Research on Air Traffic Flow Forecast Based on ELM Non-Iterative Algorithm

Zhaoyue Zhang¹ · An Zhang¹ · Cong Sun¹ · Shuaida Xiang² · Jichen Guan³ · Xuedong Huang²

Accepted: 20 October 2020 / Published online: 6 November 2020
© Springer Science+Business Media, LLC, part of Springer Nature 2020

Abstract

In this paper, the chaotic characteristics of air traffic flow are studied, ADS-B data easily available to ground aviation users are selected as the basic data of traffic flow, and a high-dimensional prediction model of air traffic flow time series based on the non-iterative PSR-ELM algorithm is established. The prediction results of the proposed algorithm are then compared with those of the SVR algorithm, which requires iteration. Moreover, airspace operation data before and after the outbreak of the COVID-19 epidemic are selected as the experimental scene, and the prediction effects of time series with different degrees of chaos are comparatively analyzed. The experimental results reveal that the PSR-ELM algorithm achieves fast and accurate results, and, when the traffic flow state is sparse, the degree of chaos is reduced and the prediction effect is improved. The findings of this research provide a reference for air traffic flow theory.

Keywords Air traffic flow · Chaotic · Time-series prediction · ELM · ADS-B data

1 Introduction

Short-term air traffic flow prediction is essential in the field of air traffic flow management. Many researchers have focused on traffic flow prediction to aid in management and planning, especially in the civil aviation domain. As a result, numerous relevant methods have been proposed in academic literature. These methods can be classified into three general categories [1, 2], namely statistical methods, model-based methods, and machine learning-based methods.

Due to their outstanding performance in reflecting nonlinear relationships between input and output data, many machine learning algorithms, including the random forest (RN), support vector machine (SVM) [3], and artificial neural network (ANN) [4] algorithms, have been developed to solve nonlinear problems.

The extreme learning machine (ELM), which is extensively used in the short-term prediction domain, has been introduced for the training of single-layer feed-forward neural networks (SLFNs), which randomly choose hidden nodes and the output weight. Thus, the ELM method is superior to the traditional backpropagation (BP) learning algorithm and the SLFN. Due to its significant performance, the ELM method has been widely applied in different fields [5, 6]. The first online ELM method created by Huang et al. was the online sequential ELM (OS-ELM) [7]. In theory, this algorithm, which uses an incremental construction method to simplify the updating process of hidden-layer nodes, tends to provide outstanding generalization performance at an extremely fast learning speed. In 2007, Chen et al. [8] took into account the training complexity of the network and proposed the convex incremental ELM (CI-ELM). This method can generate a universal single hidden layer according to any continuous sampling distribution with a faster convergence speed. Feng et al. [9] proposed the error-minimized ELM (EM-ELM) for the automatic determination of network architectures. This method can add random hidden nodes to SLFNs in groups according to the number of samples in the data segment, which significantly reduces the computational complexity. Rong et al. [10] used the fuzzy inference system (FIS) to calculate the membership function of neural network parameters, then developed the OS-Fuzzy-ELM method, which has a significantly reduced training time. To improve the adaptability to

✉ Zhaoyue Zhang
zy_zhang@cauc.edu.cn

¹ School of Aeronautics, Northwestern Polytechnical University, Xi'an 710072, China

² College of Air Traffic Management, Civil Aviation University of China, Tianjin 300300, China

³ College of Civil Aviation, Nanjing University of Aeronautics & Astronautics, Nanjing 211106, China

unknown data, Miche et al. [11] proposed an optimally-pruned ELM (OP-ELM) based on a two-step regularization penalty, in which L1 and L2 norms are respectively used to simplify the hidden-layer nodes and output-layer weights. Zhang et al. [12] used a sparse ELM as an alternative solution for classification, thereby reducing the storage space and testing time. Moreover, kernel matrix optimization was introduced in this method to decompose the quadratic programming problem into multiple small sub-problems, thereby enhancing the processing capacity of the ELM method for large-scale data. Zhang et al. [13] proposed an adaptive dynamic ELM (D-ELM) in which the hidden nodes can be dynamically recruited or deleted according to their significance to the network performance; this method combines the advantages of EM-ELM and AG-ELM [14]. Even if the objective function is not explicitly expressed, the generalization approximation ability can still be guaranteed under a minimum error condition. Liu et al. introduced the forgetting parameters ELM (FP-ELM) to measure the effect of known data on the current data segment, and quantified the degree of concept drift in the online learning environment [15]. Moreover, this method can be used to avoid estimator windup. In general, the methods developed in these previous studies resulted in improved prediction model adaptability and parameter adjustment according to the arriving data to avoid the retraining of all the data.

The chaotic dynamic characteristics of ground traffic flow have received substantial research attention in recent decades. In 1984, Disbro et al. first introduced chaos theory into the traffic system [16]. Later, Low et al. examined the concept of chaotic behavior in a deterministic car-following model [17], and Tang et al. improved the chaos forecasting method to render it effective in forecasting traffic conflict flow [18].

With the development of ground traffic flow theory, scholars began to explore the nonlinear characteristics of air traffic systems. The investigation of the time series of air traffic flow based on measured data is an effective method by which to study its nonlinear characteristics. Frank et al. found that the calculation of the Lyapunov exponent provides a clear indication of chaos [19].

Li et al. proposed an improved maximum Lyapunov exponent algorithm based on the small-data-volume method and wavelet noise reduction theory, identified the chaotic characteristics in flight conflict time series, and verified the feasibility of the application of chaos theory to flight conflict prediction [20]. Cong et al. used traffic flow time-series data in the airspace sector to study the chaotic characteristics of the air traffic system; they solved the correlation dimension and the maximum Lyapunov index of traffic flow time series with the G-P algorithm and the small-data-volume method, ultimately proving the existence of chaos and fractal characteristics in air traffic flow time series [21].

Due to the nonlinear characteristic of traffic, the ELM theory has also been applied in the field of intelligent transportation systems (ITSs) [22], which effectively and proactively play crucial roles in the operation of traffic management systems and dynamic traffic assignment. Ban et al. [23] implemented the ELM method to design a real-time traffic index for real large-scale traffic data to predict congestion, and the experimental results indicated an outstanding generalization performance as compared with existing state-of-art algorithms. Ma et al. [24] improved the applicability of OS-ELM by involving sequential updating and network reconstruction for traffic flow forecasting; this model can be adaptively updated via correction with real-time trajectories. Wang et al. [25] applied the ensemble real-time sequential ELM (ERS-ELM) with an SLFN structure for traffic flow prediction under a peak freeway traffic condition and non-stationary condition. The model was found to achieve excellent prediction performance on large traffic volume datasets collected from urban intersection arteries with a shorter training time and sufficient accuracy. Zhang et al. [26] found that ELMs with heterogeneous data exhibit improvements over linear models in terms of both the level and directional accuracy when handling traffic flow time-series data. Shang et al. [27] proposed a short-term traffic flow prediction model called SSA-KELM (singular spectrum analysis kernel ELM) to reduce the influences of uncertainty and nonlinearity on the expressway system.

Most studies relevant to traffic prediction that utilized the ELM method were focused on ground traffic, whereas the air traffic domain, which is characterized by much more chaos and less predictability, has been neglected. The input format of the ELM model affects its performance and training speed. Phase-space reconstruction (PSR) [28] is the basis of chaotic time-series analysis, and significantly dominates the prediction performance. It can be used to map scalar time series to the multi-dimensional phase space and thoroughly mine the implicit information. The delay time and embedding dimension are the key parameters of PSR, and there currently exist many methods for choosing these two parameters. The methods that can be used to calculate the delay time include the autocorrelation function method [29], the mutual information method [30], and the average displacement method [31]. The methods that can be used to calculate the embedding dimension include the G-P method [32], the false nearest neighbors method [28], and the CAO method [33]. However, the C-C method [34] is different from these methods, and can simultaneously estimate the delay time and embedding dimension based on statistical results. The advantages of the C-C method are its small required amount of calculation, strong anti-interference ability, and easy use. Thus, the C-C method was selected for use in the present study to determine the two parameters for PSR. Additionally, the phase-space reconstruction ELM (PSR-ELM) method was employed to analyze air traffic flow data.

The aims of this research are (I) to use PSR to establish the mapping relationship between time series of high and low dimensions to improve the ELM and achieve better experimental results; (II) to compare the performances of the ELM, PSR-ELM, and SVR algorithms in the field of air traffic flow prediction in terms of their error and time consumption; (III) to analyze, for the first time, the changes of the chaos characteristics of air traffic flow and evaluate the prediction effect in different periods based on the impact of COVID-19, and to accordingly provide strong support for air traffic flow theory.

The remainder of this work is organized as follows. Related work is discussed in detail in the subsequent section, including research on the nonlinear characteristics of air traffic, the forecast of air traffic flow, and the related use of the deep learning algorithm. In Section 3, the relationship between the flow state and the degree of chaos of the air traffic flow time series is proven. Section 4 provides descriptions of the methodologies of the ELM, PSR-ELM, and SVR algorithms. The precision and time consumption of the three methods are investigated in Section 5. In Section 6, the relationship between air traffic flow predictability and the flow state is verified. Finally, Section 7 outlines the conclusions of this work.

2 Research on the Chaos characteristics of air traffic flow time series

2.1 Extraction of traffic time series

To benchmark the proposed methodology, the ADS-B (automatic dependent surveillance-broadcast) flight trajectory data of the east China waypoint P449 for one week before and after the outbreak of the COVID-19 epidemic (December 2019 and March 2020) were utilized for experimental analysis. After decoding, the ADS-B data contained 10 parameters, namely the date, time, ICAO address code, flight number, latitude, longitude, altitude, speed, course, and rate of altitude increase/decrease. The time interval of the data is 1 s. Table 1 reports the aircraft trajectory data from December 7, 2019.

The original ADS-B trajectory data have many noise points and abnormal loss points, and the data quality is not high; thus, the preprocessing of the data set was required, and the workflow is presented in Fig. 1. The first step was to split the multiple trajectories that may have generated new scattered trajectories, and then to delete them altogether. In the case of sufficient data points, the distance between two adjacent trajectory points in a trajectory of the aircraft will not be more than 20 km, but, based on the distance alone, the data may be defective; thus, they cannot be readily separated. However, if the time between two adjacent trajectory points exceeds 100 s, and if the average speed is greater than 120 km/h, then the distance is also more than 20 km, and it can be considered as multiple data separation. In this case, interpolation will supplement nearly 100 points. After calculating the average number of trajectory points, each trajectory was found to have only 200–300 trajectory points on average. In this case, interpolation will lead to trajectory distortion, so the data were not considered to be defective at this time. After the dispersion of the trajectory data, the scattered trajectory data were screened. In the same research area, although the numbers of trajectory points of different trajectory data were different, there was no significant difference (a difference of 10 times or more). The number of trajectory points was counted and averaged, and one-tenth of them was taken as the threshold value. Incomplete data were deleted as scattered trajectory data. The acute angles in the trajectory were then identified, the misplaced data points were determined for smoothing processing, and the defective data (data for which the distance between adjacent trajectory points was more than 10 km) were ultimately identified for linear interpolation processing.

After data cleaning, the air traffic flow trajectories near waypoint P449 on December 14, 2019 and March 14, 2020 were extracted, as shown in Fig. 2, and the time series of traffic passing through the waypoint were extracted, as shown in Fig. 3. It is evident that the air traffic in March 2020 was affected by the COVID-19 epidemic (Fig. 3(b)), and the number of flight operations was greatly reduced by 58.3%, which can be regarded as a low-flow-level operating state. On the contrary, December 2019 represents the normal operating state (Fig. 3(a)). The

Table 1 ADS-B data format

Date/Time	Flight number	Latitude (°)	Longitude (°)	...	$V_{\perp}$ (ft/min)
2019-04-01 00:00:00	CSZ9123	116.6789843	39.1948700	.	128
2019-04-01 00:00:00	CHH7136	116.5876903	39.2397766	.	64
...					
2019-04-01 23:58:16	CSN3191	116.5701957	39.2841797		−64

Fig. 1 Data processing

considered dataset is moderate in size, and the obtained results are convincing.

2.2 Comparison of the degrees of Chaos of different flow states

In recent years, chaotic time-series analysis methods have been extensively used in many scientific research and engineering fields. There are many methods by which to determine the chaotic properties of time series, and they primarily rely on the maximum Lyapunov exponent λ . The Lyapunov exponent is an important indicator by which to measure the dynamics of a system, and it represents the average exponential rate of

system convergence or divergence between adjacent orbits in phase space. If the value is greater than 0, it indicates that the system is in a chaotic state. For the calculation of the Lyapunov exponent in time-series research, the Wolf method and the small-data-amount method are more applicable.

To calculate the Lyapunov exponent of air traffic flow time series, it is first necessary to reconstruct the phase space. Consider a univariate time series $\{Tr_f(t), t = 1, 2, \dots, n\}$, where n is the length of the time series. The phase space is reconstructed with the delay time m and embedding dimension τ to obtain a new time series $Tr_f^{new}(t) = \{Tr_f(t), Tr_f(t + \tau), \dots, Tr_f(t + (m - 1)\tau)\}$. Therefore, the choice of m and τ is very important; if chosen incorrectly, the information

Fig. 2 Aircraft trajectories near waypoint P449

contained in the original time series cannot be fully mined. According to Kim et al., there is a certain correlation between m and τ , so they should be determined simultaneously. While the C-C algorithm has been proposed for this task, in view of its shortcomings, some scholars have proposed improvement measures [35].

The time series of air traffic flow that passed through waypoint P449 from December 14 to December 16, 2019, and from March 14 to March 16, 2020, were extracted. The Wolf method was used to calculate the maximum Lyapunov exponent, and the results are reported and compared in Table 2.

It can be observed from the calculation results presented in Table 2 that the larger the time scale, the larger the maximum Lyapunov exponent, indicating that the degree of chaos in the system was also greater. By comparing different research subjects on the same time scale, it is evident that the overall degree of chaos from March 14–16, 2020 was weaker than that from December 14–16, 2019. After the outbreak of COVID-19, the air traffic flow became sparse, and the degree of chaos was weakened.

To verify the accuracy of these results, all the air traffic flow that passed the North China VYK navigation station and the East China waypoint BONGI from December 14 to December 16, 2019, and from March 14 to March 16, 2020, was selected as research subjects. The maximum Lyapunov exponent was calculated at different time scales, and the results are presented in Tables 3 and 4. Similar to the previous results, it was found that the air traffic flow became sparse and

the degree of chaos was weakened after the outbreak of COVID-19.

Different traffic states affected the chaotic characteristics. The traffic state in the peak period was relatively dense and congested, and the degree of chaos at this time was stronger. This reflects the air traffic flow in the dense traffic state, for which uncertainty and unpredictability were more prominent.

2.3 Degree of Chaos and predictable scale of time series

Due to the sensitivity of the chaotic system to the initial value, a small error in the initial value will cause a large error in the predicted value of the future state. Moreover, as the prediction time continues to increase, the prediction error will gradually increase; thus, only short-term predictions of chaotic systems can be made. Yue et al. focused on the time scale of the future state prediction of chaotic systems, and determined the average and longest predictable scales of chaotic time series [36].

The Lyapunov exponent can not only characterize the chaos of a system, but can also indicate the degree of divergence or convergence of the system's adjacent orbits. The maximum predictable scale of the system is related to the reciprocal of the largest Lyapunov exponent.

The predictability of a nonlinear system will fall within a range. Because the calculation method of the maximum Lyapunov index is not unique, it is difficult to accurately measure the predictability of the system; however, the larger the maximum Lyapunov exponent, the worse the

Fig. 3 Air traffic flow in different periods

Table 2 Comparison of the degrees of chaos in the time series of air traffic passing waypoint P449

Sample time maximum	Time scale	Lyapunov exponent λ
December 14–16, 2019	2 min	0.053890
March 14–16, 2020	2 min	0.019699
December 14–16, 2019	5 min	0.118728
March 14–16, 2020	5 min	0.094058
December 14–16, 2019	10 min	0.121855
March 14–16, 2020	10 min	0.097164
December 14–16, 2019	15 min	0.275710
March 14–16, 2020	15 min	0.235692

Table 3 Comparison of the degrees of chaos of the time series of air traffic passing the VYK navigation station

Sample time maximum	Time scale	Lyapunov exponent λ
December 14–16, 2019	2 min	0.053890
March 14–16, 2020	2 min	0.019699
December 14–16, 2019	5 min	0.118728
March 14–16, 2020	5 min	0.094058
December 14–16, 2019	10 min	0.121855
March 14–16, 2020	10 min	0.097164
December 14–16, 2019	15 min	0.275710
March 14–16, 2020	15 min	0.235692

predictability of the system. Here, predictability refers to the prediction steps of the system itself.

For two nonlinear systems, the comparison of the respective Lyapunov exponents can reveal the degree of chaos in the system, but it is not sufficient to determine the system's prediction results. In the following section, it is discussed whether there is a relationship between the flow state and the prediction accuracy of air traffic flow time series.

3 Methodology

3.1 Extreme learning machine model

The ELM model is a special feedforward neural network. Because its training process does not require iterative calculation, its operation speed is greatly improved, and the network has good generalization performance. It has been proven that this model can perform better than popular gradient-based approaches in the cases of overfitting, generalization, and local minima.

The structure of a typical ELM is illustrated in Fig. 4. The network consists of an input layer, a hidden layer, and an

Table 4 Comparison of the degrees of chaos of the time series of air traffic passing waypoint BONGI

Sample time maximum	Time scale	Lyapunov exponent λ
December 14–16, 2019	2 min	0.121590
March 14–16, 2020	2 min	0.004231
December 14–16, 2019	5 min	0.315970
March 14–16, 2020	5 min	0.021881
December 14–16, 2019	10 min	0.715972
March 14–16, 2020	10 min	0.053267
December 14–16, 2019	15 min	0.760721
March 14–16, 2020	15 min	0.149549

Fig. 4 The structure of a typical ELM

output layer. The neurons in the input layer and hidden layer, and those in the hidden layer and output layer, are fully connected.

The input layer has n neurons corresponding to n input variables, the hidden layer has l neurons, and the output layer has m neurons corresponding to m output variables. Without a loss of generality, $\mathbf{w}$ is the connection weight between the input layer and the hidden layer, β is the connection weight between the hidden layer and the output layer, and $\mathbf{b}$ is the threshold of hidden-layer neurons, as follows:

$$\mathbf{w} = \begin{bmatrix} w_{11} & w_{12} & w_{13} & \cdots & w_{1n} \\ w_{21} & w_{22} & w_{23} & \cdots & w_{2n} \\ w_{31} & w_{32} & w_{33} & \cdots & w_{3n} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ w_{l1} & w_{l2} & w_{l3} & \cdots & w_{ln} \end{bmatrix}_{l \times n} \quad (1)$$

$$\beta = \begin{bmatrix} \beta_{11} & \beta_{12} & \cdots & \beta_{1m} \\ \beta_{21} & \beta_{22} & \cdots & \beta_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ \beta_{l1} & \beta_{l2} & \cdots & \beta_{lm} \end{bmatrix}_{l \times m} \quad (2)$$

$$\mathbf{b} = \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_l \end{bmatrix}_{l \times 1} \quad (3)$$

where w_{ji} is the connection weight between the i -th neuron of the input layer and the j -th neuron of the hidden layer.

If the activation function of the hidden-layer neuron is $g(x)$, then it can be determined from Fig. 4 that the output of the network is $\mathbf{T}$:

$$\mathbf{T} = [t_1, t_2, \dots, t_q]_{m \times q} \quad (4)$$

$$t_j = \begin{bmatrix} t_{1j} \\ t_{2j} \\ \vdots \\ t_{mj} \end{bmatrix}_{m \times 1} = \begin{bmatrix} \sum_{i=1}^l \beta_{i1} g(w_i x_j + b_i) \\ \sum_{i=1}^l \beta_{i2} g(w_i x_j + b_i) \\ \vdots \\ \sum_{i=1}^l \beta_{im} g(w_i x_j + b_i) \end{bmatrix}_{m \times 1} \quad j = 1, 2, \dots, q \quad (5)$$

where $\mathbf{w}_i = [w_{i1}, w_{i2}, \dots, w_{in}]$ and $\mathbf{x}_j = [x_{1j}, x_{2j}, \dots, x_{nj}]^T$.

$$\mathbf{H}\beta = \mathbf{T}' \quad (6)$$

where $\mathbf{T}'$ is the transpose of the matrix $\mathbf{T}$, and the hidden-layer output matrix $\mathbf{H}$, called the neural network, is

$$\mathbf{H}(\mathbf{w}_1, \mathbf{w}_2, \dots, \mathbf{w}_l, \mathbf{b}_1, \mathbf{b}_2, \dots, \mathbf{b}_l, \mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_q) = \begin{bmatrix} g(\mathbf{w}_1 \mathbf{x}_1 + \mathbf{b}_1) & g(\mathbf{w}_2 \mathbf{x}_1 + \mathbf{b}_2) & g(\mathbf{w}_l \mathbf{x}_1 + \mathbf{b}_l) \\ g(\mathbf{w}_1 \mathbf{x}_2 + \mathbf{b}_1) & g(\mathbf{w}_2 \mathbf{x}_2 + \mathbf{b}_2) & g(\mathbf{w}_l \mathbf{x}_2 + \mathbf{b}_l) \\ \vdots & \vdots & \vdots \\ g(\mathbf{w}_1 \mathbf{x}_q + \mathbf{b}_1) & g(\mathbf{w}_2 \mathbf{x}_q + \mathbf{b}_2) & g(\mathbf{w}_l \mathbf{x}_q + \mathbf{b}_l) \end{bmatrix}_{q \times l} \quad (7)$$

Based on prior studies, Huang et al. proposed that, given any number of different samples (x_i, y_i) $i = 1, 2, 3, \dots, q$, and an infinitely differentiable activation function $g: \mathbf{R} \rightarrow \mathbf{R}$ in any interval, there is always an SLFN with $K^{(K \leq q)}$ hidden-layer neurons. In the case of any assignment $w_i \in \mathbf{R}$ and $b_i \in \mathbf{R}$, there is $\|\mathbf{H}\beta - \mathbf{T}'\| < \varepsilon$. Therefore, when the activation function

$g(x)$ is infinitely differentiable, the parameters of the SLFN do not need to be fully adjusted, and w and b can be randomly selected before training and remain unchanged during the training process. The connection weight β between the hidden layer and the output layer can be obtained by solving the least-squares solution of the following system of equations:

$$\min_{\beta} \|H\beta - T'\| \Rightarrow \hat{\beta} = H^{-1} T' \quad (8)$$

where H^{-1} is the Moore-Penrose generalized inverse of the hidden-layer output matrix.

The basic principle of time-series prediction is a regression prediction method, in which the continuity of the development of things is recognized, past time-series data are used for statistical analysis, and the development trends of things are inferred; moreover, randomness due to the influence of accidental factors is fully considered. To eliminate the impact of random fluctuations, historical data are used for statistical analysis, and the data are appropriately processed for trend prediction. Traditional time-series prediction and time-series prediction after PSR are respectively given by Eqs. (9) and (10):

$$Tr_f(t+1) = F(Tr_f(t), Tr_f(t-1), \dots, Tr_f(t-k)) \quad (9)$$

$$Tr_f_{predict}^{new}(t+1) = F'(Tr_f^{new}(t)) \quad (10)$$

where t is time, k is the delay length, i.e., the length of the historical time series that affects the current input, i.e., the number of neurons in the input layer of the network, and $F(\cdot)$ is the mapping function. The steps of ELM and PSR-ELM are listed in Algorithm 1.

3.2 Support vector regression model

The support vector machine (SVM) was proposed for binary classification problems, and SVR (support vector regression) is an important application branch of SVMs. The difference between SVR and SVM classification is that the sample points of SVR are ultimately of one type. The optimal hyperplane sought by SVR is not the “most open,” as it is for SVM, which divides two or more types of sample points; instead, it makes all samples.

For a given set of training samples $\{(x_i, y_i) | x_i \in R^N, y_i \in R, i = 1, 2, 3, \dots\}$, the regression problem can be attributed to finding a function $f(x)$ so that the error between the function value $f(x_i)$ and the expected value y_i in the training sample is not greater than a given value of ε .

Supposing that $f(x) = w^T \varphi(x) + b$, with $w, x \in R^N, b \in R$, SVR can be expressed as the following planning problem:

$$\begin{aligned} \min & \frac{1}{2} \|w\|^2 + C \sum_{i=1}^l (\zeta_i + \zeta_i^*) \\ \text{s.t.} & \begin{cases} w^T \varphi(x_i) + b - y_i \leq \varepsilon + \zeta_i, \\ y_i - w^T \varphi(x_i) - b \leq \varepsilon + \zeta_i^*, \\ \zeta_i, \zeta_i^* \geq 0, i = 1, 2, 3, \dots, l. \end{cases} \end{aligned} \quad (11)$$

where C, ε , and $\zeta_i(\zeta_i^*)$ are respectively the trade-off cost between the empirical error and the flatness, the size of the ε -tube, and slack variables. With the use of the duality principle, the Lagrange multipliers α_i, α_j^* , and the kernel function, Eq. (12) is transformed into a duality problem, as follows.

Algorithm 1 ELM and PSR-ELM

Input:	ELM: $Tr_f(t)$	PSR-ELM: $Tr_f^{new}(t)$
Output:	ELM: $Tr_f(t+1)$	PSR-ELM: $Tr_f_{predict}^{new}(t+1)$
1:	The parameters w and b of hidden neurons are initialized at random;	
	The hidden-layer output matrix H is calculated according to Eq.	
Train 2:	(16) by the values of w_i and b_i determined by the previous step and $g(x)$;	
	The output weights are computed by applying T target values to	
3:	Eq. (19);	
Predict 4:	The output vector T is calculated according to Eqs. (15) and (16).	

$$\min \frac{1}{2} \sum_i \sum_j (\alpha_i^* - \alpha_i) (\alpha_j^* - \alpha_j) K(x_i, x_j) + \varepsilon \sum_i (\alpha_i^* - \alpha_i) - \sum_i y_i (\alpha_i^* - \alpha_i) \quad s.t. \begin{cases} \sum_i y_i (\alpha_i^* - \alpha_i) = 0, \\ 0 \leq \alpha_i^*, \alpha_i \leq C. \end{cases} \quad (12)$$

Then, $w = \sum_i (\alpha_i^* - \alpha_i) \varphi(x_i)$, and, if $0 < \alpha_i < C$, the following can be acquired: $b = y_i - w^T \varphi(x_i) - \varepsilon$.

The final function is

$$f(z) = \sum_{i=1}^l (\alpha_i^* - \alpha_i) \cdot K(x_i, z) + b \quad (13)$$

Three parameters control the quality of SVR, namely the error cost C , the width of the tube ε , and the kernel parameter. Here, ε is a prior-given value that defines the ε -insensitive loss function, and the value of ε controls the number of support vectors. The wider the tube ε , the fewer the number of support vectors, and the approximation function will not adapt to a real function. A new problem, therefore, is determining how to select the value of ε .

4 Comparison of the prediction results of different models

4.1 Selection of experimental objects and indicators

The 10-min time series of the air traffic flow passing waypoint P449 from December 14 to 20, 2019 (a total of 1008 sets of data) was extracted from the traffic flow database. To verify the universality of the model's prediction results, two sets of experiments were conducted.

- Using the data of the four days before the training to predict the fifth.

There were 576 sets of data in the first four days, and each day there was a period in which no flights passed through the waypoint. To intuitively reflect the accuracy of the prediction model, it was verified whether the model could capture the daily zero-flow period. 600 sets of data before training and after the test 150 sets of data.

- There was a total of 951 sets of data six days before training and the seventh day before 14:30, and a total of 57 sets of data after the seventh day of the test at 14:30.

The calculation time of the model may have a stronger relationship with the determination of the parameters. To obtain high-precision and realistic results, the algorithm parameters must be adjusted. By adjusting the precision parameters, the experimental results can be changed to select appropriate

results for evaluation and analysis; this reflects the multi-resolution of the algorithm. Calculation Example A was taken as an example, and the parameters of PSR-ELM and SVR that had a greater influence on the calculation were recorded. Compared with ELM, PSR-ELM includes an additional PSR step, and the calculation of m and τ takes a long time. SVR requires the determination of the optimal parameters c and g during the calculation. The comparison results are presented in Table 5.

The average calculation time of the PSR-ELM model was 6.421 s, and the average calculation time required for the SVR model to find the best values of c and g was 165.877 s, except for the heavy requirements of the two models, the calculation time are both about 0.17 s. However, the PSR-ELM model simply needs to calculate the optimal PSR parameters, and it therefore has an absolute advantage in practical applications.

4.2 Comparative analysis

The proposed PSR-ELM was evaluated by a series of experiments designed to benchmark it against non-randomized approaches. All programs were explored using MATLAB 2019b, and were run on a mainframe with 16 GB of RAM, an AMD 6-core processor, and a Windows 10 OS. To derive fair comparisons, the statistical setup was kept similar to the competent schemes. For all experiments involving training and testing, 10-fold cross-validation was performed.

To facilitate the performance comparison between the proposed approach and other competent approaches, several evaluation metrics, including time consumption and precision, were taken into consideration. Moreover, the two measures of MSE (mean square error) and R^2 (coefficient of determination) were used to evaluate the proposed model and its counterparts; the better prediction effect, the smaller the value of MSE, and the larger value of R^2 .

$$MSE = \frac{1}{m} \sum_{i=1}^m (\hat{y}(i) - y(i))^2 \quad (14)$$

$$R^2 = 1 - \frac{\sum_{i=1}^m (\hat{y}(i) - y(i))^2}{\sum_{i=1}^m (y(i) - \bar{y})^2} \quad (15)$$

Table 5 Comparison of prediction models

Model	Example A		Example B	
	MSE	t (s)	MSE	t (s)
ELM	1.2467	0.136728	2.4737	0.221724
PSR-ELM	0.62	6.583368	1.0893	12.045819
SVR	1.06	166.046421	2.5789	432.796697

where m is the number of test samples, $y(i)$ is the i -th sample, $\hat{y}(i)$ is the predicted value of $y(i)$, and $\bar{y}(i)$ is the average value of all $y(i)$ values.

For the ELM and PSR-ELM models, the input vector is $Tr_f^{new}(t)^T$ and the output vector is $Tr_f^{new}_{predict}(t+1)^T$. For SVR, the input vector is $[Tr_f(t), \dots, Tr_f(t-k)]^T$ and the output vector is $Tr_f(t+1)$.

The three algorithms were used to predict the calculation Example A.

Figure 5 presents the results of air traffic flow time-series prediction based on the ELM algorithm, and Fig. 5(a) shows the comparison between the predicted and real traffic flow. It is evident that the trend of the predicted values is similar to that of the real values, and the model accurately captured the

state of zero-flow while being insensitive to the prediction of high peaks. The blue lines shown in Fig. 5(b) are the training set, and the red line is the testing set. Two high peaks of traffic flow can be seen in the training set, which is consistent with the original time series, but the high peaks are not sufficiently accurate. Figure 5(c) presents the real traffic flow, which fluctuates significantly with time. Moreover, peaks and troughs are interlaced, and there is no obvious law that reflects the chaotic characteristics of air traffic flow.

Figure 6 presents the prediction results of air traffic flow time series based on the PSR-ELM algorithm. By comparison with Fig. 5, it can be found that the two prediction models correctly predicted the zero-flow period and the flow change trend, but the MSE value of the PSR-ELM algorithm prediction result was much smaller than that of the ELM model

(a) Comparison of traffic flow prediction results of test set

(b) Predicted time series

(c) Real time series

Fig. 5 Prediction results (ELM)

(a) Comparison of traffic flow prediction results of test set

(b) Predicted time series

(c) Real time series

Fig. 6 Prediction results (PSR-ELM)

($0.62 < 1.2467$); therefore, the prediction accuracy of PSR-ELM was higher and better fit the actual situation.

Figure 7 exhibits the prediction results of air traffic flow time series based on the SVR algorithm. Via comparison of Fig. 7(b) and (c), it can be found that the model's peak flow prediction accuracy was poor, and the MSE value was 1.06, which was greater than that of the PSR-ELM algorithm.

A comparative analysis of calculation Example B was carried out. The calculation results were synthesized, and the error (MSE) and calculation time of the ELM, PSR-ELM, and SVR algorithms were calculated, as reported in Table 5.

(a) Comparison of traffic flow prediction results of test set

(b) Predicted time series

(c) Real time series

Fig. 7 Prediction results (SVR)

ELM is a non-iterative machine learning algorithm. Its prediction time was found to be significantly shorter than that of the SVR algorithm, but its accuracy was sometimes slightly worse than that of the SVR algorithm. Due to the timeliness of traffic flow prediction, the calculation time of 2 min is not sufficient for practical applications; therefore, for air traffic flow prediction, the ELM algorithm is superior to the SVR algorithm. The PSR-ELM is an improvement of ELM in which PSR calculations are added; therefore, while the prediction time of the algorithm was slightly increased, the improvement of accuracy was more obvious, so the performance of the PSR-ELM algorithm was found to be the best.

Because the principle of the ELM algorithm is based on randomness, the prediction results are different each time. To accurately evaluate each model, the prediction results of the ELM evolution model in 100 high-dimensional phase spaces were counted 100 times, and the prediction results of the two calculation examples were recorded, as shown in Table 5. The results reveal that the prediction time was very stable, which is consistent with the previous conclusions. The average MSE values of the 100 prediction results of the two examples were respectively 0.7167333 and 2.2992856, which still represented the highest accuracy. The MSE distribution of each result was calculated, and the results are exhibited in Fig. 8.

Figure 8(a) and (b) respectively display the probability distributions of the MSE of the 100 prediction results of Examples A and B, and the results reveal the randomness in the ELM algorithm. Figure 8(c) and (d) present the box plots of 100 prediction results in the medium- and long-term, and in the short-term, respectively. Although the ELM prediction results had a certain randomness, the prediction accuracy was concentrated in a good range.

5 Comparison of prediction results with different degrees of Chaos

The relationships between different traffic flow states and the degree of chaos of the system were discussed in Section 3. The best PSR-ELM algorithm in Section 4 was subsequently used to evaluate the prediction accuracy based on the measured data.

5.1 Selection of experimental objects and indicators

It was determined in Section 3 that air traffic flow time series have chaotic characteristics, and with the sharp decrease in the amounts of different types of air traffic flow at the same location after the COVID-19 outbreak, the degree of chaos in the time series also weakened to a certain degree. This reflects the relationship between the flow state and the degree of chaos to some extent. In this section, the time series (1008 sets of data) of the air traffic flow at a 10-min time scale of a certain

Fig. 8 Distributions of MSE

waypoint from December 14 to 20, 2019, and from March 14 to 20, 2020, were selected as the research objects:

- 504 sets of training data (3.5 days), 72 sets of testing data (0.5 days);
- 648 sets of training data (4.5 days), 72 sets of testing data (0.5 days);
- 792 sets of training data (5.5 days), 72 sets of testing data (0.5 days).

The two calculation examples presented in Section 4 reflect the prediction effect of the same air traffic flow time series. Via the comparison of the MSE values, the PSR-ELM algorithm displayed the best performance. However, when the experimental object is no longer of the same time series, the choice of the evaluation index should also be reconsidered.

Therefore, to evaluate the results of the same model for the prediction of the time series of different periods in December 2019 and March 2020, another index determination coefficient R^2 was considered, which is used in statistics to measure

the variation of the dependent variable. The proportion is used to judge the explanatory power of a statistical model.

5.2 Comparative analysis

Each group of experiments was fitted with a time-series evolution model in high-dimensional phase space, and the determination coefficient R^2 and the maximum Lyapunov exponent of the training set time series of each group of experimental objects were calculated. The results are presented in Fig. 9.

By comparing the R^2 values of the graphs in either column of Fig. 9, it is evident that the prediction results of the air traffic flow time series in March were better than those in

Fig. 9 Comparison of the fitting results in December 2019 and March 2020. The results of Example A in (a) December and (b) March; The results of Example B in (c) December and (d) March; The results of Example C in (e) December and (f) March

December. After the COVID-19 epidemic, the traffic state was sparse and the traffic flow was better-predicted. The comparison results echo that the air traffic flow time series in December was more chaotic than that in March, as determined by the maximum Lyapunov indexes. The stronger the chaos of the air traffic flow time series in different periods, the more unpredictable it is.

It should be noted that, because the results of the ELM algorithm are different each time, there are sometimes unique cases. The degree of chaos and the advantages and disadvantages of the prediction results have always been popular research topics. For the time series of measured data, the sample size is certain; to capture the evolution law of nonlinear time series is actually a problem of randomness. In particular, the occurrence of public emergencies, such as the sudden decrease in air traffic after the outbreak of COVID-19, has created an opportunity for research in this field; even on weekdays and holidays, it is difficult to collect continuous and long-term actual measured data of air traffic flow with different flow states.

6 Conclusion

The research object of this paper was on-route air traffic flow in the air traffic control sector, and, based on measured ADS-B data, air traffic flow time series were extracted. A random non-iterative learning method was used based on its chaotic characteristics; the ELM model was used to predict the evolution of time series, and the prediction results of different algorithms at low- and high-dimensional levels, as well as in different periods, were compared. Based on this investigation, different flow states, the nonlinear dynamic characteristics of traffic flow, and the relationships between traffic flow prediction results were revealed. The research results demonstrate that the ELM algorithm can predict air traffic flow at a high-dimensional angles, and that the investigated air traffic flow time series exhibited varying degrees of chaotic characteristics under different conditions. After the outbreak of COVID-19, the air traffic flow was found to become sparse, the degree of chaos was weakened, and the prediction effect was improved.

The innovations of the findings of this research can be considered from the following aspects.

- 1) To obtain qualitative information on dynamic systems, it is often necessary to know sufficient state evolution information. In this paper, PSR was used to map time series from low to high dimensions, and the PSR-ELM algorithm was used to predict the evolution law of air traffic flow time series with a better prediction effect.
- 2) In recent years, randomness-based non-iterative methods have attracted widespread attention, but there exists a lack of comparative experiments conducted with measured

data. In this paper, the error, calculation time, and regression fitting of the random non-iterative ELM, and classical SVR algorithms for measured air traffic flow time series were compared.

- 3) For the first time, changes in air traffic flow chaos characteristics based on the impact of COVID on the air traffic flow state were analyzed. A relationship between the flow state, degree of chaos, and prediction accuracy was found, which provides a reference for air traffic flow theory.

Acknowledgements This study was supported in part by the National Natural Science Foundation of China (Nos. 71801215), and in part by the Research Funds for Interdisciplinary subject, Northwestern Polytechnical University.

References

1. Van Hinsbergen CPI, Sanders FM (2007) Short term traffic prediction models [C]/14th World Congress on International Transport Systems, ITS2007, vol.7, 11
2. Dia H (2001) An object-oriented neural network approach to short-term traffic forecasting[J]. *Eur J Oper Res* 131(2):253–261
3. Suykens JAK, Vandewalle J (1999) Least squares support vector machine classifiers[J]. *Neural Process Lett* 9(3):293–300
4. Park DC, El-Sharkawi MA, Marks RJ et al (1991) Electric load forecasting using an artificial neural network[J]. *IEEE Trans Power Syst* 6(2):442–449
5. Tian Z, Ren Y, Wang G (2019) Short-term wind speed prediction based on improved PSO algorithm optimized EM-ELM[J]. *Energy Sour Part A: Recover Utiliz Environ Effects* 41(1):26–46
6. Huang GB, Zhu QY, Siew CK (2006) Extreme learning machine: theory and applications[J]. *Neurocomputing* 70(1–3):489–501
7. Lan Y, Soh YC, Huang GB (2009) Ensemble of online sequential extreme learning machine[J]. *Neurocomputing* 72(13–15):3391–3395
8. Huang GB, Chen L (2007) Convex incremental extreme learning machine[J]. *Neurocomputing* 70(16–18):3056–3062
9. Feng G, Huang GB, Lin Q et al (2009) Error minimized extreme learning machine with growth of hidden nodes and incremental learning[J]. *IEEE Trans Neural Netw* 20(8):1352–1357
10. Rong HJ, Huang GB, Sundararajan N et al (2009) Online sequential fuzzy extreme learning machine for function approximation and classification problems[J]. *IEEE Trans Syst Man Cybernet Part B (Cybernetics)* 39(4):1067–1072
11. Miche Y, Van Heeswijk M, Bas P et al (2011) TROP-ELM: a double-regularized ELM using LARS and Tikhonov regularization[J]. *Neurocomput* 74(16):2413–2421
12. Bai Z, Huang GB, Wang D, Wang H, Westover MB (2014) Sparse extreme learning machine for classification[J]. *IEEE Trans Cybernetics* 44(10):1858–1870
13. Zhang R, Lan Y, Huang GB, Xu ZB, Soh YC (2013) Dynamic extreme learning machine and its approximation capability[J]. *IEEE Trans Cybernetics* 43(6):2054–2065
14. Zhang R, Lan Y, Huang G et al (2012) Universal approximation of extreme learning machine with adaptive growth of hidden nodes[J]. *IEEE Trans Neural Networks Learn Syst* 23(2):365–371
15. Shao Z, Er MJ (2016) An online sequential learning algorithm for regularized extreme learning machine[J]. *Neurocomputing* 173: 778–788

16. Disbro JE, Frame M (2018) Traffic flow theory and chaotic behavior. *Transportation research record*. Transp Res Rec 1225(1):109–115
17. Low D, Addison P (1997) Chaos in a car-following model with a desired headway time. *Int Symposium Automotive Technol Automation Att/its Advance Enhanc Passenger Freight Int Transport Syst* 1225(1):175–182
18. Tang YS et al (2005) Chaos forecast for traffic conflict flow. *Jilin Daxue Xuebao (Gongxueban)/J Jilin Univ (Eng Technol Edition)* 35(6):646–648
19. Frank GW, Lookman T, Nerenberg MAH, Essex C, Lemieux J, Blume W (1990) Chaotic time series analyses of epileptic seizures. *Physica D* 46(3):427–438
20. ShanMei L, Xu XH, LingHang M (2012) Flight conflict forecasting based on Chaotic time series. *Trans Nanjing Univ Aeronautics Astronautics* 29(4):388–394
21. Wei C, Hu M (2014) Chaotic characteristics analysis of air traffic system. *Trans Nanjing Univ Aeronautics Astronautics* 31(6):636–642
22. Figueiredo L, Jesus I, Machado JAT et al (2001) Towards the development of intelligent transportation systems[C]//ITSC 2001. 2001 IEEE Intell Transp Syst. Proceedings (Cat. No. 01TH8585). IEEE 2001:1206–1211
23. Ban X, Guo C, Li G (2016) Application of extreme learning machine on large scale traffic congestion prediction[M]//proceedings of ELM-2015 volume 1. Springer, Cham, pp 293–305
24. Ma Z, Luo G, Huang D (2016) Short term traffic flow prediction based on on-line sequential extreme learning machine[C]//2016 Eighth International Conference on Advanced Computational Intelligence (ICACI), Chiang Mai, IEEE 2016: 143–149
25. Wang D, Xiong J, Xiao Z, et al. (2016) Short-term traffic flow prediction based on ensemble real-time sequential extreme learning machine under nonstationary condition[C]//2016 IEEE 83rd Vehicular Technology Conference (VTC spring), NanJing, IEEE 2016: 1–5
26. Zhang Q, Jian D, Xu R, et al. (2017) Integrating heterogeneous data sources for traffic flow prediction through extreme learning machine[C]//2017 IEEE International Conference on Big Data (Big Data), Boston, IEEE 2017:4189–4194
27. Shang Q, Lin C, Yang Z, Bing Q, Zhou X (2016) A hybrid short-term traffic flow prediction model based on singular spectrum analysis and kernel extreme learning machine[J]. *PLoS One* 11(8): e0161259
28. Kennel MB, Brown R, Abarbanel HDI (1992) Determining embedding dimension for phase-space reconstruction using a geometrical construction[J]. *Phys Rev A* 45(6):3403–3411
29. Holzfuss J, Mayer-Kress G (1986) An approach to error-estimation in the application of dimension algorithms[M]//dimensions and entropies in chaotic systems. Springer, Berlin, pp 114–122
30. Fraser AM, Swinney HL (1986) Independent coordinates for strange attractors from mutual information[J]. *Phys Rev A* 33(2): 1134–1140
31. Rosenstein MT, Collins JJ, De Luca CJ (1994) Reconstruction expansion as a geometry-based framework for choosing proper delay times[J]. *Physica-Section D* 73(1):82–98
32. Grassberger P, Procaccia I (1983) Characterization of strange attractors[J]. *Phys Rev Lett* 50(5):346–349
33. Cao L (1997) Practical method for determining the minimum embedding dimension of a scalar time series[J]. *Physica D: Nonlinear Phenomena* 110(1–2):43–50
34. Kim HS, Eykholt R, Salas JD (1999) Nonlinear dynamics, delay times, and embedding windows[J]. *Physica D: Nonlinear Phenomena* 127(1–2):48–60
35. Ning H, Ma L (2009) The prediction model of metro vault settlement based on developed C-C method phase space reconstruction and LS-SVM. *Math Pract Theory* 44(20):130–139
36. Hong YY, Xiu HW (2002) Study on period doubling bifurcation based on systematic Lyapunov exponent analysis[J]. *Control & Decision (S1)*:814–816+819

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.