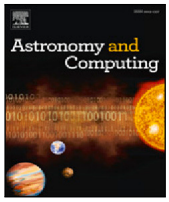
Astronomy and Computing 46 (2024) 100782

Contents lists available at ScienceDirect

Astronomy and Computing

journal homepage: www.elsevier.com/locate/ascom



Full length article

Data-driven satellite orbit prediction using two-line elements

M. Thammawichai [a], T. Luangwilai [b,*]

[a] Electrical Engineering Department, Navaminda Kasatriyadhiraj Royal Air Force Academy, Bangkok, 10220, Thailand
[b] Mathematics Department, Navaminda Kasatriyadhiraj Royal Air Force Academy, Bangkok, 10220, Thailand

ARTICLE INFO

Dataset link: https://celestrak.org

Keywords:
Data-driven
Nonlinear programming
Orbit prediction
Two-line elements

ABSTRACT

Orbit prediction is crucial for space situational awareness operations. Low earth orbit satellites are subjected to external forces such as atmospheric drag, radiation, and gravity. However, the well-known Kepler propagation model ignores these external forces. The simplified perturbation model included only the main external forces. In this study, a nonlinear programming model for orbit prediction using public two-line elements (TLE) is proposed. It has been proven that our models exhibit better performance than the standard Kepler and SPG4 models in terms of orbit prediction accuracy. Moreover, the proposed models were simple, computationally effective, and robust to disturbances. The sensitivity analysis indicates that a right ascension of the ascending node, a perigee argument, and a mean anomaly of the orbital elements are the most sensitive parameters in our models. The results also revealed that our method can be generalized to any low-earth orbit satellite with adequate data.

1. Introduction

The number of satellites has increased rapidly in recent decades because the cost of manufacturing and launching satellites into orbit has substantially decreased, especially those in low-earth orbit (LEO) (McDowell, 2020). The LEO satellite is usually smaller at lower altitudes and shorter orbit periods (Wertz et al., 2011). The lifespan was also shorter than that of higher orbits. A shorter lifespan can be seen as a positive point, as one may test and experiment with technology or equipment at a cheaper cost. The LEO satellites can also be used to demonstrate new technologies. Despite their small size, LEO satellites can still perform typical satellite tasks such as observation, surveillance, and data relay.

In low-earth orbit, satellites are affected by various external forces more than those orbiting in deep space. There are many types of external forces, such as atmospheric drag, gravitational force, solar flares, and radiation force. Therefore, it is necessary to precisely predict the orbit of a satellite in low-earth orbit. The ability to accurately predict satellite orbit is crucial because it corresponds to space situational awareness (SSA) operations such as collision avoidance, observation tracking, satellite control, communication planning, and payload operations (Wen et al., 2020; Zeng et al., 2021). Collisions among satellites are becoming increasingly concerning because of the rapidly growing number of space objects (Marshall, 2011; McDowell, 2020). These collisions were primarily due to inaccurate orbit predictions.

Several methods have been proposed to solve the satellite orbit determination and prediction problems. The simplest model for orbit prediction is the Kepler propagation model. This model considers the conservation of sweeping out areas of the orbital plane for an interval of time by orbiting objects (Wertz et al., 2011). The second most common model is simplified general perturbation (SGP) (Vallado and Crawford, 2008). This model considers the equation of motion for a small object attracted by a large spheroid. This model also cooperates with perturbation forces, which improves the accuracy of LEO satellites over Kepler propagation.

In addition to the two standard orbit propagation routines (Kepler and SGP), intensive work has been conducted on orbit prediction and orbit determination using machine learning. Examples of such studies are presented in Barton and McLaughlin (2018) and Peng and Bai (2018, 2020). Peng and Bai (2020) applied a support vector machine (SVM) technique to improve satellite orbit accuracy. Li et al. (2020) modeled a pattern of debris orbit prediction errors using machine learning to improve performance. To determine the orbit of GEO spacecraft, Jiang (2021) proposed a new machine-learning method called distribution regression. Zhai et al. (2022) demonstrated a novel model based on extreme gradient boosting and principal component analysis to improve the accuracy of orbit prediction. However, the machine-learning approach requires an adequate training dataset to obtain satisfactory results. Furthermore, the trained model cannot predict far into the future (Peng and Bai, 2017).

Another approach is based on dynamic modelings. An example is the work of Levit and Marshall (2011), in which a high-precision

* Corresponding author.
E-mail address: thiansiri.L@gmail.com (T. Luangwilai).

https://doi.org/10.1016/j.ascom.2023.100782
Received 13 September 2023; Accepted 1 December 2023
Available online 6 December 2023

numerical propagator was used to fit an orbit as state vectors from a two-line element (TLE). Then Bennett et al. (2012) improved the model further by improving the bias terms of the TLEs. Elhag et al. (2013) employed an unscented Kalman filter (UKF) in conjunction with TLE data to estimate the satellite orbit. Goh and Low (2018) proposed a finite difference based on an extended Kalman filter (EKF) to estimate a satellite's position. Muldoon et al. (2009) relied on the data-driven modeling technique by using the TLEs data as input for predicting debris motion. Nonetheless, the accuracy of the prediction depends on the accuracy of the model.

In this study, we propose a nonlinear programming (NLP) model to estimate the six orbital elements. Specifically, we investigate and propose the process of using TLE data as an observation of real satellite positions, as in the study of Marshall (2011). Then, nonlinear programming is applied to develop the orbit prediction model, which is based on the publicly available TLE. The prediction accuracy was compared with that of the test TLE dataset. The performance of our proposed model was validated against standard orbit propagation models, that is, the Kepler and SPG4 models.

The contributions of this work are summarized as follows:

- Our proposed method is simple and computationally efficient compared to the machine learning approach.
- NLP models are robust to external disturbances such as atmospheric drag, gravitational force, and solar flare because we take in the new data so that changes in the dynamics of the satellite can be captured.
- The validation of our models illustrates the superior performance of the Kepler and SPG4 models.
- The sensitivity analysis is performed to study the robustness of our models.
- The results also show that our method has the potential for generalization to different LEO satellites.

The remainder of this paper is organized as follows. In Section 2, background knowledge of satellite orbit is described. The proposed methodology is presented in Section 3. In Section 4, the performance of our models is analyzed and compared with that of typical models. In Section 5, a sensitivity analysis and discussions are provided to measure the robustness of our models. Conclusions and future work are presented in Section 6

2. Background

In this section, relevant knowledge of satellite orbits is discussed. It includes two-line elements (TLE), satellite coordinate systems, and standard satellite propagation models. The basics of our proposed model of satellite prediction are also discussed such as nonlinear programming for parameter estimation and data processing and a comparison of prediction model performance.

2.1. Two line elements (TLE)

The Two-Line Element sets (TLEs) are the text format of the orbit and position of objects orbiting the earth. This is considered a standard and fundamental data sets for space situational awareness (SSA). It can also be used for satellite monitoring and other analytical activities (Liu et al., 2021). TLEs are created by the United States Space Force by tracking space objects in Earth's orbit. These data can be obtained from many space tracking websites, such as https://celestrak.org (Vallado et al., 2013).

In this study, the TLE data are assumed to be real observations of the satellite locations, that is, pseudo-observations, as in Pearlman et al. (2002). The TLE sets consisted of a two-line text-data format. The first line contains nine data fields, that is, the satellite name, international designator, epoch year and Julian day fraction, 1st derivative of mean motion, 2nd derivative of mean motion, drag term, ephemeris type, element number, and check sum. The second line consists of nine data fields, that is, the satellite number, inclination, right ascension of the ascending node, eccentricity, argument of perigee, mean motion, mean anomaly, and revolution number at the epoch and check sum. An example of TLE of an international space station and its fields is shown in Fig. 1.

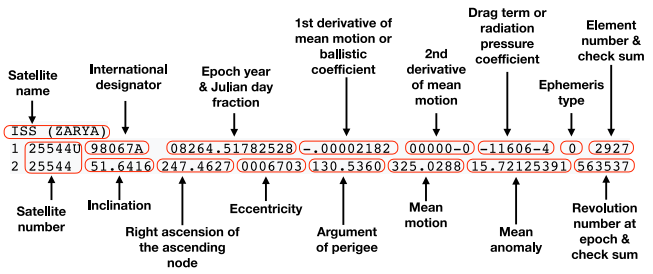


Fig. 1. The example of Two Line Element (TLE) of the international space station.

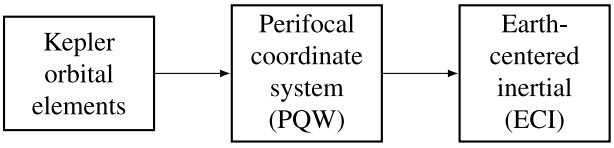


Fig. 2. The reference frames.

2.2. Orbital coordinate systems

The first step into studying objects orbiting Earth is to understand the reference frames. In this study, we mainly used three types of reference frames: Kepler orbital coordinate, perifocal coordinate (PQW), and Earth-centered inertial coordinate (ECI). Because the TLE data describe the satellite's position in Kepler's six orbital elements, we first discuss this reference frame. The satellite position coordinate is then transformed into a perifocal coordinate system (PQW), which is on the same orbital plane as the six orbital elements. However, the satellite movement in PQW is described in the vector form of position r and velocity v. To compare the accuracy of the prediction model, the satellite positions are transformed further to Earth-centered inertial coordinates (ECI) as shown in Fig. 2.

2.2.1. Kepler orbital elements

In astronomy, one of the most common ways of describing a two-body orbit is using Kepler orbital elements. The Kepler orbital elements are described in Wertz et al. (2011). The orbit can be described using the following six parameters: The first two parameters, which describe the orbit size and shape, are the semi-major axis a and eccentricity e.

The value of the semi-major axis a can be calculated from the value of the mean motion n. The mean motion n is the number of revolutions per day. This value can be converted into radians per second. Then, the value of the semi-major axis a is described as follows, where μ is a gravitational constant.

$$a = \mu^{1/3}/(2n\pi^{2/3}/86400) \tag{1}$$

Another three parameters that describe the orientation of the orbit plane are the perigee argument ω, inclination i, and right ascension of the ascending node (RAAN) Ω. The last parameter is the true anomaly ν, which defines the actual satellite position in the orbital plane at the epoch time. Sometimes, the mean anomaly M is used instead of the true anomaly ν. The value of the mean anomaly M can be calculated from the value of the eccentric anomaly E and true anomaly ν as follows:

$$E = 2tan^{-1}\left(\sqrt{\frac{1-e}{1+e}}tan\left(\frac{\nu}{2}\right)\right) \tag{2}$$

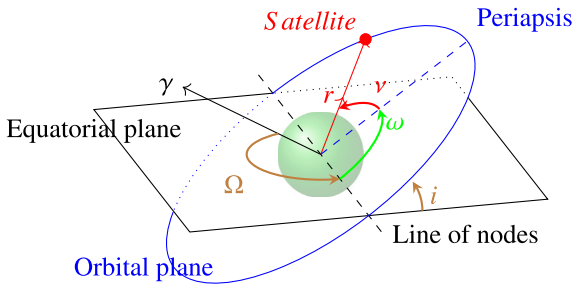


Fig. 3. Orbital elements or Keplerian elements.

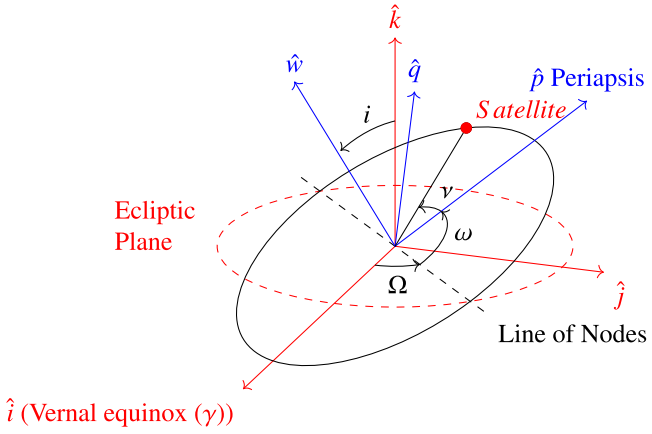


Fig. 4. Perifocal coordinate (PQW) and Earth-centered inertial (ECI).

$$M = E - esin(E). \tag{3}$$

These parameters are illustrated in Fig. 3.

2.2.2. Perifocal coordinate (PQW) system

The Perifocal coordinate (PQW) system is another common orbit reference frame for small objects that orbit large objects. The position of a satellite in this coordinate system is described by two sets of vectors: the position and velocity vectors. This coordinate system has three axes. The $\hat{p}$ and the $\hat{q}$ axes lie on the orbital plane. The $\hat{p}$ axis points to the periapsis, and the $\hat{q}$ axis is orthogonal to the $\hat{p}$ axis. The $\hat{w}$ axis is orthogonal to the orbital plane, as shown in Fig. 4. The vectors $\vec{r}_{PQW}$ and $\vec{v}_{PQW}$ of this coordinate system can be computed from Eqs. (4) and (5) as follows, where ν is a true anomaly and μ is a gravitational constant. Parameter p is the semi-lactus rectum of the orbital plane.

$$\vec{r}_{PQW} = \begin{bmatrix} rcos(\nu) \\ rsin(\nu) \\ 0 \end{bmatrix} \tag{4}$$

$$\begin{aligned} \vec{v}_{PQW} &= \begin{bmatrix} cos(-\nu) & sin(-\nu) & 0 \\ -sin(-\nu) & cos(-\nu) & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \dot{r} \\ r\dot{\nu} \\ 0 \end{bmatrix} \\ &= \begin{bmatrix} \dot{r}cos(\nu) & -r\dot{\nu}sin(\nu) \\ rsin(\nu) & r\dot{\nu}cos(\nu) \\ 0 & 0 \end{bmatrix} \end{aligned} \tag{5}$$

where

$$r = \frac{p}{1 + ecos(\nu)},$$

$$\dot{r} = \sqrt{\frac{\mu}{p}}(esin(\nu)),$$

$$r\dot{\nu} = \sqrt{\frac{\mu}{p}}(1 + ecos(\nu)).$$

2.2.3. Earth-centered inertial coordinate system (ECI)

The Earth-centered inertial coordinate system (ECI) is another common coordinate system. This reference frame has $\hat{i}$ and $\hat{j}$ axes lying on the same plane as Earth's equatorial plane. The $\hat{i}$ axis is orthogonal to the $\hat{j}$ axis and points to the vernal equinox γ. Finally, the $\hat{k}$ axis is orthogonal to the plane, as can be seen in Fig. 4. The position and velocity vectors in the Earth-centered inertial (ECI) coordinate can be obtained by rotating the perifocal coordinate (PQW) three times. Specifically, rotate by an Ω degree about the z-axis, an i degree about the x-axis, and an ω degree about the z-axis, as shown in Eq. (6).

$$R_{\text{PQW}\to\text{ECI}} = \vec{r}_{\text{ECI}} = R_3(\Omega)R_1(i)R_3(\omega)\vec{r}_{PQW} \tag{6}$$

where

$$R_3(\Omega) = \begin{bmatrix} cos(\Omega) & sin(\Omega) & 0 \\ -sin(\Omega) & cos(\Omega) & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$R_1(i) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & cos(i) & sin(i) \\ 0 & -sin(i) & cos(i) \end{bmatrix}$$

$$R_3(\omega) = \begin{bmatrix} cos(\omega) & sin(\omega) & 0 \\ -sin(\omega) & cos(\omega) & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

2.3. Standard propagation models

2.3.1. Kepler propagation

Kepler propagation is the simplest model for predicting the future position of objects orbiting Earth (Wertz et al., 2011). In this model, it is assumed that there is no perturbation force or external force acting on the objects or satellites. All orbit elements can be determined from a single observation at time t_0. Six orbital elements were used to determine orbital shape, size and orientation. Subsequently, the value of true anomaly ν was used to identify the satellite location at time t_0. Finally, the position is iterated over time until the desired final time t_f. The Kepler propagation model is illustrated in Algorithm 1.

Algorithm 1 Kepler propagation model

Input: Orbital elements at time t_0
Output: Predicted satellite position at time t_f

1: Find a mean anomaly (M) from a true anomaly (ν) at time (t_0)
2: **for** i = 1 to t_f **do**
3: $M(t_i + \Delta t) = M(t_i) + n(t_i - t_{i-1})$
4: **end for**
5: Find a true anomaly (ν) at time t_f using the mean anomaly (M) at time t_f.
$\nu = 2tan^{-1}\left(\sqrt{\frac{1+e}{1-e}}tan\left(\frac{E}{2}\right)\right)$

2.3.2. Simplified general perturbation 4 (SGP4)

Simplified general perturbation 4 (SGP4) is another standard model prediction for satellites and objects orbiting the earth. It has been widely used with TLE as the input data. This model was first published in 1965 (Lane, 1965; Lane and Cranford, 1969). This model is used to calculate the orbital space or orbital object relative to Earth-centered inertial (ECI) coordinates. The model is based on Brouwer's gravity solution (Brouwer, 1959) and the Lane atmospheric model (Lyddane, 1963). The propagation model considered the main perturbation effect, which improved the accuracy of the Kepler model. However, this model is not effective for other secondary forces.

This model has gained popularity over the years. Therefore, there are many improved versions of SGP4 written in many computer languages. Examples and details of this routine, code, and test cases can be found in Vallado and Crawford (2008).

Table 1
Satellite details.

NORAD ID	Perigee	Apogee	Inclination	Period	Semi major axis	Launch date
24793	782.2 km	785.4 km	86.4°	100.4 min	7154 km	May 5, 1997
27944	680.3 km	699.3 km	98.2°	98.4 min	7060 km	Sep 27, 2003
35871	822.9 km	829.6 km	98.6°	101.3 min	7197 km	Sep 17, 2009
39452	472.9 km	476.4 km	87.4°	93.9 min	6845 km	Nov 22, 2013
40025	595.6 km	610.4 km	98.0°	96.6 min	6973 km	Jun 19, 2014

3. Methodology

This section explains details on our problem formulation and data preprocessing method.

3.1. Nonlinear programming for parameter estimation

The nonlinear programming (NLP) problem (Avriel, 2003; Bazaraa et al., 2013) is an optimization problem in which the objective function and/or constraints are nonlinear. Generally, an optimization problem aims to select n decision variables $x_1, x_2, \dots, x_n$ from a feasible solution space $\mathcal{R}^n$ to optimize (minimize or maximize) the given objective function $f(x_1, x_2, \dots, x_n)$. Let n, m, and p be the positive integers. Let $X \subseteq \mathcal{R}^n$, and let f, g_i, h_j be functions on X for $i \in \{1, \dots, m\}$ and $j \in \{1, \dots, p\}$, where at least one of f, g_i, and h_j is nonlinear. Thus, in minimization form, general nonlinear programming is stated as follows:

$$\begin{aligned} \underset{x \in X}{\text{minimize}} \quad & f(x) \\ \text{subject to} \quad & g_i(x) \le 0 \quad \forall i \in \{1, \dots, m\} \\ & h_j(x) = 0 \quad \forall j \in \{1, \dots, p\}. \end{aligned} \tag{7}$$

where g_i and h_j are inequality and equality constraints, respectively. A linear program is a special case of the above problem, where the objective function and constraints are linear.

Next, we provide details on how to apply nonlinear programming to a parameter estimation (curve fitting) problem. Suppose we have a set of m measurements $\{(t_1, y_1), (t_2, y_2), \dots, (t_m, y_m)\}$ and a nonlinear model function $\hat{y}(t, x)$ of an independent variable t and a vector of n parameters x. Let u and ℓ be upper and lower bounds on x, respectively. Then, we can define our curve fitting problem as an NLP with the objective of minimizing the sum of the weighted squared errors between the data y_i and the value of the model function $\hat{y}(t, x)$, that is,

$$\begin{aligned} \underset{x \in \mathcal{R}^n}{\text{minimize}} \quad & f(x) := \sum_{i=1}^{m} (y(t_i) - \hat{y}(t_i, x))^2 \\ \text{subject to} \quad & g_1(x) := x - u \le 0 \\ & g_2(x) := \ell - x \le 0 \end{aligned} \tag{8}$$

Several methods have been proposed to address this problem. The most widely used algorithms are the generalized reduced gradient (GRG) and sequential quadratic programming (SQP). When the NLP problem has a convex objective function, the optimal solution is obtained. If the problem is non-convex, typically, only a local optimal solution is found. It should be noted that the nonlinear model function $\hat{y}$ represents a candidate function for measurement data. For example, $\hat{y}$ may consist of a constant, polynomial, or sinusoidal function.

3.2. Data preprocessing

Data preprocessing refers to the steps required to transform or encode data for a model to be accurate and precise in its predictions. In our work, data cleaning was performed by smoothing noisy data, resolving inconsistencies, and removing outliers. Moreover, by examining several datasets, the graphs of RAAN Ω, the argument of perigee ω and the mean anomaly M often behaved like sawtooth waves ranging from 0° to 360°. To obtain a better fitted model, we performed a data transform on these orbital elements as specified in Algorithm 2.

Algorithm 2 Unwinding algorithm

Input: Dataset, $y(:)$
Output: Transformed dataset, $\tilde{y}(:)$

Find break points of the dataset $p = [p_1, \dots, p_n]$
Find the slope of the dataset s
Initialize the new dataset $\tilde{y}(:) = y(:)$
for i = 1 to length(p)-2 **do**
  if $(s > 0)$ **then**
    $\tilde{y}(p(i+1)+1 : p(i+2)) = y(p(i+1)+1 : p(i+2)) + i \cdot 360$
  else
    $\tilde{y}(p(i+1)+1 : p(i+2)) = y(p(i+1)+1 : p(i+2)) - i \cdot 360$
  end if
end for

The purpose of Algorithm 2 is to transform (unwind) a sawtooth signal into a straight line based on the arrival time. Specifically, if the dataset has an increasing waveform, the signal before the breaking point leads to the signal after the breaking point by 360°. Conversely, the signal before the breaking point lagged the signal after the breaking point by 360° in the case of a decreasing waveform. It should be noted that the transformed data require modulo operation at the end to be in the normal range from 0° to 360°.

3.3. The performance comparison

In this research, the accuracy of the proposed algorithm was compared with the pseudo-real locations of the TLE data. The absolute errors from the proposed algorithm and the TLE data were calculated from the ECI coordinates. The performance of the proposed algorithm was then compared with the errors of the Kepler and SGP4 routines.

4. Numerical results

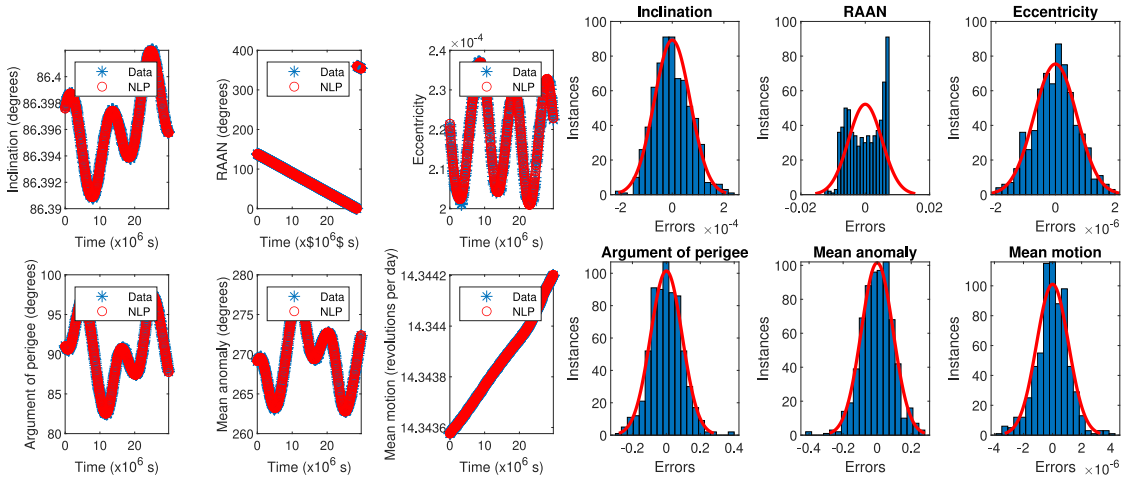
4.1. Satellite datasets

For this study, we requested satellite TLE datasets from the CelesTrak website (Kelso, 2022). The details of our tested satellites are shown in Table 1. For each satellite, we split the dataset into training and test sets. The training set consists of approximately 600–800 data points. The test set contains 30 to 40 data points, which is approximately 10 days of orbit prediction.

4.2. Nonlinear programming model

In this work, the MATLAB (R2022a) implementation of the function fit is used (MATLAB, 2022) to solve the NLP. Depending on the characteristics of the dataset, we choose a model function expressed in the form of a polynomial function:

$$\hat{y}(t, x) = a_0 + a_1 t + a_2 t^2 + \cdots + a_p t^p \tag{9}$$



(a) NLP fitted models (b) Errors Plots

Fig. 5. NORAD ID: 24793.

where $x = [a_0, a_1, \ldots, a_p]$, p is the degree of the polynomial, $a_0, a_1, \ldots, a_p$ are polynomial coefficients, t is the input time (s), and in the form of a Fourier series:

$$\hat{y}(t, x) = a_0 + \sum_{i=1}^{k} a_i \cos(i f t) + b_i \sin(i f t) \tag{10}$$

where $x = [a_0, a_1, \ldots, a_k, b_1, \ldots, b_k, f]$, t is the input time (s), a_0 is the constant term, k is the number of harmonic terms of the series, f is the fundamental frequency of the signal, $a_1, a_2, \ldots, a_k$ are coefficients of the cosine terms and $b_1, b_2, \ldots, b_k$ are coefficients of the sine terms.

Given a set of training data of the six orbital elements, the NLP estimated models are obtained by solving nonlinear programming problems specified by Eq. (8). The optimal solutions are variables of Eqs. (9) and (10). The implementation of the estimated NLP models are summarized in the Algorithm 3

Algorithm 3 Fitting algorithm

Input: Dataset, $y(:) = \{(t_1, y_1), (t_2, y_2), \ldots, (t_m, y_m)\}$
Output: NLP model, $\tilde{f}(t, x)$

1: **if** $y(:)$ is sawtooth **then**
2: $y(:) = \text{Unwinding}(y(:))$
3: **end if**
4: Define objective function $f(x) = \sum_{i=1}^{m} (y(t_i) - \hat{y}(t_i, x))^2$
5: Setting upper bound $ub := x(:) \leq u(:)$
6: Setting lower bound $lb := x(:) \geq \ell(:)$
7: Guess the initial point x_0
8: Solve the NLP $\tilde{f}(t, x) = \text{fit}(f(x), ub, lb, x_0)$

Figs. 5–9 present the NLP models of the six orbital parameter fittings of the tested satellites. For the NLP-fitted models, the x-axis is time in seconds, with time zero being the first data point. The y-axes are orbital elements of the TLE, that is, inclination, RAAN, eccentricity, argument of perigee, mean anomaly, and mean motion. The error plots are illustrated as histograms. It is apparent from the error plots that the NLP models are generally well-fitted. However, there are some error spikes from the error plots on two parameters, that is, the argument of perigee and the mean anomaly owing to the end-point fitting errors. The magnitude of the errors was small relative to the data. Furthermore, because the standard deviations are small, it can be implied that our NLP models are highly accurate, meaning that the estimated value is close to the true value.

4.3. Validation of the trained models

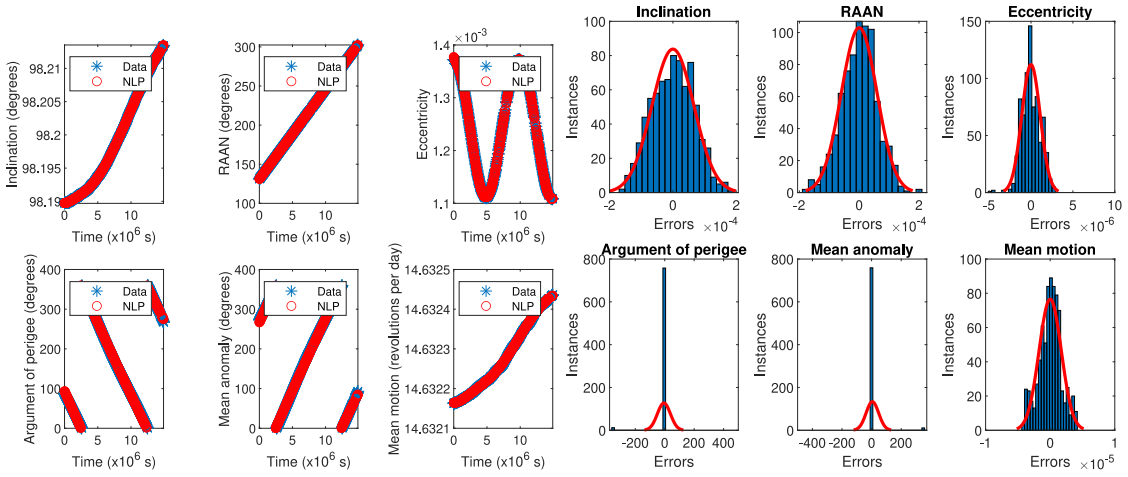
To validate the models obtained in Section 4.2, we predicted the next to 30–40 points after the trained dataset. This equates to 10–20 days of orbit determination depending on the satellite revisit period. Subsequently, the predictions were compared with the measured real TLE data. The predicted values for each tested satellite are illustrated in Figs. 10(a) through 14(a). The x-axis represents time in seconds and the y-axis represents the six orbital elements of the TLE. The star markers represent the measured TLE data points, whereas the plus markers are predicted data from our NLP models. It is clear that our models can predict the future trends of parameters such as RAAN, perigee argument, and mean anomaly for all satellites. Nevertheless, for almost all tested satellites, the predicted values of the parameters, that is, inclination, eccentricity, and mean motion, are diverted from the true TLE data points.

Next, we transform the obtained orbital element predictions into the Earth-centered inertial coordinate (ECI) as described in Section 2.2 to compare our performances with the well-known Kepler and SGP4 models. The results are presented in Figs. 10(b) through 14(b). The x-axis represents time in days, and the y-axis represents the position error in kilometers.

Satellite orbits are very dynamic depending on many external forces at that moment. Therefore, the error values when comparing the prediction locations against the pseudo-observation locations of the satellites (TLEs data) vary. This makes the prediction a very challenging task. To overcome this difficulty, the United States Space Force generates the TLE data with coefficients for external perturbations to use with the SGP4 routine. This makes the SGP4 routine reasonably good for a certain period of time. In this research, the TLE data is also assumed to be pseudo-observations for both training and validation purposes. For all cases, the proposed model is still able to predict the future locations of all satellites more accurately and has shown substantial improvement over the standard orbit propagation routines (Kepler and SGP4). The error value range of the proposed model is considered comparable to the popular machine learning technique as in Peng and Bai (2020) and Zhai et al. (2022) with simpler and fewer data points training.

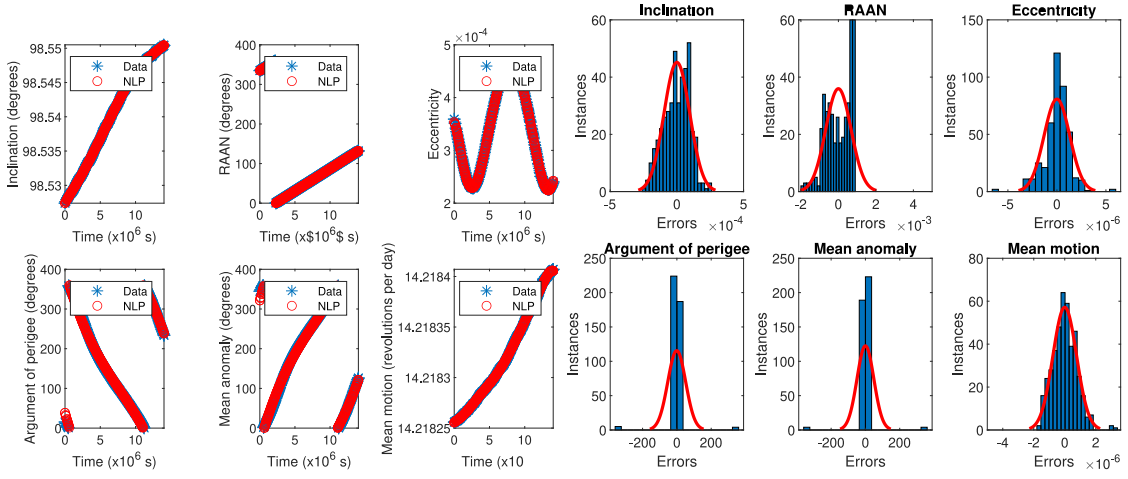
5. Sensitivity analysis

To understand the robustness of our models, we performed a sensitivity analysis to measure the impact of the input variables on the output of the model. The input variables are from two line elements: eccentricity (e), inclination (i), right ascension of the ascending node



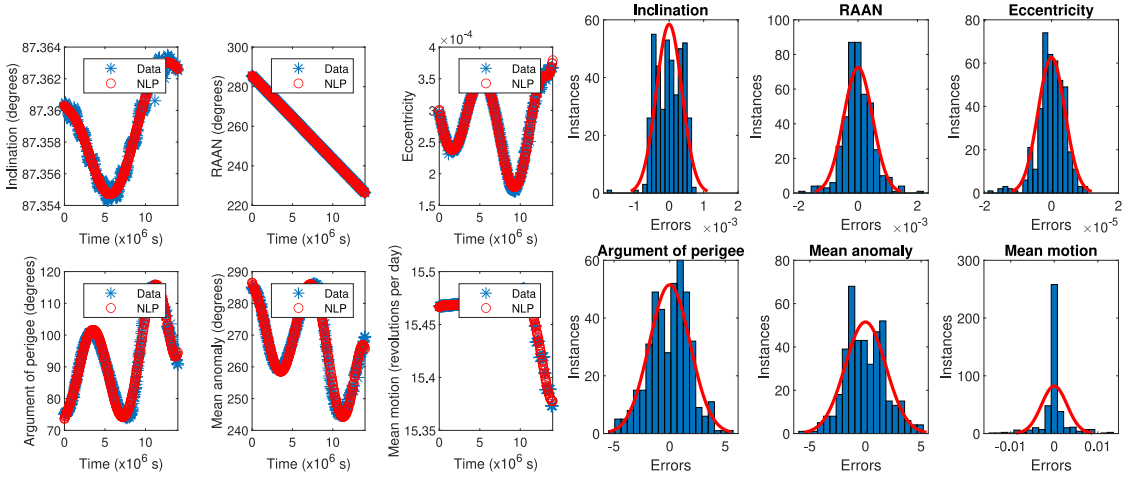
(a) NLP fitted models (b) Errors Plots

Fig. 6. NORAD ID: 27944.



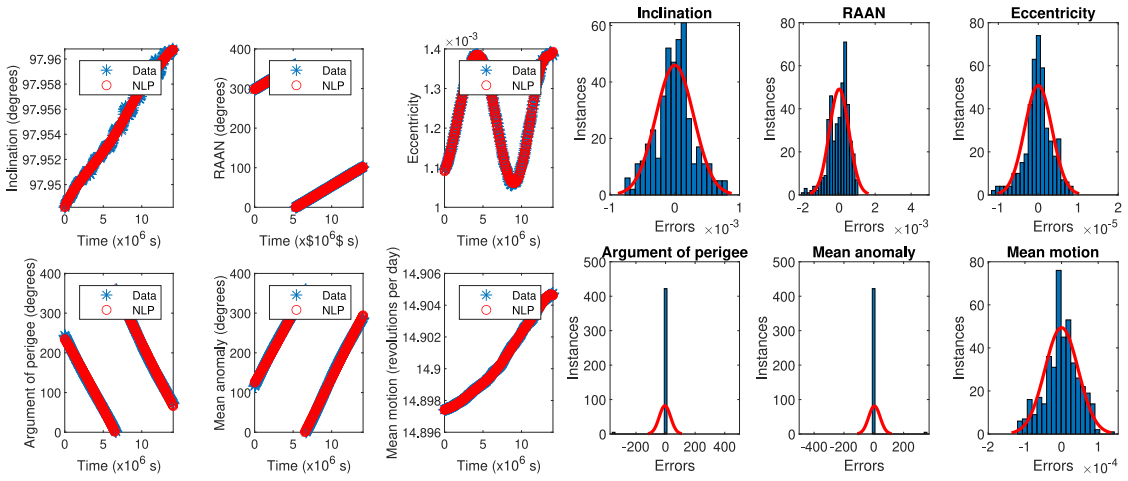
(a) NLP fitted models (b) Errors Plots

Fig. 7. NORAD ID: 35871.



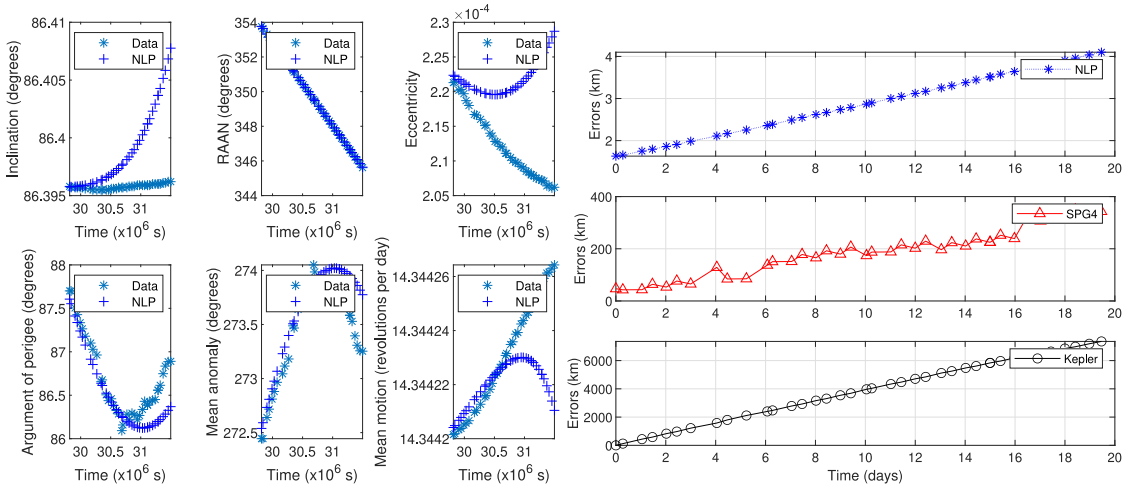
(a) NLP fitted models (b) Errors Plots

Fig. 8. NORAD ID: 39452.



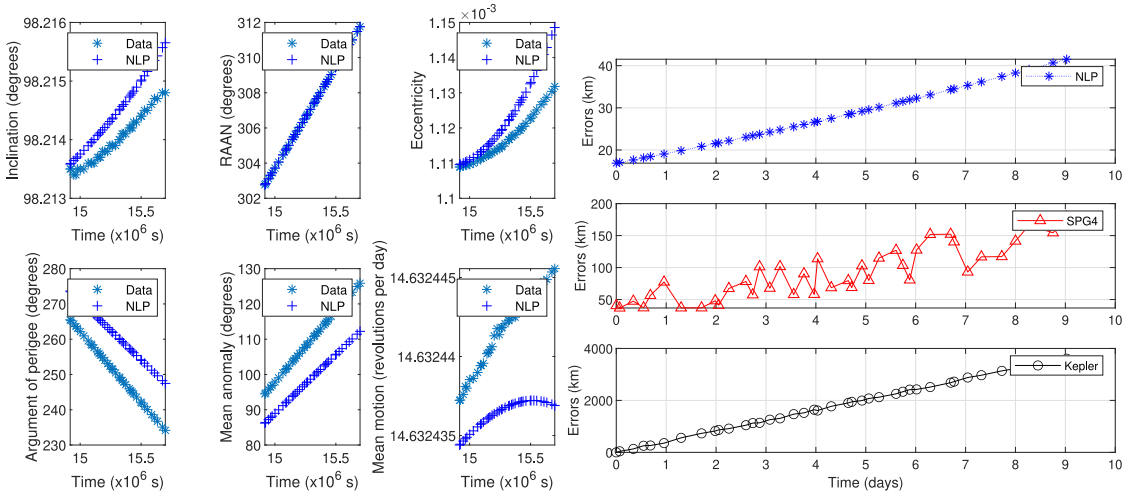
(a) NLP fitted models (b) Errors Plots

Fig. 9. NORAD ID: 40025.



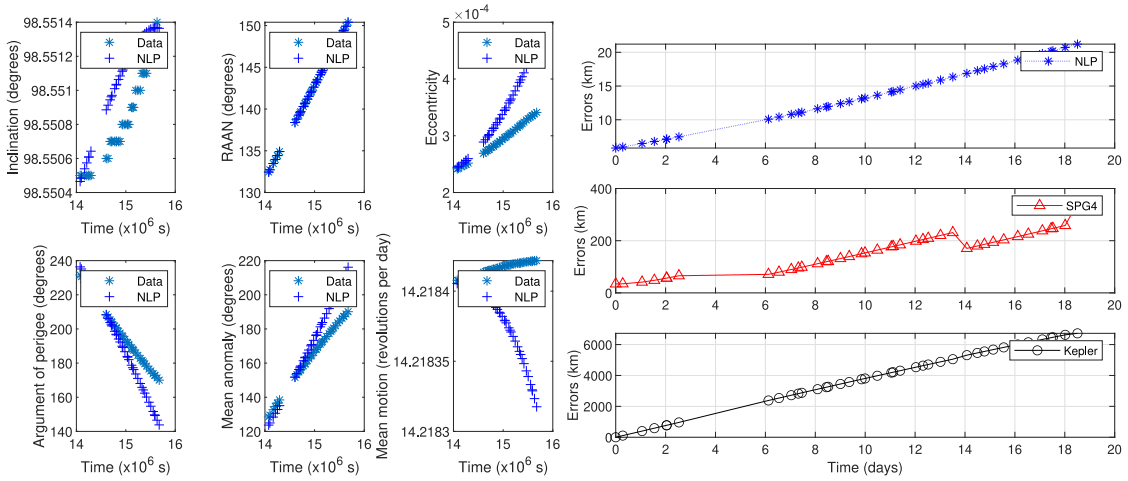
(a) Predicted values (b) Performance comparisons

Fig. 10. NORAD ID: 24793.



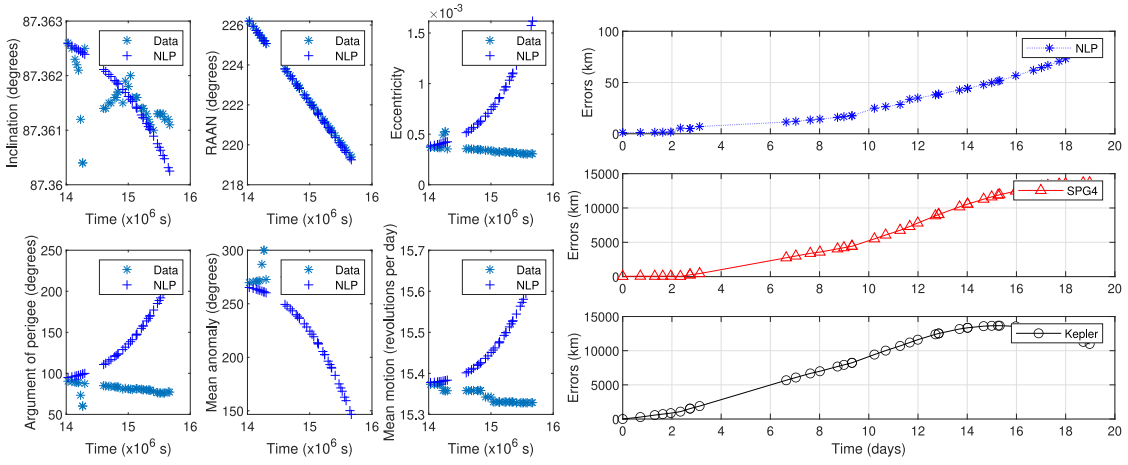
(a) Predicted values (b) Performance comparisons

Fig. 11. NORAD ID: 27944.



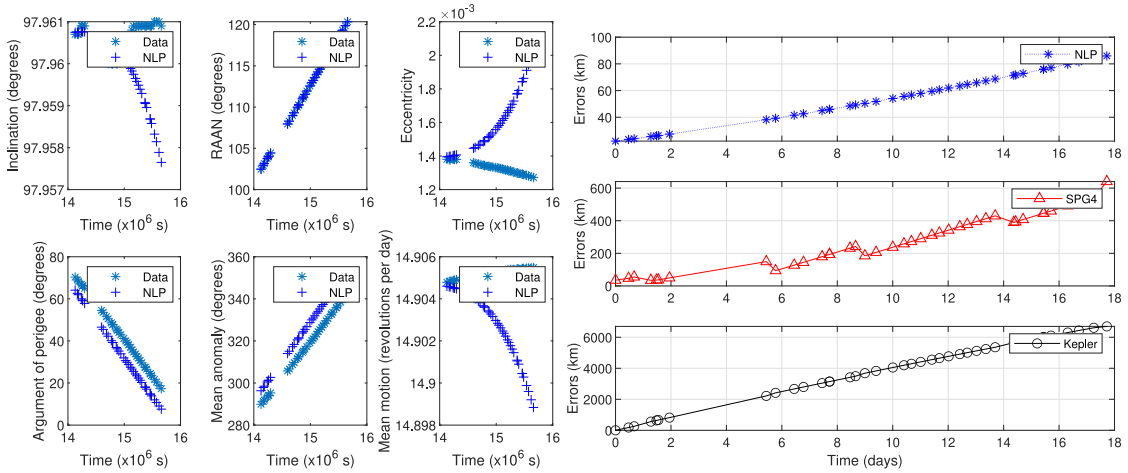
(a) Predicted values (b) Performance comparisons

Fig. 12. NORAD ID: 35871.



(a) Predicted values (b) Performance comparisons

Fig. 13. NORAD ID: 39452.



(a) Predicted values (b) Performance comparisons

Fig. 14. NORAD ID: 40025.

(RAAN, Ω), argument of perigee (ω), mean anomaly (M), and mean motion (n). The model output was a geocentric equatorial position r_{ijk}. Satellite NORAD ID: 24793 was chosen as the test unit. Each orbital element sample space was generated by perturbation around the mean value of data points in the dataset described in Section 4.1. That is, the normal distribution of each variable is characterized by the mean value of that variable, and the standard deviation is in the order of errors in the predictions of our model. Fig. 15 shows scatter plots of each orbital element sample space with 500 sample points.

5.1. Analysis methods

For this work, we analyzed the sensitivity using two methods: Pearson product-moment correlation (Pearson, 1895) and partial correlation method (Kendall and Stuart, 1945). The Pearson Correlation method calculates the correlation coefficient, R to measure a monotonic (linear) relationship between two random variables. The Pearson correlation coefficient R is a dimensionless measure of the covariance of the variables, ranging from −1 to 1. A value of R less than 0.1 indicates a negligible relationship, and greater than 0.9 a very strong relationship. Given a pair of random variables $(x, y) := \{(x_1, y_1), \dots, (x_m, y_m)\}$, the correlation coefficients between random variables x and y (R_{xy}) are defined as follows:

$$R_{xy} = \frac{\sum_{i=1}^{m}(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{n}(x_i - \bar{x})^2}\sqrt{\sum_{i=1}^{m}(y_i - \bar{y})^2}} \tag{11}$$

where m is the sample size, x_i, y_i are sample points, $\bar{x}$ and $\bar{y}$ are the means of the sample points x and y, respectively.

Partial correlation measures the linear relationship between two random variables, while controlling the effects of other random variables. To compute a partial correlation, one can solve multiple linear regressions, obtain the residuals of the two variables, and determine the Pearson correlation between the residuals. Let x, y be random variables, and $\mathbf{z} \in \mathcal{R}^{\mathbf{n}}$ be n-dimensional controlled variable. Denote x_i, y_i and $\mathbf{z_i}$ as the ith of m sample points of random variables x, y and $\mathbf{z}$. Let $[\mathbf{z_i}|1]$ be an augmented matrix of $\mathbf{z_i}$ and 1 to accommodate a constant term in the regression. Solve the following linear regressions:

$$\mathbf{w_x}^* = \arg\min_{\mathbf{w_x}} \sum_{\mathbf{i=1}}^{\mathbf{m}} (\mathbf{x_i} - \mathbf{w_x} \cdot [\mathbf{z_i}|1])^2 \tag{12}$$

$$\mathbf{w_y}^* = \arg\min_{\mathbf{w_y}} \sum_{\mathbf{i=1}}^{\mathbf{m}} (\mathbf{y_i} - \mathbf{w_y} \cdot [\mathbf{z_i}|1])^2 \tag{13}$$

where w_x^*, w_y^* are the optimal (n+1)-dimensional regression coefficient vectors of random variables x and y, respectively. Then, the residuals are calculated as

$$e_{x_i} = x_i - \mathbf{w_x}^* \cdot [\mathbf{z_i}|1] \tag{14}$$

$$e_{y_i} = y_i - \mathbf{w_y}^* \cdot [\mathbf{z_i}|1] \tag{15}$$

The partial correlation $\rho_{xy\cdot\mathbf{z}}$ is given by the Pearson correlation coefficients (Eq. (11)) of the residuals that is,

$$\rho_{xy\cdot\mathbf{z}} = \frac{m\sum_{i=1}^{m} e_{x_i} e_{y_i}}{\sqrt{m\sum_{i=1}^{m} e_{x_i}^2}\sqrt{m\sum_{i=1}^{m} e_{y_i}^2}} \tag{16}$$

5.2. Sensitivity analysis discussion

Tables 2 and 3 summarize the correlation and partial correlation coefficients between our inputs and the output of our model when perturbed. The inputs are six orbital elements: Ω, ω, M, and n. The output of the model is the position in the geocentric equatorial coordinates $r_{ijk} := (r_i, r_j, r_k)$, where the origin is at the center of the Earth. The x and y axes define the Earth's equator plane, and the z axis is along the north pole. According to the Pearson correlation analysis, orbital elements e, i, and n have little effect on the output, whereas Ω has a significant linear relationship with the position on the x and y axes. The ω and M display moderate correlation with x–y coordinates but have a strong impact on the z direction. The partial correlation analysis results illustrate that the inputs i and n have no effect on the output. The input Ω indicates a strong linear relationship in x and y directions, while the inputs ω and M exhibit highly partial correlation on all three axes. Both sensitivity analyses suggest that Ω, ω, and M are the most sensitive parameters to our model as they exhibit strong linear relationships to the output. In particular, the results imply that our model is robust to linear perturbation of the parameters e, i, and n. However, it must be pointed out that since both sensitivity analyses measure linear dependence between variables, the results cannot be inferred about nonlinear relationships.

To further study the robustness of our model, we vary the number of training points to observe its performance. Fig. 16 shows the performance analysis of the satellite with NORAD ID 39452 when the number of training data points is 400, 300, 200, and 100. The x-axis represents the predicted time in days, and the y-axis represents the predicted position errors in kilometers. It can be seen from the figure that increasing the number of training points does not guarantee better performance. The graph also suggests there is an optimal number of

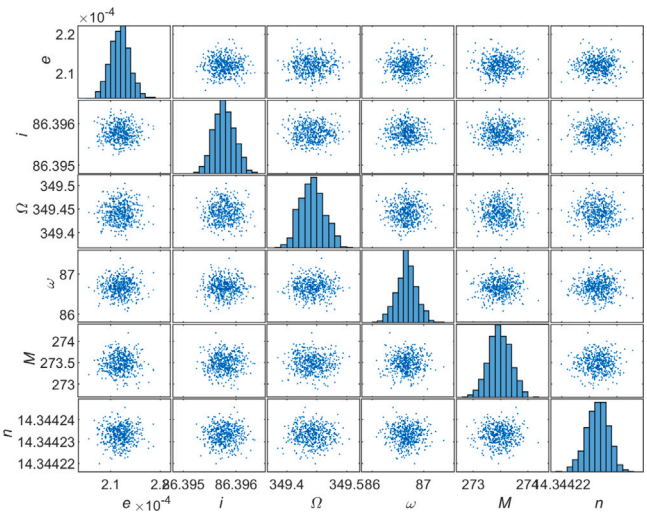


Fig. 15. Sample space scatter plots.

Table 2
Correlation coefficients.

R_{xy}	Geocentric equatorial position (r_{ijk})		
	r_i	r_j	r_k
e	−0.010078	−0.0048689	−0.030963
i	−0.010042	−0.022399	−0.078761
Ω	0.82559	0.76712	−0.042789
ω	0.37521	0.46152	0.71013
M	0.41134	0.48615	0.69602
n	0.013738	0.018622	0.048332

Table 3
Partial correlation coefficients.

$\rho_{xy\cdot\mathbf{z}}$	Geocentric equatorial position (r_{ijk})		
	r_i	r_j	r_k
e	−0.048135	0.047135	−0.67195
i	0.00049971	−0.0029364	−0.00085377
Ω	0.96775	0.99996	0.014987
ω	0.87168	0.9999	1
M	0.87195	0.99989	1
n	0.0060379	−0.0057882	0.06207

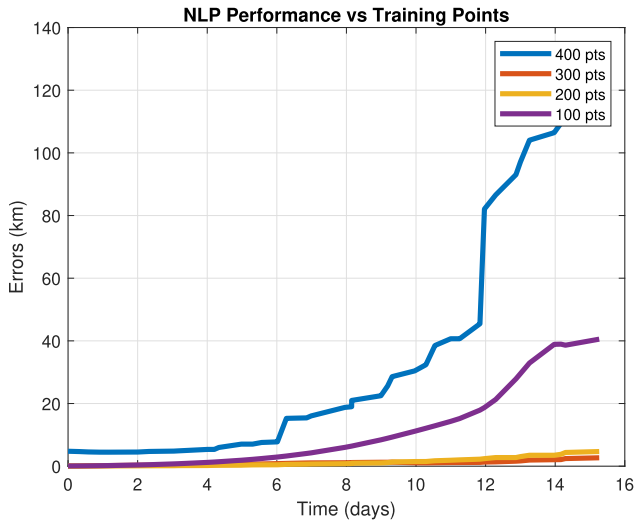


Fig. 16. NORAD ID: 39452 performance analysis.

training points for this particular satellite. This is due to the fact that the TLE data obtained are usually not distributed uniformly, and the effects of external forces such as atmospheric drag, gravitational force, and radiation force can cause abrupt changes in the satellite dynamics. Therefore, it is better to apply the concept of adaptive modeling to be responsive to unanticipated changes.

6. Conclusion

In this study, we investigated the use of nonlinear programming to predict the position of a satellite at a desired time. The value of the TLEs data is assumed to be the pseudo-real position of the satellite. It is clear that our NLP models are generally well fitted for all tested satellites. The fitted NLP models were used to estimate the future positions of the satellites. The results demonstrate that our models can predict far into the future with fewer errors than well-known models. It has been observed that a less noisy satellite dataset yields better prediction. Sensitivity analysis revealed that our model is robust to the linear perturbation of some orbital elements, that is, eccentricity, inclination, and mean motion. In future research, one can apply the proposed method to deep-space satellites. Nonlinear sensitivity analysis can be performed under a nonlinear relationship among the orbital elements. Our model can be applied as a model predictive control (MPC) scheme for satellite attitude control operations to cope with rapid changes in satellite dynamics due to external forces.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

The data is available through https://celestrak.org website.

References

Avriel, M. (Ed.), 2003. Nonlinear Programming: Analysis and Methods. Courier Corporation, USA.
Barton, K.E., McLaughlin, C.A., 2018. Long short term memory neural networks for the prediction of localized atmospheric density for orbit determination. In: Proc. 2018 AAS/AIAA Astrodyn. Spec. Conf..
Bazaraa, M.S., Sherali, H.D., Shetty, C.M. (Eds.), 2013. Nonlinear programming. Theory and algorithms. John Wiley and Sons, USA.
Bennett, J., Sang, J., Smith, C., Zhang, K., 2012. Improving low-earth orbit predictions using two-line element data with bias correction. In: Advanced Maui Optical and Space Surveillance Technologies Conference, Vol. 1. p. 46.
Brouwer, D., 1959. Solution of the problem of artificial satellite theory without drag. Astron. J. 64, 378–397.
Elhag, M.A., Yassin, A.A., Babiker, M.E., Elmageed, E.A.E.A., 2013. The unscented Kalman filter applied to satellite orbit determination using only publically available two-line element sets. In: Proc. 2013 Int. Conf. Comput. Electr. Electron. Eng. (ICCEEE). pp. 464–469.
Goh, S.T., Low, K.-S., 2018. Real-time estimation of satellite's two-line elements via positioning data. In: Proc. 2018 IEEE Aerosp. Conf.. pp. 1–7.
Jiang, C., 2021. An orbit determination method of spacecraft based on distribution regression. Open Astron. 30, 159–167.
Kelso, T.S., 2022. CelesTrak: Current GP element sets. https://celestrak.org/NORAD/elements/. [Accessed 05-Aug-2022].
Kendall, M.G., Stuart, A., 1945. The Advanced Theory of Statistics, Vol. 2. Hafner Publishing Company, New York.
Lane, M.H., 1965. The development of an artificial satellite theory using power-law atmospheric density representation. In: 2nd Aerosp. Sci. Meet.. p. 35.
Lane, M.H., Cranford, K.H., 1969. An improved analytical drag theory for the artificial satellite problem. In: Proc. Astrodyn. Conf.. p. 925.
Levit, C., Marshall, W., 2011. Improved orbit predictions using two-line elements. Adv. Space Res. 47, 1107–1115.
Li, B., Huang, J., Feng, Y., Wang, F., Sang, J., 2020. A machine learning-based approach for improved orbit predictions of LEO space debris with sparse tracking data from a single station. IEEE Trans. Aerosp. Electron. Syst. 56, 4253–4268.
Liu, J., Liu, L., Du, J., Sang, J., 2021. TLE outlier detection based on expectation maximization algorithm. Adv. Space Res. 68 (7), 2695–2712.
Lyddane, R.H., 1963. Small eccentricities or inclinations in the brouwer theory of the artificial satellite. Astron. J. 68 (8), 555–558.
Marshall, C.L.W., 2011. Improved orbit predictions using two-line elements. Adv. Space Res. 47 (7), 1107–1115.
MATLAB, 2022. version 9.12.0.1884302 (R2022a). The MathWorks Inc., Natick, Massachusetts.
McDowell, J.C., 2020. The low earth orbit satellite population and impacts of the spacex starlink constellation. Astrophys. J. Lett. 892 (2), L36.
Muldoon, A., Elkaim, G., Rickard, I., Weeden, B., 2009. Improved Orbital Debris Trajectory Estimation Based on Sequential TLE Processing. Paper IAC-09 A, p. 6.
Pearlman, M.R., Degnan, J.J., Bosworth, J.M., 2002. The international laser ranging service. Adv. Space Res. 30 (2), 135–143.
Pearson, K., 1895. VII. Note on regression and inheritance in the case of two parents. In: Proc. R. Soc. Lond., Vol. 58. pp. 240–242.
Peng, H., Bai, X., 2017. Limits of machine learning approach on improving orbit prediction accuracy using support vector machine. In: Proc. Adv. Maui Optic. Space Surveill. (AMOS) Technol. Conf..
Peng, H., Bai, X., 2018. Artificial neural network–based machine learning approach to improve orbit prediction accuracy. J. Spacecr. Rocket 55, 1248–1260.
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.
Vallado, D.A., Crawford, P.S., 2008. SGP4 orbit determination. In: Proc. AIAA/AAS Astrodyn. Spec. Conf. Exhib.. p. 6770.
Vallado, D.A., Virgili, B.B., Flohrer, T., 2013. Improved SSA through orbit determination of two-line element sets. In: Proc. ESA Space Debris Conf..
Wen, T., Zeng, X., Circi, C., Gao, Y., 2020. Hop reachable domain on irregularly shaped asteroids. J. Guid. Control Dyn. 43 (7), 1269–1283.
Wertz, J.R., Everett, D.F., Puschell, J.J. (Eds.), 2011. Space Mission Engineering: The New SMAD. Microcosm Press, Hawthorne, USA.
Zeng, X., Wen, T., Yu, Y., Circi, C., 2021. Potential hop reachable domain over surfaces of small bodies. Aerosp. Sci. Technol. 112, 106600.
Zhai, M., Huyan, Z., Hu, Y., Jiang, Y., Li, H.A., 2022. Improvement of orbit prediction accuracy using extreme gradient boosting and principal component analysis. Open Astron. 31, 229–243.