MRSOP: A MULTI-REGIME SATELLITE ORBIT PREDICTION MODEL BASED ON TENSOR DECOMPOSITION

Anonymous authors
Paper under double-blind review

ABSTRACT

Accurate satellite orbit prediction is critical for collision avoidance and sustainable space operations. Existing learning-based methods, however, are typically trained on few satellites or a single orbital regime. We construct a Two-Line Element (TLE) dataset of 18,004 satellites across LEO, MEO, NSO, and GEO, and propose **MrSOP**: a single multi-regime forecaster with physics-anchored routing and a Decomposed Mixture of Experts (DMoE). It recovers semi-major axis and eccentricity from the input, and needs no separate learned gating network or orbital-regime label at inference. MrSOP attains the lowest error in all eight in-domain regime–metric comparisons, reducing MAE by up to **60.2%** over the second-best model. It achieves the lowest error among learned models in all six zero-shot source–metric comparisons. A 16-expert DMoE cuts parameters by 94.2% relative to the full-rank model, with a gain in mean in-domain accuracy.

1 INTRODUCTION

Earth orbit is becoming increasingly crowded. As of September 2026, Starlink alone has more than 11,000 satellites in orbit (Reuters, 2026). Conjunction assessment and collision avoidance depend on accurately forecasting the future positions of orbiting objects (Boley & Byers, 2021; Reiland et al., 2021; Choumos et al., 2024). Collisions generate additional fragments that can further increase the probability of subsequent collisions, leading to the cascading process described by the Kessler syndrome (Kessler & Cour-Palais, 1978). Satellite orbit prediction has traditionally relied on analytical and numerical propagation. The Simplified General Perturbations 4 (SGP4) model remains one of the most widely used physics-based propagators for Two-Line Element (TLE) sets (Hoots & Roehrich, 1980; Vallado et al., 2006). Recent work explores differentiable propagation, learned corrections, and compensation for propagation errors (Acciarini et al., 2025; Peng & Bai, 2018; Curzi et al., 2022; Caldas & Soares, 2024), but these methods are generally evaluated on an individual spacecraft, constellation, or orbital family (Haidar-Ahmad et al., 2022; Hassan & Hassan, 2025; Zhang et al., 2025). Low Earth Orbit (LEO), Medium Earth Orbit (MEO), Navigation Satellite Orbit (NSO), and Geostationary Earth Orbit (GEO) differ in altitude, orbital geometry, and the shape of their Cartesian trajectories (Wertz & Larson, 2005).

Training one forecaster across these regimes has remained largely unattempted, and maintaining a separate forecaster for every spacecraft or regime does not scale. A multi-regime model cannot be trained directly on the raw TLE archive. The records are irregularly sampled, with sampling density and gaps that vary across objects, and tracking or sensor coverage can leave orbital elements unavailable or less timely (Space-Track.org, 2026). Without choosing which objects and mission phases to retain and where to split discontinuous tracks, a larger archive only multiplies noisy, heterogeneous series. We therefore reconstruct uniformly sampled operational trajectories from TLEs across LEO, MEO, NSO, and GEO (Space-Track.org, 2026; European Space Agency, 2026), remove non-operational phases, and apply a model-independent regularity filter. We assess reconstruction quality across all four regimes using external ephemerides, including Starlink for LEO, GNSS precise-orbit products for MEO/NSO, and GFZ precise ephemerides for BeiDou satellites in GEO.

A cleaned multi-regime dataset still leaves a learning problem. The same architecture must represent orbital patterns that do not share a common time scale or shape. Transformer-based models (Zhou et al., 2021; Nie et al., 2023), linear and multiscale architectures (Zeng et al., 2023; Wang et al., 2024a), frequency-domain models (Zhou et al., 2022), and specialized forecasting architec-

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

tures (Murad et al., 2025; Hu et al., 2025) have advanced long-horizon forecasting. Time-series foundation models target transfer to previously unseen time series and datasets (Woo et al., 2024; Das et al., 2023; Ansari et al., 2024; 2025; Shi et al., 2025; Auer et al., 2026; Liu et al., 2025). These architectures treat an orbit as a generic sequence. They do not represent orbital elements or the distinct time scales and shapes of the regimes.

We propose **MrSOP**[1], a single multi-regime satellite orbit prediction model that combines physics-anchored routing with a Decomposed Mixture of Experts (DMoE) (Shazeer et al., 2017; Fedus et al., 2022; Oseledets, 2011). The main contributions of this work are summarized as follows:

- **Large-scale multi-regime orbit dataset.** We systematically collect and preprocess historical TLE orbits for 18,004 satellites across LEO, MEO, NSO, and GEO, and report the main zero-shot evaluation on three ephemeris sources.
- **Physics-anchored DMoE.** MrSOP assigns experts from the input orbit using physically initialized anchors refined during training, without a separate learned gating network or an orbital-regime label at inference. DMoE reduces the 16-expert model from 33.85M to 1.96M parameters (94.2%, $17.23\times$) while slightly improving mean in-domain error.
- **In-domain and zero-shot forecasting.** We compare MrSOP with 15 baselines: seven pretrained time-series foundation models, seven supervised forecasters, and SGP4. It attains the lowest error in all eight in-domain regime–metric comparisons and reduces MAE by up to 60.2% over the second-best model. It attains the lowest error among learned models in all six zero-shot source–metric comparisons.

2 Dataset

2.1 In-domain TLE Dataset

Figure 1 (top) summarizes our dataset construction pipeline. We begin by collecting historical TLE sets from the Space-Track annual snapshot archives (Space-Track.org, 2026). We retain active payloads and exclude rocket bodies and debris. Orbital regimes follow the ESA DISCOS classification (European Space Agency, 2026). The nominal collection period is 2019–2025 for LEO and 2004–2025 for MEO, NSO, and GEO. We deduplicate TLEs with the same NORAD identifier and epoch, then order the remaining records chronologically for each satellite.

The reconstructed dataset after operational slicing contains 18,004 satellites (Table 1). Rows are 1-minute Cartesian states. LEO dominates the row count, with 24.7 billion states. A TLE preceding 1 January 2019 initializes the first continuous segment, and the final TLEs from 31 December 2025 can be propagated for at most 72 hours. This produces the 31 December 2018 – 3 January 2026 interval reported in the table. MEO, NSO, and GEO follow the 2004–2025 collection window.

Table 1: Trajectory statistics for the four regimes after reconstruction and operational slicing.

Regime	Satellites	Rows	Temporal coverage (UTC)
LEO	16,678	24,672,720,728	2018-12-31 – 2026-01-03
MEO	73	372,004,319	2004-01-01 – 2025-12-31
NSO	294	2,102,897,102	2004-01-01 – 2025-12-31
GEO	959	6,348,439,186	2004-01-01 – 2025-12-31

Raw TLE epochs are irregular. We therefore reconstruct each satellite as a uniformly sampled Cartesian state sequence $\mathbf{s}_t = [x_t, y_t, z_t, v_{x,t}, v_{y,t}, v_{z,t}]^\top$, with position in kilometers and velocity in kilometers per second. We assign consecutive TLEs separated by more than 72 hours to different trajectory parts. Within each part, we generate states at 1-minute intervals. We use an Orekit-based propagation pipeline for LEO and Simplified Deep-space Perturbation 4 (SDP4) for the other regimes (CS GROUP and Orekit contributors, 2026; Vallado et al., 2006). We discard TLEs that cause numerical failures and remove objects represented by a single TLE and objects yielding less than 30 days of cumulative reconstructed data.

We remove non-operational phases before applying the regularity filter. For LEO, we exclude the initial orbit-raising interval and the terminal decay or deorbiting interval. We also remove states below 300 km altitude because trajectories near re-entry are increasingly dominated by atmospheric

[1]The code is available here: `https://anonymous.4open.science/r/MrSOP-MrSOP-A-Multi-Regime-Satellite-Orbit-Prediction-Model-Based-on-Tensor-Decomposition-6F3E/README.md`

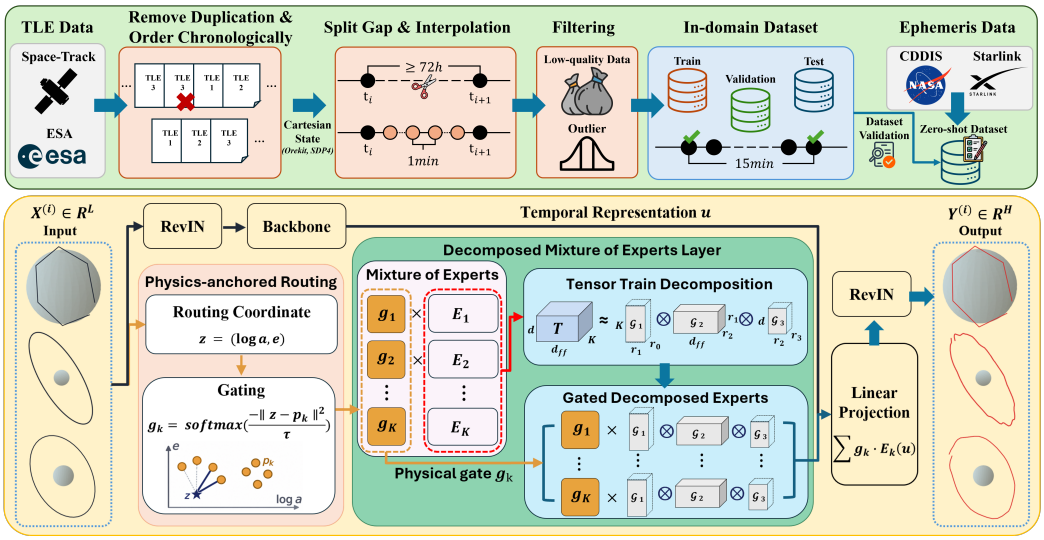

Figure 1: Dataset construction pipeline and overall architecture of MrSOP. (Top) TLEs are collected, ordered, split at gaps exceeding 72 hours, propagated onto a 1-minute Cartesian grid, filtered, and windowed into satellite-disjoint in-domain splits. External ephemerides are used only for held-out zero-shot evaluation and independent trajectory validation. (Bottom) Each input window $\mathbf{X}^{(i)}$ is normalized by RevIN and encoded by the backbone, while physics-anchored routing recovers $z = (\log a, e)$ and assigns distance-based gates g_k from the learned anchors. DMoE is gated, projected to the output window, and denormalized to $\mathbf{Y}^{(i)}$.

drag (Mahapatra, 2025). For MEO, NSO, and GEO, we derive the semi-major axis, eccentricity, and inclination from the reconstructed Cartesian states and identify the dominant stable orbital interval. This removes launch, transfer, and post-mission disposal phases. Appendix B.1 provides the detailed rolling-statistics-based slicing procedure. The resulting dataset consists of independently continuous trajectory parts, and no input or output window may cross a part boundary. Operational slicing removes large mission-phase transitions, but locally irregular trajectory parts can remain. We therefore apply an additional model-independent regularity filter based on orbital-cycle statistics of altitude, speed, eccentricity, and inclination. We convert the resulting feature scores to rank-Gaussian normal scores via an inverse-normal transformation of their empirical ranks.

We select 3σ as the smallest integer threshold satisfying the nominal four-gate Bonferroni tail criterion, $4[1 - \Phi(k)] < 0.01$, where Φ denotes the standard normal cumulative distribution function. Here, σ denotes one standard deviation in the rank-Gaussian standard-normal score space rather than the standard deviation of a raw orbital variable. The selected threshold removes 750 of 183,863 eligible trajectory parts (0.4079%) from 551 satellites. Appendix B.1 provides detailed feature construction, threshold calibration over 1–4σ, and independent diagnostics comparing the 3σ and 4σ thresholds. The regularity filter is applied before model training and does not use model outputs, output residuals, or test-time errors.

2.2 External Ephemeris Datasets

We use operational ephemerides from SpaceX Starlink (SpaceX, 2026b) and ESA/ESOC GPS orbits (ESA Navigation Support Office, 2026) for held-out zero-shot evaluation. For GEO evaluation, we use precise ephemerides from eight BeiDou geostationary satellites, which are not in the GPS set. These external ephemerides are not used for model fitting or model selection. Appendix B.3 provides the dataset statistics for these sources.

2.3 Forecasting Dataset Construction

We construct satellite-disjoint training, validation, and test splits by satellite identity. All trajectory parts from the same spacecraft remain within a single split, preventing spacecraft-level leakage

across model development and evaluation. For forecasting, we retain every 15th state of the 1-minute series, resulting in an effective sampling interval of 15 minutes. We generate windows only within retained continuous trajectory segments. For external evaluation, the ephemeris files are sampled at 60 s for Starlink, 900 s for GNSS, and 300 s for BeiDou GEO. These sequences are used without adaptation to the external ephemeris data. Section 4 specifies the input-window length, the output-window length, and model-specific preprocessing.

3 METHOD

3.1 MULTI-REGIME SATELLITE ORBIT PREDICTION (MRSOP)

Based on TimeMixer as the backbone, we propose the Multi-regime Satellite Orbit Prediction (MrSOP) model with several improvements. First, in each of the past-decomposable mixing blocks, we replace the cross-scale feed-forward layer with a mixture of K experts ($K = 16$ in the main model), while all remaining components of the backbone are left unchanged. Next, we introduce physics-anchored routing, in which two orbital elements are recovered from the input window in closed form and each expert is assigned a region of that orbital-element space through a set of anchors, so that the gate is determined by where the input window lies in physical coordinates. Furthermore, we stack the experts into a single tensor and apply Tensor-Train Decomposition (TTD) to it, which we refer to as DMoE. The overall architecture of the MrSOP model is shown in the bottom panel of Figure 1, and the algorithm of MrSOP is described in Algorithm 1.

The input is a window of L consecutive state vectors $\mathbf{X} = (\mathbf{s}_1, \ldots, \mathbf{s}_L)$, where each state vector is $\mathbf{s}_t = [x_t, y_t, z_t, v_{x,t}, v_{y,t}, v_{z,t}]^\top$. The position $\mathbf{r}_t = [x_t, y_t, z_t]^\top$ is measured in km and the velocity $\mathbf{v}_t = [v_{x,t}, v_{y,t}, v_{z,t}]^\top$ in $\mathrm{km\,s^{-1}}$, both expressed in an Earth-centered inertial frame. The output is a window of the following H state vectors, $\mathbf{Y} = (\mathbf{s}_{L+1}, \ldots, \mathbf{s}_{L+H})$, where H denotes the output length. RevIN normalizes each input channel using statistics of the input window and restores the output to physical units.

3.2 PHYSICS-ANCHORED ROUTING

3.2.1 ORBITAL-ELEMENT ROUTING COORDINATE

A state vector determines a Keplerian orbit, and two of its orbital elements are sufficient to locate that orbit within the regime structure the model has to span. From the last state vector $\mathbf{s}_L$ of the input window, the semi-major axis a is obtained from the vis-viva relation as follows:

$$a \;=\; \left(\frac{2}{\|\mathbf{r}_L\|} - \frac{\|\mathbf{v}_L\|^2}{\mu} \right)^{-1}, \tag{1}$$

where μ denotes the standard gravitational parameter of the Earth. The eccentricity e is then obtained from the norm of the eccentricity vector $\mathbf{e}$ as follows:

$$\mathbf{e} \;=\; \frac{1}{\mu} \left[\left(\|\mathbf{v}_L\|^2 - \frac{\mu}{\|\mathbf{r}_L\|} \right) \mathbf{r}_L - (\mathbf{r}_L \cdot \mathbf{v}_L)\, \mathbf{v}_L \right], \qquad e \;=\; \|\mathbf{e}\|. \tag{2}$$

The semi-major axis describes the size of the orbit and the eccentricity describes its shape, and the two together are what our router uses to distinguish one regime from another. The semi-major axis alone does not determine the trajectory shape. A highly eccentric satellite can have the same a as a circular one while following a different trajectory. Eccentricity distinguishes these cases. The two elements are therefore collected into the routing coordinate $\mathbf{z} = (\log a,\ e) \in \mathbb{R}^2$, which is standardized as $\tilde{\mathbf{z}} = (\mathbf{z} - \boldsymbol{\mu}_z) \oslash \boldsymbol{\sigma}_z \in \mathbb{R}^2$, where $\oslash$ denotes element-wise division and $\boldsymbol{\mu}_z, \boldsymbol{\sigma}_z \in \mathbb{R}^2$ are constants fixed once from the range that each element takes across the regimes, so that neither coordinate dominates the distances computed below. We take the logarithm of a to reduce the scale difference between low and high orbits. Here $\log a$ denotes $\log_{10}(a/\mathrm{km})$. We use $\boldsymbol{\mu}_z = (4.20, 0.02)$ and $\boldsymbol{\sigma}_z = (0.30, 0.05)$.

3.2.2 ANCHOR-BASED GATING

Routing is defined by a set of anchors $\mathcal{P} = \{\mathbf{p}_1, \ldots, \mathbf{p}_K\} \subset \mathbb{R}^2$ placed in the same coordinates as $\tilde{\mathbf{z}}$, with one anchor assigned to each expert. The dense weight q_k of the k-th expert is computed as a softmax over the squared distances from the routing coordinate of the input window to the anchors:

$$q_k(\tilde{\mathbf{z}}) \;=\; \frac{\exp\!\big(-\|\tilde{\mathbf{z}} - \mathbf{p}_k\|_2^2 \,/\, \tau\big)}{\sum_{j=1}^{K} \exp(-\|\tilde{\mathbf{z}} - \mathbf{p}_j\|_2^2 \,/\, \tau)}, \tag{3}$$

where $\tau = 0.15$ is a fixed temperature. Let $\mathcal{I}_2$ contain the indices of the two largest dense weights. The gate used by the experts is

$$g_k(\tilde{\mathbf{z}}) = \frac{q_k(\tilde{\mathbf{z}})\,\mathbf{1}[k \in \mathcal{I}_2]}{\sum_{j \in \mathcal{I}_2} q_j(\tilde{\mathbf{z}})}. \tag{4}$$

Thus each input window activates two experts, with their weights renormalized to sum to one. Given the temporal representation $\mathbf{u} \in \mathbb{R}^d$ that the backbone produces from the input window and passes into the layer, where d denotes the model dimension of the backbone, the output of the mixture is the gated combination of the experts:

$$\mathrm{MoE}(\mathbf{u}) \;=\; \sum_{k=1}^{K} g_k(\tilde{\mathbf{z}})\, \mathcal{E}_k(\mathbf{u}), \tag{5}$$

where $\mathcal{E}_k$ denotes the k-th expert, which is defined in Section 3.4. The anchors are initialized at positions that span the regime range along the semi-major axis coordinate, so that the experts are ordered by altitude from the first training step, and they are refined during training. Each layer learns $2K$ anchor coordinates, giving 64 trainable routing parameters across the two layers when $K = 16$. The orbital coordinates and distance-based form provide a physical prior, while training adapts the anchor locations to the forecasting objective.

3.3 TENSOR-TRAIN DECOMPOSITION

Similar to matrix decomposition methods such as Singular Value Decomposition (SVD) (Golub & Reinsch, 1970), Tensor Decomposition decomposes a high-dimensional tensor into smaller core tensors. Among the various types of Tensor Decomposition, we adopt the TTD (Oseledets, 2011), which is known to be efficient for model compression, helping reduce the number of parameters while maintaining, or even improving, the model's performance. Let $\mathcal{A} \in \mathbb{R}^{n_1 \times n_2 \times \cdots \times n_D}$ be a D-dimensional tensor, where n_m denotes the size of dimension m. Its Tensor-Train representation writes each entry as a product of D core slices as follows:

$$\mathcal{A}[i_1, i_2, \ldots, i_D] \;=\; \mathcal{G}_1[i_1]\, \mathcal{G}_2[i_2] \cdots \mathcal{G}_D[i_D], \tag{6}$$

where $\mathcal{G}_m \in \mathbb{R}^{r_{m-1} \times n_m \times r_m}$, so that each slice $\mathcal{G}_m[i_m]$ is an $r_{m-1} \times r_m$ matrix. Here, in order to keep the result of all matrix multiplications as a scalar, $r_0 = r_D = 1$, and $\{r_m\}_{m=0}^{D}$ refers to the TT-ranks. The decomposition replaces the $\prod_m n_m$ entries of $\mathcal{A}$ with $\sum_m r_{m-1} n_m r_m$ core parameters, so that the parameter count grows linearly in each dimension size rather than multiplicatively across dimensions. The TT-ranks control how much of the original tensor is retained, and reducing a rank constrains the dimensions on either side of it to interact only through a low-dimensional channel.

3.4 DECOMPOSED MIXTURE OF EXPERTS

Each expert is a two-layer position-wise network with an inner width d_{ff}, which is defined as follows:

$$\mathcal{E}_k(\mathbf{u}) \;=\; \mathbf{W}_2^{(k)}\, \phi\Big(\mathbf{W}_1^{(k)}\mathbf{u} + \mathbf{b}_1^{(k)}\Big) + \mathbf{b}_2^{(k)}, \tag{7}$$

where $\mathbf{W}_1^{(k)} \in \mathbb{R}^{d_{\mathrm{ff}} \times d}$, $\mathbf{W}_2^{(k)} \in \mathbb{R}^{d \times d_{\mathrm{ff}}}$, and ϕ denotes the activation of the feed-forward layer of the backbone. When these weight matrices are kept at full rank, each additional expert costs $2d\, d_{\mathrm{ff}}$

weight parameters per layer, in addition to its bias vectors. The resulting linear growth in the expert bank motivates a shared tensor representation.

To apply TTD to the mixture, we stack the weight matrices of all K experts so that the expert index becomes the first dimension of a third-order tensor $\mathcal{T}_1 \in \mathbb{R}^{K \times d_{\text{ff}} \times d}$, with $\mathcal{T}_1[k,:,:] = \mathbf{W}_1^{(k)}$. Similarly, the weight matrices of the second layer form $\mathcal{T}_2 \in \mathbb{R}^{K \times d \times d_{\text{ff}}}$. Applying Equation 6 to $\mathcal{T}_1$ with $D = 3$ and TT-ranks $\{r_m\}_{m=0}^{3}$ yields the core tensors $\mathcal{G}_1 \in \mathbb{R}^{K \times r_1}$, $\mathcal{G}_2 \in \mathbb{R}^{r_1 \times d_{\text{ff}} \times r_2}$, and $\mathcal{G}_3 \in \mathbb{R}^{r_2 \times d}$. The weight matrix of the k-th expert is then recovered as follows:

$$\mathcal{T}_1[k,i,j] \;=\; \sum_{\alpha=1}^{r_1} \sum_{\beta=1}^{r_2} \mathcal{G}_1[k,\alpha]\; \mathcal{G}_2[\alpha,i,\beta]\; \mathcal{G}_3[\beta,j], \tag{8}$$

where $i \in \{1,\dots,d_{\text{ff}}\}$ and $j \in \{1,\dots,d\}$ are the weight dimension indices, $k \in \{1,\dots,K\}$ is the expert index, and $\alpha \in \{1,\dots,r_1\}$ and $\beta \in \{1,\dots,r_2\}$ are the middle ranks of the TT-ranks $\{r_0, r_1, r_2, r_3\}$, with the boundary ranks absorbed. The same decomposition is applied to $\mathcal{T}_2$.

Since the expert index occupies the first dimension, $\mathcal{G}_2$ and $\mathcal{G}_3$ are shared by all experts, whereas $\mathcal{G}_1$ holds the coefficients that are specific to each expert. Writing $\mathbf{B}_\alpha \in \mathbb{R}^{d_{\text{ff}} \times d}$ for the transformation obtained by contracting $\mathcal{G}_2[\alpha]$ with $\mathcal{G}_3$, Equation 8 is equivalently written as follows:

$$\mathbf{W}_1^{(k)} \;=\; \sum_{\alpha=1}^{r_1} \mathcal{G}_1[k,\alpha]\, \mathbf{B}_\alpha. \tag{9}$$

Every expert is therefore a linear combination of the same r_1 basis transformations, and what distinguishes one expert from another is the r_1 coefficients it carries, together with its own bias vectors, which are kept outside the decomposition. For fixed TT-ranks, the shared cores do not grow with K. When the two weight stacks use the same first rank r_1, an additional expert adds $2r_1 + d + d_{\text{ff}}$ expert-bank parameters per layer, including both first-core slices and the bias vectors. The router adds two anchor coordinates per expert per layer. In the implementation, $r_1 = \min(K, 32)$, so changing K below 32 also changes the shared-core size.

Substituting Equation 8 into Equation 7, and the result into Equation 5, gives the complete layer that replaces each cross-scale feed-forward layer of the backbone. Each term of the sum is an expert reconstructed from the shared cores and scaled by its gate, and we refer to these terms collectively as the gated decomposed experts:

$$\mathrm{DMoE}(\mathbf{u}, \tilde{\mathbf{z}}) \;=\; \sum_{k=1}^{K} \underbrace{\frac{\exp\!\big(-\|\tilde{\mathbf{z}} - \mathbf{p}_k\|_2^2/\tau\big)}{\sum_j \exp(-\|\tilde{\mathbf{z}} - \mathbf{p}_j\|_2^2/\tau)}}_{\text{routing coordinate of the window}} \underbrace{\Big[\mathbf{W}_2^{(k)}\, \phi\big(\mathbf{W}_1^{(k)}\mathbf{u} + \mathbf{b}_1^{(k)}\big) + \mathbf{b}_2^{(k)}\Big]}_{\text{hidden representation}}, \tag{10}$$

where $\mathbf{W}_1^{(k)}$ and $\mathbf{W}_2^{(k)}$ are obtained from the core tensors as in Equation 8. In Equation 10, the gate depends only on the routing coordinate of the input window and the experts depend only on the temporal representation, and the two arguments are supplied by separate paths through the network that share no parameters.

4 Experimental Setup

We use the forecasting dataset of Section 2.3. The input window has length L=192 and the output window has length H=96. At the 15-minute in-domain stride these are a 48-hour input and a 24-hour output. Zero-shot input windows are read at a 900-second step, so the same lengths are a 48-hour input and a 24-hour output for Starlink, GNSS, and BeiDou. Consecutive zero-shot input windows start every 96 steps, equal to the output length, to limit overlap between output windows. Each channel is normalized by the mean and standard deviation of the input window, and the output is denormalized before scoring. We report *6-channel MAE*, the mean absolute error over position, velocity, and output steps, which mixes kilometers with kilometers per second. We also report *three-dimensional position MAE* (Pos3D MAE), the mean Euclidean distance on (x_t, y_t, z_t).

Table 2: Forecasting results on the in-domain test set across orbital regimes. Denormalized MAE in km. The second column shows the total parameter count of each model, and the bottom row "Improvements" presents the relative performance difference between the best and second-best models for each metric. The best result is highlighted in **bold**, and the second-best is highlighted with underline.

Model	Params	6-channel MAE				Pos3D MAE			
		LEO	MEO	NSO	GEO	LEO	MEO	NSO	GEO
SGP4	–	2239.13	268.30	33.00	537.43	8900.15	1091.81	130.78	2487.25
Chronos-2	119.5M	101.35	165.97	601.05	209.15	410.59	681.15	2367.09	924.24
TimesFM	231.3M	1688.02	1142.77	59.12	96.23	6682.67	4729.69	234.47	440.58
Time-MoE	453.2M	326.11	581.86	1373.61	2172.75	1315.00	2490.02	5431.45	9937.58
Sundial	128.3M	888.76	915.24	2251.02	708.31	3552.30	3923.93	8885.55	3160.33
Chronos-Bolt	205.3M	1742.70	2454.22	1733.85	297.47	6942.30	10153.43	6889.99	1338.17
TiRex	35.3M	634.03	663.47	2421.65	3284.94	2534.87	2702.59	9521.52	14700.97
Moirai	91.4M	1977.71	3647.28	5474.14	12421.95	7988.44	15256.07	21850.58	56443.30
TimeFilter	3.05M	55.68 ± 9.49	113.33 ± 37.26	107.19 ± 43.41	157.16 ± 97.14	225.73 ± 38.76	469.92 ± 155.76	431.92 ± 175.58	707.18 ± 438.32
MultiPatchFormer	15.39M	64.91 ± 1.22	101.74 ± 7.31	114.63 ± 11.39	223.64 ± 24.45	260.50 ± 5.05	419.83 ± 29.52	456.96 ± 42.98	1004.55 ± 109.34
WPMixer	2.94M	129.89 ± 6.37	242.66 ± 36.45	289.87 ± 27.27	139.61 ± 10.72	521.48 ± 25.64	997.46 ± 151.45	1144.08 ± 109.28	623.60 ± 50.47
Mamba	0.44M	136.08 ± 5.41	347.75 ± 11.27	209.37 ± 7.83	144.41 ± 1.67	552.69 ± 22.14	1441.41 ± 48.83	846.85 ± 31.77	641.37 ± 6.56
CoRA	356.5M	1799.12 ± 12.89	3239.85 ± 2.73	6741.35 ± 20.22	9423.82 ± 46.54	7152.50 ± 50.45	13520.41 ± 15.65	26488.68 ± 87.31	42573.14 ± 198.17
TimeXer	4.93M	116.11 ± 9.42	212.78 ± 10.42	364.42 ± 16.96	426.58 ± 56.53	468.41 ± 38.14	870.41 ± 42.67	1433.21 ± 65.05	1920.32 ± 257.48
TimeMixer	2.21M	62.80 ± 2.06	85.24 ± 6.19	48.43 ± 5.85	72.84 ± 0.97	250.24 ± 8.41	346.90 ± 25.48	194.49 ± 23.56	324.62 ± 3.66
MrSOP (ours)	1.96M	**44.55 ± 2.14**	**47.27 ± 5.89**	**28.34 ± 3.49**	**29.02 ± 3.23**	**177.69 ± 8.62**	**194.88 ± 24.02**	**113.60 ± 14.13**	**129.41 ± 14.33**
Improvements	–	20.0 ± 14.2%	44.5 ± 8.0%	14.1 ± 10.6%	60.2 ± 4.5%	21.3 ± 14.0%	43.8 ± 8.1%	13.1 ± 10.8%	60.1 ± 4.4%

We train with AdamW on normalized MSE, for at most 100 epochs with early stopping of patience 10 on validation 6-channel MAE. The learning rate is 4×10^{-4} with cosine decay, the weight decay is 10^{-5}. Architectural hyperparameters follow each original paper. MrSOP uses a TimeMixer backbone (Wang et al., 2024a) (d_{model}=512, d_{ff}=1024, two layers), with the main model fixed at $K = 16$ experts. This configuration is used across all main in-domain and zero-shot evaluations. The expert-count ablation in Appendix E separately varies K within the same MrSOP architecture. All runs use a single NVIDIA GeForce RTX 3090 GPU.

We compare 15 baselines: Chronos-2 (Ansari et al., 2025), TimesFM (Das et al., 2023), Time-MoE (Shi et al., 2025), Sundial (Liu et al., 2025), Chronos-Bolt (Ansari et al., 2024), TiRex (Auer et al., 2026), Moirai (Woo et al., 2024), TimeFilter (Hu et al., 2025), MultiPatchFormer (Naghashi et al., 2025), WPMixer (Murad et al., 2025), Mamba (Jung & Kim, 2026), CoRA (Cheng et al., 2026), TimeXer (Wang et al., 2024b), and TimeMixer (Wang et al., 2024a). SGP4 is the physics baseline (Vallado et al., 2006). For each input window we select the latest TLE whose epoch is at most the last time of the input window and evaluate it on the output timestamps, as in its operational use, rather than initializing it from the Cartesian input window. The tables report mean ± sample standard deviation over three runs. Deterministic methods have zero standard deviation, which is omitted for brevity. Pretrained foundation models use their released checkpoints, so repeated runs are identical and the standard deviation is zero. Appendix G reports fine-tuning of the four foundation models with a public training recipe, and Appendix H reports inference time under the same protocol.

5 Results

5.1 Main results

Table 2 reports the in-domain forecasting error across the four orbital regimes. MrSOP achieves the lowest error in all eight regime–metric comparisons among the 15 baselines. The seven pretrained time-series foundation models show substantial regime-dependent error. TimesFM, for example, performs considerably better on NSO and GEO than on LEO, whereas other pretrained models exhibit different failure patterns across the altitude regimes. These results indicate that generic pretraining alone does not resolve the heterogeneity of multi-regime orbital dynamics. SGP4 also exhibits strong regime dependence. It is competitive on NSO, reaching 33.00 km of 6-channel MAE and 130.78 km of Pos3D MAE, but its errors are substantially larger on the remaining regimes under the same evaluation protocol. This variation motivates reporting performance separately by orbital regime rather than relying only on an aggregate score. Against the TimeMixer backbone, MrSOP improves all eight in-domain regime–metric comparisons. The difference is particularly large on MEO and GEO: 6-channel MAE decreases from 85.24 ± 6.19 to 47.27 ± 5.89 km on MEO and from 72.84 ± 0.97 to 29.02 ± 3.23 km on GEO. These results motivate the routing analysis in Section 5.2.

Table 3 reports the results on the zero-shot ephemerides. The Starlink, GNSS, and BeiDou ephemeris datasets are excluded from training. MrSOP is applied to these datasets without fine-tuning and re-

Table 3: Forecasting results on three zero-shot ephemeris datasets. Denormalized MAE in km. Trained models report mean ± sample standard deviation over three runs. The bottom row "Improvements" presents the relative performance difference between MrSOP and the best other model for each metric. Positive values indicate lower error for MrSOP. Improvements relative to SGP4 are omitted. The best result is highlighted in **bold**, and the second-best is highlighted with underline.

Model	6-channel MAE			Pos3D MAE		
	Starlink	GNSS	BeiDou	Starlink	GNSS	BeiDou
SGP4	20009.84	107.96	**0.52**	77858.93	419.74	**2.18**
Chronos-2	77.03	271.24	173.77	306.95	1047.88	780.69
TimesFM	1661.34	75.57	107.16	6505.07	299.20	497.41
Time-MoE	258.97	442.11	2069.27	1033.02	1753.63	9526.47
Sundial	802.40	262.72	676.28	3166.95	1034.39	3038.77
Chronos-Bolt	1687.92	296.60	302.39	6607.55	1162.49	1382.28
TiRex	781.27	1954.31	3092.60	3076.72	7649.60	14201.02
Moirai	1961.69	4236.18	12398.08	7732.32	16977.23	58014.75
TimeFilter	39.00 ± 8.38	122.16 ± 45.62	113.50 ± 79.32	157.31 ± 33.27	487.95 ± 182.19	514.93 ± 358.60
MultiPatchFormer	45.59 ± 1.20	129.24 ± 8.25	190.43 ± 18.29	181.57 ± 4.09	509.73 ± 30.32	878.93 ± 83.59
WPMixer	110.56 ± 16.94	276.49 ± 15.91	103.31 ± 13.14	444.77 ± 67.70	1084.49 ± 66.08	471.90 ± 62.43
Mamba	84.56 ± 5.71	233.64 ± 12.09	91.28 ± 3.43	338.76 ± 23.39	931.52 ± 43.31	412.05 ± 12.59
CoRA	1770.54 ± 13.09	6869.17 ± 20.89	9250.31 ± 140.57	6929.32 ± 51.52	26763.55 ± 88.23	42686.59 ± 662.16
TimeXer	94.63 ± 12.54	367.01 ± 14.26	389.65 ± 49.46	378.91 ± 48.61	1435.17 ± 55.92	1790.21 ± 220.92
TimeMixer	30.18 ± 1.28	66.88 ± 6.47	47.18 ± 1.88	120.92 ± 5.31	265.66 ± 25.39	218.07 ± 8.24
MrSOP (ours)	**21.57 ± 0.91**	**45.99 ± 3.80**	33.02 ± 15.78	**85.12 ± 3.74**	**181.51 ± 14.97**	152.09 ± 72.14
Improvements	28.5 ± 4.3%	31.2 ± 8.7%	-	29.6 ± 4.4%	31.7 ± 8.6%	-

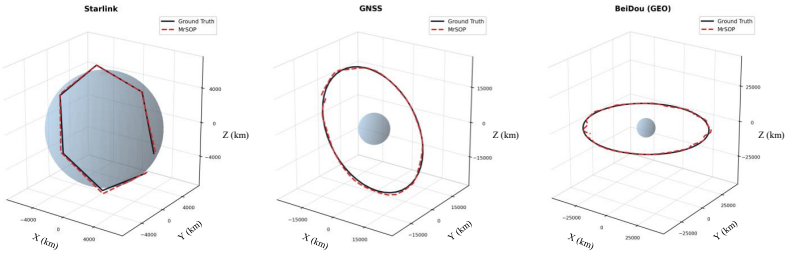

Figure 2: Zero-shot 3D output trajectories on Starlink (left), GNSS (center), and BeiDou (right). Solid curves show ground-truth trajectories, and dashed curves show the MrSOP output.

quires no orbital-regime label at inference. On BeiDou, MrSOP attains 33.02 ± 15.78 km 6-channel MAE and 152.09 ± 72.14 km Pos3D MAE, the lowest errors among the learned models in this comparison. SGP4 remains more accurate, with 0.52 km and 2.18 km, respectively. MrSOP achieves the lowest error on Starlink and GNSS for both metrics. Figure 2 shows example 3D output trajectories for Starlink (left), GNSS (center), and BeiDou (right). Solid curves show the ground-truth trajectories, and dashed curves show the corresponding MrSOP outputs.

5.2 Ablation Study

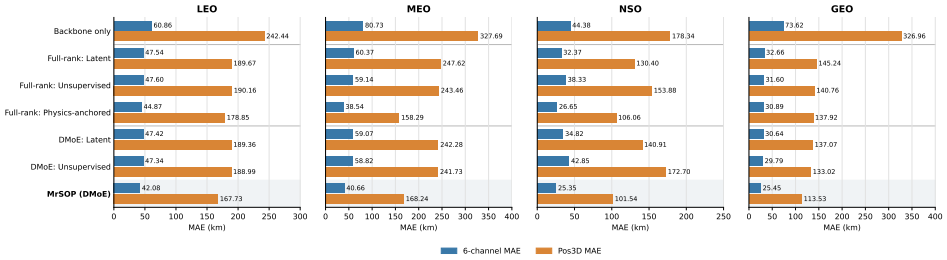

Figure 3: In-domain routing ablation on LEO, MEO, NSO, and GEO (left to right). Bars show denormalized 6-channel and Pos3D MAE in km. MrSOP has the lowest errors on LEO, NSO, and GEO, while full-rank physics-anchored routing has the lowest errors on MEO.

We compare seven variants: the TimeMixer backbone and three routing strategies with either full-rank or DMoE experts. All variants share the same backbone, and all MoE variants use $K = 16$ experts. Figures 3 and 4 report 6-channel and Pos3D MAE across the four in-domain regimes and three zero-shot sources, respectively. Detailed definitions and training settings are provided in Appendix D. In Figure 3, physics-anchored routing achieves the lowest 6-channel and Pos3D MAEs within each expert family across all four regimes. MrSOP has the lowest errors overall on LEO,

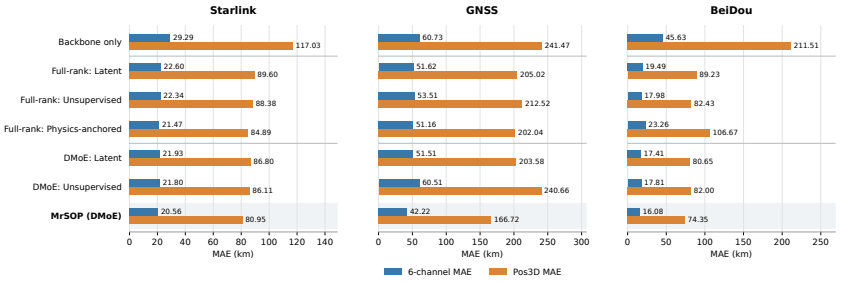

Figure 4: Zero-shot routing ablation on Starlink (left), GNSS (center), and BeiDou (right). Bars show denormalized 6-channel and Pos3D MAE in km. MrSOP achieves the lowest 6-channel and Pos3D MAE among the compared MoE variants on all three datasets.

NSO, and GEO, while the full-rank physics-anchored variant performs best on MEO. In the zero-shot comparison (Figure 4), MrSOP achieves the lowest errors on Starlink, GNSS, and BeiDou for both metrics. These comparisons support the benefit of anchor-based routing beyond supplying orbital coordinates to a learned gate.

Table 4: Parameter cost of routing and expert representations. Router params denote trainable parameters used only by the routing function, summed over the two DMoE layers.

Variant	Router params	Full-rank total	DMoE total	Reduction
Backbone	–	2,205,585		–
Latent	16,416	33,863,169	1,980,929	17.09× (94.2%)
Unsupervised	96	33,846,849	1,964,609	17.23× (94.2%)
MrSOP (ours)	64	33,846,817	**1,964,577**	17.23× **(94.2%)**

Table 4 separates the parameter cost of routing from that of the expert representation. Physics-anchored routing learns only 64 anchor coordinates across the two layers. The unsupervised orbital-coordinate router uses 96 parameters, while latent routing requires 16,416 parameters for projections from the hidden representation. In all variants, the expert bank accounts for most of the model size. For MrSOP, DMoE reduces the full-rank model from 33.85M to 1.96M parameters, a $17.23\times$ (94.2%) reduction. As a result, MrSOP has the fewest router parameters and the fewest total parameters among the compared MoE variants, and it is also smaller than the single-expert backbone (2.21M).

6 Discussion

Despite the low in-domain GEO errors in Figure 3, generalization to unseen geostationary satellites remains challenging. SGP4 and fine-tuned Moirai achieve lower mean errors there than MrSOP (Tables 3 and 14). For future work, we aim to extend the routing coordinate with quantities that vary within the geostationary regime, such as inclination, to address the difficulty of forecasting geostationary orbits. Furthermore, Table 15 shows that MrSOP requires 3.38 ms per input window, which is faster than every pretrained foundation model in our comparison. This efficiency makes MrSOP well suited to ground-based orbital monitoring and forecasting systems, where large numbers of satellite trajectories must be updated and evaluated continuously.

7 Conclusion

In this work, we propose MrSOP, a multi-regime satellite orbit prediction model that routes each input window by its orbital elements through physics-anchored routing and represents the experts with DMoE. Trained on a dataset of 18,004 satellites reconstructed from historical TLEs, MrSOP achieves the lowest error across all four in-domain orbital regimes for both metrics. It also achieves the lowest mean error among learned models on all three zero-shot datasets for both metrics, while using 94.2% fewer parameters than the full-rank model. We believe our work demonstrates that physically structured routing enables a single compact model to forecast across heterogeneous orbital regimes.

AI USE STATEMENT

We used GPT-6 Astra and Grammarly for text rephrasing and grammar checks, and GPT-6 Astra and Claude Opus 5 for assistance with code debugging. All research ideas and manuscript content were originally developed and written by the authors. The authors reviewed, revised, and verified all AI-assisted outputs and take full responsibility for the final content of this work, including its text, code, and claims.

ETHICS STATEMENT

This work does not involve human subjects or personally identifiable information. The in-domain dataset is reconstructed from publicly available Two-Line Element sets. Held-out evaluation uses operational ephemerides from SpaceX Starlink and ESA/ESOC GPS orbits, together with precise ephemerides of eight BeiDou geostationary satellites; these external ephemerides are not used for model fitting or model selection. The study is intended to support satellite orbit prediction for collision avoidance and sustainable space operations. Its forecasts are research outputs and are not a substitute for operational conjunction assessment.

REPRODUCIBILITY STATEMENT

The source code is provided through the anonymous repository linked in Section 1. It includes the scripts for training, inference, in-domain evaluation, and zero-shot evaluation, together with the configuration files that specify the model and training hyperparameters. The routing mechanism is described in Section 3.2, and the forward computation of MrSOP is summarized in Algorithm 1. The reconstruction and windowing of the in-domain dataset are described in Sections 2.1 and 2.3. Appendix B details the operational slicing and the calibration of the regularity-filter threshold. The sources, processing, and coverage of the external ephemerides used for zero-shot evaluation are given in Appendix B.3. The evaluation protocol and metrics are defined in Section 4. The settings of the routing and expert-count ablations are given in Appendices D and E. The SGP4 evaluation procedure is described in Appendix F, the fine-tuning of the foundation models in Appendix G, and the inference-time measurement setup in Appendix H.

REFERENCES

Giacomo Acciarini, Atılım Güneş Baydin, and Dario Izzo. Closing the gap between SGP4 and high-precision propagation via differentiable programming. *Acta Astronautica*, 226:694–701, 2025. doi: 10.1016/j.actaastro.2024.10.063.

Abdul Fatir Ansari, Lorenzo Stella, Ali Caner Turkmen, Xiyuan Zhang, Pedro Mercado, Huibin Shen, Oleksandr Shchur, Syama Sundar Rangapuram, Sebastian Pineda Arango, Shubham Kapoor, Jasper Zschiegner, Danielle C. Maddix, Hao Wang, Michael W. Mahoney, Kari Torkkola, Andrew Gordon Wilson, Michael Bohlke-Schneider, and Bernie Wang. Chronos: Learning the language of time series. *Transactions on Machine Learning Research*, 2024. ISSN 2835-8856. URL https://openreview.net/forum?id=gerNCVqqtR. Expert Certification.

Abdul Fatir Ansari, Oleksandr Shchur, Jaris Küken, Andreas Auer, Boran Han, Pedro Mercado, Syama Sundar Rangapuram, Huibin Shen, Lorenzo Stella, Xiyuan Zhang, et al. Chronos-2: From univariate to universal forecasting. *arXiv preprint arXiv:2510.15821*, 2025.

Andreas Auer, Patrick Podest, Daniel Klotz, Sebastian Böck, Günter Klambauer, and Sepp Hochreiter. TiRex: Zero-shot forecasting across long and short horizons with enhanced in-context learning. *Advances in Neural Information Processing Systems*, 38:57529–57580, 2026.

Aaron C. Boley and Michael Byers. Satellite mega-constellations create risks in low earth orbit, the atmosphere and on earth. *Scientific Reports*, 11(1):10642, 2021. ISSN 2045-2322. doi: 10.1038/s41598-021-89909-7. URL https://doi.org/10.1038/s41598-021-89909-7.

Francisco Caldas and Cláudia Soares. Machine learning in orbit estimation: A survey. *Acta Astronautica*, 220:97–107, 2024.

Hanyin Cheng, Xingjian Wu, Yang Shu, Zhongwen Rao, Lujia Pan, Bin Yang, and Guo Guo. CoRA: Boosting time series foundation models for multivariate forecasting through correlation-aware adapter. In *International Conference on Learning Representations*, volume 2026, pp. 106657–106680, 2026.

George Choumos, Konstantinos Tsaprailis, Vaios Lappas, and Charalampos Kontoes. Artificial intelligence for a safe space: Data and model development trends in orbit prediction and collision avoidance. In *AIAA SciTech 2024 Forum*, pp. 2066, 2024.

CS GROUP and Orekit contributors. Orekit: An accurate and efficient core layer for space flight dynamics applications, 2026. URL https://www.orekit.org/. Accessed: 2026-09-08.

Giacomo Curzi, Dario Modenini, and Paolo Tortora. Two-line-element propagation improvement and uncertainty estimation using recurrent neural networks. *CEAS Space Journal*, 14(1):197–204, 2022.

Abhimanyu Das, Weihao Kong, Rajat Sen, and Yichen Zhou. A decoder-only foundation model for time-series forecasting. *arXiv preprint arXiv:2310.10688*, 2023.

Zhiguo Deng, Thomas Nischan, and Markus Bradke. Multi-gnss rapid orbit-, clock-& eop-product series. 2017.

ESA Navigation Support Office. GNSS based products, 2026. URL https://navigation-office.esa.int/GNSS_based_products.html. Accessed: 2026-09-08.

European Space Agency. Space environment statistics, 2026. URL https://sdup.esoc.esa.int/discosweb/statistics/. Accessed: 2026-09-08.

William Fedus, Barret Zoph, and Noam Shazeer. Switch transformers: Scaling to trillion parameter models with simple and efficient sparsity. *Journal of Machine Learning Research*, 23(120):1–39, 2022.

G. H. Golub and C. Reinsch. Singular value decomposition and least squares solutions. *Numer. Math.*, 14(5):403–420, April 1970. ISSN 0029-599X. doi: 10.1007/BF02163027. URL https://doi.org/10.1007/BF02163027.

Jamil Haidar-Ahmad, Nadim Khairallah, and Zaher M Kassas. A hybrid analytical-machine learning approach for leo satellite orbit prediction. In *2022 25th International Conference on Information Fusion (FUSION)*, pp. 1–7. IEEE, 2022.

Tarek Hassan and Amr Hassan. A deep learning approach for gps orbital error prediction in offline real-time positioning. *Advances in Space Research*, 2025.

Sture Holm. A simple sequentially rejective multiple test procedure. *Scandinavian Journal of Statistics*, 6(2):65–70, 1979.

Felix R. Hoots and Ronald L. Roehrich. Models for propagation of NORAD element sets. Technical Report Spacetrack Report No. 3, Aerospace Defense Command, United States Air Force, December 1980.

Yifan Hu, Guibin Zhang, Peiyuan Liu, Disen Lan, Naiqi Li, Dawei Cheng, Tao Dai, Shu-Tao Xia, and Shirui Pan. TimeFilter: Patch-specific spatial-temporal graph filtration for time series forecasting. In *Forty-second International Conference on Machine Learning*, 2025. URL https://openreview.net/forum?id=490VcNtjh7.

Yoo-Min Jung and Leekyung Kim. MambaSL: Exploring single-layer Mamba for time series classification. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=YDl4vqQqGP.

Donald J. Kessler and Burton G. Cour-Palais. Collision frequency of artificial satellites: The creation of a debris belt. *Journal of Geophysical Research*, 83(A6):2637–2646, 1978. doi: 10.1029/JA083iA06p02637.

Yong Liu, Guo Qin, Zhiyuan Shi, Zhi Chen, Caiyin Yang, Xiangdong Huang, Jianmin Wang, and Mingsheng Long. Sundial: A family of highly capable time series foundation models. In *Forty-second International Conference on Machine Learning*, 2025. URL https://openreview.net/forum?id=LO7ciRpjI5.

Sukdev Mahapatra. Modeling orbital decay of low-earth orbit satellites due to atmospheric drag: A simplified analytical approach. *arXiv preprint arXiv:2508.19549*, 2025. doi: 10.48550/arXiv.2508.19549.

Md Mahmuddun Nabi Murad, Mehmet Aktukmak, and Yasin Yilmaz. WPMixer: Efficient multi-resolution mixing for long-term time series forecasting. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pp. 19581–19588, 2025.

Vahid Naghashi, Mounir Boukadoum, and Abdoulaye Banire Diallo. A multiscale model for multivariate time series forecasting. *Scientific Reports*, 15(1):1565, 2025.

Yuqi Nie, Nam H. Nguyen, Phanwadee Sinthong, and Jayant Kalagnanam. A time series is worth 64 words: Long-term forecasting with transformers. In *The Eleventh International Conference on Learning Representations*, 2023.

Ivan V. Oseledets. Tensor-train decomposition. *SIAM Journal on Scientific Computing*, 33(5):2295–2317, 2011. doi: 10.1137/090752286.

Hao Peng and Xiaoli Bai. Exploring capability of support vector machine for improving satellite orbit prediction accuracy. *Journal of aerospace information systems*, 15(6):366–381, 2018.

Nathan Reiland, Aaron J. Rosengren, Renu Malhotra, and Claudio Bombardelli. Assessing and minimizing collisions in satellite mega-constellations. *Advances in Space Research*, 67(11):3755–3774, 2021. ISSN 0273-1177. doi: https://doi.org/10.1016/j.asr.2021.01.010. URL https://www.sciencedirect.com/science/article/pii/S0273117721000326. Satellite Constellations and Formation Flying.

Reuters. Starlink rival eutelsat plans oneweb expansion after €1 billion satellite order. Reuters, September 2026. September 10, 2026.

Juan F San-Juana, Iván Pérezb, Eliseo Vergarac, Montserrat San Martınd, Rosario Lópeze, Alexander Wittigf, and Dario Izzog. Hybrid sgp4 propagator based on machine-learning techniques applied to galileo-type orbits. In *69th international astronautical congress, bremen, Germany*, volume 4, 2018.

Noam Shazeer, Azalia Mirhoseini, Krzysztof Maziarz, Andy Davis, Quoc V. Le, Geoffrey E. Hinton, and Jeff Dean. Outrageously large neural networks: The sparsely-gated mixture-of-experts layer. In *International Conference on Learning Representations*, 2017.

Xiaoming Shi, Shiyu Wang, Yuqi Nie, Dianqi Li, Zhou Ye, Qingsong Wen, and Ming Jin. Time-MoE: Billion-scale time series foundation models with mixture of experts. In *International Conference on Learning Representations*, volume 2025, pp. 34635–34667, 2025.

Space-Track.org. Space-Track.org Help Documentation, 2026. URL https://www.space-track.org/documentation. Accessed: 2026-09-08.

SpaceX. Starlink collision avoidance CONOPS, 2026a. URL https://space-safety.starlink.com/docs/space-safety-articles/starlink_cola_conops/. Accessed: 2026-09-26.

SpaceX. Satellite operators, 2026b. URL https://starlink.com/to/satellite-operators. Accessed: 2026-09-08.

David Vallado, Paul Crawford, Ricahrd Hujsak, and Thomas Sean Kelso. Revisiting spacetrack report# 3. In *AIAA/AAS astrodynamics specialist conference and exhibit*, pp. 6753, 2006.

Shiyu Wang, Haixu Wu, Xiaoming Shi, Tengge Hu, Huakun Luo, Lintao Ma, James Zhang, and Jun Zhou. TimeMixer: Decomposable multiscale mixing for time series forecasting. In *International Conference on Learning Representations*, volume 2024, pp. 38626–38652, 2024a.

Yuxuan Wang, Haixu Wu, Jiaxiang Dong, Guo Qin, Haoran Zhang, Yong Liu, Yunzhong Qiu, Jianmin Wang, and Mingsheng Long. TimeXer: Empowering transformers for time series forecasting with exogenous variables. *Advances in Neural Information Processing Systems*, 37:469–498, 2024b.

Dong Wei and Chang-Yin Zhao. An accuracy analysis of the SGP4/SDP4 model. *Chinese Astronomy and Astrophysics*, 34(1):69–76, 2010. doi: 10.1016/j.chinastron.2009.12.009.

James R. Wertz and Wiley J. Larson. *Space Mission Analysis and Design*. Microcosm Press and Kluwer Academic Publishers, 3rd edition, 2005. ISBN 9781881883104.

Gerald Woo, Chenghao Liu, Akshat Kumar, Caiming Xiong, Silvio Savarese, and Doyen Sahoo. Unified training of universal time series forecasting transformers. In *Forty-first International Conference on Machine Learning*, 2024. URL https://openreview.net/forum?id=Yd8eHMY1wz.

Ailing Zeng, Muxi Chen, Lei Zhang, and Qiang Xu. Are transformers effective for time series forecasting? *Proceedings of the AAAI Conference on Artificial Intelligence*, 37(9):11121–11128, 2023. doi: 10.1609/aaai.v37i9.26317.

Wei Zhang, Keke Zhang, Xingxing Li, and Jiande Huang. Improving leo short-term orbit prediction using lstm neural network. *Advances in Space Research*, 76(1):481–496, 2025.

Haoyi Zhou, Shanghang Zhang, Jieqi Peng, Shuai Zhang, Jianxin Li, Hui Xiong, and Wancai Zhang. Informer: Beyond efficient transformer for long sequence time-series forecasting. *Proceedings of the AAAI Conference on Artificial Intelligence*, 35(12):11106–11115, 2021. doi: 10.1609/aaai.v35i12.17325.

Tian Zhou, Ziqing Ma, Qingsong Wen, Xue Wang, Liang Sun, and Rong Jin. FEDformer: Frequency enhanced decomposed transformer for long-term series forecasting. In Kamalika Chaudhuri, Stefanie Jegelka, Le Song, Csaba Szepesvari, Gang Niu, and Sivan Sabato (eds.), *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pp. 27268–27286. PMLR, 17–23 Jul 2022.

Xingyu Zhou, Tong Qin, Mingjiang Ji, and Dong Qiao. A lstm assisted orbit determination algorithm for spacecraft executing continuous maneuver. *Acta Astronautica*, 204:568–582, 2023.

APPENDIX

A RELATED WORK

A.1 MACHINE LEARNING FOR SATELLITE ORBIT PREDICTION

Satellite orbit prediction has traditionally relied on analytical and numerical propagation models derived from orbital mechanics. Among analytical approaches, SGP4 remains a standard tool for propagating TLE sets because of its computational efficiency (Vallado et al., 2006). Its accuracy depends on the quality of the input orbital information and on how well the propagation model captures the relevant dynamics. This has motivated machine-learning methods that either learn propagation errors or augment conventional orbit-estimation and prediction pipelines(Caldas & Soares, 2024).

One line of work learns the residual errors of physics-based propagators. Peng and Bai (Peng & Bai, 2018) used support vector machines to learn orbit-prediction errors. San-Juan et al. (San-Juana et al., 2018) combined SGP4 with machine-learning-based error forecasting for Galileo-type orbits. Curzi et al. (Curzi et al., 2022) used a recurrent neural network to improve TLE propagation and estimate prediction uncertainty, with a separate neural-network corrector trained for each space object. Recurrent models have also been incorporated into orbit determination. Zhou et al. (Zhou et al., 2023) used an LSTM-assisted filtering framework for spacecraft executing continuous maneuvers.

These studies show that learned models can improve physics-based orbit prediction, but most evaluations focus on specific objects, orbital families, or maneuver scenarios (Haidar-Ahmad et al., 2022; Hassan & Hassan, 2025; Zhang et al., 2025). They do not directly address the setting considered here, where a single forecasting model is trained across multiple orbital regimes and evaluated on satellites that were not observed during training.

A.2 TIME-SERIES FORECASTING AND FOUNDATION MODELS

Satellite Cartesian states form a multivariate time series, which makes modern long-horizon forecasting models natural data-driven baselines. TimeXer (Wang et al., 2024b) models endogenous and exogenous information through patch-wise self-attention and variate-wise cross-attention. TimeMixer (Wang et al., 2024a) uses an MLP-based architecture that decomposes and mixes temporal patterns across multiple sampling scales. MultiPatchFormer (Naghashi et al., 2025) combines multi-scale temporal patches with channel-wise representation learning. WPMixer (Murad et al., 2025) integrates wavelet decomposition, patching, and MLP mixing. TimeFilter (Hu et al., 2025) constructs patch-level spatial-temporal graphs and adaptively filters irrelevant dependencies. Mamba (Jung & Kim, 2026) provides a general sequence-modeling alternative based on input-dependent selective state spaces with linear scaling in sequence length. CoRA (Cheng et al., 2026) introduces a lightweight adapter that models time-varying and time-invariant cross-variable correlations. We use these methods as supervised forecasters covering Transformer, MLP, wavelet, graph, state-space, and correlation-aware architectures.

Recent work has also developed time-series foundation models that transfer forecasting capability to unseen datasets without task-specific training. TimesFM (Das et al., 2023) pretrains a patched decoder-only forecasting model on a large time-series dataset. Moirai (Woo et al., 2024) uses a masked encoder designed to handle different frequencies, distributions, and numbers of variables. Chronos-Bolt (Ansari et al., 2024) improves inference efficiency through context patching and direct multi-step quantile prediction. Chronos-2 (Ansari et al., 2025) extends zero-shot forecasting to multivariate and covariate-informed settings using group attention. Time-MoE (Shi et al., 2025) scales forecasting capacity through sparse Mixture-of-Experts routing. Sundial (Liu et al., 2025) learns continuous predictive distributions with a flow-matching objective. TiRex (Auer et al., 2026) combines xLSTM with a masking-based pretraining strategy for zero-shot forecasting across short and long horizons.

These foundation models are designed to transfer across diverse time-series datasets, but they do not explicitly use orbital quantities or orbital-regime information in their architectures or routing mechanisms. The difference is particularly clear for Time-MoE. Time-MoE uses learned sparse routing and an auxiliary balancing loss to reduce routing collapse(Shi et al., 2025). Our model derives routing coordinates from orbital states and assigns experts by distance to physically initialized anchors that

Table 5: Threshold calibration and observed filtering yield on 183,863 eligible trajectory parts. The Bonferroni column is the nominal four-gate normal-tail bound $4[1-\Phi(k)]$. Reject rates are observed data-filtering rates.

Threshold	Bonferroni bound	Rejected parts	Reject rate	Rejected satellites
1σ	0.6346	56,631	30.8007%	8,998
2σ	0.09100	10,210	5.5530%	2,969
3σ	0.005400	750	0.4079%	551
4σ	0.0001267	14	0.00761%	13

are refined during training. We therefore evaluate all seven pretrained foundation models zero-shot, together with the seven supervised forecasters, to test whether general-purpose time-series models are sufficient for multi-regime satellite orbit prediction.

B Dataset Construction and Validation Details

B.1 In-domain TLE Dataset

For MEO, NSO, and GEO, the chronologically ordered TLE sequence of each satellite is first divided into independently continuous trajectory parts whenever consecutive TLE epochs are separated by more than 72 hours, as described in Section 2.1. Each resulting part is reconstructed as a 1-minute Cartesian state sequence before operational slicing.

Using the reconstructed states, we derive the semi-major axis, eccentricity, and inclination and compute 361-minute rolling median and median absolute deviation (MAD) statistics. We identify the dominant stable semi-major-axis band and retain the contiguous interval corresponding to the operational orbit, thereby removing launch, transfer, and post-mission disposal phases. Operational slicing preserves the pre-existing 72-hour part boundaries; retained states from different trajectory parts are never reconnected, and input and output windows never cross a part boundary.

Operational slicing removes large mission-phase transitions, but locally irregular trajectory parts can remain. We therefore apply an additional model-independent regularity filter before constructing the forecasting dataset. The filter operates on orbital-cycle statistics from four feature groups:

$$\mathcal{F} = \{\text{altitude}, \text{speed}, \text{eccentricity}, \text{inclination}\}. \tag{11}$$

Altitude and speed each aggregate multiple cycle-level MAD statistics into one feature-level score. Eccentricity and inclination each contribute one feature-level score. The resulting empirical scores are converted to rank-Gaussian normal scores through an inverse-normal transformation of their empirical ranks. Let z_{ij} denote the transformed score of part i for feature group j. For a candidate threshold k, the rejection rule is

$$\text{Reject}_k(i) = \mathbf{1}\left[\max_{j\in\mathcal{F}} z_{ij} > k\right]. \tag{12}$$

We calibrate k over the integer candidates $\{1, 2, 3, 4\}$. Under the rank-Gaussian normal-score interpretation, the one-sided tail probability for one gate is $1 - \Phi(k)$, where Φ denotes the standard normal cumulative distribution function. With four feature-level gates, the Bonferroni bound is

$$B_k = 4[1 - \Phi(k)]. \tag{13}$$

This quantity is used as a nominal family-wise tail bound for threshold calibration and is not an empirically measured false-rejection rate. We require $B_k < 0.01$. The corresponding values are

$$B_2 = 0.09100, \qquad B_3 = 0.005400, \qquad B_4 = 1.267 \times 10^{-4}. \tag{14}$$

Thus, 3σ is the smallest integer candidate that satisfies the 1% constraint.

Table 5 reports the observed filtering results for the final threshold-analysis population of 183,863 eligible trajectory parts. The selected 3σ threshold removes 750 parts (0.4079%) from 551 satellites. In comparison, 4σ removes only 14 parts (0.00761%) from 13 satellites.

Table 6: Matched comparisons between cleaned TLE-derived trajectories and external ephemeris products. A pair denotes one spacecraft–epoch position comparison.

Group / reference	Spacecraft	Pairs	Matched period (UTC)
GEO / BeiDou SP3	8	47,862	2025-12-07 to 2025-12-31
NSO / GNSS SP3	157	97,650,867	2014-01-05 to 2025-12-31
LEO / Starlink MEME	6,657	48,931,727	2024-11-25 to 2024-12-01

We further examine the 736 parts retained by 4σ but removed by 3σ. Among the three valid independent diagnostics, cycle-phase position RMSE and the local two-body acceleration residual show significantly greater instability after Holm correction (Holm, 1979), with adjusted one-sided permutation p-values of 1.50×10^{-4} and 0.00320, respectively. These results provide additional empirical support for selecting 3σ over the more permissive 4σ threshold.

The regularity filter is applied before model training and does not use model outputs, output residuals, or test-time errors. It is therefore independent of any post-hoc filtering performed during model evaluation.

B.2 Ephemeris-Based Validation of Cleaned Trajectories

We assess the cleaned TLE-derived trajectories against externally sourced ephemeris products for spacecraft with matching identifiers and overlapping epochs. The comparison covers representative spacecraft from LEO, NSO, and GEO. LEO trajectories are compared with 2024 SpaceX Starlink MEME ephemerides (SpaceX, 2026b;a), NSO trajectories with GNSS SP3 precise-orbit products (ESA Navigation Support Office, 2026), and GEO trajectories with GFZ multi-GNSS SP3 orbits for eight BeiDou satellites (Deng et al., 2017).

For epoch alignment, each UTC reference epoch is rounded to the nearest minute to locate the corresponding cleaned state. The reference ephemerides are not interpolated. Instead, the cleaned six-component state is propagated by at most 30 s to the exact reference epoch before computing the residual. GEO and NSO comparisons are evaluated in ITRS, while the LEO comparison is evaluated in GCRS. Table 6 summarizes the resulting matched comparisons.

Component-wise agreement. Let $e_{i,k} = r_{i,k}^{\mathrm{clean}} - r_{i,k}^{\mathrm{eph}}$ denote the paired position residual along axis $k \in \{x, y, z\}$. For each orbital group g, we compute

$$\mathrm{MAE}_{g,k} = \frac{1}{N_g} \sum_{i=1}^{N_g} |e_{i,k}|. \tag{15}$$

Uncertainty is estimated with 4,000 hierarchical bootstrap replicates, resampling spacecraft and then UTC calendar three-day blocks within each spacecraft while preserving the original paired-row weights. We additionally evaluate the one-sided margin criterion

$$H_0 : \mathrm{MAE}_{g,k} \geq \Delta_g, \qquad H_1 : \mathrm{MAE}_{g,k} < \Delta_g, \tag{16}$$

using Bonferroni-adjusted upper confidence limits.

We use regime-specific positional reference margins of $\Delta_{\mathrm{GEO}} = 27.9$ km, $\Delta_{\mathrm{NSO}} = 8.6$ km, and $\Delta_{\mathrm{LEO}} = 40$ km, based on the three-day prediction-error scales reported by Wei & Zhao (2010). The Bonferroni allocation is $0.05/12$, corresponding to the original four-group, three-axis analysis. Table 7 reports the component-wise MAE, RMSE, and Bonferroni-adjusted upper limits for the three evaluated orbital groups.

All nine component-wise Bonferroni-adjusted upper limits remain below their corresponding regime margins. GEO shows component-wise MAEs of 0.577–1.701 km, NSO 1.153–1.280 km, and LEO 4.268–4.651 km. These results show consistent component-wise agreement between the cleaned TLE-derived trajectories and the corresponding external ephemerides across the evaluated LEO, NSO, and GEO spacecraft.

Table 7: Component-wise agreement between cleaned trajectories and external ephemerides. All values are in km. U denotes the Bonferroni-adjusted one-sided bootstrap upper limit for MAE.

Group	Axis	MAE	RMSE	U	Δ
GEO	x	1.701	4.424	3.712	27.9
	y	1.427	3.512	3.327	27.9
	z	0.577	0.874	1.090	27.9
NSO	x	1.156	48.731	2.027	8.6
	y	1.153	48.755	2.020	8.6
	z	1.280	50.986	2.144	8.6
LEO	x	4.651	56.343	6.248	40.0
	y	4.592	56.049	6.293	40.0
	z	4.268	45.493	5.560	40.0

Table 8: Three-dimensional positional agreement at the regime-specific reference radius Δ. Lower denotes the Bonferroni-adjusted one-sided bootstrap lower limit.

Group	Δ (km)	Within Δ	Observed (%)	Lower (%)
GEO	27.9	47,297	98.820	94.863
NSO	8.6	97,014,720	99.349	99.105
LEO	40.0	47,654,663	97.390	97.085

Three-dimensional agreement. We additionally evaluate the proportion of matched states satisfying

$$\|\mathbf{e}_i\|_2 \leq \Delta_g. \tag{17}$$

Using the same spacecraft/three-day-block hierarchical bootstrap, we compute Bonferroni-adjusted one-sided lower confidence limits for the within-margin fraction. Table 8 reports the resulting three-dimensional agreement.

Table 9: Statistics of the external ephemeris datasets used for zero-shot evaluation. Temporal coverage denotes the minimum and maximum epochs contained in each processed dataset.

Regime	Satellites	Rows	Temporal coverage
LEO (Starlink)	10,682	45,797,059	2026-06-23 – 2026-06-26
MEO/NSO (GNSS)	34	554,096	2025-12-07 – 2026-06-06
GEO (BeiDou)	8	54,160	2025-12-06 – 2025-12-31

The observed within-margin fractions are 98.820% for GEO, 99.349% for NSO, and 97.390% for LEO. Together with the component-wise MAE analysis, these results provide an external ephemeris-based assessment of the reconstructed trajectory quality over the matched spacecraft and epochs.

B.3 External Ephemeris Datasets for Zero-Shot Evaluation

The external ephemerides used for forecasting evaluation are held out from model training and validation and are converted only into zero-shot evaluation sequences.

For LEO, we use public SpaceX Starlink ephemerides (SpaceX, 2026b). For the GNSS zero-shot set, we use ESA/ESOC final orbits of 36 GPS satellites (ESA Navigation Support Office, 2026). The BeiDou zero-shot set contains eight geostationary satellites and is constructed separately from the GPS set.

Table 9 summarizes the processed ephemeris datasets used for zero-shot evaluation. The Starlink set contains 10,682 satellites and 45,797,059 state records, the GNSS set contains 36 satellites and 554,112 records, and the BeiDou set contains 8 satellites and 53,871 states across 15 segments at the native 300 s spacing. Their temporal coverage spans 2026-06-23 to 2026-06-26, 2025-12-07 to 2026-06-06, and 2025-12-06 to 2025-12-31, respectively.

C Algorithm of Multi-regime Satellite Orbit Prediction Model

Algorithm 1 Algorithm of Multi-regime Satellite Orbit Prediction Model (MrSOP). L denotes the input-window length, H the output-window length, K the number of experts, d the model dimension of the backbone, d_{ff} the inner width of its feed-forward layer, $\mathbf{r}$ the TT-ranks, and μ the standard gravitational parameter of the Earth. $\mathrm{DMoE}(\cdot)$ denotes the DMoE layer.

Input: $\mathbf{X} \in \mathbb{R}^{L\times 6}$
Output: $\mathbf{Y} \in \mathbb{R}^{H\times 6}$
(i) Physics-anchored Routing
$(\mathbf{r}_L, \mathbf{v}_L) \leftarrow \mathbf{s}_L$
$a \leftarrow \big(2/\|\mathbf{r}_L\| - \|\mathbf{v}_L\|^2/\mu\big)^{-1}, \quad e \leftarrow \|\mathbf{e}(\mathbf{r}_L, \mathbf{v}_L)\|$
$\mathbf{z} \leftarrow (\log a,\, e), \quad \tilde{\mathbf{z}} \leftarrow (\mathbf{z} - \boldsymbol{\mu}_z) \oslash \boldsymbol{\sigma}_z \in \mathbb{R}^2$
$g_k \leftarrow \mathrm{softmax}_k\big(-\|\tilde{\mathbf{z}} - \mathbf{p}_k\|_2^2/\tau\big), \quad k = 1, \ldots, K$
(ii) DMoE
$\mathcal{T}_1 \leftarrow \mathrm{stack}(\mathbf{W}_1^{(1)}, \ldots, \mathbf{W}_1^{(K)}) \in \mathbb{R}^{K\times d_{\text{ff}}\times d}$
$\mathcal{T}_2 \leftarrow \mathrm{stack}(\mathbf{W}_2^{(1)}, \ldots, \mathbf{W}_2^{(K)}) \in \mathbb{R}^{K\times d\times d_{\text{ff}}}$
$(\mathcal{G}_1, \mathcal{G}_2, \mathcal{G}_3) \leftarrow \mathrm{TTD}(\mathcal{T}_1, \mathbf{r}), \quad (\mathcal{G}'_1, \mathcal{G}'_2, \mathcal{G}'_3) \leftarrow \mathrm{TTD}(\mathcal{T}_2, \mathbf{r})$
(iii) Backbone with DMoE Layers
$\{\mathbf{u}^{(s)}\}_{s=1}^{S} \leftarrow \mathrm{MultiScaleDownsample}(\mathbf{X})$
for each past-decomposable mixing block **do**
    Apply seasonal and trend mixing across scales
    $\mathbf{u} \leftarrow \mathrm{DMoE}(\mathbf{u}, \{g_k\}, \mathcal{G}_1, \mathcal{G}_2, \mathcal{G}_3, \mathcal{G}'_1, \mathcal{G}'_2, \mathcal{G}'_3)$
end for
(iv) Output
$\mathbf{Y} \leftarrow \mathrm{Projection}(\mathbf{u}) \in \mathbb{R}^{H\times 6}$
return $\mathbf{Y}$

D Routing Ablation Settings

The routing ablation compares two learned routing variants with the proposed physics-anchored routing, together with a single-expert TimeMixer backbone without mixture routing. Unsupervised routing operates on the orbital coordinate

$$\mathbf{z} = (\log a, e), \tag{18}$$

whereas latent routing predicts the gate directly from the temporal representation entering the mixture layer and does not use the orbital coordinate. The unsupervised variant operates directly on the unstandardized coordinate $\mathbf{z}$, while the physics-anchored router follows the standardized coordinate and anchor-based formulation defined in Section 3.2. These variants allow us to distinguish the effect of the routing signal from that of the anchor-based geometry built on it.

Unsupervised routing. Unsupervised routing replaces the anchor-based router with a learnable linear gate,

$$g_k(\mathbf{z}) = \mathrm{softmax}_k\left(\mathbf{W}\mathbf{z} + \mathbf{b}\right), \qquad \mathbf{W} \in \mathbb{R}^{K\times 2}. \tag{19}$$

The gate contains $3K$ trainable routing parameters, consisting of $2K$ weights and K biases. It is optimized using the forecasting loss and a load-balancing objective,

$$\mathcal{L}_{\text{unsup}} = \mathcal{L}_{\text{forecast}} + \lambda_{\text{lb}}\mathcal{L}_{\text{lb}}, \tag{20}$$

where $\lambda_{\text{lb}} = 0.01$. No orbital-regime labels are used at any stage.

Latent routing. Latent routing follows the conventional mixture-of-experts formulation, in which the gate is a learned function of the temporal representation entering the mixture layer rather than of an externally derived orbital coordinate. Let $\mathbf{U} = [\mathbf{u}_1, \ldots, \mathbf{u}_T]$ denote the hidden sequence entering the layer, with $\mathbf{u}_t \in \mathbb{R}^d$. To produce a single gate per input window, consistent with the other routing variants, we average the representation over the temporal dimension,

$$\bar{\mathbf{u}} = \frac{1}{T}\sum_{t=1}^{T}\mathbf{u}_t. \tag{21}$$

Table 10: Expert-count ablation within MrSOP. The main model uses $K = 16$. The ablation varies K while retaining the same backbone, physics-anchored routing, and DMoE formulation. Denormalized MAE is reported in km. Values are mean ± sample standard deviation over three runs. The best result is highlighted in **bold**, and the second-best is highlighted with underline.

K	Params	In-domain		Zero-shot 6-channel			Zero-shot Pos3D		
		6-channel	Pos3D	Starlink	GNSS	BeiDou	Starlink	GNSS	BeiDou
4	747,057	36.75 ± 2.54	**151.35 ± 10.40**	23.58 ± 2.72	50.87 ± 8.31	**19.01 ± 1.39**	93.47 ± 11.33	201.11 ± 33.26	**87.42 ± 6.35**
8	1,152,769	38.95 ± 0.97	160.40 ± 3.60	22.23 ± 1.61	63.63 ± 5.83	21.75 ± 1.15	87.87 ± 6.54	251.25 ± 23.61	100.09 ± 5.86
16 (Ours)	1,964,577	37.29 ± 3.57	153.87 ± 14.83	**21.26 ± 0.62**	**45.45 ± 2.82**	33.02 ± 15.78	**83.90 ± 2.61**	**179.36 ± 11.06**	152.09 ± 72.14
32†	3,589,729	**36.69 ± 2.60**	151.95 ± 11.77	21.36 ± 0.62	53.41 ± 9.51	25.72 ± 8.89	84.23 ± 2.61	210.38 ± 37.12	119.08 ± 41.83

The gate is then defined as

$$g_k(\bar{\mathbf{u}}) = \mathrm{softmax}_k\left(\mathbf{W}_{\mathrm{lat}}\bar{\mathbf{u}} + \mathbf{b}_{\mathrm{lat}}\right), \qquad \mathbf{W}_{\mathrm{lat}} \in \mathbb{R}^{K\times d}. \tag{22}$$

This router contains $K(d+1)$ trainable routing parameters. It is trained end-to-end using the forecasting objective together with the same load-balancing term used by unsupervised routing,

$$\mathcal{L}_{\mathrm{latent}} = \mathcal{L}_{\mathrm{forecast}} + \lambda_{\mathrm{lb}}\mathcal{L}_{\mathrm{lb}}, \tag{23}$$

with $\lambda_{\mathrm{lb}} = 0.01$. Comparing latent routing with unsupervised routing isolates the contribution of explicitly providing $(\log a, e)$ to the learned gate, since both variants are trained using the forecasting and load-balancing objectives and differ only in the input to the gate.

E Effect of the Number of Experts

The main MrSOP model is fixed at $K = 16$ experts for all main evaluations. Table 10 is an ablation within this model: we vary the expert count over $K \in \{4, 8, 16, 32\}$ while keeping the backbone, physics-anchored routing rule, and DMoE formulation fixed. This comparison measures sensitivity to the expert count and the associated parameter cost. The in-domain results remain relatively stable across the four configurations. The 6-channel MAE ranges from 36.69 to 38.95 km, and the Pos3D MAE ranges from 151.35 to 160.40 km. Increasing the number of experts therefore does not produce a consistent improvement on the in-domain test set.

The effect of the expert count depends on the zero-shot source. In Table 10, 16 experts give the lowest errors on Starlink and GNSS. The 6-channel and Pos3D MAEs are 21.26 ± 0.62 and 83.90 ± 2.61 km on Starlink, and 45.45 ± 2.82 and 179.36 ± 11.06 km on GNSS. Increasing the expert count to 32 does not improve these results.

On BeiDou, four experts give the lowest errors, 19.01 ± 1.39 and 87.42 ± 6.35 km; eight experts rank second at 21.75 ± 1.15 and 100.09 ± 5.86 km. The 16-expert configuration has larger errors and variability (33.02 ± 15.78 and 152.09 ± 72.14 km). These ablation results show how the effect of expert count varies across sources. The $K = 16$ main model remains fixed across the evaluations; the other expert counts are ablation variants of MrSOP.

F Analysis of SGP4 Errors

The large mean SGP4 error on some splits is not dynamical blow-up over the output window. The physics baseline evaluates a TLE and does not consume the Cartesian input window used by the learned models (Section 4). The operational product is a TLE, so for each input window we evaluate the latest TLE whose epoch is at most the last time of the input window, on the output timestamps. When that TLE disagrees with the Cartesian target, the output can already be wrong at the first output step. Learned models condition on an input window of 192 state vectors and therefore do not inherit TLE–ephemeris epoch mismatch.

On in-domain LEO the Pos3D MAE of each input window already has median $9{,}858$ km at the first output step and stays flat over the output window (95.2% of input windows exceed $1{,}000$ km; median TLE age 7 h). The archived series and the SGP4 trace share orbital radius and period; the miss distance is the chord of a nearly constant along-track offset, not SGP4 drift. Starlink shows the same along-track offset at the first output step (median $6{,}623$ km) because every sampled file is paired with a TLE about 77 days older than the ephemeris. In-domain MEO and GEO are typical

Table 11: GNSS Pos3D MAE versus TLE age at t_0.

TLE age	n	Median (km)	Mean (km)
$<$1 day	563	1.4	2.0
1–7 days	353	1.4	3.2
7–30 days	714	10.9	25.3
$>$30 days	3,927	289	589

Table 12: MrSOP versus SGP4. Denormalized MAE in km. MrSOP values are mean $\pm$ sample standard deviation over three runs, copied from Tables 2 and 3. SGP4 is deterministic.

Split	MrSOP		SGP4	
	6-channel	Pos3D	6-channel	Pos3D
LEO	44.55 ± 2.14	177.69 ± 8.62	2239.13	8900.15
MEO	47.27 ± 5.89	194.88 ± 24.02	268.30	1091.81
NSO	28.34 ± 3.49	113.60 ± 14.13	33.00	130.78
GEO	29.02 ± 3.23	129.41 ± 14.33	537.43	2487.25
Starlink	21.57 ± 0.91	85.12 ± 3.74	20009.84	77858.93
GNSS	45.99 ± 3.80	181.51 ± 14.97	107.96	419.74
BeiDou	33.02 ± 15.78	152.09 ± 72.14	0.52	2.18

when read as medians (31 km and 60 km), but the table means (1,092 km and 2,487 km) are driven by a $\sim$13% heavy tail. GNSS input windows with TLE age $<$1 day have median Pos3D 1.4 km; the reported mean of 420 km comes from the 71% of input windows whose TLE is older than 30 days (Table 11).

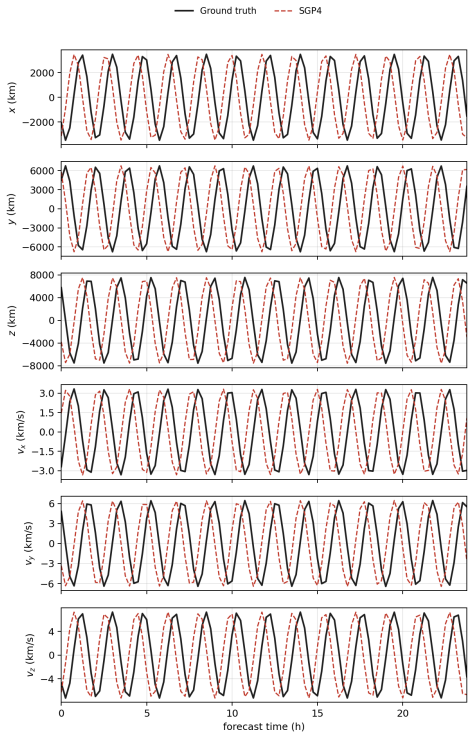

Figure 5: In-domain OneWeb-0137, TLE age 8.1 h: the same period and amplitude, with an along-track offset already at the first output step (geocentric angle 79°; Pos3D MAE 9,858 km; 6-channel MAE 2,455 km).

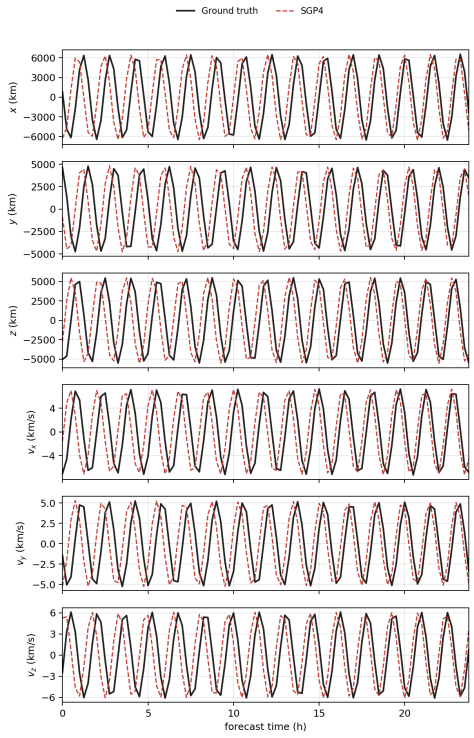

Figure 6: Zero-shot Starlink-3445, TLE age 77 d: the same period and amplitude, with an along-track offset already at the first output step (Pos3D MAE 6,623 km; 6-channel MAE 1,707 km).

Figures 5 and 6 plot all six channels of the output window for one input window per split. The two traces overlay in amplitude and period and differ by a time shift that is already present at the

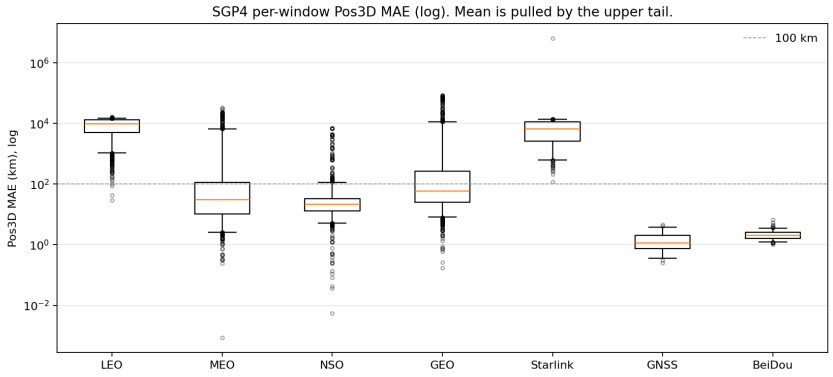

Figure 7: SGP4 Pos3D MAE for each input window (log scale). MEO/GEO medians are tens of kilometers; means are pulled by the upper tail.

first output step; the error does not grow over the output window. Figure 7 shows the Pos3D MAE distribution of each input window: in-domain LEO and Starlink boxes lie at 10^3–10^4 km (the typical input window, not outliers). MEO/GEO medians are tens of kilometers; means are tail-dominated. The BeiDou box stays near 2 km. Table 12 restates the headline comparison. The large SGP4 means on in-domain LEO and Starlink match the along-track offset in Figures 5 and 6 and the high boxes in Figure 7; NSO remains competitive for the physics baseline, and SGP4 achieves lower error than MrSOP on BeiDou.

G Fine-tuning Time-Series Foundation Models

Tables 2 and 3 evaluate seven pretrained time-series foundation models from their public checkpoints, without task-specific training. Two of those models, Sundial (Liu et al., 2025) and TiRex (Auer et al., 2026), do not provide a public fine-tuning recipe, so they cannot be adapted on our training split. Of the remaining five, Chronos-Bolt (Ansari et al., 2024) and Chronos-2 (Ansari et al., 2025) belong to the same family. Chronos-2 is the multivariate successor, and we therefore fine-tune Chronos-2 rather than both Chronos checkpoints. This leaves four models—Chronos-2, TimesFM (Das et al., 2023), Time-MoE (Shi et al., 2025), and Moirai (Woo et al., 2024)—which we fine-tune as an additional experiment.

The evaluation setup is identical to the main evaluation in Section 4: input-window length $L{=}192$, output-window length $H{=}96$, in-domain stride 15, and the same 6-channel MAE and Pos3D MAE. Fine-tuning uses only the in-domain training split. Each model is then scored on the in-domain test set and on the zero-shot ephemeris sources (Starlink, GNSS, and BeiDou). We follow each model's public fine-tuning interface rather than the 100-epoch schedule used for the supervised forecasters. Zero-shot rows are copied from the main tables so that fine-tuning can be read against the released checkpoint.

Table 13: In-domain MAE (km) before and after fine-tuning. Zero-shot rows copy Table 2. Fine-tuned values are mean ± sample standard deviation over three runs. Lower is better.

Model	Setting	6-channel MAE				Pos3D MAE			
		LEO	MEO	NSO	GEO	LEO	MEO	NSO	GEO
Chronos-2	Zero-shot	101.35	165.97	601.05	209.15	410.59	681.15	2367.09	924.24
	Fine-tuned	70.43 ± 0.68	99.42 ± 11.04	104.83 ± 5.78	118.61 ± 4.55	285.82 ± 2.76	410.97 ± 44.06	413.75 ± 22.23	518.68 ± 20.09
TimesFM	Zero-shot	1688.02	1142.77	59.12	96.23	6682.67	4729.69	234.47	440.58
	Fine-tuned	83.99 ± 2.28	177.06 ± 37.61	54.28 ± 2.10	66.38 ± 0.51	337.94 ± 9.72	731.45 ± 156.40	218.53 ± 8.12	299.82 ± 2.43
Time-MoE	Zero-shot	326.11	581.86	1373.61	2172.75	1315.00	2490.02	5431.45	9937.58
	Fine-tuned	2099.85 ± 32.67	1654.30 ± 18.76	1111.63 ± 9.44	767.42 ± 15.13	8335.49 ± 128.78	6919.76 ± 68.52	4369.48 ± 37.81	3479.76 ± 87.04
Moirai	Zero-shot	1977.71	3647.28	5474.14	12421.95	7988.44	15256.07	21850.58	56443.30
	Fine-tuned	753.65 ± 16.97	2298.93 ± 58.03	531.55 ± 210.01	77.92 ± 2.18	3120.58 ± 68.91	9797.12 ± 249.64	2160.32 ± 855.36	347.76 ± 9.71
MrSOP (ours)		**44.55 ± 2.14**	**47.27 ± 5.89**	**28.34 ± 3.49**	**29.02 ± 3.23**	**177.69 ± 8.62**	**194.88 ± 24.02**	**113.60 ± 14.13**	**129.41 ± 14.33**

Table 13 shows that fine-tuning reduces in-domain error relative to the released checkpoints for Chronos-2, TimesFM, and Moirai. Time-MoE improves on NSO and GEO, but its mean error across

the four regimes is higher than that of its public checkpoint. MrSOP remains lower on every in-domain regime–metric comparison in this table.

Table 14: Zero-shot ephemeris MAE (km) for the same checkpoints as Table 13. Fine-tuned and MrSOP values are mean ± sample standard deviation over three runs. The MrSOP row matches Table 3. The best result is bold and the second-best is underlined. Lower is better.

Model	Setting	6-channel MAE			Pos3D MAE		
		Starlink	GNSS	BeiDou	Starlink	GNSS	BeiDou
Chronos-2	Zero-shot	77.03	271.24	173.77	306.95	1047.88	780.69
	Fine-tuned	49.48 ± 0.83	88.93 ± 2.91	76.23 ± 0.77	199.91 ± 3.81	350.44 ± 10.86	346.55 ± 4.31
TimesFM	Zero-shot	1661.34	75.57	107.16	6505.07	299.20	497.41
	Fine-tuned	59.60 ± 4.28	65.74 ± 4.68	61.12 ± 1.59	238.30 ± 17.60	263.29 ± 19.09	281.72 ± 6.97
Time-MoE	Zero-shot	258.97	442.11	2069.27	1033.02	1753.63	9526.47
	Fine-tuned	2144.03 ± 40.75	1151.09 ± 20.25	771.57 ± 21.84	8383.82 ± 158.85	4489.07 ± 83.45	3566.44 ± 120.59
Moirai	Zero-shot	1961.69	4236.18	12398.08	7732.32	16977.23	58014.75
	Fine-tuned	626.53 ± 5.60	130.05 ± 21.55	**30.50 ± 1.90**	2562.49 ± 21.29	543.24 ± 93.02	**144.43 ± 9.74**
MrSOP (ours)		**21.57 ± 0.91**	**45.99 ± 3.80**	33.02 ± 15.78	**85.12 ± 3.74**	**181.51 ± 14.97**	152.09 ± 72.14

Table 14 scores the same fine-tuned checkpoints on the held-out ephemerides. On Starlink and GNSS, MrSOP retains the lowest mean error for both metrics. On BeiDou, fine-tuned Moirai has lower mean 6-channel and Pos3D errors (30.50 ± 1.90 and 144.43 ± 9.74) than MrSOP (33.02 ± 15.78 and 152.09 ± 72.14). The advantage of task-specific architecture is therefore source-dependent. These results motivate incorporating orbital elements and regime structure into forecasting models, while showing that the relative benefit must be assessed separately for each external source.

H Inference Time

Table 15 reports wall-clock inference under the evaluation protocol of Section 4: input-window length $L=192$ and output-window length $H=96$, on a single NVIDIA GeForce RTX 3090. Input windows are preloaded onto the device, so the numbers exclude I/O. Five warmup batches are discarded; the reported milliseconds per input window are then timed forwards with CUDA synchronization. SGP4 is CPU-only. In-domain and zero-shot ephemeris inputs have the same tensor shape, so the two columns are a repeatability check rather than two different graphs. The table covers the pretrained foundation models, the supervised forecasters, and SGP4. Five supervised forecasters finish an input window in under a millisecond. TimeMixer, MrSOP, and CoRA take about 1.18, 3.38, and 6.72 ms, respectively. The pretrained foundation models range from 5.67 to 458.4 ms, so all seven are slower than MrSOP. TimesFM is the slowest at 458 ms per input window, which would be about 12.7 hours for the $100{,}000$ input windows of the in-domain evaluation. Chronos-2, Chronos-Bolt, Moirai, TiRex, Sundial, and Time-MoE sit between 5.67 and 81.3 ms. CoRA is closer to the foundation models than to the other supervised rows because it wraps a frozen 356.5M-parameter encoder. The in-domain and zero-shot timings are similar, as expected from the shared input shape.

Table 15: Inference cost. Milliseconds per input window exclude data loading. Lower is faster.

Model	Params	In-domain (ms)	Zero-shot (ms)
SGP4	–	0.186	0.197
Chronos-2	119.5M	5.67	5.67
TimesFM	231.3M	458.4	457.0
Time-MoE	453.2M	81.0	81.3
Sundial	128.3M	56.0	55.2
Chronos-Bolt	205.3M	14.91	14.95
TiRex	35.3M	30.55	30.39
Moirai	91.4M	11.95	11.89
TimeFilter	3.05M	0.411	0.411
MultiPatchFormer	15.39M	0.122	0.126
WPMixer	2.94M	0.106	0.106
Mamba	0.44M	0.035	0.035
CoRA	356.5M	6.72	6.77
TimeXer	4.93M	0.113	0.118
TimeMixer	2.21M	1.178	1.177
MrSOP (ours)	1.96M	3.383	3.358