12th Annual Space Traffic Conference 2026
18-19 February 2026, Austin, TX, USA

STC26-20

OrbX: A Framework for Orbital Capacity Management

Sam White[(1)], **Samya Bagchi**[(2)], **and Yasir Latif**[(3)]

[(1)] *Adelaide University, Adelaide, Australia,* sam.white@student.adelaide.edu.au

[(2)] *Space Protocol, Adelaide, Australia,* samb@spaceprotocol.com

[(3)] *Space Protocol, Adelaide, Australia,* yasir@spaceprotocol.com

https://importsam.github.io/OrbX

Orbital volume is a finite resource whose mismanagement leads to escalating collision risk, reduced orbital carrying capacity, and an increased likelihood of the Kessler syndrome, threatening the long-term viability of space operations. We refer to orbital volume as the region of orbital space occupied by satellites, with Low Earth Orbit (LEO) being of primary concern due to the large and growing satellite population and operational value. Conventional congestion assessment tools rely on altitude shell counts as proxies for occupancy, but this approach is one-dimensional and discards geometric information about orbital distributions.

In this work, we propose OrbX: a framework for orbital capacity management, built around an orbital similarity metric originally developed for identifying celestial objects of common origin. The metric operates on the six-dimensional orbital space providing a geometrically faithful model of orbital distribution. OrbX offers local orbital density estimation, orbital neighborhood computation, and enables occupancy estimation with the assurance of underlying mathematical correctness. OrbX enables not only the identification of crowded regions but the recognition of clustering structures, orbital highways, and sparsely occupied zones. We present results in crowded orbital regimes for orbital clustering, identifying most and least populated orbits, and enabling optimization for launch planning.

By revealing orbital distribution, OrbX supports mission life-cycle decisions, while exposing appropriate orbital corridors for on-orbit rendezvous servicing infrastructure siting and space traffic logistic connectivity. Furthermore, this supports evidence-based policy development grounded in occupancy profiles, facilitates orbit assignment processes and mitigation guidelines. In this way, OrbX serves as an analytical tool, linking operational risk modeling, governance, regulatory oversight, and the wider space traffic management architecture.

Keywords: Non-Euclidean Spaces, Manifold-Based Clustering, Orbital Capacity

1 Introduction

The orbital space is experiencing rapid and sustained growth. Driven by large commercial constellations, reduced launch costs, and expanding civil and defense missions, there has been a sharp increase in successful satellite deployments [10]. This increase is concentrated primarily in Low Earth Orbit (LEO), where specific altitude-inclination regimes have become highly congested, elevating collision risk and complicating space situational awareness (SSA). High local object density increases the frequency of close approaches and the likelihood of catastrophic debris-generating collisions [5]. Studies have shown that collision risk is driven primarily by local spatial density [9, 7], rather than by global object counts alone [4,5]. As a result, robust estimation of orbital density is a foundational requirement for conjunction assessment, debris mitigation planning, sensor tasking, and broader space traffic management.

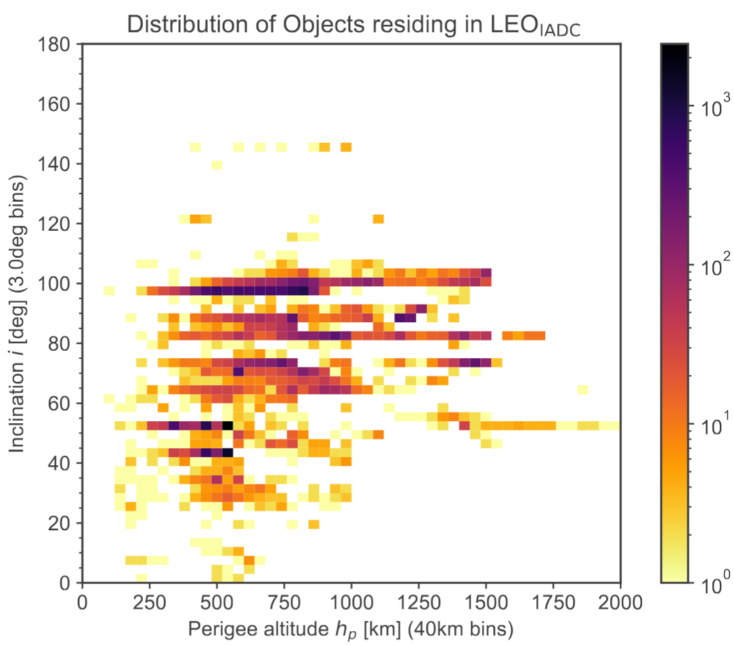

Figure 1: Distribution of objects in LEO from [2]

Traditional methods for characterizing orbital density have largely relied on altitude- and inclination-based binning of space object populations (see Fig. 1 for an example). Such approaches are widely used in debris environment models, including NASA's ORDEM [8] and ESA's MASTER [3] models, which aggregate object counts into discrete orbital shells to estimate flux and long-term debris evolution. However, these models are not designed to resolve density at the level of individual satellites or to capture localized clustering effects within orbital element space. Analytical collision risk models estimate encounter rates and conjunction probabilities using relative motion theory and probabilistic uncertainty models [13]. These methods accurately assess pairwise collision probability but do not directly characterize the surrounding orbital environment of a satellite. One of the inherent difficulties when dealing with orbits is the lack of a meaningful distance metric, that is, we are interested in formulating a distance metric that takes into account the orbit traced by the space object over time, as opposed to the instantaneous distances between them, as is done for collision assessment. Prior work has looked at this problem in the context of finding celestial objects of common origin [6]. In this work, we show the effectiveness of the distance metric when applied to nearly circular orbits (LEO) around Earth and the resulting analysis enabled. Precise modeling of the orbital space enables better policy and governance structures.

1.1 Orbital Representation and Similarity: Why is it hard?

Mathematically, a distance metric is a function that quantifies the notion of distance between elements of a set. Let $X \in \mathcal{R}^N$ be a set (for example, a set of orbital states or orbital representations). A function

$$d : X \times X \to \mathbb{R}$$

is called a distance metric if, for all $x, y, z \in X$, it satisfies the following three axioms:

1. **Non-negativity and identity of indiscernibles:**
$$d(x, y) \geq 0 \quad \text{and} \quad d(x, y) = 0 \iff x = y.$$
2. **Symmetry:**
$$d(x, y) = d(y, x).$$
3. **Triangle inequality:**
$$d(x, z) \leq d(x, y) + d(y, z).$$

These axioms ensure that the function d behaves consistently with the intuitive geometric notion of distance, enabling meaningful comparison, clustering, and neighborhood definitions in both physical space and abstract spaces such as orbital element space. First, we address existing methods of measuring distance between objects in space:

Point to point: Close approach distance computation Traditional methods for computing distances between orbits in LEO rely on representing each orbit using a set of orbital elements or state vectors and defining a metric in either physical or element space. A common approach is to propagate Two-Line Element (TLE) sets to a common epoch and compute Euclidean distances between position vectors in an Earth-centered inertial (ECI) frame, often averaged over one or more orbital periods to account for relative phasing. Closely related methods evaluate minimum distance of approach by propagating both orbits and identifying the point of closest separation in time. While useful for conjunction analysis, the point-to-point distances are unable to capture the overall orbital densities in a region of space.

Inter-orbit distance In element space, distances are frequently computed using differences in classical orbital elements—such as semi-major axis, inclination, eccentricity, right ascension of the ascending node, and argument of perigee, sometimes with empirical weighting to reflect their relative physical importance. However, this is mathematically troublesome, as all these parameters affect the orbit non-linearly and the resulting distance is a simplified linear approximation in the 6D space which does not form a metric, as defined above.

The work of [6] provides the history of orbital distance: methods for computing distances between orbits (not between where objects are in those orbits) and proposed a true metric that is applicable to circular orbits. Let us define the orbit $\mathbf{O}$ using the 6 parameter set as consisting of semi-major axis (a), pericentral distance (q), semi-latus rectum (p), eccentricity (e), inclination (i), argument of the pericentre (ω), and longitude of the ascending node (Ω). The Kholshevnikov [6] D_k between two orbits $\mathbf{o}_1$,$\mathbf{o}_2$ is given by

$$D_k(\mathbf{o}_1, \mathbf{o}_2) = (1 + e_1^2)p_1 + (1 + e_2^2)p_2 - 2\sqrt{p_1 p_2}(cosI + e_1 e_2 cosP) \quad (1)$$

where

$$\begin{aligned} cosP =& s_1 s_2 \ sin\omega_1 \ sin\omega 2 \\ &+ (cos\omega_1 \ cos\omega_2 + c_1 c_2 \ sin\omega_1 \ sin\omega_2) \ cos(\Omega_1 - \Omega_2)) \\ &+ (c_2 \ cos\omega_1 - c_1 \ sin \ \omega_1 \ cos\omega_2) \ sin(\Omega_1 - \Omega_2) \end{aligned}$$

and

$$cosI = c_1 c_2 \ cos(\Omega_1 - \Omega_2).$$

Readers are referred to [6] for proofs on the D_k being a metric and its properties. A well-defined distance metric in orbital space underpins a wide range of analytical and operational tasks. In k-nearest-neighbor (kNN) analysis, the metric enables object-centric characterization of local orbital environments by quantifying proximity between Resident Space Objects (RSOs), supporting applications such as density estimation, anomaly detection, and identification of satellites occupying sparsely populated regimes. In clustering, the same metric provides the geometric foundation for grouping objects with similar orbital characteristics, allowing constellations, debris families, and anomalous orbits to be identified in a principled and physically interpretable manner. Beyond catalog analysis, distance metrics also play a role in orbital design and planning, where candidate orbits can be evaluated relative to existing populations or mission constraints. The rest of the paper explores these applications.

2 Identifying Objects in Sparsely Populated Orbits

The search for a meaningful distance metric was initially motivated by collaboration with the Space Domain Awareness (SDA) TAP Lab[1], driven by interest in identifying RSOs occupying sparsely populated orbital regimes. From an SDA perspective, objects operating in such regions are of particular interest, as satellites seeking to minimize observation, avoid routine monitoring, or conduct non-cooperative activities may deliberately select or transition into orbits with low local object density. Identifying these objects therefore requires object-centric measures of orbital isolation rather than reliance on global population statistics. This problem can naturally be formulated as a nearest-neighbor distance problem in a space equipped with an appropriate orbital distance metric. Let

$$\mathcal{O} = \{o_1, o_2, \ldots, o_N\}$$

denote a catalog of RSOs, and let

$$D_K(o_i, o_j)$$

be a valid distance metric defined between any two objects o_i and o_j. For each object o_i, we define its k-nearest neighbors (k-NN) as the set

$$\mathcal{N}_k(o_i) = \{o_{i_1}, o_{i_2}, \ldots, o_{i_k}\}$$

such that

$$D_K(o_i, o_{i_1}) \leq D_K(o_i, o_{i_2}) \leq \cdots \leq D_K(o_i, o_{i_k}),$$

and for any object $o_j \notin \mathcal{N}_k(o_i)$,

$$D_K(o_i, o_{i_k}) \leq D_K(o_i, o_j).$$

A scalar measure of orbital isolation for o_i can then be defined as a function of the distances to its nearest neighbors, using the mean k-nearest-neighbor distance,

$$\rho_k(o_i) = \frac{1}{k} \sum_{o_j \in \mathcal{N}_k(o_i)} D_K(o_i, o_j) \tag{2}$$

Objects with unusually large values of $\rho_k(o_i)$ relative to the population are interpreted as occupying sparsely populated regions of orbital space. Ranking RSOs according to this quantity provides a principled mechanism for identifying candidates of interest for further analysis. The parameter k controls the spatial scale at which sparsity is evaluated, as orbital isolation is inherently context-dependent.

Framing the problem in this manner avoids reliance on predefined orbital bins or clusters, produces interpretable object-level scores, and aligns naturally with operational SDA objectives by highlighting RSOs whose orbital environments deviate most strongly from nominal population structure.

[1] `https://sdataplab.org`

NORAD	Name	Isolation Measure (ρ_k)
54880	EROS C3	2.68×10^3
39469	MCUBED-2	9.93×10^2
39467	ALICE	9.65×10^2

Table 1: Most isolated orbits in 300-700 km altitude shell

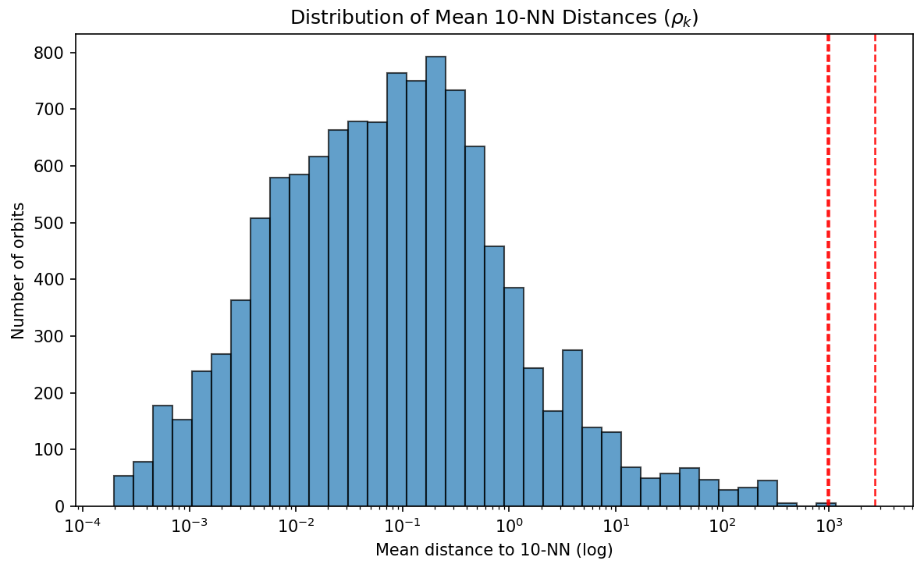

Figure 2: Histogram of the 10-NN average distance: Top 3 (most isolated) indicated with red lines. (See also Fig. 3)

Results for the k-NN based analysis with k=10 are depicted in Fig. 3 showing a histogram of the isolation measure. All the objects within the 300-700 km altitude range in the public catalog[2] were considered for the analysis. t-SNE helps visualize higher dimensional data in 2D by preserving local structure in the space. The most isolated orbits thus found are given in Table 1 and represented as red dots in the t-SNE (Appendix A) based visualization in Fig. 3.

[2] `https://www.space-track.org/`, Accessed: 01 Jan 2026.

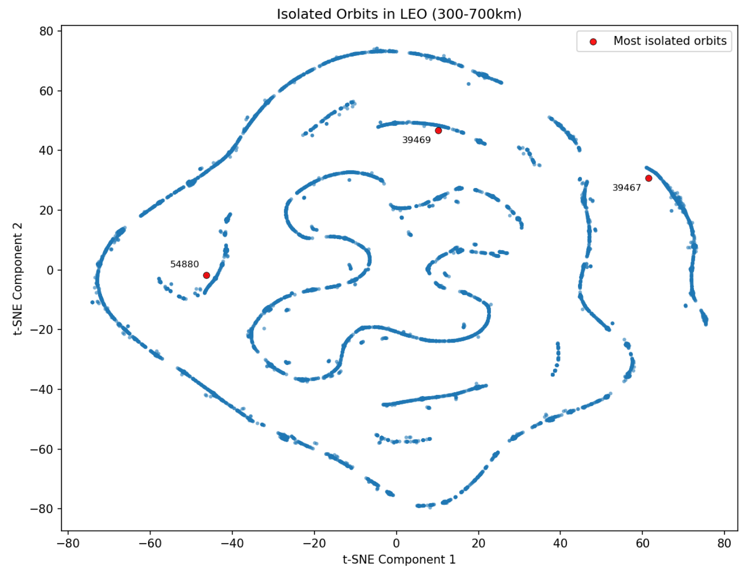

Figure 3: t-SNE visualization of isolated orbits. t-SNE enables 2D projection of the orbital space, preserving local relationships. Most isolated orbits are shown in red. Continuous lines represent orbital clusters.

3 Clustering and Density Estimation in the Orbital Space

Satellites are placed in orbits that provide high mission value such as optimizing coverage, minimizing house-keeping maneuvers, or reducing communication latency. This leads to an emergence of dense orbital neighborhoods. Discovering and characterizing these orbital "clusters" provides insights into how the orbital volume is being utilized and can help characterize the risk based on the size and local density of the cluster.

Conventional methods of characterizing orbital congestion often rely on altitude-shell based binning [2], whereby satellites are grouped into altitude bands (See Fig. 1), reducing the 6D orbit into just two dimensions. However, orbits occupy distinct spatial footprints within and outside these bins. The simplified binning-based analysis obscures these higher dynamics. With the improved distance metric in hand that allows faithful distance computation between orbits, we explore the problem of orbital density estimation using a clustering-based approach.

3.1 Unsupervised orbital clustering

Local density alone (Fig. 1) does not provide information about the structure of orbital populations. Regions of comparable density may arise from fundamentally different configurations: a single coherent orbital family, overlapping but distinct orbital regimes, or diffuse background populations. As such, density metrics are insufficient for identifying operationally meaningful orbital groupings. While density characterizes *how crowded* orbital space is, clustering characterizes *how that crowding is structured*.

The aim of unsupervised clustering is to discover orbital clusters based on orbital similarity and quantify closeness within these groups using density estimation, thereby creating an structural interpretation of orbital congestion. We make no assumptions about the relationship between different satellites and only utilize their orbital parameters for structure discovery. Let $\mathcal{O} = \{o_1, o_2, \dots, o_N\}$ denote a set of objects embedded in a space equipped with a distance metric

$$D_K : \mathcal{O} \times \mathcal{O} \to \mathbb{R}.$$

A clustering is defined as a partition

$$\mathcal{C} = \{C_1, C_2, \dots, C_M\}$$

such that

$$C_m \neq \emptyset, \quad C_m \cap C_n = \emptyset \text{ for } m \neq n, \quad \bigcup_{m=1}^{M} C_m = \mathcal{O}.$$

For each cluster C_m, a representative object (prototype) c_m is defined. The clustering problem can then be formulated as the minimization of within-cluster dispersion:

$$\min_{\{C_m\}_{m=1}^{M}, \{c_m\}_{m=1}^{M}} \sum_{m=1}^{M} \sum_{o_i \in C_m} D_K(o_i, c_m)^2.$$

When the distance metric D_K is non-Euclidean, the squared distance term may be replaced by $D_K(o_i, c_m)$ directly. If the representative is constrained to be a member of the dataset (medoid-based clustering), then

$$c_m \in C_m \subseteq \mathcal{O}.$$

We apply our analysis to the altitude shell of between 300 and 700 km (containing 11,527 satellites) as this is the most density occupied region according to Fig. 1. This region also warrants a deeper look as it has been identified as a runaway risk that cannot be adequately handled with current bureaucratic capabilities [15]. Moreover, the region exhibits spatial overlap between nearby orbital configurations and therefore meaningful clustering in this region will serve as validation for other not-so-congested regions.

3.2 Algorithm Selection

Unsupervised clustering can be performed via centroid-based or density-based methods, among others. Orbital populations do not strictly conform to predefined shapes or an expected structure. LEO is an ecosystem of constellations, fragmentation clouds, and resonant corridors, rather than compact, convex clusters of roughly equal density. In such settings, centroid-based methods are poorly matched to the geometry of the problem and often struggle to accommodate to the noise saturation without pre-processing. Density-based clustering algorithms, by contrast, define clusters as contiguous regions of high

Algorithm	Clusters	Rejected (noise)	DBCV↑	S_{Dbw} Index ↓
HDBSCAN	1816	2209	**0.6865**	**0.0031**
OPTICS	1867	2495	0.6519	0.0032
DBSCAN	345	424	0.5861	0.0129

Table 2: Results of unsupervised clustering for satellites between 300 and 700 km altitude. HDBSCAN finds the most meaningful clusters with highest DBCV score and lowest S_{Dbw} score, rejecting 2209 satellites as noise (not clustered).

point density, which enables the discovery of arbitrarily shaped clusters, and explicitly treats isolated or weakly supported objects as noise instead of forcing them into a group. We therefore focus on density based methods: DBSCAN [14], HDBSCAN [11], and OPTICS [1]. Readers are referred to Appendix B for a description of each method and the associated hyper-parameters.

Unsupervised clustering is sensitive to algorithm specific hyper-parameters. Therefore an external metric, namely, Density-Based Clustering Validation (DBCV) [12] is utilized to assess the output. DBCV evaluates how well clusters are defined in terms of local density variations and structural coherence and serves as an algorithm-agnostic objective for finding the best clustering structure. Additionally, we employ the Scatter-Density-based-within/between S_{Dbw} index [4] as a secondary cluster quality validity metric. It is particularly effective at determining the optimal number of clusters (k) by minimizing both within-cluster scattering (compactness) and inter-cluster density (separation).

3.3 Analysis

A grid search was conducted over the set of hyper-parameters for each clustering algorithm, with performance measured by the DBCV validation metric, whereby higher DBCV values indicate more informative clustering within the range [-1, 1]. The results are presented in Table. 2. HDBSCAN provides the best results based on the DBCV measure as well as the S_{Dbw} metric (lower is better), while identifying 1816 clusters. OPTICS performances similarly the quality metric scores and in cluster count, which indicates strength in hierarchical clustering methods for orbital clustering. The distribution of cluster sizes is represented in Fig. 4 showing the majority presence of a significantly small clusters (<5 objects), with an exponential decay in the number of clusters found, as the cluster size grows. Based on the trend seen in the plot, we nominate various categories: Mega (more than 50), Major (20-49), Minor (5-19) and Micro (<5).

Table 3 shows statistics for the largest 5 clusters discovered by HDBSCAN. It is interesting, albeit not surprising, that majority of the these clusters are composed of Starlink satellites. This serves as validation that HDBSCAN is correctly discovering satellite groups that should belong together.

Cluster ID	Tier	Size	Altitude (km)	Avg. Incl (°)	σ_c^2	Comment
525	Major	29	355.53 – 355.98	43.00	7.15×10^8	All Starlinks
348	Major	27	489.43 – 559.52	97.56	4.96×10^8	Mostly Starlinks
55	Major	22	547.62 – 573.16	97.60	4.16×10^8	All Starlinks
495	Major	22	319.05 – 319.86	43.00	6.92×10^8	All Starlinks
538	Major	21	547.71 – 548.95	53.21	4.03×10^8	All Starlinks

Table 3: Major clusters and corresponding altitude-inclination ranges with density estimates (HDBSCAN)

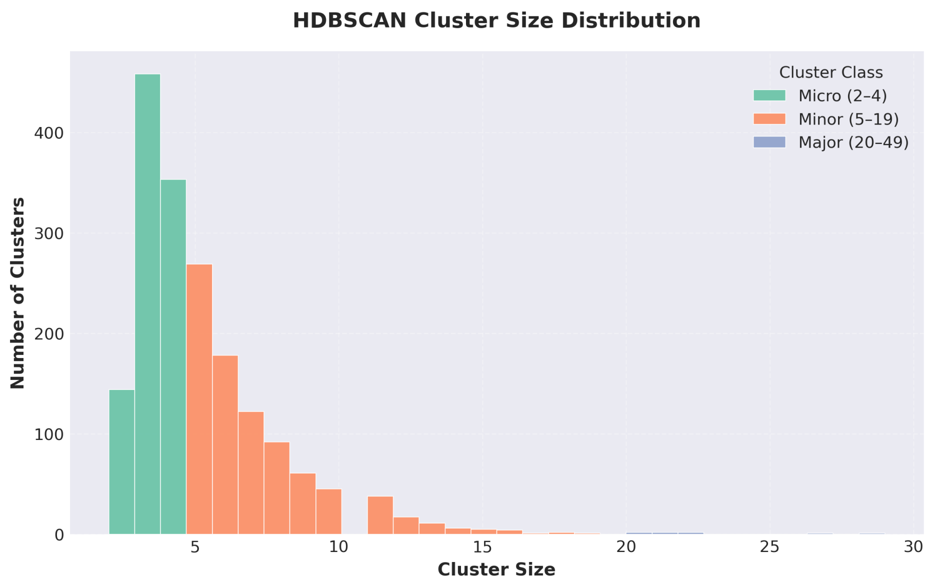

Figure 4: HDBSCAN cluster size histogram showing cluster categorization

3.4 Density estimation

Cluster size alone does not indicate within-cluster orbital density. We therefore devise a density metric based on the ‘variance’ of the cluster. We utilize the Fréchet mean, a generalization of centroids to metric spaces (See Sec. 4), to compute the “mean” orbit, from which the spread (variance) of the cluster is computed. This enables a scalar representation the density of each cluster. Given the Fréchet mean $\mathbf{o}_F$ of a cluster, we compute the spread as σ_c^2 for a cluster C of size $||C||$ using:

$$\sigma_c^2 = \frac{1}{||C|| - 1} \sum_{i=1}^{||C||} Dk(\mathbf{o}_F, \mathbf{o}_i)^2 \tag{3}$$

Fig. 5 shows the distribution of σ_C^2 for clusters of varying sizes. It can be seen that large clusters are less tightly packed but smaller cluster exhibit all range to packed-ness.

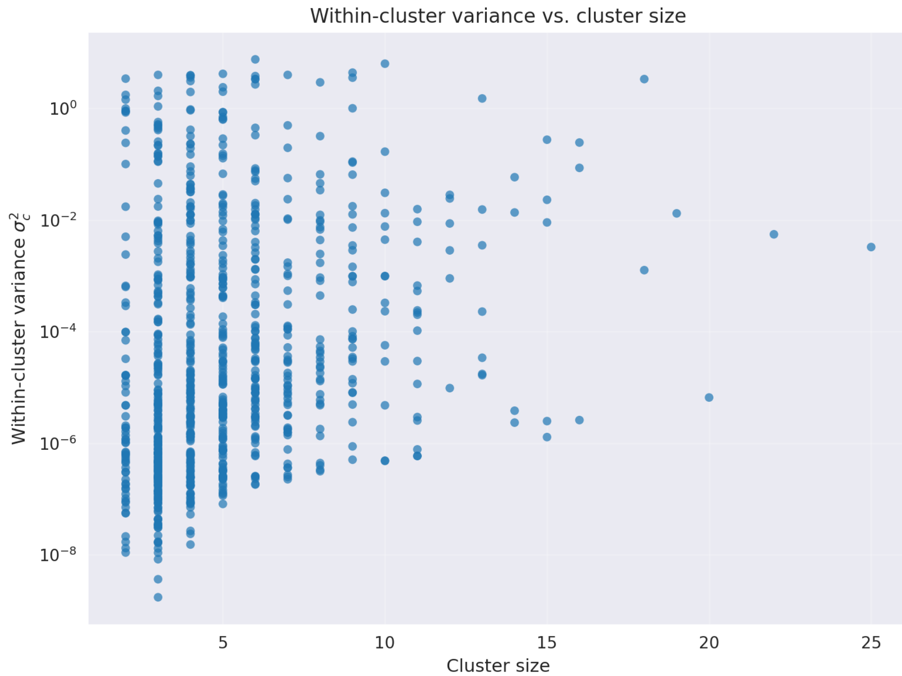

Figure 5: Cluster density for cluster sizes: Larger clusters tend to be less densely packed (lower σ_c^2 correlates to higher density).

4 Fréchet Mean Orbit

Imagine you wanted to place a refueling asset to serve known orbits. The orbit that you want to occupy should make all the orbits accessible. This is the use-case of sitting in a "mean" orbit. For orbital space, we utilize the Fréchet mean, which is a generalization of centroids to metric spaces, giving a single representative point or central tendency for a cluster of points.

Given a set of orbits of interest, $\mathcal{O} = \{o_1, o_2, \ldots, o_N\}$ we wish to determine the orbit which represents the Fréchet mean of $\mathcal{O}$. Formally, the Fréchet mean orbit $\mathbf{o}_F$ is defined as the minimizer of the average squared distance to all input orbits,

$$\mathbf{o}_{\mathcal{F}} = \arg\min_{\mathbf{o}\in\mathcal{B}} \frac{1}{N}\sum_{i=1}^{N} D_k\big(\mathbf{o}, \mathbf{o}_i\big)^2 \tag{4}$$

Intuitively, $\mathbf{o}_\mathrm{F}$ is the physically realizable orbit in $\mathcal{B}$ that best summarizes the cluster by minimizing its mean squared separation from all members.

As an optimum starting point, we first identify the medoid orbit: the orbit which on average is closest to all the other orbits in the cluster:

$$o_{\text{medoid}} = \arg\min_{\mathbf{o}_i} \frac{1}{N}\sum_{j=1}^{N} D_k\big(\mathbf{o}_i, \mathbf{o}_j\big). \tag{5}$$

The medoid serves as a strong initial reference point as it already exists near the center of the cluster. To reduce the risk of landing in a poor local minimum, we also use each orbit $\mathbf{o}_i$ as an additional starting point. From every initial orbit we then solve a bound-constrained non-linear least-squares problem,

$$\mathbf{o}_{\mathcal{F}} = \arg\min_{\mathbf{o}\in\mathcal{B}} \sum_{i=1}^{N} D_k\big(\mathbf{o}, \mathbf{o}_i\big)^2, \tag{6}$$

using a gradient-based optimizer that naively enforces the solution to remain in $\mathcal{B}$. The orbit with the lowest final objective value over all starts is adopted as the Fréchet orbit $\mathbf{o}_F$ of the cluster. To verify that this mean orbit is more central than any individual member of the cluster, we compute

$$\bar{\Phi}(\mathbf{o}_i) = \frac{1}{N}\sum_{j=1}^{N} D_k\big(\mathbf{o}_i, \mathbf{o}_j\big)^2, \qquad \bar{\Phi}(\mathbf{o}_{\mathcal{F}}) = \frac{1}{N}\sum_{j=1}^{N} D_k\big(\mathbf{o}_{\mathcal{F}}, \mathbf{o}_j\big)^2, \tag{7}$$

and verify that

$$\bar{\Phi}(\mathbf{o}_{\mathcal{F}}) \le \min_i \bar{\Phi}(\mathbf{o}_i). \tag{8}$$

To demonstrate its applicability, we computed the Fréchet for each cluster, with 99.72% of clusters producing a Fréchet mean orbit candidate which satisfies ((8)), while only 5 clusters stuck in local minima. To visualize what the Fréchet mean looks like, Fig. 6 demonstrates for a randomly selected cluster of size 15 (Starlink satellites) the Fréchet mean orbit relative to the members of cluster using a t-SNE projection.

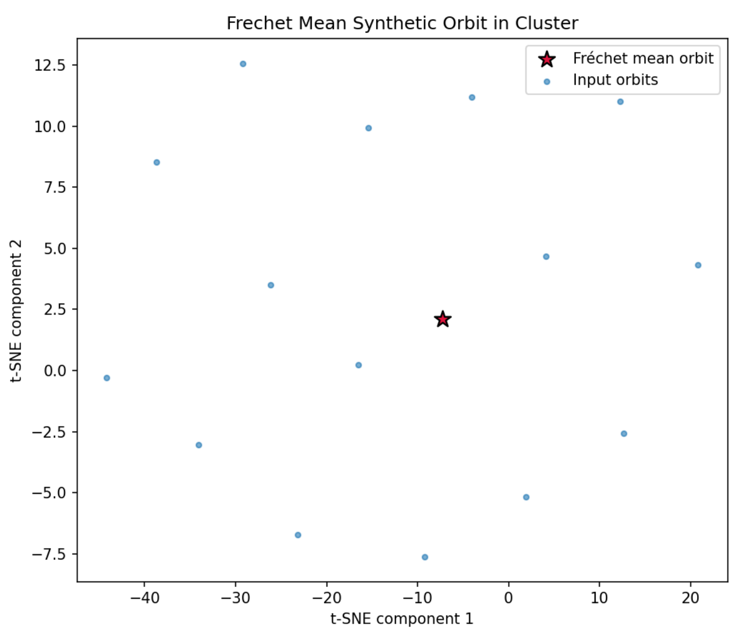

Figure 6: The Fréchet mean orbit for a cluster of size 15 using t-SNE visualization

5 Maximally Separated Orbit

While the Fréchet mean orbit provides a central reference for an existing population, space traffic management concerns often pull in the opposite direction: operators would like to place new spacecraft as far as practicable from their neighbors while remaining in the same mission-feasible region. We propose a methodology for finding the "maximally separated orbit": an orbit that occupies the largest locally empty region of orbital volume inside a given bound $\mathcal{B}$. Intuitively, this orbit sits in the "biggest gap" among all orbits by maximizing its minimum distance to any other orbit. This is useful at the planning step or as a potential maneuver target for existing satellites seeking to reduce local congestion without leaving their operational regime.

Given a set of input orbits $\mathcal{O} = \{o_1, o_2, \ldots, o_N\}$, we wish to determine the maximally geometrically separated orbit $o_{\max}$ within the feasible region $\mathcal{B}$. What we pose is, in essence, the largest empty sphere problem, whereby we seek the hyper-sphere of largest radius in the metric space, confined to a convex region which in this case is defined by $\mathcal{B}$. In this setting, the radius R is equal to the distance to the nearest neighbor. We define the separation function as

$$R(o) := \min_{1 \le i \le N} D_k(o, o_i), \quad o \in \mathcal{B}. \tag{9}$$

The maximally separated synthetic orbit is then

$$o_{\max} = \arg\max_{o \in \mathcal{B}} R(o) = \arg\max_{o \in \mathcal{B}} \min_{1 \le i \le N} D_k(o, o_i). \tag{10}$$

We approach this maximally separated orbit by solving a max–min optimization problem using a Monte Carlo search over $\mathcal{B}$, followed by a bound-constrained local optimization. We generate a set of N_s candidate orbits by sampling Keplerian element vectors

$$o_s = (a_s, e_s, i_s, \omega_s, \Omega_s, M_s) \in \mathcal{B}, \quad s = 1, \ldots, N_s, \tag{11}$$

where each element is drawn independently from a uniform distribution over its allowed range, so that all points in $\mathcal{B}$ are equally likely.

We construct an element-wise bound from the data:

$$p_{\min} = \min_i p_i, \qquad p_{\max} = \max_i p_i, \quad p \in \{a_i, e_i, i_i, \omega_i, \Omega_i, M_i\} \tag{12}$$

with

$$\mathcal{B} = [a_{\min}, a_{\max}] \times [e_{\min}, e_{\max}] \times [i_{\min}, i_{\max}] \times [\omega_{\min}, \omega_{\max}] \times [\Omega_{\min}, \Omega_{\max}] \times [0, 2\pi] \tag{13}$$

For each sample we compute its nearest-neighbor distance

$$R(o_s) = \min_{1 \le i \le N} D_k(o_s, o_i). \tag{14}$$

We retain the sample with the largest separation as the starting point for local optimization,

$$o_0 = \arg\max_{1 \le s \le N_s} R(o_s). \tag{15}$$

Starting from o_0, we refine the solution by minimizing the negative separation

$$f(o) = -R(o) = -\min_{1 \leq i \leq N} D_k(o, o_i). \tag{16}$$

We use the L-BFGS-B algorithm, which handles boundary constraints. In optimal cases, this produces the solution o_{max} with maximum separation radius:

$$o_{max} = \arg\min_{o \in \mathcal{B}} f(o). \tag{17}$$

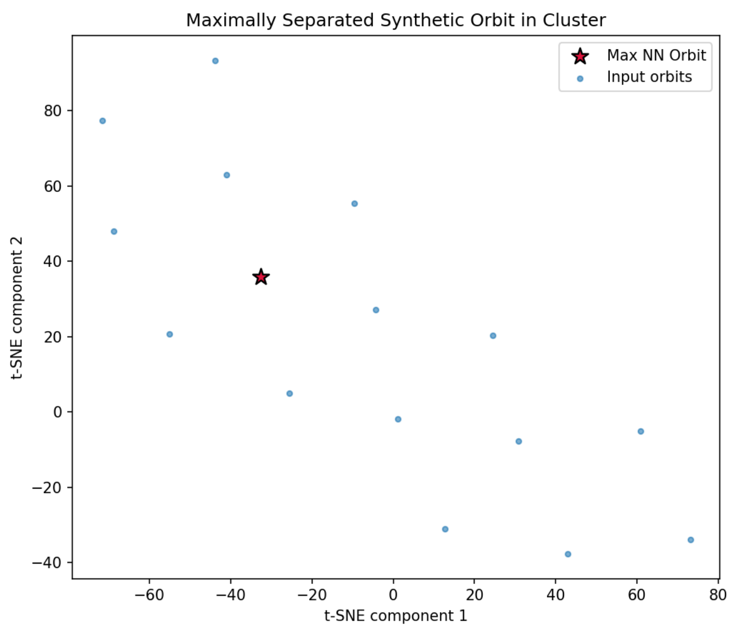

Figure 7: The maximally separated synthetic orbit for a cluster

In figure 7, o_{max} sits in the largest open available space within the cluster, which represents a viable orbit for a future launch.

6 Discussion and Policy Implications

Effective Space Traffic Management (STM) and Orbital Capacity Management (OCM) are fundamentally dependent on the transition from static regulatory regimes to dynamic, data-driven oversight. With more eyes towards the sky, managing and interpreting data through effective tools an invaluable addition to a policy makers toolkit.

The OrbX framework provides a technical bridge between raw orbital data and actionable governance by replacing oversimplified, one-dimensional altitude bins with a geometrically faithful, six-dimensional model of orbital distribution. This higher-fidelity approach enables a shift from static, reactive regulations to a dynamic system of evidence-based policy development grounded in real-time occupancy profiles. By utilizing precise clustering algorithms like HDBSCAN, policymakers can move beyond global object counts to identify specific "orbital highways", measure the "packed-ness" of mega-constellations, and recognize runaway risks in congested zones like the 300–700 km shell that exceed current bureaucratic capabilities. Furthermore, the framework introduces the "maximally separated orbit" concept, offering a proactive tool for orbital capacity management by mathematically identifying the biggest gaps for new launches or maneuver targets. For space domain awareness, the ability to quantify orbital isolation allows authorities to flag objects in sparsely populated regimes that may be attempting to evade routine monitoring. Ultimately, these tools transform abstract sustainability goals into a quantifiable architecture for orbit assignment and regulatory oversight.

References

[1] M. Ankerst, M. Breunig, H.-P. Kriegel, R. Ng, and J. Sander. Ordering points to identify the clustering structure. In *Proc. Acm Sigmod*, volume 99, 2008.

[2] European Space Agency. Esa space environment report 2025. Technical Report Issue 9, ESA Space Debris Office, Darmstadt, Germany, 2025. URL `https://www.sdo.esoc.esa.int/environment_report/Space_Environment_Report_latest.pdf`. Accessed: 2026-02-03.

[3] S. Flegel, J. Gelhaus, C. Wiedemann, P. Vorsmann, M. Oswald, S. Stabroth, H. Klinkrad, and H. Krag. The master-2009 space debris environment model. In *Fifth European conference on space debris*, volume 672, pages 1–8. Europe Space Agency/European Spanace Operations Centre Darmstadt, Germany, 2009.

[4] M. Halkidi and M. Vazirgiannis. Clustering validity assessment: Finding the optimal partitioning of a data set. In *Proceedings of the 2001 IEEE International Conference on Data Mining*, pages 187–194, 2001.

[5] D. J. Kessler, N. L. Johnson, J. Liou, and M. Matney. The kessler syndrome: implications to future space operations. *Advances in the Astronautical Sciences*, 137(8): 2010, 2010.

[6] K. V. Kholshevnikov, G. I. Kokhirova, P. B. Babadzhanov, and U. H. Khamroev. Metrics in the space of orbits and their application to searching for celestial objects of common origin. *Monthly Notices of the Royal Astronomical Society*, 462 (2):2275–2283, 08 2016. ISSN 0035-8711. doi: 10.1093/mnras/stw1712. URL `https://doi.org/10.1093/mnras/stw1712`.

[7] P. H. Krisko. Proper implementation of the 1998 nasa breakup model. *Orbital Debris Quarterly News*, 15(4):1–10, 2011.

[8] P. H. Krisko. The new nasa orbital debris engineering model ordem 3.0. In *AIAA/AAS Astrodynamics Specialist Conference*, page 4227, 2014.

[9] J.-C. Liou. An active debris removal parametric study for leo environment remediation. *Advances in space research*, 47(11):1865–1876, 2011.

[10] J. C. McDowell. The low earth orbit satellite population and impacts of the spacex starlink constellation. *The Astrophysical Journal Letters*, 892(2):L36, 2020.

[11] L. McInnes, J. Healy, S. Astels, et al. hdbscan: Hierarchical density based clustering. *J. Open Source Softw.*, 2(11):205, 2017.

[12] D. Moulavi, P. A. Jaskowiak, R. J. Campello, A. Zimek, and J. Sander. Density-based clustering validation. In *Proceedings of the 2014 SIAM international conference on data mining*, pages 839–847. SIAM, 2014.

[13] W. E. Parker, M. Harris, G. Lavezzi, and R. Linares. Defining orbital capacity through operational feasibility. In *11th Annual Space Traffic Management Conference*, 2025.

[14] E. Schubert, J. Sander, M. Ester, H. P. Kriegel, and X. Xu. Dbscan revisited, revisited: why and how you should (still) use dbscan. *ACM Transactions on Database Systems (TODS)*, 42(3):1–21, 2017.

[15] R. Stilwell. Aftershocks: Disruptive growth in low earth orbit creates new policy challenges. Technical report, CIGI Papers, 2025.

A t-Distributed Stochastic Neighbor Embedding (t-SNE)

t-Distributed Stochastic Neighbor Embedding (t-SNE) is a non-linear dimensionality reduction technique widely used for visualizing high-dimensional data in two or three dimensions. Unlike linear methods such as Principal Component Analysis (PCA), which preserve global variance structure, t-SNE is designed to preserve local neighborhood relationships, making it particularly effective for revealing clusters and substructure in complex datasets.

t-SNE operates by converting pairwise distances between points in high-dimensional space into conditional probability distributions that represent neighborhood similarity. A similar probability distribution is then constructed in the low-dimensional embedding space, and the algorithm minimizes the Kullback–Leibler divergence between the two distributions using gradient descent. This process encourages points that are close in the original space to remain close in the embedded space, while allowing distant points to be placed arbitrarily far apart. As a result, t-SNE excels at exposing local clustering behavior but does not preserve global distances or relative cluster separation.

In the context of orbital analysis, t-SNE is useful as an exploratory visualization tool for high-dimensional orbital representations, such as feature vectors derived from orbital elements, relative geometry metrics, or learned embeddings. When applied to satellite catalogs, t-SNE can reveal natural groupings corresponding to constellations, shared orbital regimes, or common mission profiles, as well as isolate objects that deviate from dominant patterns. Importantly, t-SNE is not used as a distance metric itself, nor should distances in the embedded space be interpreted quantitatively. Instead, it provides qualitative insight into structure that may inform subsequent clustering, anomaly detection, or density estimation analyses.

B Clustering algorithms

DBSCAN (Density-Based Spatial Clustering of Applications with Noise) labels points within clusters based on two fixed parameters: the ε distance and the minimum samples. A point is assigned to a cluster if the number of points within the ε distance is equal to the minimum samples. This makes it effective for discovering arbitrarily shaped, contiguous orbital neighborhoods defined by a single density scale, and for labeling isolated or sparsely supported orbits as noise instead of forcing them into a cluster.

HDBSCAN (Hierarchical Density-Based Spatial Clustering of Applications with Noise) extends DBSCAN by building a hierarchy of density-based clusters over a continuum of distance scales and then selecting clusters that are stable across this hierarchy. It can then recover clusters of varying density without requiring a single global ε, and it naturally prefers groups that persist as the density threshold is varied, which tends to suppress spurious, low-support clusters. For orbital data, this is particularly useful because bright commercial constellations, smaller specialized constellations, and diffuse debris families coexist in the same phase space at very different local densities

OPTICS (Ordering Points To Identify the Clustering Structure) constructs an ordering of the data together with a reachability distance profile that encodes cluster structure across a wide range of neighborhood radii. By analyzing valleys in this reachability plot, one can extract clusters at multiple density levels without committing to a single ε while still treating outliers as noise.

B.1 Hyperparameter Selection and Validation

Each clustering algorithm has a set of input parameters which define clustering selection criteria. In the classification of desirable cluster characterization, we do not attempt to learn a universal noise fraction, because we cannot preemptively predict the nature of data and noisiness. Instead, we declare as admissible only those solutions that identify more than a trivial number of clusters and leave no more than an a minimum of 50% share of the population clustered. Among those, we select the hyper-parameters that maximize the density-based validation index DBCV.

For DBSCAN, this involved coarse-to-fine sweeps across the distance threshold parameter ε and the minimum samples, while for HDBSCAN, the minimum cluster size was set to 2 and the minimum samples were explored to assess clustering stability and cluster significance. For OPTICS, a grid was evaluated over the xi parameter without a limit on the ε value, which controls cluster sensitivity in hierarchical extraction. In each case, DBCV was employed as the primary quantitative measure of clustering quality.