A Machine Learning-based Approach for Improved Orbit Predictions of LEO Space Debris With Sparse Tracking Data From a Single Station

Bin Li, Jian Huang, Yanming Feng, Fuhong Wang, and Jizhang Sang

Abstract—Accurate orbit prediction (OP) of space debris is vital in Space Situation Awareness (SSA) related tasks, like space collision warnings. However, due to the sparse and low precision observations, unknown geometrical and physical features of debris, and effects of incomplete force models, OP based on the orbital mechanics theory or physics-based OP of space debris suffers from rapid error growth over a long duration, limiting the period of validity of debris OP for precision space applications. Considering that the tracking arcs of a debris object over a single station often share a similar temporal and spatial distribution in the inertial space, the resultant OP errors possibly have a coherent relationship with the temporal and spatial distribution of tracking arcs. This paper proposes a machine learning (ML)-based approach to model the underlying pattern of debris OP errors from historical observations and apply it to modify the future physics-based OP results. The approach includes three steps: constructing a historical OP error set, training a ML model to fit the historical OP error set, and correcting the future physics-based OP with ML-predicted orbital errors. The ensemble learning algorithm of boosting tree (BT) is studied as the primary ML method for the error modeling and predicting process. Experiments with three low-Earth-orbit (LEO) objects, tracked by a single radar station, demonstrate that the trained ML models can capture more than 80% of the underlying pattern of the historical OP errors. More importantly, the errors of physics-based OP over the future 7 days reduce from thousands of meters to hundreds or even tens of meters through the error correction with the learned error pattern, achieving at least 50% accuracy improvement. Such dramatic OP improvements show the promising potential of ML for enhanced SSA capability.

Index Terms—Space debris, sparse data, orbit determination and prediction, boosting tree, support vector regression.

I. INTRODUCTION

SPACE debris includes nonfunctional payloads, abandoned launch vehicle stages, mission-related debris, and fragmentation debris. There are more than 23,000 orbital objects larger than 10cm in diameter tracked by the U.S. space surveillance network (SSN), of which space debris comprises more than 95 percent [1]. The population of space debris is rising rapidly, and the threats from space debris are reported frequently, highlighted by the well-known collision between operational Iridium 33 and defunct Cosmos 2251 on February 10, 2009 [2]. The requirements for safe and sustainable space operations drive the need to establish strong Space Situation Awareness (SSA) and Space Traffic Management (STM) capabilities in this increasingly crowded space environment. While detecting and tracking space debris requires very costly space-based and ground-based hardware facilities to gather data, how to accurately determine and predict the trajectories of space debris using the limited space surveillance facilities is the next biggest challenge. Accurate orbital knowledge of space debris can be widely applied in SSA related tasks such as collision warnings [3], maneuvering target tracking [4], [5], debris detumbling [6], and the future active debris removal [7], [8].

The North American Aerospace Defence Command (NO-RAD) two-line element (TLE) catalog has been the sole orbital data source of space debris for public access via https://www.space-track.org. But the accuracy of TLE propagations through the Simplified General Perturbations-4 (SGP4) propagator decreases dramatically with time due to the simplifications made in the perturbation force models [9], [10]. For example, the 7-day maximum error of TLE propagations of Larets orbit (altitude 691km) can grow up to 8km, when comparing TLE propagations to high precision ephemerides [11]. Hence, the TLE data is becoming gradually insufficient for the growing demands of space applications requiring higher accuracy, such as debris collision warnings, since missing such a warning could result in a catastrophic space event.

Alternatively, accurate orbital information of a space object can be generated from an orbit determination (OD) and prediction (OP) process with the orbit observations relating to the space object’s positions and velocities [12]. In the OD process, the optimal estimation of the space object’s truth orbit is obtained by adjusting the initial state parameters and force model parameters to fit the available observations best [13]. With the orbit state and force model parameters estimated from the OD process, the OP process begins by propagating the orbital solutions to the future time through the computation of the equations of motion. Theoretically, approaches to OP can be broadly categorized as numerical, analytical, and semi-analytic methods [14]. The numerical methods provide accurate predictions of the positions and velocities of space objects by integrating the equations of motion with high fidelity force models, such as the Cowell integrator and the Runge-Kutta 4 (RK4) integrator [13], but...
come with the disadvantage of slow computing speed. The analytical methods, like the SGP4 propagator, can be used to solve the equations of motion efficiently, but the orbit solutions are not accurate as only simplified force models are employed. The semi-analytic methods produce reasonably accurate but quickly delivered orbit predictions, which allows the numerical integration of the equations of mean elements to proceed with long step sizes of half a day to one day, while the short-periodic terms are recovered analytically. Typical semi-analytic methods include the averaging-based DSST integrator [15], [16] and the multiscaling-based semi-analytic integrator [17], [18]. All of these OP approaches are essentially physics-based, which employs a series of physical models to quantify the effects of perturbation forces on the orbits. Consequently, the accuracy of any such OP heavily depends on the OD accuracy and the fidelity of perturbation force models, such as Earth gravity, third-body gravitational attractions, solid Earth and ocean tides, atmospheric drag, solar radiation pressure, etc.

In theory, accurate OP can only be achievable when a good knowledge of the orbit state at the start of OP and the temporal and spatial variations of the space environment is available [13], [14]. For the satellites, their orbit states can be precisely determined with centimeter-level and globally distributed observations [19], [20] and then propagated in their high precision OP (HPOP) system. However, for the debris orbit problem, various sources could bring in large errors in the OD and OP processing. In an ideal case, the difference between the real observation and its theoretical values should be caused by the noise that occurs in making the observations. Yet, the sensors or instruments employed in the debris surveillance are not completely accurate, which inevitably introduces some discrepancy or error in the debris observations. More importantly, our knowledge of the temporal and spatial variations of the space environment is limited, resulting in the dynamic model errors [21], namely the differences between the nominal model parameters and the real model parameters, e.g., the atmospheric density model. The accuracy of widely used atmospheric density models, such as Jacchia71, MSIS86, and DTM94, is usually estimated at a 15% level [13]. As a result, the atmospheric drag, as the most significant non-gravitational perturbation acting on LEO objects, impacts the OP results seriously, especially for long-duration prediction.

Although more complicated force models can be taken into account, the orbit estimation error is still an important factor affecting the accuracy of the subsequent OP, because of the erroneous debris observations. Radar and optical tracking are the two most popular tracking techniques in current space surveillance [22]–[25]. The accuracy of the radar tracking is typically at an order of tens of meters/arcseconds for the range/angle (azimuth and elevation) observations, according to ground-based radar station parameters in Table 4-4 in the citation [14]. While the accuracy of the optical observations is usually at the level of 5-10 arcsec according to the Ground-based Electro-Optical Deep Space Surveillance (GEODSS) [14], which is equivalent to 25-50m at a distance of 1000km between the station and the space objects. Additionally, due to the limited tracking facilities, the accurate OD and OP of space debris also confronts two main challenges [26], [27]: (1) the tracking data is mostly sparse and ill-distributed, leading to the OD computation difficult to converge, for instance, there are only one or two passes of tracking data observed per day; (2) the ballistic coefficient or area-to-mass ratio needed for computing the non-conservative forces is unknown for most tracked debris.

To address the inaccurate debris OP challenges, multiple research efforts have been made through the modification of classical physical models. The major one is to improve current atmospheric density models through calibration methods [28]–[30] or develop new models [31]–[33]. As for the problem of unknown ballistic coefficients, research efforts have been made to use long-term historical TLE sets to estimation them [5], [34]. Besides, to overcome the geometrical weakness of sparsely distributed tracking arcs, the TLE-predicted positions are used as pseudo-observations to constrain the OD process so that the OP accuracy improves [26], [27]. However, these methods are still far from satisfactory for space safety operations to some extent, since developing high precision physical models is a very challenging task, which requires high quality and long-term observations with sufficient temporal and spatial resolutions.

Considering a single-station observation scenario, a piece of space debris may be tracked once or twice each day. The tracking arcs from the single station appear stable in terms of the temporal and spatial distribution in the inertial space. Thus, it is likely that the OP errors using the single-station observations follow some underlying pattern if the space environment does not change much. The long-term space surveillance has accumulated huge amounts of debris tracking data, which makes it possible to study the underlying pattern of debris OP errors in the short term and use it to improve the future OP accuracy. Fortunately, with the development of machine learning (ML) technique [35], [36], through which a mathematical model is automatically learned from past experiences without being explicitly programmed to perform the task, it provides a novel solution for the pattern recognition of debris OP errors.

Unlike the physics-based OP method, the ML method is essentially a data-driven modeling and predicting approach. It learns a function or mapping from labeled samples, known as training data, in order to make predictions or decisions for a new instance. A ML algorithm, also called model, is a mathematical expression \( \hat{y} = f(X) \) that represents the relationship between the input \( X \) and the output \( Y \) in the context of a problem. Various ML algorithms and tools have been developed and used in a wide variety of space applications, such as automatic space objects characterization [37], planning and optimization of spacecraft trajectories [38], [39], orbit determination [40], [41]. These applications clearly show the excellent capability of ML in the orbit dynamics field, which prompts researchers to a keen interest of applying the ML method for accurate orbit predictions of resident space objects (RSOs). Peng and Bai studied the support vector machine (SVM), Gaussian Proceses (GPs), and artificial neural network (ANN) to improve the OP accuracy [42]–[44], together with
the performance comparisons of these three ML algorithms on the OP accuracy improvement [45].

In this study, we propose a hybrid method to achieve accurate debris OP results with single-station sparse tracking data through integrating the ML method into the physics-based OP system. With the evolution pattern of historical OP errors learned by the ML method, it is used to produce orbital error predictions for the future physics-based orbit predictions, improving their accuracy via the error compensation or correction. A complete ML framework for improved physics-based OP results is implemented in three steps, including constructing the historical OP error set in a sliding OD/OP window using sparse tracking data, training an error model with the supervised ML algorithms, and correcting the future physics-based orbit predictions using ML-predicted orbital errors. In the developed ML framework, the learned ML model can be treated as an error corrector to the physics-based OP system. Thus, accurate physics-based orbit predictions of space debris can be achieved over a long duration, if the learned ML model captures the majority of the error pattern.

The rest of this paper starts with the description of debris OD and OP process using sparse tracking data in Section II. Next, the framework of the ML approach in modeling the debris OP error pattern is provided. In Section IV, the experimental result of the ML approach in improving the physics-based OP accuracy is presented. Finally, some conclusions are given.

II. DEBRIS OD AND OP USING SPARSE TRACKING DATA

A. Single-station Radar Tracking Scenario

As an important part of the SSN, the ground-based radar system is best suited for object tracking in the LEO region, and some very high powered radars are also capable of tracking objects beyond the LEO region [1], [14], [23]. Considering a single ground radar station and one debris, a set of radar tracking data includes the pointing angle and two-way range measurement of the debris, where the ground radar station serves as the signal transmitter and receiver. Essentially, the radar pointing angle is obtained in the topocentric coordinate system of the station by measuring the direction of the maximum signal amplitude of a space object, while the radar ranging measurement is the distance computed from the two-way flight time of a radar signal fired from the station to the space object and radiated back to the station. Thus, given the radar station location and debris position in the Earth Centered Earth Fixed (ECEF) coordinate system, the radar observations, including the azimuth/elevation $O_{Az/El}$ and the distance $O_{Dis}$ of a debris object with respect to (WRT) the radar station, can be derived according to the observational model, which can be referred to Section 6.2 in the citation [13].

Since the purpose of this study is to develop a ML framework for improving the debris OP accuracy under the condition of sparse tracking data, only a single ground radar station is considered. To reflect the realistic tracking scenario, strict configurations are assumed for the radar station by referring to the radar system parameters given in Table 4-4 in [14], as follows:

- The debris must be visible to the ground radar station;
- The accuracy of radar angular/ranging observation is 100 arcsec/100m, respectively;
- The maximum two-way ranging distance is 5000km;
- The minimum elevation mask is 20 degrees;

In the experiments, the ground radar station is located in Wuhan, China, with the geodetic latitude of 30.542 degrees, longitude of 114.346 degrees, and altitude 39.3m. The laser ranging satellites in the LEO region are selected as representatives in the radar tracking. Their precise orbits in the consolidated prediction format (CPF) [46] are downloaded from the International Laser Ranging Service (ILRS) data center (ftp://cddis.gsfc.nasa.gov/) and used to simulate the radar observations. A typical CPF file usually contains 3-5 days of accurate satellite positions and these positions are generated by propagating the precisely determined satellite state in its HPOP system. In our study, only the satellite positions from the first day in each CPF file are used, since they are more accurate than those of later days. Then, a precise CPF orbit can be produced by combining the first-day positions from a series of CPF files. Lastly, a precise OD process with high fidelity force models is implemented to fit the combined positions, in order to avoid the discontinuities that occur in making the precise CPF orbit. Following these procedures, a precise and smooth CPF orbit over long arcs is obtained, which will be used as truth orbit to compute the radar observations according to the preset configurations.

B. Orbit Estimation Strategy

The equations of motion of an Earth-orbiting object in the Earth-centered inertial (ECI) coordinate system is given by a system of three second order differential equations of [14]

$$ f(x, c, t) = \ddot{r} = -\frac{\mu}{r^3} + a_{per} + \Gamma $$

(1)

where $r$, $\dot{r}$, and $\ddot{r}$ are the space object’s position, velocity, and acceleration vectors in the ECI frame, respectively; $r = ||r||$, with $|| \cdot ||$ denoting the Euclidean norm of a vector; $\mu = GM$ is the Earth gravitational constant; $a_{per}$ is the perturbation acceleration vector caused by non-spherical gravity, third-body gravity attraction, atmospheric drag, solar radiation pressure, and so on; $\Gamma$ is the thrust acceleration vector; $f(x, c, t)$ denotes the equations of motion, in which $x$ is the space object’s state (6-dimensional), $c$ is the force model parameter, which refers to the drag coefficient $C_d$ used as a critical parameter to compute the atmospheric drag effect; $t$ is the time.

For a LEO object, given the initial condition of Eq.(1), i.e., the initial state $x_0 = x(t_0)$ and the drag coefficient $C_d$, $x_t$ at a future time epoch $t$ can be uniquely predicted with a numerical, analytical, or semi-analytical orbit propagator. In practice, the true value of $x_0$ is never exactly known, but can be accurately estimated using a batch or filtering algorithm. Assume that $n$ radar angular observations $O_{Az/El,i}$, $i = 1, 2, \cdots, n$, and $m$ radar distance observations $O_{Dis,j}$, $j = 1, 2, \cdots, m$ of a tracked object are available, with their corresponding weights of $P_{Az/El,i}$ and $P_{Dis,j}$ characterizing their relative accuracy. Let $v_{Az/El,i}$, $v_{Dis,j}$ denote the residuals between real observations $O_{Az/El,i}$, $O_{Dis,j}$ and their theoretical values. Given the
initial condition \((x_0, C_d)\), \(v_{Az/El,i}\) and \(v_{Dis,j}\) are computed as
\[
v_{Az/El,i} = O_{Az/El,i} - C_{Az/El,i} \tag{2}
v_{Dis,j} = O_{Dis,j} - C_{Dis,j} \tag{3}
\]
where \(C_{Az/El,i}\) and \(C_{Dis,j}\) correspond to the theoretical values of \(O_{Az/El,i}\) and \(O_{Dis,j}\), which are computed by the space object’s state prediction and the radar station location according to the radar angle and range observational models.

After obtaining all the residuals of available observations, a weighted sum of squares of the residuals (WSSR) is given by
\[
WSSR = \sum_{i=1}^{n} P_{Az/El,i} v_{Az/El,i}^2 + \sum_{j=1}^{m} P_{Dis,j} v_{Dis,j}^2 \tag{4}
\]

Obviously, \(WSSR\) is a function of the initial condition. In the sense of batch least squares (BLS), the optimal estimations for the initial condition can be obtained by minimizing \(WSSR\) with respect to the initial state and force model parameters, as
\[
\dot{x}_0 = x_0 + \delta x_0, \ddot{C}_d = C_d + \delta C_d \tag{5}
\]
where
\[
\begin{bmatrix}
\frac{\partial x_0}{\partial C_d} \\
\frac{\partial x_0}{\partial \delta C_d}
\end{bmatrix} = (B^T P B)^{-1} B^T P l \tag{6}
\]
\[
B = \frac{\partial H}{\partial \dot{x}_t}, \quad \dot{x}_t = \frac{\partial H}{\partial [x_0, C_d]_i} = H_{t,t} \cdot \Phi_{t, t_0} \tag{7}
\]
\(
\delta x_0 \) and \(\delta C_d\) are the estimated corrections to the initial condition of \(x_0\) and \(C_d\), respectively; \(B\) is the error equation matrix; \(P\) is the weight matrix of the observations which is typically a diagonal matrix with its elements being the inverse of the observation variance; \(I\) is the observation residual vector; \(H_{t,t}\) is the observation matrix, \(H_{t,t} = \frac{\partial H}{\partial \dot{x}_t}\); \(\Phi_{t, t_0}\) is the state transition matrix (STM), containing the variational partial of the initial condition.

In our experiments, an orbit determination and analysis software for processing the angular and ranging observations is developed by the authors \([47], [48]\). It applies the Cowell numerical orbit propagation method for the state solutions in Eq.(1) and the (BLS) algorithm for the optimal estimations in Eq.(5). In the BLS OD processing, 3 days’ radar tracking arcs of space debris are used to compute the optimal estimation \((\hat{x}_0, \hat{C}_d)\), and the OD fit span is set starting at the epoch time of the first tracking arc in radar tracking scenario. The initial orbit state in the BLS OD system is computed from the latest TLE before the OD fit span. As two types of observations, \(O_{Az/El,i}\) (2-dimensional) and \(O_{Dis,j}\) (1-dimensional), are weighted in the BLS OD processing, they provide different geometrical strengths when forming the normal equations Eq.(6). Thus, their relative weights should be properly assigned for obtaining the optimal estimation, regarding to the observation accuracy and the geometrical strengths. In our BLS OD processing, a weight strategy of \(P_{Az/El} = 1.0\), \(P_{Dis} = 10^{-10}\) is adopted by referring to the citation \([27]\), so that the accuracy of \(O_{Dis}\) (in meters) is at the same order as that of \(O_{Az/El}\) (in radians) when fitting the angle and range observations together (3-dimensional observations).

The BLS OD computation is an iterative process and converges when changes between the estimated initial state and its previous estimate are less than the preset values of 1.0m for the position vectors and 0.01m/s for the velocity vectors. After the OD convergence, an outlier detection procedure is performed to ensure no observation with gross error is used in the OD processing. After the completion of OD computation, the OD time window will slide to the next one with the starting time epoch set at the beginning time of the next tracking arc, as illustrated in Fig. 1. Following this OD procedure, all the pairs of the estimations \((\hat{x}, \hat{C}_d)\) of the available tracking arcs are obtained.

To reflect the orbit error pattern relative to the high fidelity dynamical system of CPF orbits in simulating the radar tracking scenario, a simplified dynamical system (SDS) is adopted in Eq.(1) for the debris OD and OP processing so that any variations of space environment could introduce errors in the OD and OP processing. In the SDS, the main perturbations acting on LEO objects are considered, including Earth gravity, Sun and Moon attraction, solar radiation pressure, and atmospheric drag. The Joint Gravity Model-3 (JGM-3), truncated to 20/20 degrees/orders, is used to compute the Earth gravity effect. The Sun and Moon positions are provided by the DE406 planetary ephemeris. When computing the solar radiation pressure and drag effects, the area-to-mass ratio of a space object is determined as \(BC = \frac{2}{C_d}\), where \(BC\) is the TLE-derived ballistic coefficient \([34]\). Initial values of the force model parameters, including the drag coefficient \(C_d\) and the solar radiation pressure coefficient \(C_r\), are set with \(C_d = 2.2\) and \(C_r = 1.1\). The NRLMSISE-00 atmospheric density model \([49]\) is applied to compute the atmospheric mass density \(\rho\), which will be then added into a random error from a Gaussian distribution with zero mean and 0.1\(\rho\) standard deviation, in order to reflect the effect of atmospheric uncertainty.

C. Construction of historical OP Error Set

After obtaining the estimation \((\hat{x}, \hat{C}_d)\) of each tracking arc following the above OD strategy, an orbit integration of Eq.(1) with \((\hat{x}, \hat{C}_d)\) as initial values is performed in the SDS through the Cowell integrator, yielding the predicted state as
\[
\ddot{x}_{j,i} = \phi(t_{j,i}; \hat{x}_i, t_i) \tag{8}
\]
where \(\hat{x}_i\) is the estimated state of the \(i\)-th tracking arc, with its initial time at \(t_i\); \(\ddot{x}_{j,i}\) is the predicted state at a future time
$t_j$ based on $\hat{x}_j$, and $t_j - t_i$ is the prediction time length; $\phi$ is the implicit state solution of Eq. (1), i.e., state propagation [13], [21], [48].

In Eq. (8), the error of the predicted state $\hat{x}_{j;i}$ WRT the truth orbit can be interpreted as a result of the mutual effect of the orbit estimation error and the dynamic model error. Since the true orbit of space debris is unknown in the real data processing, the estimation $\hat{x}_j$ is used instead as the reference to assess the accuracy of $\hat{x}_{j;i}$, expressed as

$$e_{j;i} = \hat{x}_{j;i} - \hat{x}_j$$

(9)

where $e_{j;i}$ is the error of $\hat{x}_{j;i}$, i.e., the OP error; $\hat{x}_j$ is the state estimation at $t_j$; $t_j > t_i + 3$ days, since the OP starts from the terminal time of the 3-day OD window. In Eq. (9), $e_{j;i}$ is computed in the ECI coordinate system, which will be then transformed into the RSW coordinate system [14] for orbital error pattern analysis. In this way, an OP error set in the sliding OD window is constructed by combining $e_{j;i}$ of all the pairs of state estimations, as illustrated in Fig. 2.

![Fig. 2. Illustration of OP error set construction in the sliding OD window.](image)

III. ML IN MODELING THE OP ERROR PATTERN

Due to the observation error and incomplete perturbation force models, the estimated orbit states of space debris still contain errors. This causes the subsequently predicted orbit to further deviate from its truth orbit as a result of the effect of dynamic model error. According to the orbital error propagation characterization [21], [48], there would be some patterns existed in the OP errors. Inspired by the ML theory, such patterns can be modeled through the data analysis based on large amounts of OP error samples. Essentially, the ML method learns a mathematical model $Y = f(X)$ between the input $X$ and the output $Y$ of the labeled samples and then makes predictions when a new input is given. In this section, we will first discuss the data preparation for learning an error model $Y = f(X)$ using the ML method, followed by the implementation of ML algorithms in modeling the evolution pattern of historical OP error set.

A. Data Preparation

The ML method works on the sample data (training data and testing data) which are independent and identically distributed, even if their probability distribution is not known beforehand. Hence, if a pattern is contained in these samples, it can be learned by the ML algorithm. For a specific problem, the success of the pattern learning strongly relies on data preparation.

Generally, the training data is composed of the input $X \in \mathbb{R}^N$ and the output $Y \in \mathbb{R}$. The purpose of a learning process is to find the underlying mapping relationship between $X$ and $Y$ if it exists, i.e., $Y = f(X)$. Therefore, the training data has to be well prepared such that this underlying mapping relationship can be captured. It is always easy to determine the target variable $Y$. On the contrary, the choice of learning variables in constructing $X$ is not straightforward, as there are many latent features associated with $Y$. Theoretically, the more learning variables are included in $X$, the more accurate but complex the trained model would be. But it would result in the overfitting problem [50], in which the trained model works well for the training data, but produce bad predictions for the unseen data. In this sense, a desirable model should be trained to avoid overfitting, but at the same time take into account the most important factors affecting the target variable $Y$.

As expressed in Eqs. (1), (8), and (9), the OP error $e_{j;i}$ is computed by comparing the predicted state $\hat{x}_{j;i}$ to the reference state, while $\hat{x}_{j;i}$ is obtained by numerically integrating the equations of motion with the given initial condition $(\hat{x}_i, \dot{C}_{d;i})$. In this process, the factors that directly contribute to $e_{j;i}$ should be included to construct $X$. Thus, at a specific epoch $t_j$, $X$ includes the learning variables as

- OP time length $\Delta t_{j;i} = t_j - t_i$, $\Delta t_{j;i} \leq 7$ days, since the accuracy of state propagation accuracy decreases as $\Delta t_{j;i}$ increases. The longer $\Delta t_{j;i}$ is, the larger the magnitude of $e_{j;i}$ will be.
- Initial orbit state $\hat{x}_i$ for state propagation. $\hat{x}_i$ in both the Keplerian orbit element (KOE) (6-dimensional) form $\hat{x}^\text{KOE}_i$ and ECI position and velocity (6-dimensional) form $\hat{x}^\text{PV}_i$ are used. The combination of $\hat{x}^\text{KOE}_i$ and $\hat{x}^\text{PV}_i$ contains all the necessary orbit information of a space object, such as the orbit altitude, period, shape, and state.
- Force model parameter $\dot{C}_{d;i}$ for state propagation. It is a critical parameter to account for the effect of atmospheric drag on LEO objects.
- The predicted state $\hat{x}_{j;i}$ based on $\hat{x}_i$, also expressed in both the KOE (6-dimensional) form $\hat{x}^\text{KOE}_{j;i}$ and ECI position and velocity (6-dimensional) form $\hat{x}^\text{PV}_{j;i}$. It is reasonable that $\hat{x}_{j;i}$ should be included, since it is used to compute the OP error.

Obviously, the target variable $Y$ corresponding to $X$ refers to the OP error as

- $e_{j;i}$ at time $t_j$, expressed in the RSW coordinate system as $e = [e_A, e_C, e_R, e_{vA}, e_{vC}, e_{vR}]^T$, where $e_A$, $e_C$, and $e_R$ denote the along-track, cross-track, and radial position errors, respectively, and $e_{vA}$, $e_{vC}$, and $e_{vR}$ denote the along-track, cross-track, and radial velocity errors, respectively. It means that six error models need to be trained for all the error components.

Given the learning and target variables defined above, a sample of the training data is constructed in the form of $(X, Y)$, where $X = [\Delta t_{j;i}, \hat{x}_i, \hat{x}^\text{KOE}_i, \hat{x}^\text{PV}_i, \dot{C}_{d;i}, \hat{x}_{j;i}, \hat{x}^\text{KOE}_{j;i}, \hat{x}^\text{PV}_{j;i}]$ and $Y = e_{j;i}$. It means that each target variable corresponds to 26 learning variables in the training data so that the underlying
error pattern can be sufficiently modeled. Hence, with a series of samples \((X, Y)\) being well-prepared, an underlying mapping relationship \(Y = \hat{f}(X)\) can be trained by the ML approach in the learning system. More importantly, if the testing data beyond the learning period is correlated to the training data, the trained ML model can be applied to the testing data to predict new errors, which are then used to correct the state predictions during the testing period, as illustrated in Fig. 3.

In Fig. 3, the ML-predicted orbital error, denoted as \(e_{ML}\), is computed by

\[
e_{ML} = \hat{Y} = \hat{f}(X)
\]

where \(\hat{Y}\) is the error prediction corresponding to \(X\) of the training or testing data with the trained ML model. Once \(e_{ML}\) is obtained, the error correction is performed to modify the physics-based state predictions \(\tilde{x}_{j;i}\) by removing \(e_{ML}\), as

\[
\tilde{x}^{ML}_{j;i} = \tilde{x}_{j;i} - e^{ECI}_{ML}
\]

where \(\tilde{x}^{ML}_{j;i}\) is the improved physics-based state predictions; \(e^{ECI}\) is the \(e_{ML}\) term expressed in the ECI frame.

The residual error \(\Delta e\) corresponding to \(\tilde{x}^{ML}_{j;i}\) after error correction is equivalent to

\[
\Delta e = e - e_{ML}
\]

where \(e\) is the original error of \(\tilde{x}_{j;i}\).

When assessing the performance of learned ML models in modifying \(\tilde{x}_{j;i}\) via error correction, the improvement percentage is quantified by the performance metric \(P_{ML}\) as

\[
P_{ML} = (1 - \frac{RMS_{\Delta e}}{RMS_{e}}) \times 100\%
\]

where \(RMS_{e} = \frac{1}{n} \sum_{i=0}^{n} e_i^2\) is the root mean square (RMS) value of \(e\); \(RMS_{\Delta e} = \frac{1}{n} \sum_{i=0}^{n} \Delta e_i^2\) is the RMS value of \(\Delta e\); \(n\) is the number of training or testing samples. In Eq. (13), all the samples are included in computing \(P_{ML}\), giving an overall performance assessment of the ML model on the training or testing data. The higher the \(P_{ML}\) value is, the better the trained ML model is obtained. In an ideal case, \(P_{ML}\) can reach 100%, indicating all the underlying patterns contained in \(e\) are perfectly extracted by the trained ML model, namely \(e_{ML} = e\). However, \(P_{ML}\) could have a negative value if the OP error pattern cannot be found by the trained ML model.

The learned error pattern is strongly dependent on the configuration of the learning variables in the sense that whether the learning variables can characterize the features that affect OP errors, as well as the temporal and spatial distribution of the training and testing data. Thus, the trained ML model can be applicable to the time periods if they share a similar space environment. To guarantee the period of validity of the trained ML model for future error prediction, in the experiments, the time periods of the training data and the testing data is formed by a sliding time window, as shown in Fig. 4.

As illustrated in Fig. 4, the training data over the past 30 days is used to train a ML model. The trained model is then used to predict the OP errors over the 7 days following the training period. In this case, only the tracking data in the 3 days before the last tracking arc, as well as the learning and target variables in the past 30 days, is needed. The tracking data is processed in the standard physics-based OD/OP process, and the OP errors are predicted by the trained ML model with learning variables derived from the OD/OP. The ML-predicted errors are then applied to correct the physics-based orbit predictions.

B. ML Algorithms for Training the OP Error Model

Modeling the relationship between the input \(X\) and the output \(Y\) of the prepared OP error set is a multivariable regression problem. Of all the types of ML methods, the supervised ML methods appear suitable to learn a mapping from \(X\) to \(Y\). In the work of [42], the SVM [51], [52] was studied to improve the satellite OP accuracy, but the errors of improved position predictions were still at an order of tens of kilometers, although three radar stations were employed in the satellite tracking. Such an OP accuracy cannot be applicable to the precision space applications. In this paper, we apply the ensemble learning method based on the boosting algorithm instead to mine the underlying OP error pattern of space debris under a single-station tracking scenario. In this section, the idea of the boosting tree (BT) algorithm is outlined, and more details are referred to the citations.

The BT algorithm is a popular ensemble learning method that can be applied for both classification and regression...
the loss function, expressed as 

\[ f_m(x) = f_{m-1}(x) + T(x; \Theta_m) \]  

where \( x \) is the input variable, and \( x = X \) in our experiments; \( T(x; \Theta_m) \) is the decision tree (base learner) with its parameter of \( \Theta_m \); \( M \) is the number of \( T(x; \Theta_m) \). Here, \( T(x; \Theta_m) \) refers to the regression tree, since a regression task is performed in our study.

Given a set of training data, the BT model is trained using the forward stage-wise algorithm as

\[ f_m(x) = f_{m-1}(x) + T(x; \Theta_m) \]  

where \( f_{m-1}(x) \) and \( f_m(x) \) are the predictions at step \( m - 1 \) and \( m \), with an initial value \( f_0(x) = 0 \). \( T(x; \Theta_m) \) is the decision tree learner at step \( m \), which is estimated by minimizing the loss function, expressed as

\[ \Theta_m = \arg \min_{\Theta_m} \sum_{i=1}^{N} L[y_i, f_{m-1}(x_i) + T(x_i; \Theta_m)] \]  

where \( N \) is the number of the training samples; \( y_i \) is the output variable corresponding to the input variable \( x_i \) of the training data, and \( y_i = Y_i \) in our study according to its definition.

The quadratic loss function \( L(y, f(x)) = [y - f(x)]^2 \) is used to determine \( \Theta_m \) in Eq. (16) by minimizing the loss of

\[ L[y_i, f_{m-1}(x_i) + T(x_i; \Theta_m)] = [r - T(x_i; \Theta_m)]^2 \]  

where \( r = y - f_{m-1}(x) \) denotes the residual at step \( m - 1 \).

In Eqs. (14)-(17), the BT algorithm is interpreted that the first learner \( T(x; \Theta_1) \) is used to fit the whole space of input data, and then the second learner \( T(x; \Theta_2) \) is applied to fit the residuals of the first fitting process, thus overcoming the drawbacks of the first learner. This fitting process is repeated until meeting the stopping criterion. Finally, a BT model, as expressed in Eq. (14), is obtained to approximate the target function \( Y = \hat{f}(X) \) by the sum of the predictions of each base learner.

The implementation of the BT algorithm is summarized as:

1. Initialization \( f_0(x) = 0 \).
2. For \( m = 1; m \leq M; m + + \) do
   1. Compute the residuals \( r_{mi} = y_i - f_{m-1}(x_i), i = 1, 2, \ldots, N; \)
   2. Fit \( r_{mi} \) to train a regression tree model \( T(x; \Theta_m) \);
   3. Update \( f_m(x) = f_{m-1}(x) + T(x; \Theta_m) \);
3. End for
4. Obtain the BT model \( f_M(x) = \sum_{m=1}^{M} T(x; \Theta_m) \);

In this research, the BT algorithm for training the OP error model is programmed in the MATLAB (R2018a) environment. In the implementation of the BT algorithm, the least squares boosting (LSBoost) is adopted for solving the regression problem [55]. A grid search method is adopted to adjust the parameter values in the BT model. In the experiments, the size of the OP error set is very limited, less than 900 samples, which will be given in Section IV.B. In this case, an amount of 30 base learners, with the leaf size being minimum 8 and learning rate being 0.1, are determined to fit them in the learning process. We note that these settings are not unique. It is a trial-and-error process, since the larger the leaf size and/or the number of employed learners is, the slower the training speed will be.

IV. EXPERIMENTAL RESULTS

Three LEO space objects, Larets (altitude 691km), Starlette (altitude 815km-1100km), and HY-2A (altitude 971km), are tested in the experiments. Their precise CPF-format orbits are available in the ILRS data center [56]. In modeling the OP errors of the satellites, the BT algorithm is used as main ML algorithm for the training tasks. In addition, the performance of ML based on the support vector regression (SVR) in the error pattern learning is also presented, in order to validate our developed ML framework for improved orbit predictions of LEO space debris through modeling and transferring the historical orbit error patterns in the single-station radar tracking scenario.

A. Accuracy of OD and OP Using Sparse Tracking Arcs

In the experiments, a 90-day single-station radar tracking scenario is simulated for the test satellites, and up to two tracking arcs are picked each day to determine the orbits. In this radar tracking scenario, a total of 150 radar tracking arcs are collected by the radar station for each satellite. The average time length of each tracking arc is 305.7s for Larets orbit, 483.6s for Starlette orbit, and 502.0s for HY-2A orbit, which are very short when comparing to their orbital periods.

According to the orbit estimation strategy in Section II, the BLS OD processing using tracking arcs in a 3-day fit span is performed to obtain the optimal state estimation \( \hat{x} \) for the three satellites. Since \( x \) will be used as the reference to compute the subsequent 7-day OP errors, an accuracy assessment of \( \hat{x} \) is necessary to verify that \( \hat{x} \) is sufficiently precise as the approximation of the truth orbit (precise CPF-format orbit). The true estimation error, denoted as \( \Delta \), is computed as the difference between \( x \) and the truth orbit, and then transformed into the RSW coordinate system. Fig. 5 presents the true position error, \( (\Delta_A, \Delta_C, \Delta_R) \), of the estimated state for the collected tracking arcs.

As shown in Fig. 5, the true estimation errors of 150 tracking arcs remain steady within a certain range in each position component. The along-track position errors \( \Delta_A \) are slightly large, but most of them are within [-50m, 50m],...
while the majority of $\Delta_C$ and $\Delta_R$ are within [-10m, 10m]. It indicates the correctness of the developed BLS OD processing using sparse radar observations. Considering the magnitude of 7-day OP errors at such selected orbit altitudes is usually at hundreds or even thousands of meters, the true estimation errors hardly affect the OP error pattern learning. Hence, the state estimations $\hat{x}$ can be regarded as accurate enough in representing the truth orbit to compute the OP errors. This is important for the OP error computation of space debris, as only the state estimations $\hat{x}$ are available as references for the orbital accuracy assessments.

With the state estimations $\hat{x}$ as references, a 30-day OP error set is constructed for the learning process. Fig. 6 presents an example of the OP error set over the first 30 days (1st-30th day) of the Larets orbit.

In Fig. 6, the along-track position prediction error $e_A$ grows dramatically as the OP time length increases, with the maximum value of the whole OP errors approximating 7km over the 7-day OP time span, while the cross-track and radial position prediction errors, $e_C$ and $e_R$, scatter around the zero value and are all within [-50m, 50m]. Obviously, $e_A$ contributes to the most of the position prediction error, indicating the significance of reducing $e_A$ to improve the OP accuracy. Therefore, in the following learning process, $e_A$ will be used as the target variable to be learned by the ML algorithms.

**B. OP Error Pattern Learning Using Training Data**

The purpose of ML is to learn a desirable OP error model $Y = \hat{f}(X)$, which not only fits well with the training data, but also makes good predictions with new inputs. To avoid the overfitting problem and generalize the learned OP error pattern to future epochs, the cross-validation (CV) is necessary for the learning process. It is a statistical method used to obtain an accurate estimate of the ML model which generally results in a low bias [57].

Fig. 5. True estimation error $\hat{x}$ of each of the 150 tracking arcs in the single-station radar tracking scenario.
A common CV is the holdout CV that splits the original dataset into two parts (training and testing) and uses the testing score as a generalization measure [36]. It usually works well when the original dataset is large. However, due to the small size (less than 900 samples) of the OP error set, the k-fold CV [58] is more suitable for the error pattern learning in this study, as it generates a less biased estimate of the ML model than the holdout CV methods. In the k-fold CV process, the original OP error set is randomly partitioned into k disjoint sets with equal-sized subsamples. Of the k subsamples, a model is trained using the k - 1 subsamples (training data) while the remaining single subsample (validation or test data) is used to assess the performance of the trained model. Then, the CV is repeated for k times, with each of the k subsamples being used exactly once as the validation data. The performance metric of the trained model is finally computed as the average test error over all folds.

With \(e_A\) being the target variable to be modeled, a ML-predicted error \(e_{ML,A}\) can be computed by the learned error model \(Y = f(X)\) and then used to correct \(e_A\). Figs. 7-9 present the ML results of the BT and SVR methods on the training data for the test orbits. In each figure, the lower scatter plot shows the original error \(e_A\) (blue cross) and ML-predicted error \(e_{ML,A}\) (red dot), and the upper scatter plot presents the residual error \(\Delta e_A\) (green cross) after the error correction. Here, \(e_{ML,A}\) refers to BT-predicted error \(e_{BT,A}\) or SVR-predicted error \(e_{SVR,A}\).

In Figs. 7-9, it is clear that the ML-predicted error cluster \(e_{ML,A}\) and the original error cluster \(e_A\) are located closely, with their difference \(\Delta e_A\) being evenly distributed around the zero value within a boundary. Taking Larets orbit for example, the worst \(e_A\) is -6722.3m. However, after the error correction with the trained BT/SVR models, the worst residual error \(\Delta e_A\) reduces to -622.7m/-671.8m through removing \(e_{BT,A}/e_{SVR,A}\) from \(e_A\), respectively. A decrease of 90.7%/90.0% is obtained for the worst error by the trained BT/SVR models. To demonstrate the ML performance on the whole in Figs. 7-9, the \(P_{ML}\) values of the trained BT and SVR models on the training data are computed and used as the main metric to assess the performance of the trained BT and SVR models, as given in TABLE I.

### TABLE I: Performance of the Learned BT and SVR Models in Fitting the Training Data, in meters.

| Orbits | \(RMS_{e_A}\) | \(RMS_{\Delta e_A}\) | \(P_{ML}\) |
|--------|---------------|----------------|---------|
|        | BT | SVR | BT | SVR |
| Larets | 1342.6 | 181.6 | 234.7 | 86.5% | 82.5% |
| Starlette | 376.4 | 51.6 | 58.7 | 86.3% | 84.4% |
| HY-2A | 469.0 | 73.2 | 88.4 | 84.4% | 81.2% |

In TABLE I, the performance metric \(P_{ML}\) is computed by \(RMS_{e_A}\) and \(RMS_{\Delta e_A}\) according to its definition. It can be seen that \(RMS_{\Delta e_A}\) is significantly smaller than \(RMS_{e_A}\) and the computed \(P_{ML}\) value exceed 81% for all the trained BT and SVR models. Moreover, according to the metrics, the trained BT models show stronger learning capability than the trained SVR models on the same training data. Although such superiority of the trained BT models is small, it demonstrates the proposed BT method has the ability to train a ML model with lower bias. All in all, the developed ML framework is capable of capturing the dominating components of the underlying error pattern when only 30 days’ training data is available. If the learned pattern agrees well with the realistic patterns, it can predict accurate orbital errors beyond the training period, which will be evaluated in the following section.

### C. Improving the Future OP Accuracy With the Trained ML Model

From the application perspective, one is more concerned with the performance of the trained ML model with new data that is not used in the model learning. In this case, a desirable ML model should be learned with good generalization capability, that is, producing accurate predictions for unseen data. In terms of the OP accuracy improvement, after the trained error model is obtained from the historical OP error set, it can be generalized to future epochs to predict new error terms corresponding to the latest orbit predictions. Thus, the improved orbit predictions are achieved via error correction.

Now, the BT and SVR models, learned by the first 30 days’ training data in Figs. 7-9, are applied to predict the future 7 days’ orbital errors. As illustrated in Fig. 4, the physics-based orbit predictions during this period are used as the testing data to input the models. Figs. 10-12 plot the ML-predicted error \(e_{ML,A}\), the original error \(e_A\), and the difference between \(e_{ML,A}\) and \(e_A\), over the future 7 days (31st-37th day) for these three objects. In each figure, the generalization capability of the trained models can be seen from the performance metrics \(P_{ML}\).

We observed that in Figs. 10-12, the error predicted by the trained BT and SVR models with the testing data follows the same growing trend of the original error over
the time span of 31st-37th day. After correcting the predicted state with the ML-predicted error, the maximum absolute error of Larets/Starlette/HY-2A over 7 days reduces from 2186.3m/548.9m/396.7m to 465.7m/154.4m/103.5m for the trained BT model and 707.0m/145.5m/145.6m for the trained SVR model, respectively. The performance metric $P_{ML}$ of the trained BT/SVR model on the testing data reaches 79.7%/75.1% for Larets, 80.9%/78.5% for Starlette, and 75.2%/60.8%. It means that the ML-predicted errors compensate the majority of the real errors of the physics-based orbit predictions during the future 7 days. More importantly, it proves the effectiveness of the learned error pattern. Besides, it rationalizes that performances of the learned error models on the testing data are slightly worse than that on the training data, since these two datasets are from different time spans. Hence, the error pattern learned from the past period may be not exactly the same as that in the future period, because of the variations of the orbital environment such as the atmospheric density. Fortunately, the orbital environment does not change much within short terms, indicating that the orbital error evolutions during these two periods share the common pattern in most cases. As a result, the learned error pattern can predict the majority of the real OP errors in the future 7 days, as revealed in Figs. 10-12.

To further illustrate the effectiveness of the learned error models in predicting the future OP errors, the ML training and testing processings in a sliding time window of 50 days are performed. In this sliding time window, 100 combinations...
Fig. 9. Results of the trained BT and SVR models on training data (808 samples) for HY-2A.

Fig. 10. Future 7 days’ error predictions of the BT and SVR models learned by the first 30 days’ training data for Larets.

Fig. 11. Future 7 days’ error predictions of the BT and SVR models learned by the first 30 days’ training data for Starlette.

of the training and testing data are constructed, as illustrated in Fig. 4. That is, the combination of tracking arcs in one ML training and testing is different from other ML processings. For each dataset combination, a ML model is trained with the training data over the previous 30 days’ time span, and applied to the testing data in the following 7 days. Fig. 13 plots the RMS values of the future 7 days’ OP errors before and after the error correction with ML-predicted errors for the 100 ML
TABLE II: Distribution of $P_{ML}$ (in %) for the 100 ML Runs for the Testing Data Sets, in Number.

| Orbits | Model | $<0$ | 0-25% | 25%-50% | 50%-75% | 75%-100% |
|--------|-------|------|-------|----------|----------|----------|
| Larets | BT    | 5    | 8     | 7        | 35       | 45       |
|        | SVR   | 11   | 10    | 8        | 46       | 25       |
| Starlette | BT  | 8    | 4     | 6        | 28       | 54       |
|        | SVR   | 11   | 6     | 13       | 34       | 36       |
| HY-2A  | BT    | 7    | 8     | 9        | 32       | 44       |
|        | SVR   | 15   | 5     | 9        | 38       | 33       |

TABLE II summarizes the distribution of the performance metric $P_{ML}$ of the 100 ML runs for the three satellites. For Larets orbit, taking the trained BT models as examples, 95 combinations show the improvement of the OP results, while only 5 combinations show the worse results after error correction. After the examination of the OP results in Fig. 13, the original OP errors of these 5 combinations are all at the level of tens of meters. Since most of the OP errors in the training data are large, they have a dominant effect on the error pattern learning. Therefore, it is expectable that most of the large OP errors will be better modeled, and the few small errors will be modeled less accurately. On the other hand, a minor accuracy degradation in the originally accurate OP results is a worthy sacrifice for the overall OP accuracy improvement.

D. Effect of Training Data Size on the Performance Metric

More training data usually contributes to a better ML model. This is conditional on the assumption that the whole dataset follows the same distribution or shares the common underlying pattern. In practice, additional data could hardly improve the ML performance if it brings uncorrelated information or its related space weather condition in the training period and testing period changes heavily. To investigate the effect of training data size on the performance metric $P_{ML}$, a series of BT and SVR models are trained with different sizes of training data and then evaluated on the same testing data.

Fig. 14 gives an example of Larets orbit for the trained BT model using a training data size of 10 days, 20 days, 30 days, 40 days, 50 days, and 60 days. In Fig. 14, $e$ denotes the original OP error $e_A$ of the testing data; $e_{BT|10days}$ denotes the predicted error $e_{BT,A}$ using the BT model trained by 10 days’ training data, and so on for $e_{BT|20days}$, $e_{BT|30days}$, $e_{BT|40days}$, $e_{BT|50days}$, $e_{BT|60days}$.

As revealed in Fig. 14, the effect of the training data size on the performance of the trained ML model is demonstrated by using 60, 50, 40, 30, 20, and 10 days’ training data, with their corresponding performance metric $P_{ML}$ being 90.5%, 87.9%, 86.0%, 84.3%, 70.1%, 52.6%, respectively. To demonstrate more clear variations of $P_{ML}$, the training data size is varied from 10 days to 60 days with a step size of 5 days. 30 dataset combinations of training data and testing data are processed.
using 30 days’ training data in the previous experiments is further improved when a sufficient size of training data is used. The performance of the trained BT and SVR models will not be improved once the training data reaches and exceeds 30 days. This indicates the importance of the training data size for Larets orbit.

In Fig. 15, PMML increases quickly when the data size increases from 10 to 25 days for both the trained BT and SVR models, but tends to stabilize once the size of the training data reaches and exceeds 30 days. This indicates the performance of the trained BT and SVR models will not be further improved when a sufficient size of training data is used in the training process. It also rationalizes that our choice of using 30 days’ training data in the previous experiments is sufficient in achieving the expected ML performance.

V. Conclusion

Providing accurate OP information of space debris is vital in enhancing the SSA capabilities, like the reliable space conjunction warning. Towards accurate debris OP results with sparse observations from a single tracking station, this paper proposes a hybrid method of integrating the ML method into the physics-based OP system for improved orbit predictions of space debris when only sparse tracking data is available. This method displays the good generalization capability that the trained ML model is able to predict the future physics-based OP errors, which can be used to correct the physics-based OP errors.

This research has resulted in a complete computational framework that embeds the ML methods into the debris BLS OD/OP system. Two ML algorithms, including the BT method and the SVR method, are implemented to model the underlying pattern of past OP errors. Experimental results of three LEO satellites have demonstrated that both the trained BT and SVR models are capable of capturing most features of the underlying error pattern. The predicted errors by the trained BT and SVR models agree well with their truth values. By applying the model-predicted OP errors to correct the physics-based predicted states, the 7-day OP errors reduce from hundreds or even thousands of meters to tens of meters through the error correction with the learned error models, achieving at least 50% accuracy improvement.

The current research has assumed the sparseness of tracking data from a single station. In the next phase of the research, data sets from a regional network of several stations in China will be processed, in order to assess the achievable OP accuracy of tracked space objects through the developed ML computing framework. As part of the orbital knowledge, the orbital uncertainty or error covariance knowledge plays an important role in the precision space applications such as space collision probability computation and data association. The application of ML technique to the uncertainty propagation problem of space debris will be also studied in future work, thus providing the user groups with whole orbital information.

REFERENCES

[1] B. Lal, A. Balakrishnan, B. M. Caldwell, R. S. Buenconsejo, and S. A. Carion, “Global trends in space situational awareness (SSA) and space traffic management (STM),” 2018.
[2] T. Kelso, “Analysis of the iridium 33 cosmos 2251 collision,” Advances in the Astronautical Sciences, vol. 135, pp. 1099–1112, 2009.
[3] J.-C. Liou, “Collision activities in the future orbital debris environment,” Advances in Space Research, vol. 38, no. 9, pp. 2102–2106, 2006.
[4] X. R. Li and V. P. Jilkov, “Survey of maneuvering target tracking: part i. dynamic models,” IEEE Transactions on aerospace and electronic systems, vol. 39, no. 4, pp. 1333–1364, 2003.
[5] J. Bennett, J. Sang, C. Smith, and K. Zhang, “Accurate orbit predictions for debris orbit manoeuvre using ground-based lasers,” Advances in Space Research, vol. 52, no. 11, pp. 1876–1887, 2013.
[6] M. Vetrisano, N. Thiry, and M. Vasile, “Detumbling large space debris via laser ablation,” in 2015 IEEE Aerospace Conference. IEEE, 2015, pp. 1–10.
[7] J.-C. Liou, “An active debris removal parametric study for leo environment remediation,” Advances in Space Research, vol. 47, no. 11, pp. 1865–1876, 2011.
[8] Z. Meng, P. Huang, and J. Guo, “Approach modeling and control of an autonomous maneuverable space net,” Journal of Guidance, Control, and Dynamics, vol. 38, no. 9, pp. 2102–2106, 2006.
[9] A. Carioscia, “Global trends in space situational awareness (SSA) and space traffic management (STM),” Advances in Space Research, vol. 53, no. 6, pp. 2651–2661, 2017.
[10] F. R. Hoots and R. L. Roehrich, “Models for propagation of nomal element sets,” Aerospace Defense Command Peterson AFB CO Office of Astrodynamics, Tech. Rep., 1980.
[11] S. T. Goh and K.-S. Low, “Real-time estimation of satellite’s two-line elements via positioning data,” in 2018 IEEE Aerospace Conference. IEEE, 2018, pp. 1–7.
[12] J. Sang and J. C. Bennett, “Achievable debris orbit prediction accuracy using laser ranging data from a single station,” Advances in Space Research, vol. 54, no. 1, pp. 119–124, 2014.
[13] J. Raol and N. Sinha, “On the orbit determination problem,” IEEE Transactions on Aerospace and Electronic Systems, vol. 3, pp. 274–291, 1965.
[14] O. Montenbruck and E. Gill, Satellite orbits: models, methods and applications. Springer Science & Business Media, 2012.
[15] D. A. Vallado, Fundamentals of astrodynamics and applications. Springer Science & Business Media, 2001, vol. 12.
[16] P. Cefola, J. San Juan, S. Setty, R. Proulx et al., “Review of the draper semi-analytical satellite theory (dsst),” in AIAAC18: 18th Australian International Aerospace Congress (2019): HUMS-11th Defence Science and Technology (DST) International Conference on Health and Usage Monitoring (HUMS 2019): IFSSFD-27th International Symposium on Space Flight Dynamics (ISSFD). Engineers Australia, Royal Aeronautical Society, 2019, p. 1399.

[17] A. Awad, A. Narang-Siddarth, and R. Weisman, “Method of multiple scales for orbit propagation with nonconservative forces,” Journal of Guidance, Control, and Dynamics, vol. 40, no. 6, pp. 1457–1465, 2017.

[18] B. Li, J. Sang, and J. Ning, “A multiscaling-based semi-analytic orbit propagation method for the catalogue maintenance of space debris,” Journal of Spatial Science, pp. 1–23, 2018.

[19] Y. Lou, Y. Liu, C. Shi, B. Wang, X. Yao, and F. Zheng, “Precise orbit determination of beidou constellation: method comparison,” GPS solutions, vol. 20, no. 2, pp. 259–268, 2016.

[20] Y. Yang, X. Yue, and A. G. Dempster, “Gps-based onboard real-time orbit determination for leo satellites using consider kalman filters,” IEEE Transactions on Aerospace and Electronic Systems, vol. 52, no. 2, pp. 769–777, 2016.

[21] Y. Luo and Z. Yang, “A review of uncertainty propagation in orbital mechanics,” Progress in Aerospace Sciences, vol. 89, pp. 23–39, 2017.

[22] F. Piergentili, A. Ceruti, F. Rizzitelli, T. Cardona, M. L. Battagliere, and F. Santoni, “Space debris measurement using joint mid-latitude and equatorial optical observations,” IEEE Transactions on Aerospace and Electronic Systems, vol. 50, no. 1, pp. 664–675, 2014.

[23] D. F. Crouse, “On measurement-based light-time corrections for bistatic orbital debris tracking,” IEEE Transactions on Aerospace and Electronic Systems, vol. 51, no. 3, pp. 2502–2518, 2015.

[24] T. Castaing, B. Pannetier, F. Muller, and M. Rombaut, “Track initiation in low-earth-orbit objects using statistical modelling of sparse observations,” IEEE Transactions on Aerospace and Electronic Systems, vol. 51, no. 1, pp. 258–269, 2015.

[25] M. Grassi, E. Cetin, and A. G. Dempster, “Enabling orbit determination of space debris using narrowband radar,” IEEE Transactions on Aerospace and Electronic Systems, vol. 51, no. 2, pp. 1231–1240, 2015.

[26] J. Sang, J. C. Bennett, and C. Smith, “Experimental results of debris orbit predictions using sparse tracking data from mt. stromlo,” Acta Astronautica, vol. 102, pp. 258–268, 2014.

[27] J. C. Bennett, J. Sang, C. Smith, and K. Zhang, “An analysis of very short-arc orbit determination for low-earth objects using sparse optical and laser tracking data,” Advances in Space Research, vol. 55, no. 2, pp. 617–629, 2015.

[28] M. F. Storz, B. R. Bowman, M. J. J. Branson, S. J. Casali, and W. K. Tobiska, “High accuracy satellite drag model (basdm),” Advances in Space Research, vol. 36, no. 12, pp. 2497–2505, 2005.

[29] E. Doornbos, “Thermosphere density model calibration,” in Space Weather. Springer, 2007, pp. 107–114.

[30] C. Shi, W. Li, M. Li, Q. Zhao, and J. Sang, “Calibrating the scale of the nrlmsise-00 model during solar maximum using the two line elements dataset,” Advances in Space Research, vol. 56, no. 1, pp. 1–9, 2015.

[31] B. R. Bowman, W. K. Tobiska, F. A. Marcos, and C. Valladares, “The jh2006 empirical thermospheric density model,” Journal of Atmospheric and Solar-Terrestrial Physics, vol. 70, no. 5, pp. 774–793, 2008.

[32] J. Sang, C. Smith, and K. Zhang, “Towards accurate atmospheric mass density determination using precise positional information of space objects,” Advances in Space Research, vol. 49, no. 6, pp. 1088–1096, 2012.

[33] J. Chen and J. Sang, “Thermospheric mass density measurement from precise orbit ephemeris,” Geodesy and Geodynamics, vol. 7, no. 3, pp. 210–215, 2016.

[34] J. Sang, J. C. Bennett, and C. H. Smith, “Estimation of ballistic coefficients of low altitude debris objects from historical two line elements,” Advances in Space Research, vol. 52, no. 1, pp. 117–124, 2013.

[35] C. M. Bishop, Pattern recognition and machine learning. springer, 2006.

[36] R. Stuart and N. Peter, “Artificial intelligence—a modern approach 3rd ed.” 2016.

[37] W. D. Bennett, K. Zeliff, and J. Raquepas, “Classification of objects in geosynchronous earth orbit via light curve analysis,” in 2017 IEEE Symposium Series on Computational Intelligence (SSCI). IEEE, 2017, pp. 1–6.

[38] D. Izzo, D. Hennes, L. F. Simões, and M. Mårtens, “Designing complex interplanetary trajectories for the global trajectory optimization competitions,” in Space Engineering. Springer, 2016, pp. 151–176.

[39] A. Mereta, D. Izzo, and A. Wittig, “Machine learning of optimal low-thrust transfers between near-earth objects,” in International Conference on Hybrid Artificial Intelligence Systems. Springer, 2017, pp. 543–553.

[40] S. Sharma and J. W. Cutler, “Robust orbit determination and classification: A learning theoretic approach,” Interplanetary Network Progress Report, vol. 203, p. 1, 2018.

[41] B.-S. Lee, W.-G. Kim, J. Lee, and Y. Hwang, “Machine learning approach to initial orbit determination of unknown leo satellites,” in 2018 SpaceOps Conference, 2018, p. 2566.

[42] H. Peng and X. Bai, “Improving orbit prediction accuracy through supervised machine learning,” Advances in Space Research, vol. 61, no. 10, pp. 2628–2646, 2018.

[43] H. Peng and X. Bai, “Artificial neural network–based machine learning approach to improve orbit prediction accuracy,” Journal of Spacecraft and Rockets, vol. 55, no. 5, pp. 1248–1260, 2018.

[44] H. Peng and X. Bai, “Gaussian processes for improving orbit prediction accuracy,” Acta Astronautica, vol. 161, pp. 44–56, 2019.

[45] H. Peng and X. Bai, “Comparative evaluation of three machine learning algorithms on improving orbit prediction accuracy,” Astrodynamics, vol. 3, no. 4, pp. 325–343, 2019.

[46] R. L. Rickerle, “Consolidated laser ranging prediction format,” ILRS Data Format and Procedures Working Group, 2006.

[47] B. Li, J. Sang, and J. Chen, “Achievable orbit determination and prediction accuracy using short-arc space-based observations of space debris,” Advances in Space Research, vol. 62, no. 11, pp. 3065–3077, 2018.

[48] B. Li, J. Sang, and H. Liu, “Accurate propagation of debris orbit error via dynamic calibration and its cataloguing,” Advances in Space Research, vol. 63, no. 8, pp. 2422–2435, 2019.

[49] J. Picone, A. Hedin, D. P. Drob, and A. Aikin, “Nrlmsise-00 empirical model of the atmosphere: Statistical comparisons and scientific issues,” Journal of Geophysical Research: Space Physics, vol. 107, no. A12, pp. SIA–15, 2002.

[50] D. M. Hawkins, “The problem of overfitting,” Journal of chemical information and computer sciences, vol. 44, no. 1, pp. 1–12, 2004.

[51] V. Vapnik, The nature of statistical learning theory. Springer science & business media, 2013.

[52] A. J. Smola and B. Schölkopf, “A tutorial on support vector regression,” Statistics and computing, vol. 14, no. 3, pp. 199–222, 2004.

[53] J. Friedman, T. Hastie, R. Tibshirani et al., “Additive logistic regression: a statistical view of boosting (with discussion and a rejoinder by the authors),” The annals of statistics, vol. 28, no. 2, pp. 337–407, 2000.

[54] Y. Freund and R. E. Schapire, “A decision-theoretic generalization of on-line learning and an application to boosting,” Journal of computer and system sciences, vol. 55, no. 1, pp. 119–139, 1997.

[55] J. H. Friedman, “Greedy function approximation: a gradient boosting machine,” Annals of statistics, pp. 1189–1232, 2001.

[56] M. R. Pearlman, J. J. Degnan, and J. Bosworth, “The international laser ranging service,” Advances in Space Research, vol. 30, no. 2, pp. 135–143, 2002.

[57] S. Arlot, A. Celisse et al., “A survey of cross-validation procedures for model selection,” Statistics surveys, vol. 4, pp. 40–79, 2010.

[58] T.-T. Wong, “Performance evaluation of classification algorithms by k-fold and leave-one-out cross validation,” Pattern Recognition, vol. 48, no. 9, pp. 2839–2846, 2015.
Jian Huang was born in Suizhou, Hubei, China in 1986. He received the B.S degree in electronic engineering in 2007, the M.S degree in communications engineering in 2009, and the Ph.D degree in information and communications engineering in 2013, all from National University of Defense Technology, China. He has been an Assistant Research Fellow with Beijing Institute of Tracking and Telecommunications Technology since 2013. His research interests include space objects surveillance system design and information processing.

Yanming Feng received the Ph.D. degree in satellite geodesy from Wuhan Technical University of Surveying and Mapping (now part of Wuhan University), Wuhan, China. He is currently a Professor in Data Science at the School of Electrical Engineering and Computer, Queensland University of Technology, Brisbane QLD, Australia. His research interests and publications cover the areas of satellite orbit determination, GNSS models and algorithms and integrity determination, vehicular communications and time synchronization, Internet of Things, and GNSS solutions and data analytics.

Fuhong Wang received the B.S. degree in engineering surveying from the Jiaozuo Institute of Technology, Jiaozuo, China, in 1986, and the M.S. and Ph.D. degrees in geodesy and surveying engineering from Wuhan University, Wuhan, China, in 2000 and 2006, respectively, where he is currently a Professor with the School of Geodesy and Geomatics. His research interests include space-borne GPS high-precision autonomous orbit determination theory and method, and GPS/INS integrated navigation.

Jizhang Sang was born in Yuyao, Zhejiang, China in 1963. He received the B.S. and M.S. degrees in geodesy from Wuhan Technical University of Surveying and Mapping (now part of Wuhan University), Wuhan, China, in 1983 and 1986, respectively, and the Ph.D degree in satellite navigation from Queensland University of Technology, Brisbane QLD, Australia in 1997. He was an Associate Lecturer from 1986 to 1990, and a Lecturer from 1990 to 1993, both at Department of Geodesy, Wuhan Technical University of Surveying and Mapping, China. He was a Post-doctoral Research Fellow with Space Centre of Satellite Navigation, Queensland University of Technology, Australia, from 1996 to 1998. From 1998 to 2013, he was a Senior/Principal Research Engineer at EOS Space Systems, Canberra, Australia. Since 2013, he has been a Professor with School of Geodesy and Geomatics, Wuhan University, China. He has published widely in geodesy, satellite navigation, and space debris orbit determination. His research interests include space debris surveillance, orbit determination, atmospheric mass density modeling, space geodesy, and satellite navigation.