FROM SPACE WEATHER TO ORBITS: AN UNCERTAINTY-AWARE FRAMEWORK FOR PREDICTING SATELLITE TRAJECTORIES

Sergio Sanchez-Hurtado[(1)], **William E. Parker**[(2)], **Victor Rodriguez-Fernandez**[(1)], **and Richard Linares**[(2)]

[(1)] *Universidad Politécnica de Madrid, ETSI Sistemas Informáticos, Madrid, Spain, Emails:s.shurtado@alumnos.upm.es, victor.rfernandez@upm.es*

[(2)] *Massachusetts Institute of Technology, Cambridge, Massachusetts 02139, Emails: {wparker, linaresr}@mit.edu*

ABSTRACT

Accurate satellite orbit prediction is becoming increasingly critical due to the growing density of objects in low-Earth orbit (LEO) and the unpredictable effects of space weather. Traditional physics-based models often struggle under extreme solar conditions, largely because they lack mechanisms to quantify predictive uncertainty. In this work, we propose an uncertainty-aware framework for predicting orbital perturbations using Graph Neural Networks (GNNs) enriched with space weather indicators and satellite-specific data. Our architecture incorporates a Bayesian multi-relational Graph Convolutional Network (GCN) that captures dynamic inter-satellite relationships through multiple graph views, based on proximity, orbital similarity, and operator classification. Combined with a temporal forecasting module, this framework enables accurate orbit predictions and real-time uncertainty estimation via Monte Carlo sampling. We also explore a complementary uncertainty estimation approach using time series classification, which leverages probabilistic outputs to infer model confidence. Preliminary experiments with SWARM and GRACE-FO datasets show that our model performs on par with traditional propagators while offering robust confidence intervals, particularly during geomagnetic storms. These findings demonstrate the promise of GNN-based methods for reliable orbit forecasting, anomaly detection, and large-scale satellite catalog maintenance in complex, data-rich environments.

Keywords: Space Weather, GNNs, Satellite Propagation.

Every decade, the activity of our Sun attracts significant attention, especially when extreme events occur and their impacts are felt on Earth. During these events, even general and popular press outlets begin publishing articles explaining the effects of Solar Storms. In some cases, they analyze how solar storms are addressed and their potential impacts on our way of life [1, 2]. However, that is often the extent of public awareness.

For the average person, these phenomena may seem to cause only minor technological inconveniences, such as brief electrical blackouts, mobile phones failing to track locations accurately, or the familiar static noise on the radio when no signal can be received. Some people might also be aware that during these storms, auroras can be seen in locations far from the usual regions near the North and South Poles. However, solar storms are far from trivial. They can lead to severe problems, such as satellite malfunctions or unexpected orbital decay. In extreme cases, this may result in satellites crashing back to Earth in an uncontrolled manner, which is not only costly but also highly dangerous [3, 4]. Furthermore, solar storms can cause significant damage to power grids, potentially leading to permanent infrastructure failures. Such failures can have long-term economic and safety consequences [5].

During 2024, solar storms once again captured the scientific community's attention, thanks to unusually extreme events occurring in May, August, and October. These episodes not only challenged our understanding of space weather but also underscored the tangible impacts on satellite operations. Intense solar activity during these periods led to significant geomagnetic disturbances, which in turn contributed to enhanced atmospheric density variations and increased satellite drag. According to Parker and Linares [6], who analyze the impact of the last May solar storm, the Ganon Solar was triggered by several X-Class solar flares and coronal mass ejections (CMEs) from some solar active regions that severely impacted Earth's magnetosphere and upper atmosphere. This event poses a serious risk, as such extreme and sudden occurrences can disrupt both ground and space operations by producing high voltage inductions, sudden electronic failures, and unexpectedly extreme drag on satellites traversing Earth's atmosphere.

With this project, we aim to focus on the challenge of predicting when satellites are likely to experience an unexpected decay, enabling us to better prepare for their consequences. As highlighted by the scientific community and even by media outlets [7], our current models are not sufficiently equipped to predict these events with the necessary lead time.

Moreover, these reports emphasize the urgency of tackling this issue as soon as possible. The paradigm in the field of space operations is shifting, with an increasing

Proc. 9th European Conference on Space Debris, Bonn, Germany, 1–4 April 2025, published by the ESA Space Debris Office
Editors: S. Lemmens, T. Flohrer & F. Schmitz, (http://conference.sdo.esoc.esa.int, April 2025)

number of satellites populating Low Earth Orbit (LEO) [3, 6]. This growing congestion significantly raises the risk of catastrophic impacts if such events remain unpredictable, in terms that we are not able to be aware when satellites can suffer from major drag effects, not even in a probabilistic way, as there is an important lack of reliable uncertainty aware methods with real-world implications already being observed [8].

In response to these challenges, our project leverages Graph Neural Networks (GNNs) alongside advanced time series methodologies to predict satellite drag with an uncertainty-aware approach. By integrating these techniques, we aim to deliver more reliable forecasts that account for the inherent variability of solar-driven phenomena, ultimately supporting improved decision-making in satellite mission planning and operations.

1. CURRENT LITERATURE COMPARISON

Traditionally, a wide set of approaches have been employed to predict and prepare for such solar events, but the main ones can be described as a mixture of an empirical atmospheric model embedded within a propagator or a hybrid method that combines physics-based simulations with data-driven corrections. Following the research trend we can find mainly three primary methodologies for predicting satellite drag or altitude loss that necessitate maneuvers: numerical integration, semi-analytical methods, and machine learning models.

1.1. Numerical Integration Methods

Numerical integration methods involve solving the satellite's equations of motion through detailed, step-by-step calculations that account for atmospheric drag and other perturbative forces [9]. However, these approaches struggle during periods of high variance in solar activity with sudden peaks, as they are primarily effective for short-term predictions under stable solar conditions. Although these methods offer high accuracy, which is essential for precise orbit predictions, their computational cost can become significant, particularly for long-term forecasts or when managing large satellite constellations [10]. This common approximation relies on atmospheric models to perform numerical orbit propagation, a process that depends on multiple factors: accurate solar activity predictions, reliable models for retrieving atmospheric densities, and high-fidelity propagators. At each stage, the limited accuracy of the available methods results in accumulated errors across the different tools, ultimately leading to a higher overall error during challenging periods.

However, Flores et al. [9] point out something crucial for maneuver planning: the development of uncertainty-aware methods that can guide satellite operators in choosing the optimal time to maneuver, thereby adding confidence to the decision-making process.

1.2. Semi-Analytical Methods

Another tendency in orbit prediction involves semi-analytical methods, which blend analytical formulations with numerical corrections to balance accuracy and computational efficiency. They average perturbations over orbital periods, making them particularly useful for long-term predictions and for maintaining space object catalogs [11]. These authors especially recognize the performance of their model for debris orbit prediction, which is coherent with expectations since debris, lacking maneuver capabilities, exhibits more predictable behavior. This predictability is further enhanced by the separation of short-term and long-term error components, allowing the model to effectively integrate historical events into current data for accurate orbit propagation.

Despite their advantages, these methods may encounter challenges when dealing with highly perturbed orbits, where the underlying simplifications can reduce prediction accuracy [12]. Furthermore, they are less suited for preventing the events mentioned at the start of this paper, as their focus on long-term dependencies and patterns limits their ability to provide real-time predictions of imminent orbital decay.

1.3. Machine Learning/Data-Driven Models

Lastly, numerous machine learning approaches leverage historical satellite data to predict orbital decay and drag-induced maneuvers. By using techniques such as Support Vector Machines, Artificial Neural Networks, and Gaussian Processes, these models are capable of learning complex patterns and correcting errors in traditional physics-based models [13]. Their adaptability is particularly promising for handling uncertainties introduced by solar events, although their performance critically depends on the quality and volume of available data [14]. While the large number of satellites orbiting Earth provides a wealth of data, the lack of synchronization in the reporting process poses a challenge, necessitating significant efforts in data preprocessing and preparation before the data can be fed into the model.

As we have seen, there are numerous ways to approach orbital prediction, or at least to reduce uncertainty and help operators make safer decisions. In low Earth orbit (LEO), quantifying uncertainty is crucial due to the inherently variable nature of space weather and its impact on orbital trajectories. Understanding how reliable a model is in its predictions allows for a more complete assessment of whether rapid maneuvers are necessary to avoid potential issues.

Supporting this, as noted by Parker et al. [15], poor predictions of geomagnetic conditions can hinder reliable conjunction assessments, delaying timely collision avoidance maneuvers. Accurate estimation of atmospheric drag in the thermosphere is especially vital in a congested LEO environment, where sudden density changes caused by solar activity significantly raise collision risks, as highlighted by Matsuo et al. [16] and Hypolite et al. [17]. The growing number of satellites and space debris calls for robust probabilistic models to guide operational decisions under uncertainty, a need stressed by Smith and Doe [18] and Space.com [19].

Recent advances in deep evidential frameworks (Li et al. [20]) offer promising solutions, providing both predictive accuracy and explicit uncertainty estimates for thermospheric density during geomagnetic storms. More broadly, as emphasized by several authors [21, 22, 23, 24], integrating uncertainty-aware techniques into space weather forecasting is essential for maintaining safe and efficient satellite operations in the increasingly crowded LEO environment.

That's why we are starting to develop a framework that incorporates what we believe are good practices, or at least key points, for this task: having uncertainty determination estimates, integrating all data within the same framework to avoid the prediction pipeline relying on too many steps (with their associated errors), and being able to update a large portion of the satellite catalogue in a short period of time. This is why we suggest the use of GNNs to address the problem, as they can profit from all the information present in the network of satellites and are capable of making accurate predictions. Moreover, they are well-suited to exploit all possible inputs that can affect satellite drag, such as space weather data and the states of neighboring satellites, and they are able to capture the spatio-temporal relationships among satellites that are critical for predicting their future states.

2. OUR APPROACH

From our perspective, the traditional pipeline for satellite orbit and drag prediction, beginning with space weather indices to construct an atmospheric model, which is then fed into an orbit propagator, introduces significant sources of error and often fails to deliver satisfactory performance. This approach is commonly used by satellite operators for orbit prediction. However, as highlighted by Parker & Linares [6], a clear example of its limitations was observed during the Ganon Storm in May 2024, when most LEO satellites had to perform corrective maneuvers within just one day to recover their nominal orbits. This response effort poses a major operational challenge, as a large number of satellites require timely maneuvering over a short time window due to severe altitude loss induced by increased atmospheric drag.

As shown in the previous section, one of the key challenges to address is the lack of uncertainty quantification in this type of prediction. That is why we prioritized this problem from the beginning, aiming to incorporate uncertainty-aware solutions early in the model design process.

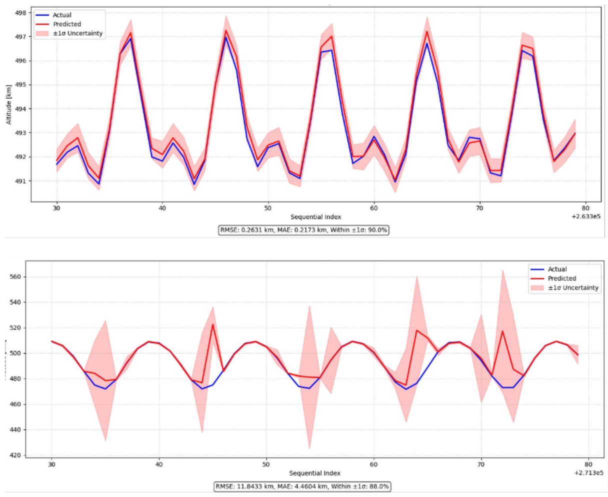

Figure 1. Uncertainty estimates for $t + 1$ altitude predictions, concatenated over 80 steps of 10-minute intervals.

To this end, we propose an end-to-end architecture based on Graph Neural Networks (GNNs), capable of processing standard inputs and learning how to effectively apply them to orbit prediction tasks. In this simplified case study, the goal is to predict variations in satellite altitude using multiple temporal snapshots of a graph. The nodes represent satellites and remain constant, while the model updates their altitude state by learning from evolving relationships and features over time.

In parallel, we explored a complementary approach to uncertainty estimation, based on the idea of decoupling it from the main model. This allows for more flexible experimentation with techniques better suited for representing predictive uncertainty, especially in scenarios where end-to-end uncertainty propagation may be difficult to achieve or interpret.

2.1. Uncertainty Estimation

From the early stages of model development, we focused on enabling reliable uncertainty predictions. This decision was also motivated by our intent to separate the uncertainty estimation process from the core predictive model, allowing us to experiment with alternative uncertainty modeling strategies.

2.1.1. Uncertainty Estimation Using a Time Series Classifier

To estimate uncertainty, we explored training a separate model dedicated to this task. Specifically, we used a time series classifier, leveraging its inherent probabilistic outputs to derive uncertainty metrics. This type of model provides a distribution over possible class labels, which can be analyzed statistically to estimate confidence in the predictions.

This idea was inspired by a participant in a machine learning competition who used a similar approach to predict pressure values [25]. The method is also supported by Hacibeyoglu et al. [26], who proposed using classifiers for continuous variable prediction by discretizing the target and interpreting the resulting probability distribution as an uncertainty estimate.

What we did was discretized the target variable, altitude, into several bins. After testing different configurations, we found that using 200 bins for a 300 km range (yielding a bin size of approximately 660 meters) provided a good balance between resolution and performance. Our goal was to predict only the next time step ($t + 1$) with an associated uncertainty estimate. To do so we used ResNetPlus architecture extrated from the tsai python library [27], which we configure with its presets without further arch configuration. To this model, we fed sliding windows of 64 steps in the past of Ap and altitude values.

While this approach performed well over various time periods, we eventually decided to temporarily set it aside. The main limitations were its poor scalability when increasing resolution (as more bins were required) and its limited suitability for multivariate extensions. Additionally, the method occasionally produced unexpected spikes in uncertainty at random intervals, as shown in Figure 1. Although we do not rule out revisiting this approach in the future, it is not incorporated into the final model presented here.

2.1.2. Graph Bayesian Aggregation

Following the classification approach, we explored a Graph Bayesian Aggregation (GBA) framework, inspired by the work of Hu et al. [28], who successfully applied it to a spatio-temporal extrapolation problem. Given the similarities in structure and requirements, we adapted the method to our context.

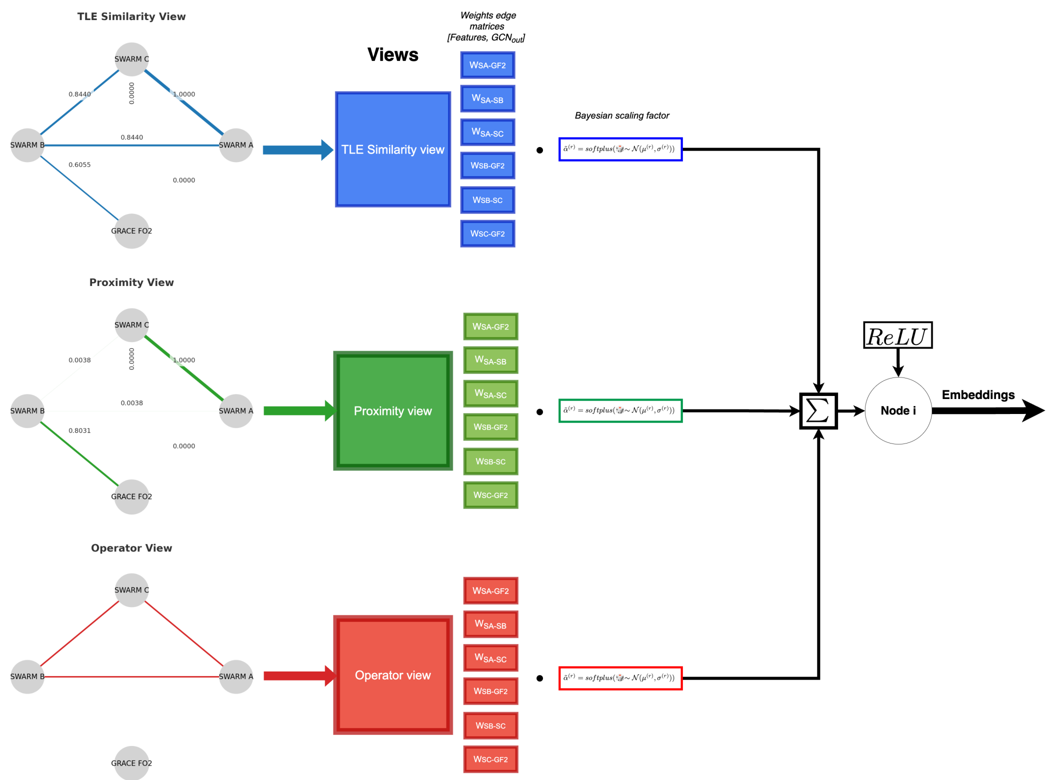

Figure 2. Graph Bayesian Aggregation example applied to our use case. Here we represent how this process works using Similarity, Proximity and Operator views.

In our case, we model multiple types of relationships between satellites, such as spatial proximity, similarity in orbital parameters, or current dynamical states. These relationships define distinct views or adjacency representations of the satellite graph. The GBA model processes each view independently, allowing it to assign different levels of relevance to each representation based on its contribution to prediction performance. These multiple views are then aggregated in a Bayesian fashion, introducing stochasticity into the output.

This probabilistic approach allows us to obtain different predictions across multiple runs, with the variance between predictions reflecting the model's confidence, more compressed distributions indicating higher certainty. A visual summary of this mechanism is provided in Figure 2, which illustrates the GBA process. Importantly, this method addresses the scalability limitations of our previous model while seamlessly integrating uncertainty estimation into the prediction process. As a result, there is no need for a separate module to assess prediction confidence, everything is handled within a single, unified model. A more detailed discussion of this approach follows in the next section.

2.2. Final model

Our model is designed to integrate satellite relational dynamics with temporal forecasting over graph-structured data. Each node in the graph represents a satellite, characterized by a set of evolving features derived from TLE parameters, physical constants, and space weather indices. Instead of explicitly encoding spatial coordinates, the model emphasizes learning the interactions and dependencies between satellites.

The architecture comprises two main components: a relational module based on a Bayesian Multi-Relational Graph Convolutional Network (GCN), which processes multiple graph views capturing different types of satellite relationships, and a temporal module that employs an LSTM with multi-head self-attention to model the temporal evolution of node features. A high-level schematic of the architecture is presented in Figure 3.

2.2.1. Relational Module – Bayesian Multi-Relational GCN

At each time step t, we have a graph snapshot with:

$$X^{(t)} \in \mathbb{R}^{N \times F},$$

where N is the number of satellites and each row $x_i^{(t)}$ represents the feature vector of satellite i.

For each relation r (e.g., proximity, TLE similarity), we compute a dynamic adjacency matrix:

$$A^{(r)}(t) \in \mathbb{R}^{N \times N},$$

and include static relationships (e.g., common operator) as precomputed matrices $A^{(r)}_{\text{static}}$.

Each relation has a learnable weight matrix:

$$W^{(r)} \in \mathbb{R}^{F \times H},$$

where H is the embedding dimension. In addition, we model a per-relation scalar $\alpha^{(r)}$ as a random variable:

$$\alpha^{(r)} \sim \mathcal{N}\big(\mu^{(r)}, (\sigma^{(r)})^2\big), \quad \text{with } \sigma^{(r)} = \exp\big(\text{agg\_logstd}^{(r)}\big),$$

and sample a scaling factor during propagation:

$$\tilde{\alpha}^{(r)} = \text{softplus}\Big(\mu^{(r)} + \epsilon^{(r)}\,\sigma^{(r)}\Big), \quad \epsilon^{(r)} \sim \mathcal{N}(0,1).$$

The message from neighbors for node i at time t is computed as:

$$m_i^{(r)}(t) = \sum_{j \in \mathcal{N}^{(r)}(i,t)} A_{ij}^{(r)}(t)\, W^{(r)} x_j^{(t)},$$

and the aggregated representation is:

$$h_i(t) = \text{ReLU}\left(\sum_{r=1}^{R} \tilde{\alpha}^{(r)}\, m_i^{(r)}(t) + b\right),$$

where $b \in \mathbb{R}^H$ is a bias term. This results in a new embedding matrix for each time step:

$$H^{(t)} \in \mathbb{R}^{N \times H}.$$

2.2.2. Temporal Module – LSTM with Multi-Head Self-Attention

For each satellite i, we now have a sequence of embeddings over a sliding window of W time steps:

$$\{h_i(1), h_i(2), \ldots, h_i(W)\}, \quad h_i(t) \in \mathbb{R}^H.$$

This sequence is processed by an LSTM, which produces a series of hidden states:

$$s_i(1), s_i(2), \ldots, s_i(W) \in \mathbb{R}^D,$$

where D is the LSTM's hidden dimension.

On top of the LSTM outputs, we apply a multi-head self-attention mechanism to enable the model to focus on the most relevant regions of the embedding space. This design is motivated by our objective to input a large number of variables, allowing the model to learn to selectively attend to the most informative features for prediction. The outputs from all attention heads are concatenated and passed through a final linear projection layer. Subsequently, temporal aggregation (e.g., via mean pooling) is applied to derive a fixed-size context vector for each satellite.

$$c_i = \frac{1}{W} \sum_{t=1}^{W} \tilde{s}_i(t),$$

where $\tilde{s}_i(t)$ is the output from the attention layer.

The final forecast is produced by a fully connected layer:

$$\hat{y}_i = f(c_i) \in \mathbb{R}^L,$$

with L being the forecast horizon.

2.2.3. Uncertainty Estimation and Overall Data Flow

Because the Bayesian GCN samples scaling factors $\tilde{\alpha}^{(r)}$ during each forward pass, the output is inherently stochastic. By performing multiple forward passes (Monte Carlo sampling), we obtain a distribution of outputs from which we can compute both the mean forecast and its standard deviation as an uncertainty estimate. Doing this way, the model learns how to weight the different view representations of each node's edges, and it captures uncertainty by detecting deviations from the typical weighting.

This combined approach adapts the ideas used in Spatio-Temporal Convolution Recurrent Neural Networks [30] and applies them to leverage contextual data from other satellites instead of traditional spatial information. This allows our model to capture the complex inter-satellite relationships and their time-evolving behavior, while also quantifying uncertainty, a powerful tool for reliable orbit prediction in noisy and dynamic environments.

2.3. Data preparation

For our initial approximation, we used data from the SWARM and GRACE-FO missions, as they provide some of the most accessible and well-documented datasets. These datasets offer satellite position and velocity vectors at a 10-minute resolution over extended time periods, which were extracted using the HAPI interface provided by the VirES for SWARM service [29]. This resolution aligns well with our target forecasting interval. In our dataset, each row represents a satellite's state at a given epoch. The primary objective in this initial stage is to predict altitude variations with high precision and performance.

Each satellite is represented as a node with a feature vector that includes:

- **Orbital and TLE-derived Parameters:** Parameters such as mean motion, eccentricity, inclination, right ascension of the ascending node (RAAN), argument of pericenter, and mean anomaly capture the orbit's geometry and behavior.
- **Physical and Operational Characteristics:** We compute the approximate ballistic coefficient for each satellite, which is critical for understanding its interaction with atmospheric drag. Additionally, categorical information such as operator/mission (e.g., Communication, Navigation, Military) is incorporated through one-hot encoding or embedding.
- **Positional Data:** Each node contains latitude, longitude, and altitude (in km), with optional inclusion of velocity components to capture dynamic behavior.
- **Space Weather Indices (Global Context):** These indices are modeled as separate global nodes connected to every satellite node.

We also create multiple "views" of the graph by constructing different adjacency matrices that reflect various relationships between satellites:

1. Proximity View: For each epoch, we extract a satellite's position vector

$$p_i = [\text{latitude, longitude, altitude\_km}],$$

compute the Euclidean distance between satellites:

$$d_{ij} = \|p_i - p_j\|,$$

and convert it into a similarity measure using a Gaussian (RBF) kernel:

$$A_{ij}^{\text{prox}} = \exp\Big(-\frac{d_{ij}^2}{2\sigma^2}\Big),$$

with σ as a scale parameter.

2. TLE Similarity View: For each satellite, we construct a vector of TLE parameters:

$$v_i = \begin{cases} \text{mean motion} \\ \text{eccentricity} \\ \text{inclination} \\ \text{RAAN} \\ \text{argument of pericenter} \\ \text{mean anomaly} \end{cases}$$

and compute cosine similarity:

$$A_{ij}^{\text{tle}} = \frac{v_i \cdot v_j}{\|v_i\| \, \|v_j\|}.$$

3. Categorical Views: For instance, the operator view connects satellites managed by the same organization:

$$A_{ij}^{\text{op}} = \begin{cases} 1 & \text{if operator}_i = \text{operator}_j, \\ 0 & \text{otherwise,} \end{cases}$$

In our current design, we include:

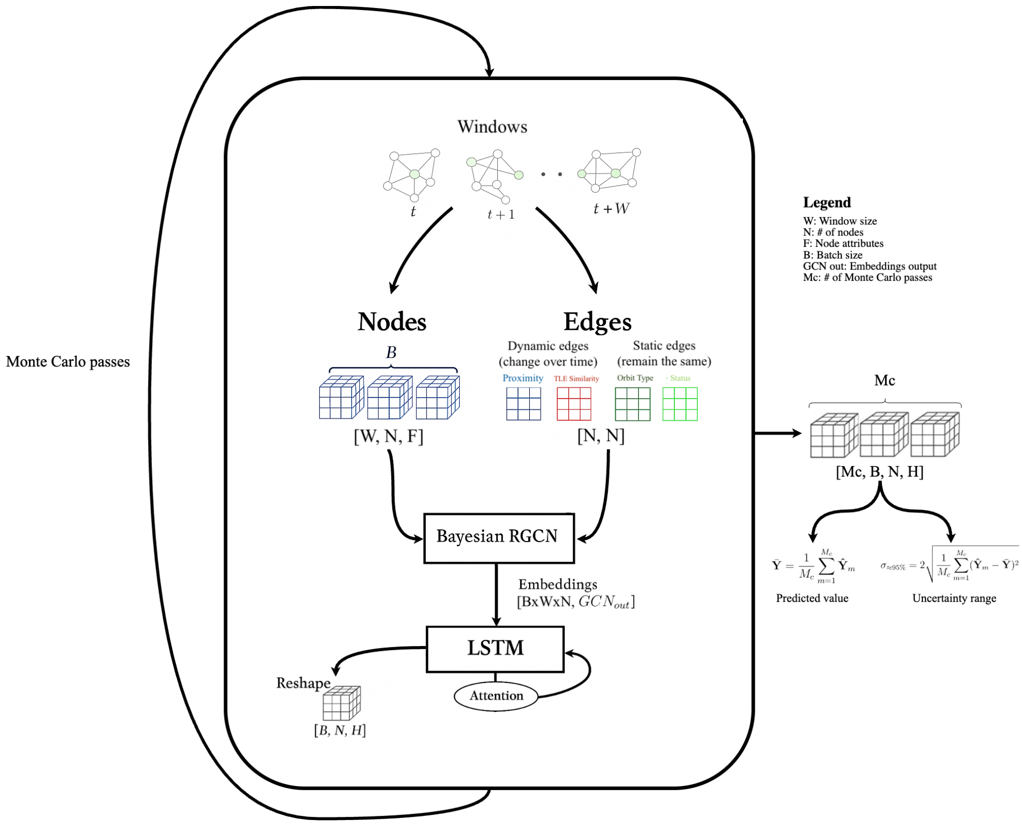

Figure 3. Bayesian Recursive Graph Convolutional Network Model (B-RGCN) architecture general description

- **Dynamic Data:** Satellite position and velocity vectors sampled every 10 minutes.

- **Orbital Data:** Daily downsampled TLE information (given that public updates are roughly three times per day, but not synchronized), from which we also compute the approximate ballistic coefficient.

- **Global Context:** Space weather information (Ap, Dst, and F10.7) is added at its native resolution.

- **Static and Categorical Data:** Such as weight or other similar fixed values.

This dataset, where each satellite is represented as a node enriched with both dynamic and static features, allows our model to integrate multiple data sources and build robust representations of inter-satellite relationships for forecasting. To prepare the data for training, we precomputed graph sequences using a sliding window of 36 time steps, capturing all three graph views (proximity, similarity, and operator) over that period. The model is trained to predict a future horizon of 18 steps, with a batch size of 64. The analysis covers the period from early 2021 to early 2025, with a chronological train-test split of 80%/20% to preserve temporal consistency.

RESULTS AND DISCUSSION

Our initial experiments, conducted on a limited dataset using only the SWARM and GRACE-FO2 satellites, show that our model achieves a validation root mean squared error (RMSE) of 37.84 when forecasting the next 18 time steps. This result is comparable to the RMSE of 36.5 obtained using the SGP4 propagator. All these results are presented in Figure 4. While this level of error is not yet competitive, it is important to emphasize that our model relies solely on a data-centric approach. No additional domain-specific filtering or handcrafted features were introduced. This performance emerges from a small fraction of the full satellite catalogue and demonstrates the potential of our architecture even when operating under sparse and partially incomplete data conditions. The model learns physical behavior implicitly, rather than depending on externally derived physics-informed constraints.

Another key strength of our model lies in its capacity to provide meaningful uncertainty estimations. As illustrated in Figure 4, the predicted uncertainty intervals cover the true values in over 85% of cases, with an average coverage of 83.2%. Notably, the model exhibits heightened uncertainty with increased geomagnetic activity. A comparison between predictions during quiet conditions and those during a G5 solar storm reveals a significant expansion of the confidence interval, from ap-

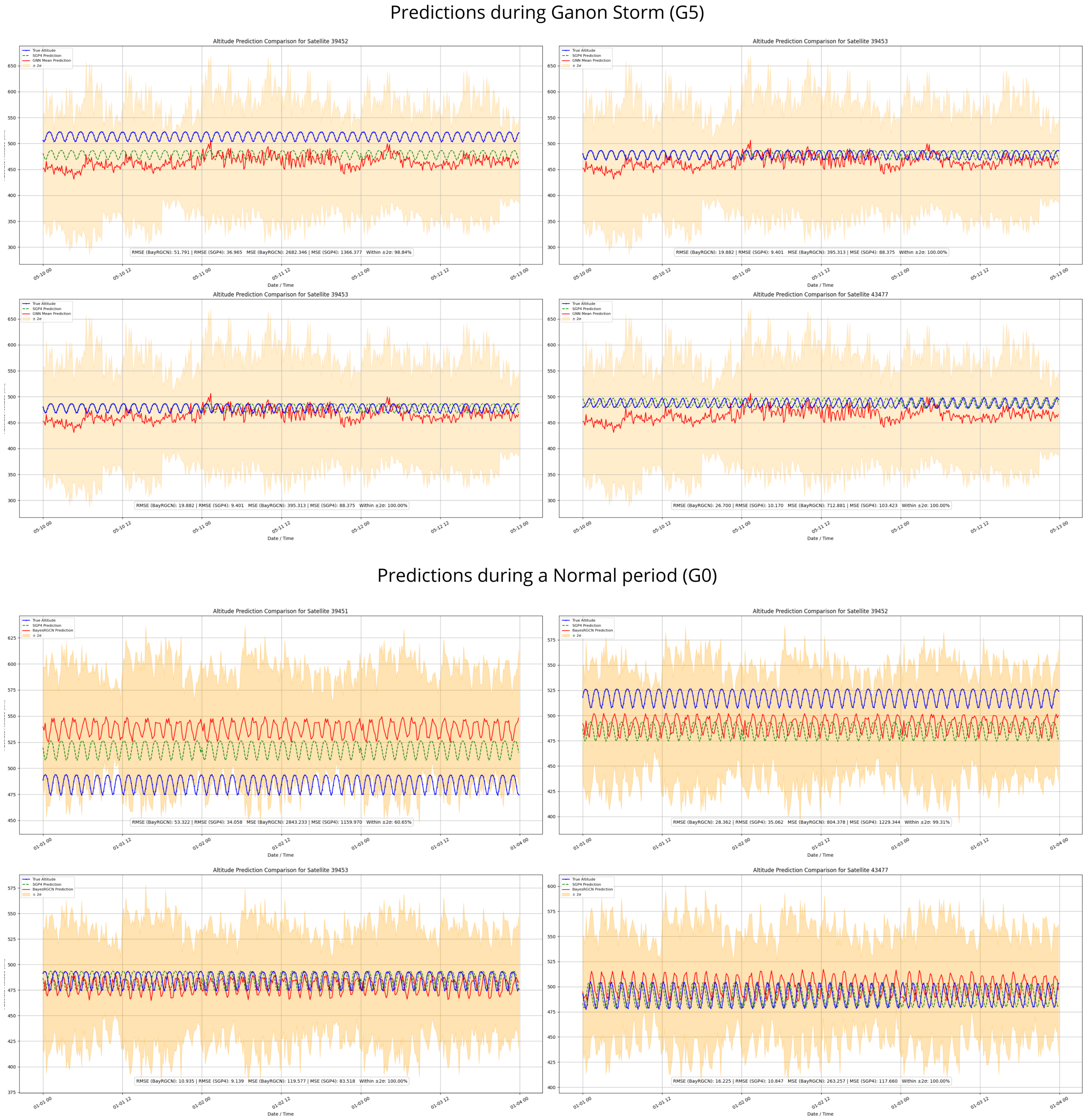

Figure 4. Altitude (km) comparison between model predictions and SGP4 propagation performance. The plots illustrate the performance during the Ganon Solar Storm (May 11–13) and a period of low geomagnetic activity. Note that the evaluation metrics are annotated at the bottom of each plot.

This suggests that the model is capable of detecting early signs of space weather disturbances and responding with increased caution in its predictions.

It is worth noting that we have not yet explored the effect of different input configurations, nor have we applied optimization strategies or data augmentation techniques. These directions are expected to significantly impact model performance and are planned for future experimentation. Another current limitation of the model is its relatively slow training performance, due to the sequential nature of temporal processing, which prevents parallelization beyond the embedding stage. However, this does not hinder inference performance: the model is capable of generating 3-hour forecasts for the entire graph in approximately 0.5 ms.

Given these preliminary results, we anticipate that scaling up the dataset to include a broader set of satellites, incorporating longer training sequences, optimizing hyperparameters, and testing alternative architectures will substantially improve performance. These enhancements could reduce prediction errors and potentially surpass the accuracy of traditional methods such as SGP4.

3. CONCLUSION

In conclusion, while this study presents a promising approach to the orbit determination problem, it remains in a preliminary stage and must address several challenges before becoming a robust solution for operational satellite orbit prediction. The current model demonstrates encouraging results, particularly in its ability to provide uncertainty estimates alongside predictions. However, further refinement is necessary, especially when scaling to larger satellite datasets and handling longer temporal sequences.

Graph Neural Networks offer advantages that extend beyond forecasting. They have demonstrated strong potential in anomaly detection tasks, which is particularly valuable for updating uncertain or missing states across multiple satellites, a challenge highlighted by Caldas and Soares [14]. Their capacity to model complex interdependencies and identify outliers or irregular behaviors within graph-structured data is a key strength, as noted by Wu et al. [30]. Additionally, GNNs can infer the state of nodes with missing data by leveraging information from neighboring nodes, an especially powerful feature given the common occurrence of temporal gaps in satellite telemetry due to data acquisition limitations. This dual capability for both forecasting and anomaly detection positions our framework as a versatile and scalable solution, supporting not only routine orbit prediction but also real-time satellite catalogue maintenance and updates.

REFERENCES

1. Associated Press, (2024). Severe Solar Storm May Stress Power Grids as U.S. Deals with Major Hurricanes.
2. Pultarova, T., (2024). May Solar Superstorm Caused Largest 'Mass Migration' of Satellites in History, *space.com*.
3. Emmert, J.T., (2015). Thermospheric Mass Density: A Review, *Adv. Space Res.*, 56.
4. Hapgood, M., Liu, H. & Lugaz, N., (2022). SpaceX—Sailing Close to the Space Weather?, *Space Weather*, 20(3), e2022SW003074.
5. Weiss, M. & Weiss, M., (2019). An Assessment of Threats to the American Power Grid, *Energy, Sustainability and Society*, 9(1), 1–9.
6. Parker, W.E. & Linares, R., (2024). Satellite Drag Analysis During the May 2024 Geomagnetic Storm, *arXiv preprint arXiv:2406.08617*.
7. Appell, D., (2024). May 2024 Super Geomagnetic Storm Challenges Current Space Weather Prediction Models, *phys.org*.
8. Adalian, D., (2024). Falcon 9 Return to Flight OK'd by FAA, *earthsky.org*.
9. Flores, R., Burhani, B.M. & Fantino, E., (2021). A Method for Accurate and Efficient Propagation of Satellite Orbits: A Case Study for a Molniya Orbit, *Alexandria Eng. J.*, 60(2), 2661–2676.
10. Paul, S.N., Sheridan, P.L., Licata, R.J. & Mehta, P.M., (2023). Stochastic Modeling of Physical Drag Coefficient – Its Impact on Orbit Prediction and Space Traffic Management, *Adv. Space Res.*, 72(4), 922–939.
11. Cazabonne, B. & Cefola, P.J., (2021). Towards Accurate Orbit Determination Using Semi-Analytical Satellite Theory, *AAS Paper*, 21-309.
12. Li, B. & Sang, J., (2020). Efficient and Accurate Error Propagation in the Semi-Analytic Orbit Dynamics System for Space Debris, *Adv. Space Res.*, 65(1), 285–296.
13. Peng, H. & Bai, X., (2020). Machine Learning Approach to Improve Satellite Orbit Prediction Accuracy Using Publicly Available Data, *J. Astronaut. Sci.*, 67(2), 762–793.
14. Caldas, F. & Soares, C., (2024). Machine Learning in Orbit Estimation: A Survey, *Acta Astronaut.*
15. Parker, T., Reed, M. & Johnson, R., (2024). Influences of Space Weather Forecasting Uncertainty on Satellite Conjunction Assessment.
16. Matsuo, T., Anderson, K. & Gibson, L., (2022). Uncertainty Quantification Techniques for Data-Driven Space Weather Modeling: Thermospheric Density Application.
17. Hypolite, D.B., Martinez, F. & Lee, S., (2018). Effects of Space Weather on the Ionosphere and LEO Satellites' Orbital Trajectory in Equatorial, Low and Middle Latitude.
18. Smith, J. & Doe, A., (2022). A Survey of Uncertainty Quantification in Machine Learning for Space Weather Prediction.
19. Space.com, (2024). Space Weather Can Make It Hard to Predict Satellite Trajectories. Here's Why That's a Problem.
20. Li, X., Zhang, D. & Chen, J., (2023). Thermospheric Density Predictions During Quiet Time and Geomagnetic Storm Using a Deep Evidential Model-Based Framework.
21. University of Colorado, Boulder, (2022). How Does Space Weather Impact Satellites and Debris in Low-Earth Orbit?
22. NOAA Space Weather Prediction Center, (n.d.). Satellite Drag.
23. Wikipedia, (2025). Kessler Syndrome.
24. Horne, R.B., Meredith, N.P. & Green, J.C., (2013). Space Weather Impacts on Satellites and Forecasting the Earth's Electron Radiation Belts with SPACECAST.

25. Kaggle, (2021). Discussion Post: Ventilator Pressure Prediction - Common Errors and Modeling Tips.

26. Hacibeyoglu, M., Arslan, A. & Kahramanli, S., (2011). Improving Classification Accuracy with Discretization on Data Sets Including Continuous Valued Features, *Ionosphere*, 34(351), 2.

27. Oguiza, I., (2023). *tsai – A State-of-the-Art Deep Learning Library for Time Series and Sequential Data*, GitHub.

28. Hu, J., Liang, Y., Fan, Z., Chen, H., Zheng, Y. & Zimmermann, R., (2023). Graph Neural Processes for Spatio-Temporal Extrapolation, In *Proc. 29th ACM SIGKDD Conf. on Knowledge Discovery and Data Mining*, pp. 752–763.

29. Vires Services, (n.d.). ESA Virtual Space Weather Modelling Centre.

30. Wu, L., Cui, P., Pei, J. & Zhao, L., (2022). *Graph Neural Networks: Foundations, Frontiers, and Applications*, Springer Singapore, Singapore, 725 pp.