Frontiers of Information Technology & Electronic Engineering
www.jzus.zju.edu.cn; engineering.cae.cn; www.springerlink.com
ISSN 2095-9184 (print); ISSN 2095-9230 (online)
E-mail: jzus@zju.edu.cn

Spatio-temporal correlation-based incomplete time-series traffic prediction for LEO satellite networks

Liang PENG, Jie YAN, Peng WEI, Xiaoxiang WANG[†‡]

Key Laboratory of Universal Wireless Communications, Ministry of Education, Beijing University of Posts and Telecommunications, Beijing 100876, China

[†]E-mail: cpwang@bupt.edu.cn

Received Dec. 26, 2023; Revision accepted Apr. 19, 2024; Crosschecked Mar. 20, 2025

Abstract: Accurate short-term traffic prediction is essential for improving the efficiency of data transmission in low Earth orbit (LEO) satellite networks. However, traffic values may be missing due to collector failures, transmission errors, and memory failures in complex space environments. Incomplete traffic time series prevent the efficient utilization of data, which can significantly reduce the traffic prediction accuracy. To overcome this problem, we propose a novel spatio-temporal correlation-based incomplete time-series traffic prediction (ITP-ST) model, which consists of two phases: reconstituting incomplete time series by missing data imputation and making traffic prediction based on the reconstructed time series. In the first phase, we propose a novel missing data imputation model based on the improved denoising autoencoder (IDAE-MDI). Specifically, we combine DAE with the Gramian angular summation field (GASF) to establish the temporal correlation between different time intervals and extract the structural patterns from the time series. Taking advantage of the unique spatio-temporal correlation of the LEO satellite network traffic, we focus on improving the missing data initialization method for DAE. In the second phase, we propose a traffic prediction model based on a multi-channel attention convolutional neural network (TP-CACNN) by combining the spatio-temporally correlated traffic of the LEO satellite network. Finally, to achieve the ideal structure of these models, we use the multi-verse optimizer (MVO) algorithm to select the optimal combination of model parameters. Experiments show that the ITP-ST model outperforms the baseline models in terms of traffic prediction accuracy at different data missing rates, which demonstrates the effectiveness of our proposed model.

Key words: Incomplete time series; Denoising autoencoder (DAE); Spatio-temporal correlation; Traffic prediction; LEO satellite networks

https://doi.org/10.1631/FITEE.2300873 **CLC number:** TN014

1 Introduction

Due to the high mobility of the low Earth orbit (LEO) satellites, the coverage area of each satellite varies rapidly. Factors such as population size and economic conditions in different coverage areas lead to significant differences in service traffic accessing to the satellites, and dramatic changes in the traffic load can easily lead to network congestion (Jiang et al., 2023). Therefore, it becomes necessary to introduce traffic prediction to sense future traffic changes. However, due to collector failures, transmission errors, and memory failures in the harsh space environment, traffic values may usually be lost at unexpected moments. Such incomplete traffic time series undoubtedly create difficulties in data analysis, which poses a significant challenge for traffic prediction in the LEO satellite networks (Liu W et al., 2021).

In recent decades, efforts have been devoted to solving the traffic prediction problem with incomplete time series. There are two main approaches to address the problem: the two-step approach and

[‡] Corresponding author
ORCID: Liang PENG, https://orcid.org/0009-0006-6312-4627; Xiaoxiang WANG, https://orcid.org/0000-0002-2924-2295

the synchronous approach (Wang A et al., 2023). In the former, the missing values in the incomplete time series are first filled in using a missing data imputation method. Then, predictions are made based on the reconstructed complete time series. In the latter, missing data imputation and traffic prediction are synchronized.

Missing data imputation methods are the focus of research in the two-step approach, which generally include mathematical statistics and learning-based methods (Baggag et al., 2021). Commonly used statistical methods include mean interpolation (Mukhopadhyay and Mukherjee, 2020), spline interpolation (You et al., 2020), and the nearest neighbor interpolation (Li et al., 2020). Although these algorithms are simple and easy to implement, these methods are based on simplified smoothness and linearity assumptions, which cannot fully capture the nonlinear and uncertain variations in satellite network traffic and are adapted only to some simple linear time series.

In recent years, some learning-based methods have also been widely used for missing data imputation (Miao et al., 2023), such as the k-nearest neighbor (KNN), expectation–maximization (EM), generative adversarial net (GAN), and denoising autoencoder (DAE). For example, Marchang and Tripathi (2021) proposed a spatio-temporal KNN scheme for missing data imputation. Yoon et al. (2018) proposed a novel GAN framework for missing data imputation. Tao HM et al. (2020) proposed a DAE-based model for missing data imputation and validated the model on a variety of datasets. DAE can recover input data containing noise through unsupervised learning (Nguyen et al., 2021). Data missing is a particular case of noisy input, making DAE an ideal model for missing data imputation (Ma Q et al., 2020). However, the ability of DAE to extract time-dependent information from time series is limited. Hence, a different initialization method for missing data can significantly affect the performance of missing data imputation.

After obtaining the reconstructed complete time series, some prediction algorithms need to be used to output the predicted values. A good traffic prediction algorithm will improve the overall performance of the incomplete time-series traffic prediction. Similar to the previously mentioned missing data imputation methods, traffic prediction algorithms are also divided into two main categories: the statistical method and the deep learning method (Huang et al., 2022). Statistical methods (Zhao et al., 2020) include linear regression (LR), support vector machine (SVM), and autoregressive integrated moving average (ARIMA) models. Luo et al. (2019) proposed a novel GAN framework to estimate missing data in incomplete time series and then adopted SVMs to make predictions based on the reconstructed time series. Statistical-based prediction methods require the time series to be smooth. Since the satellite traffic data are extremely dynamic and highly nonlinear, these prediction methods cannot effectively analyze the future trend of satellite network traffic.

Deep learning-based prediction algorithms extract relationships directly from the data by transforming the raw data into more abstract high-level representations by using multi-layer nonlinear modules (Lu et al., 2021). After ample transformations, complex and highly flexible function approximations can be realized and this helps in achieving more accurate prediction results. Liu ZL and Li (2018) proposed a generalized regression neural network (GRNN)-based traffic prediction model for satellite networks. Ma XL et al. (2015) introduced long short-term memory (LSTM) into traffic prediction. Some studies have also considered spatially correlated time series. For example, Ma XL et al. (2017) proposed a convolutional neural network (CNN)-based traffic prediction model that abstracts the spatio-temporal features of traffic data into $(M\times N)$-dimensional matrices to extract temporal and spatial features. Ke et al. (2020) converted traffic speed and volume data into spatio-temporal multi-channel matrices before training a two-stream multi-channel CNN model to predict lane-level traffic speeds. They have shown that the prediction algorithms considering correlated traffic can obtain higher prediction accuracy than those that do not. However, they did not consider how different spatial time series may affect the prediction results.

Currently, most synchronization methods are based on improved LSTMs and gated recurrent units (GRUs). These methods fill in the missing data by extracting the contextual relationships of the time series while making predictions. Che et al. (2018) proposed GRU-D, assuming that the missing data can be considered as a combination of the last corresponding observation and the global mean. Shen

et al. (2018) proposed a missing data imputation network with residual short paths called RIMP-LSTM, which models a longer temporal dependency by the residual structure to produce better imputation results. However, the imputation error of the missing values would be carried over to the next cycle, which leads to error accumulation.

To address the shortcomings of DAE in missing data imputation, it has been investigated how DAE can be improved to enhance the performance of missing data imputation. Ye et al. (2023) transformed the time series into the Gramian angular summation field (GASF) images, converting the time series into an image that describes the temporal correlation between each time point. By feature extraction of the GASF images, potential temporal relationships in the original time series can be mined. Deng et al. (2023) proposed an enhanced CNN model to improve CNN's temporal relation extraction capability by combining CNN with GASF. In view of this approach, GASF was combined with DAE to solve the problem of DAE's poor ability to extract the temporal relationship of the time series.

DAE often adopts the initialization method of filling in zeros for missing data. However, if the initial values are far away from the actual ones, it can seriously damage the inherent temporal characteristics of the time series. Pan et al. (2023) proposed an adaptive-learned median-filled deep autoencoder (AM-DAE), which uses the median of the input data as the initial value for the missing values. However, this scheme does not apply to satellite network traffic, where the median value of the input data may still deviate significantly from the actual value due to the high fluctuation of satellite traffic.

Unlike ad-hoc networks and wireless sensor networks, satellite networks are highly regular and periodic. Liu ZL and Li (2018) pointed out that the correlation between the real-time traffic of different satellites at the same time is low due to the significant differences in their geographic areas. However, if a certain time lag is added, the traffic from different satellites will show a large spatio-temporal correlation. Therefore, to reduce the damage to the hidden temporal correlation of the time series, we use these correlated time series to improve the missing value initialization method of DAE. This spatio-temporally correlated traffic is also used as input to the traffic prediction model for improving traffic prediction accuracy.

Model parameters play a crucial role in model performance, and these parameters are usually set manually based on the experience before model learning. This practice can diminish the model performance. Multi-verse optimizer (MVO) is a new meta-heuristic optimizer for global optimization based on the multi-verse principle (Appala Naidu et al., 2019). Due to its advantages such as having only a few control parameters and its robustness (Kumar et al., 2018), it is widely used in optimization problems in the fields of engineering and finance.

The main contributions of our work are four-fold:

1. We propose a novel spatio-temporal correlation-based incomplete time-series traffic prediction (ITP-ST) model by using the unique motion regularity of the LEO satellite network. It consists of a missing data imputation model based on an improved DAE (IDAE-MDI) and a traffic prediction model based on a multi-channel attention CNN (TP-CACNN) .

2. In the IDAE-MDI model, we combine GASF and DAE to explicitly represent the potential relationship between each timestamp by projecting a 1D time series into a 2D image representation to improve the DAE's temporal correlation extraction capability. We improve the missing value initialization scheme for DAE by exploiting the unique spatio-temporal correlation of the LEO satellite networks.

3. In the TP-CACNN model, we propose a traffic prediction model based on the channel attention mechanism CNN, which uses spatio-temporally correlated traffic to assist in traffic prediction and considers the degree of influence of different spatio-temporally correlated traffic on the forecast.

4. We use the MVO algorithm to select the optimal combination of model parameters for the IDAE-MDI and TP-CACNN models.

2 Data preparation and conversion

In this section, we introduce the dataset and then describe converting a 1D time series into a 2D image using GASF.

2.1 Data preparation

Although we do not have access to the actual traffic data for the LEO satellite networks, it can

be assumed that the potential traffic demand for satellite networks is proportional to the ground traffic, taking into account the bypass effect of satellite networks on the ground network (Tasdemir et al., 2013). At the same time, the fundamental characteristics of ground traffic (self-similarity, long correlation, and burstiness) are not changed after traffic passes through the terrestrial gateway into the satellite network (Na et al., 2015). Therefore, the present paper will translate the actual global traffic data from the 2012 Internet Census Project into satellite traffic data, which collects traffic information for almost the entire global set of IPv4 addresses for nearly six months.

We construct an Iridium-like constellation consisting of 6 orbits with 11 satellites in each orbit, with each satellite having an altitude of 550 km. Each terrestrial satellite user is served by the satellite closest to it. To simplify the satellite network scenario, we uniformly divide the Earth evenly into 6×11 geographical zones and place a ground station (GS) in the center of each zone, assuming that the traffic demands of all satellite users in each zone can all be aggregated to the GS. At any given time, each GS is served by the satellite closest to it, as shown in Fig. 1a. Fig. 1b shows the traffic load from August 8, 2012, to August 15, 2012, in the zone of 30°N–60°N and 90°E–120°E, which exhibits a clear characterization of the daily cycle. To calculate the traffic load on the satellite, the Iridium-like constellation and the above-mentioned traffic model are simulated using the network simulator OPNET 14.5 software. The traffic in the satellite is aggregated every 5 min, and the traffic time series for one of the satellites is shown in Fig. 1c, which exhibits a periodicity of approximately 24 h.

2.2 Time-series imaging

Traffic time series is represented as a GASF feature image, which is a polar coordinate coding algorithm that converts a 1D time series into a 2D feature image, where each pixel shows the temporal correlation between each data point. Given a time series of N real-valued observations $\boldsymbol{x} = [x_1, x_2, \ldots, x_N]$, the first step is to normalize the time series to the interval $[-1, 1]$ by

$$\hat{x}_i = \frac{(x_i - x_{\max}) + (x_i - x_{\min})}{x_{\max} - x_{\min}}, \tag{1}$$

where $x_{\max}$ and $x_{\min}$ are the maximum and minimum in the time series, respectively. The second step is to convert the normalized time series into a polar coordinate representation as

$$\begin{cases} \varphi_i = \arccos(\hat{x}_i), \\ r_i = \frac{i}{c}, \end{cases} \tag{2}$$

where i is the timestamp and c is a constant parameter regulating the span of the polar coordinate system. After converting to a polar coordinate system, temporal correlations over different time intervals can be identified using the trigonometric sums between each point. The GASF matrix is defined as follows (Tang et al., 2023):

$$G_{i,j} = \cos(\varphi_i + \varphi_j), \tag{3}$$

where i and j are the indices of the element $G_{i,j}$ in the GASF matrix $\boldsymbol{G}$. According to the trigonometric transformation, Eq. (3) is equivalent to

$$\boldsymbol{G} = \boldsymbol{x}'\hat{\boldsymbol{x}} - \sqrt{\boldsymbol{i} - \hat{\boldsymbol{x}}^2}' \sqrt{\boldsymbol{i} - \hat{\boldsymbol{x}}^2}, \tag{4}$$

where $\boldsymbol{i}$ is the unit row vector.

From the definition of GASF, it is a generalized visualization of the temporal correlation between all data points in the time series. On one hand, the timestamp of a data point increases as it traverses from the upper left corner to the lower right corner of the GASF matrix. On the other hand, the element $G_{i,j}$ represents the temporal correlation between x_i and x_j in the time series, and $|i - j|$ denotes the time interval between them. Moreover, the elements on the main diagonal of the GASF are denoted as $G_{i,i} = \cos(2\varphi_{i,i})$, which means that the raw data information is completely stored in the diagonal of the GASF. With the encoding matrix, the time series can be approximately reconstructed as a GASF image, and its deep hidden features can be learned in the deep neural network. Fig. 2 shows the process of converting a 1D traffic time series into a GASF image; Fig. 2a is the transformation to the polar coordinate representation, Fig. 2b is the GASF image, and Fig. 2c is the 1D time series.

3 Proposed IDAE-MDI model for missing data imputation

In this section, we introduce the basic DAE and how the missing data initialization method of the

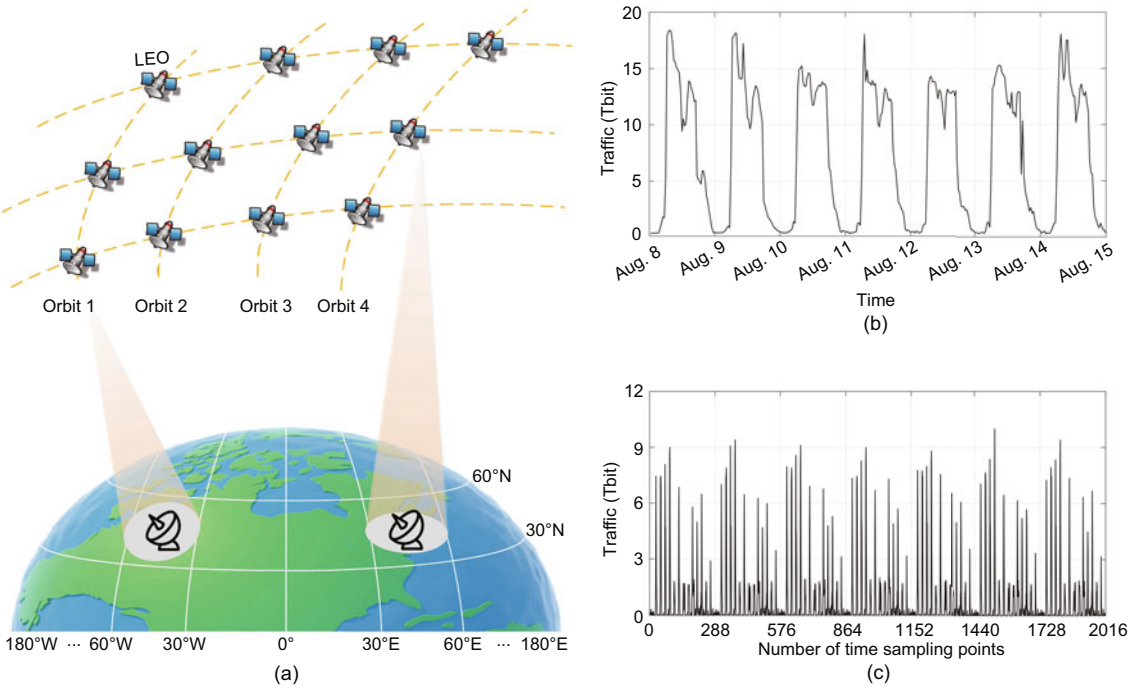

Fig. 1 LEO satellite network traffic prediction scenarios (a), traffic load in the zone of 30°N − 60°N and 90°E − 120°E (b), and traffic time series in a satellite with a total of 2016 time sampling points (c)

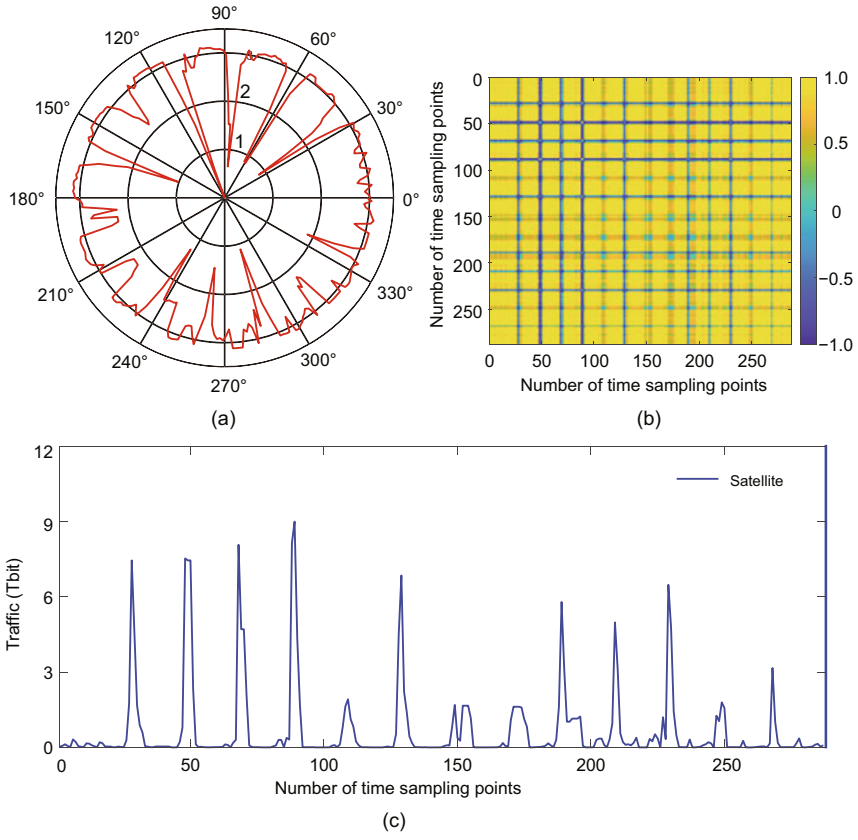

Fig. 2 Conversion of 1D time series into a 2D GASF image: (a) polar coordinate representation; (b) GASF image; (c) 1D time series

DAE is improved by exploiting the spatio-temporally correlated traffic of the LEO satellite network. We finally explain the training process of the IDAE-MDI model in detail.

3.1 DAE

DAE is an unsupervised neural network that receives corrupted data as input and outputs complete data, which is widely used for image denoising and restoration (Lee et al., 2022). DAE consists of an encoder and a decoder (Su et al., 2021), which are responsible for mapping the input data into a low-dimensional hidden representation and reconstructing the hidden representation of the original input data, respectively. The learning process of DAE is depicted in Fig. 3. We use the character $\boldsymbol{x}_{\text{actual}}$ to represent the original input data, which refers to the complete traffic time series. DAE will randomly destroy some of the data in the original input data during training to learn how to remove the noise (Wang HY et al., 2022). We use the symbol "◯" to show the location of the data in the input data that will be destroyed. In this paper, the corrupted data can be viewed as missing data. A marker vector $\boldsymbol{m}$ is introduced to indicate whether the value in the original time series $\boldsymbol{x}_{\text{actual}}$ is destroyed or not, and it is defined as

$$m_j = \begin{cases} 1, & \text{if } x_j \text{ is missing,} \\ 0, & \text{otherwise.} \end{cases} \tag{5}$$

For the missing data, some initialization methods are used to fill in the missing values, and the initialized time series $\bar{\boldsymbol{x}}_{\text{init}}$ can be represented as

$$\bar{\boldsymbol{x}}_{\text{init}} = (\boldsymbol{1} - \boldsymbol{m}) \odot \boldsymbol{x}_{\text{actual}} + \boldsymbol{m} \odot \text{init}(\boldsymbol{x}_{\text{actual}}), \tag{6}$$

where init$(\cdot)$ indicates the initialization method for missing data, which is usually initialized to 0, Gaussian, or Median, denoted by init$(\cdot) = 0$, init$(\cdot) =$ Gaussian, and init$(\cdot) =$ Median, respectively. $\odot$ denotes the element-wise multiplication.

Given an initialized time series $\bar{\boldsymbol{x}}_{\text{init}}$, the encoder maps it into a hidden representation $\boldsymbol{h}$, and then the decoder performs data reconstruction to obtain the output representation $\tilde{\boldsymbol{x}}_{\text{out}}$. The encoding and decoding functions can be defined as

$$\begin{cases} \boldsymbol{h} = \Psi\left(\bar{\boldsymbol{x}}_{\text{init}}; \theta\right), \\ \tilde{\boldsymbol{x}}_{\text{out}} = \Phi\left(\boldsymbol{h}; \gamma\right), \end{cases} \tag{7}$$

where encoder $\Psi(\cdot)$ and decoder $\Phi(\cdot)$ can be implemented by a fully connected (FC) neural network, a CNN, or a recurrent neural network. θ and γ are the parameters of the encoder and decoder, respectively.

The learning objective of DAE is to make the output $\tilde{\boldsymbol{x}}_{\text{out}}$ as close as possible to the original time series $\boldsymbol{x}_{\text{actual}}$ (Tao XL et al., 2023). Hence, the mean square error can be used as the loss function to learn the parameters, expressed as

$$\mathcal{L}(\theta, \gamma) = \frac{1}{N} \sum\nolimits_{n=1}^{N} (x_n - \tilde{x}_n)^2, \tag{8}$$

where N is the number of samples, and x_n and $\tilde{x}_n$ are the n^{th} elements of $\boldsymbol{x}_{\text{actual}}$ and $\tilde{\boldsymbol{x}}_{\text{out}}$ respectively.

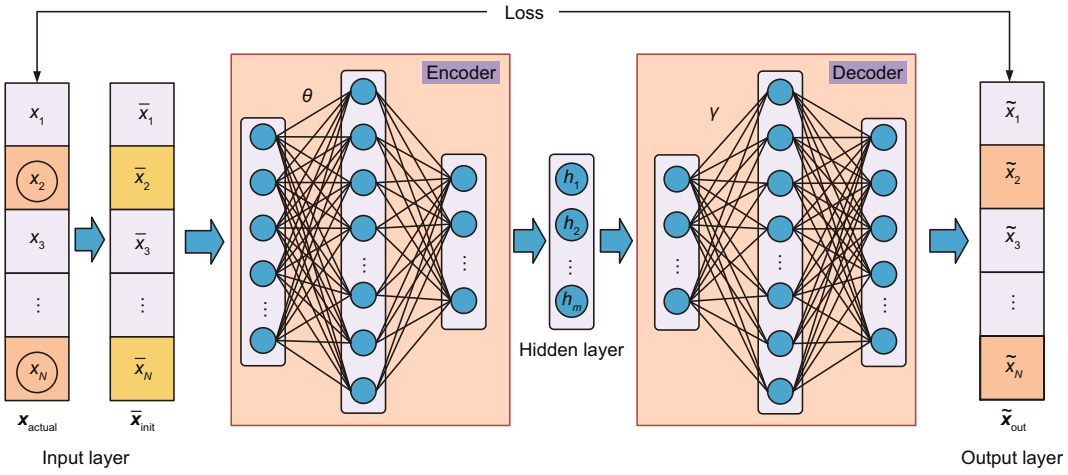

Fig. 3 Network structure of the basic DAE

3.2 Improvements of DAE in missing data initialization methods

Different initialization methods that are used for missing data in the time series can significantly affect the effectiveness of data reconstruction. The time series consisting of satellite network traffic is autocorrelated (Na et al., 2015). When we use the $\text{init}(\cdot)=0$ method, if the missing value is far from 0, this will severely destroy the potential temporal relationship of the time series, and thus the data reconstruction accuracy is reduced. Due to the highly volatile nature of the satellite network traffic, the $\text{init}(\cdot)=\text{Median}$ method will likewise lead to a large difference between the initial value and the missing value. In other words, if the initial value is closer to the missing value, it will help DAE extract more accurate potential temporal relationships, which will significantly improve the data reconstruction accuracy. Considering that the satellite network is a special network with regularity, we will improve the initialization method $\text{init}(\cdot)$ using spatially correlated traffic.

Due to the significant differences in the geographical areas covered by different satellites at the same time, the correlation of traffic time series between satellites is low. However, due to the special motion regularity of the satellites, different satellites will have ground orbits similar to each other, and traffic time series from different satellites will show greater correlation if a certain time delay is added (Liu ZL and Li, 2018). For example, we illustrate this particular traffic correlation between satellites using two neighboring satellites in the same orbit as shown in Figs. 4a and 4c, which show a 24-h traffic time series for these two satellites. We define the traffic time series of satellite i as $\boldsymbol{x}^i=\left[x_1^i,x_2^i,\ldots,x_N^i\right]$. Then, the Pearson correlation coefficient of the time series $\boldsymbol{x}^i$ lagging $\boldsymbol{x}^j$ with time lag l is calculated by

$$\rho_{i,j}(l)=\frac{\sum_{m=1}^{N}\left(x_m^i-\overline{x}^i\right)\left(x_{m-l}^j-\overline{x}^j\right)}{\sqrt{\sum_{m=1}^{N}\left(x_m^i-\overline{x}^i\right)^2}\sqrt{\sum_{m=1}^{N}\left(x_{m-l}^j-\overline{x}^j\right)^2}},\tag{9}$$

where $\rho_{i,j}(l)$ denotes the degree of correlation between the current traffic of satellite i and the historical traffic of satellite j. There is a significant correlation between the two traffic time series when $\rho_{i,j}(l)$ is close to $+1$ or -1 (Coscia, 2021). Fig. 4b demonstrates the Pearson correlation coefficients of these two time series at different time lags. We can see that the correlation coefficients of these two time series are significantly different with different time lags, and the correlation is the largest when satellite 1 lags satellite 2 by two sampling points.

Based on the correlation coefficients, a weight is assigned to each spatio-temporally correlated time series, and the missing values initialized using the K most correlated time series are as follows:

$$\text{init}(x_t^i)=\frac{\sum_{j=1}^{K}\Gamma_t x_t^j\rho_{i,j}^{-\alpha}(l)}{\sum_{j=1}^{K}\rho_{i,j}^{-\alpha}(l)},\tag{10}$$

where x_t^i denotes the traffic value of the i^{th} satellite at time t, α is a positive parameter that controls the decay rate of a satellite's weight by $\rho_{i,j}^{-\alpha}(l)$, and Γ_t is set to 0 when x_t^j is missing, and $\Gamma_t=1$ otherwise.

3.3 MVO algorithm for model parameter optimization

The parameters such as the neural network architecture, number of training iterations, and learning rate have a significant effect on the model performance. In this paper, the MVO algorithm is used to find the optimal parameter combination to maximize the model performance. The MVO algorithm is a mathematical model intended for the optimization process by using the principle of exchange of objects between wormholes in the universe (Lv et al., 2023). The MVO algorithm defines a set of universes as follows:

$$\boldsymbol{U}=\begin{bmatrix}u_1^1 & u_1^2 & \cdots & u_1^{N_{\text{o}}}\\ u_2^1 & u_2^2 & \cdots & u_2^{N_{\text{o}}}\\ \vdots & \vdots & & \vdots\\ u_{N_{\text{u}}}^1 & u_{N_{\text{u}}}^2 & \cdots & u_{N_{\text{u}}}^{N_{\text{o}}}\end{bmatrix},\tag{11}$$

where N_{u} is the number of universes, each universe represents a candidate solution, and N_{o} is the number of objects in the universe (i.e., the dimensionality of the solution).

After normalizing and ranking the expansion rates (fitness values) of the universe, the white hole is selected according to the roulette mechanism, and the black hole exchanges dimension information with it.

$$u_i^j=\begin{cases}u_k^j, & r_1<\text{NI}(\boldsymbol{U}_i),\\ u_i^j, & r_1\geq\text{NI}(\boldsymbol{U}_i),\end{cases}\tag{12}$$

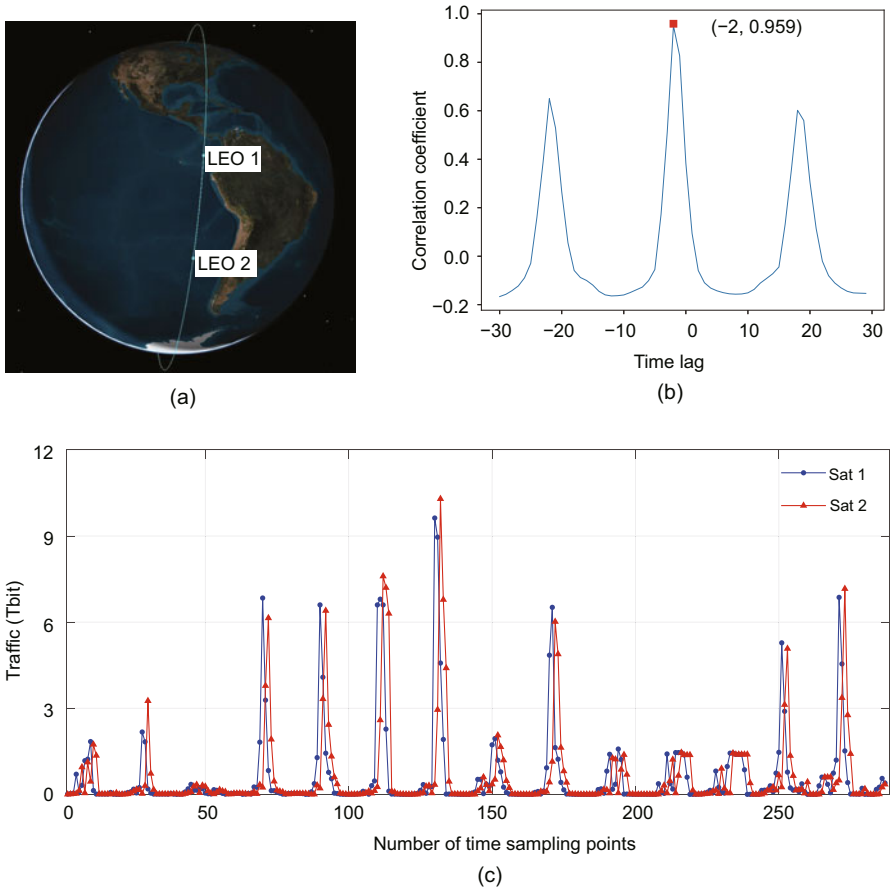

Fig. 4 Two neighboring satellites in the same orbit (a), the Pearson correlation coefficient of those two traffic time series with different time lags (b), and the 24-h traffic time series of these two satellites (c)

where NI $(\boldsymbol{U}_i)$ denotes the normalized expansion rate of universe i, r_1 is a random number in the range of $[0,1]$, and u_k^j denotes the j^{th} object of the k^{th} universe selected.

The wormholes are usually forwarded randomly to the universe, objects in the universe travel around the optimal universe through wormholes, and the iterative process is as follows:

$$u_i^j = \begin{cases} u_0^j + T\cdot\left(\left(\text{ub}^j - \text{lb}^j\right) r_2 + \text{lb}^j\right), r_3 < 0.5, r_4 < W, \\ u_0^j - T\cdot\left(\left(\text{ub}^j - \text{lb}^j\right) r_2 + \text{lb}^j\right), r_3 \geq 0.5, r_4 < W, \\ u_i^j, \quad r_4 \geq W, \end{cases} \tag{13}$$

where u_0^j is the j^{th} variable of the best universe obtained so far, ub^j and lb^j represent the upper bound and the lower bound of the j^{th} variable, respectively, and r_2, r_3, and r_4 are the random numbers in the range of $[0,1]$. W and T are the wormhole existence probability and traveling distance rate, respectively, and are expressed as follows:

$$W = W_{\min} + l_{\text{c}}\left(\frac{W_{\max} - W_{\min}}{L}\right), \tag{14}$$

$$T = 1 - \frac{l_{\text{c}}^{1/p}}{L^{1/p}}, \tag{15}$$

where $W_{\min}$ and $W_{\max}$ denote the minimum and maximum of the wormhole existence probability W, respectively, l_{c} is the index of the current iteration, L denotes the maximum number of repetitions, and p describes the local search accuracy over the iterations.

3.4 Training process of the IDAE-MDI model

The overall framework of the IDAE-MDI model is shown in Fig. 5, and the detailed training process is shown in the following steps:

Step 1: This step is the data preparation phase, which includes the preparation of the training dataset and the finding of the K most correlated

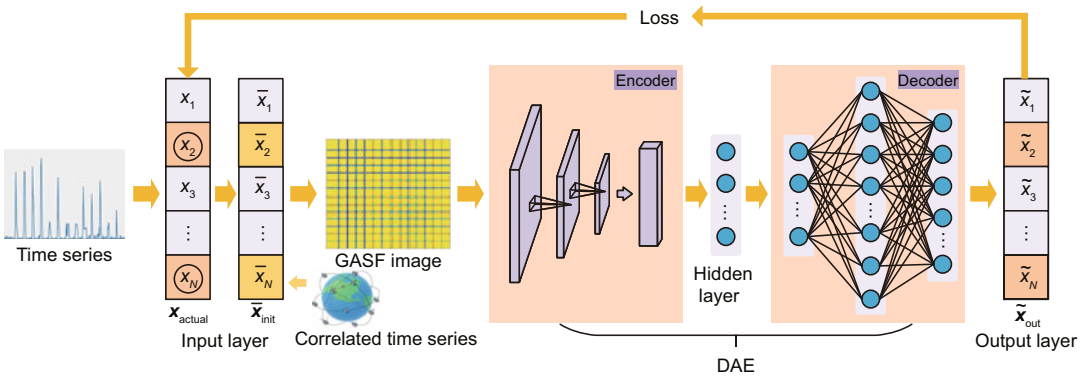

Fig. 5 Framework of the IDAE-MDI model

spatio-temporal traffic time series for each satellite using Eq. (9).

Step 2: This step initializes the relevant parameters of the MVO algorithm, and the population defines the IDAE-MDI model parameters, including network structure, learning factor, the maximum number of training iterations, and batch size.

Step 3: This step formally enters the training session. Randomly destroy part of the data in the traffic time series $\boldsymbol{x}_{\text{actual}}$ and initialize these missing values using Eq. (10), and finally obtain $\bar{\boldsymbol{x}}_{\text{init}}$ according to Eq. (6). To improve the temporal extraction capability of the DAE, the time series $\bar{\boldsymbol{x}}_{\text{init}}$ is then converted to a 2D image $\boldsymbol{G}$, i.e., GASF representation.

Step 4: In this step, the GASF matrix $\boldsymbol{G}$ is used as the input to the DAE, and the hidden representation $\boldsymbol{h}$ is mapped by the encoder. Meanwhile, we construct the encoder with a CNN to obtain a better performance in image recognition. Then the decoder outputs the reconstructed time series $\tilde{\boldsymbol{x}}_{\text{out}}$. The process is as in Eq. (7).

Step 5: This step compares the input time series $\boldsymbol{x}_{\text{actual}}$ with the output time series $\tilde{\boldsymbol{x}}_{\text{out}}$ to obtain the error function $\mathcal{L}$ in Eq. (8), and optimizes the DAE network weight parameters θ and γ using the gradient descent method.

Step 6: This step determines whether the maximum number of iterations has been reached, and if it is satisfied, then jumps to step 7, and returns to step 3 otherwise.

Step 7: This step enters the next iteration of MVO, updating the population using Eq. (13), and then jumps to step 3.

Step 8: This step determines whether the maximum number of iterations of the MVO algorithm has been reached, and if it is satisfied, ends the training, and returns to step 7 otherwise.

Step 9: In this step, the trained IDAE-MDI model is used for missing data imputation for incomplete time series.

4 Proposed TP-CACNN model for traffic prediction

After imputing the missing data through the IDAE-MDI model, we predict the satellite network's traffic values for the next moment based on these reconstructed time series. In this paper, we will use the spatio-temporally correlated traffic mentioned above to improve the traffic prediction accuracy. CNN has excellent multi-channel feature extraction capability (Zhou et al., 2021), but cannot distinguish the importance of each channel. The K spatio-temporally correlated time series have different correlations with the target time series, which will have different degrees of influence on traffic prediction. In recent years, attention-based models have been widely developed and shown their effectiveness in many fields. Zhang CH et al. (2019) and Huang et al. (2022) used an attention mechanism to extract the spatial relationship of traffic at different nodes, which significantly improves the accuracy of traffic prediction compared to using the traffic data at only one node. Inspired by this, this paper improves the CNN by introducing a channel attention mechanism to distinguish the importance of each spatio-temporally related time series. At the same time, to improve the temporal extraction ability of CNN, 1D time series

are still first converted into 2D GASF images. The overall framework is shown in Fig. 6, and the steps are given as follows:

Step 1: This step is the data processing phase, where the target time series and the K most relevant spatio-temporal time series are transformed into 2D GASF feature images. When the length of the time series is N, we can obtain a feature map $\boldsymbol{x} \in \mathbb{R}^{K\times N\times N}$. Specifically, we can denote $\boldsymbol{x} = \left[\boldsymbol{x}^1, \boldsymbol{x}^2, \ldots, \boldsymbol{x}^K\right]$, where $\boldsymbol{x}^k \in \mathbb{R}^{N\times N}$, $k = 1, 2, \cdots, K$.

Step 2: This step gives the attention weights for each channel and generates the refined feature map with attention weights. Since the K spatio-temporally correlated time series have different correlations with the target time series, they have different degrees of influence on traffic prediction. Therefore, the TP-CACNN model first adds a channel attention mechanism to adaptively assign channel weights to distinguish the importance of each spatio-temporally correlated time series. We squeeze the global feature map $\boldsymbol{x}$ using the global average pooling and the maximum pooling to generate two channel-level statistics $\boldsymbol{z}^{\text{ave}}$ and $\boldsymbol{z}^{\max}$, respectively. Formally, the statistics $\boldsymbol{z}^{\text{ave}}$ and $\boldsymbol{z}^{\max}$ are erased by shrinking $\boldsymbol{x}$ through spatio-temporal dimensions $N \times N$, where the c^{th} elements of $\boldsymbol{z}^{\text{ave}}$ and $\boldsymbol{z}^{\max}$ are calculated by

$$\boldsymbol{z}_c^{\text{ave}} = \frac{1}{N^2}\sum_{i=1}^{N}\sum_{j=1}^{N}\boldsymbol{x}^c(i,j), \tag{16}$$

$$\boldsymbol{z}_c^{\max} = \max(\boldsymbol{x}^c(i,j)), \tag{17}$$

where $\boldsymbol{z}_c^{\text{ave}}$ and $\boldsymbol{z}_c^{\max}$ represent the information of channel c. Then $\boldsymbol{z}_c^{\text{ave}}$ and $\boldsymbol{z}_c^{\max}$ are forwarded to a shared network to produce our channel attention map. The shared network is composed of a multi-layer perceptron (MLP) with one hidden layer and has the same number of neurons in the input and output layers. After the shared network is applied to each descriptor, we merge the output feature vectors using element-wise summation. In short, the channel attention $\boldsymbol{M} \in \mathbb{R}^{K\times 1\times 1}$ is computed as

$$\boldsymbol{M} = \sigma\left(\boldsymbol{W}_1\left(\boldsymbol{W}_0\left(\boldsymbol{z}^{\text{ave}}\right)\right) \oplus \boldsymbol{W}_1\left(\boldsymbol{W}_0\left(\boldsymbol{z}^{\max}\right)\right)\right), \tag{18}$$

where σ denotes the sigmoid function, and $\boldsymbol{W}_0$ and $\boldsymbol{W}_1$ are the neural network weights. Finally, the refined feature map $\boldsymbol{x}'$ after channel attention processing is obtained as

$$\boldsymbol{x}' = \boldsymbol{x} \otimes \boldsymbol{M}, \tag{19}$$

where $\otimes$ refers to the channel-wise multiplication between the feature map $\boldsymbol{x}$ and channel attention $\boldsymbol{M}$.

Step 3: This step initializes the relevant parameters of the MVO algorithm, and the population defines the TP-CACNN model parameters, including network structure, learning factor, maximum number of training iterations, and batch size.

Step 4: In this step, the TP-CACNN model enters the training phase. The refined feature map $\boldsymbol{x}'$ is inputted to the CNN, and each channel performs a convolution operation with the convolution kernel. Then, the computed results of each channel are summed up by the channel and inputted to the FC layer to obtain the predicted value. Finally, the predicted values are compared with the true values, and the training parameters are optimized using the gradient descent method.

Step 5: This step determines whether the maximum number of iterations has been reached, and if it is satisfied, then jumps to step 6, and returns to step 4 otherwise.

Step 6: This step enters the next iteration of MVO, updating the population using Eq. (13), and then jumps to step 4.

Step 7: This step determines whether the maximum number of iterations of the MVO algorithm has

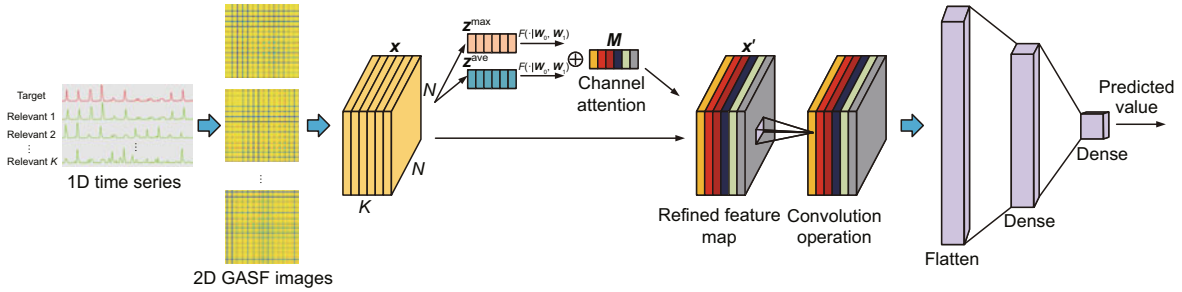

Fig. 6 Principle of the TP-CACNN model

been reached, and if it is satisfied, ends the training, and returns to step 6 otherwise.

Step 8: In this step, the trained TP-CACNN model is used for traffic prediction.

5 Experiments and results

5.1 Experimental design

We evaluate the proposed model based on the previously described dataset obtained from the 2012 Internet Census Project, where the first 80% of the data are used as the training set and the last 20% as the test set. To validate the performance of the proposed model with different data missing rates, we randomly mask some of the data in the dataset with data missing rates of 10%, 20%, 30%, 40%, and 50%. Meanwhile, to evaluate the proposed model's performance more comprehensively and avoid possible inaccuracies due to data and calculation restrictions, we conduct experiments on a total of three satellites.

The root mean square error (RMSE) is used as our evaluation metric to compare the accuracy of missing data imputation of different methods, expressed as follows:

$$\mathrm{RMSE}=\sqrt{\frac{1}{N}\sum_{n=1}^{N}(x_n-\tilde{x}_n)^2}, \tag{20}$$

where $\tilde{x}_n$ and x_n represent the imputation of missing data and real data, respectively. The smaller the RMSE value, the better the performance of the model.

5.2 Complexity analysis

In this subsection, the time complexity of the proposed ITP-ST model is analyzed, which is jointly determined by the IDAE-MDI and TP-CACNN models. The IDAE-MDI model contains mainly a CNN $\psi(\theta)$ and an FC network $\zeta(\gamma)$. Suppose that $\psi(\theta)$ contains $\mathcal{L}_\psi$ convolution layers and that $\zeta(\gamma)$ contains $\mathcal{L}_\zeta$ FC layers. The time complexity of the IDAE-MDI model is expressed as

$$O\left(\sum_{l=0}^{\mathcal{L}_\psi}N_l^2S_l^2+\sum_{l=0}^{\mathcal{L}_{\zeta-1}}\zeta_l\zeta_{l+1}\right), \tag{21}$$

where N_l denotes the input length of the l^{th} layer, S_l denotes the size of the convolution kernel of the l^{th} layer, and ζ_l denotes the number of neurons in the l^{th} layer. The TP-CACNN model is a multi-channel CNN $\varsigma(\omega)$ with an attention mechanism, and the time complexity of the TP-CACNN model can be expressed as

$$O\left(N_0^2C_l+\sum_{l=0}^{\mathcal{L}_\varsigma}N_l^2S_l^2C_l\right), \tag{22}$$

where N_0 denotes the length of the time series, C_l defines the number of input channels in the l^{th} layer, and $\mathcal{L}_\varsigma$ is the number of hidden layers.

5.3 Experimental results and analysis

1. Model parameter optimization results

Figs. 7a and 7b illustrate the search process for the optimal parameters of IDAE-MDI and TP-CACNN models, respectively. The optimal model parameters are shown in Table 1.

2. Ablation experiments on the IDAE-MDI model

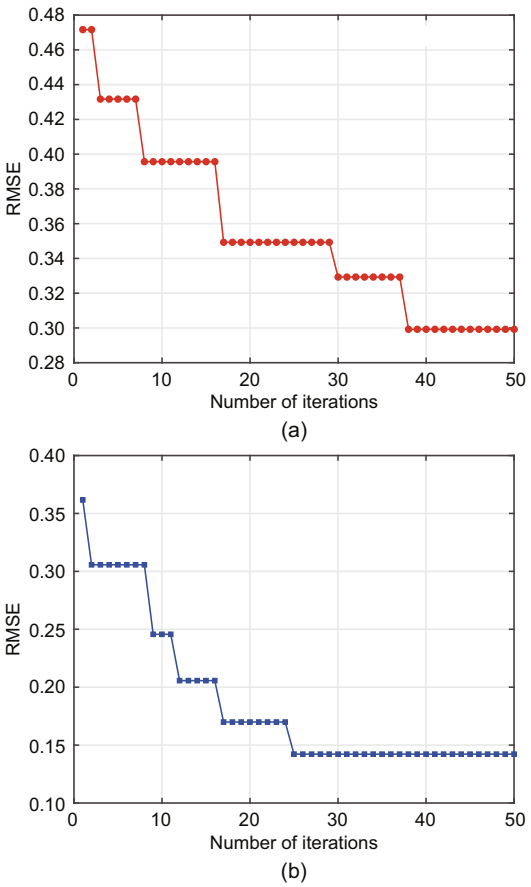

Fig. 7 Search process for the optimal parameters of the IDAE-MDI (a) and TP-CACNN (b) models

Table 1 Optimal parameters of the models

Model	Parameter	Value
IDAE-MDI	Size of the encoder	64×64
	Size of the hidden layer	28
	Size of the decoder	$28 \times 171 \times 64$
	Learning factor	3.079e-5
	Batch size	32
	Number of training iterations	116
TP-CACNN	Size of the input layer	64×64
	Size of the FC layer	$126 \times 54 \times 1$
	Learning factor	1.62e-5
	Batch size	32
	Number of training iterations	98

To verify what extent can the DAE affect the improvement of the missing data imputation, we compare the performances of the IDAE-MDI model and its variants through ablation experiments. The different variants of the IDAE-MDI model are as follows:

(1) DAE-1D-Zero: This is the base DAE, where the input is a 1D time series, and the missing data are initialized to zero.

(2) DAE-1D-Median: The input to the DAE is a 1D time series, and the missing data are initialized to the median value of the input.

(3) DAE-1D-Related: The input to the DAE is a 1D time series, and the missing data are initialized using the missing data initialization method proposed in this paper.

(4) DAE-2D-Zero: The input to the DAE is a 2D time series that has been transformed by GASF, and the missing data are initialized to zero.

(5) DAE-2D-Median: The input to the DAE is a 2D time series, and the missing data are initialized to the median of the input.

(6) IDAE-MDI: This is the model proposed in this paper, where the input is a GASF-transformed 2D time series, and the missing data are initialized using the spatio-temporally correlated time series.

To evaluate the advantages of the IDAE-MDI model in terms of computational efficiency and accuracy, we conduct experiments on these two aspects separately. Fig. 8 illustrates the comparison of the convergence speed of the IDAE-MDI model and its variants at different data missing rates, and we can see that the IDAE-MDI model has a higher computational efficiency. We visualize the training process with a missing data rate of 10%, as shown in Fig. 9, where each point indicates the RMSE value of the test set during the training process.

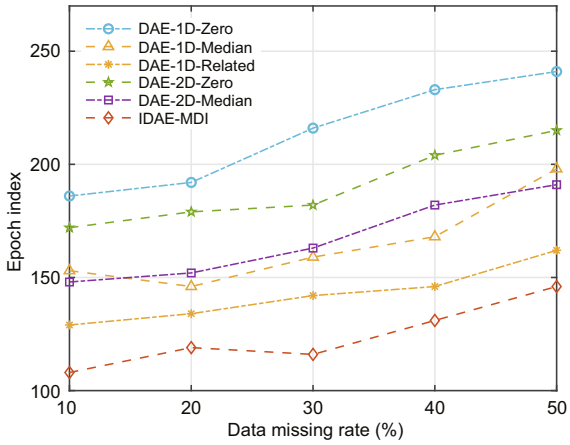

Fig. 8 Comparison of convergence speed of the IDAE-MDI model and its variants at different data missing rates

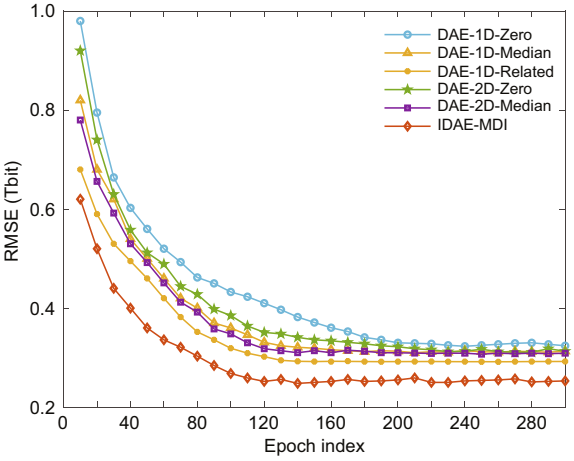

Fig. 9 RMSE comparison of the IDAE-MDI model and its variants during training at 10% data missing rate

Table 2 shows in detail the RMSE of the IDAE-MDI model and its variants for missing data imputation for a total of three satellites. We average the RMSE values of the three satellites as shown in Fig. 10. It can be seen that the IDAE-MDI model outperforms its variants in terms of accuracy of missing data imputation at different missing rates. When DAE uses the same missing data initialization method, e.g., DAE-1D-Zero and DAE-2D-Zero, transforming a 1D time series into GASF matrices has better accuracy in missing data imputation. When DAE uses the same input method, such as DAE-2D-Zero, DAE-2D-Median, and IDAE-MDI, it is better to use the initialization of the correlated traffic rather than the median and zero. Moreover, DAE-1D-Related has better missing data imputation accuracy than DAE-2D-Zero and DAE-2D-Median,

Table 2 RMSE comparison between the IDAE-MDI model with its variants and the baseline models for missing data imputation with data missing rates of 10%, 20%, 30%, 40%, and 50%

Model	RMSE (Tbit)														
	Satellite 1					Satellite 2					Satellite 3				
	10%	20%	30%	40%	50%	10%	20%	30%	40%	50%	10%	20%	30%	40%	50%
DAE-1D-Zero	0.3262	0.3529	0.4080	0.4826	0.5780	0.3400	0.3618	0.4270	0.4728	0.5903	0.3494	0.3708	0.4125	0.4710	0.5724
DAE-1D-Median	0.3380	0.3386	0.3987	0.4287	0.5360	0.3190	0.3650	0.3870	0.4480	0.5493	0.3200	0.3914	0.3903	0.4290	0.5350
DAE-1D-Related	0.3090	0.3180	0.3302	0.4060	0.4500	0.2971	0.3219	0.3280	0.3960	0.4420	0.2972	0.2978	0.3022	0.4124	0.4541
DAE-2D-Zero	0.3279	0.3440	0.3594	0.4630	0.5603	0.3183	0.3250	0.3720	0.4322	0.5411	0.3308	0.3308	0.3740	0.4225	0.5600
DAE-2D-Median	0.3200	0.3325	0.3603	0.4250	0.5020	0.3370	0.3270	0.3520	0.4080	0.4953	0.3208	0.3470	0.3630	0.4127	0.4730
Spline	0.3905	0.4787	0.5836	0.7560	0.9856	0.4203	0.4669	0.6035	0.6890	1.1090	0.4148	0.4698	0.5704	0.7314	1.1460
KNN-ST	0.3190	0.3580	0.4398	0.4860	0.6530	0.3287	0.3627	0.4260	0.5030	0.6050	0.3127	0.3763	0.4148	0.4860	0.5910
GRU-D	0.3104	0.3360	0.3840	0.4400	0.5670	0.2980	0.3263	0.3620	0.4896	0.6030	0.3190	0.3147	0.3777	0.4580	0.6269
IDAE-MDI(WM)	0.2939	0.2989	0.3270	0.3350	0.3980	0.2939	0.3087	0.3094	0.3603	0.4020	0.2568	0.3070	0.3040	0.3991	0.4617
IDAE-MDI	**0.2403**	**0.2668**	**0.2826**	**0.3199**	**0.3824**	**0.2309**	**0.2693**	**0.2932**	**0.3201**	**0.3787**	**0.2474**	**0.2516**	**0.2947**	**0.3344**	**0.3790**

The best results are in bold

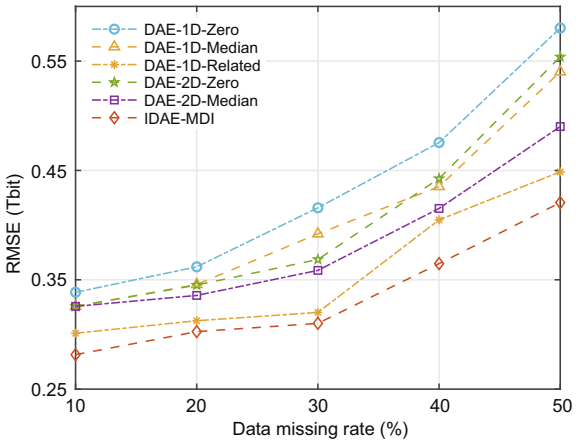

Fig. 10 RMSE comparison between the IDAE-MDI model and its variants at different data missing rates

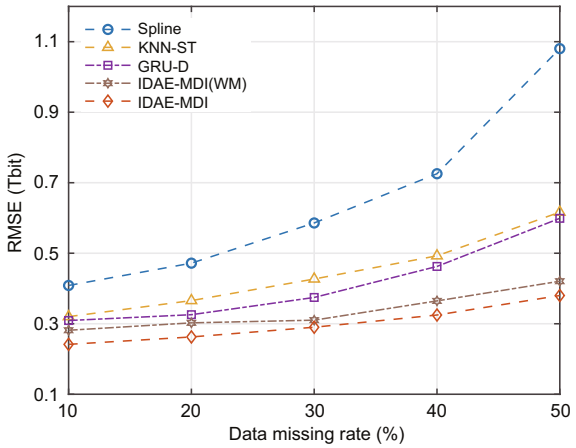

Fig. 11 RMSE comparison between the IDAE-MDI model and the baseline models at different data missing rates

indicating that the initialization method using correlated traffic has a more significant impact compared to transforming 1D time series into GASF matrices.

3. Performance comparison of the IDAE-MDI model with the baseline models for missing data imputation

To evaluate the performance of the IDAE-MDI model, we compare it with the following baseline methods commonly used for missing data imputation:

(1) Spline: The spline interpolation is used for missing data imputation.

(2) KNN-ST: The missing data are imputed using KNN.

(3) GRU-D: Learn the temporal properties of a time series to compute missing data using an improved GRU.

(4) IDAE-MDI(WM): IDAE-MDI model without parameter optimization by the MVO algorithm.

Fig. 11 shows the RMSE comparison between the IDAE-MDI model and the baseline models for missing data imputation at different data missing rates, and it can be seen that the IDAE-MDI model outperforms the baseline models in missing data imputation at different data missing rates. The RMSEs of different models increase with the increase of data missing rate, but the RMSE of the IDAE-MDI model grows the most slowly. The spline interpolation method performs well at a low data missing rate, but the accuracy decreases rapidly as the data missing rate increases. As the data missing rate increases, the GRU-D model accumulates too much error, which leads to a rapid decrease in the accuracy of missing data imputation.

4. Performance comparison of the TP-CACNN model with the baseline models for time-series traffic prediction

To validate the performance of the TP-CACNN model, we compare it with the following baseline models based on the complete time series:

(1) SVM: A traditional statistics-based machine learning algorithm.

(2) LSTM (Ma XL et al., 2015): A recurrent neural network commonly used for time-series prediction, which can extract temporal relationships in time series.

(3) CNN-MC (Ke et al., 2020): Spatio-temporally relevant traffic is used as CNN input, but the degree of influence of different spatio-temporal traffics on the traffic prediction is not considered.

(4) TP-CACNN(WM): TP-CACNN model without parameter optimization by the MVO algorithm.

Table 3 shows the RMSE of the TP-CACNN model and the baseline models for time-series traffic forecasts for a total of three LEO satellites. We can see that the traffic prediction accuracy is higher due to the consideration of spatio-temporally correlated traffic by the CNN-MC model and the TP-CACNN model. TP-CACNN has a lower RMSE than CNN-MC, which proves the effectiveness of the introduced multi-channel attention mechanism. Fig. 12 shows the comparison curves between the true and predicted values of five different models for one of the satellites.

5. Performance comparison of the ITP-ST model with the baseline models for incomplete time-series traffic prediction

To validate the overall performance of the ITP-ST model, we compare it with the following baseline models:

(1) Spline-GRU: The spline interpolation is used for missing data imputation, and GRU is used for traffic prediction.

(2) GAN-SVM (Luo et al., 2019): GAN is used to estimate missing data in incomplete time series and then SVM is used to predict traffic based on the reconstructed time series.

(3) GRU-D (Che et al., 2018): A synchronization method that implements missing data imputation and traffic prediction by means of an improved GRU to extract temporal attributes of incomplete time series.

(4) RIMP-LSTM (Shen et al., 2018): A synchronization method that fuses the LSTM structure with residual shortcut connection and takes into account the previously observed values and the overall observed values in a sample set.

(5) DRMI-GRU (Zhang JJ et al., 2019): The GRU structure is fused with residual shortcut connection, and the previously observed values and the overall observed values in a sample set are taken into account.

Table 4 shows the RMSE comparison between the ITP-ST model and the baseline models for incomplete time-series traffic prediction at different data missing rates for a total of three satellites. We average the RMSE values of the three satellites as shown in Fig. 13. We can see that the ITP-ST model outperforms the baseline models under different data missing rates. When the data missing rate is small (e.g., 10%), the difference in the performance of the different models is not significant. This is mainly because different models perform well in missing data imputation at a low data missing rate. RMSE of each model grows with the increase of data missing rate, but the RMSE of the ITP-ST model grows the most slowly. The traffic prediction accuracy of the spline-GRU model decreases rapidly with the increase of data missing rate. For the GRU-D and RIMP-LSTM

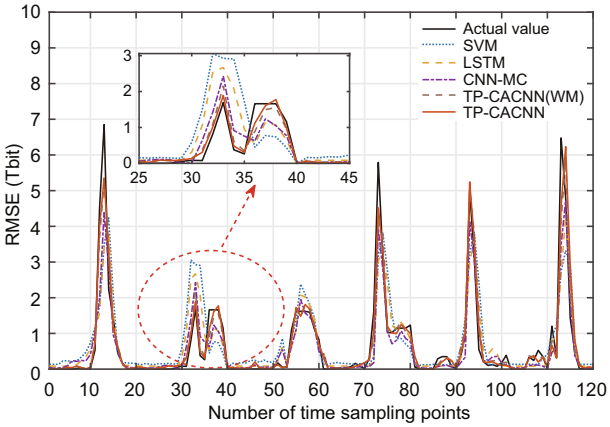

Fig. 12 Comparison curves of the true and predicted values for five different models

Table 3 RMSE for different models for each satellite

Model	RMSE (Tbit)		
	Satellite 1	Satellite 2	Satellite 3
SVM	0.2814	0.2604	0.2908
LSTM	0.2305	0.2114	0.2349
CNN-MC	0.2121	0.1935	0.2197
TP-CACNN(WM)	0.1682	0.1802	0.1814
TP-CACNN	**0.1506**	**0.1492**	**0.1449**

The best results are in bold

Table 4 RMSE comparison between the ITP-ST model and the baseline models for incomplete time-series traffic prediction with data missing rates of 10%, 20%, 30%, 40%, and 50%

Model	RMSE (Tbit)														
	Satellite 1					Satellite 2					Satellite 3				
	10%	20%	30%	40%	50%	10%	20%	30%	40%	50%	10%	20%	30%	40%	50%
Spline-GRU	0.4258	0.4425	0.4820	0.5063	0.6049	0.3914	0.4360	0.5019	0.5460	0.5820	0.4084	0.4170	0.4736	0.5841	0.6207
GAN-SVM	0.3853	0.3659	0.4426	0.4588	0.5780	0.3524	0.3761	0.4372	0.4730	0.5533	0.3679	0.4035	0.4277	0.4924	0.5263
GRU-D	0.3566	0.3750	0.4308	0.4431	0.4589	0.3328	0.3692	0.4180	0.4620	0.4819	0.3476	0.3512	0.3872	0.4226	0.4949
RIMP-LSTM	0.3106	0.3363	0.3522	0.4027	0.4339	0.3028	0.3004	0.3666	0.4055	0.4290	0.3036	0.3403	0.3569	0.3775	0.3975
DRMI-GRU	0.3025	0.3106	0.3392	0.3698	0.4059	0.2806	0.3261	0.3402	0.3709	0.4022	0.2858	0.3211	0.3410	0.3837	0.3936
ITP-ST	**0.2356**	**0.2783**	**0.2599**	**0.2907**	**0.3162**	**0.2501**	**0.2690**	**0.3080**	**0.3002**	**0.3144**	**0.2689**	**0.2604**	**0.2725**	**0.2935**	**0.3011**

The best results are in bold

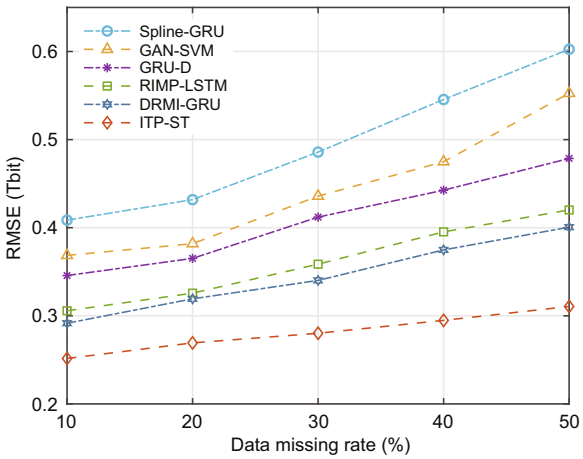

Fig. 13 RMSE comparison between the ITP-ST model and the baseline models at different data missing rates

models, it does not lead to significant performance degradation as the data missing rate increases due to the strong temporal extraction capability of GRU and LSTM.

6 Conclusions

This paper proposed a novel spatio-temporal correlation-based incomplete time-series traffic prediction model for LEO satellite networks, i.e., ITP-ST. It consisted of an IDAE-MDI model for missing data imputation and a TP-CACNN model for traffic prediction. The IDAE-MDI model was based on an improved DAE. First, we combined GASF and DAE to explicitly represent the potential relationship between each timestamp by projecting the 1D time series into a 2D image representation. Second, we took advantage of the unique spatio-temporal correlation of the LEO satellite network to improve the missing value initialization scheme of DAE. After obtaining the reconstructed time series, we performed traffic prediction using the proposed TP-CACNN, which takes into account the degree of influence of different spatio-temporally correlated traffic on traffic prediction. Finally, to achieve the ideal structure of these models, we used the MVO algorithm to select the optimal combination of model parameters. Experiments on the real traffic dataset showed that the ITP-ST model outperforms the baseline models in terms of traffic prediction accuracy at different data missing rates, which demonstrated the effectiveness of our proposed model.

Contributors

Liang PENG designed the research. Jie YAN and Peng WEI processed the data. Liang PENG drafted the paper. Jie YAN helped organize the paper. Xiaoxiang WANG revised and finalized the paper.

Conflict of interest

All the authors declare that they have no conflict of interest.

Data availability

Due to the nature of this research, the authors of this study did not agree for their data to be shared publicly, so supporting data are not available.

References

Appala Naidu T, Raj Arya S, Maurya R, 2019. Dynamic voltage restorer with quasi-Newton filter-based control algorithm and optimized values of PI regulator gains. *IEEE J Emerg Sel Top Power Electron*, 7(4):2476-2485. https://doi.org/10.1109/JESTPE.2018.2890415

Baggag A, Abbar S, Sharma A, et al., 2021. Learning spatiotemporal latent factors of traffic via regularized tensor factorization: imputing missing values and forecasting. *IEEE Trans Knowl Data Eng*, 33(6):2573-2587. https://doi.org/10.1109/TKDE.2019.2954868

Che ZP, Purushotham S, Cho K, et al., 2018. Recurrent neural networks for multivariate time series with missing

values. *Sci Rep*, 8(18):6085. https://doi.org/10.1038/s41598-018-24271-9

Coscia M, 2021. Pearson correlations on complex networks. *J Compl Netw*, 9(6):cnab036. https://doi.org/10.1093/comnet/cnab036

Deng BW, Xu TW, Yan MD, 2023. UWB NLOS identification and mitigation based on Gramian angular field and parallel deep learning model. *IEEE Sens J*, 23(22):28513-28525. https://doi.org/10.1109/JSEN.2023.3323564

Huang J, Luo K, Cao LB, et al., 2022. Learning multiaspect traffic couplings by multirelational graph attention networks for traffic prediction. *IEEE Trans Intell Transp Syst*, 23(11):20681-20695. https://doi.org/10.1109/TITS.2022.3173689

Jiang B, Yan YC, You L, et al., 2023. Robust secure transmission for satellite communications. *IEEE Trans Aerosp Electron Syst*, 59(2):1598-1612. https://doi.org/10.1109/TAES.2022.3203027

Ke RM, Li W, Cui ZY, et al., 2020. Two-stream multi-channel convolutional neural network for multi-lane traffic speed prediction considering traffic volume impact. *Transp Res Rec*, 2674(4):459-470. https://doi.org/10.1177/0361198120911052

Kumar P, Garg S, Singh A, et al., 2018. MVO-based 2-D path planning scheme for providing quality of service in UAV environment. *IEEE Int Things J*, 5(3):1698-1707. https://doi.org/10.1109/JIOT.2018.2796243

Lee WK, Seo HJ, Seo SC, et al., 2022. Efficient implementation of AES-CTR and AES-ECB on GPUs with applications for high-speed FrodoKEM and exhaustive key search. *IEEE Trans Circ Syst II Expr Briefs*, 69(6):2962-2966. https://doi.org/10.1109/TCSII.2022.3164089

Li FY, Shang CJ, Li Y, et al., 2020. Interpolation with just two nearest neighboring weighted fuzzy rules. *IEEE Trans Fuzzy Syst*, 28(9):2255-2262. https://doi.org/10.1109/TFUZZ.2019.2928496

Liu W, Ren C, Xu Y, 2021. PV generation forecasting with missing input data: a super-resolution perception approach. *IEEE Trans Sustain Energy*, 12(2):1493-1496. https://doi.org/10.1109/TSTE.2020.3029731

Liu ZL, Li X, 2018. Short-term traffic forecasting based on principal component analysis and a generalized regression neural network for satellite networks. *J China Univ Posts Telecommun*, 25(1):15-28. https://doi.org/10.19682/j.cnki.1005-8885.2018.0002

Lu BL, Liu ZH, Wei HL, et al., 2021. A deep adversarial learning prognostics model for remaining useful life prediction of rolling bearing. *IEEE Trans Artif Intell*, 2(4):329-340. https://doi.org/10.1109/TAI.2021.3097311

Luo YH, Zhang Y, Cai XR, et al., 2019. E^{2}GAN: end-to-end generative adversarial network for multivariate time series imputation. Proc 28th Int Joint Conf on Artificial Intelligence, p.3094-3100.

Lv ZL, Peng LH, Cao YJ, et al., 2023. Weak fault feature extraction method of rolling bearings based on MVO-MOMEDA under strong noise interference. *IEEE Sens J*, 23(14):15732-15740. https://doi.org/10.1109/JSEN.2023.3277516

Ma Q, Lee WC, Fu TY, et al., 2020. MIDIA: exploring denoising autoencoders for missing data imputation. *Data Min Knowl Disc*, 34:1859-1897. https://doi.org/10.1007/s10618-020-00706-8

Ma XL, Tao ZM, Wang YH, et al., 2015. Long short-term memory neural network for traffic speed prediction using remote microwave sensor data. *Transp Res Part C Emerg Technol*, 54:187-197. https://doi.org/10.1016/j.trc.2015.03.014

Ma XL, Dai Z, He ZB, et al., 2017. Learning traffic as images: a deep convolutional neural network for large-scale transportation network speed prediction. *Sensors*, 17(4):818. https://doi.org/10.3390/s17040818

Marchang N, Tripathi R, 2021. KNN-ST: exploiting spatio-temporal correlation for missing data inference in environmental crowd sensing. *IEEE Sens J*, 21(3):3429-3436. https://doi.org/10.1109/JSEN.2020.3024976

Miao XY, Wu YY, Chen L, et al., 2023. An experimental survey of missing data imputation algorithms. *IEEE Trans Knowl Data Eng*, 35(7):6630-6650. https://doi.org/10.1109/TKDE.2022.3186498

Mukhopadhyay S, Mukherjee A, 2020. ImdLMS: an imputation based LMS algorithm for linear system identification with missing input data. *IEEE Trans Signal Process*, 68:2370-2385. https://doi.org/10.1109/TSP.2020.2983162

Na ZY, Liu Y, Cui Y, et al., 2015. Research on aggregation and propagation of self-similar traffic in satellite network. *Int J Hybr Inform Technol*, 8:325-338. https://doi.org/10.14257/ijhit.2015.8.1.29

Nguyen HD, Vu TL, Slotine JJ, et al., 2021. Contraction analysis of nonlinear DAE systems. *IEEE Trans Autom Contr*, 66(1):429-436. https://doi.org/10.1109/TAC.2020.2981348

Pan ZF, Wang YL, Wang K, et al., 2023. Imputation of missing values in time series using an adaptive-learned median-filled deep autoencoder. *IEEE Trans Cybern*, 53(2):695-706. https://doi.org/10.1109/TCYB.2022.3167995

Shen LF, Ma QL, Li S, 2018. End-to-end time series imputation via residual short paths. Proc 10th Asian Conf on Machine Learning Research, p.248-263.

Su T, Liu YB, Zhao JB, et al., 2021. Probabilistic stacked denoising autoencoder for power system transient stability prediction with wind farms. *IEEE Trans Power Syst*, 36(4):3786-3789. https://doi.org/10.1109/TPWRS.2020.3043620

Tang D, Wang SY, Liu BR, et al., 2023. GASF-IPP: detection and mitigation of LDoS attack in SDN. *IEEE Trans Serv Comput*, 16(5):3373-3384. https://doi.org/10.1109/TSC.2023.3266757

Tao HM, Deng QQ, Xiao SZ, 2020. Reconstruction of time series with missing value using 2D representation-based denoising autoencoder. *J Syst Eng Electron*, 31(6):1087-1096. https://doi.org/10.23919/JSEE.2020.000081

Tao XL, Liu ZY, Zhao F, et al., 2023. An SSA-LC-DAE method for extracting network security elements. *IEEE Trans Netw Sci Eng*, 10(2):1175-1185. https://doi.org/10.1109/TNSE.2023.3233986

Tasdemir Y, Kolay E, Kayabali K, 2013. Comparison of three artificial neural network approaches for estimating of slake durability index. *Environ Earth Sci*, 68:23-31. https://doi.org/10.1007/s12665-012-1702-3

Wang A, Ye YC, Song XZ, et al., 2023. Traffic prediction with missing data: a multi-task learning approach. *IEEE Trans Intell Transp Syst*, 24(4):4189-4202. https://doi.org/10.1109/TITS.2022.3233890

Wang HY, Zhao JP, Su YS, et al., 2022. scCDG: a method based on DAE and GCN for scRNA-seq data analysis. *IEEE/ACM Trans Comput Biol Bioinform*, 19(6):3685-3694. https://doi.org/10.1109/TCBB.2021.3126641

Ye L, Hu SB, Yan TT, et al., 2023. GAF representation of millimeter wave drone RCS and drone classification method based on deep fusion network using ResNet. *IEEE Trans Aerosp Electron Syst*, 59(1):336-346. https://doi.org/10.1109/TAES.2022.3182303

Yoon J, Jordon J, Schaar M, 2018. GAIN: missing data imputation using generative adversarial nets. Proc 35th Int Conf on Machine Learning, p.5689-5698.

You WB, Ding YH, Yao Y, 2020. Static explosion field reconstruction based on the improved biharmonic spline interpolation. *IEEE Sens J*, 20(13):7235-7240. https://doi.org/10.1109/JSEN.2020.2978502

Zhang CH, Yu JJQ, Liu Y, 2019. Spatial-temporal graph attention networks: a deep learning approach for traffic forecasting. *IEEE Access*, 7:166246-166256. https://doi.org/10.1109/ACCESS.2019.2953888

Zhang JJ, Mu XD, Fang JS, et al., 2019. Time series imputation via integration of revealed information based on the residual shortcut connection. *IEEE Access*, 7:102397-102405. https://doi.org/10.1109/ACCESS.2019.2928641

Zhao L, Song YJ, Zhang C, et al., 2020. T-GCN: a temporal graph convolutional network for traffic prediction. *IEEE Trans Intell Transp Syst*, 21(9):3848-3858. https://doi.org/10.1109/TITS.2019.2935152

Zhou X, Shi J, Gong K, et al., 2021. A novel quench detection method based on CNN-LSTM model. *IEEE Trans Appl Supercond*, 31(5):4702105. https://doi.org/10.1109/TASC.2021.3070735