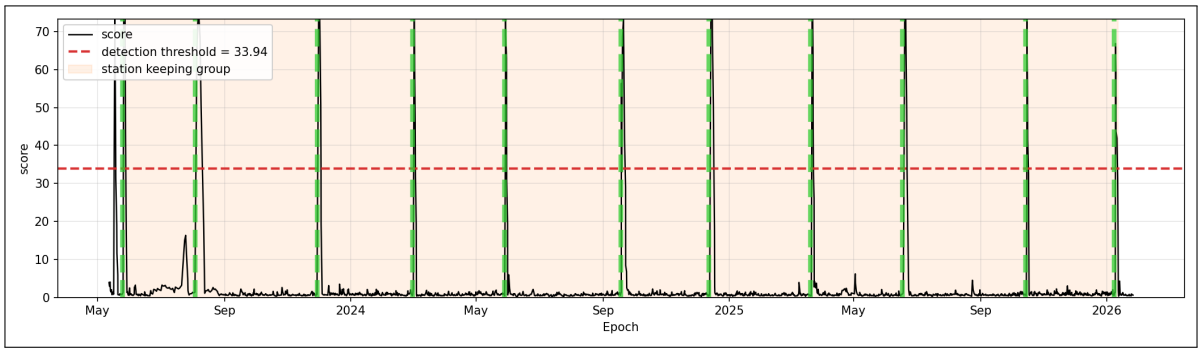

Figure 5.32: Station-keeping group of manoeuvres for SEN6A satellite, with a median frequency of 91 days in the T direction

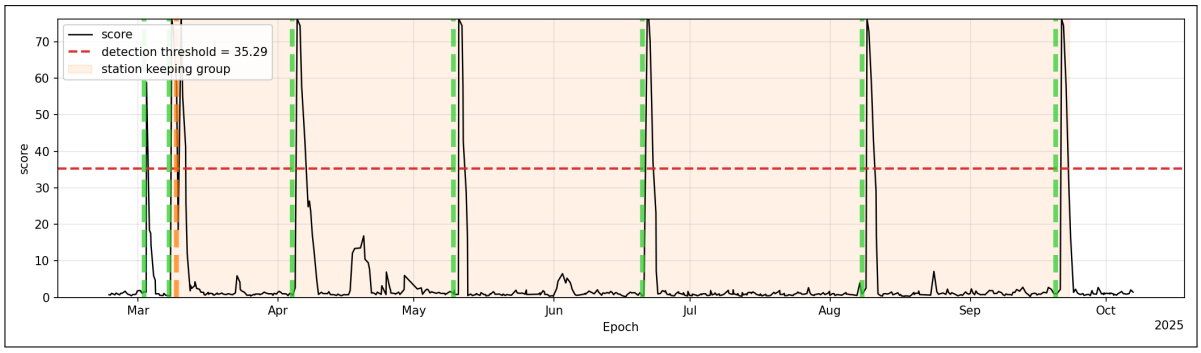

Figure 5.33: Station-keeping group of manoeuvres for SWOT1 satellite, with a median frequency of 41 days in the R+T direction

It is important to note that some of the satellites shown in these examples were previously classified at the satellite level as primarily performing non-station-keeping manoeuvres. This does not represent a contradiction, but rather highlights the aforementioned limitation of assigning a single behavioural label to an entire mission lifetime. A satellite may perform different types of manoeuvres during different operational phases, and a global frequency-directionality descriptor cannot fully capture this temporal variability. The individual event-level classification therefore provides a more flexible and physically meaningful representation, which is potentially more useful for future orbit-prediction frameworks where the propagation strategy could be adapted according to the nature of each manoeuvre rather than according to a fixed satellite-level category.

Nevertheless, despite the promising results obtained, the proposed analysis cannot unambiguously determine the true operational purpose of an individual manoeuvre using TLE data alone. The classification presented here should therefore be interpreted as a behavioural characterization based on

orbital consistency, temporal recurrence, and directional patterns, rather than as a definitive identification of manoeuvre intent. The only way to determine whether a manoeuvre was actually performed for station-keeping or for another operational purpose would be to have access to the mission operator's intent and planning information.

5.7. Evaluation of unknown satellites

Finally, the trained framework is applied to previously unseen satellites in order to assess its behaviour in a realistic operational scenario. Although no reference manoeuvre catalogue is available for these objects, the resulting classifications can be compared against the known operational characteristics of the satellites. In addition, the evolution of the semi-major axis and inclination is analysed to provide an independent indication of whether detected events correspond to physically meaningful orbital changes. The following examples illustrate three representative cases: a passive satellite, a satellite exhibiting irregular operational manoeuvring activity, and a satellite dominated by recurring station-keeping behaviour.

STELLA is a passive geodetic satellite with no expected manoeuvring activity. The resulting anomaly-score profile, shown in Figure 5.34, agrees with this expectation, as the score remains below the detection threshold throughout the complete analysed period and no manoeuvre events are detected. This indicates that the framework is capable of not generating spurious detections for objects without active orbital control.

This behaviour is also consistent with the semi-major axis and inclination evolution shown in Figure 5.35. Both orbital parameters exhibit smooth secular variations, with no abrupt discontinuities associated with impulsive orbital corrections. Therefore, the absence of detected manoeuvres is supported both by the score evolution and by the direct analysis of the orbital elements.

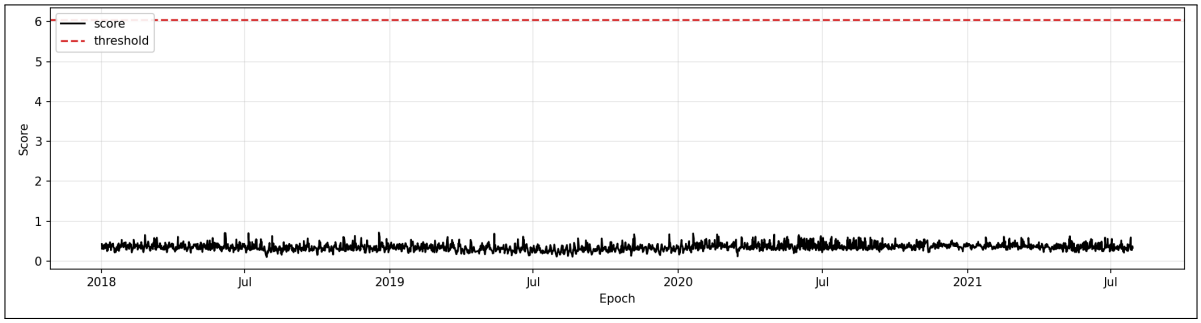

Figure 5.34: Score plot for STELLA satellite from 01/01/2018 to 31/12/2022

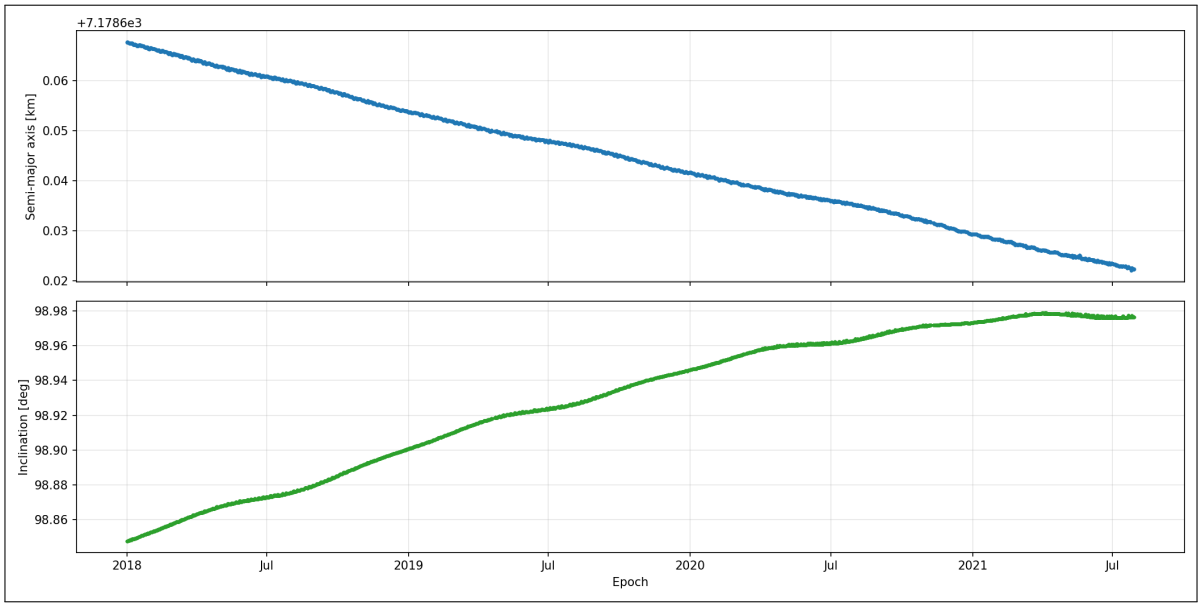

Figure 5.35: Semi-major axis and inclination evolution for STELLA satellite from 01/01/2018 to 31/12/2022

SENTINEL-2B is an Earth-observation satellite operating in a Sun-synchronous LEO, for which orbit-maintenance manoeuvres are expected during nominal operations. The anomaly-score evolution shown in Figure 5.36 presents clear score excursions above the nominal noise level, indicating epochs with behaviour inconsistent with the surrounding TLE sequence. However, unlike satellites dominated by regular station-keeping, these events do not exhibit a constant temporal spacing. Therefore, the detected pattern is not classified as a recurring station-keeping sequence according to the criteria introduced in the previous chapter.

The comparison with the orbital-element evolution in Figure 5.37 provides additional insight. While most of the detected score increases coincide with visible variations in semi-major axis or inclination, others do not produce clearly identifiable discontinuities in these parameters. This illustrates that TLE-based residual analysis can detect changes in orbital consistency that are not always directly observable through simple inspection of individual orbital elements.

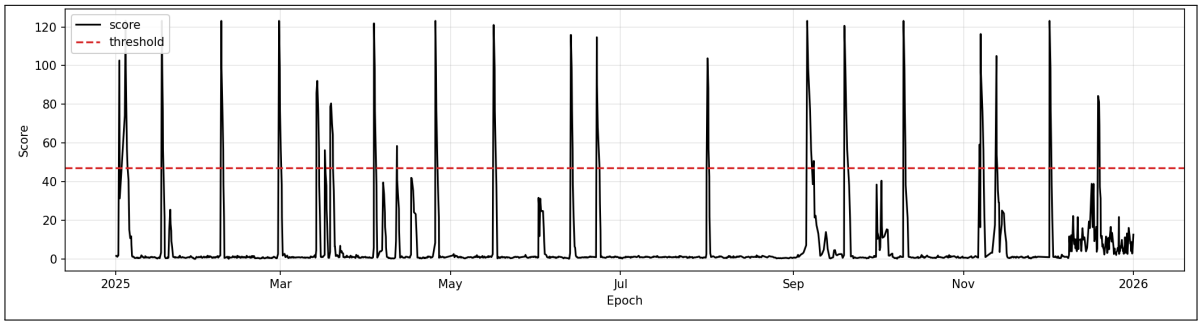

Figure 5.36: Score plot for SENTINEL-2B satellite from 01/01/2025 to 31/12/2025

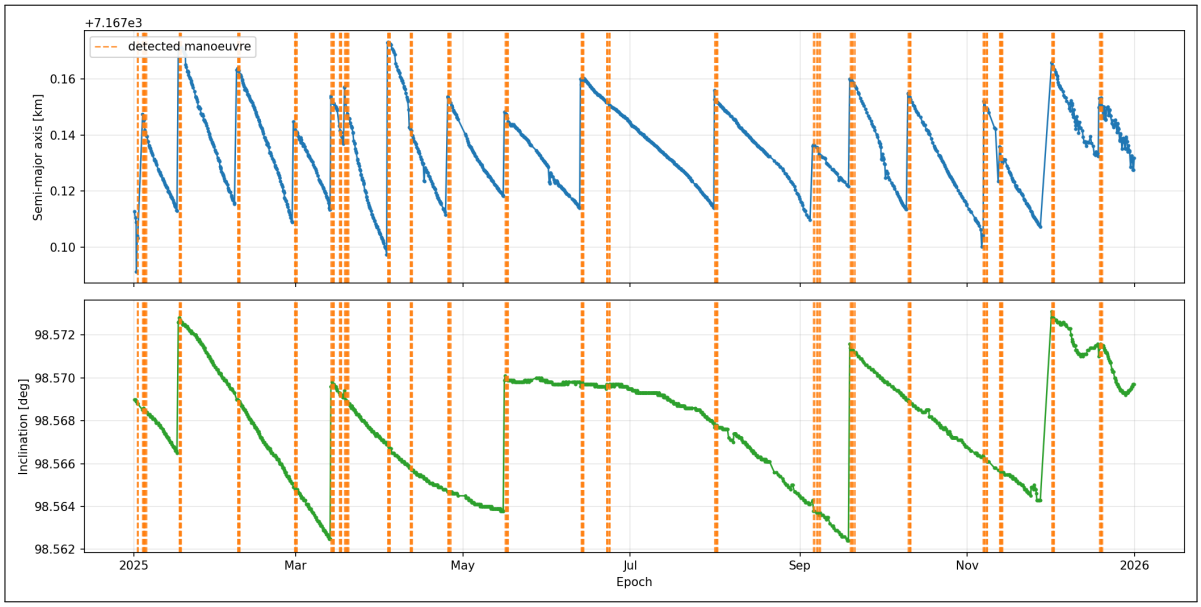

Figure 5.37: Semi-major axis and inclination evolution for SENTINEL-2B satellite from 01/01/2025 to 31/12/2025

HY-2B provides an example of a satellite exhibiting behaviour consistent with regular station-keeping operations. The spacecraft is an operational ocean-observation satellite in a Sun-synchronous LEO, requiring orbit maintenance to preserve its mission orbit. The score evolution shown in Figure 5.38 reveals a clear repetitive pattern: manoeuvre events appear as isolated score peaks emerging from a relatively stable noise background. According to the event-level classification criteria introduced previously, this regularity in temporal spacing and directional behaviour is consistent with a station-keeping manoeuvre sequence.

The physical interpretation is further supported by the evolution of the orbital elements shown in Figure 5.39. Most of the identified events coincide with abrupt changes in semi-major axis, indicating that the detected score excursions correspond to modifications of the orbital state rather than random TLE inconsistencies.

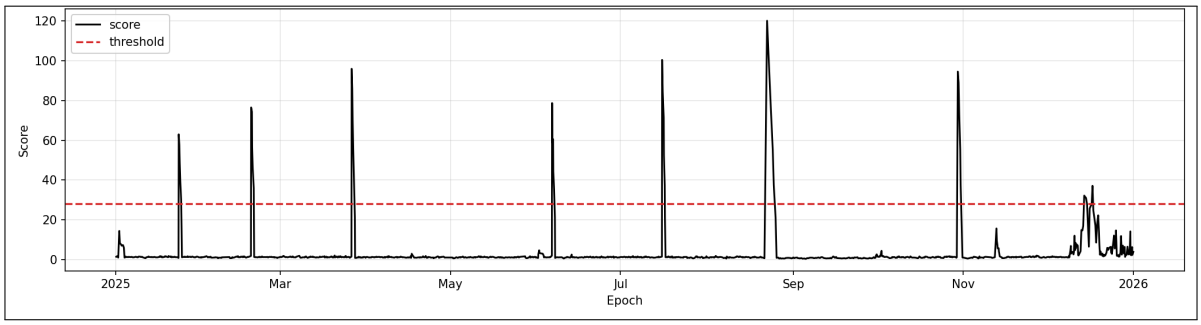

Figure 5.38: Score plot for HY-2B satellite from 01/01/2025 to 31/12/2025

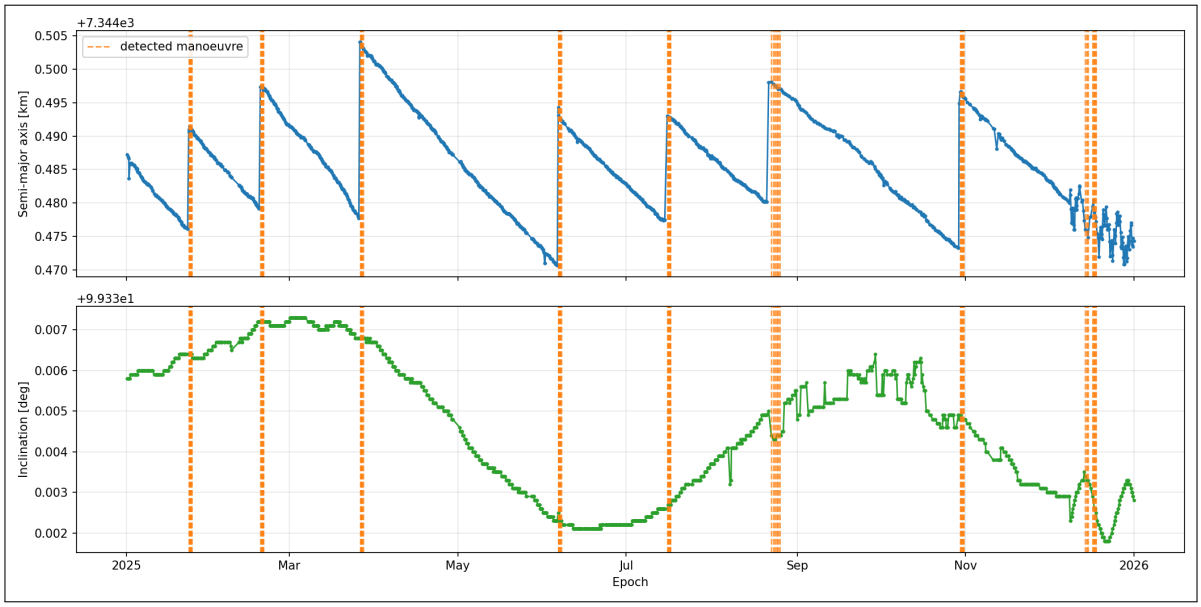

Figure 5.39: Semi-major axis and inclination evolution for HY-2B satellite from 01/01/2025 to 31/12/2025

5.8. Integration into ISOPA

The final stage of the evaluation consists of integrating the manoeuvre-detection framework into the ISOPA orbit-prediction pipeline and assessing its impact on prediction accuracy. For each satellite, ISOPA was executed both in its original configuration and in its manoeuvre-aware version, where epochs affected by detected manoeuvres were excluded from the training dataset. The resulting orbit-prediction errors were then compared against the available CPF reference ephemerides [78, 79]. All accuracy figures reported in this section refer to the root-mean-square (RMS) of the 14-day-horizon position error, which penalizes the large error excursions around manoeuvres more strongly than a simple average and is therefore the more representative metric for this application.

It is worth noting that the ISOPA implementation includes a built-in consistency check that discards epochs for which no CPF data are available. Since both forward and backward propagations are used by the algorithm, a single CPF gap may remove up to approximately 28 days of data from the analysis. Consequently, some flat segments or discontinuities may appear in the following plots and should not be interpreted as algorithmic effects.

Another important aspect concerns the different temporal signatures of the developed manoeuvre detection algorithm and ISOPA's orbit-prediction degradation. The proposed detector is designed to identify the first TLE update affected by a manoeuvre, resulting in a localized response: after the residual calculation, normalization, and whitening procedures, the manoeuvre typically appears as a sharp score increase concentrated around a single epoch. In contrast, the ISOPA error metric is not associated with a single TLE-to-TLE transition, but evaluates the average prediction error obtained from the propagation of 14 initial conditions at each epoch, with a cadence of 1 day. Consequently, before a manoeuvre occurs, several consecutive ISOPA predictions are affected, since propagations initiated before the manoeuvre continue to accumulate the manoeuvre-induced error until the corresponding information is incorporated into the initial TLE. This produces a more extended increase in the ISOPA error metric, which is temporally shifted and spread over several epochs compared with the localized response of the manoeuvre detector.

To account for this difference in temporal behaviour, each detected manoeuvre was expanded by five epochs on either side before removing the corresponding training data. In addition, the centre of the contaminated window was shifted by eight TLE epochs backwards with respect to the detected manoeuvre epoch. Both the expansion size and the temporal shift were determined empirically by analysing the typical extent and alignment of manoeuvre-induced error growth observed in the ISOPA outputs.

The first evaluation was performed on satellites for which the manoeuvre detector had already demonstrated good performance during the validation phase: HY-2C (2025), HY-2D (2025), SENTINEL-6A (2024), and SWOT (2025). Table 5.2 summarizes the accuracy gains obtained by the manoeuvre-aware version of ISOPA for these four satellites, expressed as the reduction in the RMS position error, both in absolute terms and relative to the baseline.

Satellite	Year	Absolute reduction [m]	Relative reduction [%]
SWOT	2025	730	38
HY-2D	2025	515	32
HY-2C	2025	326	34
SENTINEL-6A	2024	74	11

Table 5.2: Reduction in the RMS of the 3D position prediction error over the 14-day horizon achieved by the manoeuvre-aware ISOPA with respect to the baseline, for the four validation satellites.

The integration of manoeuvre awareness into ISOPA produced consistent RMS improvements across all four satellites. The largest gains were obtained for SWOT (730 m, 38%) and HY-2D (515 m, 32%), with HY-2C also benefiting substantially (326 m, 34%). SENTINEL-6A showed a more modest but still clearly positive reduction (74 m, 11%). In every case the manoeuvre-aware version outperformed the baseline, confirming that removing manoeuvre-contaminated epochs from the training set is beneficial regardless of the magnitude of the effect.

To illustrate the typical behaviour behind these figures, SWOT and HY-2D are examined in detail as representative examples of satellites known to perform well. For each of them, three plots are shown: the evolution of the RMS error for both ISOPA configurations (Figures 5.40 and 5.43), the corresponding RMS improvement (Figures 5.41 and 5.44), and the RMS improvement broken down into the three RTN components (Figures 5.42 and 5.45). In the second set of plots, detected manoeuvre epochs are indicated by dashed green lines; since the associated data are removed from the manoeuvre-aware training set, the prediction error is not defined at those epochs.

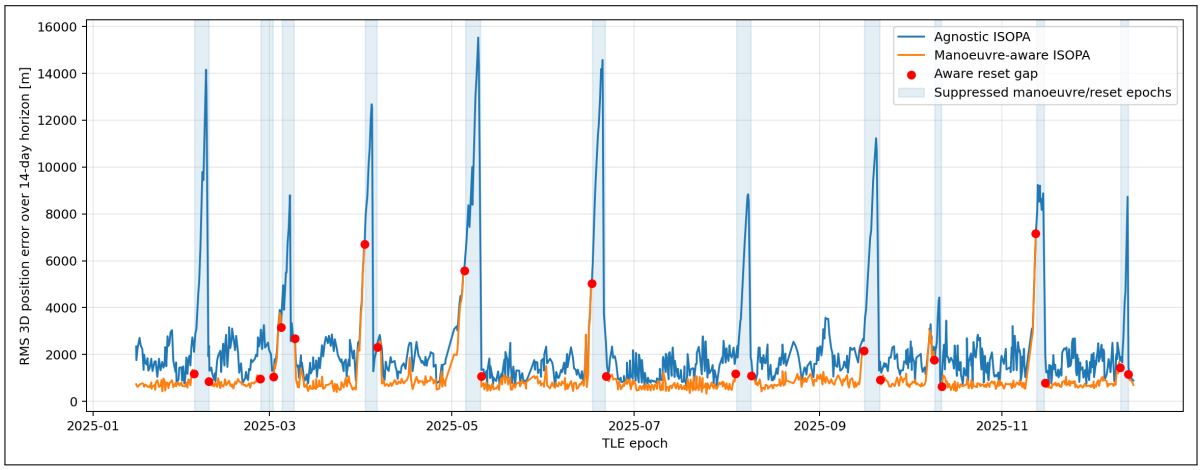

Figure 5.40: Evolution of the RMS of the 3D position prediction error over the 14-day horizon, for the baseline and manoeuvre-aware ISOPA, for the SWOT satellite in 2025.

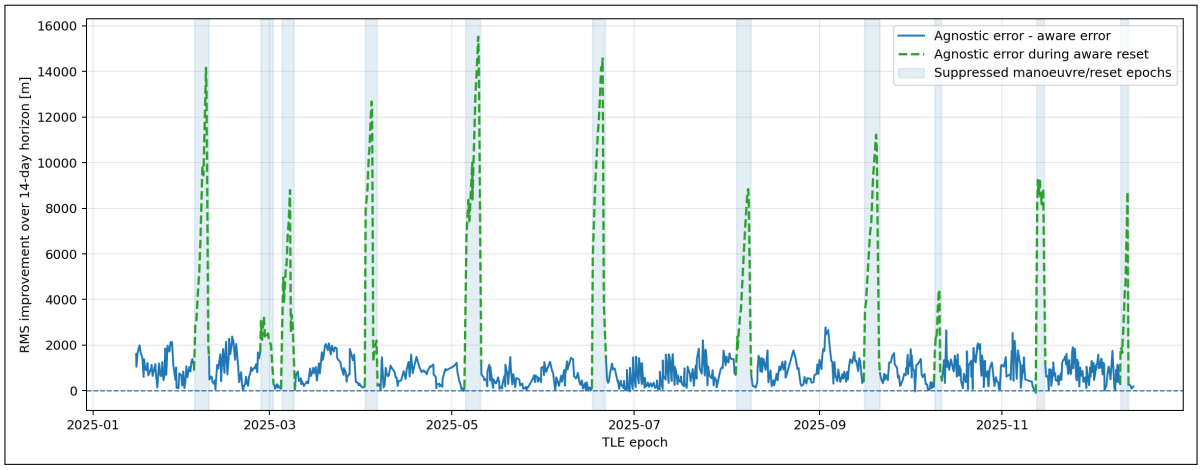

Figure 5.41: Evolution of the reduction in the RMS position error (baseline minus manoeuvre-aware) over the 14-day horizon, for the SWOT satellite in 2025.

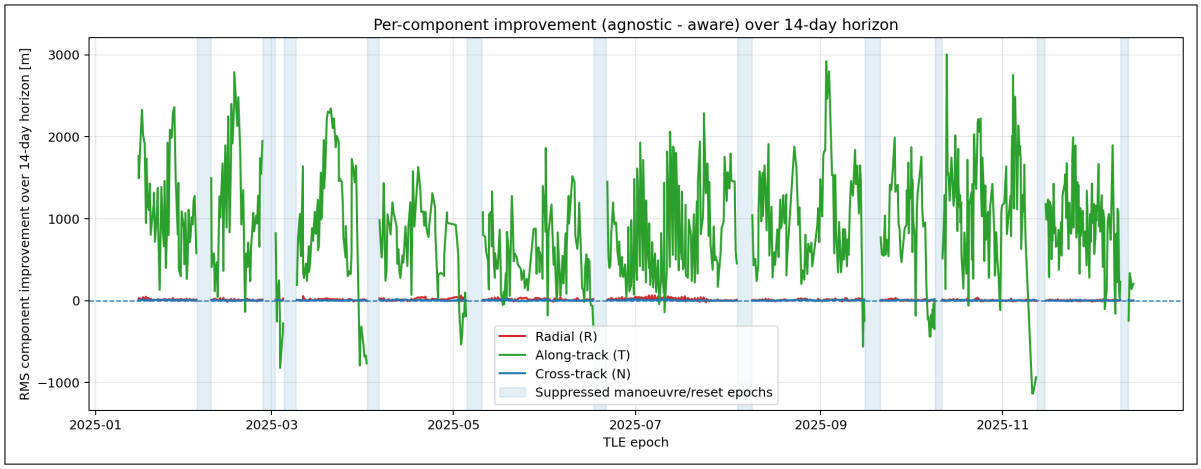

Figure 5.42: Evolution of the reduction in the RMS position error per RTN component (baseline minus manoeuvre-aware) over the 14-day horizon, for the SWOT satellite in 2025.

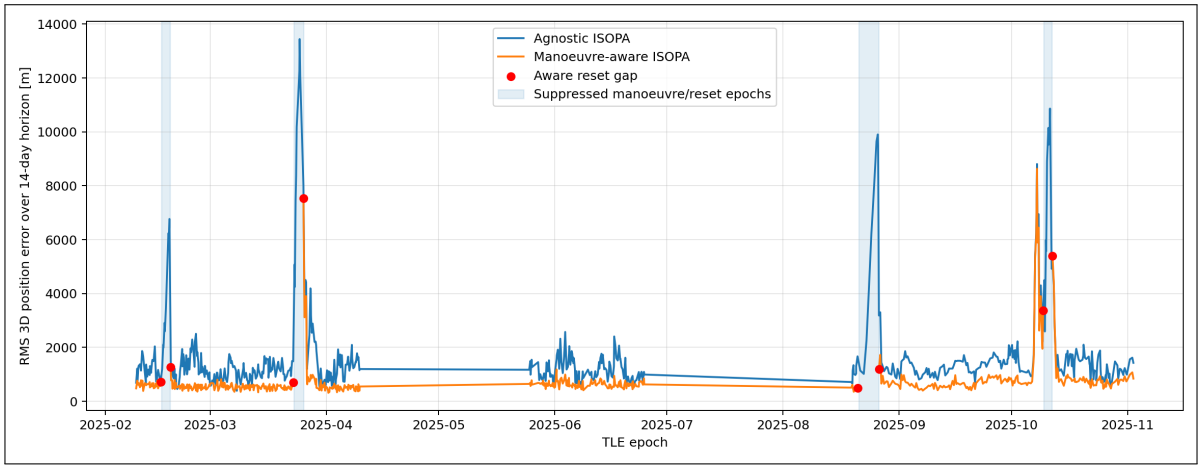

Figure 5.43: Evolution of the RMS of the 3D position prediction error over the 14-day horizon, for the baseline and manoeuvre-aware ISOPA, for the HY-2D satellite in 2025.

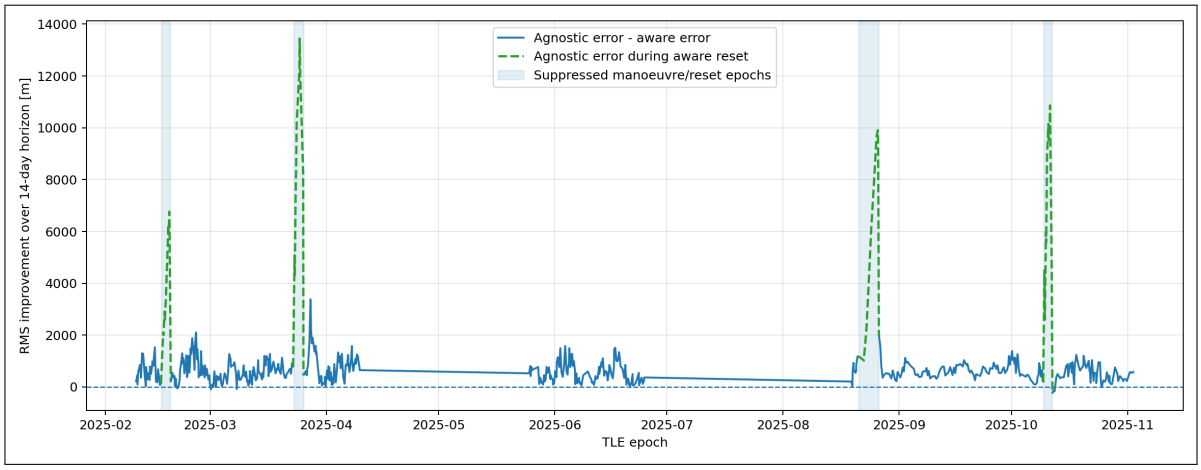

Figure 5.44: Evolution of the reduction in the RMS position error (baseline minus manoeuvre-aware) over the 14-day horizon, for the HY-2D satellite in 2025.

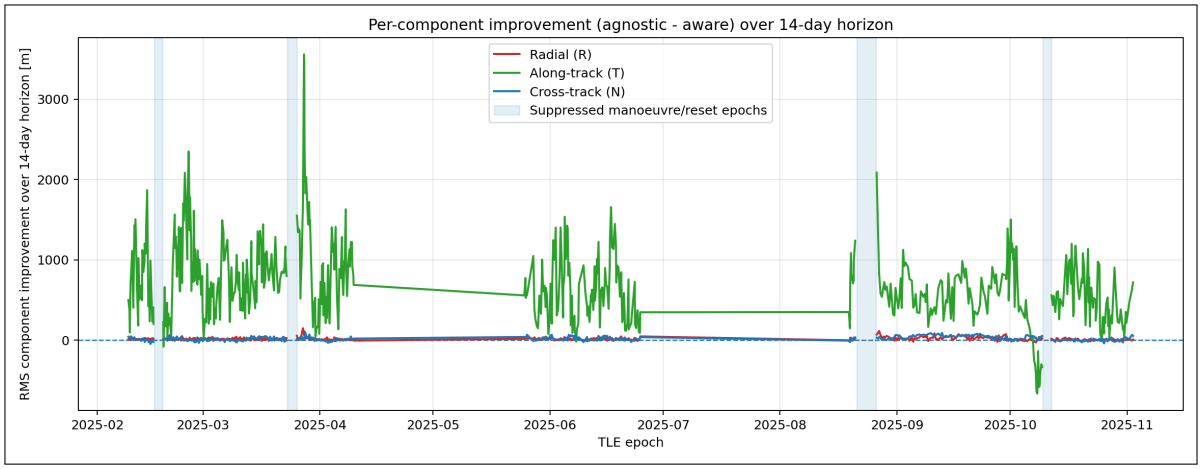

Figure 5.45: Evolution of the reduction in the RMS position error per RTN component (baseline minus manoeuvre-aware) over the 14-day horizon, for the HY-2D satellite in 2025.

The component-wise decomposition (Figures 5.42 and 5.45) reveals a consistent pattern: the largest error reduction occurs in the along-track (transverse) direction. This is both expected and desirable, for two complementary reasons. First, the orbit-maintenance manoeuvres performed by these low-Earth-orbit satellites are predominantly tangential, as they are designed to counteract atmospheric drag and to maintain the semi-major axis, and therefore act mainly along the velocity vector; the manoeuvre-induced contamination of the training data is thus concentrated in the along-track component. Second, the along-track direction is precisely the one along which orbit-prediction errors grow fastest: an error in the semi-major axis, or equivalently in the mean motion, translates into an along-track position error that accumulates secularly with propagation time, whereas radial and cross-track errors remain bounded and oscillatory. Over the 14-day horizon considered here, the along-track component therefore dominates the total position error.

As observed throughout these examples, manoeuvre awareness not only improves prediction accuracy around manoeuvre epochs by removing manoeuvre-contaminated training data, but also reduces the error during nominal orbital evolution. This suggests that training the prediction model exclusively on non-manoeuvring periods leads to a cleaner representation of the underlying perturbation-driven dynamics and therefore to improved predictive performance.

In addition to the satellites used during the manoeuvre-detection validation phase, the proposed framework was also evaluated on previously unseen satellites. As an illustrative example, Figures 5.46–5.48

present the results obtained for HY-2B during 2025: the evolution of the RMS error for both ISOPA configurations, the corresponding RMS improvement, and the per-component breakdown of that improvement.

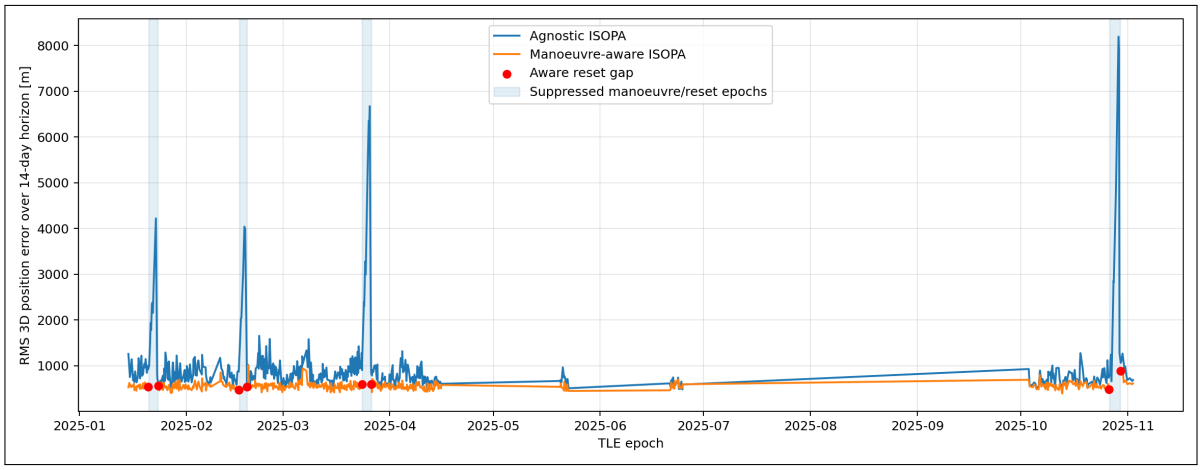

Figure 5.46: Evolution of the RMS of the 3D position prediction error over the 14-day horizon, for the baseline and manoeuvre-aware ISOPA, for the HY-2B satellite in 2025.

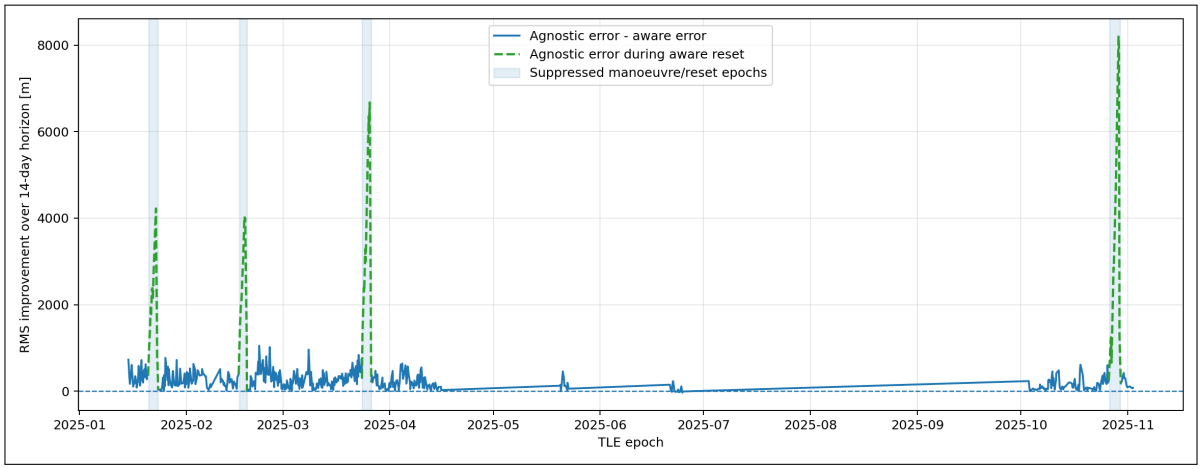

Figure 5.47: Evolution of the reduction in the RMS position error (baseline minus manoeuvre-aware) over the 14-day horizon, for the HY-2B satellite in 2025.

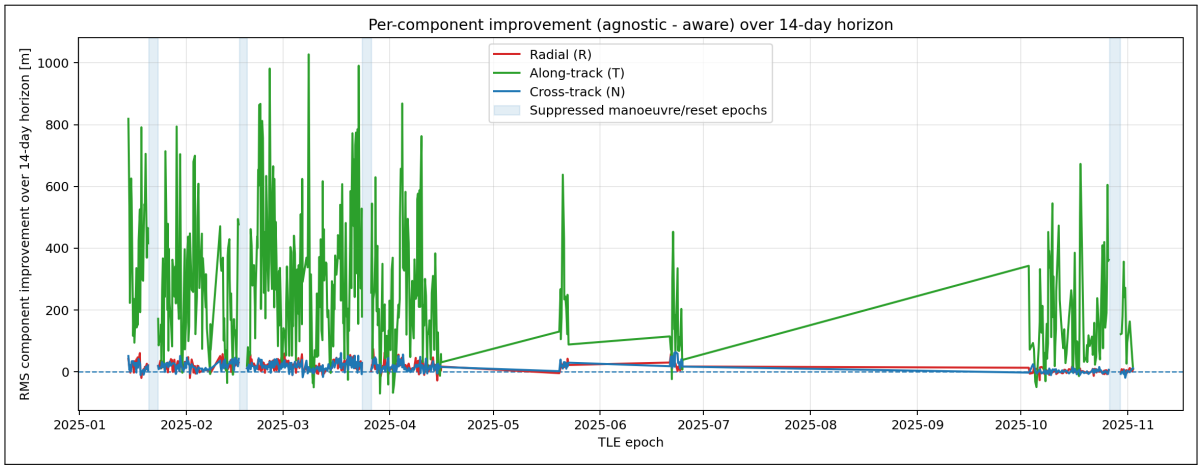

Figure 5.48: Evolution of the reduction in the RMS position error per RTN component (baseline minus manoeuvre-aware) over the 14-day horizon, for the HY-2B satellite in 2025.

HY-2B achieves an RMS error reduction of 252 m, corresponding to an improvement of approximately 31% with respect to the baseline ISOPA implementation. The fact that a comparable gain is obtained on a satellite that did not take part in the detector validation confirms that the framework generalizes beyond the specific objects used to tune it. As for the validation satellites, the component-wise decomposition (Figure 5.48) shows that the along-track direction concentrates most of the reduction, in agreement with the reasoning above.

The benefits of the proposed approach are also visible for satellites whose manoeuvring behaviour is more difficult to characterize. In these cases, the detected manoeuvre intervals are not always as clearly defined as in the previous examples, and the resulting epoch removal may therefore be less accurate. Nevertheless, the overall improvement remains positive for a large fraction of the analysed period, indicating that the manoeuvre detector is still capable of identifying and filtering a significant portion of the manoeuvre-affected observations.

A representative example is provided by SWARM-B. Figures 5.49–5.51 show the evolution of the RMS error for both ISOPA configurations, the corresponding RMS improvement, and its per-component breakdown. Although the detected manoeuvre intervals are less sharply defined than in the previous cases, the manoeuvre-aware solution still outperforms the baseline for most of the analysed time span. Manoeuvre awareness decreases the position error by an RMS value of 283 m, corresponding to an improvement of approximately 3% with respect to the baseline ISOPA implementation. Even in this more challenging case, the component-wise decomposition (Figure 5.51) shows that the along-track direction remains the one where most of the (smaller) reduction is concentrated, consistent with the dynamical argument discussed above.

It is also worth noting that the error peaks observed for SWARM-B are both more numerous and substantially larger than those seen in the previous satellites. This behaviour is likely related to the relatively high manoeuvre frequency of the satellite. Since each detected manoeuvre removes ten days of training data, satellites performing manoeuvres every few weeks can rapidly lose a significant fraction of the available training set. As a result, the amount of nominal data available to ISOPA becomes limited, potentially reducing the effectiveness of the prediction process. Addressing this trade-off between manoeuvre filtering and training-data availability constitutes an important area for future improvements of the framework.

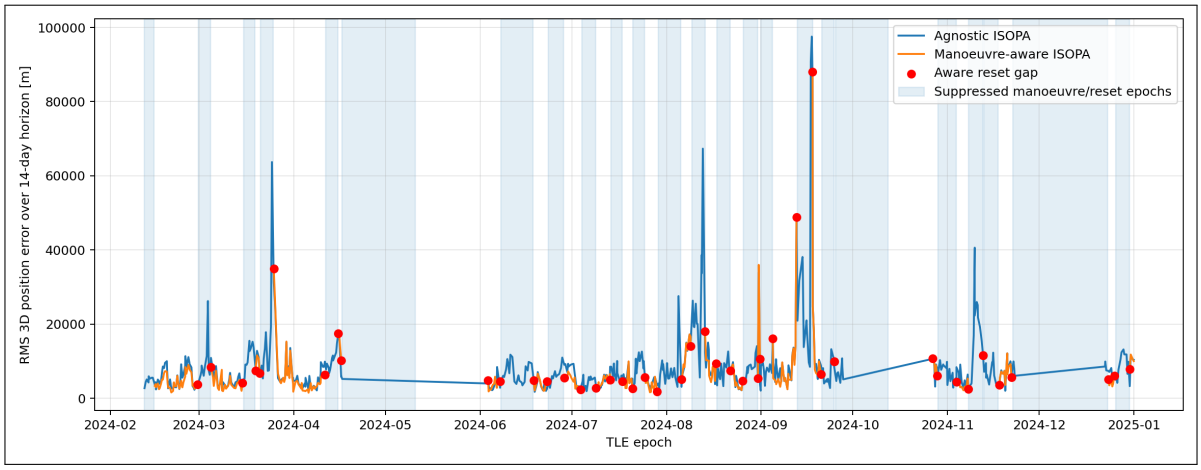

Figure 5.49: Evolution of the RMS of the 3D position prediction error over the 14-day horizon, for the baseline and manoeuvre-aware ISOPA, for the SWARM-B satellite in 2024.

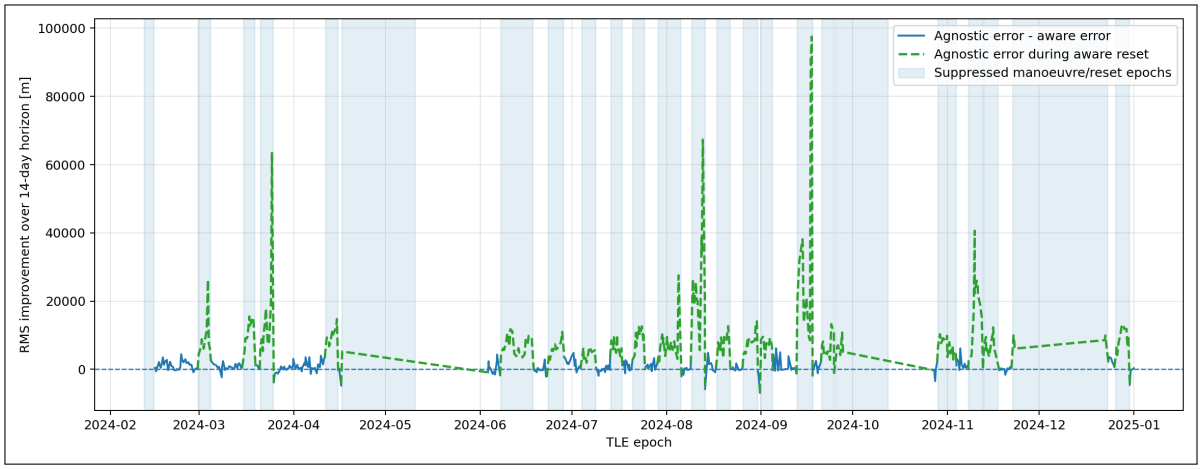

Figure 5.50: Evolution of the reduction in the RMS position error (baseline minus manoeuvre-aware) over the 14-day horizon, for the SWARM-B satellite in 2024.

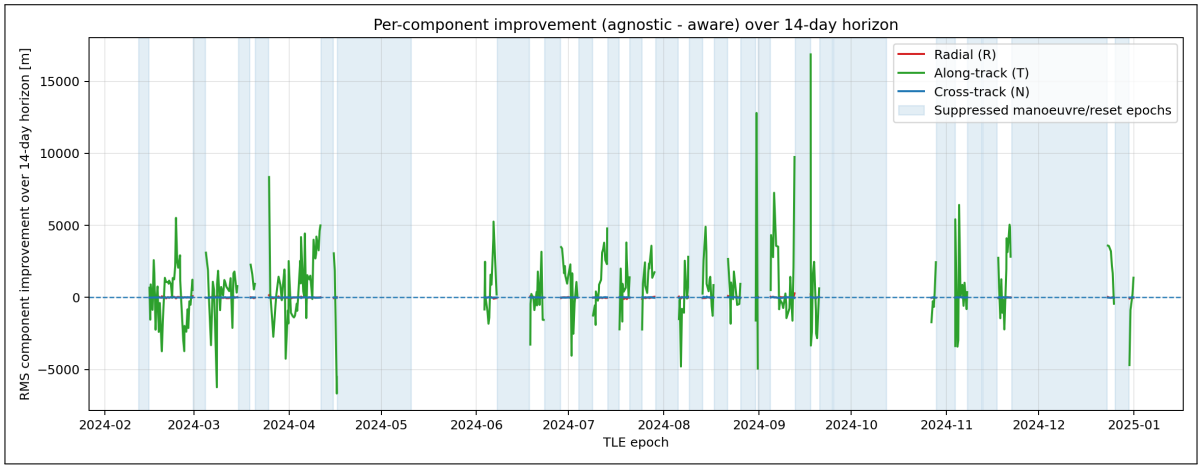

Figure 5.51: Evolution of the reduction in the RMS position error per RTN component (baseline minus manoeuvre-aware) over the 14-day horizon, for the SWARM-B satellite in 2024.

6

Conclusions and answers to Research Questions

6.1. Conclusions

The results presented in Chapter 5 support several main findings.

With respect to the algorithm training

- The results demonstrate that meaningful manoeuvre detection can be achieved using only publicly available TLE data. Despite the inherent noise, estimation errors, and limited physical information contained in TLEs, the proposed methodology is able to identify manoeuvre-related signatures with sufficient reliability to support downstream orbit-prediction applications.
- A significant fraction of the remaining detection errors appears to be attributable to limitations in the underlying TLE data rather than to the detection methodology itself. This is particularly evident in the case of SARAL, where the exclusion of a known problematic interval leads to a substantial increase in performance. After applying this refinement, the framework reaches its best overall results,

$$F_2^{\mathrm{VAL,ALL}} = 0.7359, \qquad F_2^{\mathrm{TEST,ALL}} = 0.7892, \tag{6.1}$$

  while the high-performing/low-performing partition improves to twelve high-performing satellites and only one low-performing satellite, apart from the four correctly identified passive satellites. These results suggest that future gains may depend as much on improvements in the quality of the available orbital data as on further algorithmic refinements.
- The dominant performance gains originate from progressively richer physical representations of TLE-consistency information. Moving from the raw baseline score to innovation-based formulations, and subsequently to normalized multi-backstep representations, accounts for most of the improvement observed on both validation and test datasets. These gains demonstrate that manoeuvre signatures are more effectively captured when the long-term consistency of multiple propagated TLEs is explicitly exploited rather than summarized through a single residual measure. In particular, the tuned whitened innovation score constitutes the best purely physics-based formulation and provides strong generalization performance, since it offers the most complete representation of the information contained in the historical sequence of propagated TLEs.
- The machine-learning branch provides only incremental gains beyond the best physical scores. Under the current TLE noise regime, the physical representations already capture most of the exploitable manoeuvre-related information, while the learned correction acts primarily as a modest refinement. This behaviour is consistent with the fact that the model is trained on nominal epochs that remain affected by TLE noise, orbit-determination errors, and other sources of variability unrelated to manoeuvres. As a result, the learning algorithm does not observe a perfectly clean representation of nominal behaviour, limiting its ability to extract highly discriminative fea-

tures. Nevertheless, the onset-sensitive formulation provides a further consistent improvement and becomes the best general configuration prior to any satellite-specific refinement.

- The most successful score formulations are those that achieve scale consistency between validation and test datasets. Early approaches, such as the median back-innovation score, may produce significant discrepancies in score magnitude across data splits, leading to poor threshold transferability and, in extreme cases, complete detection failure. In contrast, the whitened vector innovation introduces an implicit normalization that stabilizes score distributions over time and makes threshold-based detection considerably more robust.
- Test performance is consistently higher than validation performance across all evaluated configurations. This reinforces the conclusion that the selected parameters do not overfit the validation subset and that the proposed methodology generalizes well to unseen future samples.

With respect to the long-term behaviour modelling, manoeuvre classification, and evaluation of unknown satellites

- The behavioural-classification framework successfully separates satellites according to their manoeuvring activity and operational nature, providing a meaningful characterization of long-term satellite behaviour.
- The algorithm is able to distinguish between frequently manoeuvring, infrequently manoeuvring, and passive satellites in most cases. When misclassifications occur, they are generally associated with an overestimation of manoeuvre frequency rather than with missed manoeuvres. From an orbit-prediction perspective, this represents the most desirable failure mode, since false manoeuvre detections are generally less harmful than undetected manoeuvres.
- The algorithm is able to identify manoeuvre patterns according to both their temporal regularity and directional structure. At the long-term satellite level, however, this classification can be insufficiently informative in some cases, as summarising several years of manoeuvre activity using only two aggregate values may conceal significant variations in the characteristics of individual manoeuvres over time. The classification of individual manoeuvres therefore provides a more informative level of analysis, as it allows the temporal and directional characteristics of specific events to be examined directly. Nevertheless, the true operational purpose of an individual manoeuvre cannot be determined with certainty from orbital data alone. Distinguishing between station-keeping activities, collision-avoidance manoeuvres, orbit transfers, and other mission-specific operations ultimately requires information from the satellite operator. Even so, the consistency observed across all analysed station-keeping cases is encouraging: the individual manoeuvres systematically appear as pronounced score peaks emerging from relatively low and stable noise floors, while also exhibiting clear temporal and directional regularity. Although not conclusive, these recurring patterns provide the strongest indication of manoeuvre nature that can be obtained from publicly available orbital data alone.
- The framework is not only designed for supervised evaluation on satellites with known manoeuvre histories, but can also be applied to previously unseen satellites using only their NORAD identifier and the desired analysis period. Although quantitative performance assessment is not possible in the absence of reference manoeuvre data, the resulting classifications remain consistent with the known operational characteristics of the analysed satellites.

With respect to the integration into ISOPA

- The ultimate value of the proposed framework lies not only in its manoeuvre-detection capability but also in its operational impact. The integration into ISOPA demonstrates that the detected manoeuvre information can be translated into measurable improvements in orbit-prediction accuracy.
- By identifying manoeuvre-contaminated epochs and excluding them from the training process, the manoeuvre-aware version of ISOPA consistently outperforms the original behaviour-agnostic implementation. For several satellites, substantial reductions in the RMS of the 14-day position prediction error were achieved, reaching up to 38% (for SWOT).
- When manoeuvres occur very frequently, the removal of affected epochs may substantially re-

duce the amount of training data available to the orbit-prediction framework. This can limit the effectiveness of the prediction process, particularly for satellites performing manoeuvres every few weeks. Future work could address this issue by refining the way manoeuvre effects are represented and localized in time by ISOPA, thereby reducing the amount of data that must be discarded.

- The availability of CPF reference ephemerides constitutes a limitation for the evaluation of the proposed framework. Missing CPF data create extended gaps in the accuracy assessment, since ISOPA cannot generate valid reference-based prediction-error scores for a significant period around each missing epoch. This limits the temporal coverage over which the accuracy of the resulting predictions can be quantitatively assessed. Nevertheless, this limitation does not affect the operation of the proposed method itself. CPF ephemerides are used exclusively as reference truth for evaluating prediction accuracy and are not provided as input to either the manoeuvre-detection or orbit-prediction stages. The manoeuvre detector operates exclusively on TLE data, while the manoeuvre-aware training strategy relies only on the detected manoeuvre epochs, apart from TLE data. Consequently, the complete pipeline, comprising manoeuvre detection followed by manoeuvre-aware orbit prediction, remains fully operational in the absence of CPF data.
- Building on this distinction, a natural direction for future work is the integration of additional or alternative sources of reference truth into the evaluation of ISOPA. Operator-provided ephemerides or GNSS-based precise orbits could complement or replace the CPF ephemerides, extending the temporal coverage of the accuracy assessment and reducing the evaluation gaps caused by missing CPF data. A broader and more continuous set of truth sources would enable a more robust and representative quantification of the improvements achieved by the manoeuvre-aware approach, and would also make it possible to validate the framework on satellites for which CPF data are not available. Where several truth sources overlap, suitable interpolation or fusion strategies could be applied.

6.2. Answers to Research Questions

All research questions and subquestions proposed in Chapter 3 have been successfully addressed through the development, validation, and operational integration of the proposed manoeuvre-detection framework.

RQ1: Manoeuvre Detection Analysis

How can data-driven analysis be used to identify manoeuvre events and characterise satellite manoeuvring behaviour?

The results demonstrate that manoeuvre events can be identified from publicly available TLE data through the analysis of propagation residuals. By combining physically motivated anomaly scores with machine-learning refinements, the proposed framework is capable of detecting manoeuvre-related signatures and characterising satellite behaviour according to both the frequency and operational nature of the detected manoeuvres.

- **Which TLE consistency representations provide the most informative features for manoeuvre detection?**

  The results indicate that the most informative features are the residuals obtained by propagating seven historical TLEs to a common reference epoch and comparing their predicted states with the current TLE. The resulting position- and velocity-difference vectors are projected onto the RTN frame defined by the reference TLE, ensuring that the residuals are expressed in physically meaningful radial, transverse, and normal directions. These residuals provide information about the consistency of the orbital evolution over time and are significantly more informative for manoeuvre detection than the absolute orbital parameters contained in the TLEs themselves.

  Furthermore, the results show that manoeuvre signatures are better captured when the complete temporal structure of the residual history is retained rather than reduced to a single aggregate statistic. The largest performance improvements were obtained when moving from the baseline representation to multi-backstep residual representations, and subsequently from simple summary statistics to formulations that explicitly exploit the full set of historical residuals. Therefore,

the most effective time-series representations are those that preserve the relative behaviour of multiple historical backsteps simultaneously, allowing the detector to identify temporal consistency breaks rather than relying solely on the magnitude of individual residuals. In the proposed framework, a history of seven backsteps was found to provide sufficient temporal context for effective manoeuvre detection.

- **How do different residual-based score representations influence the performance and reliability of manoeuvre detection?**

  The results show a clear progression in performance as more physically meaningful score formulations are introduced. Raw propagation-error scores provide limited discriminative capability, whereas innovation-based representations substantially improve detection performance by explicitly measuring inconsistencies between historical and current orbital states. Additional gains are obtained through whitening and normalization procedures, which improve score stability across satellites and time periods. The machine-learning refinement further improves performance, although its contribution remains relatively modest compared with the gains achieved through the physical score formulations, suggesting that most of the exploitable manoeuvre information is already captured by the innovation-based features. Finally, onset-sensitive formulations provide the best general performance by emphasizing the abrupt score increases that typically accompany manoeuvre events. A further improvement is obtained through the SARAL-specific refinement; however, this modification primarily affects a single problematic satellite and has only a minor impact on the performance of the remaining satellites. This indicates that, although satellite-specific refinements can improve performance in individual cases, the main gains are obtained from the general physical formulation of the score rather than from ad hoc corrections.

- **How does the presence of outliers affect detection performance, and how can outliers be mitigated?**

  Outliers were found to strongly influence score distributions and threshold estimation, thereby affecting both the stability of the detection scores and the reliability of the resulting thresholds. Their impact can be mitigated through percentile-based normalization and by separating high-performing and low-performing satellites during the optimization process.

  The SARAL case study further demonstrated that not all outlier-related problems can be effectively addressed through purely statistical corrections. In this case, the anomalous data did not consist of isolated points, but rather of extended periods of poor and erratic orbital data. These intervals distorted the score statistics and led to spurious detections. The appropriate mitigation therefore consisted of identifying and removing the affected periods before the detection stage, so that score normalization and threshold estimation were based only on trustworthy TLE segments. Once these intervals were excluded, the detection performance for SARAL, and consequently for the overall dataset, improved markedly.

  This result highlights that the appropriate treatment of outliers depends on their origin and temporal structure. Isolated anomalous samples may be mitigated through robust statistical processing, whereas sustained periods of unreliable input data should be identified and excluded before applying the detection and orbit-prediction stages. Consequently, a data-quality pre-screening step is recommended when extending the framework to additional satellites, as it can improve robustness and reduce the need for satellite-specific corrections.

- **How many subsequent observations (or TLE updates) are required after a manoeuvre to reliably detect its occurrence?**

  Since the proposed scores are computed using only past information, a manoeuvre-induced discontinuity is expected to appear in the first TLE update following the manoeuvre. In principle, therefore, a single post-manoeuvre TLE should be sufficient to generate a detectable increase in the anomaly score.

  However, a safety margin of 3600 minutes is introduced to account for the practical way in which TLEs are generated. The epoch associated with a TLE may result from an orbit-determination process that uses observations acquired both before and after the manoeuvre. Consequently,

TLEs located very close to the manoeuvre epoch may not always reflect the discontinuity immediately, and the score peak can appear one or a few updates later. For this reason, the detection logic allows nearby post-manoeuvre score increases to be associated with the same manoeuvre event, improving robustness against small timing offsets in the TLE sequence.

- **How consistent is the detection performance across different satellites?**

  Detection performance varies across satellites due to differences in manoeuvring behaviour and TLE quality. Nevertheless, the final framework achieves consistent performance for the majority of the analysed satellites, reaching a final partition of twelve high-performing satellites and only one low-performing satellite. Importantly, the framework is not tailored to a specific set of objects: the residual, normalization, and scoring logic is formulated in a satellite-agnostic way and, as demonstrated by its successful application to satellites that did not take part in the tuning, it is expected to generalize to previously unseen satellites. This generalization should nonetheless be interpreted with the size of the available sample in mind. The methodology was developed and assessed on thirteen active and four passive satellites, a limitation imposed by the scarcity of reliable manoeuvre ground-truth data rather than by the method itself. A broader validation, enabled by the incorporation of additional satellites and more comprehensive manoeuvre references, would be required to confirm the extent of this generalization and to further characterize its limits.

RQ2: Manoeuvre-Aware Orbit Prediction

How can manoeuvre awareness be incorporated into a hybrid orbit-prediction framework to improve operational performance?

Manoeuvre awareness can be incorporated by using the manoeuvre detector as a supervisory layer for the orbit-prediction process. Once manoeuvre-contaminated epochs are identified, they are excluded from the ISOPA training dataset, ensuring that the prediction model is trained primarily on nominal orbital evolution. Additional behavioural information regarding manoeuvre frequency and operational nature can also be extracted and provided to the prediction framework.

- **How should detected behaviour classes be used to adapt the prediction strategy within the ISOPA pipeline?**

  When a manoeuvre is identified as part of a structured station-keeping pattern, the associated frequency and directional information can be used to infer how the orbit is expected to evolve in the future. This information could be provided to the orbit-prediction framework to adapt the propagation strategy and anticipate future orbital corrections. Conversely, when a manoeuvre is classified as a larger operational manoeuvre, such as an orbit transfer, collision-avoidance action, or other mission-specific event, the most appropriate strategy is to disregard the pre-manoeuvre orbital history and adapt the prediction process to the new orbital configuration.

  However, this behaviour-specific adaptation was not implemented within ISOPA during the present work. Modifying the internal prediction strategy of the framework was considered beyond the scope of the thesis. Consequently, all detected manoeuvres were treated identically during the integration phase, and the manoeuvre-awareness information was used exclusively to identify and remove manoeuvre-contaminated epochs from the training dataset. The results therefore demonstrate the benefit of manoeuvre awareness itself, while the exploitation of manoeuvre-type information for behaviour-specific prediction remains an avenue for future work.

- **To what extent does manoeuvre-aware adaptation improve prediction accuracy compared with a behaviour-agnostic orbit-prediction model?**

  The integration of manoeuvre awareness into ISOPA consistently improves orbit-prediction performance. Depending on the satellite and time period analysed, reductions exceeding 35% in the RMS 14-day position prediction error were achieved. The results show that the proposed framework is able to identify and remove a large fraction of the epochs associated with pronounced prediction-error peaks, which are likely caused by manoeuvre-induced dynamical discontinuities, while preserving most of the epochs characterized by smooth nominal orbital evolution.

Therefore, the benefits of manoeuvre awareness extend beyond the manoeuvre epochs themselves. By removing training samples affected by orbital discontinuities, the prediction model is trained on a dataset that more accurately represents the nominal dynamics of the satellite. This results in a more accurate representation of the underlying orbital behaviour and leads to improved prediction performance across the entire time span, including epochs well separated from manoeuvre events.

References

[1] European Space Agency. *ESA Space Environment Report 2024*. Tech. rep. Accessed: 2025-11-24. European Space Agency, 2024. URL: https://www.esa.int/Space_Safety/Space_Debris/ESA_Space_Environment_Report_2024.

[2] David A. Vallado. *Fundamentals of Astrodynamics and Applications*. 4th ed. Hawthorne, CA: Microcosm Press, 2013. URL: https://archive.org/details/FundamentalsOfAstrodynamicsAndApplications.

[3] Maurice Uteg et al. “Machine Learning Approach for Accurate and Robust Satellite Tracking in Optical Space-to-Ground Communication using Time-Series Prediction for LEO Satellites”. In: (2025). URL: https://elib.dlr.de/217757/.

[4] European Space Agency. *Space Situational Awareness*. https://www.esa.int/About_Us/Ministerial_Council_2012/Space_Situational_Awareness_SSA. 2025.

[5] David A. Vallado and Paul Crawford. “SGP4 Orbit Determination”. In: *AIAA* (2008). URL: https://celestrak.org/publications/AIAA/2008-6770/.

[6] Lao Zhendi et al. “Experimental Study on Short-arc Initial Orbit Determination of Space Debris Based on Commercial Space-based and Ground-based Electro-optical Monitoring Data”. In: *Chinese Journal of Space Science* (2025). URL: https://ui.adsabs.harvard.edu/abs/2025ChJSS..45.1407L/abstract.

[7] C. Bergmann et al. “Detection of Satellite Manoeuvres Using Non-Linear Kalman Filters on Passive-Optical Measurements”. In: *Acta Astronautica* (2012). URL: https://elib.dlr.de/188523/.

[8] Oliver Montenbruck and Eberhard Gill. *Satellite Orbits: Models, Methods, and Applications*. Springer, 2000. URL: https://ftp.space.dtu.dk/pub/nio/Macau/Montenbruck_2000_SatelliteOrbits.pdf.

[9] Felix R. Hoots and Ronald L. Roehrich. *Models for Propagation of NORAD Element Sets*. Tech. rep. Spacetrack Report No. 3. U.S. Air Force Aerospace Defense Command, 1980. URL: https://apps.dtic.mil/sti/html/tr/ADA093554/index.html.

[10] Srinivas J. Setty et al. “Application of Semi-analytical Satellite Theory orbit propagator to orbit determination for space object catalog maintenance”. In: *Advances in Space Research* (2016). URL: https://www.sciencedirect.com/science/article/pii/S0273117716300072?via%3Dihub.

[11] Martin Lara and Santiago Ferrer. “HEOSAT: a mean elements orbit propagator for highly elliptical orbits”. In: *CEAS Space Journal* 10 (2018), pp. 79–95. URL: https://link.springer.com/article/10.1007/s12567-017-0152-x.

[12] Ahmed M. Atallah et al. “Accuracy and Efficiency Comparison of Six Numerical Integrators for Propagating Perturbed Orbits”. In: *The Journal of the Astronautical Sciences* (2019). URL: https://link.springer.com/article/10.1007/s40295-019-00167-2.

[13] Oliver Montenbruck. “Numerical integration methods for orbital motion”. In: *Celestial Mechanics and Dynamical Astronomy* 53 (1992), pp. 59–69. URL: https://link.springer.com/article/10.1007/BF00049361.

[14] T. Papanikolaou et al. “Assessment of numerical integration methods in the context of low Earth orbits and inter-satellite observation analysis”. In: *Acta Geodaetica et Geophysica* 51 (2016), pp. 479–497. URL: https://link.springer.com/article/10.1007/s40328-016-0159-3.

[15] Giulio Ba‘u, Hodei Urrutxua, and Jes’us Pel’aez. “EDromo: An Accurate Propagator for Elliptical Orbits in the Perturbed Two-Body Problem”. In: *Advances in the Astronautical Sciences* 152 (2014). AAS 14-227, pp. 379–399.

[16] M. Kiani. “Simultaneous approximation of a function and its derivatives by Sobolev polynomials: applications in satellite geodesy and precise orbit determination for LEO CubeSats”. In: *Geodesy and Geodynamics* 11.4 (2020), pp. 303–312. URL: https://www.sciencedirect.com/science/article/pii/S167498472030046X.

[17] Kan Wang et al. “Real-Time LEO Satellite Orbits Based on Batch Least-Squares Orbit Determination with Short-Term Orbit Prediction”. In: *Remote Sensing* 14.22 (2022), p. 5853. URL: https://www.mdpi.com/2072-4292/15/1/133.

[18] Oliver Montenbruck et al. “Reduced dynamic orbit determination using GPS code and carrier measurements”. In: *Aerospace Science and Technology* 9 (2005), pp. 261–271. URL: https://www.sciencedirect.com/science/article/pii/S1270963805000106.

[19] J. Ferguson. “A Comparison of the Extended Kalman Filter and Weighted Least Squares in Early-Orbit Determination”. In: *Journal of Guidance and Control* (1971). URL: https://archive.org/details/DTIC_AD0741457/page/n7/mode/2up.

[20] Pooja Patil et al. “Orbit Determination Using Batch Sequential Filter”. In: *International Journal of Engineering Research & Technology* 7.9 (2018). URL: https://www.ijert.org/research/orbit-determination-using-batch-sequential-filter-IJERTCONV1IS04040.pdf.

[21] E. Park et al. “Satellite orbit determination using a batch filter based on the unscented transformation”. In: *Aerospace Science and Technology* 14.2 (2010), pp. 93–102. URL: https://www.sciencedirect.com/science/article/pii/S1270963810000428.

[22] Yang Guo et al. “Bayesian Adaptive Extended Kalman-Based Orbit Determination for Optical Observation Satellites”. In: *Sensors* (2025). URL: https://www.mdpi.com/1424-8220/25/8/2527.

[23] Max I. Hallgarten La Casta and Davide Amato. “Debiasing of two-line element sets for batch least squares pseudo-orbit determination in MEO and GEO”. In: *Advances in Space Research* (2025). URL: https://www.sciencedirect.com/science/article/pii/S0273117725002856?via%3Dihub.

[24] Francisco M. Caldas et al. “Machine Learning in Orbit Estimation: A Survey”. In: *arXiv preprint arXiv:2203.12345* (2022). URL: https://www.sciencedirect.com/science/article/pii/S0094576524001917.

[25] J. Sang et al. “Analytical representations of precise orbit predictions for Earth orbiting space objects”. In: *Advances in Space Research* 59.8 (2017), pp. 2082–2094. URL: https://www.sciencedirect.com/science/article/pii/S0273117716306251.

[26] Xia Ren et al. “Comparing satellite orbit determination by batch processing and extended Kalman filtering using inter-satellite link measurements of the next-generation BeiDou satellites”. In: *GPS Solutions* 23.4 (2019), pp. 1–14. URL: https://link.springer.com/article/10.1007/s10291-018-0816-9.

[27] C. Levit and W. Marshall. “Improved orbit predictions using two-line elements”. In: *Advances in Space Research* 46.10 (2010), pp. 132–143. URL: https://www.sciencedirect.com/science/article/pii/S0273117710006964.

[28] Xinyuan Mao, Wenbing Wang, and Yang Gao. “Precise orbit determination for low Earth orbit satellites using GNSS: Observations, models, and methods”. In: *Astrodynamics* 8 (2024), pp. 349–374. URL: https://link.springer.com/article/10.1007/s42064-023-0195-z.

[29] Hao Peng, Xiaoli Bai, and Ying Yu. “Fusion of a machine learning approach and classical orbit predictions”. In: *Acta Astronautica* 184 (2021), pp. 151–162. URL: https://www.sciencedirect.com/science/article/pii/S0094576521001636.

[30] Hao Peng and Xiaoli Bai. “Machine learning approach to improve satellite orbit prediction accuracy using publicly available data”. In: *Journal of the Astronautical Sciences* (2019). URL: https://link.springer.com/article/10.1007/s40295-019-00158-3.

[31] Hao Peng and Xiaoli Bai. “Gaussian Processes for improving orbit prediction accuracy”. In: *Acta Astronautica* (2019). URL: https://www.sciencedirect.com/science/article/pii/S0094576518320344.

[32] Hao Peng et al. “Comparative Evaluation of Three Machine Learning Algorithms on Improving Orbit Prediction Accuracy”. In: *Astrodynamics* (2019). URL: https://link.springer.com/article/10.1007/s42064-018-0055-4.

[33] Bin et al. Li. “A machine learning-based approach for improved orbit predictions of LEO space debris with sparse tracking data from a single station”. In: *IEEE Transactions on Aerospace and Electronic Systems* (2020). URL: https://ieeexplore.ieee.org/document/9076032.

[34] X. Bai. “Improving Orbit Prediction Accuracy through Supervised Machine Learning”. In: *arXiv preprint arXiv:1811.05540* (2018). URL: https://www.sciencedirect.com/science/article/pii/S0273117718302035.

[35] X. Xi, X. Li, and C. Nan. “Satellite orbit prediction model based on VMD-RC-LSTM”. In: *Remote Sensing* (2025). URL: https://www.spiedigitallibrary.org/conference-proceedings-of-spie/13561/3058477/Satellite-orbit-prediction-model-based-on-VMD-RC-LSTM/10.1117/12.3058477.full.

[36] Vatsal Sorathiya et al. “Machine Learning-Powered Satellite Trajectory Prediction: A Novel Approach for Enhanced Accuracy”. In: *2024 International Conference on Intelligent Systems*. 2024. URL: https://ieeexplore.ieee.org/abstract/document/10594087.

[37] Jamil A. Haidar-Ahmad and Z. Kassas. “A Hybrid Analytical-Machine Learning Approach for LEO Satellite Orbit Prediction”. In: *25th International Conference on Information Fusion*. 2022. URL: https://ieeexplore.ieee.org/abstract/document/9841298.

[38] C. Wang et al. “Confidence-Based Fusion of AC-LSTM and Kalman Filter for Accurate Space Target Trajectory Prediction”. In: (2024). URL: https://www.mdpi.com/2226-4310/12/4/347.

[39] Ruibo Wei et al. “BDS-3 Satellite Orbit Prediction Method Based on Ensemble Learning and Neural Networks”. In: *Computers, Materials & Continua* (2025). URL: https://www.techscience.com/cmc/v84n1/61747.

[40] Yang Guo et al. “Enhancing Medium-Orbit Satellite Orbit Prediction: Application and Experimental Validation of the BiLSTM-TS Model”. In: *Electronics* (2025). URL: https://www.mdpi.com/2079-9292/14/9/1734.

[41] Hao Peng et al. “Limits of Machine Learning Approach on Improving Orbit Prediction Accuracy Using Support Vector Machine”. In: *Advances in Space Research* (2017). URL: https://www.semanticscholar.org/paper/Limits-of-Machine-Learning-Approach-on-Improving-Peng-Bai/f3b57da461195fea8a1e69a3c7e385b4d0671b5f.

[42] P. Lemos et al. “Rediscovering Orbital Mechanics with Machine Learning”. In: *Machine Learning: Science and Technology* (2022). URL: https://iopscience.iop.org/article/10.1088/2632-2153/acfa63.

[43] Zhiwei Qin et al. “A Method to Determine BeiDou GEO/IGSO Orbital Maneuver Time Periods”. In: *Sensors* (2019). URL: https://www.mdpi.com/1424-8220/19/12/2675.

[44] Fei Ye et al. “A Robust Method to Detect BeiDou Navigation Satellite System Orbit Maneuvering/Anomalies and Its Applications to Precise Orbit Determination”. In: *Sensors* (2017). URL: https://www.mdpi.com/1424-8220/17/5/1129.

[45] Guanwen Huang et al. “A Real-Time Robust Method to Detect BeiDou GEO/IGSO Orbital Maneuvers”. In: *Sensors* (2017). URL: https://www.mdpi.com/1424-8220/17/12/2761.

[46] Stijn Lemmens and Holger Krag. “Two-Line-Elements-Based Maneuver Detection Methods for Satellites in Low Earth Orbit”. In: *Journal of Guidance, Control, and Dynamics* (2014). URL: https://arc.aiaa.org/doi/epdf/10.2514/1.61300.

[47] Xue Bai et al. “Mining Two-Line Element Data to Detect Orbital Maneuver for Satellite”. In: *IEEE Access* (2019). URL: https://ieeexplore.ieee.org/document/8830454.

[48] A. Pastor et al. “Satellite maneuver detection and estimation with optical survey observations”. In: *Journal of the Astronautical Sciences* (2022). URL: https://link.springer.com/article/10.1007/s40295-022-00311-5.

[49] Lorenzo Porcelli et al. “Satellite maneuver detection and estimation with radar survey observations”. In: *Acta Astronautica* (2022).

[50] T. Kelecy and M. Jah. “Detection and orbit determination of a satellite executing low thrust maneuvers”. In: *Acta Astronautica* (2010). URL: https://www.sciencedirect.com/science/article/pii/S0094576509004287.

[51] C. Bergmann et al. “Detection of Satellite Manoeuvres Using Non-Linear Kalman Filters on Passive-Optical Measurements”. In: *Advances in Space Research* (2016). URL: https://elib.dlr.de/188523/.

[52] P. Zhang et al. “Input matrix compensated strong tracking filter for maneuvering spacecraft tracking”. In: *Aerospace Science and Technology* (2017). URL: https://www.sciencedirect.com/science/article/pii/S1270963825010582?ref=pdf_download&fr=RR-2&rr=9a50cb2e4b7cf958.

[53] Jianglong Gui et al. “Satellite maneuver detection using dual-moving window curve fitting”. In: *Acta Astronautica* 238 (2026), pp. 724–738. URL: https://www.sciencedirect.com/science/article/pii/S0094576525005478?ref=pdf_download&fr=RR-2&rr=9a50cc517b26f958.

[54] Zhen Yang et al. “Spacecraft maneuver detection based on nonlinear uncertainty propagation along decussated orbital arc”. In: *Advances in Space Research* 76 (2025), pp. 2472–2487. URL: https://www.sciencedirect.com/science/article/pii/S0273117725006064?ref=cra_js_challenge&fr=RR-1.

[55] J. M. Montilla, Rafael Vázquez, and Pierluigi Di Lizia. “Maneuver Detection with Two Mixture-Based Metrics for Radar Track Data”. In: *Journal of Guidance, Control, and Dynamics* (2025). URL: https://arc.aiaa.org/doi/epdf/10.2514/1.G008333.

[56] Marcus J. Holzinger, Daniel J. Scheeres, and Kyle T. Alfriend. “Object Correlation, Maneuver Detection, and Characterization Using Control-Distance Metrics”. In: *Journal of Guidance, Control, and Dynamics* (2012). URL: https://arc.aiaa.org/doi/pdf/10.2514/1.53245.

[57] Daniel P. Lubey and Daniel J. Scheeres. “Towards Real-Time Maneuver Detection: Automatic State and Dynamics Estimation with the Adaptive Optimal Control Based Estimator”. In: *Advanced Maui Optical and Space Surveillance Technologies Conference (AMOS)*. 2015.

[58] A. Dehadraya et al. “Geosynchronous Satellite Pattern-of-Life Characterization Through Machine Learning-based Mode Change Detection and Classification”. In: *Acta Astronautica* (2023). URL: https://ieeexplore.ieee.org/document/10667695.

[59] Kelvin Siew et al. “AI SSA Challenge Problem: Satellite Pattern-of-Life Characterization Dataset and Suite”. In: *Acta Astronautica* (2024). URL: https://www.researchgate.net/publication/374083350_AI_SSA_Challenge_Problem_Satellite_Pattern-of-Life_Characterization_Dataset_and_Benchmark_Suite.

[60] Riccardo Cipollone, Carolin Frueh, and Pierluigi Di Lizia. “LSTM-based maneuver detection for resident space object catalog maintenance”. In: *Neural Computing and Applications* (2025). URL: https://link.springer.com/article/10.1007/s00521-025-11177-7.

[61] Xingyu Zhou et al. “A LSTM assisted orbit determination algorithm for spacecraft executing continuous maneuver”. In: *Acta Astronautica* (2022). URL: https://www.sciencedirect.com/science/article/pii/S0094576522005069.

[62] Fen Li et al. “A station-keeping maneuver detection method of non-cooperative geosynchronous satellites”. In: *Advances in Space Research* (2023). URL: https://www.sciencedirect.com/science/article/pii/S0273117723008141.

[63] R. Abay et al. “Maneuver detection of space objects using Generative Adversarial Networks”. In: *Acta Astronautica* (2018). URL: https://www.semanticscholar.org/paper/Maneuver-detection-of-space-objects-using-Networks-Abay-Gehly/6758e20bcdb1307ad7c89539b3ab0e9cf15962fd.

[64] Yang Ying et al. “Maneuver Detection of Cataloged Space Object Based on Machine Learning”. In: *Proceedings of the 43rd Chinese Control Conference*. 2024. URL: https://ieeexplore.ieee.org/abstract/document/10662282.

[65] Nicholas Perovich et al. “Satellite maneuver detection using machine learning and neural network methods”. In: *2022 IEEE Aerospace Conference*. 2022. URL: https://ieeexplore.ieee.org/document/9843412.

[66] IDS. “IDS: International Doris Service - Documents for the Data Analysis”. In: (2026). URL: https://ids-doris.org/resources/technical-documents.html.

[67] Kelecy et al. “Satellite Maneuver Detection Using Two-line Element (TLE) Data”. In: (2007). URL: https://www.researchgate.net/publication/242742404_Satellite_Maneuver_Detection_Using_Two-line_Elements_Data.

[68] Arvind Mukundan and Hsiang-Chen Wang. “Simplified Approach to Detect Satellite Maneuvers Using TLE Data and Simplified Perturbation Model Utilizing Orbital Element Variation”. In: (2021). URL: https://www.mdpi.com/2076-3417/11/21/10181.

[69] Daniel Jaunzemis, Jack Mathew, and Marcus Holzinger. “Control Cost and Mahalanobis Distance Binary Hypothesis Metrics for Maneuver Detection”. In: *Acta Astronautica* (2024). URL: https://arc.aiaa.org/doi/10.2514/1.G001616.

[70] D. Wang et al. “A machine learning method for the orbit state classification of large LEO constellation satellites”. In: *Advances in Space Research* (2022). URL: https://www.sciencedirect.com/science/article/pii/S0273117722008961.

[71] Q.M. Zhang R. Wang J. Liu. “Propagation errors analysis of TLE data”. In: (2008). URL: https://www.sciencedirect.com/science/article/pii/S0273117708006121?via%3Dihub.

[72] NORAD. *North American Aerospace Defense Command*. Accessed January 29, 2026. 2025. URL: https://www.norad.mil/.

[73] CS Group. *Orekit: An Accurate and Efficient Core Library for Space Flight Dynamics*. Accessed December 15, 2025. 2024. URL: https://www.orekit.org.

[74] Joe H. Ward. “Hierarchical Grouping to Optimize an Objective Function”. In: *Journal of the American Statistical Association* 58.301 (1963), pp. 236–244. DOI: 10.1080/01621459.1963.10500845.

[75] AVISO+. *SARAL/AltiKa Orbit Description*. https://www.aviso.altimetry.fr/en/missions/current-missions/saral/orbit-1.html. Accessed: 2026-03-24. 2023.

[76] AVISO+. *DUACS Product Changes and Updates – SARAL*. https://www.aviso.altimetry.fr/en/data/product-information/updates-and-reprocessing/ssalto/duacs-product-changes-and-updates/archives.html. Accessed: 2026-03-24. 2015.

[77] Jean Verron et al. “The SARAL/AltiKa Altimetry Satellite Mission”. In: *Remote Sensing* 10.7 (2018), p. 1051. DOI: 10.3390/rs10071051. URL: https://www.mdpi.com/2072-4292/10/7/1051.

[78] NASA CDDIS. *SLR Consolidated Prediction Format (CPF) Orbit Prediction Archive*. https://cddis.nasa.gov/archive/slr/cpf_predicts_v2/. Accessed: 2026-06-04. 2026.

[79] United States Space Force. *Space-Track: Space Surveillance and Tracking Data Service*. 2026. URL: https://www.space-track.org/ (visited on 06/04/2026).

[80] Peter L. Bartlett, Cristina Butucea, and Johannes Schmidt-Hieber. “Mathematical Foundations of Machine Learning”. In: *Oberwolfach Reports* 18.1 (2022), pp. 853–894. URL: https://ems.press/content/serial-article-files/46893.

[81] Sepp Hochreiter and Jürgen Schmidhuber. “Long Short-Term Memory”. In: *Neural Computation* 9.8 (1997), pp. 1735–1780. URL: https://www.researchgate.net/publication/13853244_Long_Short-term_Memory.

[82] Benjamin Lindemann et al. “A Survey on Long Short-Term Memory Networks for Time Series Prediction”. In: *Procedia CIRP* 99 (2021), pp. 73–78. URL: https://www.sciencedirect.com/science/article/pii/S2212827121003796.

[83] Fabio Bonassi et al. “On the Stability Properties of Gated Recurrent Units Neural Networks”. In: *Systems & Control Letters* 144 (2020), pp. 104–748. URL: https://www.sciencedirect.com/science/article/pii/S0167691121001791.

[84] Shaojie Bai, J. Zico Kolter, and Vladlen Koltun. “An Empirical Evaluation of Generic Convolutional and Recurrent Networks for Sequence Modeling”. In: *arXiv preprint* (2018). URL: https://arxiv.org/pdf/1803.01271v1.

[85] Leo Breiman et al. *Classification and Regression Trees*. Wadsworth, 1984. URL: https://www.taylorfrancis.com/books/mono/10.1201/9781315139470/classification-regression-trees-leo-breiman-jerome-friedman-olshen-charles-stone.

[86] Leo Breiman. “Random Forests”. In: *Machine Learning* 45.1 (2001), pp. 5–32. URL: https://doi.org/10.1023/A:1010933404324.

[87] Tianqi Chen and Carlos Guestrin. “XGBoost: A Scalable Tree Boosting System”. In: *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. 2016, pp. 785–794. URL: https://arxiv.org/pdf/1603.02754.

[88] Corinna Cortes and Vladimir Vapnik. “Support-Vector Networks”. In: *Machine Learning* 20.3 (1995), pp. 273–297. URL: https://link.springer.com/article/10.1007/BF00994018.

[89] Stuart Lloyd. “Least Squares Quantization in PCM”. In: *IEEE Transactions on Information Theory* 28.2 (1982), pp. 129–137. URL: https://ieeexplore.ieee.org/document/1056489.

[90] Ricardo J. G. B. Campello, Davoud Moulavi, and Jörg Sander. “Density-Based Clustering Based on Hierarchical Density Estimates”. In: *Pacific-Asia Conference on Knowledge Discovery and Data Mining*. Springer, 2013, pp. 160–172. URL: https://link.springer.com/chapter/10.1007/978-3-642-37456-2_14.

[91] A. P. Dempster, N. M. Laird, and D. B. Rubin. “Maximum Likelihood from Incomplete Data via the EM Algorithm”. In: *Journal of the Royal Statistical Society, Series B* 39.1 (1977), pp. 1–38. URL: https://academic.oup.com/jrsssb/article/39/1/1/7027539?login=true.

[92] Geoffrey E. Hinton and Ruslan R. Salakhutdinov. “Reducing the Dimensionality of Data with Neural Networks”. In: *Science* 313.5786 (2006), pp. 504–507. URL: https://dbirman.github.io/learn/hierarchy/pdfs/Hinton2006.pdf.

[93] Raghavendra Chalapathy and Sanjay Chawla. “Deep Learning for Anomaly Detection: A Survey”. In: *arXiv preprint* (2019). URL: https://arxiv.org/abs/1901.03407.

[94] Ian J. Goodfellow et al. “Generative Adversarial Nets”. In: *Advances in Neural Information Processing Systems*. Vol. 27. 2014, pp. 2672–2680. URL: https://arxiv.org/pdf/1406.2661.

[95] Kuang Luo et al. “Physics-Informed Neural Networks for PDE Problems: A Comprehensive Review”. In: *Artificial Intelligence Review* 58.10 (2025), p. 323. URL: https://link.springer.com/article/10.1007/s10462-025-11322-7.

[96] Guy Revach et al. “KalmanNet: Neural Network Aided Kalman Filtering for Partially Known Dynamics”. In: *IEEE Transactions on Signal Processing* 70 (2022), pp. 1532–1547. URL: https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=9733186.

A

Project Planning

This chapter describes the overall planning of the thesis work, including the work breakdown structure, the project timeline (Gantt chart), milestone dates, expected deliverables, and the duration of each project phase.

A.1. Work packages

The following Table A.1 reflects the necessary work packages (WP) for the success of the thesis, as well as their dependencies and the phase at which they should be completed.

WP #	Phase	Title	Dependencies
1	Literature Review	ML theory papers	-
2	Literature Review	Classical OD papers	-
3	Literature Review	ML OD papers	-
4	Literature Review	Classical manoeuvre detection papers	-
5	Literature Review	ML manoeuvre detection papers	-
6	Literature Review	Gap detection and algorithmic decision	2, 3, 4, 5
7	Mid-Term Phase	TLE and manoeuvre labels collection	-
8	Mid-Term Phase	Residual generation and feature extraction	6, 7
9	Mid-Term Phase	Anomaly scoring algorithm design	6, 8
10	Mid-Term Phase	Event detection algorithm design	9
11	Mid-Term Phase	Analysis of performance and selection of configuration	9, 10
12	Green Light Phase	HMM design	11
13	Green Light Phase	Hierarchical agglomerative clustering algorithm design	12
14	Green Light Phase	Integration of classifier and detector into ISOPA	11, 13
15	Green Light Phase	Assessment of potential prediction improvement	14
16	Finalization Phase	Refinement of thesis report and final deliverables	All

Table A.1: Work packages for the thesis and their dependencies.

The work packages listed in Table A.1 are grouped into four main phases, each with clearly defined objectives and tangible outcomes:

1. **Literature Review Phase (WP 1–6)**: It focuses on establishing a theoretical and methodological basis for the thesis. By the end of this phase, the relevant literature on classical and machine-learning-based orbit determination, and classical and machine-learning-based manoeuvre detection will have been reviewed, as well as machine learning theory, in order to provide the reader with a solid background information. This phase concludes with an identified research gap and a justified algorithmic strategy.

2. **Mid-Term Phase (WP 7–11)**: It is dedicated to data preparation and the development of the core detection methodology. During this phase, TLE datasets and manoeuvre labels are collected, followed by the generation of consistency-based residuals and the extraction of relevant features. Based on these inputs, the anomaly scoring framework is designed, progressively evolving from simple physically motivated formulations to more advanced scoring strategies. Subsequently, an event detection algorithm is developed to convert the continuous anomaly scores into discrete manoeuvre events. The phase concludes with a systematic analysis of performance and the selection of the most suitable configuration based on validation results.
3. **Green Light Phase (WP 12–15)**: It focuses on extending and validating the selected methodology, as well as integrating it into an operational context. In this phase, additional modelling layers are explored, including the design of a Hidden Markov Model (HMM) and a hierarchical agglomerative clustering approach to further structure and refine detection outputs. These components are then integrated into the ISOPA framework, enabling end-to-end processing from TLE ingestion to manoeuvre-aware prediction. Finally, the impact of the proposed detection pipeline on orbit prediction performance is assessed, with the objective of quantifying potential improvements over behaviour-agnostic baselines.
4. **Finalization Phase (WP 17)**: It is dedicated to refining the thesis and delivering the final results. This includes finalizing figures and tables, addressing feedback, and producing the final written thesis and associated deliverables.

A.2. Timeline and Gantt chart

The following Figure A.1 shows the timeline and Gantt chart of all WP. LRP stands for Literature Review Phase, MTP stands for Mid-Term Phase, GLP stands for Green Light Phase and FP stands for Finalization Phase.

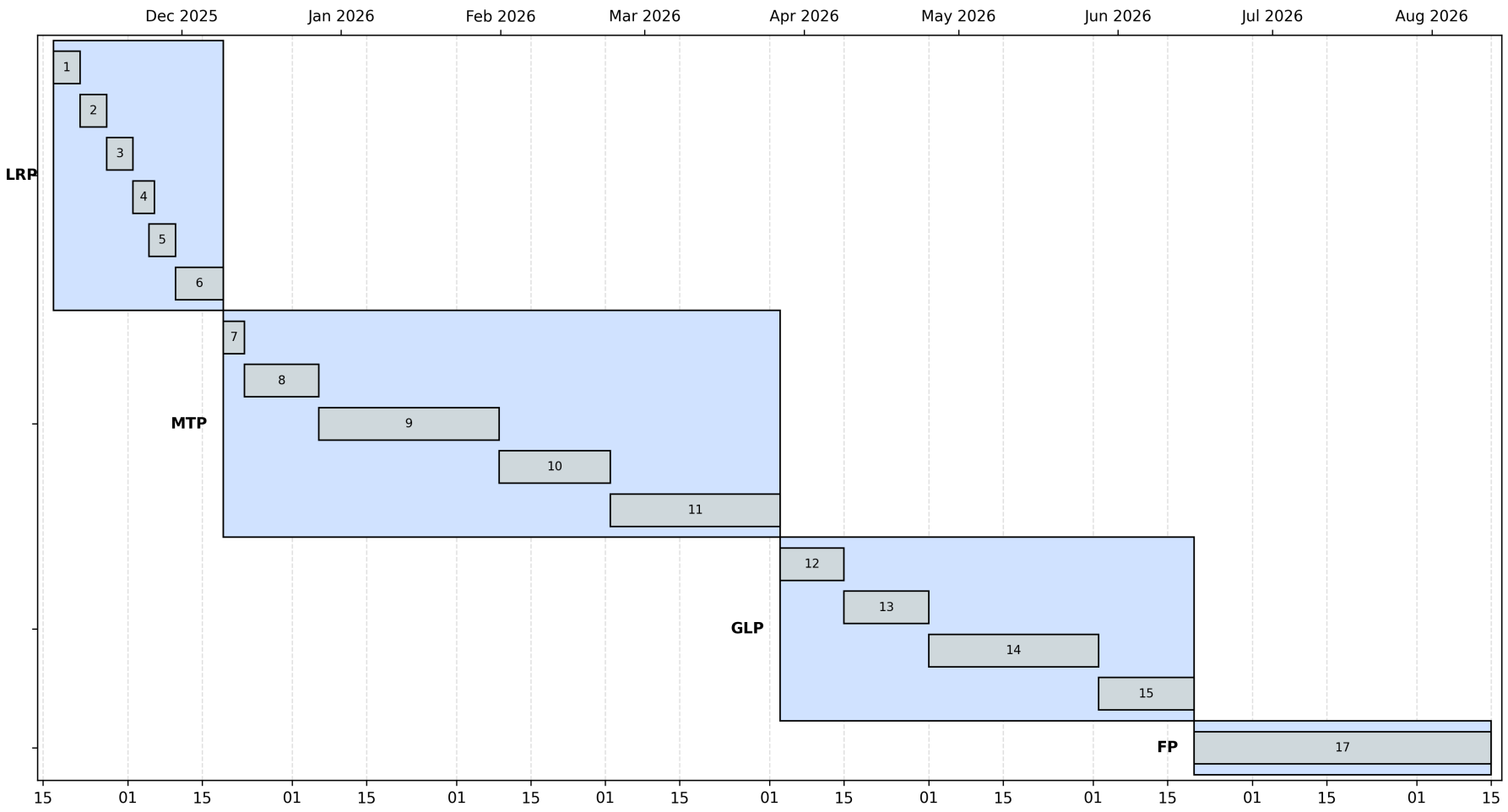

Figure A.1: Thesis Gantt chart

A.3. Milestones dates and deliverables

Table A.2 summarizes the milestone dates, the corresponding deliverables, and the duration of each phase of the thesis project.

Phase	Milestone date	Deliverables	Duration
LRP	18 Dec	Written report	17 Nov – 18 Dec (5 working weeks)
MTP	2 Apr	Oral presentation; written report	19 Dec – 2 Apr (13 working weeks)
GLP	25 Jun	Thesis draft; short presentation summarizing main outcomes	7 Apr – 25 Jun (11 working weeks)
FP	31 Jul	Final thesis document; defence presentation	28 Jun – 31 Jul (4 working weeks)
Defence	14 Aug	—	—

Table A.2: Milestone dates, corresponding deliverables, and phase durations of the thesis.

B

ML theory

B.1. Learning paradigms and modelling dimensions

Machine-learning models can be grouped along several independent dimensions. A first axis is the learning paradigm, which depends on whether ground-truth labels are available during training. In supervised learning, the model receives input–output pairs and learns to predict a target variable, typically by minimizing an expected risk functional [80]. Both sequence models and feature-based classifiers fall under this category. In contrast, unsupervised learning assumes that no labels are provided; the goal is instead to discover structure in the data. These methods learn representations, clusters or anomaly scores from raw or engineered features without explicit supervision. Semi-supervised learning lies in between both cases, exploiting small labelled sets together with larger unlabelled sets, although it is not the main focus here.

A second, independent axis concerns how machine learning interacts with physical models. Hybrid Physics–ML methods form a separate category because they combine a physics-based model with a data-driven component that learns residuals, noise properties or latent behavioural representations. They may internally use supervised models, unsupervised models or both. What distinguishes them is not the presence or absence of labels, but the role of physics in structuring the learning problem and constraining the model.

In the present thesis, hybrid Physics–ML approaches receive particular emphasis because the available datasets consist of residuals and propagated orbital states. For this reason, hybrid methods are presented separately, as a modelling strategy that complements—rather than replaces—the standard supervised/unsupervised taxonomy.

B.2. ML model families

B.2.1. Sequence models

B.2.1.1. Long-Short Term Memory (LSTM)

Standard Recurrent Neural Networks (RNN) suffer when dealing with long-term dependencies because gradients tend to vanish when backpropagated through many time steps [81]. LSTMs address this issue, introducing a cell state that carries information over long horizons and using gating mechanisms to control information flow.

Each LSTM has a cell state (c_t, long-term memory) and a hidden state (h_t, short-term representation). Three gates regulate updates: the input gate decides how much of the new candidate information enters the cell, the forget gate decides how much of the previous cell is kept and the output gate decides how much of the cell state becomes visible in the hidden state.

The key idea is that the cell update combines the old cell state and the new candidate in a nearly linear way, so that gradients can propagate through time with limited decay.

Inputs for this approach could be time series of orbital elements or residuals and outputs could be pre-

dicted future residuals, corrections, or anomaly scores. They are especially useful for orbital dynamics because of their robust handling of long temporal context and ability to model nonlinear couplings, but they carry a higher risk of overfitting and require higher training and inference cost compared to simpler models [82].

B.2.1.2. Gated Recurrent Unit (GRU)

GRUs are basically a simplified alternative to LSTM, with fewer gates and a single hidden-state vector. Input and forget gates are merged into an update gate, and a reset gate controls how much of the past state is used when computing the candidate update [83]. The update rule mixes the previous hidden state and the candidate state according to the update gate. This way, it retains the ability to capture long-term dependencies while reducing the number of parameters and operations compared to LSTMs.

For SSA applications, GRUs work especially well when datasets are relatively small (compared to full LSTM), latency constraints exist (e.g. near-real-time processing) and hardware resources are limited (e.g. in onboard applications). In practice, GRUs often match LSTMs while being faster to train and less prone to overfitting [82, 83].

B.2.1.3. Temporal Convolutional Networks (TCN)

This type of ML framework is tailored for sequences. They use causal convolutions, ensuring that the output at a certain moment depends only on inputs up to that moment, and dilated convolutions, where the convolution skips over time steps according to a dilation factor.

By stacking layers with increasing dilation factors, TCNs achieve a large receptive field using few layers. Residual connections further improve training stability [84]. Because gradients propagate through a feed-forward computation graph rather than through recurrent connections, TCNs tend to have more stable gradients and can be easier to train than deep RNNs.

For orbital dynamics, they can model long-term patterns such as periodic perturbations, focus on local deviations indicative of maneuvers and serve as residual-correction on top of a physical propagator. Design choices determine how many orbits are effectively seen by the model. It is important to choose the receptive field length carefully [84].

B.2.2. Feature-based models

B.2.2.1. Decision trees

A decision tree recursively partitions the feature space into axis-aligned regions. At each node, a split is chosen (feature and threshold) to maximize reduction. Leaves correspond to regions with relatively homogeneous labels [85].

Trees are highly interpretable: if a path is followed from root to leaf as a sequence of “if-then” rules, one can identify a trend. For instance: “if residual RMS over 2 orbits $\geq$ threshold and inferred drag proxy is high, then classify as maneuver”.

In an SSA context, trees can serve as simple baselines for maneuver detection and interpretable models operating on features produced by representation learning. However, they tend to overfit if grown deep without pruning [85].

B.2.2.2. Random Forest (RF)

RFs are ensembles of decision trees trained on randomly resampled versions of the dataset, where each split considers only a random subset of features [86]. This randomness reduces variance, while each individual tree remains a high-variance, low bias learner.

Some of the main properties of RFs are that training involves growing many trees, classification predictions are made by majority vote over the trees and regression predictions are averages over tree outputs. Results show that RF ensembles substantially reduce variance compared to single trees [86].

In SSA applications, RFs are useful when the input is a set of hand-crafted features (like statistics of residuals, changes in orbital elements, etc.), when the task is to classify time windows or satellites into maneuvers vs. no-maneuvers or into maneuver types and when interpretability is needed (feature importance scores). However, large ensembles can be memory- and computer-intensive.

B.2.2.3. Gradient-boosted trees

Gradient boosting-e.g., eXtreme Gradient Boosting (XGBoost)-builds an ensemble of shallow decision trees in a sequential way. Each new tree is trained to predict the negative gradient of the loss with respect to the current ensemble prediction [87].

The objective combines data fit with regularization terms that control tree complexity. Shrinkage (learning rate), subsampling and early stopping prevent from overfitting. The fact that they are tree-based, these methods handle heterogeneous features very well.

In the SSA context, these methods are well suited to datasets of moderate size with tabular features derived from orbital histories. They perform reliably when the SNR is moderate, when interactions between features are important, and when strong performance is required together with fast training and interpretable feature contributions.

B.2.2.4. Support Vector Machines (SVM)

These methods try to find decision boundaries that best separate the classes. Specifically, the one that leaves the largest margin between them. To do this, they solve an optimization problem that penalizes misclassifications while keeping the boundary as simple as possible. Since it is a convex problem, the algorithm is guaranteed to find the global optimum [88].

Kernels are a mathematical trick that let SVMs handle nonlinear data. Instead of relying on the original features, kernels let the model work in a higher-dimensional space where classes are easier to separate.

For SSA, SVMs work well when labelled data is limited but features are informative (summary statistics of residuals, differences in orbital elements) or when a robust classifier is needed with few parameters. However, kernel methods scale poorly to very large datasets due to dense kernel matrices.

B.2.3. Unsupervised learning models

B.2.3.1. K-means clustering

K-means separates data into K clusters by minimizing the sum of squared distances between points and their assigned cluster centroids. The standard algorithm alternates between assigning each point to its nearest centroid and recomputing centroids as means of their assigned points [89].

Theoretically, K-means converges to a local minimum, finding the global optimum is extremely computationally hard, but in practice good solutions are often obtained with careful initialization.

For orbital data, K-means is useful for features summarizing residual patterns or orbital elements over windows. For maneuver detection, they may appear as points that fall far from any centroid. Nevertheless, some limitations of this method are that clusters are assumed to be roughly spherical and equally varianced, and that they are very sensitive to feature scaling and number of clusters.

B.2.3.2. Hierarchical Density-Based Clustering (HDBSCAN)

HDBSCAN is a density-based clustering framework that builds clusters hierarchically organized. It builds this hierarchy from mutual reachability distances and then extracts a flat clustering that optimizes cluster stability [90].

With respect to K-means, it can find clusters of varying density and arbitrary shapes and naturally identifies outliers as points that belong to no stable cluster. It does not require a specific number of clusters beforehand and is more robust in the presence of irregular cluster structure.

In SSA applications, this algorithm is useful for discovering different behavioural modes in a high-dimensional space and for automatically identifying anomalous windows or satellites as noise points [90].

B.2.3.3. Gaussian Mixture Models (GMM)

A GMM models the data distribution as a mixture of Gaussian components, each with its own mean, covariance and mixing weight. The learning algorithm is Expectation-Maximization (EM). The E-step consists on computing responsibilities of each component for each data point and the M-step consists on updating the parameters to maximize the expected log-likelihood [91].

In a maneuver detection framework, GMM could be used to fit a distribution to nominal behaviour and then flag samples with low likelihood as anomalies. This approach is useful due to its probabilistic foundation and interpretable covariance structure, but a Gaussian component assumption may be too restrictive and the algorithm is prone to converging to local optima [91].

B.2.3.4. Autoencoders (AE)

Autoencoders learn a compressed representation (latent code) of the data by training a neural network to reconstruct its inputs. An encoder maps the input into a latent code, and then a decoder reconstructs the original data from this code. Training minimizes reconstruction error [92].

For anomaly detection, it would be useful to train the autoencoder only on nominal data and then, at test time, reconstruction error would serve as an anomaly score: maneuvering sequences would reconstruct poorly.

They are also very powerful representation learners: they can form the front-end of hybrid models where latent space is constrained by physics combined with tree-based classifiers. They are central to representation-learning-based hybrids in SSA [92, 93].

B.2.3.5. Generative Adversarial Networks (GAN)

GANs consist of a generator that produces synthetic samples from random noise and a discriminator that tries to distinguish real from generated samples. Training is formulated as a game where the generator aims to fool the discriminator, and the discriminator aims to classify correctly. At convergence, the generator ideally learns to reproduce the data distribution. In practice, training is very challenging due to issues like mode collapse and instability [94].

For anomaly detection, a GAN can be trained on nominal behaviour, the discriminator could be used to define an anomaly score and maneuver sequences would tend to be scored as ”unlikely” or ”fail” by the discriminator.

Conceptually, they are very powerful combined with physics constraints or residual-learning ideas, but their complexity and instability often make simpler methods more suitable for these applications [94, 93].

B.3. Hybrid Physics-ML models

B.3.1. Residual learning (Physics-informed ML)

In residual learning, the state evolution is decomposed into a baseline physics-based model (SGP4 or similar) and a data-driven residual that captures systematic mismodelling. Then, the true next state is approximated as the physics propagator’s output plus an ML-predicted residual. The ML model is trained on residuals between propagated and observed states. This turns the problem into a supervised learning task where labels are residuals.

The benefits of this approach are that the ML model focuses on correction rather than learning the full dynamics, it preserves physical interpretability, and it can be seamlessly integrated into existing pipelines [95].

B.3.2. ML-assisted Kalman Filtering

The classical KF provides good estimates for linear Gaussian state-space models. Nevertheless, it suffers when the model is not accurate for system dynamics and noise covariance.

A hybrid ML-KF approach retains the essence of KF, but uses ML to learn or adapt process and measurement noise covariances based on statistics, to learn bias or unmodelled dynamics and to interpret innovation sequences to detect maneuvers.

An example is the KalmanNet, where a neural network learns to map innovations, measurements and predictions to corrections in the Kalman gain or state update, while preserving the KF structure. For orbit determination, similar ideas could be used: the core propagation and update would remain physically grounded and ML would handle unknown or time-varying components and help detect regime changes.

The proposed hybridization keeps the best of both approaches: statistical optimality and structure of the KF and the flexibility of ML [96].

B.3.3. Latent-space Physics–ML hybrid models

Unlike residual-learning or ML-assisted filtering, latent-space hybrid models operate by first learning a compressed representation of orbital behaviour and then evaluating physical consistency in this latent space. The hybridisation therefore occurs at the level of representation rather than in state propagation or filtering, making these methods suitable for high-dimensional time series in which manoeuvre signatures or behavioural modes become more separable after nonlinear dimensionality reduction.

In these approaches, unsupervised models (typically autoencoders) compress windows of orbital-element or residual time series into lower-dimensional latent codes that capture the dominant structure of nominal behaviour [92, 93]. Physics-based constraints or checks are then applied to these latent codes, and downstream ML models—often clustering methods—identify deviations from the nominal manifold. Anomalies or manoeuvres appear as latent points lying far from the learned manifold, in density valleys, or in regions corresponding to physically implausible state changes.

C

Per-satellite results of the selected configuration

This appendix provides the detailed per-satellite results for the final selected configuration, corresponding to the onset-sensitive ranker with the SARAL temporal refinement. Results are reported separately for the validation and test splits. Table C.1 shows the results for the validation set, Table C.2 shows the results for the test set, and Table C.3 shows the combined results for the whole timespan of each satellite.

C.1. Validation results

Satellite	TP	FP	FN	Precision	Recall	F_2
CRYO2	97	31	18	0.758	0.843	0.825
ENVI1	55	58	14	0.487	0.797	0.707
HY-2A	30	87	5	0.256	0.857	0.584
HY-2C	21	30	1	0.412	0.955	0.755
HY-2D	21	22	0	0.488	1.000	0.827
JASO1	26	37	11	0.413	0.703	0.616
JASO2	29	20	9	0.592	0.763	0.721
JASO3	20	11	10	0.645	0.667	0.662
SARAL	9	2	0	0.818	1.000	0.957
SEN3A	37	20	7	0.649	0.841	0.794
SEN3B	37	3	8	0.925	0.822	0.841
SEN6A	9	7	9	0.562	0.500	0.511
SWOT1	28	0	14	1.000	0.667	0.714

Table C.1: Per-satellite performance on the validation set for the final selected configuration.

C.2. Test results

Satellite	TP	FP	FN	Precision	Recall	F_2
CRYO2	102	18	3	0.850	0.971	0.944
ENVI1	60	56	34	0.517	0.638	0.610
HY-2A	19	103	4	0.156	0.826	0.444
HY-2C	22	9	0	0.710	1.000	0.924
HY-2D	21	5	0	0.808	1.000	0.955
JASO1	33	46	24	0.418	0.579	0.537
JASO2	22	5	14	0.815	0.611	0.643
JASO3	36	0	10	1.000	0.783	0.818
SARAL	7	16	0	0.304	1.000	0.686
SEN3A	91	25	1	0.784	0.989	0.940
SEN3B	76	10	7	0.884	0.916	0.909
SEN6A	9	0	0	1.000	1.000	1.000
SWOT1	18	4	1	0.818	0.947	0.918

Table C.2: Per-satellite performance on the test set for the final selected configuration.

C.3. Overall results

Satellite	TP	FP	FN	Precision	Recall	F_2
CRYO2	199	49	21	0.802	0.905	0.882
ENVI1	115	114	48	0.502	0.706	0.653
HY-2A	49	190	9	0.205	0.845	0.520
HY-2C	43	39	1	0.524	0.977	0.833
HY-2D	42	27	0	0.609	1.000	0.886
JASO1	59	83	35	0.415	0.628	0.569
JASO2	51	25	23	0.671	0.689	0.685
JASO3	56	11	20	0.836	0.737	0.755
SARAL	16	18	0	0.471	1.000	0.816
SEN3A	128	45	8	0.740	0.941	0.893
SEN3B	113	13	15	0.897	0.883	0.886
SEN6A	18	7	9	0.720	0.667	0.677
SWOT1	46	4	15	0.920	0.754	0.782

Table C.3: Per-satellite performance considering the combined validation and test datasets for the final selected configuration.