ORIGINAL ARTICLE

Clustering astronomical orbital synthetic data using advanced feature extraction and dimensionality reduction techniques

Eraldo Pereira Marinho · Nelson Callegari Junior · Fabricio Aparecido Breve · Caetano Mazzoni Ranieri

Received: 30 September 2025 / Accepted: 25 February 2026

Abstract
The dynamics of Saturn's satellite system offer a rich framework for studying orbital stability and resonance interactions. Traditional methods for analysing such systems, including Fourier analysis and stability metrics, struggle with the scale and complexity of modern datasets. This study introduces a machine learning-based pipeline for clustering $\sim 22,300$ simulated satellite orbits, addressing these challenges with advanced feature extraction and dimensionality reduction techniques. The key to this approach is using MiniRocket, which efficiently transforms 400 timesteps into a 9,996-dimensional feature space, capturing intricate temporal patterns. Additional automated feature extraction and dimensionality reduction techniques refine the data, enabling robust clustering analysis. This pipeline reveals stability regions, resonance structures, and other key behaviours in Saturn's satellite system, providing new insights into their long-term dynamical evolution. By integrating computational tools with traditional celestial mechanics techniques, this study offers a scalable and interpretable methodology for analysing large-scale orbital datasets and advancing the exploration of planetary dynamics.

Keywords Unsupervised clustering · Time-series feature extraction · Celestial mechanics · Outlier detection

Mathematics Subject Classification 68T09 · 37N05 · 85-08 · 62H30 · 65T50

1 Introduction

The temporal evolution of celestial bodies' orbits, particularly satellites' orbits around planets, is a complex process influenced by numerous dynamical factors. Departing with initial states close to the real orbit of a determined body, a sheer ensemble with typically tens of thousands of initial orbits is numerically integrated for very long periods. As a result, a massive amount of time series of representative orbital elements are generated for each clone orbit. A powerful tool often applied in dynamical astronomy is the construction of dynamical maps. This technique consists of plotting some stability criterion on a two-dimensional map, ready to furnish the main regions of the phase space with physical interest in terms of the long-term stability and interactions of clones of real satellites [1]. Clustering behaviours among simulated orbits offer a powerful framework for deciphering

Published online: 22 April 2026

the dynamical structure of planetary systems. Researchers can identify stability zones, resonance structures, and evolutionary pathways by grouping orbits with similar properties furnished by the time series. For this task, traditional methods such as Fourier analysis and numerical stability metrics can be applied and are effective, though computationally expensive and struggle to scale with the large, high-dimensional datasets generated by modern simulations (e.g. [2–4]).

This study presents a novel machine-learning-based pipeline designed to analyse and cluster the time series, effectively addressing the scalability and complexity limitations of traditional methods in celestial mechanics. At the core of this pipeline is MiniRocket [5], a state-of-the-art method that performs feature extraction from time series using random convolutional kernels. In our case, MiniRocket transforms raw 400-step time series data into a high-dimensional feature space with 9,996 features. By utilising convolutional kernels with precisely tuned dilation and weights, MiniRocket efficiently captures intricate temporal patterns, making it particularly suited to the time series characteristics of orbital data. While recent advancements in time series clustering, such as random convolutional kernels, have demonstrated remarkable improvements in efficiency and scalability [6], this study adopts traditional convolutional kernels for their interpretability and established effectiveness in capturing key dynamical features. Additionally, TSFresh automates the extraction of interpretable features, enabling the identification of statistically significant patterns across the orbital dataset [8].

Dimensionality reduction techniques refine the high-dimensional features that MiniRocket, and TSFresh extract. Linear methods, such as Principal Component Analysis (PCA), reduce dimensionality while preserving the most significant variances [9]. Non-linear approaches, including t-SNE [10], UMAP [11], and autoencoders [12], uncover complex, non-linear relationships within the data. These combined methods enable a comprehensive and interpretable clustering analysis of the satellite assembly of orbits.

The remainder of this paper is structured as follows. Section 2 provides a comprehensive review of related work in the context of time series clustering and orbital dynamics. In Sect. 3, we detail the proposed methodology, including the combined feature extraction procedures and dimensionality reduction strategies employed. Section 4 addresses the hyperparameter optimisation process adopted throughout the study. A preliminary validation of the approach is presented in Sect. 5, through the clustering of time series generated by the classical simple pendulum. Section 6 presents the application of the proposed clustering framework to a dataset composed of numerically integrated orbital time series corresponding to an ensemble of fictitious Saturnian satellites, inspired by discoveries from the Cassini-Huygens Planetary Mission. Finally, Sect. 7 summarises the main conclusions and outlines potential directions for future work.

2 Related work

The analysis of orbital dynamics in celestial mechanics has traditionally relied on numerical simulations and stability metrics, such as those derived from Fourier analysis [1]. These methods have proven effective for understanding resonance structures and stability zones, particularly in planetary systems like Saturn's satellite system. However, their computational cost and inability to scale with large datasets pose significant challenges in the era of modern astronomical simulations.

Recent advances in machine learning have paved the way for more efficient and scalable approaches to analysing high-dimensional time series data. Feature extraction techniques like TSFresh [8].

The introduction of random convolutional kernels for time series analysis and clustering, as in [6], represents a further step in improving scalability and efficiency. These methods leverage randomly initialized convolutional filters to extract meaningful patterns without the need for extensive training, offering competitive performance in clustering tasks. Similarly, MiniRocket [5] has established itself as a state-of-the-art feature extractor, transforming raw time series data into a high-dimensional feature space that captures both local and global temporal dynamics with exceptional efficiency.

While these advancements have shown promise, gaps remain in applying them to large-scale astronomical datasets, such as those involving orbital dynamics. Many studies focus on general-purpose time series data or limited domains, leaving the integration of machine learning with classical astronomical methods underexplored. This study addresses these gaps by introducing a comprehensive pipeline that combines the scalability of MiniRocket with interpretable feature extraction methods and robust clustering techniques specifically tailored to the complex dynamical interactions in Saturn's satellite system.

3 Methodology

This study employs a machine learning-based pipeline to analyse and cluster the orbital dynamics of Saturn satellites. The data represent the orbits of small satellites in the Saturnian system, specifically focusing on the dynamics of Anthe (a small moon of Saturn) and its resonance interactions with Mimas (a mid-sized moon of Saturn). The numerical simulations model the orbital motion of the test satellites as they orbit Saturn, considering the gravitational influences of both the planet and its larger moons [2].

The dataset D consists of 22, 288 samples, each comprising a time series $d \in D$ with 400 timesteps. Each sample corresponds to a pair of variables representing the initial state of the system. The variables contained in the time series are two angles φ_1 and φ_2, characterizing the system's capture in the so-called Corotation and Lindblad resonances, respectively, where motion remains confined around stable equilibria states. The definition of the angles and more details on the resonant dynamics is given in Sect. 6.

3.1 Machine learning pipeline

The machine learning pipeline, shown in Fig. 1, was designed to integrate advanced feature extraction, dimensionality reduction, and clustering methods for data processing. The diagram is intended as a *conceptual pipeline*, rather than a representation of parallel execution or data-flow concurrency. The subsequent subsections describe each step in the pipeline, including specific implementation details for reproducibility.

3.1.1 Feature extraction

Before feature extraction, each time series is normalized using per-series z-score normalization: the mean and standard deviation are computed over the 400 timesteps of each orbit, and the orbit is standardized independently. No explicit detrending or windowing procedure is applied. This choice ensures that the feature extraction stage focuses on the temporal structure of the signal rather than absolute offsets or scale differences, and it also affects the interpretation of Euclidean and cosine distances after concatenating heterogeneous feature blocks.

As shown in Fig. 1, the proposed feature engineering pipeline is executed sequentially, with each module receiving the output of the previous step as input. Therefore, unlike parallel feature extraction schemes based on concatenation, each transformation progressively refines the time-series representation, and the resulting feature dimensionality evolves across stages according to the specific operation applied. Z-score standardization is applied to the input of each feature extraction module to normalize the signal scale. This preprocessing step does not alter the representation's dimensionality.

Each experiment activates a specific subset of extractors, while the remaining ones are disabled. This design implements an ablation strategy over feature representations. Conceptually, this stage is represented by a vertical feature bus, from which individual extractors can be selectively enabled. Each enabled extractor maps the normalized time series into a feature space of fixed dimensionality. When more than one extractor is active, the output of the previous block is fed to the input of the next one.

Four techniques were considered. These techniques were the MiniRocket method [5], the Fast Fourier Transform (FFT) and DWT [13], and TSFresh [8]. The resulting feature shapes at each step are detailed in Table 1.

 https://doi.org/10.1007/s00521-026-12057-4

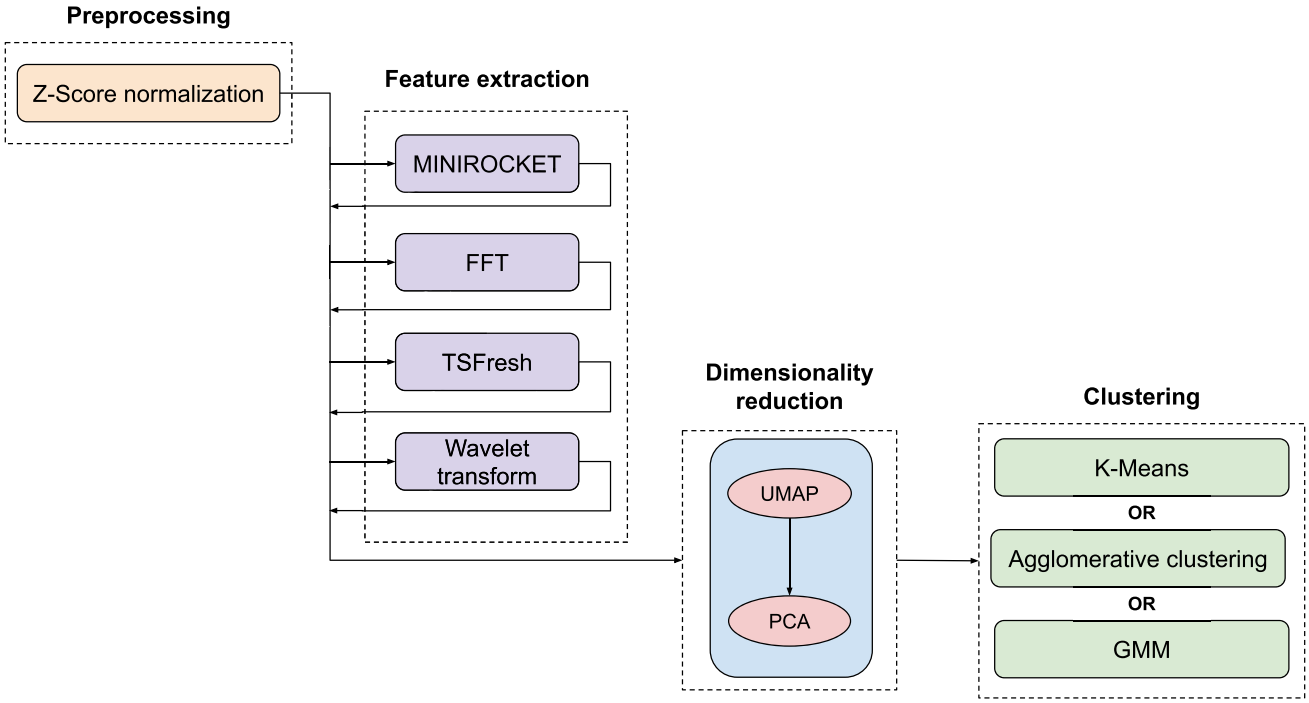

Fig. 1 Proposed pipeline for clustering time-series data. The data shape at each stage is shown to improve reproducibility. Feature extraction modules are optional and are enabled or disabled to form different feature subsets (ablation study). Only the selected subset is concatenated to build the final feature vector; see Table 1 for the evaluated configurations and best-performing pipelines

Table 1 Feature dimensions at each extraction step

Feature Extraction Method	Feature Dimensions
MiniRocket	(22,288, 9,996)
FFT	(22,288, 201)
Wavelet Transform	(22,288, 401)
TSFresh	(22,288, 794)
Combined Features	**(22,288, 11,375)**

Bold values indicate the best performance for each metric

MiniRocket is a method initially designed for time series classification. It transforms time series data into a feature space using convolutional kernels, enabling linear classifiers like ridge regression or logistic regression to achieve high accuracy with the generated feature vectors. Unlike its predecessor (i.e., Rocket [14]), MiniRocket employs a small, fixed set of carefully designed kernels, making it computationally efficient and up to 75 times faster on large datasets while maintaining comparable accuracy to state-of-the-art methods.

In this work, we adapt MiniRocket as a feature extractor for time series clustering. By processing its output feature vectors through dimensionality reduction techniques, we obtain a lower-dimensional representation suitable for clustering methods. This approach leverages MiniRocket's efficiency and effectiveness, extending its utility beyond classification to unsupervised tasks.

We have also applied other techniques to extract features from the time series and enhance the representation fed to the clustering methods. The Fast Fourier Transform (FFT) breaks down a signal into its frequency components by efficiently computing the Discrete Fourier Transform (DFT) or its inverse, the Inverse Discrete Fourier Transform (IDFT). FFT converts a time-domain signal into its frequency-domain representation, revealing characteristics such as periodic patterns, dominant frequencies, and spectral energy distribution. This transformation is beneficial for analysing time series data, as it enables the extraction of frequency-based features that capture

underlying patterns and trends. Applying FFT produces a frequency vector with half the dimensionality of the input signal.

DWT decomposes a signal into scaled and shifted versions of a wave-like function known as a wavelet. Unlike the Fourier transform, which only provides frequency information and assumes that the signal is stationary, the DWT delivers a time-frequency representation. This representation reveals how different frequency components change over time. Wavelets offer a versatile framework for analysing real-world signals' timing and frequency characteristics, allowing for localized analysis across various scales.

Finally, TSFresh is a framework for automating feature extraction and selection in time series data. It combines signal processing and statistical techniques to uncover meaningful patterns. Central to TSFresh is the FRESH algorithm, which uses hypothesis testing to assess the relevance of extracted features for classification or regression tasks. This approach ensures that only significant features are retained, minimizing the risk of overfitting and improving model generalization. The framework offers 63 feature calculators that generate a total of 794 features, covering metrics such as distribution, entropy, and stationarity.

The dimensionality of the resulting feature vectors depends on the last feature extractor applied in the pipeline, following the dimensions shown in Table 1. Only the best-performing configurations are reported in the benchmark results (Table 2).

Table 2 summarises a selection of the most relevant pipeline configurations, ranked according to the Silhouette score, followed by the Davies–Bouldin (DB) and Calinski–Harabasz (CH) indices. Each result corresponds to a pipeline whose parameters were optimised through an exhaustive grid search over the hyperparameter space. For each configuration, the table reports the feature extraction methods used, the clustering algorithm, the dimensionality reduction parameters (PCA and UMAP), the achieved clustering scores, and the sample distribution across the resulting clusters.

3.1.2 Dimensionality reduction

Dimensionality reduction (**DR**) is the next step in the pipeline. The feature vectors generated by the techniques discussed in the previous subsection are often high-dimensional, which complicates their application in clustering methods. This study utilised two complementary techniques to address this issue: Principal Component Analysis (PCA) for linear DR and Uniform Manifold Approximation and Projection (UMAP) for non-linear DR.

Principal Component Analysis (PCA)

PCA reduces the dimensionality of data by projecting it onto a set of orthogonal axes that maximize variance. It is particularly effective for identifying the directions of the highest variance in a dataset, which often represent the most informative aspects of the data. Given a dataset with n samples and d features, represented as a two-dimensional matrix $\mathrm{X} \in \mathbb{R}^{n \times d}$, it begins by centring the data to have zero mean and computing the covariance matrix C, as shown in Eq. 1.

$$\mathrm{C} = \frac{1}{n}\mathrm{X}^{\top}\mathrm{X} \tag{1}$$

Next, the eigenvalues λ_i and eigenvectors v_i of C are computed to satisfy Eq. 2, where λ_i indicates the amount of variance explained by the corresponding principal component v_i.

Table 2 Compact Benchmark Table (Landscape + Shrunk)

Features	Method	PCA	UMAP	Silhouette	DB	CH	Clusters
Minirocket, wavelet, tsfresh	K-means	2	55	0.6830	0.4176	115173.8695	C0:9671, C1:4452, C2:2528, C3:5637
Minirocket, tsfresh	Agglomerative	3	50	0.6830	0.4424	52505.8749	C0:7847, C1:10152, C2:2527, C3:1762
Minirocket, fft, tsfresh	GMM	2	90	0.6810	0.4612	95632.9438	C0:9316, C1:5582, C2:4893, C3:2497
Minirocket, tsfresh	K-means	2	20	0.6809	0.4338	109598.8835	C0:5618, C1:9647, C2:2528, C3:4495
Minirocket, fft, tsfresh	K-means	2	60	0.6722	0.4334	106145.4625	C0:9685, C1:5625, C2:2525, C3:4453
Minirocket, wavelet, tsfresh	GMM	2	50	0.6650	0.4308	104599.3775	C0:6254, C1:2528, C2:9208, C3:4298

$$\mathrm{C}\mathrm{v}_i = \lambda_i \mathrm{v}_i \tag{2}$$

The data is projected onto the first k principal components (i.e., eigenvectors with the largest eigenvalues) as defined in Eq. 3, where $\mathrm{W} \in \mathbb{R}^{d \times k}$ contains the top k eigenvectors. The parameter k, arbitrarily defined by the practitioner, is proportional to the amount of variance preserved in the new representation.

$$\mathrm{Z} = \mathrm{XW} \tag{3}$$

The resulting matrix Z comprises a linear approximation of the original data represented by X, simplifying clustering by reducing noise and redundant dimensions.

Uniform Manifold Approximation and Projection (UMAP)

UMAP is a non-linear DR technique that captures complex, non-linear structures in the data by preserving local and global relationships [11]. Unlike PCA, which assumes linearity, UMAP seeks to uncover the best low-dimensional representation by approximating the data's intrinsic manifold. The algorithm involves two main stages: graph construction and graph layout optimisation.

Graph construction is performed by modelling the data as a weighted k-neighbour graph to represent the local structure of the data. A probabilistic measure derived from the distance between points determines the graph's weights. In the graph layout optimisation stage, UMAP optimises a low-dimensional representation of the data using a force-directed graph layout. The optimisation minimises the cross-entropy between the high-dimensional and low-dimensional representations, effectively preserving the topology of the original dataset.

As a result, UMAP finds the best-curved surface (i.e., subspace) to represent the intra-data structure. By preserving local neighbourhoods and global data dispersion, UMAP (i) provides a precise representation of non-linear relationships and (ii) enhances cluster separability in low-dimensional space.

Alternative DR control (PaCMAP)

As a robustness check, we replaced UMAP with PaCMAP while keeping all other stages of the pipeline fixed (feature stack, PCA dimensionality, clustering configuration, and hyperparameters). For φ_1, using 170 embedding components and 100 neighbors followed by PCA (2 components) and K-means ($k = 4$), we obtained a Silhouette score of 0.5645, a Davies–Bouldin index of 0.5971, and a Calinski–Harabasz index of 52583.1. These values are comparable to those obtained with UMAP, indicating that cluster separability is not materially affected by the specific choice of neighborhood-preserving DR method.

Combining UMAP and PCA

In the proposed pipeline, DR is performed by connecting the UMAP output to the PCA input. UMAP is used to identify and capture non-linear patterns in the high-dimensional feature space, while PCA subsequently refines this reduction linearly. The parameters for both methods are fine-tuned to maximize the silhouette score, ensuring optimal cluster cohesion and separation. This approach leverages the strengths of both techniques, providing an effective representation of the data for clustering.

We apply UMAP before PCA (see Fig. 1) because UMAP is the main non-linear manifold learning step and benefits from operating directly on the full high-dimensional feature representation. PCA is then applied as a lightweight linear refinement/compression step to obtain a compact representation (2–3 dimensions) for visualization and to improve clustering stability. In contrast, applying PCA before UMAP would impose an a priori linear projection that may discard low-variance components that could still be relevant for defining neighbourhood relations in the original feature space. This design treats PCA not as a DR precursor, but as a final orthogonal projection of the UMAP embedding.

3.1.3 Clustering

The reduced representations are clustered using standard unsupervised algorithms (K-Means, Agglomerative Clustering, or Gaussian Mixture Models). No feedback from the clustering stage is propagated upstream: feature extraction, DR, and clustering are strictly decoupled steps in the pipeline.

The clustering problem addressed in this work is formulated and solved in a fully unsupervised setting. No physical labels are assigned a priori to individual trajectories, and no dynamical diagnostic (e.g., resonance indicators, frequency analysis, or chaos indicators) is used during feature extraction, DR, or clustering.

In this context, standard external validation in the supervised learning sense, namely, comparison against a complete and predefined ground truth partition, is not directly applicable, as no such global labelling exists for the ensemble under study.

In Hamiltonian systems such as the one considered here, dynamical regimes including corotation, Lindblad resonances, and chaotic behaviour are well characterised locally in phase space and extensively documented in the literature. However, these regimes do not define a global, exhaustive, and mutually exclusive partition of all trajectories. Many orbits occupy transitional regions, exhibit mixed behaviour depending on the timescale considered, or require multiple diagnostic tools to be meaningfully characterised. Consequently, assigning a single, unambiguous dynamical label to each of the 22,288 orbits would require additional modelling assumptions and long-term integrations beyond the scope of the present study.

For this reason, clustering performance is primarily assessed using internal validation indices–namely the Silhouette score, the Davies–Bouldin index, and the Calinski–Harabasz index–which are appropriate in fully unsupervised scenarios and are used for model selection and pipeline comparison.

The number of clusters k is not treated in this work as a purely data-driven hyperparameter, but rather as a physics-informed modelling choice. In Hamiltonian dynamical systems, the phase space is often structured into a small number of well-characterised regimes (e.g., libration zones, circulation domains, chaotic layers, and non-resonant regions), which motivates selecting k according to the expected qualitative partition of the dynamics. In the simple pendulum validation test, $k = 3$ was intentionally chosen to match the three canonical regimes visible in the Hamiltonian portrait (oscillation, prograde circulation, and retrograde circulation). In the Saturnian resonance dataset, we set $k = 4$ to remain consistent with the dynamical mapping reported by Callegari and Yokoyama [2], who identify four dominant orbital regimes in the 11:10 Anthe–Mimas resonance region. This strategy ensures that the clustering output remains interpretable in terms of established dynamical behaviour rather than being solely driven by statistical criteria.

Physical knowledge of the system is incorporated a posteriori through the interpretation of the resulting clusters in dynamical maps and their qualitative consistency with known phase-space structures reported in previous studies. This provides a physics-informed interpretative validation complementary to the internal indices.

3.1.4 Distance metrics

The choice of distance metrics plays a pivotal role in clustering algorithms, especially for time series data. In this study, the Euclidean distance was employed as the main metric due to its computational efficiency, mathematical simplicity, and compatibility with clustering methods such as K-Means and Gaussian Mixture Models. Euclidean distance remains the default measure for these methods, as they assume centroid-based optimisation in feature space [15].

Dynamic Time Warping (DTW):

Dynamic Time Warping (DTW) is widely recognized for its ability to align time series sequences by handling temporal shifts and distortions [16]. However, DTW was not used in this study for the following reasons:

- **Computational Overhead:** Constructing a pairwise DTW distance matrix for a dataset of 22, 288 time series, each of length 400, incurs a prohibitive time complexity of $\mathcal{O}(N^2T^2)$, where N is the number of series and T is the series length [17].
- **Suboptimal Results:** Initial experiments revealed no significant performance improvement with DTW compared to Euclidean distance, likely due to the extracted feature space reducing the need for temporal alignment.

Alternative Metrics

While Euclidean distance performed well in this study, other distance metrics remain relevant for time series clustering and high-dimensional feature spaces:

- **Manhattan Distance (L1 Norm):** Robust to outliers, it computes the sum of absolute differences [18].
- **Cosine Distance:** Particularly useful in high-dimensional spaces where the angular similarity is more informative than magnitude differences [19].
- **Correlation Distance:** Suitable for clustering time series with similar trends but differing amplitudes [20].
- **Mahalanobis Distance:** Effective for feature spaces with correlated variables and differing variances by incorporating the covariance matrix [21].

Future Considerations

Although DTW and other advanced distance metrics are beneficial for time series clustering, their computational cost must be balanced against dataset size and runtime constraints. Incorporating DR techniques (e.g., PCA, UMAP) prior to clustering can alleviate computational burdens while preserving structural relationships in the data [10, 11].

Cosine or Mahalanobis distances can be beneficial in some high-dimensional settings. However, we adopted Euclidean distance primarily because (i) it is the native objective for centroid-based clustering and Gaussian mixtures, (ii) it remains stable and efficient at the scale considered, and (iii) the feature engineering stage (per-series z-normalization and heterogeneous descriptors) already mitigates scale effects. A systematic comparison of alternative metrics (e.g., cosine, whitening) is a natural extension and is left for future work.

3.2 Outlier Repositioning via Graph Diffusion (ORG-D)

The output of the clustering step often contains noise due to various factors; i.e., some elements of one cluster are mistakenly assigned to another. Particle Competition and Cooperation (PCC) [24] was originally a nature-inspired graph-based semi-supervised classification method. However, in this paper, it is used to eliminate noise by repatriating elements to their rightful cluster. This application is possible because the method was designed with the assumptions of cluster and label smoothness in mind. For instance, according to Fig. 2, there are some misplaced points in the wrong clusters. For example, the leftmost yellow point, labelled cluster 3, is located in a region typical of cluster 0.

First, a graph is built where each node represents a satellite orbit. Each node is connected to its k nearest neighbours, based on their respective values of semi-major axis and eccentricity. The graph is then fed to PCC.

The nodes are randomly divided into 10 folds. Nodes belonging to one of the folds are presented to PCC with their respective cluster labels, while the remaining are presented unlabelled. The algorithm will spread the labels from the labelled nodes to the unlabelled nodes using a transductive approach, which works as follows.

A particle is generated for each labelled node, which will be called the home node of the particle. Particles will walk around the graph, trying to dominate as many nodes as possible. Particles that have the same label are said to belong to the same team and act cooperatively, while competing with particles from other labels/teams. It is an iterative process where particles use a random-greedy approach to decide which node to visit next, prioritizing nodes closer to their home node and nodes that their team has higher domination over.

At the end of the iterative process, the long-term domination levels [25] are used as the probabilities for all labelled and unlabelled nodes. The whole process is repeated for each of the 10 folds. The entire fold division is repeated 100 times, for a total of 1000 PCC executions, so the probabilities for each node are the average of 1000 executions, maximizing their reliability.

After all executions, if an orbit has a higher probability of belonging to another cluster than the one to which it was assigned in the clustering step, it is repatriated, i.e., re-grouped to its rightful cluster.

Figure 3 shows the results of this process applied to the image in Fig. 2 using $k = 24$. Similarly, Fig. 4. shows the results of this process applied to the image in Fig. 5 using $k = 24$. From the comparison of the two pairs of figures, it is clear that the outliers were effectively relabelled.

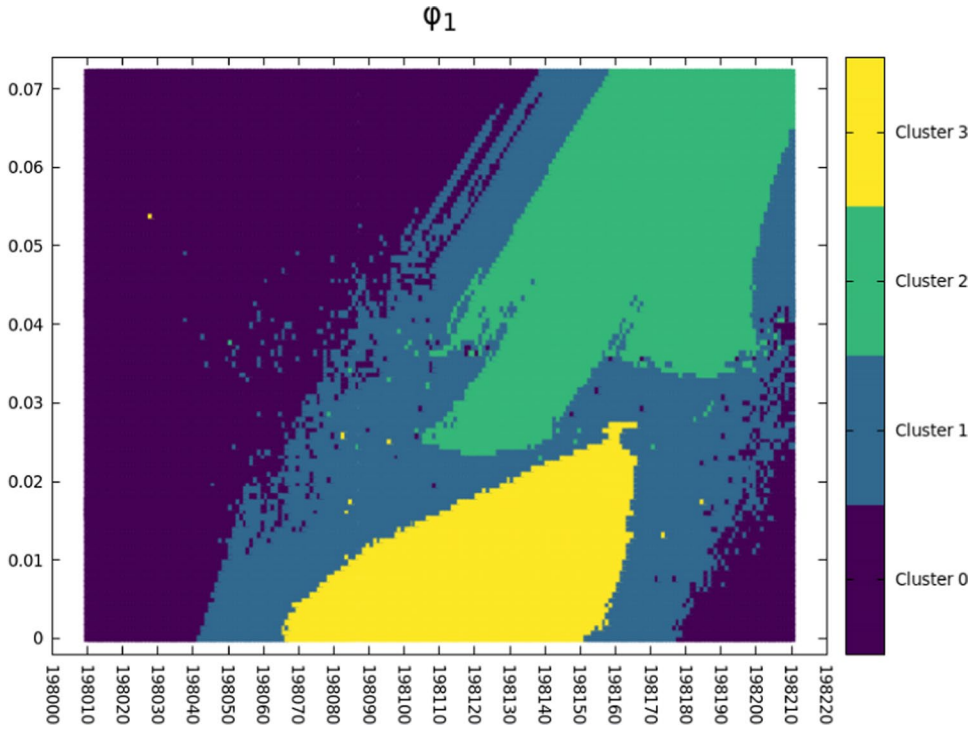

Fig. 2 Dynamical map representing the clustering of initial orbital conditions in the space of semi-major axis versus initial eccentricity for the angle φ_1. The colour bar indicates different clusters obtained through the K-Means algorithm, capturing distinct dynamical behaviours in the orbital phase space. The horizontal axis represents the semi-major axis, and the vertical axis represents the eccentricity. Outliers are evident throughout the dynamical map

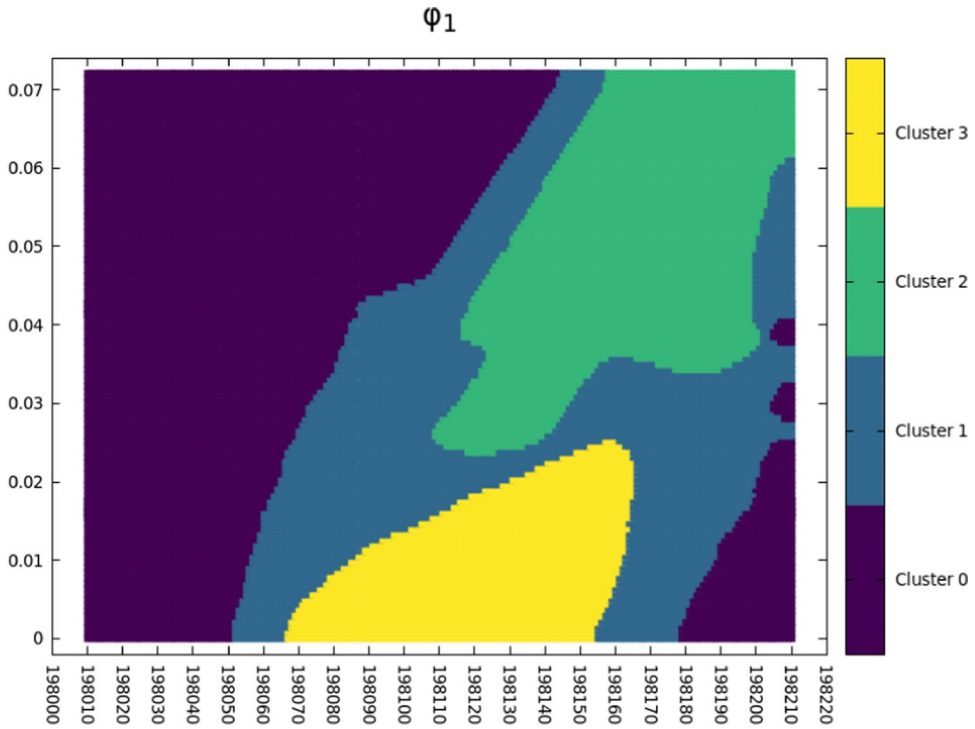

Fig. 3 The same context of Fig. 2 with outliers repositioned using $K = 24$ nearest neighbours

3.2.1 PCC-based soft labels and per-orbit entropy

To quantify the confidence of the PCC-based repatriation step, we compute a *per-orbit* uncertainty score from the PCC "dominance levels". For each orbit (graph node) *i*, PCC yields a soft-label vector $\mathrm{p}_i = (p_{i,1}, \ldots, p_{i,K})$, where $p_{i,k} \in [0, 1]$ represents the (normalized) long-term domination level of team *k* at node *i*, and $\sum_{k=1}^{K} p_{i,k} = 1$.

 https://doi.org/10.1007/s00521-026-12057-4

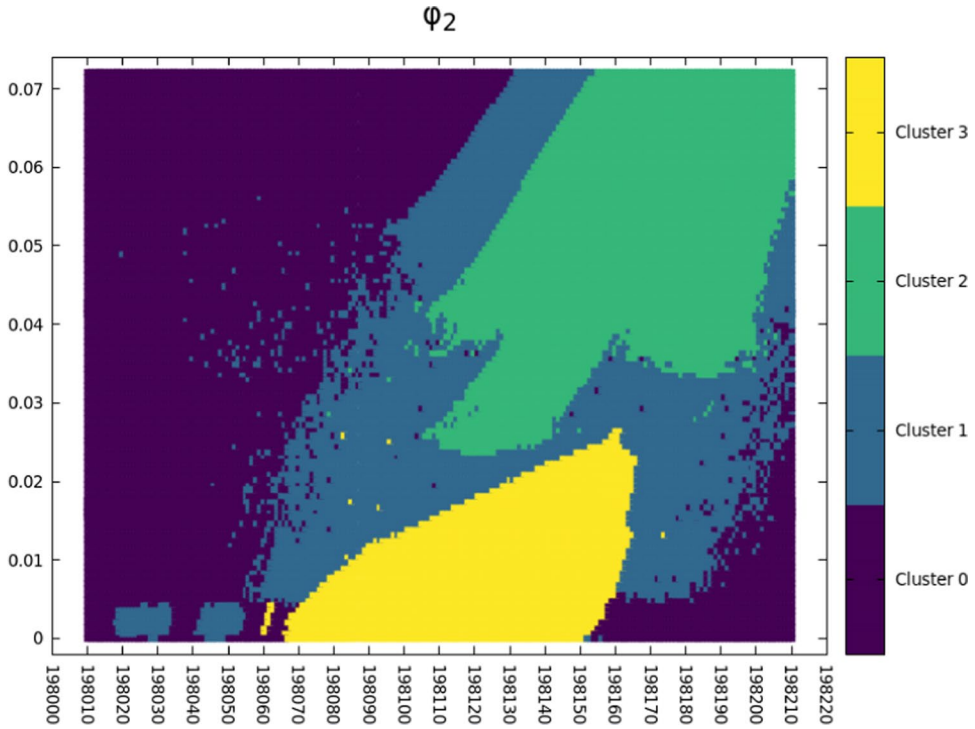

Fig. 4 As in Fig. 2 but for the angle φ_2

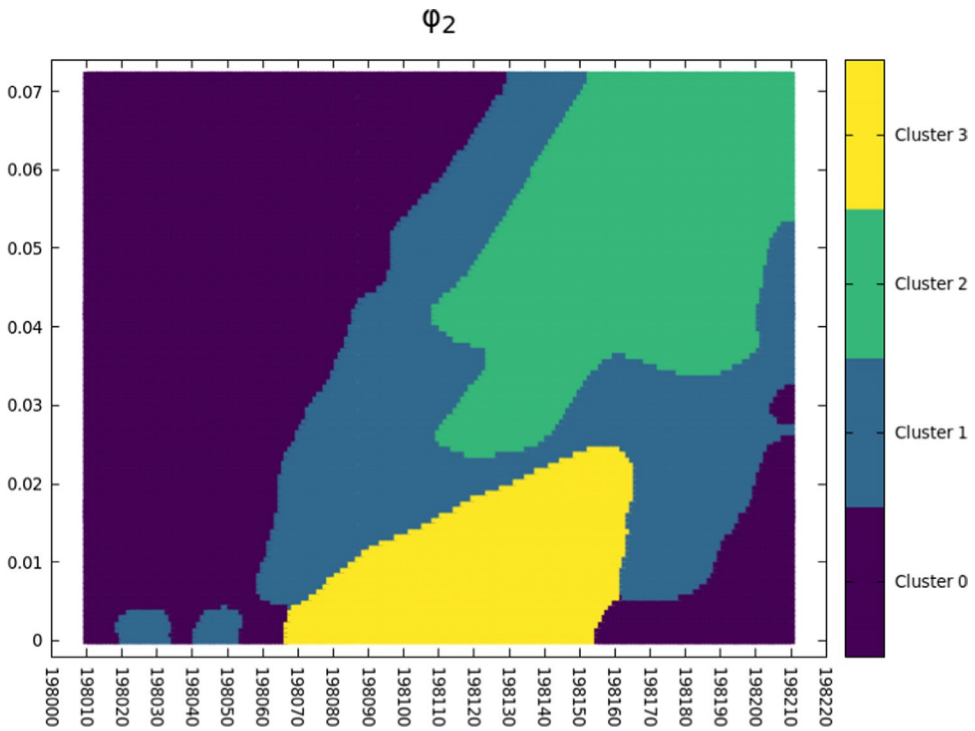

Fig. 5 The same context of Fig. 4 with outliers repositioned using $K = 24$ nearest neighbours

In our implementation, the final p_i is obtained by averaging multiple PCC runs (different random fold assignments) and then renormalizing.

We then define the Shannon entropy of the soft labels as

$$H_i = -\sum_{k=1} p_{i,k} \log(p_{i,k} + \varepsilon), \tag{4}$$

where ε is a small constant used only for numerical stability. Unless stated otherwise, $\log(\cdot)$ denotes the natural logarithm; using $\log_2$ changes H_i only by a constant scaling factor. Entropy is computed *per orbit*: $H_i \approx 0$ indicates a highly dominant team (near-deterministic assignment), while larger values indicate ambiguous membership across teams. The theoretical maximum is $H_i = \log K$ for the uniform distribution $p_{i,k} = 1/K$.

For visualization in the (a, e) plane, we define an entropy field $H(a, e)$ by binning orbits on a regular grid in (a, e) and plotting the mean entropy per bin (this binning is used *only* for visualization purposes). Figures 6–7 show entropy maps for φ_1 and φ_2. Importantly, these maps do not aim to reproduce the categorical dynamical map; instead, they summarize the spatial distribution of *soft-label uncertainty* induced by the PCC-based repatriation method.

Most of the trajectories reassigned by the PCC/ORG-D procedure are located in regions of elevated entropy (lighter lanes) in the PCC-induced soft labelling, consistently observed for both angular variables, φ_1 and φ_2. This indicates that the reassignment step primarily affects trajectories lying in locally ambiguous regions of the embedding space, while leaving well-defined regimes largely unchanged.

3.2.2 Partition agreement measures (ARI, NMI)

To quantify the global impact of the PCC-based ORG-D on the original clustering, we employ standard *partition agreement measures*. These indices compare two partitions of the *same dataset* – the cluster assignments obtained before and after the ORG-D step – and therefore do not rely on any external ground truth labels. Their purpose is not to assess clustering quality, but to measure how strongly the ORG-D modifies the initial partition structure.

Let $U = \{U_1, \ldots, U_K\}$ and $V = \{V_1, \ldots, V_K\}$ denote the partitions produced by K-Means before and after the PCC-based repatriation, respectively.

Adjusted Rand Index (ARI)

The Adjusted Rand Index (ARI) measures the similarity between two partitions while correcting for chance agreement [31]. It is defined as

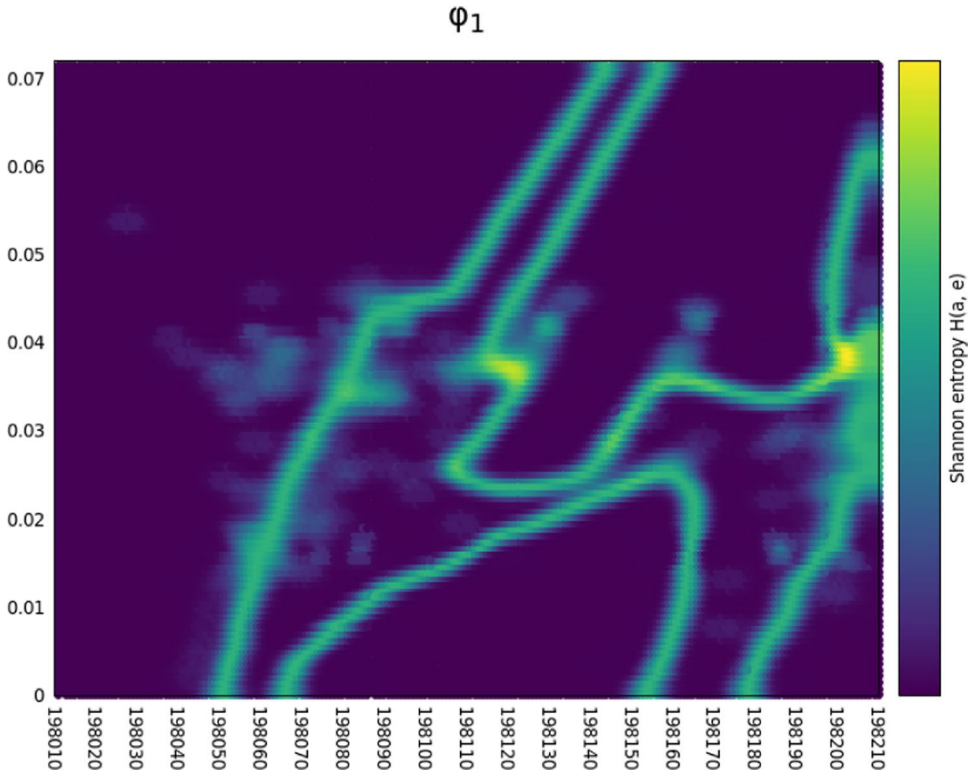

Fig. 6 Entropy map in the (a, e) plane for φ_1, computed from PCC-based soft labels using Eq. (4). The field is shown as the entropy per (a, e) bin

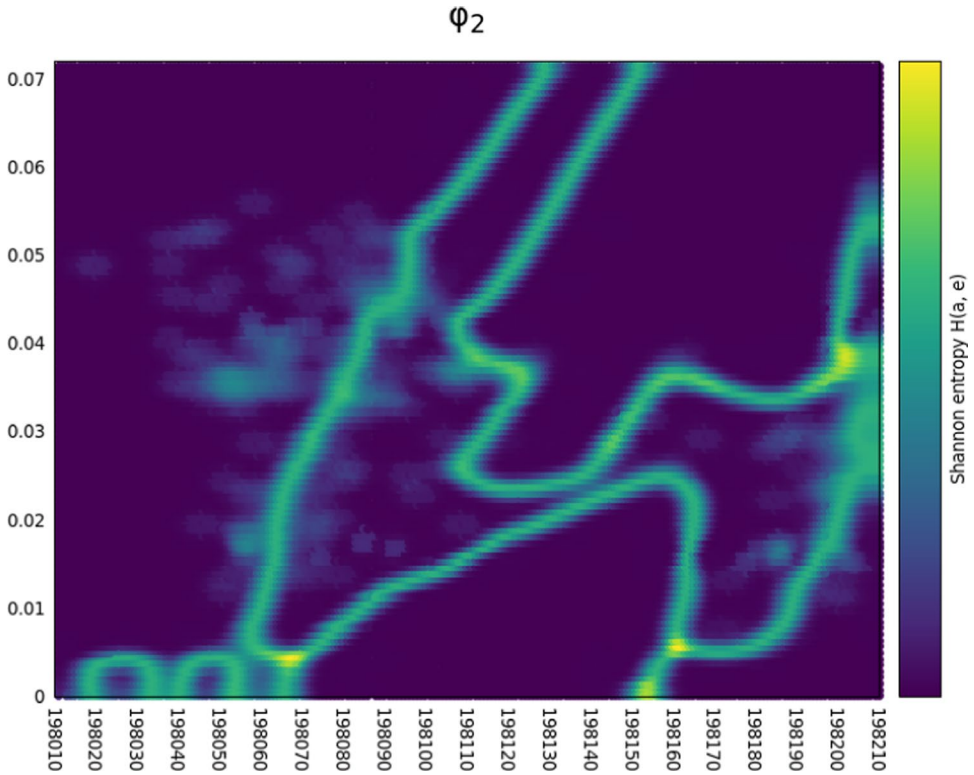

Fig. 7 Entropy map in the (a, e) plane for φ_2, computed from PCC-based soft labels using Eq. (4). The field is shown as the entropy per (a, e) bin

Table 3 Summary of ORG-D impact on the dynamical maps for φ_1 and φ_2. For each angle we report the fraction of relabelled orbits and several agreement indices between pre– and post–ORG-D cluster assignments

Angle	Relabelled	ARI	NMI	Homogeneity	Completeness	V-measure
φ_1	978 (4.39%)	0.881	0.850	0.850	0.850	0.850
φ_2	998 (4.48%)	0.875	0.850	0.849	0.850	0.850

$$\mathrm{ARI} = \frac{\mathrm{RI} - \mathbb{E}[\mathrm{RI}]}{\max(\mathrm{RI}) - \mathbb{E}[\mathrm{RI}]}, \tag{5}$$

where RI denotes the Rand Index. ARI takes values in the interval $[-1, 1]$, with $\mathrm{ARI} = 1$ indicating identical partitions and values close to zero corresponding to random agreement.

Normalized Mutual Information (NMI)

Normalized Mutual Information (NMI) quantifies the amount of shared information between two partitions using information-theoretic concepts [32]. It is defined as

$$\mathrm{NMI}(U, V) = \frac{2\, I(U, V)}{H(U) + H(V)}, \tag{6}$$

where $I(U, V)$ is the mutual information between partitions U and V, and $H(\cdot)$ denotes Shannon entropy. NMI ranges from 0 (no shared information) to 1 (perfect agreement).

In this work, ARI and NMI are used to quantify the agreement between the original K-Means clustering and the post-ORG-D partition. High values indicate that ORG-D acts as a *conservative refinement* of the initial clustering, reassigning only a limited number of trajectories while preserving the global structure of the partition.

To summarise the global impact of the PCC-based ORG-D on the dynamical maps, Table 3 reports, for both critical angles φ_1 and φ_2, the fraction of relabelled orbits together with the agreement indices between pre– and post–ORG-D cluster assignments (ARI, NMI, homogeneity, completeness and V-measure). In both cases, only about 4.4% of the 22 288 trajectories are reassigned, while the agreement scores remain high (ARI ≈ 0.88, NMI

Table 4 Confusion matrix between pre– and post–ORG-D labels for φ_1. Rows correspond to the original K-Means clusters and columns to the PCC-based ORG-D labels

	Post 0	Post 1	Post 2	Post 3
Pre 0	9315	367	3	0
Pre 1	262	5222	129	12
Pre 2	4	137	4312	0
Pre 3	1	63	0	2461

Table 5 Confusion matrix between pre– and post–ORG-D labels for φ_2. Rows correspond to the original K-Means clusters and columns to the PCC-based ORG-D labels

	Post 0	Post 1	Post 2	Post 3
Pre 0	8804	369	1	9
Pre 1	355	5248	82	12
Pre 2	0	112	4795	0
Pre 3	14	44	0	2443

and V-measure ≈ 0.85). These values confirm that ORG-D does not alter the global partition structure but rather performs targeted corrections concentrated in dynamically ambiguous regions.

The detailed redistribution of orbits between clusters is shown by the confusion matrices in Tables 4 and 5. Most entries lie on the diagonal, whereas the dominant off-diagonal terms correspond to transitions such as $0 \to 1$ and $1 \to 0$, concentrated near separatrix-like boundaries between the non-physical and chaotic domains, consistently with the high-entropy regions highlighted by the ORG-D uncertainty maps.

3.3 Evaluation metrics

Two distinct classes of evaluation metrics are employed in this study, reflecting different methodological stages of the proposed pipeline.

The first class comprises *internal clustering validity indices*, namely the silhouette score, the Davies–Bouldin (DB) index, and the Calinski–Harabasz (CH) index. These metrics assess cluster compactness and separation directly from the embedded feature space and do not rely on any external labels. They are therefore used exclusively to compare different feature-extraction, dimensionality-reduction, and clustering configurations during model selection.

The **silhouette score** [26] S measures cluster cohesion and separation. Equation 7 defines the computation of S, where $a(i)$ is the average intra-cluster distance, and $b(i)$ is the average nearest-cluster distance.

$$S = \frac{b(i) - a(i)}{\max(a(i), b(i))} \tag{7}$$

The **DB** index [27] assesses compactness and separation. It is computed as in Eq. 8, where σ_i is the intra-cluster scatter of cluster i.

$$DB = \frac{1}{K} \sum_{i=1}^{K} \max_{j \neq i} \frac{\sigma_i + \sigma_j}{\|\mu_i - \mu_j\|} \tag{8}$$

The **CH** index [28] measures the ratio of dispersion between clusters to dispersion within clusters. It is computed as in Eq. 9, where B and W are the scatter matrices between clusters and within clusters.

$$CH = \frac{\mathrm{tr}(\mathrm{B})/(K-1)}{\mathrm{tr}(\mathrm{W})/(n-K)} \tag{9}$$

The second class comprises *partition agreement measures*, which are employed only to quantify the impact of the ORG-D based on Particle Competition and Cooperation (PCC). In this case, the comparison is performed between two partitions of the same dataset: the original clustering obtained by K-Means and the refined clustering obtained after PCC-based relabelling.

These measures do not assume any ground truth physical labels. Instead, they quantify the degree of consistency between pre– and post–ORG-D assignments, allowing us to assess whether the repatriation procedure preserves the global cluster structure.

3.4 Feature ablation and DR sensitivity

To address the referee's request regarding feature sensitivity and DR ordering, we conducted controlled ablation experiments under frozen DR and clustering hyperparameters. In contrast to the optimisation procedure reported in Table 2, where hyperparameters were individually tuned for each configuration, the present analysis was performed with identical UMAP, PCA, and clustering settings across all feature combinations. This controlled protocol ensures that performance differences reflect the intrinsic contribution of the feature sets rather than secondary effects introduced by hyperparameter re-tuning.

We evaluated four representative feature configurations defined in the feature dictionary:

1. MiniRocket features only,
2. MiniRocket + FFT,
3. MiniRocket + FFT + Wavelets,
4. Full feature stack (MiniRocket + FFT + Wavelets + TSFresh).

For each case, DR parameters were fixed, and clustering was performed using K-Means with identical settings across all runs. The resulting internal validation indices are reported in Table 6 for φ_1 and φ_2.

The ablation results indicate that MiniRocket features alone already capture a substantial portion of the dynamical structure. The addition of FFT descriptors improves cluster compactness and separation, as reflected by higher Silhouette and Calinski–Harabasz indices. The inclusion of wavelet and TSFresh features produces incremental but diminishing gains. Importantly, the large-scale dynamical partition remains qualitatively stable across configurations, indicating that the clustering is not driven by a single dominant feature family.

To further assess sensitivity to DR ordering, we evaluated the reversed configuration (PCA→ UMAP). In this experiment, the output dimensionalities were correspondingly swapped to maintain comparable embedding scales (i.e., PCA was first applied with 30 components followed by UMAP with 3 components, whereas in the original configuration UMAP was applied with 30 components followed by PCA with 3 components). This adjustment preserves the final embedding dimensionality while isolating the effect of transformation order.

Although the reversed configuration yielded higher Silhouette values in certain cases, the corresponding dynamical map (Fig. 8) exhibits increased fragmentation within the chaotic transition region. In particular, the boundaries between resonant and chaotic domains become less coherent and more spatially dispersed. This behaviour indicates that improvements in geometric compactness within the embedding space do not necessarily correspond to improved dynamical consistency.

For this reason, the original UMAP→ PCA ordering was retained in the main analysis, as it preserves the large-scale dynamical structures more faithfully.

It is worth emphasising that higher internal validation indices do not automatically imply physically more meaningful partitions in Hamiltonian dynamical systems. The Silhouette score measures geometric compactness

Table 6 Controlled feature ablation under fixed DR and clustering hyperparameters for φ_1 and φ_2

Angle	Feature Configuration	Silhouette	Davies–Bouldin	Calinski–Harabasz
φ_1	MiniRocket	0.5656	0.6593	42851.35
φ_1	MiniRocket + FFT	0.5761	0.6466	43808.21
φ_1	MiniRocket + FFT + Wavelet	0.5601	0.6695	35547.92
φ_1	Full Feature Stack	0.5186	0.6979	33617.59
φ_2	MiniRocket	0.6228	0.5108	89090.91
φ_2	MiniRocket + FFT	0.6191	0.5155	89704.76
φ_2	MiniRocket + FFT + Wavelet	0.6134	0.5129	76760.33
φ_2	Full Feature Stack	0.6141	0.5062	78333.39

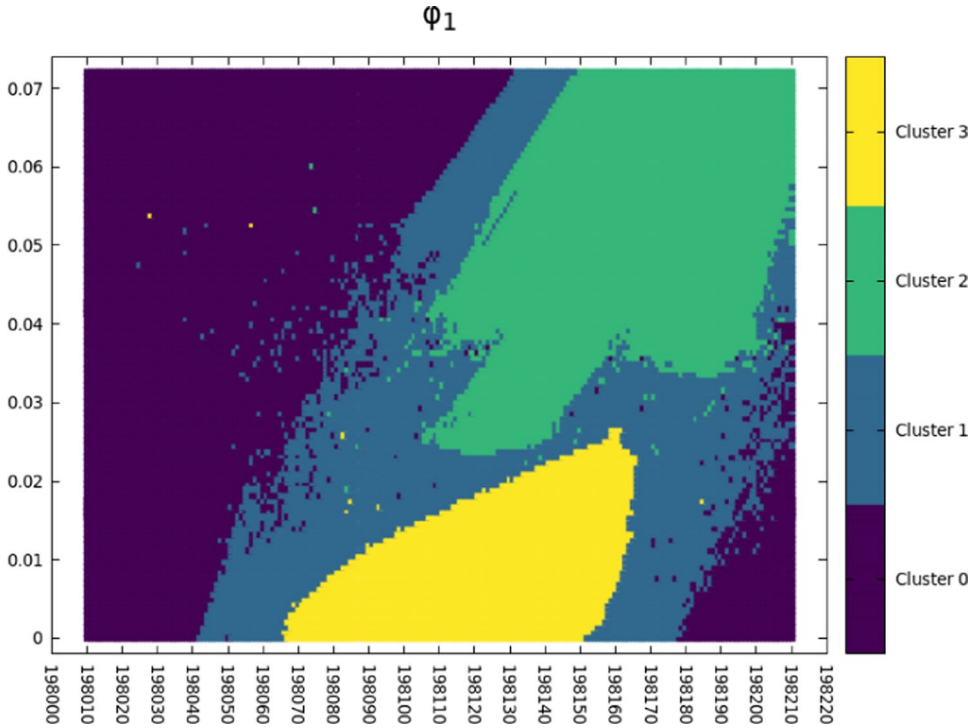

Fig. 8 Dynamical map obtained using the reversed DR ordering (PCA→ UMAP) for φ_1. Although this configuration yields slightly higher internal validation indices, the chaotic transition region exhibits increased fragmentation and irregular cluster boundaries. This illustrates that improved geometric compactness in the embedding space does not necessarily correspond to enhanced dynamical coherence

in embedding space rather than dynamical coherence. Therefore, moderate numerical improvements under alternative DR orderings should not be interpreted as superior physical segmentation.

Overall, these experiments demonstrate that the proposed framework is robust with respect to feature composition and DR ordering, and that the principal dynamical regimes identified in this work remain stable under controlled perturbations of the pipeline.

4 Fine tuning

The DR and clustering pipeline was fine-tuned using a grid search methodology. The process involved optimizing key parameters for both UMAP and PCA to maximize the silhouette score, ensuring optimal cluster cohesion and separation. This section outlines the fine-tuning procedure.

Initially, feature vectors were extracted from the dataset using various combinations of preprocessing methods. The feature combinations were systematically evaluated, with the most effective set determined based on clustering performance metrics. The explored feature combinations included FFT, wavelet, TSFresh, MiniRocket features, and their combinations. Each combination was assigned an identifier $hkey$ and processed in batches to handle the dataset's size.

The fine-tuning process began with UMAP, where the following parameters were varied:

- **n_components**: The tested values ranged from 30 to 100, by intervals of 10.
- **n_neighbours**: The tested values ranged from 30 to 100, by intervals of 10.
- **min_dist**: The tested values were 0.0003125, 0.000625, and 0.00125.

UMAP was applied to reduce the dataset's dimensionality while preserving local and global structures. The output from UMAP was then further reduced using PCA to refine the dimensionality linearly. PCA was evaluated with parameters **n_components** = 2, 3, 4, and 10.

For clustering, several methods were tested, including K-Means, Agglomerative Clustering, Gaussian Mixture Models (GMM), DBSCAN, and HDBSCAN. Each method underwent a grid search to identify the optimal

hyperparameters for the clustering algorithm. For instance, K-Means was fine-tuned with variations in the initialization method, number of clusters, and maximum iterations.

The evaluation metrics for each configuration included:

- **Silhouette Score**: A measure of cluster cohesion and separation.
- **Davies-Bouldin Index (DB)**: A metric for cluster compactness and separation.
- **Calinski-Harabasz Index (CH)**: The ratio of between-cluster dispersion to within-cluster dispersion.

These metrics were used to identify the best combination of feature vectors, UMAP parameters, PCA parameters, and clustering methods. This systematic approach ensured that the pipeline was optimised for both DR and clustering, yielding robust and meaningful clustering results for the dataset.

To address UMAP stochasticity and dimensionality-reduction stability, we fix the `random_state` parameter in all UMAP executions, ensuring reproducible embeddings under the same configuration. UMAP hyperparameters were systematically explored via grid search, including `n_neighbors` in the range $30 - -100$ and `min_dist` in $\{0.0003125, 0.000625, 0.00125\}$, while varying the embedding dimensionality `(n_components)` between 30 and 100. Since the goal of this study is to provide a deterministic and reproducible pipeline, we report results obtained under fixed-seed embeddings for the selected hyperparameters.

5 Test: The dynamics of the simple pendulum

A simple pendulum consists of a point mass m suspended by a massless string of length l from a fixed point A. Assuming that the motion is frictionless and confined to a vertical plane – and taking the pendulum's lowest position (point B) as the reference level for potential energy – the Hamiltonian formulation of the problem is given by

$$H(\theta, p) = \frac{p^2}{2ml^2} + mgl(1 - \cos\theta), \tag{10}$$

where (θ, p) is the pair of conjugate canonical variables. Here, θ is the angle between the string and the vertical line through B, and the conjugate momentum is defined as $p = ml^2\dot{\theta}$ (with $\dot{\theta}$ denoting the time derivative of θ); g represents the acceleration due to gravity.

The equations of motion are given by

$$\begin{aligned} \dot{\theta} &= \frac{\partial H}{\partial p} = \frac{p}{ml^2}, \\ \dot{p} &= -\frac{\partial H}{\partial \theta} = -mgl\sin\theta. \end{aligned} \tag{11}$$

It is well known that the analytical solution of the initial value problem (11) is cumbersome since it involves elliptic functions (e.g., [29]). For this reason, in many applications, numerical methods – such as Runge-Kutta algorithms – are often employed to obtain the trajectories (e.g., [30]). However, since the Hamiltonian has one degree of freedom, several aspects of the global dynamics of the pendulum can be inferred directly from its properties (10).

Figure 9 displays several level curves of the Hamiltonian in the representative phase space of the dynamical system, namely the plane (θ, p). In each of the three coloured regions in Fig. 9, the motion of the pendulum is characterized by a specific type of behaviour: prograde circulation (yellow), retrograde circulation (green), and oscillations about the equilibrium point (pink level curves). The circulating regimes are separated from the oscillatory motion by the separatrix, which is indicated by the red curves[1]

Applying the clustering pipeline shown in the main text, we have the following dynamical map shown in Fig. 10, where the number of clusters, $K = 3$, was intentionally chosen to match the number of equilibrium states depicted in Fig. 9.

[1] The motion of the pendulum starting from initial conditions corresponding to the Hamiltonian value at the separatrix ($H = 9.799804688$ in Fig. 9) tends asymptotically to the unstable equilibrium points located at $n\pi$, where $n \in \mathbb{Z}$.

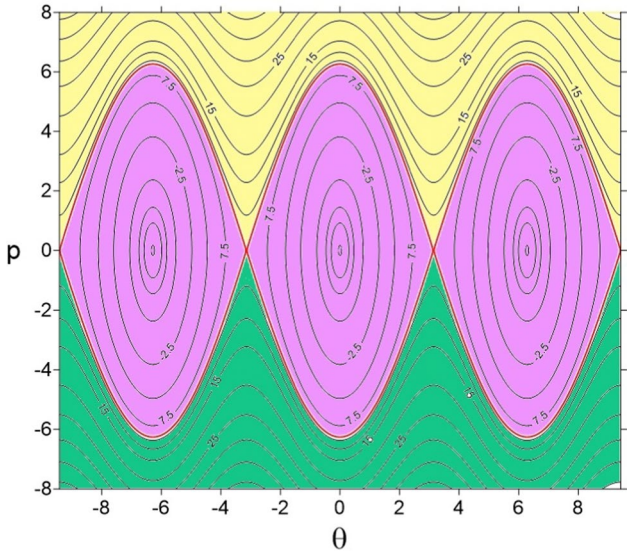

Fig. 9 Level curves of the Hamiltonian (10). Pink, green, and yellow indicate areas of the phase space corresponding to distinct regimes of motion: oscillation (pink), prograde circulation (yellow), and retrograde circulation (green). The red curves indicate the separatrix between the oscillatory and circulating regimes. Here, $m = l = 1$ and $g = 9.8$ (arbitrary units)

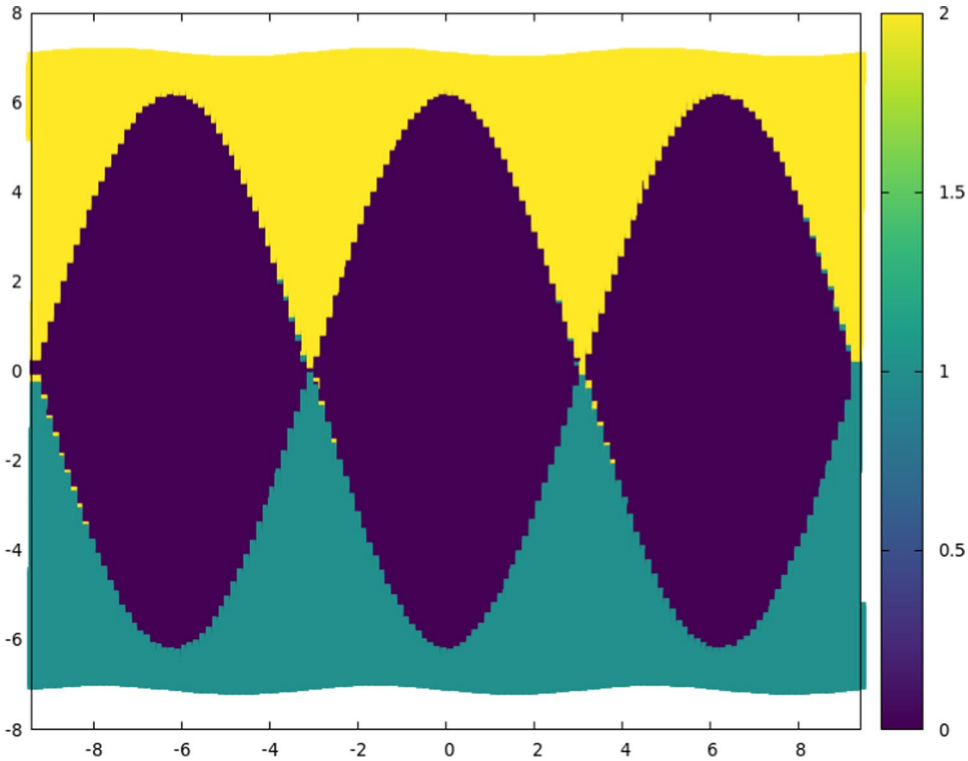

Fig. 10 The resulting K-Means clustering, K = 3, for the simple pendulum time series, which is numerically computed in Fig. 9

6 Application: cluster analysis of simulations of clones of natural Saturn satellites

This section presents the application of the methods described above. As noted in Sect. 3, the dataset analysed here was generated by numerical simulations of clones of natural satellites, incorporating the gravitational effects of Saturn's zonal harmonics and major satellites. It comprises two variables – the corotation and Lindblad resonant angles[2], which are representative variables whose time variations provide informative descriptors of the dynamics of the particle ensemble.

[2] They are defined in terms of elliptic orbital elements as $\varphi_1 = 11\lambda_S - 10\lambda_M - \bar{\omega}_M$ and $\varphi_2 = 11\lambda_S - 10\lambda_M - \bar{\omega}_S$, where λ_S and λ_M are the mean longitudes of a test particle and the satellite Mimas, and $\bar{\omega}_M$ and $\bar{\omega}_S$ are the longitudes of

Figure 11 shows two dynamical maps resulting from the techniques proposed in this work. The top (bottom) plot results from the time series analyses of the φ_1 (φ_2) angles. The methods were able to detect at least four very well-defined regions in the phase space of the semi-major axis versus the initial eccentricity of the test particles[3]. These regions are denoted, respectively, by Corotation resonance (yellow - cluster 3), Lindblad resonance (green - cluster 2), chaotic motion (blue - cluster 1), and non-physical meaning (purple - cluster 0). In order to illustrate the nature of these dynamical states of the particles, Fig. 12 shows the time variations of φ_1 and φ_2 for four initial conditions within the domains of the main regions. The Corotation resonance is characterized by the perpetual oscillation of the angle φ_1 around zero while φ_2 circulates. The Lindblad resonance is characterized by the oscillation of the angle φ_2 around π while φ_1 circulates. In the region designated as chaotic, both angles alternate their regimes between oscillation and circulation in their respective ranges. The initial conditions far from resonance domains define areas without any physical interest since no signature of the resonances appears. In these cases, the critical angles suffer short-period time variations compared to the typical time-scale of resonant signatures.

It is noteworthy that the application of the K-Means algorithm with short time series with 400 timesteps analysed in this work was able to reproduce the results of the dynamical maps given originally in Callegari & Yokoyama's work, where huge time series have been utilised. In fact, a comparison of Fig. 11 with the mapping of the 11:10 Anthe-Mimas resonance shown in Fig. 12 in [2] gives good agreement with Fig. 11.

In terms of computational cost, to provide actionable guidance without overstating hardware-dependent benchmarks, we performed an indicative profiling run for the representative case of 22,288 $\times$ 400 time series, using the same software stack as in the main experiments. The measurements were obtained on a server-class machine running an Intel Xeon E5-2698 v4 with 80 cores and 1 TB of RAM; the pipeline is entirely CPU-based and does not rely on GPU acceleration.

In this configuration, the end-to-end pipeline required on the order of 10 minutes of wall-clock time per run, with a peak resident memory (RSS) of approximately 6.8 GB, which includes all threads (threads share the same address space). Feature extraction dominated the runtime, while UMAP and k-means contributed a smaller fraction of the total cost. As expected, absolute timings vary with CPU model, thread parallelism, and I/O; therefore, these values should be interpreted as order-of-magnitude guidance, while the relative cost across pipeline stages is robust.

7 Conclusive remarks

This work presented a fully reproducible pipeline for clustering large collections of astronomical orbital time series. We combined complementary feature extractors (MiniRocket, FFT, DWT, and TSFresh), non-linear dimensionality reduction (UMAP), and classical clustering (e.g., K-Means), and assessed the results with Silhouette, Davies–Bouldin, and Calinski–Harabasz indices. The resulting embeddings separate major dynamical regimes in a way that is consistent with domain knowledge, providing compact visual summaries that facilitate the inspection of resonant structures and transition regions.

A practical contribution of this study is a simple *Outlier Repositioning via Graph Diffusion* (ORG-D) for placing rare or out-of-distribution (OOD) time series into an *already learned* low-dimensional manifold *without* distorting its geometry. The method decouples *detection* from *placement*: (i) learn the embedding on an in-distribution set only; (ii) flag potential outliers in the original feature space using a robust distance to the nearest cluster centre (e.g., median/MAD-normalized or Mahalanobis) and a high-quantile threshold; (iii) obtain two-dimensional coordinates for flagged items by applying the frozen UMAP transform and computing their position via k-NN barycentric interpolation over inlier neighbours; and (iv) render them with an uncertainty score (e.g.,

pericentre of Mimas and of the test satellite, respectively [2].

[3] The choice of $k = 4$ is justified after inspection of results given in Figure 7 in the paper Callegari Jr. and Yokoyama [2], where the main regimes of motion of particles around the 11:10 resonance with the satellite Mimas have been mapped.

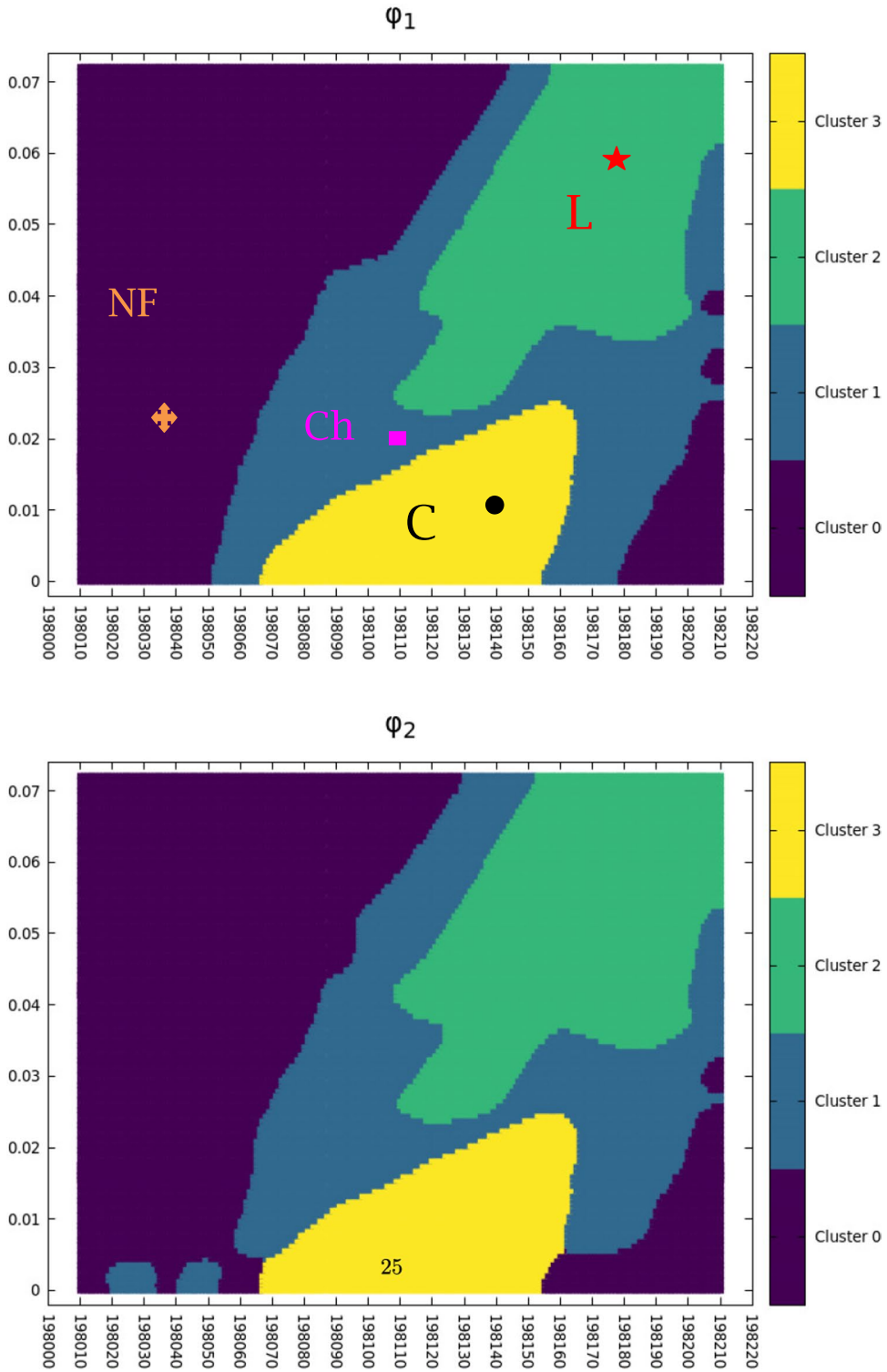

Fig. 11 Dynamical maps representing the clustering of initial orbital conditions in the space of semi-major axis versus initial eccentricity. The top and bottom panels show the mapping for φ_1 and φ_2, respectively. The colour-coded regions indicate different clusters obtained through the K-Means algorithm, capturing distinct dynamical behaviours indicated by C, L, Ch, and NP, meaning corotation resonance, Lindblad resonance, chaotic motion, and non-physical meaning, respectively. At the top panel, four coloured symbols of the initial conditions of representative temporal series of each region are shown (see Fig. 12)

inverse local density) while avoiding any re-optimisation of the embedding. In practice, this keeps the global cluster layout stable and prevents outliers from pulling dense regions apart.

In our experiments, ORG-D consistently repositioned rare orbits at the periphery of the closest cluster or along separatrix-like boundaries, making them visually salient without altering the centroids or neighbourhood relations of in-distribution data. Computationally, once features are extracted, neighbour search and interpolation

 https://doi.org/10.1007/s00521-026-12057-4

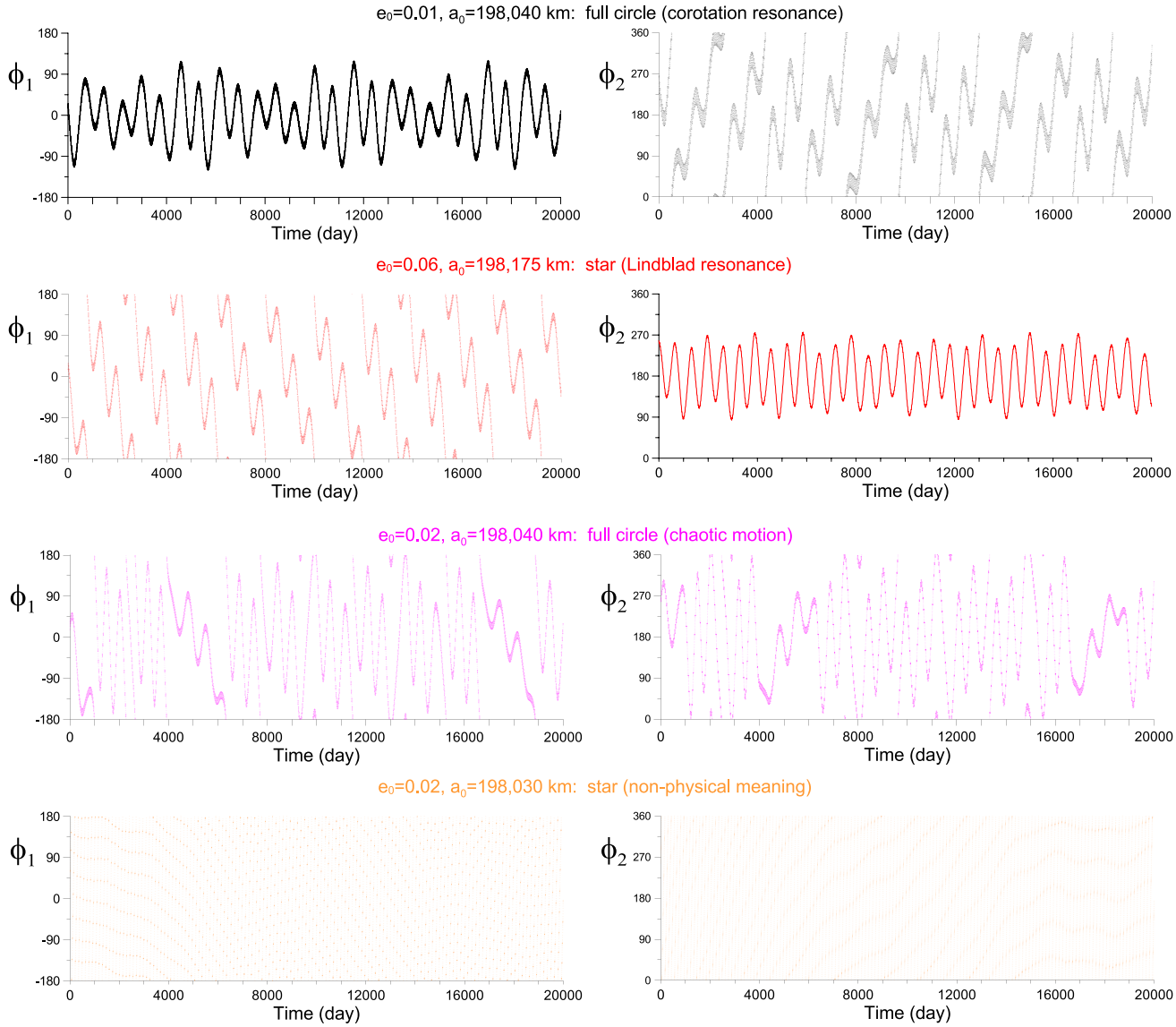

Fig. 12 Time variations of the critical angles φ_1 and φ_2 corresponding to the initial conditions indicated the symbols given in Fig. 11. From top to bottom: full circle, star, rectangle, crux. The corresponding initial values of the semi-major axis and the eccentricity of each orbit are given at the top of the panels

dominate the cost and scale sub-quadratically with dataset size when using standard approximate nearest-neighbour (ANN) indices.

The overall quality remains sensitive to hyperparameters (UMAP neighbourhood size, interpolation k, and the outlier threshold) and to the choice of base features. Dynamic Time Warping (DTW) was considered but not included due to its $\mathcal{O}(N^2T^2)$ runtime on long series and because feature-space alternatives preserved temporal information sufficiently well for clustering at this scale.

A further contribution of this study is to consolidate and extend the application of clustering techniques within celestial mechanics. Earlier work in this area demonstrated that unsupervised learning can be used to classify orbital behaviours, separate resonance zones and the chaotic layers that surround them. Such an approach provides a complementary perspective to analytical and perturbative methods by revealing families of orbits and transitional dynamics that are otherwise difficult to capture. By embedding these ideas into a broader machine learning pipeline that integrates modern feature extraction, dimensionality reduction and validation indices, the

present work strengthens the role of clustering as a rigorous tool for exploring large ensembles of resonant and chaotic dynamical systems.

Furthermore, starting from short, raw time series (400 samples) of the resonant angles, our pipeline qualitatively reproduces the phase-space maps that are traditionally obtained only after very long and computationally expensive integrations, recovering the main resonance domains and transitional regions (cf. Figure 11).

Future improvements include: (a) benchmarking ORG-D with alternative detectors (e.g., Isolation Forest) and calibration strategies for uncertainty; (b) streaming/incremental embeddings to accommodate continuously generated orbits; and (c) physics-informed features that better capture libration/rotation switching and other resonance-driven phenomena.

Incremental and streaming extensions of the proposed framework constitute a natural direction for future work. In particular, an important practical scenario is the continuous generation of new orbital trajectories, which requires integrating out-of-sample time series into an already learned low-dimensional manifold without distorting its geometry. A systematic evaluation of such a streaming setup would require dedicated experimental protocols to assess embedding stability, cluster consistency, and robustness under potential distribution shifts, and is, therefore, left for future investigation.

Appendix A Clustering algorithms

Three algorithms were considered in this study: K-Means, agglomerative clustering, and Gaussian Mixture Models (GMM) clustering. This section briefly describes each of them.

K-Means partitions data into K clusters by minimizing within-cluster variance $\mathcal{V}$ defined in Eq. A1, where μ_k is the centroid of cluster C_k [22].

$$\mathcal{V} = \sum_{k=1}^{K} \sum_{\mathrm{z} \in C_k} \|\mathrm{z} - \mu_k\|^2 \tag{A1}$$

Agglomerative clustering builds a hierarchy of clusters using linkage criteria such as the Ward's method [23], which computes the distance $d_{\mathrm{ward}}(C_i, C_j)$ between clusters C_i and C_j based on Eq. A2, where C_i and C_j are clusters, and μ_i, μ_j are their centroids.

$$d_{\mathrm{ward}}(C_i, C_j) = \frac{|C_i||C_j|}{|C_i| + |C_j|} \|\mu_i - \mu_j\|^2 \tag{A2}$$

Gaussian Mixture Models (GMM) transform data using a mixture of Gaussians, optimising the log-likelihood $\mathcal{L}$ defined in Equation A3, where π_k, μ_k, and σ_k are the weight, mean, and covariance of the k-th Gaussian, respectively [15].

$$\mathcal{L} = \sum_{i=1}^{n} \log \left(\sum_{k=1}^{K} \pi_k \mathcal{N}(\mathrm{z}_i | \mu_k, \sigma_k) \right) \tag{A3}$$

Experiments were performed using each of these clustering algorithms, and the results were assessed according to a set of evaluation metrics. These metrics were employed to allow for comparing the results between different combinations of the techniques presented so far.

Acknowledgements NCJ thanks the São Paulo Research Foundation (FAPESP) for funding projects 2020/06807-7 and 2025/02325-1.

Author contributions Eraldo Pereira Marinho: conceptualisation, methodology, software, formal analysis, visualisation, writing – original draft. Nelson Callegari Junior: investigation (celestial mechanics), validation, supervision, writing – review & editing. Fabricio Aparecido Breve: methodology (machine learning), validation, supervision, writing – review & editing. Caetano Mazzoni Ranieri: simulations, data curation, visualisation, writing – review & editing.

Funding The Article Processing Charge (APC) for the publication of this research was funded by the Coordenação de

Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) (ROR identifier: 00x0ma614). São Paulo Research Foundation (FAPESP), grant numbers 2020/06807-7 and 2025/02325-1 (to N. Callegari Junior).

Data availability The datasets and full source code required to reproduce the experiments are publicly available at https://github.com/epmarinho/ts-saturn-orbits.

Declarations

Conflict of interest The authors declare that they have no competing interests.

References

1. Callegari N Jr, Yokoyama T (2010) Numerical exploration of resonant dynamics in the system of Saturnian inner satellites. Planet Space Sci 58:1906–1921
2. Callegari N Jr, Yokoyama T (2020) Dynamics of the 11:10 corotation and Lindblad resonances with Mimas, and application to Anthe. Icarus 348:113820
3. Callegari N Jr, Rodríguez A, Ceccatto DT (2021) The current orbit of Methone (S/2004 S 1). Celest Mech Dyn Astron 133:49
4. Callegari N Jr, Rodríguez A (2023) The orbit of Aegaeon and the 7:6 Mimas-Aegaeon resonance. Celest Mech Dyn Astron 133:49
5. Dempster A, Schmidt DF, Webb GI (2021) Minirocket: A very fast (almost) deterministic transform for time series classification. In: Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining. KDD '21, (pp 248–257). Association for Computing Machinery, New York, NY, USA . https://doi.org/10.1145/3447548.3467231
6. Jorge M, Rubén C (2024) Time series clustering with random convolutional kernels. Data Min Knowl Discov 38:1862–1888. https://doi.org/10.1007/s10618-024-01018-x
7. Bai S, Kolter JZ, Koltun V (2018) An Empirical Evaluation of Generic Convolutional and Recurrent Networks for Sequence Modeling. arXiv preprint. arXiv:1803.01271
8. Christ M, Braun N, Neuffer J, Kempa-Liehr AW (2018) Time Series FeatuRe Extraction on Basis of Scalable Hypothesis Tests (TSFresh)–A Python Package. Neurocomputing 307:72–77
9. Jolliffe IT, Cadima J (2016) Principal component analysis: a review and recent developments. Philos Trans R Soc Lond A Math Phys Eng Sci 374(2065):20150202
10. Maaten L, Hinton G (2008) Visualizing data using t-SNE. J Mach Learn Res 9(Nov):2579–2605
11. McInnes L, Healy J, Melville J (2018) UMAP: Uniform Manifold Approximation and Projection for Dimension Reduction. arXiv preprint. arXiv:1802.03426
12. Hinton GE, Salakhutdinov RR (2006) Reducing the dimensionality of data with neural networks. Science 313(5786):504–507
13. Rao KD, Swamy MNS (2018) Digital Signal Processing: Theory and Practice. Springer, Cham
14. Dempster A, Petitjean F, Webb GI (2020) Rocket: exceptionally fast and accurate time series classification using random convolutional kernels. Data Min Knowl Disc 34(5):1454–1495
15. Bishop CM (2006) Pattern Recognition and Machine Learning. Springer, New York. https://doi.org/10.1007/978-0-387-45528-0
16. Berndt DJ, Clifford J (1994) Using dynamic time warping to find patterns in time series. KDD workshop 10(16):359–370
17. Sakoe H, Chiba S (1978) Dynamic programming algorithm optimisation for spoken word recognition. IEEE Trans Acoust Speech Signal Process 26(1):43–49
18. Jain AK, Murty MN, Flynn PJ (1999) Data clustering: a review. ACM computing surveys (CSUR) 31(3):264–323
19. Tan P-N, Steinbach M, Kumar V (2006) Introduction to Data Mining. Pearson Education India, New Delhi
20. Shokoohi-Yekta M, Hu B, Jin H, Wang J, Keogh E (2015) A non-parametric motif discovery algorithm for time series analysis. Data Min Knowl Disc 29:838–865

21. De Maesschalck R, Jouan-Rimbaud D, Massart DL (2000) Mahalanobis distance. Chemom Intell Lab Syst 50(1):1–18
22. MacQueen J (1967) Some methods for classification and analysis of multivariate observations. In: Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability, Volume 1: Statistics, (pp 281–297). University of California Press, Berkeley, CA
23. Ward JH (1963) Hierarchical grouping to optimise an objective function. J Am Stat Assoc 58(301):236–244. https://doi.org/10.1080/01621459.1963.10500845
24. Breve F, Quiles M, Zhao L, Pedrycz W, Liu J (2012) Particle competition and cooperation in networks for semi-supervised learning. IEEE Trans Knowl Data Eng 24(09):1686–1698. https://doi.org/10.1109/TKDE.2011.119
25. Breve F, Zhao L (2013) Fuzzy community structure detection by particle competition and cooperation. Soft Comput 17(4):659–673
26. Rousseeuw PJ (1987) Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. J Comput Appl Math 20:53–65
27. Davies DL, Bouldin DW (1979) A cluster separation measure. IEEE Trans Pattern Anal Mach Intell 1(2):224–227
28. Calinski T, Harabasz J (1974) A dendrite method for cluster analysis. Commun Stat 3(1):1–27
29. Ferraz-Mello S (2007) Canonical Perturbation Theories: Degenerate Systems and Resonance. Astrophysics and Space Science Library, (vol 345). Springer, New York, NY
30. Stoer J, Bulirsch R (1993) Introduction to Numerical Analysis, 2nd edn. Springer, New York
31. Hubert L, Arabie P (1985) Comparing partitions. J Classif 2(1):193–218
32. Strehl A, Ghosh J (2002) Cluster ensembles – a knowledge reuse framework for combining multiple partitions. J Mach Learn Res 3:583–617

Authors and Affiliations

Eraldo Pereira Marinho[1] · Nelson Callegari Junior[1] · Fabricio Aparecido Breve[1] · Caetano Mazzoni Ranieri[1]

✉ Eraldo Pereira Marinho
pereira.marinho@unesp.br

[1] Institute of Geosciences and Exact Sciences, São Paulo State University (Unesp), Rio ClaroAvenida 24A, 1515, DEMAC, São Paulo 13506-900, Brazil