SSC21-IX-02

Improved Orbital Propagator Integrated with SGP4 and Machine Learning

Zac Yung-Chun Liu, Scott Tarlow, Mohammad Akbar, Quentin Donnellan, Darek Senkow
Hypergiant Industries LLC
101 W 6th St #400 Austin TX 78701; (737) 808-4055
zac.liu@hypergiant.com

ABSTRACT

The current industry standard orbital propagator, the Simplified General Perturbation Model- 4 (SPG4), relies completely on physics-based orbital mechanics, can only provide accurate orbital predictions ~12 hours in advance. We developed a novel hybrid model, combining the SGP4 baseline with two machine learning estimators, autoencoder and random forest, in order to reduce the errors of the SGP4 propagator. The sources of errors in SGP4 propagators come from incomplete perturbation calculations and low-order of series expansions. The time-series nature of these error patterns are modeled by our machine learning estimators and then are used to make corrections to the SGP4 propagation, which result in more accurate orbit predictions. We tested our hybrid model on 3 satellite objects with the corresponding TLE (Two Line Element) data. The improvement on orbit prediction achieved 20-30% over the future 30 days period. The limitation of this hybrid approach is the requirement of at least 3 years of historical TLE data for the machine learning models, but could be overcome by creating synthetic orbital data from a similar space object. This hybrid model can be easily packaged into a software tool for space mission operation planning and facilitate mission autonomy.

1. INTRODUCTION

Accurate satellite orbit prediction is essential for mission operation, autonomy, and prioritization, as well as space situational awareness and collision avoidance.

The current industry standard to create orbital prediction is to implement the orbital propagation through the Simplified General Perturbation Model- 4 (SPG4) [1,2] with Two-Line Element (TLE) [3] data set. The TLE includes information about the satellite and its orbit, such as satellite number, orbit inclination, eccentricity, argument of perigee, derivatives of the mean motion, mean anomaly, drag (BSTAR), ballistic coefficient, ascending node, and revolution number. TLEs have been the sole public source of orbital observations in the past decades. The propagation of TLE needs to be done through the SGP4 software, which is the propagator specially adapted to TLE specifications. SGP4 is originally based on Brouwer's theory [4] of satellite motion perturbed by the Earth zonal harmonics.

Since SGP4 only considers the main perturbing effects and TLEs are mainly based on mean elements instead of osculating elements, the associated uncertainty causes the dramatic decrease in orbital prediction accuracy as the propagation horizon increases, for a low earth orbit (LEO) object, the SGP4 simulator could gain a distance error of 1-3 km for each day in the future [2,5], in other words, it can only provide accurate and reliable orbital predictions (< 1 km distance error) ~12 hours in advance in most practical cases, thus, significantly limits the period of validity of TLEs as well as the propagation horizon for the satellite operation planning.

To increase the precision of longer term orbital projections for essential planning payload tasks, mission autonomy, and collision analysis, we developed a novel hybrid model, combining the SGP4 baseline with two machine learning estimators, autoencoder and random forest, in order to reduce the errors of the SGP4 propagator. The source of errors in SGP4 propagator comes from the incomplete perturbation calculations and low-order of series expansions. The time-series nature of these error patterns are modeled by our machine learning estimators and then are used to make corrections to the SGP4 propagation, which result in more accurate orbit predictions.

Similar approaches [6,7,8] were used to improve the SGP4 propagations by modeling the orbital error patterns. Different from these hybrid models, instead of relying solely on the machine learning regression to model the propagation errors, here we implement a neural network based autoencoder [9] to learn the embedding of historical TLE propagations, which are treated as error waveforms. Using these waveforms to offset the propagation errors can potentially maintain the improvement of the orbital projection for the entire prediction time period up to 30 days.

The remaining part of the paper is organized as follows. In Section 2, the methodology of our modeling approach, the design of machine learning and target variables, and the TLE data used to test on our hybrid model are presented. In Section 3, the model results are presented and discussed. At the last section, conclusions, and future studies are provided.

2. METHODOLOGY

TLE data

We selected 3 objects in low earth orbit (LEO) and the corresponding TLEs to be tested in our hybrid model. These include a research CubeSat (QuakeSat by Stanford University, NORAD ID: 27845), a satellite payload (COSMOS 2098, NORAD ID: 20774), and a satellite debris (PEGASUS DEB debris, NORAD ID: 23975). Note that we constrain our testing object selection to those lack propulsive capability so we can focus on the natural orbital decay resulting from the external forces acting alone on the satellite.

The TLE data set was collected from the utilization of Space-Track API [10], which allows the access to its publicly available TLE data catalog. Each selected satellite object has more than 20 years of historical daily TLEs in the Space-Track database, which were used to calculate the distance error in the SGP4 propagation.

SGP4 setup and error calculation

SGP4 models predict the effect of perturbations caused by the Earth's shape, drag, radiation, and gravitation effects from other bodies such as the sun and moon. The complete documentation of all mathematical equations in SGP4 was published in 2004 [11]. In this work, we used the most updated code in Python programming language [12].

We input a TLE set of an orbital object at an initial time step $\mathbf{t}_0$ to SGP4 model that transforms the TLE into osculating orbital elements, which are composed of the estimation χ_0 position and velocity, where χ represents the set of six variables of position and velocity ($\mathbf{P}_x$, $\mathbf{P}_y$, $\mathbf{P}_z$, $\mathbf{V}_x$, $\mathbf{V}_y$, $\mathbf{V}_z$) that can be referred to canonical coordinate system. We then define a time period for SGP4 to propagate from the time step $\mathbf{t}_0$ to the time step $\mathbf{t}_n$ and obtain the estimation χ_n, which we called an ephemerid, representing the future position and velocity of the orbital object. In our work, ephemerides are taken with a sampling period of one minute.

Since there are daily TLEs available for our selected satellite objects, we can use the ephemerid into the future time step of roughly 24 hours as pseudo-observations, which we adopt as our ground truth baseline. For example, the ephemerid calculated by the TLE obtained on Oct 10th, 2020 ($\mathbf{t}_0$) in SGP4 is considered valid pseudo-observations through Oct 11th, 2020 ($\mathbf{t}_1$). To get the next time step's valid pseudo-observations through Oct 12th, 2020 ($\mathbf{t}_2$), the TLE obtained from Oct 11th ($\mathbf{t}_1$) is required. Any ephemerid estimated from TLEs beyond 24 hours has inherent distance and velocity errors from SGP4 model, which is calculated by subtracting the ephemerides of SGP4 prediction to pseudo-observations:

| SGP4 χ_n prediction – SGP4 χ_n pseudo-observations |

With the aim of modeling the error of SGP4 with respect to the pseudo-observations, the TLE is propagated with SGP4 for 30-day time period. Therefore, we calculate the distance error from the ephemerid estimated from $\mathbf{t}_0$ to $\mathbf{t}_{30}$ (n = 30 days) where we have 30 daily TLEs as well as 30-day orbital propagation. **Figure 1** demonstrates the satellite debris (NORAD 23975) distance error in *x*-axis over an 8-day time period. Note that this time series of the error shows a systematic pattern that repeats in every orbiter revolution; depending on each case, a trend, usually linear, can also exist.

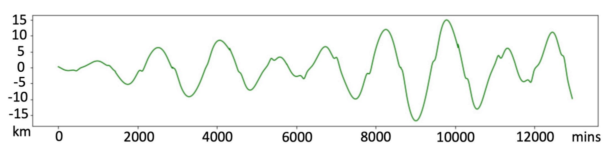

Figure 1. 8 days x-axis distance error of debris (NORAD 23975)

We then used the historical TLEs data that spans a full year to calculate the average distance error in each day out, from $\mathbf{t}_0$ to $\mathbf{t}_1$ (1-day out) up to $\mathbf{t}_{30}$ (30-days out). **Figure 2** is the average distance error for satellite debris 23975 from 1-day out to 30-day out, which shows the distance error was averaging 50 km for 15-day out and reaches > 200 km of distance error for 30-day out. The TLEs used to calculate the average distance error displayed in **Figure 2** were from June 2019- June 2020. We took the distance error curve as the benchmark and aimed to develop our machine learning approach to beat this benchmark.

Machine Learning framework

The machine learning method is designed to extract knowledge from large dataset [13], which is similar to human cognition in learning from past experience to predict future events. Supervised learning structure has been developed to learn a function from pairs of input and its output. Several studies [7,8,14,15] used supervised machine learning approach to improve orbital prediction accuracy based on historical measurements with a regression scheme.

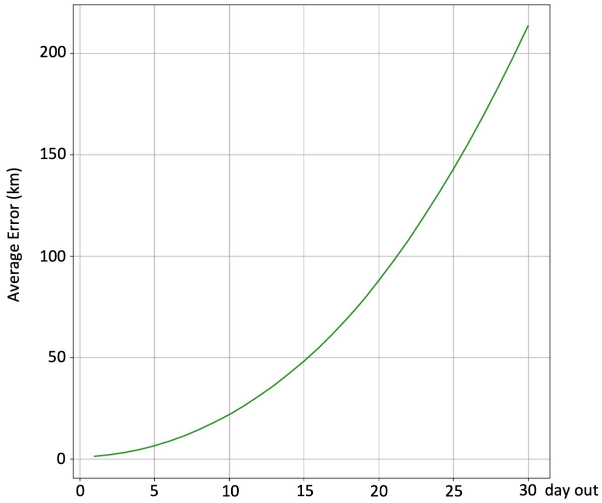

Figure 2. Average distance error 1 day out to 30 day out for debris (NORAD 23975)

In this work, we developed a novel hybrid model, combining the SGP4 baseline with two machine learning estimators: autoencoder and random forest, in order to reduce the distance errors of the SGP4 propagator.

First, an autoencoder [16] is a type of artificial neural networks that can learn efficient data codings in an unsupervised manner. The aim of an autoencoder is to learn a representation (encoding) for a set of data. In our work, we developed an autoencoder to encode the distance error to the latent space. We treat the 30-day time-series distance error (**Figure 1, 2**) as a representation of a waveform, which was used as the input in the autoencoder. It has an internal (hidden) layer that describes a vector used to represent the input, and it is constituted by two main parts: an encoder that maps the input into a embedding vector, and a decoder that maps the embedding vector to a reconstruction of the input, e.g. distance error.

Second, random forest is a tree-based algorithm that generates unpruned classification trees to predict a response and it also implements bootstrapping samples of the data and randomized subsets of predictors [17]. In this work, we used a random forest model to learn the trend of historical embedding vectors in each position axis (*x, y, z*) and predict the embedding vector for the next time step.

Next, the decoder was used to decode the predicted embedding vector back to the distance error waveform, which then was used to offset the distance errors in a 30-day time period. **Figure 3** shows an example of the predicted waveform (orange color) that was used to offset the SGP4 distance error (blue color, *x*-axis), for the satellite debris 23975 over the 30-day time period. The idea is that the distance error waveform (**Figure 1,3**) for next time step likely exists in the historical data, because the satellite orbit tends to repeat itself so periodic error patterns exist in the data, which can be fully utilized by our machine learning framework.

Note that we modeled all 3 axes of satellite positions (*x, y, z*), thus, we had 3 autoencoders and 3 random forest models for each modeled object. The historical TLE data used in the model are from June 2017-June 2020, which included 3-year of daily TLEs for autoencoders to learn the embedding representation and for random forest models to learn the time series trend of the embedding vectors.

We implemented autoencoder model and training in the Python development environment, with the Keras package [18], which provides a high-level application programming interface to access Google's deep learning framework, TensorFlow [19] as well as the random forest model from scikit-learn library [20].

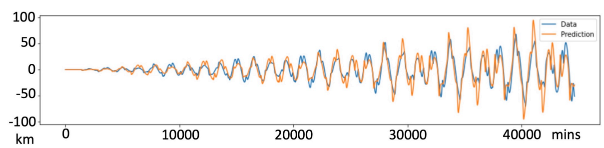

Figure 3. An example of predicted waveform to offset SGP4 distance error for debris (NORAD 23975) 30-day x-axis

3. RESULTS

Following our machine learning framework and TLEs data we collected from Space-Track, we modeled the SGP4 propagator errors and produced improved orbital predictions for 3 LEO objects (NORAD ID: 27845, 20774, and 23975). The TLEs used in this analysis were all from the time period of June 2017- June 2020. The results are presented in **Table 1** and **Table 2**, summarizing the 3 selected satellite objects' improved distance error and improvement percentage for 15-day out and 30-day out, respectively.

Figure 4 displays the improved average distance error for satellite debris 23975 from 1-day out to 30-day out, which shows the improved orbital prediction has distance error reduced from 50 km to 35 km (30% improvement) for 15-day out and from 200 km to 150 km (25% improvement) for 30-day out.

In sum, these results clearly suggest that our hybrid model consistently beat the SGP4 benchmark in the 30-day orbital propagation.

Table 1: Model results 15-day out

Satellite object (NORAD ID)	15-day out distance error	Improved 15-day out distance error	Improvement %
27845	2 km	1.5 km	25%
20774	12 km	8 km	33%
23975	50 km	35 km	30%

Table 2: Model results 30-day out

Satellite object (NORAD ID)	30-day out distance error	Improved 30-day out distance error	Improvement %
27845	5 km	4 km	25%
20774	70 km	45 km	35%
23975	200 km	150 km	25%

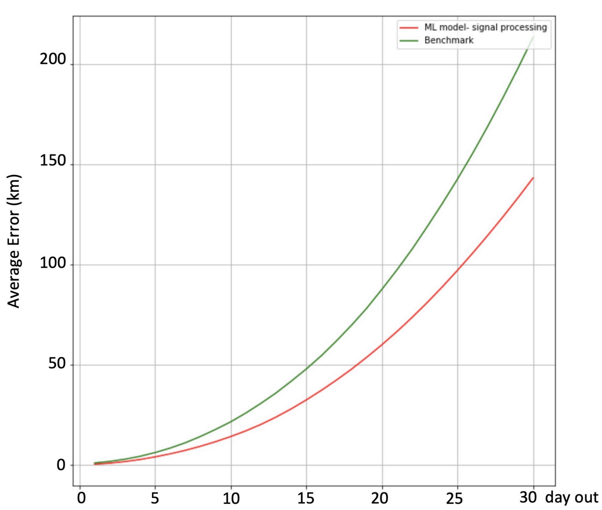

Figure 4. Improved average distance error of debris (NORAD 23975)

4. CONCLUSION AND DISCUSSION

In this study, we show that the time-series orbital error patterns can be modeled by our machine learning estimators, which then produce an improved orbital predictions that can be used to make corrections to the SGP4 propagation. This results in more accurate orbit projections. We tested our hybrid model on 3 LEO satellite objects with the corresponding 3-year TLE data (June 2017- June 2020) for the autoencoder and random forest models training. The improvement on orbit prediction achieved 20-30% in average over the future 30 days period.

Since the machine learning models rely on learning from large amount of historical TLEs data, in our initial analysis, we found that at least 3 years of TLEs data is required to produce meaningfully better orbital projections than SGP4 propagator since our machine learning models need to learn from the sufficiently large amount of examples of the periodic orbit error patterns. However, this limitation could be overcome by creating synthetic orbital data from a similar space object in orbit, which is in need of more explorations and will be our next step of development.

This hybrid model can be easily packaged into a software tool for space mission planning and facilitate mission autonomy, including the following general use cases in satellite operation: ground station bandwidth usage forecasts, better AOS (acquisition of signal) and LOS (loss of signal) prediction, mission planning prioritization and optimization, measuring ground station reliability, improved trajectory designing, and collision risk analysis.

Acknowledgments

The authors thank Brent Horine, Chris Tackle, Adrian Glover, Marv LeBlanc for their suggestions and comments for this project.

References

1. Hoots, F.R. and Roehrich, R.L. “Models for propagation of the NORAD element sets, Spacetrack Report #3” *U.S. Air Force Aerospace Defense Command, Colorado Springs, CO, USA*. 1980.
2. Vallado, David, et al. "Revisiting spacetrack report# 3." *AIAA/AAS Astrodynamics Specialist Conference and Exhibit*. 2006.
3. Vallado, David A., and Paul J. Cefola. "Two-line element sets–practice and use." *63rd International Astronautical Congress, Naples, Italy*. 2012.
4. Brouwer, Dirk. Solution of the problem of artificial satellite theory without drag. YALE UNIV NEW HAVEN CT NEW HAVEN United States, 1959.
5. Dong, Wei, and Zhao Chang-yin. "An accuracy analysis of the SGP4/SDP4 model." *Chinese Astronomy and Astrophysics* 34.1 (2010): 69-76.
6. San-Juan, Juan Félix, et al. "Hybrid SGP4 orbit propagator." *Acta Astronautica* 137 (2017): 254-260.
7. San-Juana, Juan F., et al. "Hybrid SGP4 propagator based on machine-learning techniques applied to GALILEO-type orbits." (2018).

8. Peng, Hao, and Xiaoli Bai. "Comparative evaluation of three machine learning algorithms on improving orbit prediction accuracy." *Astrodynamics* 3.4 (2019): 325-343.

9. Kramer, Mark A. "Nonlinear principal component analysis using autoassociative neural networks." *AIChE journal* 37.2 (1991): 233-243.

10. Hoddinott, Philip James Bernard. *Tracking of Space Debris from Publicly Available Data*. Rensselaer Polytechnic Institute, 2018.

11. Hoots, Felix R., Paul W. Schumacher Jr, and Robert A. Glover. "History of analytical orbit modeling in the US space surveillance system." *Journal of Guidance, Control, and Dynamics* 27.2 (2004): 174-185.

12. https://pypi.org/project/sgp4/1.4/

13. Abu-Mostafa, Yaser S., Malik Magdon-Ismail, and Hsuan-Tien Lin. *Learning from data*. Vol. 4. New York, NY, USA:: AMLBook, 2012.

14. Peng, Hao, and Xiaoli Bai. "Improving orbit prediction accuracy through supervised machine learning." *Advances in Space Research* 61.10 (2018): 2628-2646.

15. Peng, H., and X. Bai. "Limits of Machine Learning Approach on Improving Orbit Prediction Accuracy using Support Vector Machine." *Advanced Maui Optical and Space Surveillance (AMOS) Technologies Conference*. 2017.

16. Tschannen, Michael, Olivier Bachem, and Mario Lucic. "Recent advances in autoencoder-based representation learning." *arXiv preprint arXiv:1812.05069* (2018).

17. Breiman, Leo. "Random forests." *Machine learning* 45.1 (2001): 5-32.

18. Chollet, François. "Keras: Deep learning library for theano and tensorflow." *URL: https://keras. io/k* 7.8 (2015): T1.

19. Abadi, Martín, et al. "Tensorflow: A system for large-scale machine learning." *12th {USENIX} symposium on operating systems design and implementation ({OSDI} 16)*. 2016.

20. Pedregosa, Fabian, et al. "Scikit-learn: Machine learning in Python." *the Journal of machine Learning research* 12 (2011): 2825-2830.