TRR
JOURNAL OF THE TRANSPORTATION RESEARCH BOARD

Research Article

Transportation Research Record
1–11

DOI: 10.1177/03611981221089298
journals.sagepub.com/home/trr
SAGE

A Method for Granular Traffic Data Imputation Based on PARATUCK2

Mina Nouri[1], Mostafa Reisi-Gahrooei[2], and Mohammad Ilbeigi[1]

Abstract

Imputing missing data is a critical task in data-driven intelligent transportation systems. During recent decades there has been a considerable investment in developing various types of sensors and smart systems, including stationary devices (e.g., loop detectors) and floating vehicles equipped with global positioning system (GPS) trackers to collect large-scale traffic data. However, collected data may not include observations from all road segments in a traffic network for different reasons, including sensor failure, transmission error, and because GPS-equipped vehicles may not always travel through all road segments. The first step toward developing real-time traffic monitoring and disruption prediction models is to estimate missing values through a systematic data imputation process. Many of the existing data imputation methods are based on matrix completion techniques that utilize the inherent spatiotemporal characteristics of traffic data. However, these methods may not fully capture the clustered structure of the data. This paper addresses this issue by developing a novel data imputation method using PARATUCK2 decomposition. The proposed method captures both spatial and temporal information of traffic data and constructs a low-dimensional and clustered representation of traffic patterns. The identified spatiotemporal clusters are used to recover network traffic profiles and estimate missing values. The proposed method is implemented using traffic data in the road network of Manhattan in New York City. The performance of the proposed method is evaluated in comparison with two state-of-the-art benchmark methods. The outcomes indicate that the proposed method outperforms the existing state-of-the-art imputation methods in complex and large-scale traffic networks.

Keywords

data and data science, data analytics, statistical methods, modeling, operations, highway traffic monitoring, regional transportation systems management and operations, traffic operations

In light of recent advances in intelligent transportation systems, a wide range of technology-enhanced traffic data collection methods have been integrated into traffic monitoring and disruption detection solutions. Large quantities of traffic data are collected every day on a continuous basis through stationary devices, including loop detectors and traffic cameras and, floating vehicles equipped with global positioning system (GPS) trackers (*1*). A major and inevitable challenge in analyzing the collected traffic data is missing values. Collected data may not include observations from all road segments at every sampling time for a variety of reasons, including sensor failure, transmission error, and because GPS-equipped vehicles may not always travel through all road segments. Missing traffic data may significantly affect the accuracy and reliability of traffic monitoring and forecasting models (*2*). Therefore, many studies in recent years have aimed to address this issue by developing data imputation methods.

Li et al. (*3*) categorized the existing traffic data imputation methods into three groups: (i) prediction-based, (ii) interpolation-based, and (iii) statistical learning-based methods. Prediction-based methods employ predictive models, including time series (*4*, *5*), Bayesian networks (*6*, *7*), feedforward neural networks (*8*, *9*), and support vector regression (*10*, *11*), to estimate missing values based on the relationship between past and future data

[1]Department of Civil, Environmental, and Ocean Engineering (CEOE), Stevens Institute of Technology, Hoboken, NJ
[2]Department of Industrial and Systems Engineering (ISE), University of Florida, Gainesville, FL

Corresponding Author:
Mohammad Ilbeigi, milbeigi@stevens.edu

points extracted from historical data. Interpolation-based methods perform data imputation using the information obtained from available neighboring data. Historical average method and k-nearest neighbors (KNN) model are two popular interpolation-based methods. The historical average method is based on temporal-neighboring analysis (*12*, *13*). This approach estimates missing values using the average of available historical data collected from the same location at the same time of neighboring days. The KNN model, on the other hand, performs pattern-similarity analysis (*14*) and completes missing values using the average (or weighted average) of data obtained from neighboring points. Statistical learning-based methods, such as probabilistic principal component analysis (PPCA) and Markov Chain Monte Carlo imputation methods, estimate missing values by inferring statistical characteristics of traffic data from observed values (*15*, *16*). These methods assume a probability distribution over the collected data (including missing values). The missing values and consequently the distribution parameters are then estimated by using the expectation maximization algorithm in an iterative process.

Li et al. (*3*) empirically examined some of the most common methods from these three categories and concluded that prediction and interpolation methods cannot capture variations in daily traffic flow and may not perform well compared with the statistical learning-based methods. Prediction-based models are most suitable for short-term data prediction purposes, and they may not provide satisfactory results in situations where long-term predictions are necessary because of the persistence of a long chain of missing values. In addition, they do not consider the data observed after the missing value, which makes them less effective in data completion applications (*17*). Interpolation-based methods rely on the assumption that neighboring data points are similar to each other. This assumption is not always valid, especially when we are dealing with missing values in large and interconnected networks. The imputation performance of these methods will also reduce severely when neighboring traffic data is missing as well (*18*).

In addition to the three categories of data imputation methods identified by Li et al. (*3*), recently, decomposition-based techniques, such as low-rank matrix/tensor completion methods, have been used for traffic data imputation (*19*, *20*). These techniques use a multiway array to model the traffic data and attempt to impute missing values by reconstructing data profiles from a low-dimensional representation of the collected data. This low-dimensional representation is created by minimizing the rank of the multiway array (*21*) or by imposing a predefined decomposition form. (*22*, *23*). The decomposition-based methods can capture the spatiotemporal correlation of traffic data and use this information to recover missing values. Using this feature, these methods have the potential to outperform the foregoing conventional data imputation methods.

A popular decomposition-based approach is to represent a matrix $X \in \mathbb{R}^{m \times n}$ in a bilinear form (*23*), that is, $X = UV^T$, where $U \in \mathbb{R}^{m \times k}$, $V \in \mathbb{R}^{n \times k}$, and k is the rank of the matrix. The goal is to find factor matrices U and V so that the Frobenius norm of $X - UV^T$ is minimized at the non-missing values of X. Even though this approach is very popular for its simplicity, it requires both factor matrices to have the same rank. Therefore, it cannot be used when the data contains complex clustering structures along both rows and columns of the matrix.

Another data imputation method based on matrix decomposition is the singular value decomposition (SVD) of a matrix (*24*). This approach is tightly related to the principal component analysis. SVD decomposes a matrix as $X = U\Sigma V^T$, where U and V are square orthogonal matrices and Σ is a diagonal matrix. Although this approach is more flexible than bilinear clustering, like bilinear form, it cannot capture the complex clustering structures because of the consistency required in the dimensions of the factor matrices.

Another group of methods focuses on completing multi-dimensional arrays (tensors). These methods use tensor decomposition techniques (e.g., Tucker and canonical polyadic decomposition [CPD]) (*25*, *26*) or low-rank penalties (e.g., nuclear norm penalty) (*27*, *28*) to project the tensor on a lower dimensional space where the missing values are imputed. Tensor decomposition techniques based on Bayesian inference were introduced to address the data sparsity issue in these data completion problems (*29*). These approaches are considered the state-of-the-art in data imputation. However, they cannot fully capture the spatiotemporal clustered patterns as they use the global information for data completion.

Another type of decomposition-based data imputation method is the decomposition into directional component (DEDICOM) that can capture the asymmetric relationship between several objects, for example, nodes in a transportation network (*30*). This approach decomposes a matrix $X \in \mathbb{R}^{n \times n}$ as ARA^T, where $A \in \mathbb{R}^{n \times k}$, $R \in \mathbb{R}^{k \times k}$, and k is the number of similarity groups. Even though the main application of this method is not for matrix imputation, it can be extended to impute missing values in a matrix. Unlike bilinear and singular value decomposition methods, this approach is capable of clustering the data into similarity groups. However, it only allows for square matrices and is not applicable to traffic data that often is not in square form.

While the existing decomposition-based methods have shown some levels of effectiveness, they are still not able to fully mine the clustered spatiotemporal information of

traffic data. When performing imputation, these methods use all the data globally without emphasizing roads with similar spatiotemporal characteristics. This issue may limit their effectiveness in estimating missing values, especially in large and complex urban road networks that may show strong clustered patterns.

This paper addresses this issue by proposing a new matrix imputation approach that captures clustered spatiotemporal characteristics of the traffic data. The objective of this study is to extend the existing body of knowledge in traffic data imputation by developing a novel imputation method based on PARATUCK2 matrix decomposition. Harshman and Lundy (*31*) introduced PARATUCK2 decomposition as a generalization of DEDICOM by combining the parallel factors (PARAFAC) and TUCKER2 models. This decomposition is appropriate for problems that involve interactions between different factors and leads to a set of latent factors that capture the underlying low-dimensional patterns of the data. For example, in the case of traffic data, the interaction between spatial and temporal factors and their underlying patterns is captured by PARATUCK2.

In this study, a method is developed as an extension to the PARATUCK2 matrix decomposition approach for data imputation. The proposed method captures spatiotemporal information of traffic data as a matrix and constructs a low-dimensional representation of traffic patterns. At the same time, the proposed approach allows for both spatial and temporal clustering to identify potential road segments and time intervals that have similar traffic patterns. Using this clustered low-dimensional structure, the proposed method estimates missing values by recovering network traffic profiles. To demonstrate the applicability of the proposed approach, the method is applied to the traffic data in the road network of Manhattan in New York City. The performance of the proposed method is compared with two existing state-of-the-art data imputation approaches.

The remainder of the paper is structured as follows. After a brief review of the notations, the proposed data imputation method based on PARATUCK2 is presented. The method is then implemented using traffic data of the Manhattan road network. Next, the performance of the proposed method is empirically evaluated by comparing it with two state-of-the-art data imputation methods. Finally, the outcomes of this study, its contribution, and potential future works are reviewed in the conclusion section.

Notations and Preliminaries

The notation used throughout this paper is very similar to that adopted by Kolda and Badar (*32*) and is commonly used in other publications in the area of matrix decomposition. Matrices and vectors are denoted by boldface capital letters and boldface lowercase letters, for example, $\boldsymbol{A}$ and $\boldsymbol{a}$, respectively. Third-order tensors are denoted by boldface Euler script letters, for example, $\mathcal{X}$, and scalars are represented using lower case letters, for example, a. Given a matrix $\boldsymbol{A}$, the (i, j)th entry of $\boldsymbol{A}$ is denoted by a_{ij}, and $\boldsymbol{a}_i$ is used to denote its ith row.

The transpose of a matrix $\boldsymbol{A} \in \mathbb{R}^{I \times J}$ is represented by $\boldsymbol{A}^T \in \mathbb{R}^{J \times I}$, and its inverse is denoted by $\boldsymbol{A}^{-1}$. The vectorization operator vec$(\cdot)$ converts a matrix into a column vector, and $\|\boldsymbol{A}\|$ denotes the Frobenius norm of a given matrix $\boldsymbol{A}$, which is defined as the square root of the sum of the squares of all entries of matrix $\boldsymbol{A}$.

$$\|\boldsymbol{A}\| = \sqrt{\sum_i \sum_j a_{ij}{}^2} \tag{1}$$

Given two matrices $\boldsymbol{A} \in \mathbb{R}^{I \times J}$ and $\boldsymbol{B} \in \mathbb{R}^{K \times L}$, their Kronecker product is represented by $\boldsymbol{A} \otimes \boldsymbol{B}$, which results in a matrix of size $\mathbb{R}^{IK \times JL}$.

$$\boldsymbol{A} \otimes \boldsymbol{B} = \begin{bmatrix} \boldsymbol{a}_{11}\boldsymbol{B} & \boldsymbol{a}_{12}\boldsymbol{B} & \cdots & \boldsymbol{a}_{1J}\boldsymbol{B} \\ \boldsymbol{a}_{21}\boldsymbol{B} & \boldsymbol{a}_{22}\boldsymbol{B} & \cdots & \boldsymbol{a}_{2J}\boldsymbol{B} \\ \vdots & \vdots & \ddots & \vdots \\ \boldsymbol{a}_{I1}\boldsymbol{B} & \boldsymbol{a}_{I2}\boldsymbol{B} & \cdots & \boldsymbol{a}_{IJ}\boldsymbol{B} \end{bmatrix} \tag{2}$$

PARATUCK2 Data Imputation Model

This section introduces the proposed traffic data imputation method that estimates missing values by capturing clustered spatiotemporal patterns in the traffic data. The proposed method is designed as an extension to the PARATUCK2 matrix decomposition approach. In this method, the traffic data is modeled as a matrix to capture spatiotemporal information. The PARATUCK2 matrix decomposition method constructs a low-dimensional decomposition of a matrix (*31*). This decomposition leads to a set of latent factors that capture the underlying low-dimensional patterns of the data. More specifically, PARATUCK2 represents a matrix $\boldsymbol{X} \in \mathbb{R}^{I \times J}$ as a product of three matrices, as follows:

$$\boldsymbol{X} = \boldsymbol{A}\boldsymbol{R}\boldsymbol{B}^{\boldsymbol{T}}, \tag{3}$$

where $\boldsymbol{A}$, $\boldsymbol{R}$, and $\boldsymbol{B}$ are factor matrices of size $\mathbb{R}^{I \times P}$, $\mathbb{R}^{P \times Q}$, and $\mathbb{R}^{J \times Q}$. This decomposition attempts to group the mode-one (i.e., rows) and mode-two (i.e., columns) of matrix $\boldsymbol{X}$ into P and Q latent factors, respectively. The columns in $\boldsymbol{A}$ and $\boldsymbol{B}$ correspond to the latent factors, such that a_{ip} indicates the association of object i with group p in the first mode of the matrix, and b_{jq} shows the association of object j with group q in the second mode of the matrix. The rectangular matrix $\boldsymbol{R}$ represents the interaction between the P latent factors in $\boldsymbol{A}$ and the Q latent factors in $\boldsymbol{B}$.

PARATUCK2 estimates the factor matrices by minimizing the following loss function:

$$\operatorname{argmin} f(\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{R}) = \left\| \boldsymbol{X} - \boldsymbol{A}\boldsymbol{R}\boldsymbol{B}^T \right\|^2 \tag{4}$$

This loss function is a non-convex optimization problem that is usually solved by an iterative alternating least squares (ALS) algorithm (*33*). The key idea of the ALS algorithm is to optimize f over a selected factor matrix (e.g., $\boldsymbol{A}$) while having the other matrices (e.g., $\boldsymbol{B}$ and $\boldsymbol{R}$) fixed at their last estimated values. Using this approach, the factor matrices are updated iteratively until a convergence criterion is satisfied. The updating rules for the PARATUCK2 decomposition are as follows:

$$\begin{cases} \boldsymbol{A} \leftarrow \boldsymbol{X}\boldsymbol{B}\boldsymbol{R}^{\boldsymbol{T}}\left(\boldsymbol{R}\boldsymbol{B}^{\boldsymbol{T}}\boldsymbol{B}\boldsymbol{R}^{\boldsymbol{T}}\right)^{-1} \\ \boldsymbol{B} \leftarrow \boldsymbol{X}^{\boldsymbol{T}}\boldsymbol{A}\boldsymbol{R}\left(\boldsymbol{R}^{\boldsymbol{T}}\boldsymbol{A}^{\boldsymbol{T}}\boldsymbol{A}\boldsymbol{R}\right)^{-1} \\ \boldsymbol{R} \leftarrow \left(\boldsymbol{A}^{\boldsymbol{T}}\boldsymbol{A}\right)^{-1}\boldsymbol{A}^{\boldsymbol{T}}\boldsymbol{X}\boldsymbol{B}\left(\boldsymbol{B}^{\boldsymbol{T}}\boldsymbol{B}\right)^{-1} \end{cases} \tag{5}$$

PARATUCK2 was not originally developed for situations where the data matrix contains missing values. The proposed method extends PARATUCK2 to perform the decomposition and consequently complete the matrix in the presence of missing values. That is, the extended method can be used for missing traffic data imputation. To extend PARATUCK2 to a decomposition method that can handle missing values, this study attempts to minimize the sum of reconstruction errors over all the observed values in the original matrix, ignoring the missing ones. Therefore, the regularized optimization problem is formulated as follows:

$$\operatorname{argmin} f(\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{R}) = \sum\nolimits_{\Omega}\left(x_{ij} - \boldsymbol{a}_i\boldsymbol{R}\boldsymbol{b}_j^T\right)^2 + \lambda\left(\sum\nolimits_i \|\boldsymbol{a}_i\|^2 + \sum\nolimits_j \|\boldsymbol{b}_j\|^2\right) \tag{6}$$

where Ω is an index set denoting the indices of observed values in matrix $\boldsymbol{X}$. The first term of this objective minimizes the reconstruction errors of the observed entries of $\boldsymbol{X}$, and the second term penalizes the loss function to discourage the complexity of the model and prevent overfitting.

To solve the above problem, the ALS algorithm is used. By alternately fixing two of the factor matrices, this non-convex minimization problem is turned into a set of quadratic problems, which can be solved for the remaining matrix. Therefore, all three matrices can be updated iteratively.

The rows of matrix **A** are updated as follows:

$$\boldsymbol{a}_i = \left(\sum\nolimits_{j \in J_i} x_{ij}\boldsymbol{b}_j\boldsymbol{R}^T\right)\left(\sum\nolimits_{j \in J_i} \boldsymbol{R}\boldsymbol{b}_j^T\boldsymbol{b}_j\boldsymbol{R}^T + \lambda\boldsymbol{I}_p\right)^{-1} \tag{7}$$

where $J_i = \{j \mid x_{ij} \text{ is not missing}\}$ and $\boldsymbol{I_p}$ is the identity matrix of size p.

Similarly, the rows of the matrix $\boldsymbol{B}$ are given by

$$\boldsymbol{b}_j = \left(\sum\nolimits_{i \in I_j} x_{ij}\boldsymbol{a}_i\boldsymbol{R}\right)\left(\sum\nolimits_{i \in I_j} \boldsymbol{R}^T\boldsymbol{a}_i^T\boldsymbol{a}_i\boldsymbol{R} + \lambda\boldsymbol{I}_q\right)^{-1} \tag{8}$$

where $I_j = \{i \mid x_{ij} \text{ is not missing}\}$.

Finally, $\boldsymbol{R}$ can be updated by solving the following systems of linear equations:

$$\operatorname{vec}\left(\sum\nolimits_{\Omega} \boldsymbol{a}_i^T x_{ij}\boldsymbol{b}_j\right) = \sum\nolimits_{\Omega}\left(\left(\boldsymbol{b}_j^T\boldsymbol{b}_j\right)^T \otimes \left(\boldsymbol{a}_i^T\boldsymbol{a}_i\right)\right)\operatorname{vec}(\boldsymbol{R}) \tag{9}$$

In each iteration, the alternating procedure decreases the objective function in Equation 6, which is bounded from below by zero. Therefore, iterating this procedure leads to a solution for $\boldsymbol{A}$, $\boldsymbol{B}$, and $\boldsymbol{R}$ such that either the objective function cannot be improved any further by the updating rules or a maximum number of iterations is reached. Note that this solution may not be a global optimum because of the non-convexity of the problem.

Having the estimated matrices $\boldsymbol{A}$, $\boldsymbol{B}$, and $\boldsymbol{R}$, missing values are then imputed by reconstructing the original data matrix from the predicted latent factors.

$$\hat{\boldsymbol{X}} = \boldsymbol{A}\boldsymbol{R}\boldsymbol{B}^{\boldsymbol{T}} \tag{10}$$

The pseudocode of the proposed method for missing value imputation is outlined below. The advantage of the proposed method over the existing matrix completion methods reviewed previously is its capability to capture both spatial and temporal patterns, which often exist in traffic data.

Empirical Performance Evaluation

This section presents the implementation of the proposed traffic data imputation method using traffic data in the Manhattan road network and the evaluation of its performance in comparison with two state-of-the-art benchmark methods. In the remainder of this section, first, the

The Proposed Data Imputation Algorithm

Algorithm: imputer ($\boldsymbol{X}$, P, Q, λ)

Initialize $\boldsymbol{A} \in \mathbb{R}^{I \times P}$, $\boldsymbol{B} \in \mathbb{R}^{J \times Q}$ and $\boldsymbol{R} \in \mathbb{R}^{P \times Q}$ as small random values
repeat
  for $i = 1$ **to** I **do**
    update $\boldsymbol{a}_i$ by **Equation 7**
  update $\boldsymbol{R}$ by **Equation 9**
  for $j = 1$ **to** J **do**
    update $\boldsymbol{b}_j$ by **Equation 8**
until fit ceases to improve or maximum iterations is exhausted
return imputed matrix $\hat{\boldsymbol{X}}$, set of factor matrices $\{\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{R}\}$

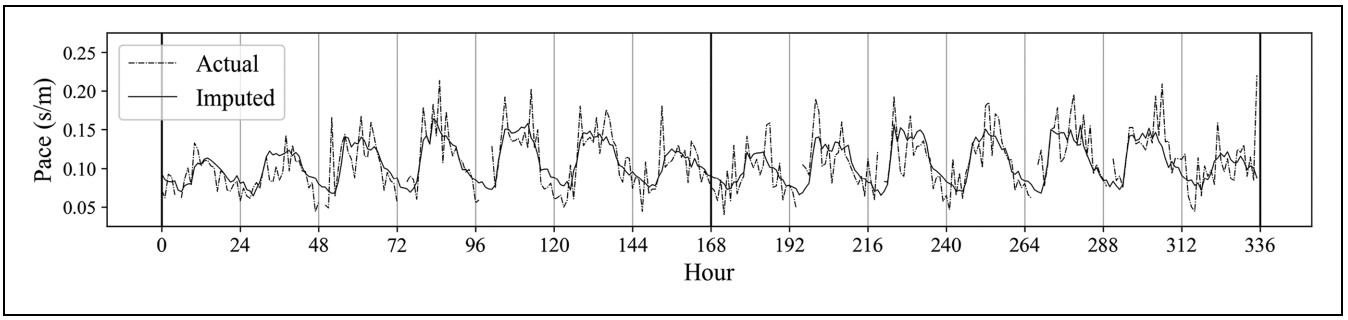

Figure 1. Example of data imputation in a road segment.

dataset is introduced. Next, the steps conducted to implement the method are explained. The results are then presented, and finally, the performance of the proposed method is compared with the benchmark methods. The benchmarks are two state-of-the-art existing data imputation methods.

The first benchmark is the low-rank matrix completion using UV-decomposition (UVD) (*23*). This method is a matrix completion model, in which factor matrices $\boldsymbol{U}$ and $\boldsymbol{V}$ represent the low-rank structure of the traffic data matrix $\boldsymbol{X}$, and they are obtained by minimizing the Frobenius norm of $\boldsymbol{X} - \boldsymbol{U}\boldsymbol{V}^T$ over the observed values.

The second benchmark is the tensor recovery based on robust CP-decomposition (RCPD) (*27*, *34*). This approach is a low-rank tensor completion model, in which the traffic data tensor $\mathcal{T}$ is decomposed as the sum of a low-rank tensor (i.e., $\mathcal{X}$), and a sparse outlier tensor (i.e., $\mathcal{E}$). Here, $\mathcal{X}$ and $\mathcal{E}$ are estimated by minimizing the weighted loss that penalizes the rank of tensor $\mathcal{X}$ and sparsity level of $\mathcal{E}$, using nuclear norm and l_1 norm approximations while imposing the decomposition to match the traffic data at available entries.

Manhattan Road Network Experiment

The New York City (NYC) road traffic dataset was prepared by Donovan and Work (*35*). The traffic data in this dataset is estimated using taxi GPS traces. The dataset contains hourly average travel time on individual road segments of the NYC road network for four years, from January 2010 to December 2013. It also includes information about the coordinates and length of each road segment. For the purpose of this study, the traffic data for a window of two weeks, from Sunday, April 4, 2010, to Sunday, April 18, 2010, in the borough of Manhattan is used. Previous studies (*36*, *37*) indicated that the NYC road network did not experience any extreme events and unusual disruptions during this window of time. The road network in Manhattan consists of 8,822 road segments represented in the dataset using 3,910 nodes and 8,822 links.

Because of the varying length of links in road networks, normalized travel time against distance, called pace, is often used to measure network performance in urban traffic monitoring (*36*, *37*). This study also use pace or travel time (in seconds) per meter. Using the traffic dataset, a matrix $\boldsymbol{M}$ of size $\mathbb{R}^{N \times T}$ is created, where $N = 8{,}822$ represents the number of road segments in the Manhattan network, and $T = 336$ is the number of data points for each road segment over the selected period of time (i.e., April 4, 2010, to April 18, 2010). This matrix contains average pace values of all the N links in the network from hour 1 to hour 336. However, not all entries of $\boldsymbol{M}$ are known; 32.5% of traffic values are missing.

Missing Traffic Data Imputation. To assess the performance of the proposed method for missing traffic data imputation, it is applied to the network matrix $\boldsymbol{M}$ described in the previous section. Using the proposed method, the network matrix is decomposed into three matrices (i.e., $\boldsymbol{A} \in \mathbb{R}^{N \times P}$, $\boldsymbol{B} \in \mathbb{R}^{T \times Q}$ and $\boldsymbol{R} \in \mathbb{R}^{P \times Q}$), each of which contains information about spatial or temporal correlations that may exist between the road segments in the network. Matrix $\boldsymbol{A}$ attempts to group the road segments into P spatial clusters. The $(i,p)^{th}$ entry of matrix $\boldsymbol{A}$, denoted by a_{ip}, indicates the association of road segment i with cluster p. Similarly, matrix $\boldsymbol{B}$ captures the existing temporal patterns in the traffic data by grouping the time intervals (i.e., hours) into Q latent factors. Finally, matrix $\boldsymbol{R}$ shows the interaction between P spatial groups in A and Q temporal groups in $\boldsymbol{B}$.

To choose an appropriate number of latent factors (i.e., P and Q), a preliminary assessment is conducted. Decomposition results for different values of P and Q show that even when P is large, the network road segments are primarily associated with five latent factors (i.e., spatial clusters). It also shows that for Q greater than or equal to seven, the method could successfully capture the underlying temporal patterns. Therefore, $P = 5$ and $Q = 7$ were selected to perform the decomposition. Having the matrices $\boldsymbol{A}$, $\boldsymbol{B}$, and $\boldsymbol{R}$, data imputation is first performed. For example, Figure 1 shows the

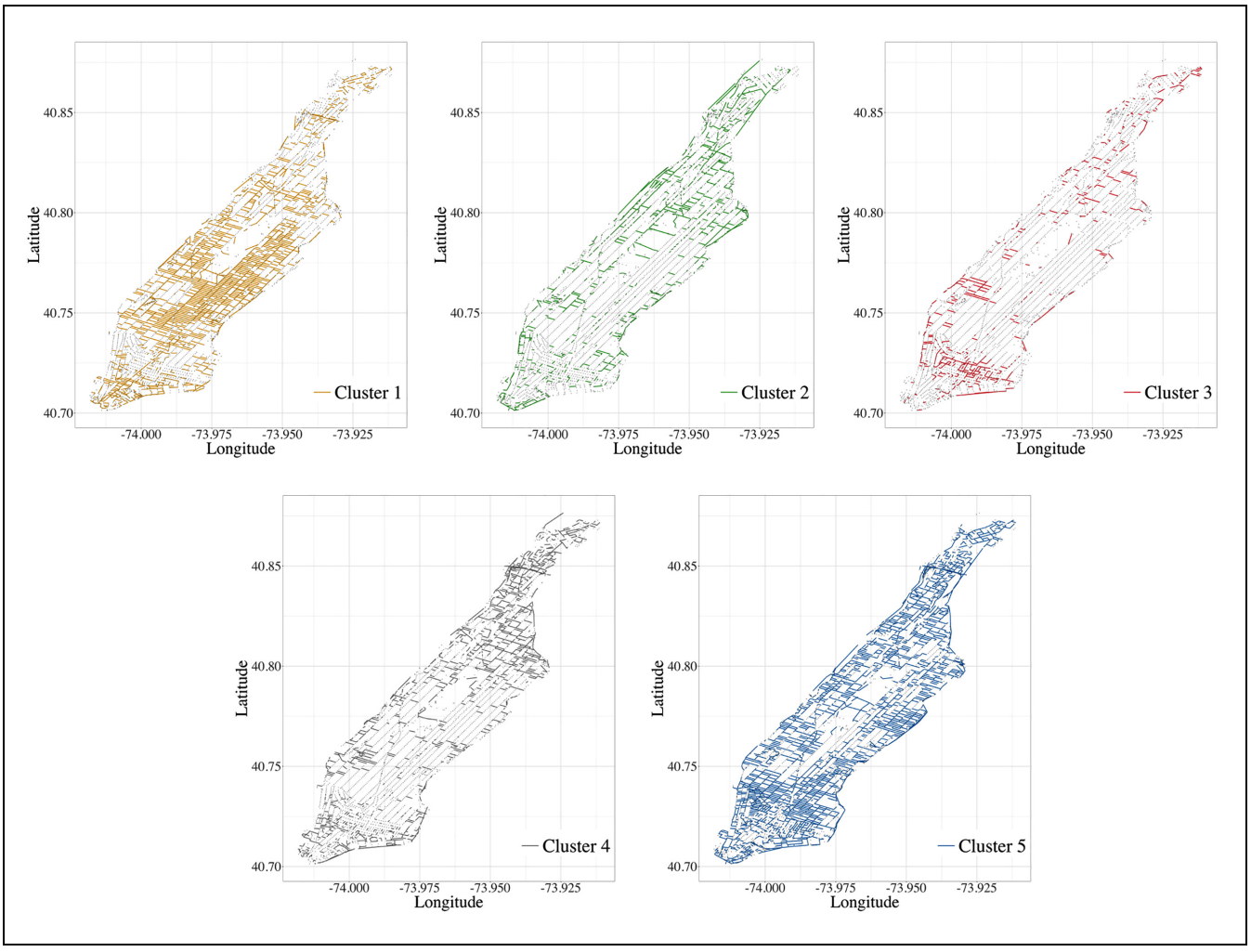

Figure 2. Spatial clusters determined by the PARATUCK2 imputation model on the Manhattan road network.

actual (containing some missing values) and estimated (including imputed values for missing data) pace for a road segment during the analysis period.

Next, the imputation process is performed individually for each of the five spatial clusters identified in the previous step. A cluster contains a group of road segments that are similar in their traffic patterns. The road i is assigned to the cluster p, when a_{ip} is greater than a_{ij} for any other cluster j. Figure 2 shows the spatial clusters determined by the method when $P = 5$ in the decomposition.

It should be noted that roads in a cluster are not necessarily spatially clustered, rather they are grouped based on their traffic patterns over time. To elaborate on this, Figure 3 presents the hourly average pace of each identified cluster during the selected two-week period (April 4, 2010, to April 18, 2010). For example, clusters 1 and 4 indicate busier road segments (i.e., links with higher average pace) in the network and links with the lowest average pace have been grouped into cluster 2. These observations are consistent with the empirical knowledge about the traffic patterns of the roads in clusters shown in Figure 2.

Figure 4 demonstrates the hourly average pace of the entire network from Sunday, April 4, 2010, to Sunday, April 18, 2010, along with the temporal groups. Each group contains time intervals (i.e., hours) with similar traffic patterns. As is shown in the figure, the method is able to capture the underlying temporal patterns of the traffic data. For example, groups 5 and 6 mostly represent peak hours (i.e., hours with high average pace) during weekdays, and group 2 includes off-peak hours. Peak hours during the weekends have also been mainly assigned to group 3.

To better evaluate the imputation performance of the model, it is also cross-validated by masking a certain

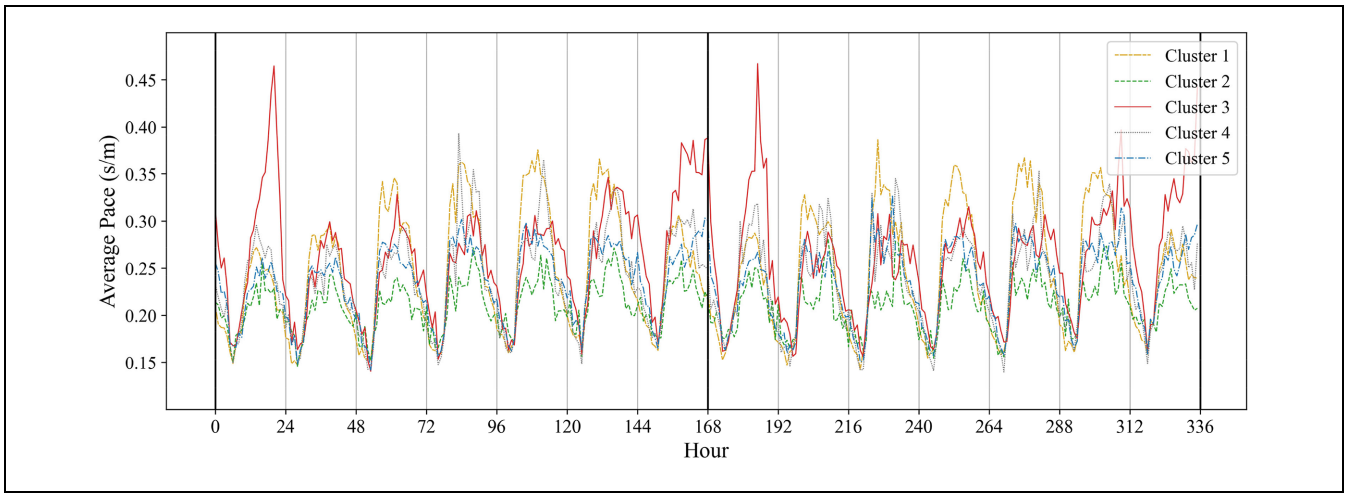

Figure 3. Hourly average pace on the five spatial clusters determined by the model during the selected two-week period (April 4, 2010, to April 18, 2010).

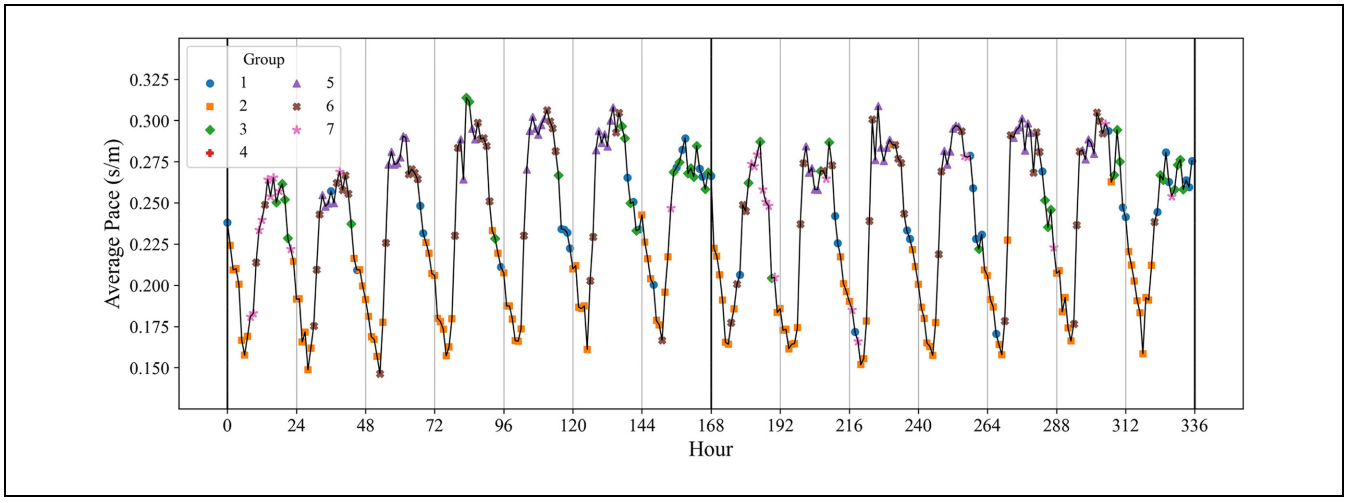

Figure 4. Illustration of the hourly average pace on the Manhattan network and temporal groups determined by the PARATUCK2 model.

number of observations as missing values. Here, 10% of the available traffic values are masked (in addition to the original missing values of the data) and used as testing data. The remaining observations are taken as training data to learn the model.

The imputation performance is evaluated using the median absolute percentage error (MdAPE), which is an evaluation metric less sensitive to outliers (*38*):

$$\text{MdAPE} = \text{median}\left(\frac{|y_i - \hat{y}_i|}{y_i} \times 100\right), \qquad (11)$$

where y_i and $\hat{y}_i$ are actual and imputed values of masked entries, respectively.

Table 1 reports the validation errors (in MdAPE) based on the 10% testing data for each cluster. Here, the actual values of the masked entries are used as the ground truth to compute MdAPEs. The results show that the proposed method can reconstruct traffic profiles with reasonable accuracy, even for clusters with high fluctuations in pace (e.g., cluster 3).

Performance Evaluation in Comparison With Benchmark Methods. To further evaluate the imputation performance of the proposed method, missing values are also imputed for the entire network using the proposed approach and the two benchmarks described before. The imputation is performed by randomly masking p% (i.e., 1%, 5%, 10%, and 20%) of available traffic data as missing values and considering them as ground truth to evaluate the accuracy of estimation. The remaining partial observations are used to feed the models. To make an accurate comparison, this procedure is repeated 10

Table 1. Validation Errors in Each Cluster Determined by the Proposed Method

Cluster	Number of links in cluster	Percentage of missing data	Validation error MdAPE
Cluster 1	2201	31.29	27.08
Cluster 2	1179	38.68	33.14
Cluster 3	579	37.64	29.65
Cluster 4	1410	52.79	40.73
Cluster 5	3453	39.37	30.69

Note: MdAPE = median absolute percentage error.

Table 2. Performance Comparison of the Proposed Method With Benchmarks

p	PARATUCK2	UV-decomposition	Robust CP-decomposition
1	30.28 (0.14)	31.84 (0.12)	38.66 (0.09)
5	30.31 (0.21)	31.88 (0.16)	38.72 (0.10)
10	30.34 (0.11)	31.97 (0.10)	42.11 (0.10)
20	30.48 (0.24)	32.13 (0.10)	54.50 (0.07)

Note: UV = ; CP = canonical polyadic; UV = bilinear factorization.

times and the validation errors (in MdAPE) and their standard deviation are calculated over the 10 trials of imputation for each method.

The data matrix which is used for the first benchmark is the same as that created for the proposed imputation method (i.e., matrix $\boldsymbol{M}$). However, for the second benchmark, the traffic data is prepared in the form of a third-order tensor, where the first, second, and third dimensions represent road segments, hours of the day, and days, respectively. Thus, a data tensor of size $8{,}822 \times 24 \times 14$ is constructed. The hyperparameters used for benchmarks are also tuned for their best performance in this setting.

Table 2 provides the mean and standard deviation of the validation errors (in MdAPE) obtained over 10 trials of imputation for the proposed method and the benchmark models (Figure 5). As reported, the proposed approach outperforms the benchmarks at all levels of missing values. For example, when $p = 10\%$, the MdAPE obtained by the proposed method is 30.34%, which is smaller than the values obtained by UVD (31.97%) and RCPD (42.11%). All Python codes developed to implement and evaluate the proposed method are publicly available at a GitHub repository: https://github.com/minanouri/Traffic_Data_Imputation_Based_on_PARATUCK2.git

Conclusion

Despite recent advances in data-driven and intelligent transportation systems, missing observations in traffic data is still a significant concern that can affect the accuracy and reliability of traffic monitoring and forecasting models. A considerable number of the existing traffic data imputation approaches are based on matrix completion techniques. However, these methods may not fully capture the clustered spatiotemporal structure of the traffic data. To address this limitation and gap in knowledge, this study extends the existing body of knowledge in traffic data imputation by developing a novel imputation method using PARATUCK2 decomposition, which can capture the spatiotemporal information embedded in traffic data. The proposed method constructs a low-dimensional and clustered representation of traffic patterns and estimates missing values using extracted low-dimensional structure of the data. The model allows for both spatial and temporal clustering to determine potential road segments and time intervals that have similar traffic patterns. The method then takes advantage of the inherent spatiotemporal information of traffic data to estimate missing values. The proposed method is implemented and evaluated empirically using the hourly road traffic data in Manhattan, NYC, for two weeks. This dataset contains 8,822 road segments with complex clustered traffic patterns. The data imputation performance of the proposed method is compared with two state-of-the-art benchmark methods. The outcomes indicate that the proposed method outperforms the existing state-of-the-art imputation methods in complex and large-scale traffic networks. The future work includes using tensors for data completion and capturing the sparsity structure of the transportation networks.

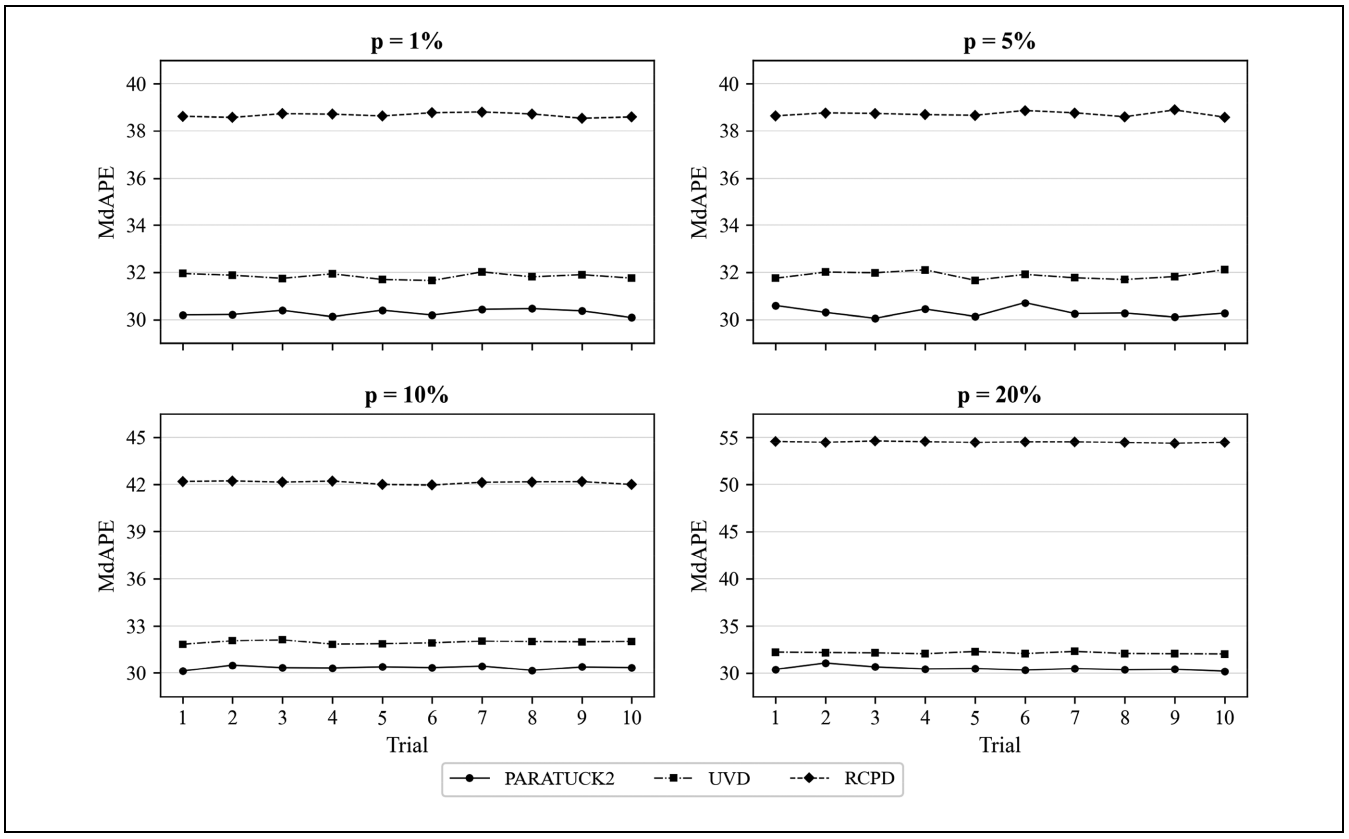

Figure 5. Performance comparison of the proposed method with benchmarks for all 10 trials.

Furthermore, techniques such as federated learning to incorporate the heterogeneity of the traffic data in imputing the data should be considered.

Author Contributions

The authors confirm contribution to the paper as follows: study conception and design: M. Nouri, M. Reisi-Gahrooei, M. Ilbeigi; data collection: M. Nouri; analysis and interpretation of results: M. Nouri, M. Reisi-Gahrooei; draft manuscript preparation: M. Nouri, M. Reisi-Gahrooei, M. Ilbeigi. All authors reviewed the results and approved the final version of the manuscript.

Declaration of Conflicting Interests

The author(s) declared no potential conflicts of interest with respect to the research, authorship, and/or publication of this article.

Funding

The author(s) disclosed receipt of the following financial support for the research, authorship, and/or publication of this article: This material is based on work supported by the National Science Foundation under Grants CMMI-2027025 and CMMI-2026795.

ORCID iDs

Mina Nouri https://orcid.org/0000-0002-6683-6260
Mostafa Reisi-Gahrooei https://orcid.org/0000-0002-7633-9575
Mohammad Ilbeigi https://orcid.org/0000-0001-6576-3808

References

1. Chen, X., Z. He, and L. Sun. A Bayesian Tensor Decomposition Approach for Spatiotemporal Traffic Data Imputation. *Transportation Research Part C: Emerging Technologies*, Vol. 98, 2019, pp. 73–84.
2. Ni, D., J. D. Leonard, A. Guin, and C. Feng. Multiple Imputation Scheme for Overcoming the Missing Values and Variability Issues in ITS Data. *Journal of Transportation Engineering*, Vol. 131, No. 12, 2005, pp. 931–938.
3. Li, Y., Z. Li, and L. Li. Missing Traffic Data: Comparison of Imputation Methods. *IET Intelligent Transport Systems*, Vol. 8, No. 1, 2014, pp. 51–57.
4. Park, B., C. J. Messer, and T. Urbanik. Short-Term Freeway Traffic Volume Forecasting Using Radial Basis Function Neural Network. *Transportation Research Record: Journal of the Transportation Research Board*, 1998. 1651: 39–47.
5. Zhong, M., S. Sharma, and P. Lingras. Genetically Designed Models for Accurate Imputation of Missing

Traffic Counts. *Transportation Research Record: Journal of the Transportation Research Board*, 2004. 1879: 71–79.
6. Zhang, C., S. Sun, and G. Yu. A Bayesian Network Approach to Time Series Forecasting of Short-Term Traffic Flows. *Proc., 7th International IEEE Conference on Intelligent Transportation Systems (IEEE Cat. No. 04TH8749)*, Washington, WA, October 3–6, 2004. IEEE, New York, 2004, pp. 216–221.
7. Ghosh, B., B. Basu, and M. O'Mahony. Bayesian Time-Series Model for Short-Term Traffic Flow Forecasting. *Journal of Transportation Engineering*, Vol. 133, No. 3, 2007, pp. 180–189.
8. Dia, H. An Object-Oriented Neural Network Approach to Short-Term Traffic Forecasting. *European Journal of Operational Research*, Vol. 131, No. 2, 2001, pp. 253–261.
9. Vlahogianni, E. I., M. G. Karlaftis, and J. C. Golias. Optimized and Meta-Optimized Neural Networks for Short-Term Traffic Flow Prediction: A Genetic Approach. *Transportation Research Part C: Emerging Technologies*, Vol. 13, No. 3, 2005, pp. 211–234.
10. Castro-Neto, M., Y. S. Jeong, M. K. Jeong, and L. D. Han. Online-SVR for Short-Term Traffic Flow Prediction Under Typical and Atypical Traffic Conditions. *Expert Systems With Applications*, Vol. 36, No. 3, 2009, pp. 6164–6173.
11. Jin, X., Y. Zhang, and D. Yao. Simultaneously Prediction of Network Traffic Flow Based on PCA-SVR. *Proc., International Symposium on Neural Networks. Advances in Neural Networks – ISSN 2007*, Springer, Berlin and Heidelberg, 2007, pp. 1022–1031.
12. Yin, W., P. Murray-Tuite, and H. Rakha. Imputing Erroneous Data of Single-Station Loop Detectors for Nonincident Conditions: Comparison Between Temporal and Spatial Methods. *Journal of Intelligent Transportation Systems*, Vol. 16, No. 3, 2012, pp. 159–176.
13. Zhong, M., S. Sharma, and Z. Liu. Assessing Robustness of Imputation Models Based on Data From Different Jurisdictions: Examples of Alberta and Saskatchewan, Canada. *Transportation Research Record: Journal of the Transportation Research Board*, 2005. 1917: 116–126.
14. Troyanskaya, O., M. Cantor, G. Sherlock, P. Brown, T. Hastie, R. Tibshirani, D. Botstein, and R. B. Altman. Missing Value Estimation Methods for DNA Microarrays. *Bioinformatics*, Vol. 17, No. 6, 2001, pp. 520–525.
15. Ni, D., and J. D. Leonard. Markov Chain Monte Carlo Multiple Imputation Using Bayesian Networks for Incomplete Intelligent Transportation Systems Data. *Transportation Research Record: Journal of the Transportation Research Board*, 2005. 1935: 57–67.
16. Qu, L., L. Li, Y. Zhang, and J. Hu. PPCA-Based Missing Data Imputation for Traffic Flow Volume: A Systematical Approach. *IEEE Transactions on Intelligent Transportation Systems*, Vol. 10, No. 3, 2009, pp. 512–522.
17. Shang, Q., Z. Yang, S. Gao, and D. Tan. An Imputation Method for Missing Traffic Data Based on FCM Optimized by PSO-SVR. *Journal of Advanced Transportation*, Vol. 2018, 2018, pp. 1–21. https://doi.org/10.1155/2018/2935248.
18. Chang, G., and T. Ge. Comparison of Missing Data Imputation Methods for Traffic Flow. *Proc., International Conference on Transportation, Mechanical and Electrical Engineering (TMEE)*, Changchun, China. IEEE, New York, 2011, pp. 639–642.
19. Chen, X., Z. Wei, Z. Li, J. Liang, Y. Cai, and B. Zhang. Ensemble Correlation-Based Low-Rank Matrix Completion With Applications to Traffic Data Imputation. *Knowledge-Based Systems*, Vol. 132, 2017, pp. 249–262.
20. Du, R., C. Chen, B. Yang, and X. Guan. VANET Based Traffic Estimation: A Matrix Completion Approach. *Proc., IEEE Global Communications Conference (GLOBECOM)*, Atlanta, GA. IEEE, New York, 2013, pp. 30–35.
21. Zhang, D., Y. Hu, J. Ye, X. Li, and X. He. Matrix Completion by Truncated Nuclear Norm Regularization. *Proc., IEEE Conference on Computer Vision and Pattern Recognition*, Providence, RI. IEEE, New York, 2012, pp. 2192–2199.
22. Ma, R., N. Barzigar, A. Roozgard, and S. Cheng. Decomposition Approach for Low-Rank Matrix Completion and its Applications. *IEEE Transactions on Signal Processing*, Vol. 62, No. 7, 2014, pp. 1671–1683.
23. Jain, P., P. Netrapalli, and S. Sanghavi. Low-Rank Matrix Completion Using Alternating Minimization. *Proc., Forty-Fifth Annual ACM Symposium on Theory of Computing*, Palo Alto, CA. Association for Computing Machinery, New York, 2013, pp. 665–674.
24. Wall, M. E., A. Rechtsteiner, and L. M. Rocha. Singular Value Decomposition and Principal Component Analysis. In *A Practical Approach to Microarray Data Analysis* (Berrar, D. P., W. Dubitzky, and M. Granzow, eds.), Springer, Boston, MA, 2003, pp. 91–109.
25. Gong, C., and Y. Zhang. Urban Traffic Data Imputation With Detrending and Tensor Decomposition. *IEEE Access*, Vol. 8, 2020, pp. 11124–11137.
26. Lu, W., T. Zhou, L. Li, Y. Gu, Y. Rui, and B. Ran. An Improved Tucker Decomposition-Based Imputation Method for Recovering Lane-Level Missing Values in Traffic Data. *IET Intelligent Transport Systems*, Vol. 16, 2022, pp. 363–379.
27. Goldfarb, D., and Z. Qin. Robust Low-Rank Tensor Recovery: Models and Algorithms. *SIAM Journal on Matrix Analysis and Applications*, Vol. 35, No. 1, 2014, pp. 225–253.
28. Chen, X., J. Yang, and L. Sun. A Nonconvex Low-Rank Tensor Completion Model for Spatiotemporal Traffic Data Imputation. *Transportation Research Part C: Emerging Technologies*, Vol. 117, 2020, p. 102673.
29. Chen, X., Z. He, Y. Chen, Y. Lu, and J. Wang. Missing Traffic Data Imputation and Pattern Discovery With a Bayesian Augmented Tensor Factorization Model. *Transportation Research Part C: Emerging Technologies*, Vol. 104, 2019, pp. 66–77.
30. Harshman, R. A. Models for Analysis of Asymmetrical Relationships Among N Objects or Stimuli. *Proc., First Joint Meeting of the Psychometric Society and the Society of Mathematical Psychology*, Hamilton, ON, 1978, pp. 1–25.
31. Harshman, R. A., and M. E. Lundy. Uniqueness Proof for a Family of Models Sharing Features of Tucker's Three-Mode Factor Analysis and PARAFAC/CANDECOMP. *Psychometrika*, Vol. 61, 1996, pp. 133–154.

32. Kolda, T. G., and B. W. Bader. Tensor Decompositions and Applications. *SIAM Review*, Vol. 51, No. 3, 2009, pp. 455–500.
33. Kiers, H. A. L. An Alternating Least Squares Algorithm for Fitting the Two- and Three-Way DEDICOM Model and the IDIOSCAL Model. *Psychometrika*, Vol. 54, No. 3, 1989, pp. 515–521.
34. Hu, Y., and D. B. Work. Robust Tensor Recovery With Fiber Outliers for Traffic Events. *ACM Transactions on Knowledge Discovery From Data (TKDD)*, Vol. 15, 2021, pp. 1–27.
35. Donovan, B., and D. B. Work. *Open Data: 2010–2013 New York City Traffic Estimates*. University of Illinois, June 13, 2015. https://uofi.app.box.com/v/NYC-traffic-estimates.
36. Donovan, B., and D. B. Work. Empirically Quantifying City-Scale Transportation System Resilience to Extreme Events. *Transportation Research Part C: Emerging Technologies*, Vol. 79, 2017, pp. 333–346.
37. Ilbeigi, M. Statistical Process Control for Analyzing Resilience of Transportation Networks. *International Journal of Disaster Risk Reduction*, Vol. 33, 2019, pp. 155–161.
38. Fildes, R. The Evaluation of Extrapolative Forecasting Methods. *International Journal of Forecasting*, Vol. 8, No. 1, 1992, pp. 81–98.

Any opinions, findings, and conclusions or recommendations expressed in this paper are those of the authors and do not necessarily reflect the views of the National Science Foundation.