Laplacian Convolutional Representation for Traffic Time Series Imputation

Xinyu Chen, Zhanhong Cheng, Nicolas Saunier, Lijun Sun, *Senior Member, IEEE*

Abstract—Spatiotemporal traffic data imputation is of great significance in intelligent transportation systems and data-driven decision-making processes. To make an accurate reconstruction from partially observed traffic data, we assert the importance of characterizing both global and local trends in time series. In the literature, substantial works have demonstrated the effectiveness of utilizing low-rankness property of traffic data by matrix/tensor completion models. In this study, we first introduce a Laplacian kernel to temporal regularization for characterizing local trends in traffic time series, which can be formulated as circular convolution. Then, we develop a low-rank Laplacian convolutional representation (LCR) model by putting the nuclear norm of a circulant matrix and the Laplacian kernelized temporal regularization together, which is proved to meet a unified framework that takes a fast Fourier transform (FFT) solution in a relatively low time complexity. Through extensive experiments on some traffic datasets, we demonstrate the superiority of LCR for imputing traffic time series of various time series behaviors (e.g., data noises and strong/weak periodicity) and reconstructing sparse speed field. The proposed LCR model is an efficient and effective solution to large-scale traffic data imputation over the existing baseline models. Despite the LCR's application to temporal modeling, the key idea lies in bridging the low-rank models and the Laplacian kernelized regularization through FFT, which is well-suited to spatiotemporal reconstruction.

Index Terms—Spatiotemporal traffic data, time series imputation, low-rank models, Laplacian kernelized regularization, circular convolution, discrete Fourier transform

1 Introduction

Missing data imputation is a fundamental component to a wide range of applications in modern intelligent transportation systems (ITS), including route planning, travel time estimation, and traffic flow forecasting. Typically, traffic data can be collected by fixed sensors (e.g., loop detectors, video cameras, and radars) on a continuous basis, producing a sequence of time series measurements of traffic flow. Since real-world ITS often suffers from various operational issues (e.g., sensor failure and network communication problems), these lead to data corruption and sparsity. Making accurate recovery on these data is important for supporting ITS applications, and still demands appropriate imputation approaches.

For traffic time series, the basic modeling idea of missing data imputation is to exploit the spatial and temporal correlations/dependencies from partial observations. Typically, traffic flow data always show strong global and local trends with long-term and short-term patterns [1]. The global trends usually refer to certain periodic, seasonal, and cyclical patterns, which can be well characterized by low-rank models. Unfortunately, conventional low-rank models fail to characterize the sequential dynamics of time series because the reconstruction of low-rank models (e.g., low-rank matrix completion (LRMC) [2], [3]) is invariant to the permutation of rows and columns. Therefore, recent studies presented low-rank time series completion models based on certain algebraic structures, including the Hankel matrices/tensors [4], circulant/anti-circulant matrices/tensors [5], [6], and convolution matrices [6], [7]. The sequential dependencies are implicitly captured by delay embedding when characterizing the low-rankness. However, low-rank Hankel/convolution models are always limited to small-/middle-scale problems due to the large size of delay-embedded matrices/tensors. Although circulant matrix nuclear norm minimization (CircNNM) can be efficiently solved via the use of the fast Fourier transform (FFT), compared with convolution matrices, circulant matrices are restricted and fail to characterize the local trends of time series [6].

Regarding the low-rank framework, the default structure of most low-rank models does not ensure local smoothness properties. Thus, it requires us to model both global patterns and local trends in a unified low-rank framework. In the literature, there are several ways to characterize the local spatial and temporal dependencies in data-driven machine learning models. For example, on the spatial dimension, Laplacian regularization has become a standard technique to impose local consistency on a network (e.g., [8], [9]). On the temporal dimension, the local smoothness property is often characterized using temporal regularization and time series autoregression explicitly (see e.g., [10], [11], [12]).

Considering the importance of temporal regularization techniques in regulating the behavior of global low-rank models, we are inspired to develop a tailored regularizer for CircNNM to improve its capability of local trends modeling while maintaining the algorithm efficiency through FFT. Therefore, using the fact that the Laplacian matrix of a circulant graph is a circulant matrix, we first introduce a

- *X. Chen and N. Saunier are with the Civil, Geological and Mining Engineering Department, Polytechnique Montreal, Montreal, QC H3T 1J4, Canada. E-mail: chenxy346@gmail.com (X. Chen), nicolas.saunier@polymtl.ca (N. Saunier).*
- *Z. Cheng and L. Sun are with the Department of Civil Engineering, McGill University, Montreal, QC H3A 0C3, Canada. E-mail: zhanhong.cheng@mail.mcgill.ca (Z. Cheng), lijun.sun@mcgill.ca (L. Sun).*

(Corresponding author: Nicolas Saunier)

novel Laplacian-matrix-based temporal regularization to the CircNNM problem. Next, we develop an efficient algorithm to solve the proposed Laplacian convolutional representation (LCR) using the alternating direction method of multipliers (ADMM) and FFT. The proposed imputation method is evaluated on several real-world traffic flow datasets. The experimental results demonstrate that the proposed method can achieve better performance than the state-of-the-art baseline methods in terms of both accuracy and efficiency. The contribution of this work is three-fold:

- We introduce a circular Laplacian kernel and use it to define a temporal regularization term for characterizing the local trends in traffic time series. By doing so, the Laplacian kernelized temporal regularization can be formulated with circular convolution.
- We propose a low-rank completion model—LCR—by characterizing global trends of traffic time series as the nuclear norm of a circulant matrix and modeling local trends by the temporal regularization simultaneously. According to the properties of the circulant matrix and circular convolution, we present a fast implementation of LCR through FFT.
- We demonstrate the imputation performance of LCR on some real-world traffic flow datasets and verify the importance of temporal regularization in LCR. The model is well-suited to sparse traffic flow time series that are characterized by both seasonality and non-seasonality. In addition, due to the fast implementation via FFT with relatively low complexity, the proposed model is scalable to large and high-dimensional imputation problems.

The remainder of this paper is structured as follows. Section 2 introduces the related work. Section 3 reviews the basic definitions of circulant matrix and convolution operator. In Section 4, we integrate the temporal regularization into low-rank models for characterizing both global and local trends in traffic time series. Section 5 and Section 6 conduct imputation experiments on real-world traffic time series datasets. Finally, we conclude this study in Section 7.

2 Related Work

This section reviews the related work on low-rank completion algorithms. The review consists of two aspects: 1) low-rank matrix/tensor completion models based on special algebraic structures (e.g., circulant/Hankel matrices) and 2) temporal modeling techniques in matrix/tensor completion.

2.1 Low-Rank Completion with Algebraic Structures

Recent studies show great interest in low-rank time series completion models based on certain algebraic structures, e.g., Hankel and circulant matrices. These approaches overcome the limitations of pure LRMC models, such as being incapable when an entire row/column is missing and invariant to the permutation of rows/columns, or in the case of univariate time series. For example, the model developed by Yokota *et al*. can recover the missing slices of tensor using Hankelization [4]. Sedighin *et al*. applied tensor train decomposition to tensors obtained by extended multi-way delay embedded transform and found better completion performance [13].

A critical property of a circulant matrix is that its nuclear norm can be efficiently obtained via FFT. Using this property, Yamamoto *et al*. proposed a fast tensor completion in delay embedded space [5]; Liu and Zhang used the nuclear norm minimization of circulant matrices for missing data recovery and time series forecasting [6]. Despite the fast algorithm on the circulant matrix, circulant-matrix-based models are inadequate in capturing the local trend/continuity in time series. Therefore, convolution nuclear norm minimization (ConvNNM) model shows better local trends modeling if the kernel size of the convolution matrix is set as a relative small value [6]. Further, Liu proposed a learnable and orthonormal transformation for ConvNNM to reinforce its modeling ability when the convolutional low-rankness condition is not satisfied [7].

2.2 Imputation with Temporal Modeling

A large body of previous research has leveraged temporal dynamics in low-rank models for time series imputation. A common assumption among these models is that time series and their low-rank factors have local dependencies. In terms of local dependencies, Chen and Cichocki proposed a Toeplitz-matrix-based regularizer to impose temporal smoothness in matrix factorization [14]; the regularizer penalizes the difference between the low-rank factors of consecutive time points. A similar regularizer based on the Toeplitz matrix was used by [15] for traffic data reconstruction. Chen *et al.* applied a quadratic variation (QV) to a traffic tensor completion problem to ensure temporal smoothness [16]. In terms of modeling temporal dynamics of low-rank factors, Xiong *et al.* formulated a Bayesian tensor factorization with first-order Markovian assumptions on the temporal factors [10]. Yu *et al.* developed a temporal autoregressive regularizer in matrix factorization[11]. Chen *et al.* developed a low-rank autoregressive tensor completion model for traffic data imputation [17]. The above two works assume the independent autoregression for each latent factor, Chen and Sun applied a vector autoregression (VAR) on the latent factors and developed a fully Bayesian solution for multidimensional time series prediction [18]. Chen *et al.* imposed the VAR process on seasonal differenced copy of the temporal factor matrix for nonstationary temporal matrix factorization [12].

Many studies used Gaussian Process (GP) priors to latent temporal factors for temporal/spatial smoothness [19], [20], i.e., kernelized matrix/tensor factorization. Lei *et al.* proposed a fully Bayesian solution by an efficient Markov chain Monte Carlo sampling algorithm for kernelized matrix factorization [21]. In fact, the maximum a posteriori estimation of using a proper GP prior yields the same form as a Laplacian regularization, where the Laplacian matrix is the inverse of the covariance matrix of the GP prior [22], [20]. As Laplacian regularization is of broad use in graph modeling, it is also applicable to temporal modeling. For example, Rao *et al.* proposed a scalable collaborative filtering algorithm with spatial/temporal Laplacian regularization for matrix data imputation [23]. To the best of our knowledge, we are the first to present Laplacian kernelized

temporal regularization in the form of circular convolution and thus lead to the use of FFT.

3 Preliminaries

In this section, we first introduce the basic definitions of circulant matrix, convolution matrix, and circular convolution. In the meanwhile, we give a summary of their relationships. Then, we discuss the modeling process of the convolution matrix, circulant matrix, and Hankel matrix.

3.1 Circulant Matrix

Typically, circulant matrix is an important structure that shows broad use in the field of signal processing [24], [25]. By definition, for any vector $\boldsymbol{x} = (x_1, x_2, \cdots, x_T)^\top \in \mathbb{R}^T$, the circulant matrix can be written as follows,

$$\mathcal{C}(\boldsymbol{x}) \triangleq \begin{bmatrix} x_1 & x_T & x_{T-1} & \cdots & x_2 \\ x_2 & x_1 & x_T & \cdots & x_3 \\ x_3 & x_2 & x_1 & \cdots & x_4 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ x_T & x_{T-1} & x_{T-2} & \cdots & x_1 \end{bmatrix} \in \mathbb{R}^{T\times T}, \quad (1)$$

where $\mathcal{C} : \mathbb{R}^T \to \mathbb{R}^{T\times T}$ denotes the circulant operator. The first column of $\mathcal{C}(\boldsymbol{x})$ is the vector $\boldsymbol{x}$, and the diagonal entries of $\mathcal{C}(\boldsymbol{x})$ are all about x_1.

3.2 Convolution Matrix

Convolution is vital to a variety of machine learning problems. By definition, for any vectors $\boldsymbol{x} = (x_1, x_2, \cdots, x_T)^\top \in \mathbb{R}^T$ and $\boldsymbol{y} = (y_1, y_2, \cdots, y_{\tilde{\tau}})^\top \in \mathbb{R}^\tau$ with $\tilde{\tau} \leq T$, the circular convolution of two vectors is $\boldsymbol{z} = \boldsymbol{x} \star \boldsymbol{y} \in \mathbb{R}^T$ [26], denoting the operator with the symbol $\star$; element-wise, we have

$$z_t = \sum_{k=1}^{\tilde{\tau}} x_{t-k+1} y_k, \ \forall t \in \{1, 2, \ldots, T\}, \quad (2)$$

where z_t is the tth entry of $\boldsymbol{z}$ and $x_{t-k+1} = x_{t-k+1+T}$ for $t+1 \leq k$. In particular, circular convolution is a linear operator that can be expressed as follows,

$$\boldsymbol{x} \star \boldsymbol{y} \equiv \mathcal{C}_{\tilde{\tau}}(\boldsymbol{x})\boldsymbol{y}, \quad (3)$$

where $\mathcal{C}_{\tilde{\tau}} : \mathbb{R}^T \to \mathbb{R}^{T\times\tilde{\tau}}$ denotes the convolution operator and the resultant convolution matrix [6], [7] is given by

$$\mathcal{C}_{\tilde{\tau}}(\boldsymbol{x}) \triangleq \begin{bmatrix} x_1 & x_T & x_{T-1} & \cdots & x_{T-\tilde{\tau}+2} \\ x_2 & x_1 & x_T & \cdots & x_{T-\tilde{\tau}+3} \\ x_3 & x_2 & x_1 & \cdots & x_{T-\tilde{\tau}+4} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ x_T & x_{T-1} & x_{T-2} & \cdots & x_{T-\tilde{\tau}+1} \end{bmatrix} \in \mathbb{R}^{T\times\tilde{\tau}}, \quad (4)$$

where $\tilde{\tau}$ is the kernel size.

Remark. *Following Eq. (3), for any $\boldsymbol{x} = (x_1, x_2, \cdots, x_T)^\top \in \mathbb{R}^T$ and $\boldsymbol{y} = (y_1, y_2, \cdots, y_T)^\top \in \mathbb{R}^T$ (i.e., $\tilde{\tau} = T$), then we have the following property:*

$$\boldsymbol{x} \star \boldsymbol{y} \equiv \mathcal{C}(\boldsymbol{x})\boldsymbol{y}, \quad (5)$$

where $\mathcal{C}(\boldsymbol{x})$ is the circulant matrix as defined in Eq. (1).

Essentially, a convolution matrix consists of the first $\tilde{\tau}$ columns of the circulant matrix. For any vector $\boldsymbol{x} \in \mathbb{R}^T$, it always holds that $\|\boldsymbol{x}\|_2^2 = \|\mathcal{C}_{\tilde{\tau}}(\boldsymbol{x})\|_F^2/\tilde{\tau}$ where $\|\cdot\|_2$ and $\|\cdot\|_F$ denote the ℓ_2-norm of a vector and the Frobenius norm of a matrix, respectively. Unlike the convolution operator, suppose $\mathcal{H}_{\tilde{\tau}} : \mathbb{R}^T \to \mathbb{R}^{(T-\tilde{\tau}+1)\times\tilde{\tau}}$ denotes the Hankel operator [5], then $\|\boldsymbol{x}\|_2^2 \not\propto \|\mathcal{H}_{\tilde{\tau}}(\boldsymbol{x})\|_F^2$, showing that Hankel structure cannot preserve the consistency between data and its Hankelization.

Typically, ConvNNM can be converted into a standard nuclear norm minimization with singular value thresholding [3], [2], [27], [6], [7]. In contrast to ConvNNM, CircNNM has an efficient solution through FFT and shows better generalization to high-dimensional data [7].

4 Methodology

In this section, we introduce an LCR model for imputing sparse traffic time series. The cornerstone of LCR includes the definition of the Laplacian kernel. To resolve the involved optimization problem, we seek a fast implementation through FFT in the frequency domain, instead of the time domain.

4.1 Laplacian Kernel

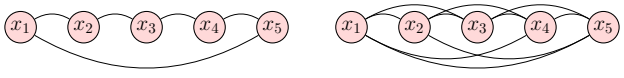
While graph modeling brings new insights into relational data analysis, we propose to characterize the temporal dependencies of time series through undirected and circulant graphs. Recall that the Laplacian matrix by definition takes $\boldsymbol{L} = \boldsymbol{D} - \boldsymbol{A}$ in which $\boldsymbol{D}$ and $\boldsymbol{A}$ are the (diagonal) degree matrix and adjacency matrix, respectively [28]. As exemplified by Fig. 1, the Laplacian matrices of these two graphs are

$$\boldsymbol{L} = \begin{bmatrix} 2 & -1 & 0 & 0 & -1 \\ -1 & 2 & -1 & 0 & 0 \\ 0 & -1 & 2 & -1 & 0 \\ 0 & 0 & -1 & 2 & -1 \\ -1 & 0 & 0 & -1 & 2 \end{bmatrix}, \quad (6)$$

and

$$\boldsymbol{L} = \begin{bmatrix} 4 & -1 & -1 & -1 & -1 \\ -1 & 4 & -1 & -1 & -1 \\ -1 & -1 & 4 & -1 & -1 \\ -1 & -1 & -1 & 4 & -1 \\ -1 & -1 & -1 & -1 & 4 \end{bmatrix}, \quad (7)$$

respectively. Notably, both two matrices are circulant matrices, and their first columns are given by $\boldsymbol{\ell} = (2, -1, 0, 0, -1)^\top$ and $\boldsymbol{\ell} = (4, -1, -1, -1, -1)^\top$, respectively.



(a) Circulant graph with degree 2. (b) circulant graph with degree 4.

Fig. 1: Undirected and circulant graphs on the relational data samples $\{x_1, x_2, \ldots, x_5\}$ with certain degrees.

In this work, we introduce a Laplacian kernel as described in Definition 1, allowing one to encode temporal dependencies of time series. In above cases, the first column

of the Laplacian matrix $\boldsymbol{L}$ is indeed a simple example of a Laplacian kernel.

Definition 1 ((Circular) Laplacian Kernel). *Given any time series $\boldsymbol{x} = (x_1, \ldots, x_T)^\top \in \mathbb{R}^T$, suppose $\tau \in \mathbb{N}^+$ ($\tau \leq \frac{1}{2}(T-1)$) be the kernel size of an undirected and circulant graph, then the Laplacian kernel is defined as*

$$\boldsymbol{\ell} \triangleq (2\tau, \underbrace{-1, \cdots, -1}_{\tau}, 0, \cdots, 0, \underbrace{-1, \cdots, -1}_{\tau})^\top \in \mathbb{R}^T, \quad (8)$$

which is also the first column of the Laplacian matrix and the inherent degree matrix is diagonalized with entries 2τ.

Essentially, according to Definition 1, the corresponding Laplacian matrix $\boldsymbol{L} \in \mathbb{R}^{T\times T}$ can be characterized as a circulant matrix. Recall that the temporal regularization through Laplacian matrix is given by

$$\mathcal{R}_\tau(\boldsymbol{x}) = \frac{1}{2}\|\boldsymbol{L}\boldsymbol{x}\|_2^2 = \frac{1}{2}\sum_{i=1}^{T}\big(\sum_{j\in\Phi(i)}(x_i - x_j)\big)^2, \quad (9)$$

where $\Phi(i)$ is a set of adjacent nodes of x_i in the graph. The temporal regularization calculates how values of $\boldsymbol{x}$ differ from their adjacent values, and can thus be used as a measure/regularization of local temporal smoothness. According to the definitions of Laplacian kernel and circulant matrix (see Eq. (1)) and the relationship between circular convolution and circulant matrix (see Eq. (5)), we can now declare the following form:

$$\mathcal{R}_\tau(\boldsymbol{x}) = \frac{1}{2}\|\boldsymbol{L}\boldsymbol{x}\|_2^2 = \frac{1}{2}\|\mathcal{C}(\boldsymbol{\ell})\boldsymbol{x}\|_2^2 = \frac{1}{2}\|\boldsymbol{\ell}\star\boldsymbol{x}\|_2^2. \quad (10)$$

Remark. *We can obtain a more general form of the kernel by relaxing Definition 1 and allowing for a more flexible design of $\boldsymbol{\ell}$. For example, if we introduce a directed Laplacian kernel in the form of random walk [10], [8] (i.e., $\boldsymbol{\ell} = (1, 0, \ldots, 0, -1)^\top \in \mathbb{R}^T$) and reinforce the local trends in time series reconstruction, then the temporal regularization is equivalent to the QV regularization, namely,*

$$\frac{1}{2}\|\boldsymbol{\ell}\star\boldsymbol{x}\|_2^2 = \frac{1}{2}\boldsymbol{x}^\top\tilde{\boldsymbol{L}}\boldsymbol{x}, \quad (11)$$

where

$$\tilde{\boldsymbol{L}} = \begin{bmatrix} 2 & -1 & 0 & \cdots & 0 & 0 & -1 \\ -1 & 2 & -1 & \cdots & 0 & 0 & 0 \\ 0 & -1 & 2 & \cdots & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & -1 & 2 & -1 \\ -1 & 0 & 0 & \cdots & 0 & -1 & 2 \end{bmatrix}. \quad (12)$$

In what follows, we utilize the Convolution Theorem (see Theorem 1) to reformulate the temporal regularization in the frequency domain.

Theorem 1 (Convolution Theorem [26]). *For any vectors $\boldsymbol{x}, \boldsymbol{y} \in \mathbb{R}^T$, a circular convolution in the time domain is a product in the frequency domain, and it always holds that*

$$\boldsymbol{x}\star\boldsymbol{y} = \mathcal{F}^{-1}(\mathcal{F}(\boldsymbol{x})\circ\mathcal{F}(\boldsymbol{y})), \quad (13)$$

where $\mathcal{F}(\cdot)$ and $\mathcal{F}^{-1}(\cdot)$ denote the discrete Fourier transform and the inverse discrete Fourier transform, respectively. $\mathcal{F}(\boldsymbol{x}), \mathcal{F}(\boldsymbol{y}) \in \mathbb{C}^T$ are the results of discrete Fourier transform on $\boldsymbol{x}, \boldsymbol{y}$ with $\mathbb{C}$ denoting the set of complex numbers. The symbol $\circ$ denotes the Hadamard product.

Typically, Theorem 1 gives a concise description of relationship between circular convolution and discrete Fourier transform, and it shows that circular convolution can be implemented in the frequency domain. Circulant matrix is advantageous because the required matrix-vector product can usually be done efficiently by leveraging the structure. Since the Laplacian kernel in this case holds the property of circulant matrix (see Eq. (5)), the temporal regularization in Eq. (9) can be therefore reformulated as follows,

$$\mathcal{R}_\tau(\boldsymbol{x}) = \frac{1}{2}\|\boldsymbol{\ell}\star\boldsymbol{x}\|_2^2 = \frac{1}{2T}\|\mathcal{F}(\boldsymbol{\ell})\circ\mathcal{F}(\boldsymbol{x})\|_2^2. \quad (14)$$

As can be seen, the temporal regularization is converted into the formula that associated with the Laplacian kernel in the frequency domain. It seems that the length-T Laplacian kernel $\boldsymbol{\ell}$ can represent the graphical relationship of T-by-T Laplacian matrix $\boldsymbol{L}$ in the temporal regularization, showing no need for constructing $\boldsymbol{L}$. In the temporal regularization, the coefficients are governed by the Laplacian kernel.

Remark. *For Eq. (14), we can prove the statement as follows. Let*

$$\begin{cases} \boldsymbol{\alpha} = \boldsymbol{\ell}\star\boldsymbol{x}, \\ \boldsymbol{\beta} = \mathcal{F}(\boldsymbol{\ell})\circ\mathcal{F}(\boldsymbol{x}), \end{cases} \quad (15)$$

then it always holds that

$$\boldsymbol{\alpha} = \mathcal{F}^{-1}(\boldsymbol{\beta}) \quad or \quad \mathcal{F}(\boldsymbol{\alpha}) = \boldsymbol{\beta}. \quad (16)$$

Furthermore, according to the Parseval's theorem [26], we get

$$\|\boldsymbol{\alpha}\|_2^2 = \frac{1}{T}\|\mathcal{F}(\boldsymbol{\alpha})\|_2^2 = \frac{1}{T}\|\boldsymbol{\beta}\|_2^2, \quad (17)$$

as claimed in Eq. (14).

4.2 Univariate Time Series Imputation

Spatiotemporal traffic data modeling is of particular interest to many real-world ITS applications. Typically, traffic flow data by nature involve certain time series characteristics, e.g., global daily/weekly rhythm and local trends. However, such kind of time series are usually incomplete or even sparse due to unpredictable data collection processes. In the univariate case, the imputation problem can be summarized as Problem 1.

Problem 1 (Univariate Time Series Imputation). *For any partially observed time series $\boldsymbol{y} \in \mathbb{R}^T$ with observed index set Ω, the goal is to reconstruct the missing values, namely, $\mathcal{P}_\Omega^\perp(\boldsymbol{y})$, from the partial observations $\mathcal{P}_\Omega(\boldsymbol{y})$. Herein, $\mathcal{P}_\Omega : \mathbb{R}^T \to \mathbb{R}^T$ denotes the orthogonal projection supported on the observed index set Ω, while $\mathcal{P}_\Omega^\perp : \mathbb{R}^T \to \mathbb{R}^T$ denotes the orthogonal projection supported on the complement of Ω.*

Remark. *On the vector $\boldsymbol{y} \in \mathbb{R}^T$ with observed index set Ω, the operator $\mathcal{P}_\Omega(\cdot)$ can be described as follows,*

$$[\mathcal{P}_\Omega(\boldsymbol{y})]_t = \begin{cases} y_t, & \text{if } t \in \Omega, \\ 0, & \text{otherwise}, \end{cases} \quad (18)$$

where $t = 1, 2, \ldots, T$.

Although ConvNNM and CircNNM can reconstruct missing values in time series, both two models fail to incorporate global and local consistency appropriately. In this work, we propose a low-rank completion model, i.e., LCR,

in which we utilize circulant matrix nuclear norm to pursue the global trends and use the temporal regularization to characterize the local trends in time series (see Fig. 2 for an illustration). Formally, the LCR model can be formulated as follows,

$$\begin{aligned} \min_{\boldsymbol{x}} \ & \|\mathcal{C}(\boldsymbol{x})\|_* + \gamma \cdot \mathcal{R}_\tau(\boldsymbol{x}) \\ \text{s.t. } & \mathcal{P}_\Omega(\boldsymbol{x}) = \mathcal{P}_\Omega(\boldsymbol{y}), \end{aligned} \tag{19}$$

where $\|\cdot\|_*$ denotes the nuclear norm of matrix, which is the sum of singular values. The vector $\boldsymbol{x} \in \mathbb{R}^T$ is the reconstructed time series corresponding to the partially observed time series $\boldsymbol{y}$. In the objective function, γ is the weight parameter.

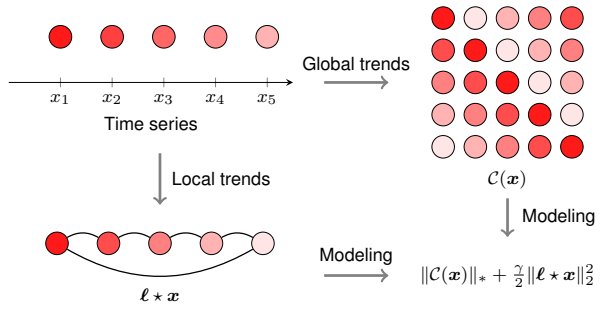


Fig. 2: Illustration of the proposed LCR model.

Since spatiotemporal traffic data are usually noisy, the strong observation constraint in Eq. (19) should be replaced by $\|\mathcal{P}_\Omega(\boldsymbol{z} - \boldsymbol{y})\|_2 \leq \epsilon$ in which $\epsilon \geq 0$ is the tolerance. Now, the optimization problem of LCR is given by

$$\begin{aligned} \min_{\boldsymbol{x}} \ & \|\mathcal{C}(\boldsymbol{x})\|_* + \gamma \cdot \mathcal{R}_\tau(\boldsymbol{x}) \\ \text{s.t. } & \|\mathcal{P}_\Omega(\boldsymbol{x} - \boldsymbol{y})\|_2 \leq \epsilon. \end{aligned} \tag{20}$$

Our LCR model stems from ConvNNM [6], [7], and it can be solved by the ADMM framework. To resolve the convex optimization problem of LCR in Eq. (20), we introduce an auxiliary variable $\boldsymbol{z}$ to preserve the observation information. Thus, the optimization problem now becomes

$$\begin{aligned} \min_{\boldsymbol{x},\boldsymbol{z}} \ & \|\mathcal{C}(\boldsymbol{x})\|_* + \gamma \cdot \mathcal{R}_\tau(\boldsymbol{x}) + \eta \cdot \pi(\boldsymbol{z}) \\ \text{s.t. } & \boldsymbol{x} = \boldsymbol{z}, \end{aligned} \tag{21}$$

where η is the weight parameter. We define $\pi(\cdot)$ corresponding to the reconstructed errors between $\boldsymbol{z}$ and $\boldsymbol{y}$ in the set Ω, which is formally given by

$$\pi(\boldsymbol{z}) = \frac{1}{2}\|\mathcal{P}_\Omega(\boldsymbol{z} - \boldsymbol{y})\|_2^2, \tag{22}$$

To reinforce both global and local trends in the reconstructed time series, the observation constraint can be related to the noisy version as shown in Eq. (20), thus leading to the denoised and smooth time series in $\boldsymbol{x}$.

Accordingly, the augmented Lagrangian function of Eq. (21) can be written as follows,

$$\begin{aligned} \mathcal{L}(\boldsymbol{x},\boldsymbol{z},\boldsymbol{w}) =& \|\mathcal{C}(\boldsymbol{x})\|_* + \gamma \cdot \mathcal{R}_\tau(\boldsymbol{x}) + \frac{\lambda}{2}\|\boldsymbol{x} - \boldsymbol{z}\|_2^2 \\ & + \langle \boldsymbol{w}, \boldsymbol{x} - \boldsymbol{z} \rangle + \eta \cdot \pi(\boldsymbol{z}), \end{aligned} \tag{23}$$

where $\boldsymbol{w} \in \mathbb{R}^T$ is the Lagrange multiplier, and λ is the hyperparameter. The symbol $\langle \cdot, \cdot \rangle$ denotes the inner product. Note that the constraint $\boldsymbol{x} = \boldsymbol{z}$ in the optimization problem is relaxed by the Lagrange multiplier.

Thus, the ADMM scheme can be summarized as follows,

$$\begin{cases} \boldsymbol{x} := \arg\min_{\boldsymbol{x}} \mathcal{L}(\boldsymbol{x},\boldsymbol{z},\boldsymbol{w}), \\ \boldsymbol{z} := \arg\min_{\boldsymbol{z}} \mathcal{L}(\boldsymbol{x},\boldsymbol{z},\boldsymbol{w}), \\ \boldsymbol{w} := \boldsymbol{w} + \lambda(\boldsymbol{x} - \boldsymbol{z}). \end{cases} \tag{24}$$

In particular, with respect to the variable $\boldsymbol{x}$, if we rewrite the regularization terms in Eq. (24) as follows,

$$f = \gamma \cdot \mathcal{R}_\tau(\boldsymbol{x}) + \frac{\lambda}{2}\|\boldsymbol{x} - \boldsymbol{z} + \boldsymbol{w}/\lambda\|_2^2, \tag{25}$$

then the formula for $\mathcal{R}_\tau(\boldsymbol{x})$ can be converted into a discrete Fourier transform copy (see Eq. (14)), and the Parseval's theorem is also applicable to the remaining term of f. Thus, the above formula for f is equivalent to

$$\begin{aligned} f =& \frac{\gamma}{2}\|\boldsymbol{\ell} \star \boldsymbol{x}\|_2^2 + \frac{\lambda}{2}\|\boldsymbol{x} - \boldsymbol{z} + \boldsymbol{w}/\lambda\|_2^2 \\ =& \frac{\gamma}{2T}\|\mathcal{F}(\boldsymbol{\ell}) \circ \mathcal{F}(\boldsymbol{x})\|_2^2 + \frac{\lambda}{2T}\|\mathcal{F}(\boldsymbol{x} - \boldsymbol{z} + \boldsymbol{w}/\lambda)\|_2^2 \\ =& \frac{\gamma}{2T}\|\hat{\boldsymbol{\ell}} \circ \hat{\boldsymbol{x}}\|_2^2 + \frac{\lambda}{2T}\|\hat{\boldsymbol{x}} - \hat{\boldsymbol{z}} + \hat{\boldsymbol{w}}/\lambda\|_2^2, \end{aligned} \tag{26}$$

where $\hat{\boldsymbol{\ell}} = \mathcal{F}(\boldsymbol{\ell})$, and we introduce the variables $\{\hat{\boldsymbol{x}}, \hat{\boldsymbol{z}}, \hat{\boldsymbol{w}}\} = \{\mathcal{F}(\boldsymbol{x}), \mathcal{F}(\boldsymbol{z}), \mathcal{F}(\boldsymbol{w})\}$ referring to the variables $\{\boldsymbol{x}, \boldsymbol{z}, \boldsymbol{w}\}$ in the frequency domain. Accordingly, the partial derivative of the function f with respect to the variable $\hat{\boldsymbol{x}}$ is given by

$$\begin{aligned} \frac{\partial f}{\partial \hat{\boldsymbol{x}}} =& \frac{\gamma}{T}\hat{\boldsymbol{\ell}}^* \circ \hat{\boldsymbol{\ell}} \circ \hat{\boldsymbol{x}} + \frac{\lambda}{T}(\hat{\boldsymbol{x}} - \hat{\boldsymbol{z}} + \hat{\boldsymbol{w}}/\lambda) \\ =& \frac{1}{T}(\gamma\hat{\boldsymbol{\ell}}^* \circ \hat{\boldsymbol{\ell}} + \lambda\mathbb{1}_T) \circ \hat{\boldsymbol{x}} - \frac{1}{T}(\lambda\hat{\boldsymbol{z}} - \hat{\boldsymbol{w}}), \end{aligned} \tag{27}$$

where $\mathbb{1}_T \in \mathbb{R}^T$ is the vector comprised of ones. The notation $\cdot^*$ represents the complex conjugate.

Letting $\frac{\partial f}{\partial \hat{\boldsymbol{x}}} = \boldsymbol{0}$ produces a closed-form solution such that

$$\hat{\boldsymbol{x}} = (\lambda\hat{\boldsymbol{z}} - \hat{\boldsymbol{w}}) \oslash (\gamma\hat{\boldsymbol{\ell}}^* \circ \hat{\boldsymbol{\ell}} + \lambda\mathbb{1}_T), \tag{28}$$

where $\oslash$ denotes the Hadamard division. Switching to the function f and using the closed-form solution in Eq. (28), it suggests an appropriate form to reformulate the remaining regularization terms together (see Eq. (25)), namely,

$$f = \frac{\lambda}{2T}\|\hat{\boldsymbol{x}} - (\lambda\hat{\boldsymbol{z}} - \hat{\boldsymbol{w}}) \oslash (\gamma\hat{\boldsymbol{\ell}}^* \circ \hat{\boldsymbol{\ell}} + \lambda\mathbb{1}_T)\|_2^2. \tag{29}$$

Lemma 1. *For any vector $\boldsymbol{x} \in \mathbb{R}^T$, the nuclear norm of the resultant circulant matrix $\mathcal{C}(\boldsymbol{x}) \in \mathbb{R}^{T\times T}$ is related to the discrete Fourier transform:*

$$\|\mathcal{C}(\boldsymbol{x})\|_* = \|\mathcal{F}(\boldsymbol{x})\|_1. \tag{30}$$

Proof. For any circulant matrix $\mathcal{C}(\boldsymbol{x})$ if and only if it is diagonalizable by the unitary matrix, the eigenvalue decomposition [25] can be written as follows,

$$\mathcal{C}(\boldsymbol{x}) = \boldsymbol{U}\text{diag}(\mathcal{F}(\boldsymbol{x}))\boldsymbol{U}^H, \tag{31}$$

where $\cdot^H$ denotes the conjugate transpose. Since $\boldsymbol{U}$ is a unitary matrix, it always holds that

$$\|\mathcal{C}(\boldsymbol{x})\|_* = \|\boldsymbol{U}\text{diag}(\mathcal{F}(\boldsymbol{x}))\boldsymbol{U}^H\|_* = \|\text{diag}(\mathcal{F}(\boldsymbol{x}))\|_* = \|\mathcal{F}(\boldsymbol{x})\|_1,$$

and we can calculate the singular values of $\mathcal{C}(\boldsymbol{x})$ from the FFT of $\boldsymbol{x}$, i.e., $\mathcal{F}(\boldsymbol{x})$. Here, FFT is an efficient algorithm for computing the discrete Fourier transform in $\mathcal{O}(T\log T)$ time. □

Going back to the ADMM scheme and using the property of circulant matrix nuclear norm in Lemma 1, the subproblem for optimizing the variable $\boldsymbol{x}$ in the time domain can be converted into the optimization over the variable $\hat{\boldsymbol{x}}$ in the frequency domain. Thus,

$$\boldsymbol{x} := \arg\min_{\boldsymbol{x}} \ \|\mathcal{C}(\boldsymbol{x})\|_* + \frac{\gamma}{2}\|\boldsymbol{\ell} \star \boldsymbol{x}\|_2^2 + \frac{\lambda}{2}\|\boldsymbol{x} - \boldsymbol{z} + \boldsymbol{w}/\lambda\|_2^2, \tag{32}$$

is equivalent to

$$\hat{\boldsymbol{x}} := \arg\min_{\hat{\boldsymbol{x}}} \|\hat{\boldsymbol{x}}\|_1 + \frac{\lambda}{2T}\|\hat{\boldsymbol{x}} - (\lambda\hat{\boldsymbol{z}} - \hat{\boldsymbol{w}}) \oslash (\gamma\hat{\boldsymbol{\ell}}^* \circ \hat{\boldsymbol{\ell}} + \lambda\mathbb{1}_T)\|_2^2. \tag{33}$$

The resultant ℓ_1-norm minimization is memory efficient, easy to compute with, and preserves the singular values of circulant matrix that are due to the FFT. The closely related analysis and results are also discussed in [5], [6], [7].

In Eq. (33), let

$$\hat{\boldsymbol{h}} \triangleq (\lambda\hat{\boldsymbol{z}} - \hat{\boldsymbol{w}}) \oslash (\gamma\hat{\boldsymbol{\ell}}^* \circ \hat{\boldsymbol{\ell}} + \lambda\mathbb{1}_T), \tag{34}$$

then the closed-form solution to $\hat{\boldsymbol{x}}$ can be found in Lemma 2. As we have the closed-form solution as described in Eq. (37) in the frequency domain, we can update the variable $\boldsymbol{x}$ by

$$\boldsymbol{x} := \mathcal{F}^{-1}(\hat{\boldsymbol{x}}). \tag{35}$$

Lemma 2. *Following Eq. (33), for any optimization problem in the form of ℓ_1-norm minimization in complex space, i.e.,*

$$\min_{\hat{\boldsymbol{x}}} \ \|\hat{\boldsymbol{x}}\|_1 + \frac{\lambda}{2T}\|\hat{\boldsymbol{x}} - \hat{\boldsymbol{h}}\|_2^2, \tag{36}$$

with complex-valued $\hat{\boldsymbol{x}}, \hat{\boldsymbol{h}} \in \mathbb{C}^T$, element-wise, the solution is given by

$$\hat{x}_t := \frac{\hat{h}_t}{|\hat{h}_t|} \cdot \max\{0, |\hat{h}_t| - T/\lambda\}, t = 1, \ldots, T. \tag{37}$$

Proof. In theory, Lemma 2 invokes the shrinkage operator in [29], [30], [6]. □

In the ADMM scheme (see Eq. (24)), the subproblem with respect to the variable $\boldsymbol{z}$ can be written as follows,

$$\min_{\boldsymbol{z}} \ \frac{\lambda}{2}\|\boldsymbol{x} - \boldsymbol{z} - \boldsymbol{w}/\lambda\|_2^2 + \frac{\eta}{2}\|\mathcal{P}_\Omega(\boldsymbol{z} - \boldsymbol{y})\|_2^2, \tag{38}$$

if we let g be the objective function, then the partial derivative with respect to the variable $\boldsymbol{z}$ can be formed by $\mathcal{P}_\Omega(\boldsymbol{z})$ and $\mathcal{P}_\Omega^\perp(\boldsymbol{z})$. Here, we have

$$\begin{aligned} \frac{\partial g}{\partial \mathcal{P}_\Omega(\boldsymbol{z})} &= \lambda\mathcal{P}_\Omega(\boldsymbol{z} - \boldsymbol{x} - \boldsymbol{w}/\lambda) + \eta\mathcal{P}_\Omega(\boldsymbol{z} - \boldsymbol{y}), \\ \frac{\partial g}{\partial \mathcal{P}_\Omega^\perp(\boldsymbol{z})} &= \lambda\mathcal{P}_\Omega^\perp(\boldsymbol{z} - \boldsymbol{x} - \boldsymbol{w}/\lambda). \end{aligned} \tag{39}$$

In what follows, $\frac{\partial g}{\partial \boldsymbol{z}} = \boldsymbol{0}$ allows one to infer a closed-form solution to the variable $\boldsymbol{z}$:

$$\boldsymbol{z} := \frac{1}{\lambda + \eta}\mathcal{P}_\Omega(\lambda\boldsymbol{x} + \boldsymbol{w} + \eta\boldsymbol{y}) + \frac{1}{\lambda}\mathcal{P}_\Omega^\perp(\lambda\boldsymbol{x} + \boldsymbol{w}). \tag{40}$$

Remark. *If $\eta \to +\infty$, then the solution would be $\boldsymbol{z} := \mathcal{P}_\Omega(\boldsymbol{y}) + \mathcal{P}_\Omega^\perp(\boldsymbol{x} + \boldsymbol{w}/\lambda)$, corresponding to the LCR model with strong observation constraint in Eq. (19). In terms of the parameter η, we can preferably set the default one as $\eta = c \cdot \lambda$ with $c \in \{10^2, 10^3\}$ to preserve the observation information.*

As mentioned above, our LCR model bridges the gap between the modeling processes of global low-rankness and local temporal trends underlying time series data, therefore reinforcing both global and local consistency. Since we utilize circulant matrix and circular convolution, it is not hard to show the appealing properties of discrete Fourier transform and lead to an elegant and fast solution algorithm. Algorithm 1 summarizes the implementation of the proposed LCR model.

Algorithm 1 Laplacian Convolutional Representation (LCR)

Input: Data $\boldsymbol{y} \in \mathbb{R}^T$ with observed index set Ω, Laplacian kernel size $\tau \in \mathbb{N}^+$, and hyperparameters $\{\gamma, \lambda, \eta\}$.
Output: Reconstructed vector $\boldsymbol{x} \in \mathbb{R}^T$.

```
Initialize {x_0, z_0, w_0}.
Construct the Laplacian kernel ℓ with τ and perform FFT on it to get ℓ̂.
for i = 0 to maximum iteration do
    Perform FFT on {z_i, w_i}.
    Compute ĥ by Eq. (34).
    Compute x̂ by the shrinkage in Eq. (37).
    Compute x_{i+1} by x_{i+1} = F^{-1}(x̂) (see Eq. (35)).
    Compute z_{i+1} by Eq. (40).
    Compute w_{i+1} = w_i + λ(x_{i+1} − z_{i+1}) (see Eq. (24)).
end for
```

4.3 Multivariate Time Series Imputation

Considering both spatial and temporal dimensions in traffic data, we have a multivariate time series imputation task as described in Problem 2. The critical question is how to characterize both spatial and temporal dependencies of traffic time series data in the modeling process.

Problem 2 (Multivariate Time Series Imputation)**.** *For any partially observed time series $\boldsymbol{Y} \in \mathbb{R}^{N\times T}$ consisting of N variables and T time steps, if its observed index set is denoted by Ω, then the goal is to reconstruct the missing values, namely, $\mathcal{P}_\Omega^\perp(\boldsymbol{Y})$, from the partial observations $\mathcal{P}_\Omega(\boldsymbol{Y})$. Herein, $\mathcal{P}_\Omega : \mathbb{R}^{N\times T} \to \mathbb{R}^{N\times T}$ denotes the orthogonal projection supported on the observed index set Ω, while $\mathcal{P}_\Omega^\perp : \mathbb{R}^{N\times T} \to \mathbb{R}^{N\times T}$ denotes the orthogonal projection supported on the complement of Ω.*

On the multivariate time series $\boldsymbol{Y}$, the first impulse is to follow the LCR model in the univariate case and formulate the multivariate model as follows,

$$\begin{aligned} \min_{\boldsymbol{X}} \ & \|\mathcal{C}(\boldsymbol{X})\|_* + \frac{\gamma}{2}\|\boldsymbol{L}\boldsymbol{X}^\top\|_F^2 \\ \text{s.t.} \ & \|\mathcal{P}_\Omega(\boldsymbol{X} - \boldsymbol{Y})\|_F \leq \epsilon, \end{aligned} \tag{41}$$

where we introduce the nuclear norm of the circulant tensor $\mathcal{C}(\boldsymbol{X})$ in Definition 2, which follows the tensor nuclear norm proposed in [31], [32]. The Laplacian matrix $\boldsymbol{L} \in \mathbb{R}^{T\times T}$ is basically given by $\mathcal{C}(\boldsymbol{\ell})$ with the prescribed Laplacian kernel $\boldsymbol{\ell} \in \mathbb{R}^T$ in Definition 1.

Definition 2 (Circulant Tensor Nuclear Norm). *For any matrix $\boldsymbol{X} \in \mathbb{R}^{N\times T}$, the corresponding circulant tensor is $\mathcal{C}(\boldsymbol{X}) \in \mathbb{R}^{N\times N\times T\times T}$, which can be factorized in the Tucker format (i.e., higher-order singular value decomposition) [33]:*

$$\mathcal{C}(\boldsymbol{X}) = \boldsymbol{\mathcal{S}} \times_1 \boldsymbol{U}_1 \times_2 \boldsymbol{U}_1 \times_3 \boldsymbol{U}_2 \times_4 \boldsymbol{U}_2, \tag{42}$$

where $\boldsymbol{\mathcal{S}} \in \mathbb{R}^{N\times N\times T\times T}$ is the core tensor (consisting of singular values [31], [32]), while $\boldsymbol{U}_1 \in \mathbb{R}^{N\times N}$ and $\boldsymbol{U}_2 \in \mathbb{R}^{T\times T}$ are unitary matrices. The notation $\times_k$, $\forall k \in \{1,2,3,4\}$ represents the mode-k product between tensor and matrix [33]. The circulant tensor nuclear norm is defined as

$$\|\mathcal{C}(\boldsymbol{X})\|_* = \sum_{n=1}^{N}\sum_{t=1}^{T} s_{n,n,t,t}, \tag{43}$$

where $s_{n,n,t,t}$ is the (n,n,t,t)-th entry of the core tensor $\boldsymbol{\mathcal{S}}$.

As mentioned in Eq. (41), the Laplacian matrix in the temporal dimension works on each time series independently. If we aim to build the connection between circulant matrix and FFT, then the problem would be reduced to

$$\begin{aligned} \min_{\boldsymbol{X}} \ & \sum_{n=1}^{N} \|\mathcal{C}(\boldsymbol{x}_n)\|_* + \frac{\gamma}{2}\sum_{n=1}^{N}\|\boldsymbol{\ell} \star \boldsymbol{x}_n\|_2^2 \\ \text{s.t.} \ & \|\mathcal{P}_\Omega(\boldsymbol{X}-\boldsymbol{Y})\|_F \le \epsilon, \end{aligned} \tag{44}$$

where the vectors $\boldsymbol{x}_n \in \mathbb{R}^T$, $n = 1,2,\ldots,N$ are the univariate time series or the rows of the data $\boldsymbol{X} \in \mathbb{R}^{N\times T}$, and $\boldsymbol{\ell} \in \mathbb{R}^T$ is the Laplacian kernel (see Definition 1). In this case, the whole problem is divided into N subproblems (defined as LCR$_N$ in the following). The reconstruction of multivariate time series is achieved by reconstructing each time series independently.

To jointly characterize the spatial and temporal dependencies for traffic time series, we consider a separable kernel in the LCR model, namely,

$$\boldsymbol{K} \triangleq \boldsymbol{\ell}_s\boldsymbol{\ell}^\top \in \mathbb{R}^{N\times T}, \tag{45}$$

with the spatial kernel $\boldsymbol{\ell}_s = (1,0,\cdots,0)^\top \in \mathbb{R}^N$ (i.e., the first column of the N-by-N identity matrix). In the case of spatial modeling, $\boldsymbol{\ell}_s$ can also be introduced as the Laplacian kernel in Definition 1. The optimization problem of two-dimensional LCR (LCR-2D) can be formulated as follows,

$$\begin{aligned} \min_{\boldsymbol{X}} \ & \|\mathcal{C}(\boldsymbol{X})\|_* + \frac{\gamma}{2}\|\boldsymbol{K}\star\boldsymbol{X}\|_F^2 \\ \text{s.t.} \ & \|\mathcal{P}_\Omega(\boldsymbol{X}-\boldsymbol{Y})\|_F \le \epsilon, \end{aligned} \tag{46}$$

where the two-dimensional (2D) circular convolution is described in Definition 3. Although the spatial kernel $\boldsymbol{\ell}_s = (1,0,\cdots,0)^\top \in \mathbb{R}^N$ does not provide any spatial dependencies, the nuclear norm of circulant operator on $\boldsymbol{X}$ can achieve implicit spatial modeling. In the meanwhile, the spatiotemporal structure of data can be preserved. If applicable, one can characterize spatial correlations by using the Laplacian kernel.

Definition 3 (2D Circular Convolution [34], [24]). *For any matrices $\boldsymbol{X} \in \mathbb{R}^{N\times T}$ and $\boldsymbol{K} \in \mathbb{R}^{\nu_1\times\nu_2}$ with $\nu_1 \le N, \nu_2 \le T$, the circular convolution of two matrices is defined as*

$$\boldsymbol{Z} = \boldsymbol{K}\star\boldsymbol{X} \in \mathbb{R}^{N\times T}, \tag{47}$$

or element-wise,

$$z_{n,t} = \sum_{i=1}^{\nu_1}\sum_{j=1}^{\nu_2} \kappa_{i,j}x_{n-i+1,t-j+1}, \tag{48}$$

where $n = 1,2,\ldots,N$ and $t = 1,2,\ldots,T$. $\kappa_{i,j}$ is the (i,j)-th entry of $\boldsymbol{K}$. The symbol $\star$ denotes the operator of convolution.

Remark. *In the field of signal processing, the 2D circular convolution also possesses the properties that associated with two-dimensional discrete Fourier transform. According to the convolution theorem and the Parseval's theorem, we have*

$$\|\boldsymbol{K}\star\boldsymbol{X}\|_F^2 = \frac{1}{NT}\|\mathcal{F}(\boldsymbol{K}) \circ \mathcal{F}(\boldsymbol{X})\|_F^2, \tag{49}$$

where $\mathcal{F}(\cdot)$ denotes the 2D discrete Fourier transform. Typically, 2D discrete Fourier transform can be computed by first transforming each column vector (or row vector) and then each row vector (or column vector) of the matrix [24].

As mentioned above, LCR in the multivariate case is also a convex problem that can be resolved by the ADMM framework. Following Eq. (24), the ADMM scheme is given by

$$\begin{cases} \boldsymbol{X} := \arg\min_{\boldsymbol{X}} \mathcal{L}(\boldsymbol{X},\boldsymbol{Z},\boldsymbol{W}), \\ \boldsymbol{Z} := \arg\min_{\boldsymbol{Z}} \mathcal{L}(\boldsymbol{X},\boldsymbol{Z},\boldsymbol{W}) \\ \quad = \frac{1}{\lambda+\eta}\mathcal{P}_\Omega(\lambda\boldsymbol{X}+\boldsymbol{W}+\eta\boldsymbol{Y}) + \frac{1}{\lambda}\mathcal{P}_\Omega^\perp(\lambda\boldsymbol{X}+\boldsymbol{W}), \\ \boldsymbol{W} := \boldsymbol{W} + \lambda(\boldsymbol{X}-\boldsymbol{Z}), \end{cases} \tag{50}$$

where $\mathcal{L}(\boldsymbol{X},\boldsymbol{Z},\boldsymbol{W})$ is the augmented Lagrangian function:

$$\begin{aligned} \mathcal{L}(\boldsymbol{X},\boldsymbol{Z},\boldsymbol{W}) =& \|\mathcal{C}(\boldsymbol{X})\|_* + \frac{\gamma}{2}\|\boldsymbol{K}\star\boldsymbol{X}\|_F^2 \\ &+ \frac{\lambda}{2}\|\boldsymbol{X}-\boldsymbol{Z}\|_F^2 + \langle\boldsymbol{W},\boldsymbol{X}-\boldsymbol{Z}\rangle \\ &+ \frac{\eta}{2}\|\mathcal{P}_\Omega(\boldsymbol{Z}-\boldsymbol{Y})\|_F^2. \end{aligned} \tag{51}$$

With respect to the variable $\boldsymbol{X}$, the subproblem is

$$\begin{aligned} \boldsymbol{X} := \arg\min_{\boldsymbol{X}} \ & \|\mathcal{C}(\boldsymbol{X})\|_* + \frac{\gamma}{2}\|\boldsymbol{K}\star\boldsymbol{X}\|_F^2 \\ & + \frac{\lambda}{2}\|\boldsymbol{X}-\boldsymbol{Z}+\boldsymbol{W}/\lambda\|_2^2. \end{aligned} \tag{52}$$

Although the nuclear norm of circulant tensor in Definition 2 is more complicated than the nuclear norm of circulant matrix as mentioned in Lemma 1, 2D discrete Fourier transform is also applicable for finding the equivalent formula. In the frequency domain, it takes

$$\begin{aligned} \hat{\boldsymbol{X}} := \arg\min_{\hat{\boldsymbol{X}}} \ & \|\hat{\boldsymbol{X}}\|_1 + \frac{\gamma}{2NT}\|\hat{\boldsymbol{K}}\circ\hat{\boldsymbol{X}}\|_F^2 \\ & + \frac{\lambda}{2NT}\|\hat{\boldsymbol{X}}-\hat{\boldsymbol{Z}}+\hat{\boldsymbol{W}}/\lambda\|_F^2, \end{aligned} \tag{53}$$

where we introduce the 2D discrete Fourier transformed $\{\hat{\boldsymbol{K}},\hat{\boldsymbol{X}},\hat{\boldsymbol{Z}},\hat{\boldsymbol{W}}\}$ referring to $\{\boldsymbol{K},\boldsymbol{X},\boldsymbol{Z},\boldsymbol{W}\}$ in the frequency domain. Without loss of generality, this subproblem for ℓ_1-norm minimization in complex space can also be solved by the shrinkage operator in Eq. (37). Algorithm 2 summarizes the whole scheme of LCR-2D.

Algorithm 2 Two-Dimensional Laplacian Convolutional Representation (LCR-2D)

Input: Data $\boldsymbol{Y} \in \mathbb{R}^{N\times T}$ with observed index set Ω, Laplacian kernel size $\tau \in \mathbb{N}^+$, and hyperparameters $\{\gamma, \lambda, \eta\}$.
Output: Reconstructed matrix $\boldsymbol{X} \in \mathbb{R}^{N\times T}$.

```
Initialize {X_0, Z_0, W_0}.
Construct the Laplacian kernel ℓ ∈ R^T with τ.
Construct the spatial kernel ℓ_s = (1, 0, ··· , 0) ∈ R^N (or ℓ_s ∈ N^T with τ) and build up a separable kernel K ≜ ℓ_s ℓ^⊤.
for i = 0 to maximum iteration do
    Perform FFT on {Z_i, W_i}.
    Compute X̂ by referring to the shrinkage in Eq. (37).
    Compute X_{i+1} by X_{i+1} = F^{-1}(X̂).
    Compute Z_{i+1} by Eq. (50).
    Compute W_{i+1} by Eq. (50).
end for
```

5 Univariate Traffic Time Series Imputation

This section evaluates the reconstruction of univariate traffic time series from partial observations. The reconstruction results of the proposed LCR model are compared with several baseline models. Experiments are based on traffic speed time series (weak periodicity and strong noises) and traffic volume time series (strong periodicity). We focus on understanding how well the reconstructed time series preserve the global and local trends of the ground truth time series in the imputation task. The adapted datasets and Python implementation are publicly available at https://github.com/xinychen/transdim.

5.1 Traffic Speed

We begin with some preliminary experiments on univariate freeway traffic speed time series collected through dual-loop detectors in Portland, USA.[1] The speed observations have 15-min time resolution (i.e., 96 expected data samples per day) over three days, and the data vector is of length 288. In particular, we consider the imputation scenarios on fully observed, 20%, 10%, and 5% observed data, respectively. To generate the partially observed time series, we randomly mask a certain number of observations as missing values.

Fig. 3 shows the reconstruction/imputation results produced by our LCR model. Fig. 3(a) demonstrates the reconstructed time series by LCR on the fully observed time series, in which the data noises can be smoothed out due to the existence of Laplacian kernelized temporal regularization and the relaxation of observation constraint $\|\mathcal{P}_\Omega(\boldsymbol{x} - \boldsymbol{y})\|_2 \leq \epsilon$ in which $\boldsymbol{y}$ and $\boldsymbol{x}$ are the partial observations and the reconstructed time series, respectively. Fig. 3(b) and 3(c) show the imputation performance by LCR with partial observations. Of these results, the reconstructed time series can accurately approximate both partial observations and missing values while preserving the trends of the ground truth time series.

Next, we test a more challenging scenario in which we aim to reconstruct 95% missing values from 5% observations (i.e., reconstructing 274 missing values from only 14 data samples). We compare the time series imputation of LCR with the following baseline models:

- CircNNM [6]. This is a nuclear norm minimization of circulant matrix via FFT, which can be regarded as a special case of our LCR model without temporal regularization.
- ConvNNM [6]. This is a standard nuclear norm minimization on the convolution matrix. In particular, we set up a baseline as ConvNNM with Laplacian kernel (i.e., ConvNNM+) whose optimization problem is given by

$$\begin{aligned} \min_{\boldsymbol{x}} \ & \|\mathcal{C}_{\tilde{\tau}}(\boldsymbol{x})\|_* + \frac{\lambda}{2}\|\boldsymbol{\ell} \star \boldsymbol{x}\|_2^2 \\ \text{s.t.} \ & \|\mathcal{P}_\Omega(\boldsymbol{x} - \boldsymbol{y})\|_2 \leq \epsilon, \end{aligned} \tag{54}$$

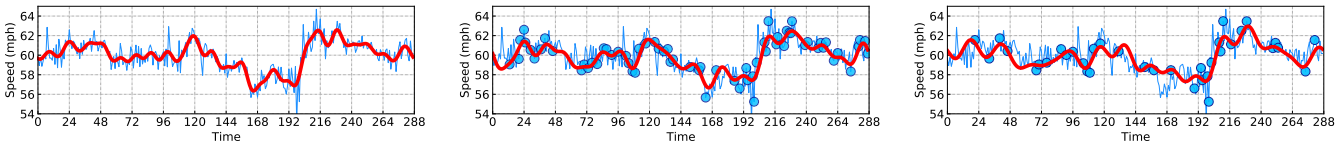
  where $\tilde{\tau} \in \mathbb{N}^+$ is the kernel size of the convolution matrix as discussed in Eq. (4).
- Gaussian process (GP, [35]). The GP regression with a squared exponential kernel. The hyperparameters are optimized via the maximum marginal likelihood method.

Essentially, comparing the proposed LCR model with these baseline models allows one to highlight a) the importance of global and local trends modeling in LCR, and b) the fast implementation of LCR via the use of FFT. The imputation results on the given time series are shown in Fig. 4, we can summarize the following findings for the baseline models:

- The reconstructed time series by CircNNM shows high fluctuations due to the lack of local trend modeling. As a result, the reconstructed time series fails to reproduce trends of the ground truth under such a high missing rate.
- ConvNNM performs better than CircNNM. The reconstructed time series fits the observed values well, but the trend is still far from the ground truth. Unlike CircNNM, ConvNNM cannot employ a fast implementation via FFT. As a result, ConvNNM is not scalable to large problems.
- ConvNNM+ outperforms ConvNNM. This demonstrates the significance of Laplacian kernel for time series modeling. However, both ConvNNM and ConvNNM+ require to implement the singular value thresholding on T-by-$\tilde{\tau}$ convolution matrices, consuming $\mathcal{O}(\tilde{\tau}T)$ space.
- The GP model reasonably reconstructs the traffic speed time series, but is computationally costly. A common fact is that GP models are not well-suited to long time series and large problems due to the time complexity $\mathcal{O}(T^3)$.

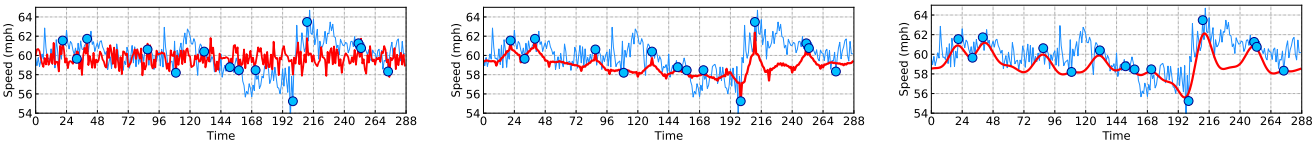
As shown in Fig. 4(e), the reconstructed time series by LCR demonstrates consistent global and local trends with the trends of ground truth time series, and our LCR model clearly outperforms the baseline models in terms of either accuracy or computational efficiency. By comparing CircNNM with LCR, the imputation results emphasize the importance of temporal regularization with Laplacian kernel. Notably, when producing the reconstructed time series $\boldsymbol{x}$, both CircNNM and LCR consume $\mathcal{O}(T)$ space, however, ConvNNM and ConvNNM+ take $\mathcal{O}(\tilde{\tau}T)$ space. In terms

1. https://portal.its.pdx.edu/

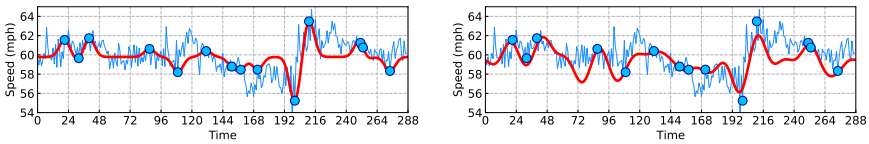


(a) Fully observed. (b) 80% missing values. (c) 90% missing values.

Fig. 3: Univariate traffic time series imputation on the freeway traffic speed time series. The blue curve represents the ground truth time series, while the red curve refers to the reconstructed time series produced by LCR. Here, partial observations are illustrated as blue circles.



(a) CircNNM (red curve). (b) ConvNNM (red curve). (c) ConvNNM+ (red curve).



(d) GP (red curve). (e) LCR (red curve) with $\tau = 2, \gamma = 2\lambda$.

Fig. 4: Univariate traffic time series imputation on the freeway traffic speed time series. In this case, we mask 95% observations as missing values and only have 14 speed observations for training the model.

of temporal modeling, both LCR and GP can produce accurate reconstruction. Nevertheless, the proposed temporal regularization via Laplacian kernel presents an FFT solution in $\mathcal{O}(T \log T)$ time, which is much more computationally efficient than GP with $\mathcal{O}(T^3)$ time.

5.2 Traffic Volume

In addition to the traffic speed time series, we evaluate the model on the freeway traffic volume time series collected through loop detectors in Portland, USA. The volume observations are with 15-min time resolution over three days, and the data vector is of length 288. The time series is characterized by a strong daily rhythm (see Fig. 5). The time series shows relatively low traffic volume at night, and the traffic volume reaches a peak during rush hours, demonstrating typical traveler's behavioral rhythms. As shown in Fig. 5, the time series possesses strong seasonality and three remarkable peaks over three days.

The task is reconstructing 95% missing values from 5% observations. As can be seen, due to the lack of explicit temporal modeling, both CircNNM and ConvNNM models cannot produce smooth time series as LCR. Observing the reconstructed time series, it is clear that LCR produces much more accurate reconstruction results than both CircNNM and ConvNNM. Yet, on the contrary to the traffic speed time series, due to the strong seasonality (i.e., daily rhythm in traffic flow) in the traffic volume time series, both CircNNM and ConvNNM can produce reasonable time series, which seems to be consistent with the results in [7].

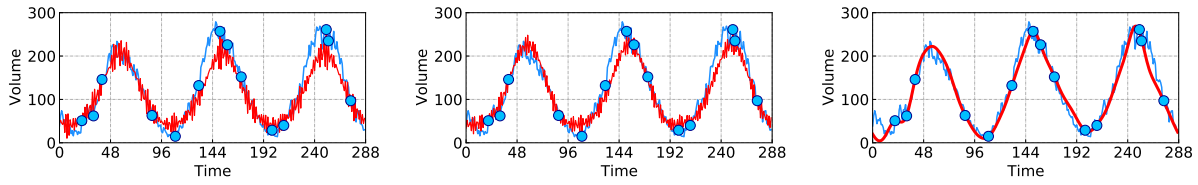
6 Multivariate Traffic Time Series Imputation

In this section, we apply the LCR model to two real-world multivariate traffic time series imputation scenarios. One is the speed field reconstruction which shows strong spatiotemporal correlations, while another is the large-scale highway traffic speed data imputation. Both two scenarios show practical implications in real-world application and these experiments in this study are expected to demonstrate the effectiveness of LCR for spatiotemporal data modeling on traffic time series.

6.1 NGSIM Speed Field Reconstruction

Speed field reconstruction is a critical problem in road traffic flow modeling as the data collection could often be far from ideal [36]. The task is to reconstruct the traffic speed distribution along a road segment during a period using the motion states (i.e., time, location, and speed) of a small portion of vehicles [37], e.g., to estimate the traffic congestion by using only the motion states of taxis. We use the NGSIM data[2] for the experiment. To evaluate the speed field reconstruction performances, we use the mean absolute

2. U.S. Department of Transportation Federal Highway Administration. (2016). Next Generation Simulation (NGSIM) Vehicle Trajectories and Supporting Data. [Dataset]. Provided by ITS DataHub through Data.transportation.gov. Accessed 2023-01-01 from http://doi.org/10.21949/1504477.



(a) CircNNM. (b) ConvNNM. (c) LCR with $\tau = 2, \gamma = 2\lambda$.

Fig. 5: Univariate time series imputation on the freeway traffic volume time series. In this case, we randomly remove 95% observations as missing values, and we only have 14 volume observations for the reconstruction.

percentage error (MAPE) and the root mean square error (RMSE):

$$\text{MAPE} = \frac{1}{n}\sum_{i=1}^{n}\frac{|y_i - \hat{y}_i|}{y_i} \times 100, \text{ RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2},$$

where n is the total number of estimated values, and y_i and $\hat{y}_i$ are the actual value and its estimation, respectively.

For comparison, we choose the following baseline models to highlight the importance of Laplacian kernels, despite the circulant tensor nuclear norm. Accordingly, we build up the following baseline models:

- Quadratic variation completion (QVC), which is given by
$$\begin{aligned} \min_{\boldsymbol{X}} \ & \frac{\gamma}{2}\|(\boldsymbol{\ell}_s\boldsymbol{\ell}^\top) \star \boldsymbol{X}\|_F^2 \\ \text{s.t.} \ & \|\mathcal{P}_\Omega(\boldsymbol{X} - \boldsymbol{Y})\|_F \leq \epsilon, \end{aligned} \tag{55}$$
with the spatial kernel $\boldsymbol{\ell}_s = (1, 0, \cdots, 0, -1)^\top \in \mathbb{R}^N$ and the temporal kernel $\boldsymbol{\ell} = (1, 0, \cdots, 0, -1)^\top \in \mathbb{R}^T$. If we let both two kernels be the Laplacian kernels parameterized by τ in Definition 1, then this Laplacian kernelized completion (LKC) model is a special case of LCR-2D without circulant nuclear norm minimization.
- Circulant tensor nuclear norm minimization (CTNNM), whose objective function is specified as $\|\mathcal{C}(\boldsymbol{X})\|_*$.

Fig. 6(a) is the ground-truth speed field, which is in the form of matrix (of size 200×500). The speed field is with the 3-meter spatial resolution (i.e., $N = 200$) and the 5-second time resolution (i.e., $T = 500$). Fig. 6(b) is the speed field from the trajectories of 20% of the vehicles chosen randomly, showing to be rather sparse. We evaluate our model on Fig. 6(b) and reconstruct sparse speed field. Through testing the model, the hyperparameters of LCR are set as $\lambda = 1 \times 10^4 NT$, $\eta = 10^2\lambda$, $\gamma = 5\lambda$, and $\tau = 2$. For the baseline models, the hyperparameters of QVC and LKC are set as $\lambda = 1 \times 10^4 NT$, $\eta = 10^2\lambda$, and $\gamma = 5\lambda$, while the hyperparameters of CTNNM are set as $\lambda = 1 \times 10^4 NT$ and $\eta = 10^2\lambda$. As shown in Fig. 6, we can summarize the following findings:

- Recall that QVC is a special case of LKC, we empirically demonstrate that LKC outperforms QVC (see Fig. 6(c) and 6(d)). This shows the modeling flexibility of Laplacian kernels over the QV regularization.
- LKC cannot produce accurate reconstruction of speed field as the proposed LCR-2D model (see Fig. 6(d) and 6(f)), and this implies the importance of circulant tensor nuclear norm in LCR-2D.
- Recall that CTNNM is a special case of LCR-2D without Laplacian kernels, we empirically demonstrate that LCR-2D performs better than CTNNM (see Fig. 6(e) and 6(f)), showing the importance of Laplacian kernels in LCR-2D.

As mentioned above, both circulant tensor nuclear norm and Laplacian kernels in the objective function of LCR-2D are of great significance. LCR-2D can achieve satisfactory results for speed field reconstruction because of the spatiotemporal correlation of traffic wave [38] characterized by both global and local trends in LCR-2D.

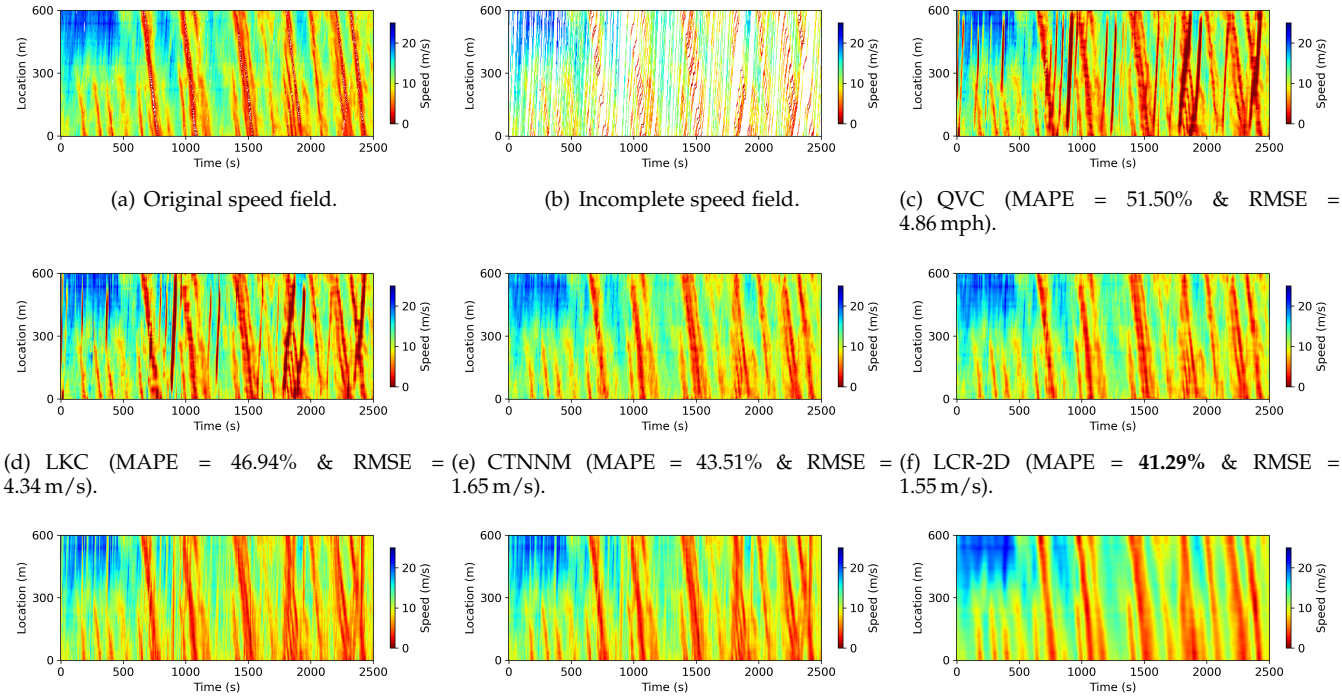
In addition, we also evaluate a sequence of matrix/tensor factorization models, e.g., matrix factorization (MF), smoothing matrix factorization (SMF), and Hankel tensor factorization (HTF, a special case of MDT-Tucker [4]). To achieve spatiotemporal modeling in SMF and HTF, we take into account smoothing/Hankelization over spatial and temporal dimensions simultaneously. Comparing Fig. 6(g) with Fig. 6(h), SMF with spatial/temporal smoothing performs better than the pure MF model. In the meanwhile, comparing Fig. 6(h) with Fig. 6(i), HTF shows better reconstruction than SMF due to the use of spatial/temporal Hankelization. Notably, HTF produces very competitive reconstruction as LCR-2D, but HTF involves high memory consumption if one builds a Hankel tensor [4].

6.2 PeMS Traffic Speed Imputation

In what follows, we study the generalization of LCR to high-dimensional data and evaluate the model on a large-scale traffic flow dataset. The data are collected by the California department of transportation through their Performance Measurement System (PeMS) [39]. This dataset contains freeway traffic speed collected from 11,160 traffic measurement sensors over 4 weeks (the first 4 weeks in the year of 2018) with a 5-minute time resolution (288 time intervals per day) in California, USA.[3] It can be arranged in a matrix of size 11160×8064. Note that this dataset contains about 90 million observations; many existing methods, such as GP and MDT-Tucker, are not well-suited to such a large-scale dataset.

To set up the imputation task, we randomly mask 30%, 50%, 70%, and 90% traffic speed observations as missing values, referred to as 30%, 50%, 70%, and 90% missing

3. The dataset is available at https://doi.org/10.5281/zenodo.3939792.



(a) Original speed field. (b) Incomplete speed field. (c) QVC (MAPE = 51.50% & RMSE = 4.86 mph).

(d) LKC (MAPE = 46.94% & RMSE = 4.34 m/s). (e) CTNNM (MAPE = 43.51% & RMSE = 1.65 m/s). (f) LCR-2D (MAPE = **41.29%** & RMSE = 1.55 m/s).

(g) MF (MAPE = 45.84% & RMSE = 2.80 m/s). (h) SMF (MAPE = 44.06% & RMSE = 2.16 m/s). (i) HTF (MAPE = 41.58% & RMSE = **1.39** m/s).

Fig. 6: Speed field reconstruction of road traffic flow with LCR-2D and some baseline models on the 80% masked trajectories. The smaller the MAPE and RMSE, the better the reconstruction quality.

rates, respectively. To assess the imputation performance, we use the actual values of the masked missing entries as the ground truth to compute MAPE and RMSE. For comparison, we choose the baseline models as CircNNM [6], LRMC [3], nonstationary temporal matrix factorization (NoTMF [12]) and high accuracy low-rank tensor completion (HaLRTC [40], [41]). Through cross validation, we have the following settings:

- LCR-2D: We set the hyperparameters as $\lambda = 1 \times 10^{-5}NT$, $\eta = 10^2\lambda$, and $\gamma = 5\lambda$.
- LCR$_N$: We set the hyperparameters as $\lambda = 5\times 10^{-3}T$ and $\eta = 10^3\lambda$; since LCR is accomplished over N subproblems independently, we set $\gamma = 5\lambda$ at all missing rates for seeking better local trends.
- LCR: We consider the LCR via the vectorization on the multivariate time series [7], which is given by

$$\begin{aligned} \min_{\boldsymbol{X}} \ & \|\mathcal{C}(\mathrm{vec}(\boldsymbol{X}^\top))\|_* + \frac{\gamma}{2}\|\boldsymbol{\ell} \star \mathrm{vec}(\boldsymbol{X}^\top)\|_2^2 \\ \text{s.t.} \ & \|\mathcal{P}_\Omega(\boldsymbol{X} - \boldsymbol{Y})\|_F \leq \epsilon, \end{aligned} \tag{56}$$

  where $\mathrm{vec}(\cdot)$ denotes the operator of vectorization. The Laplacian kernel $\boldsymbol{\ell}$ is of length NT. We set the hyperparameters as $\lambda = 5\times 10^{-6}NT$, $\eta = 10^2\lambda$, and $\gamma = 5\lambda$.
- CircNNM: We follow the nuclear norm minimization of circulant matrix via the vectorization [6] and set the hyperparameters as $\lambda = 5 \times 10^{-7}NT$ and $\eta = 10^2\lambda$.

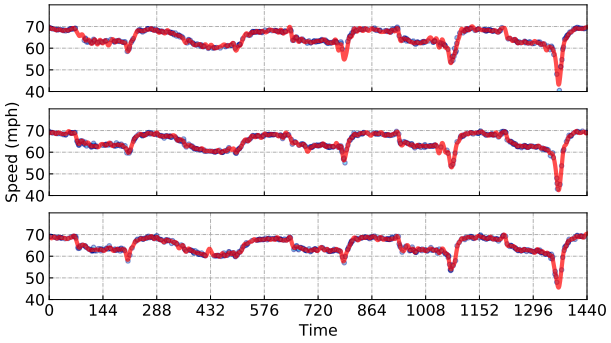
Notably, to assess the importance of local temporal modeling thoroughly, we evaluate the LCR models with different Laplacian kernels parameterized by $\tau = 1, 2, 3$. From Table 1, in the case of a relatively small missing rate (e.g., 30%), LCR-2D, LCR$_N$, and LCR perform best with the kernel size $\tau = 1$. However, when the missing rate reaches 90%, setting the kernel size as $\tau = 2, 3$ in the LCR models can produce more competitive results. These demonstrate the importance of a proper Laplacian kernel for characterizing the local trends in time series. When learning LCR from limited observations with a relatively high missing rate, local trend modeling in LCR demands more informative Laplacian kernels.

Among the baseline models in Table 1, CircNNM can be implemented by FFT, which is a special case of our LCR model without the temporal regularization. By comparing the performance of CircNNM and LCR, we can find the significant improvement of imputation achieved by LCR over CircNNM, mainly due to the existence of the temporal regularization. Moreover, our LCR models significantly outperform LRMC, NoTMF, and HaLRTC. Here, LRMC provides a well-suited framework for reconstructing incomplete matrix, however, it involves high time complexity in the process of singular value thresholding, making it extremely costly to work on the large-scale problems. NoTMF jointly models global and local trends by a unified and efficient temporal matrix factorization framework, but as can be seen, it is inferior to the LCR models.

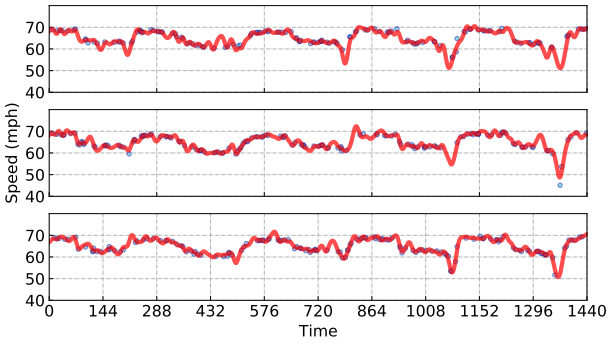
Notably, since this work takes the same experiment settings on the PeMS-4W dataset as [16], our LCR models also perform better than the state-of-the-art low-rank tensor completion models. Beyond the matrix/tensor completion models that employ the costly process of singular value thresholding, our LCR models convert the nuclear norm

TABLE 1: Imputation performance (MAPE/RMSE) on the PeMS-4W traffic speed dataset. Note that the best results are highlighted in bold fonts.

Missing rate	30%	50%	70%	90%
LCR-2D ($\tau = 1$)	1.57/1.54	1.81/1.74	2.29/2.12	3.80/3.17
LCR-2D ($\tau = 2$)	1.74/1.70	1.89/1.84	2.21/2.15	3.40/3.07
LCR-2D ($\tau = 3$)	2.00/1.94	2.12/2.06	2.37/2.31	3.32/3.13
LCR$_N$ ($\tau = 1$)	**1.51**/**1.53**	**1.73**/1.76	2.18/2.19	3.77/3.49
LCR$_N$ ($\tau = 2$)	1.70/1.69	1.84/1.85	**2.15**/2.19	3.32/3.28
LCR$_N$ ($\tau = 3$)	1.99/1.96	2.11/2.09	2.35/2.35	**3.30**/3.32
LCR ($\tau = 1$)	1.60/1.55	1.82/**1.73**	2.29/**2.10**	3.86/3.18
LCR ($\tau = 2$)	1.83/1.77	1.97/1.90	2.28/2.17	3.47/**3.05**
LCR ($\tau = 3$)	2.11/2.03	2.23/2.14	2.47/2.37	3.41/3.11
CircNNM	2.41/2.03	2.80/2.33	3.48/2.82	5.22/4.03
LRMC	2.04/1.80	2.43/2.12	3.08/2.66	6.05/4.43
HaLRTC	1.98/1.73	2.22/1.98	2.84/2.49	4.39/3.66
NoTMF	2.95/2.65	3.05/2.73	3.33/2.97	5.22/4.71



(a) 70% missing rate.



(b) 90% missing rate.

Fig. 7: Multivariate traffic time series imputation by LCR on the PeMS-4W dataset. We visualize the traffic speed time series of the first three road segments during the first five days. Note that blue circles and red curves indicate the partial observations and the reconstructed time series, respectively.

minimization as an ℓ_1-norm minimization in the frequency domain. It seems that LCR is a well-suited framework for both univariate and multivariate time series imputation of various data scales.

Fig. 7 shows the speed imputation results of three road segments under 70% and 90% missing rates. Despite the high missing rates, we can see that the reconstructed time series curves achieved by LCR preserve the overall trend of the original speed time series. Notably, the reconstructed curves even correctly depict the sharp decrease of the speed during congestion periods, demonstrating the effectiveness of LCR in capturing local information of the traffic speed time series.

7 Conclusion

In this study, we focus on reconstructing spatiotemporal traffic data from partial observations. To model the local trends in traffic time series, we introduce a Laplacian kernel for temporal regularization in the form of circular convolution. Following that definition, we propose an LCR model that integrates the temporal regularization into a circulant-matrix-based low-rank model for characterizing both global and local trends in traffic time series, bridging the gap between low-rank models and graph Laplacian models. When developing the solution algorithm, we borrow the properties of circulant matrix and circular convolution, and prove that the proposed LCR model has a fast implementation through FFT. To be specific, the nuclear norm minimization with Laplacian kernelized temporal regularization can be converted into an ℓ_1-norm minimization in complex space. Beyond univariate time series imputation, LCR can be easily adapted to multivariate and even multidimensional time series imputation tasks.

In the numerical experiments, we conduct both univariate and multivariate time series imputation tasks on several real-world traffic datasets. For the fully observed data, LCR can produce denoised and smooth time series. For the sparse and noisy traffic data, LCR can accurately reconstruct traffic time series with the elimination of data noises, demonstrating the importance of local trend modeling via the temporal regularization. On the speed field reconstruction task, the results demonstrate the importance of spatiotemporal modeling with Laplacian kernels. On the large-scale dataset, LCR outperforms the baseline models and demonstrates strong generalization to high-dimensional problems due to the efficient implementation and relatively low time complexity.

Acknowledgment

Xinyu Chen would like to thank the Institute for Data Valorisation (IVADO) and the Interuniversity Research Centre on Enterprise Networks, Logistics and Transportation (CIRRELT) for providing the PhD Excellence Scholarship to support this study.

References

[1] L. Li, X. Su, Y. Zhang, Y. Lin, and Z. Li, "Trend modeling for traffic time series analysis: An integrated study," *IEEE Transactions on Intelligent Transportation Systems*, vol. 16, no. 6, pp. 3430–3439, 2015.

[2] E. J. Candes and Y. Plan, "Matrix completion with noise," *Proceedings of the IEEE*, vol. 98, no. 6, pp. 925–936, 2010.

[3] J.-F. Cai, E. J. Candès, and Z. Shen, "A singular value thresholding algorithm for matrix completion," *SIAM Journal on optimization*, vol. 20, no. 4, pp. 1956–1982, 2010.

[4] T. Yokota, B. Erem, S. Guler, S. K. Warfield, and H. Hontani, "Missing slice recovery for tensors using a low-rank model in embedded space," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 8251–8259.

[5] R. Yamamoto, H. Hontani, A. Imakura, and T. Yokota, "Fast algorithm for low-rank tensor completion in delay-embedded space," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 2058–2066.

[6] G. Liu and W. Zhang, "Recovery of future data via convolution nuclear norm minimization," *IEEE Transactions on Information Theory*, 2022.

[7] G. Liu, "Time series forecasting via learning convolutionally low-rank models," *IEEE Transactions on Information Theory*, vol. 68, no. 5, pp. 3362–3380, 2022.

[8] D. Cai, X. He, J. Han, and T. S. Huang, "Graph regularized nonnegative matrix factorization for data representation," *IEEE transactions on pattern analysis and machine intelligence*, vol. 33, no. 8, pp. 1548–1560, 2010.

[9] X. Mao, K. Qiu, T. Li, and Y. Gu, "Spatio-temporal signal recovery based on low rank and differential smoothness," *IEEE Transactions on Signal Processing*, vol. 66, no. 23, pp. 6281–6296, 2018.

[10] L. Xiong, X. Chen, T.-K. Huang, J. Schneider, and J. G. Carbonell, "Temporal collaborative filtering with bayesian probabilistic tensor factorization," in *Proceedings of the 2010 SIAM international conference on data mining*. SIAM, 2010, pp. 211–222.

[11] H.-F. Yu, N. Rao, and I. S. Dhillon, "Temporal regularized matrix factorization for high-dimensional time series prediction," in *Advances in Neural Information Processing Systems*, 2016, pp. 847–855.

[12] X. Chen, C. Zhang, X.-L. Zhao, N. Saunier, and L. Sun, "Nonstationary temporal matrix factorization for multivariate time series forecasting," *arXiv preprint arXiv:2203.10651*, 2022.

[13] F. Sedighin, A. Cichocki, T. Yokota, and Q. Shi, "Matrix and tensor completion in multiway delay embedded space using tensor train, with application to signal reconstruction," *IEEE Signal Processing Letters*, vol. 27, pp. 810–814, 2020.

[14] Z. Chen and A. Cichocki, "Nonnegative matrix factorization with temporal smoothness and/or spatial decorrelation constraints," *Laboratory for Advanced Brain Signal Processing, RIKEN, Tech. Rep*, vol. 68, 2005.

[15] Y. Wang, Y. Zhang, X. Piao, H. Liu, and K. Zhang, "Traffic data reconstruction via adaptive spatial-temporal correlations," *IEEE Transactions on Intelligent Transportation Systems*, vol. 20, no. 4, pp. 1531–1543, 2018.

[16] X. Chen, Y. Chen, N. Saunier, and L. Sun, "Scalable low-rank tensor learning for spatiotemporal traffic data imputation," *Transportation research part C: emerging technologies*, vol. 129, p. 103226, 2021.

[17] X. Chen, M. Lei, N. Saunier, and L. Sun, "Low-rank autoregressive tensor completion for spatiotemporal traffic data imputation," *IEEE Transactions on Intelligent Transportation Systems*, 2021.

[18] X. Chen and L. Sun, "Bayesian temporal factorization for multidimensional time series prediction," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 44, no. 9, pp. 4659–4673, 2021.

[19] J. Luttinen and A. Ilin, "Variational gaussian-process factor analysis for modeling spatio-temporal data," *Advances in neural information processing systems*, vol. 22, 2009.

[20] T. Zhou, H. Shan, A. Banerjee, and G. Sapiro, "Kernelized probabilistic matrix factorization: Exploiting graphs and side information," in *Proceedings of the 2012 SIAM international Conference on Data mining*. SIAM, 2012, pp. 403–414.

[21] M. Lei, A. Labbe, Y. Wu, and L. Sun, "Bayesian kernelized matrix factorization for spatiotemporal traffic data imputation and kriging," *IEEE Transactions on Intelligent Transportation Systems*, 2022.

[22] A. J. Smola and R. Kondor, "Kernels and regularization on graphs," in *Learning theory and kernel machines*. Springer, 2003, pp. 144–158.

[23] N. Rao, H.-F. Yu, P. K. Ravikumar, and I. S. Dhillon, "Collaborative filtering with graph information: Consistency and scalable methods," *Advances in neural information processing systems*, vol. 28, 2015.

[24] P. C. Hansen, J. G. Nagy, and D. P. O'leary, *Deblurring images: matrices, spectra, and filtering*. SIAM, 2006.

[25] J. Wright and Y. Ma, *High-dimensional data analysis with low-dimensional models: Principles, computation, and applications*. Cambridge University Press, 2022.

[26] S. L. Brunton and J. N. Kutz, *Data-driven science and engineering: Machine learning, dynamical systems, and control*. Cambridge University Press, 2022.

[27] E. Candes and B. Recht, "Exact matrix completion via convex optimization," *Communications of the ACM*, vol. 55, no. 6, pp. 111–119, 2012.

[28] A. Sandryhaila and J. M. Moura, "Discrete signal processing on graphs: Graph fourier transform," in *2013 IEEE International Conference on Acoustics, Speech and Signal Processing*. IEEE, 2013, pp. 6167–6170.

[29] J. Yang, W. Yin, Y. Zhang, and Y. Wang, "A fast algorithm for edge-preserving variational multichannel image restoration," *SIAM Journal on Imaging Sciences*, vol. 2, no. 2, pp. 569–592, 2009.

[30] G. Liu, Z. Lin, S. Yan, J. Sun, Y. Yu, and Y. Ma, "Robust recovery of subspace structures by low-rank representation," *IEEE transactions on pattern analysis and machine intelligence*, vol. 35, no. 1, pp. 171–184, 2012.

[31] O. Semerci, N. Hao, M. E. Kilmer, and E. L. Miller, "Tensor-based formulation and nuclear norm regularization for multienergy computed tomography," *IEEE Transactions on Image Processing*, vol. 23, no. 4, pp. 1678–1693, 2014.

[32] C. Lu, J. Feng, Y. Chen, W. Liu, Z. Lin, and S. Yan, "Tensor robust principal component analysis with a new tensor nuclear norm," *IEEE transactions on pattern analysis and machine intelligence*, vol. 42, no. 4, pp. 925–938, 2019.

[33] T. G. Kolda and B. W. Bader, "Tensor decompositions and applications," *SIAM Review*, vol. 51, no. 3, pp. 455–500, 2009.

[34] E. O. Brigham, *The fast Fourier transform and its applications*. Prentice-Hall, Inc., 1988.

[35] C. K. Williams and C. E. Rasmussen, *Gaussian processes for machine learning*. MIT press Cambridge, MA, 2006, vol. 2, no. 3.

[36] M. Treiber and A. Kesting, "Traffic flow dynamics," *Traffic Flow Dynamics: Data, Models and Simulation, Springer-Verlag Berlin Heidelberg*, pp. 983–1000, 2013.

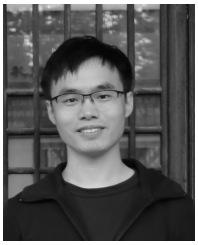
[37] C. De Fabritiis, R. Ragona, and G. Valenti, "Traffic estimation and prediction based on real time floating car data," in *2008 11th international IEEE conference on intelligent transportation systems*. IEEE, 2008, pp. 197–203.

[38] X. Wang, Y. Wu, D. Zhuang, and L. Sun, "Low-rank hankel tensor completion for traffic speed estimation," *arXiv preprint arXiv:2105.11335*, 2021.

[39] C. Chen, K. Petty, A. Skabardonis, P. Varaiya, and Z. Jia, "Freeway performance measurement system: mining loop detector data," *Transportation Research Record*, vol. 1748, no. 1, pp. 96–102, 2001.

[40] J. Liu, P. Musialski, P. Wonka, and J. Ye, "Tensor completion for estimating missing values in visual data," *IEEE transactions on pattern analysis and machine intelligence*, vol. 35, no. 1, pp. 208–220, 2012.

[41] B. Ran, H. Tan, Y. Wu, and P. J. Jin, "Tensor based missing traffic data completion with spatial–temporal correlation," *Physica A: Statistical Mechanics and its Applications*, vol. 446, pp. 54–63, 2016.



Xinyu Chen (Student Member, IEEE) received the B.S. degree in Traffic Engineering from Guangzhou University, Guangzhou, China, in 2016, and M.S. degree in Transportation Information Engineering & Control from Sun Yat-Sen University, Guangzhou, China, in 2019. He is currently a Ph.D. student with the Civil, Geological and Mining Engineering Department, Polytechnique Montreal, Montreal, QC, Canada. His current research centers on machine learning, spatiotemporal data modeling, and intelligent transportation systems.

Zhanhong Cheng received his Ph.D. degree from McGill University. He received his B.S. and M.S. degrees from Harbin Institute of Technology, Harbin, China. He is now a Postdoc researcher in the Department of Civil Engineering at McGill University, Montreal, QC, Canada. His research interests include public transportation, travel behavior modeling, spatiotemporal forecasting, and machine learning in transportation.

Nicolas Saunier received an engineering degree and a Doctorate (Ph.D.) in computer science from Telecom ParisTech, Paris, France, respectively in 2001 and 2005. He is currently a Full Professor with the Civil, Geological and Mining Engineering Department at Polytechnique Montreal, Montreal, QC, Canada. His research interests include intelligent transportation, road safety, and data science for transportation.

Lijun Sun (Senior Member, IEEE) received the B.S. degree in civil engineering from Tsinghua University, Beijing, China, in 2011, and the Ph.D. degree in civil engineering (transportation) from the National the University of Singapore in 2015. He is currently an Assistant Professor in the Department of Civil Engineering, McGill University, Montreal, QC, Canada. His research centers on intelligent transportation systems, machine learning, spatiotemporal modeling, travel behavior, and agent-based simulation.