Urban Network-Wide Traffic Speed Estimation with Massive Ride-Sourcing GPS Traces

Jingru Yu¹, Marc E. J. Stettler², Panagiotis Angeloudis², Simon Hu³, Xiqun (Michael) Chen⁴,*

¹College of Civil Engineering and Architecture, Zhejiang University, Hangzhou 310058, China
²Centre for Transport Studies, Department of Civil and Environmental Engineering, Imperial College London, London SW7 2AZ, United Kingdom
³Zhejiang University/University of Illinois at Urbana-Champaign Institute (ZJU-UlUC Institute), Haining 314400, China

Abstract: The ability to obtain accurate estimates of city-wide urban traffic patterns is essential for the development of effective intelligent transportation systems and the efficient operation of smart mobility platforms. This paper focuses on the network-wide traffic speed estimation, using trajectory data generated by a city-wide fleet of ride-sourcing vehicles equipped with GPS-capable smartphones. A cell-based map-matching technique is proposed to link vehicle trajectories with road geometries, and to produce network-wide spatio-temporal speed matrices. Data limitations are addressed using the Schatten $p$-norm matrix completion algorithm, which can minimize speed estimation errors even with high rates of data unavailability. A case study using data from Chengdu, China, demonstrates that the algorithm performs well even in situations involving continuous data loss over a few hours, and consequently, addresses large-scale network-wide traffic state estimation problems with missing data, while at the same time outperforming other data recovery techniques that were used as benchmarks. Our approach can be used to generate congestion maps that can help monitor and visualize traffic dynamics across the network, and therefore form the basis for new traffic management, proactive congestion identification, and congestion mitigation strategies.

Keywords: Ride-sourcing; GPS traces; data recovery; matrix completion; urban mobility; speed estimation

*Corresponding author. Mail address: B828 Anzhong Building, College of Civil Engineering and Architecture, Zhejiang University, 866 Yuhangtang Rd, Hangzhou 310058, China; Tel.: (+86)571-8820-8938; E-mail: chenxiqun@zju.edu.cn
1. INTRODUCTION

The monitoring, estimation, and prediction of urban traffic conditions are important functions of modern urban planning, traffic management, and environmental monitoring. Recent attempts to deploy smart mobility and intelligent traffic management systems (coordinated traffic control, emergency route planning, proactive congestion management) rely on the ability of transport authorities to precisely estimate traffic speeds across the network, with high levels of accuracy.

The emergence of new technologies, such as GPS enabled smartphones, has facilitated the collection of data from large-scale road networks with different road grades. Recent studies, for instance, have demonstrated how taxis and probe vehicles equipped with GPS units can be used to collect data for the estimation of network-wide traffic performance metric (Chen et al., 2011; Zhan et al., 2013; Zhang et al., 2013; Zhang et al., 2015). However, these studies haven’t address traffic speed estimation over a large-scale road network, which is one of the essential parts of traffic conditions.

There exists a considerable range of challenges that hinder efforts to determine urban-wide traffic speed estimates, including (a) the lack of efficient processing techniques for GPS traces sourced from large-scale urban road networks; (b) difficulties in achieving complete spatiotemporal data coverage (missing data); and (c) the need to consider the unique nature of the various link and intersection categories encountered in urban road networks.

This study addresses the above issues by developing a novel, efficient, and integrated speed estimation technique that incorporates aspects of cell-based map-matching and low-rank matrix completion for data recovery. Our map-matching technique builds upon the earlier work by He et al. (2017) on highway networks, introducing extensions that can accommodate dense road layouts and intersections that are present in urban networks. A matrix completion approach was used for data recovery, building upon a low-rank matrix completion technique that was previously developed by Chen et al. (2017) and extended to make use of network properties and temporal characteristics of traffic patterns. Finally, stochastic congestion map generation was used to illustrate the probability of congestion occurrence across the network.

The rest of this paper is organized as follows; Section 2 reviews related studies in processing GPS traces, map matching, missing data recovery, and traffic speed estimation. Section 3 proposes the map-matching algorithm, missing data recovery method, and network-wide traffic speed method. Section 4 describes the real-world ride-sourcing vehicle GPS data used in this study. Then Section 5 presents the numerical experiments to evaluate the performance of the proposed traffic speed estimation framework. Finally, Section 6 concludes this paper and outlooks on future research.

2. LITERATURE REVIEW

The effective collection and processing of GPS trace data have received increasing attention
from data mining, intelligent transportation systems, database, and ubiquitous computing communities (Zhang et al., 2011; Castro et al., 2013). A variety of research issues have been addressed leveraging large-scale GPS traces, anomalous trajectory detection (Zhang et al., 2011; Chen et al., 2011), taxi/passenger search strategies (Lee et al., 2008; Chang et al., 2010; Powell et al., 2011; Zhang et al., 2015), and characterization of urban mobility patterns (Cao et al., 2005; Jiang et al., 2009; Chang et al., 2010; Li et al., 2012) besides traffic state estimation and prediction (Castro et al., 2012; Kong et al., 2013). Recently, the advances in GPS technology have offered abundant datasets and new opportunities to improve upon such traditional means, which can provide much broader coverage of road networks.

GPS traces and advanced models have been applied to traffic state estimation across a range of settings, such as density (Papadopoulou et al., 2018), congestion (Kan et al., 2019), and travel times (Li et al., 2017). Table 1 summarizes the studies of inferring traffic state with GPS data. Also, speed estimation has been investigated by many researchers, and the scale of the research area varies from a single road to the downtown road network. Simulation and machine learning methods have been used to capture the traffic speed estimation problem (e.g., Herrera et al., 2010; Zhao et al., 2011; Tao et al., 2012; Zhang et al., 2013).

Map matching has received a lot of attention by the academic community over the years, (Quddus et al., 2007; Quddus and Washington, 2015), both for high-resolution in-vehicle navigation applications (White et al., 2000; Greenfeld, 2002; Marchal et al., 2005; Quddus et al., 2006; Chen et al., 2008; Schuessler and Axhausen, 2009; Velaga et al., 2009; Yang et al., 2011) as well as for the manipulation of probe data with a low temporal resolution (Hunter et al., 2014).

Data inputs used in map-matching GPS traces from vehicle receivers (in some cases enhanced using dead-reckoning mechanisms), alongside high-resolution spatial road network maps. Pyo et al. (2001) proposed a map-matching method using the multiple hypothesis techniques to determine a road in a probabilistic approach that provided consistent performance even in complex downtown areas, overpass/underpass areas, and in the areas where roads were adjacent in parallel.

Given the variable levels of spatiotemporal data coverage across urban transport networks, the application of data recovery methods would be essential to ensure that the traffic estimation methods produce reliable results. This is particularly pertinent in the case of ridesharing datasets, where usage might vary considerably over the course of the day. A range of techniques have been used in the past, with wavelet and principal component analysis (PCA) being the most widely used anomaly detection techniques, which can utilize prior low-rank properties, temporal traffic matrix characteristics, or other matrix completion-based methods (Lakhina et al., 2004; Ringberg et al., 2007; Keshavan et al., 2010; Pascoal et al., 2012; Hoang et al., 2018).

Recently, several studies formulated the traffic matrix estimation as a compressive sensing based problem (Chen et al., 2014; Jiang et al., 2016). Moreover, many efficient algorithms for solving the low-rank matrix completion have been proposed, e.g., singular value thresholding
algorithm (Cai et al., 2010), non-negative matrix factorization (Lee et al., 2001; Mao and Saul, 2004), low-rank matrix fitting algorithm (Wen et al., 2012), and gradient descent algorithm on the Grassman manifold (Keshavan et al., 2010).

**Table 1. An overview of inferring traffic state with GPS data studies**

<table>
<thead>
<tr>
<th>Studies</th>
<th>Data Source</th>
<th>Modes</th>
<th>Methodology</th>
<th>Variables</th>
</tr>
</thead>
<tbody>
<tr>
<td>Castro et al. (2012)</td>
<td>Taxi</td>
<td>Road network</td>
<td>Markov logic network</td>
<td>Speed, travel time, Traffic density</td>
</tr>
<tr>
<td>D'Este et al. (1999)</td>
<td>Floating car</td>
<td>A specific route</td>
<td>Location difference speed</td>
<td></td>
</tr>
<tr>
<td>Diker et al. (2012)</td>
<td>Probe car</td>
<td>Road</td>
<td>Fuzzy neighborhood density-based spatial</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>clustering of applications with noise</td>
<td></td>
</tr>
<tr>
<td>Herrera et al. (2010)</td>
<td>GPS-enabled Nokia N95 phone Taxi Intersection</td>
<td>A 10-mile stretch of</td>
<td>Virtual trip lines</td>
<td>Speed</td>
</tr>
<tr>
<td>Kan et al. (2019)</td>
<td>Taxi</td>
<td>Intersection</td>
<td>Clustering</td>
<td>Queue length, Speed</td>
</tr>
<tr>
<td>Kong et al. (2013)</td>
<td>Probe vehicle</td>
<td>Urban road network</td>
<td>Curve Fitting and Vehicle tracking</td>
<td></td>
</tr>
<tr>
<td>Li and McDonald (2002)</td>
<td>Probe vehicle</td>
<td>A stretch of freeway</td>
<td>Mathematical model</td>
<td></td>
</tr>
<tr>
<td>Li et al. (2017)</td>
<td>Probe vehicle</td>
<td>Road network</td>
<td>Compressed sensing algorithm</td>
<td>Link travel time</td>
</tr>
<tr>
<td>Necula (2015)</td>
<td>Floating car</td>
<td>Street segments</td>
<td>Clustering</td>
<td>Traffic volume, Densities, on-ramp and off-ramp flow</td>
</tr>
<tr>
<td>Papadopoulou et al. (2018)</td>
<td>Probe vehicle</td>
<td>Highway</td>
<td>Model-based approach</td>
<td></td>
</tr>
<tr>
<td>Thianniwet et al. (2009)</td>
<td>Floating car</td>
<td>Highly congested roads in Bangkok</td>
<td>Decision tree algorithm and sliding windows</td>
<td></td>
</tr>
<tr>
<td>Tao et al. (2012)</td>
<td>Probe Vehicle</td>
<td>Road network of downtown area</td>
<td>Microscopic traffic simulation-based method</td>
<td>Speed</td>
</tr>
<tr>
<td>Tong et al. (2005)</td>
<td>Probe Vehicle</td>
<td>Highway segment</td>
<td>GPS probe vehicle data integrated with GIS</td>
<td>Travel time, speed and congestion index</td>
</tr>
<tr>
<td>Work et al. (2008)</td>
<td>GPS-enabled Nokia N95 phone</td>
<td>Highway</td>
<td>An ensemble Kalman filtering approach</td>
<td></td>
</tr>
<tr>
<td>Zhao et al. (2011)</td>
<td>Probe vehicle</td>
<td>Road network</td>
<td>Curve Fitting</td>
<td>Speed</td>
</tr>
<tr>
<td>Zhang et al. (2013)</td>
<td>Field-experiment data set (Mobile Century) and three simulated data sets.</td>
<td>Freeway</td>
<td>Increasing weights of recent records and high velocity</td>
<td>Speed, travel time</td>
</tr>
</tbody>
</table>
Empirical results showed that in the case of road networks, low-rank matrix completion was able to estimate missing values with better accuracy in comparison with other imputation methods (Asif et al., 2013, Mardani and Giannakis, 2013). Roughan et al. (2012) applied rank regularization along with matrix factorization to recover a traffic matrix from partial traffic measurements and accordingly proposed the sparsity regularized matrix factorization (SRMF) algorithm. However, most of the above studies are unable to deal with the large-scale data recovery problem efficiently or effectively. Nie et al. (2012) proposed an efficient method for completion problems that solved a large-scale matrix with promising performance. In this paper, the adopted Schatten $p$-norm algorithm is characterized by the capacity for solving large-scale matrix completion problems.

3. METHODOLOGY

This section focuses on the methodology of network-wide traffic speed estimation for addressing the challenges of processing a rich source of emerging trajectories data generated by ride-sourcing vehicles equipped with smartphone GPS units. The proposed framework is illustrated in Fig. 1, and is comprised of three major components: map matching, traffic data recovery, and traffic state estimation. The mathematical notation used in Section 3 is summarized in Table 2.

![Fig. 1. The framework of trajectories-based urban road network-wide traffic speed estimation.](image_url)

3.1 Cell-Based Map Matching

This section proposes an effective and efficient cell-based map-matching method for large-scale high-frequency GPS-based vehicle traces. The following terminology is used in this section:

- **Link**: road segment between two nodes in the road network;
- **Heading angle**: the clockwise angle between the vehicle speed and north direction;
- **Link direction**: the clockwise angle between the line connecting the two adjacent points of the road and the geographic north.
Table 2. Notation and Definition

<table>
<thead>
<tr>
<th>Notation</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>B</strong></td>
<td>Binary matrix</td>
</tr>
<tr>
<td><em>d</em></td>
<td>Distance between the record point and candidate link</td>
</tr>
<tr>
<td><strong>G</strong></td>
<td>Base matrix</td>
</tr>
<tr>
<td>G</td>
<td>GPS sequence</td>
</tr>
<tr>
<td><strong>H</strong></td>
<td>Mask matrix</td>
</tr>
<tr>
<td><em>l</em></td>
<td>Road link in the urban network</td>
</tr>
<tr>
<td><em>L_c</em></td>
<td>Set of candidate links</td>
</tr>
<tr>
<td><strong>M</strong></td>
<td>Original matrix</td>
</tr>
<tr>
<td>$|M|_{S_p}$</td>
<td>Schatten $p$-norm of matrix $M$</td>
</tr>
<tr>
<td>$|M|_*$</td>
<td>Nuclear norm of matrix $M$</td>
</tr>
<tr>
<td>$M_{11}$</td>
<td>Observed values in matrix $M$</td>
</tr>
<tr>
<td>$N_c$</td>
<td>Number of candidate links</td>
</tr>
<tr>
<td>$N_s$</td>
<td>Number of point coordinate pairs of a road location sequence</td>
</tr>
<tr>
<td>$N_r$</td>
<td>Number of point coordinate pairs in the GPS record sequence</td>
</tr>
<tr>
<td><em>p</em></td>
<td>Parameter of the Schatten $p$-norm method</td>
</tr>
<tr>
<td><strong>T</strong></td>
<td>Matrix with known data sampled from the observed entries</td>
</tr>
<tr>
<td><strong>R</strong></td>
<td>Road location sequence</td>
</tr>
<tr>
<td><strong>r</strong></td>
<td>GPS record</td>
</tr>
<tr>
<td><strong>S</strong></td>
<td>Location of a GPS record</td>
</tr>
<tr>
<td><em>t</em></td>
<td>Time</td>
</tr>
<tr>
<td><strong>v</strong></td>
<td>Speed</td>
</tr>
<tr>
<td><strong>X</strong></td>
<td>Speed matrix recovered by the Schatten $p$-norm method</td>
</tr>
<tr>
<td><em>x</em>_i</td>
<td>Longitude of GPS record $i$</td>
</tr>
<tr>
<td><em>y</em>_i</td>
<td>Latitude of GPS record $i$</td>
</tr>
<tr>
<td><em>α</em></td>
<td>Heading angle of the trajectory velocity</td>
</tr>
<tr>
<td><em>θ</em></td>
<td>Heading angle of the road link</td>
</tr>
<tr>
<td><em>γ</em></td>
<td>The angle between the trajectory velocity and direction of the candidate link</td>
</tr>
<tr>
<td><strong>σ</strong></td>
<td>Singular value of $M$</td>
</tr>
<tr>
<td><strong>Ω</strong></td>
<td>Set of observed entries</td>
</tr>
<tr>
<td>$Tr(\cdot)$</td>
<td>Trace of matrix</td>
</tr>
<tr>
<td>$\circ$</td>
<td>Hadamard product</td>
</tr>
<tr>
<td>$\Delta(\cdot)$</td>
<td>Difference function</td>
</tr>
<tr>
<td><strong>Λ</strong></td>
<td>Set of links in the traffic network</td>
</tr>
</tbody>
</table>
Fig. 2. An illustration of the cell-based map-matching method.

GPS records are used to determine vehicle locations at a given moment in time, and have the following essential attributes: latitude, longitude, timestamp, heading ad instantaneous speed. The location and timestamp of GPS record \( r_i \) are denoted by \( S_i \) and \( t_i \), respectively, with \( S_i \) being the location at time \( t_i \), and containing records for longitude \( x_i \) and latitude \( y_i \), such that \( S_i = (x_i, y_i) \). Due to the satellite signal blockage and multipath effects, the positioning accuracy of \( S_i \) in an urban area is in the range of 0-40 m at the 95% confidence level (Quddus et al., 2007).

Consider a sequence of GPS records for a vehicle denoted by \( \{(S_1, t_1), \ldots, (S_N, t_N)\} \), where \( N \) is the number of point coordinate pairs in the GPS record sequence. The instantaneous speed is calculated by differencing consecutive pairs of the record sequence, denoted by \( v_j = \frac{\Delta(S_i)}{t_{i+1} - t_i} \), where \( \Delta(\cdot) \) is the difference method. The heading angle of the trajectory velocity is \( \alpha_i = \frac{\Delta(x_{i+1})}{\Delta(y_{i+1})} \) (see Fig. 2).

Digital maps are used to organize the underlying location information of each road across the network. A location sequence \( \{(x_1, y_1), \ldots, (x_{N_s}, y_{N_s})\} \) contains the latitude/longitude of the starting point and endpoint, where \( N_s \) is the number of point coordinate pairs. All of the turning points of the road can be obtained from the GIS data, while heading angles can be calculated using the expression \( \theta_i = \frac{\Delta(y_{i+1})}{\Delta(x_{i+1})} \).
Based on the parameter notation of the map-matching problem, the mapping-to-cells method is adopted to improve the map-matching efficiency in an urban area. The cell-based map-matching process consists of four steps: initialization, trajectories tracking to cells, candidate link determination, and matching result generation.

**Step 1 (Initialization):** Calculate the instantaneous speed and heading direction of each GPS record, delete invalid records, then calculate the direction of each link. Since drivers' abnormal driving behavior (e.g., parking, and U-turn) and data errors lead to the issue that the speed may not represent the instantaneous traffic state, we denote the upper/lower bounds of vehicular speed as $v_{\text{max}}$ and $v_{\text{min}}$. In this paper, the values of $v_{\text{max}}$ and $v_{\text{min}}$ are set to be 70 km/h and 1 km/h, respectively. If the travel speed violates speed bounds, i.e., $v_j > v_{\text{max}}$ or $v_j < v_{\text{min}}$, then $v_j = \text{null}$, that is, delete the invalid speed.

**Step 2 (Trajectories tracking to cells):** Construct the cell index for all GPS records using their latitude and longitude information according to the spatial range of each cell. To simplify the map-matching process, GPS records are assigned to cells. The index of a cell is denoted as $\text{row} \times 100 + \text{col}$, where $\text{row}$ and $\text{col}$ are the row and column in which the cell is located. **Step 3 (Candidate link determination):** Assign cells to links. In the case of a road, the line connecting any two adjacent points in the location sequence \( \{(x_1, y_1), ..., (x_N, y_N)\} \) passes through the cells, which are matched to links while keeping track of their respective indices. A special case might occur when road network nodes are located within a cell. Therefore, multiple links pass the same cell, and the cell pertains to multiple links. After picking out these cells and recording their cell indices, the set of candidate links for GPS record \( \{S_i, t_i\} \) is denoted as \( \mathcal{L}_c = \{l_1, ..., l_{N_c}\} \), where \( N_c \) is the number of candidate links. For instance, as shown in Fig. 2, Cell_A and Cell_B pertain to multiple links.

**Step 4 (Applying K-means clustering algorithm to distinguish candidate links):** Calculate the angle $\gamma_j$ between the trajectory velocity and the direction of candidate link $l_j$. The distance $d_j$ between the record point and candidate link $l_j$ is also calculated. For all candidate links, the one of which the angle reaches the minimum is named the generation link. However, if parallel links are located in the same cell, with the angle value not being sufficient to distinguish the different links. In this case, if the difference between any two candidate links is smaller than 10 degrees, these will be as parallel. We apply the $K$-means method to classify the GPS traces only in the situation of two parallel roads crossing the same cell. Moreover, the algorithm can solve the problem of global GPS migration. The inputs of the $K$-means algorithm are as follows: the longitude and latitude of GPS records, the value of $K$, and the initialization clustering center. The value of $K$ is determined by the number of candidate links in the cell. And the initialization clustering centers are randomly selected from the points on the candidate links. The distance of the $K$-means algorithm is Euclidean distance. The output of the algorithm is the results of the record belonging to each cluster. After $K$-means clustering, the records are clustered into $K$ categories. Then the categories are matched to the
associated links by minimizing the summation of the vertical distance from each cluster center to the candidate links.

**Step 5 (Matching result generation):** The matching result is generated according to the output of the algorithm.

### 3.2 Schatten $p$-Norm Minimization for Missing Value Recovery

The matrix completion method is efficient enough to overcome the problem of missing data, which recovers a low-rank matrix with a fraction of its entries arbitrarily corrupted. In this paper, we apply a fast-converging matrix completion method based on Schatten $p$-norm minimization to recover the large-scale network-wide traffic speed matrix.

#### 3.2.1 Problem Formulation

A spatio-temporal speed matrix is a representation of the traffic speed at the link in the traffic network or a cell on the road. Considering a network with $n$ links (or a road with $n$ cells) and $m$ time intervals, the speed matrix can be formed as an $n \times m$ matrix. The entry denotes the traffic speed of the $i$th link $l_i$ at time $t_j$, where $i = 1, 2, ..., n$, $j = 1, 2, ..., m$. Note that columns of the traffic matrix represent the links’ time-varying traffic speed, while the rows represent traffic speed over all links at the sampling time.

The original speed matrix with missing entries is calculated after applying the map-matching method to the GPS records. The harmonic mean (Salamanis et al., 2016) is preferred over the arithmetic mean, because it eliminates outliers in the form of too high speeds, thus resulting in the more accurate reconstruction of speed. Since the trajectories have been tracked to cells and links, the speed values of trajectories are aggregated into the average speed of the cells and links at time interval $t$. Supposing $n$ raw speed values are recorded at time interval $t$, we have

$$
\hat{v}_{i,j} = \frac{1}{n} \sum_{j=1}^{n} v_j
$$

where $\hat{v}_{i,j}$ is the harmonic mean speed of link $l_i$ at time $t$.

We use the binary matrix $B \in \mathbb{R}^{n \times m}$ to indicate whether the entries of speed matrix $M$ are missing. $B$ is defined by $B_{ij} = 0$ if $M_{ij}$ is missing, otherwise, $B_{ij} = 1$. Denote $M_\Omega = \{ M_{ij} \mid (i, j) \in \Omega \}$, $\Omega$ is the set of observed entries. Suppose that the observed values are given as $M_\Omega$ in matrix $M$. The matrix completion task is to predict unobserved values in matrix $M$, and satisfy the following constraint:

$$
X_{ij} = T_{ij}, \quad (i, j) \in \Omega
$$

(2)
where \( T_{ij} \) are the known data sampled from the entry set \( \Omega \). Suppose \( X \in R^{m \times n} (n \geq m) \), \( X \) is the recovered speed matrix and using the matrix form, problem (1) can be concisely written as

\[
X \circ B = M
\]

(3)

where operator \( \circ \) is the Hadamard product; \( M \in R^{m \times n} \); \( M_{ij} = T_{ij} \) if \((i, j) \in \Omega\); otherwise, \( M_{ij} = 0 \).

3.2.2 Traffic Speed Matrix Completion Based on Schatten P-Norm

Traffic data usually exhibit strong spatio-temporal correlations. For example, temporally adjacent matrix elements have similar values. Spatial affinity refers to the similarity between link \( l_i \) and the link \( l_j \), which expresses there are some columns close to each other. In this paper, missing data recovery is achieved by Schatten \( p \)-norm algorithm, a widely-used matrix completion method.

For matrix \( M \), the \( i \)th column, \( j \)th row, and \((i, j)\)th entry of \( M \) are denoted by \( m_i \), \( m_j \), and \( M_{ij} \), respectively. For vector \( v \), the \( i \)th entry of \( v \) is denoted by \( v_i \). The extended Schatten \( p \)-norm (0 < \( p \) < \( \infty \)) of \( M \in R^{m \times n} \) is defined as:

\[
\| M \|_{S_p} = \left( \sum_{i=1}^{\min\{m, n\}} \sigma_i^p \right)^{\frac{1}{p}} = \left( \text{Tr}\left( (M^T M)^{\frac{p}{2}} \right) \right)^{\frac{1}{p}}
\]

(4)

where \( \sigma_i \) is the \( i \)th singular value of \( M \). \( \text{Tr}(\cdot) \) is the trace of a matrix. Thus, the Schatten \( p \)-norm of \( M \in R^{m \times n} \) to power \( p \) is given by

\[
\| M \|_p^p = \sum_{i=1}^{\min\{m, n\}} \sigma_i^p
\]

(5)

When \( p = 1 \), the Schatten 1-norm is the trace norm or nuclear norm, which is usually denoted as \( \| M \|_1 \). When \( p \to 0 \), the extended Schatten \( p \)-norm of \( M \) approximates the rank of \( M \). If we define \( 0^0 = 0 \), then when \( p = 0 \), Eq. (4) equals the rank of \( M \).

Suppose that the observed values are defined as \( M_{ij} \) in matrix \( M \), the matrix completion task is to estimate the unobserved values in \( M \).

\[
\min_{X \in R^{m \times n}} \| X \|_{S_p}^p \quad \text{s.t.} \quad X_{ij} = T_{ij} \quad (i, j) \in \Omega
\]

(6)

where \( T_{ij} \) are the known data sampled from the entries set \( \Omega \).

Suppose \( X \in R^{m \times n} (n \geq m) \), problem (5) can be concisely written in the matrix form as

\[
\min_{X \in R^{m \times n}} \| X \|_p^p \quad \text{s.t.} \quad X \circ H = M
\]

(7)

where \( \circ \) is the Hadamard product; \( M \in R^{m \times n} \), \( M_{ij} = T_{ij} \) for \((i, j) \in \Omega\); and \( M_{ij} = 0 \) for other \((i, j)\); \( H \in R^{m \times n} \), \( H_{ij} = 1 \) for \((i, j) \in \Omega\) and \( H_{ij} = 0 \) for others \((i, j) \notin \Omega\).
The optimization algorithm is derived from the Lagrangian function of problem (6), given by

\[ L(X, \Lambda) = Tr\left( X^T X \right)^{\frac{p}{2}} - Tr\left( \Lambda^T (X \circ H - M) \right) \]  

(8)

By taking the derivative of \( L(X, \Lambda) \) with respect to \( X \) and setting the derivative to be zero, we have

\[ \frac{\partial L(X, \Lambda)}{\partial X} = 2XD - H \circ \Lambda = 0 \]  

(9)

where \( D \) is defined as \( D = \frac{p}{2} \left( X^T X \right)^{\frac{p-2}{2}} \).

By solving Eq. (8), we obtain

\[ X = \frac{1}{2} (H \circ \Lambda) D^{-1} \]  

(10)

Algorithm: An Efficient Iterative Algorithm for Solving Problem (6).

**Input:** Data \( T_{ij} \) for all \((i,j) \in \Omega\), and \( 0 \leq p \leq 2 \).

**Output:** \( H \in \mathbb{R}^{n \times m} \).

**Define** \( M \in \mathbb{R}^{n \times m} (n \geq m) \), where \( M_{ij} = T_{ij} \) for \((i,j) \in \Omega\), and \( M_{ij} = 0 \) for others \((i,j) \notin \Omega\);

**Define** \( H \in \mathbb{R}^{n \times m} \), where \( H_{ij} = 1 \) for \((i,j) \in \Omega\), and \( H_{ij} = 0 \) for others \((i,j) \notin \Omega\);

**Set** \( s = 0 \);

**Initialize** \( D_s \in \mathbb{R}^{n \times m} (n \geq m) \) as \( D_s = \frac{p}{2} \left( M^T M \right)^{\frac{p-2}{2}} \).

**Repeat**

1. Calculate \( X_{s+1} = \frac{1}{2} (H \circ \Lambda_s) D_s^{-1} \), where the \( i \)th row is calculated by \( \Lambda_s^i = 2m^i \left( H^i D_s^i H^i \right)^{-1} \);

2. Calculate \( D_s = \frac{p}{2} \left( X_{s+1}^T X_{s+1} \right)^{\frac{p-2}{2}} \);

3. \( s = s + 1 \).

**Until** Convergence criteria are satisfied.

### 3.3 Measures of Effectiveness

We adopt four widely used measures of effectiveness to evaluate the accuracy of matrix completion as follows:

**MAE** (mean absolute error) is an average of the absolute errors on the missing values after the completion, which can be calculated as
\[
\text{MAE} = \frac{\sum_{i,j,H_y=0} |X_{ij} - \hat{X}_{ij}|}{T}
\]  

(11)

**NMAE** (normalized mean absolute error) is an average of the absolute errors on the missing values after the interpolation, which can be calculated as

\[
\text{NMAE} = \frac{\sum_{i,j,H_y=0} |X_{ij} - \hat{X}_{ij}|}{\sum_{i,j,H_y=0} |X_{ij}|}
\]  

(12)

**RMSE** (root mean squared error) represents the standard deviation of the errors, which can be calculated as

\[
\text{RMSE} = \sqrt{\frac{1}{|\Phi|} \sum_{i,j,H_y=0} |X_{ij} - \hat{X}_{ij}|^2}
\]  

(13)

**NRMSE** (normalized root mean squared error) represents the standard deviation of the errors, which can be calculated as

\[
\text{NRMSE} = \sqrt{\frac{1}{|\Phi|} \sum_{i,j,H_y=0} |X_{ij} - \hat{X}_{ij}|^2} \frac{1}{T \sum_{i,j} |X_{ij}|}
\]  

(14)

In Eqs. (10)-(13), \( \hat{X} \) is the recovered speed matrix, \( H_y = 0 \) means \( X_y \) is missing, and only errors on the missing values are calculated. \( |\Phi| \) is the number of missing values. \( T \) denotes the number of entries of \( X \).

4. **RIDE-SOURCING VEHICLE GPS TRAJECTORIES**

In this section, we describe a dataset of GPS traces from ride-sourcing vehicles operating on a large-scale road network that we use to estimate network-wide traffic speeds. The probe vehicle dataset was obtained from DiDi Chuxing Technology Co., the largest on-demand ride services platform in China, for one month between November 1, 2016, and November 30, 2016. The GPS traces are collected by ride-sourcing vehicles that travel through the urban road network and upload their status information to the data center every 3 s, which can be regarded as high temporal resolution. An individual GPS record includes a real-time timestamp, order ID, driver ID, longitude, and latitude. Any identifiable information of drivers and passengers was anonymized to avoid any privacy issues.

The dataset covers approximately one quarter of the city area of Chengdu, China between 104.03°E and 104.13°E in longitude and 30.64°N and 30.74°N in latitude. There were over 6.1 million successful orders during the study period. Fig. 3(a) presents the temporal variation in the
mean number of the GPS data points on weekdays and weekends during November 2016, showing that there are over 2 million data records per hour during the daytime and that the peak period of recorded orders varies from weekday to weekend. Fig. 3(b) shows daily profiles of all the probe data during the month. The number of hourly GPS records is highest between 14:00 and 15:00 and Friday and Saturday are typically the busiest days.

As shown in Fig. 4, the study urban area is partitioned into 100 × 100 cells. Each cell is 0.001 in longitude and 0.001 in latitude (approximately 110 × 110 m²). Fig. 4(a) presents the spatial distributions and frequencies of cell-based GPS records during a typical 5-minute time period. In Fig. 4(b), the 3 s GPS data are resampled into time intervals of 5 min, 10 min, and 15 min, and the hourly proportion of missing data in the speed matrix is shown for the period between 0:00 to 23:00. Fig. 5 shows the cell-average traffic speed during two peak periods.

In summary, the descriptive statistics and empirical traffic dynamics are shown for a better understanding of temporal and spatial distributions of ride-sourcing vehicles, trajectory data on the road network, traffic speed maps, etc. These empirical results based on massive GPS traces data of on-demand ride services provide valuable information on characteristics of traffic patterns of a real-world large-scale urban road network, which is useful for the traffic manager to easily visualize traffic congestion patterns and devise traffic management strategies for better decision support.
5. RESULTS

5.1 Map Matching

To accurately estimate traffic speeds across the road network, the GPS traces first need to be mapped onto the road network. The study site contains 2,702 roads and 5,690 links, further characteristics of the road network are shown in Table 3. Fig. 6(a) shows all GPS points from 8:00 to 9:00 on November 1, 2016, plotted on top of the road links and Fig. 6(b) shows different grades of roads in the network. The GPS traces are mapped to links with the cell-based map-matching method. After mapping GPS traces to links, the instantaneous velocity records are aggregated for 5-minute time intervals, and the harmonic mean is calculated for each link. Fig. 7 presents the spatio-temporal traffic diagrams for the two directions of the 2nd ring road on November 1, 2016, where
the x-axis represents the indices of cells in order and the y-axis represents the speed values of cells.

**Table 3.** Network characteristics of the study site

<table>
<thead>
<tr>
<th>Network characteristics</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Nodes</td>
<td>1,527</td>
</tr>
<tr>
<td>Roads</td>
<td>2,702</td>
</tr>
<tr>
<td>Links</td>
<td>5,690</td>
</tr>
<tr>
<td>Minimal road length (km)</td>
<td>2.900</td>
</tr>
<tr>
<td>Maximal road length (km)</td>
<td>3.430</td>
</tr>
<tr>
<td>Average road length (km)</td>
<td>0.399</td>
</tr>
<tr>
<td>Minimal link length (km)</td>
<td>0.015</td>
</tr>
<tr>
<td>Maximal link length (km)</td>
<td>2.131</td>
</tr>
<tr>
<td>Average link length (km)</td>
<td>0.113</td>
</tr>
<tr>
<td>Minimal links per road</td>
<td>1</td>
</tr>
<tr>
<td>Maximal links per road</td>
<td>28</td>
</tr>
<tr>
<td>Average links per road</td>
<td>2.105</td>
</tr>
</tbody>
</table>

**(a)** Fig. 6. Ride-sourcing vehicle GPS records in the urban area of Chengdu, China. (a) Scatter plot of all GPS points (8:00-9:00, November 1, 2016); (b) Road network in Chengdu.
Fig. 7. Spatio-temporal traffic diagrams of the 2nd ring road on November 1, 2016. (a) Clockwise direction; (b) Anti-clockwise direction.

5.2 Traffic Data Recovery

Our experiments are performed on two matrices. The first matrix $M_1$ is the 2nd ring road speed matrix with 288 rows, representing $(12 \times 24)$ 5-minute intervals over one day, and 98 columns, representing the number of cells on the road. The second matrix $M_2$ is a one-week speed matrix selected from the original 10-minute speed matrix of primary road links, and has 588 rows, representing $(6 \times 14 \times 7 = 588)$ 10-minute time intervals, and 114 columns, representing links that do not have any missing entries. In many machine learning problems, $n$-fold ($n$-split) cross-validation is used to verify the predictive performance of a model and to evaluate the effect of different training and test sets. However, spatio-temporal matrix completion problems are slightly different in the sense that the datasets are temporally correlated. Here, we select the second matrix mentioned above as the criteria when calculating relative errors to verify the efficiency of the method. Observed data are chosen randomly, and the speed matrix is generated at certain observation rates to verify the accuracy of the method. The observation rate is defined as the observed rate of the original matrix, which is the proportion of known data entries to the total entries.

In the experiments on the matrix $M_1$, we present the numerical results for solving matrix completion with $0 < p < 2$, and compare the performance for different values of $p$ between 0.1 and 2, with an increasing step of 0.1. The number of iterations is set to 50 to ensure convergence. The convergence criterion is that the difference between two iterations is less than 0.0001. A smaller value of either NMAE or NRMSE indicates a better recovery of $X$. Fig. 8 presents the matrix recovery performance of the method with different values of $p$ when the observation rate is 70%. As can be seen, the recovery is more precise when $p$ is smaller than 1. Fig. 9 presents a comparison between the actual speed and estimated speed of 2nd ring road with the observation rates of 50% and 10%, respectively.
Fig. 8. Matrix recovery performance with different values of $p$ (observation rate = 70%).

Fig. 9. Comparison between actual speeds and estimated speeds. (a) Observation rate 50%; (b) Observation rate 10%.

To verify the algorithm performance in the case of continuous data loss (such as several hours), the starting position of the continuous missing data in the speed matrix is randomly selected in the first dataset. For example, for a road link with 24-hour data, a 25% continuous data loss means there is a 6-hour gap in the speed series of this link. Fig. 10 shows mean error and standard deviation of 10 random repeats. The standard deviation is less than 0.011 and indicates that the algorithm is robust to the point at which the data loss period begins. The experimental results show that the
algorithm the performance of the algorithm is significantly worse when the observation rate is less than 10%. In general, the performance of the algorithm in the case of continuous data loss is worse than in the case of randomly missing points. When the observation rate is less than 50%, the possibility of there being missing data for all links at a certain time period increases, which reduces the accuracy of the algorithm.

Fig. 10. Comparison of the matrix recovery performance on continuous data loss. (a) NMAE; (b) NRMSE. The error bars represent the standard deviation of the results for 10 randomly selected starting positions.

In the experiments performed on the second matrix $\mathbf{M}_2$, we demonstrate the effectiveness of the Schatten $p$-norm algorithm by comparing its performance with the following benchmark matrix completion algorithms:

**K Nearest Neighbor (KNN)** fills in missing values with adjacent data. The missing data in the matrix is replaced with a weighted mean of the k nearest neighbor columns in the algorithm. The weights are inversely proportional to the distances from the neighboring columns.

**Nonnegative Matrix Factorization (NMF)** is formulated as an alternative nonnegative least squares problem in the presence of missing entries (Mao and Saul, 2004). The NMF problem is described as follows: Given matrix $\mathbf{M} \in \mathbb{R}^{m \times n}$, find the nonnegative matrices $\mathbf{G} \in \mathbb{R}^{m \times r}$ and $\mathbf{Z} \in \mathbb{R}^{r \times n}$, which approximate $\mathbf{M} \approx \mathbf{GZ}$. Usually, the value of $r$ is smaller than $n$, with the constraint $(m+n) \cdot r < m \cdot n$. The column vectors of the original matrix $\mathbf{M}$ are the weighted sum of all the column vectors of the left matrix $\mathbf{G}$, and the weight coefficient is the element of the column vectors corresponding to the right matrix $\mathbf{Z}$. Therefore, $\mathbf{G}$ is the base matrix. Since the base matrix $\mathbf{G}$ and coefficient matrix $\mathbf{Z}$ are simultaneously determined by NMF, coefficient matrix $\mathbf{Z}$ is not a projection of data matrix $\mathbf{M}$ on $\mathbf{G}$. NMF implements nonlinear dimension reduction performs well in image analysis and processing problems due to its interpretation of the local characteristics of an image.

**Sparsity Regularized Matrix Factorization (SRMF)** utilizes the low-rank structure and
spatio-temporal properties to recover missing data (Roughan et al., 2012). It performs alternating constrained least squares to factorize a matrix, which is subject to spatial and temporal constraint matrices. It has the advantage of the sparsity regularized singular value decomposition (SVD), but is more general, such that other objectives in the traffic speed matrix approximation/interpolation algorithm may be expressed through different constraints. Standard algorithms for calculating SVD assume that the original matrix is wholly known. The SRMF achieves SVD-like factorization that allows matrix with missing data and satisfies the measurement equations. The algorithm can utilize the spatio-temporal properties of the traffic speed matrix with spatial and temporal constraint matrices.

Fig. 11. Comparison of the matrix recovery performance. (a) NMAE; (b) NRMSE.

Fig. 12. Matrix recovery performance for different road types. (a) MAE; (b) RMSE.
As shown in Fig. 11, the Schatten $p$-norm matrix completion algorithm outperforms the other benchmark algorithms over the whole range of missing data rates. The value of $K$ in the KNN algorithm is set to 6 by grid search. For the high data loss, the performance of NMF is significantly worse than the Schatten $p$-norm method. Since the data of speed matrices are insufficient, it’s difficult to extract the statistic features through such datasets with many missing entries. Therefore, higher data loss makes it more difficult to reconstruct traffic states. After demonstrating the effectiveness of the Schatten $p$-norm method, we test its urban network-wide performance on 114 primary links, 57 secondary links, and 62 tertiary links without missing entries. Fig. 12 presents the matrix recovery performance of the method with different missing rates on various grades of links. The method performs better on the primary links than on the other categories of links because of the larger amount of data and more obvious statistical features. Table 4 and Table 5 present the matrix recovery performance with different values of $p$, showing that performance is typically best when $p$ approaches 0.1.

**Table 4. MAE (km/h) of the network-wide matrix recovery**

<table>
<thead>
<tr>
<th>Value of $p$</th>
<th>10%</th>
<th>30%</th>
<th>50%</th>
<th>70%</th>
<th>90%</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>3.09</td>
<td>3.25</td>
<td>4.26</td>
<td>4.74</td>
<td>5.96</td>
</tr>
<tr>
<td>0.9</td>
<td>2.90</td>
<td>3.08</td>
<td>3.95</td>
<td>4.51</td>
<td>5.55</td>
</tr>
<tr>
<td>0.8</td>
<td>2.78</td>
<td>2.94</td>
<td>3.78</td>
<td>4.23</td>
<td>5.52</td>
</tr>
<tr>
<td>0.7</td>
<td>2.69</td>
<td>2.85</td>
<td>3.67</td>
<td>4.21</td>
<td>5.73</td>
</tr>
<tr>
<td>0.6</td>
<td>2.57</td>
<td>2.72</td>
<td>3.51</td>
<td>4.20</td>
<td>5.95</td>
</tr>
<tr>
<td>0.5</td>
<td>2.48</td>
<td>2.62</td>
<td>3.45</td>
<td>4.26</td>
<td>6.09</td>
</tr>
<tr>
<td>0.4</td>
<td>2.38</td>
<td>2.56</td>
<td>3.40</td>
<td>4.38</td>
<td>6.13</td>
</tr>
<tr>
<td>0.3</td>
<td>2.31</td>
<td>2.51</td>
<td>3.36</td>
<td>4.60</td>
<td>6.48</td>
</tr>
<tr>
<td>0.2</td>
<td>2.27</td>
<td>2.47</td>
<td>3.38</td>
<td>4.88</td>
<td>6.69</td>
</tr>
<tr>
<td>0.1</td>
<td>2.23</td>
<td>2.42</td>
<td>3.45</td>
<td>5.07</td>
<td>6.98</td>
</tr>
</tbody>
</table>

**Table 5. RMSE (km/h) of the network-wide matrix recovery**

<table>
<thead>
<tr>
<th>Value of $p$</th>
<th>10%</th>
<th>30%</th>
<th>50%</th>
<th>70%</th>
<th>90%</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>4.70</td>
<td>5.35</td>
<td>6.04</td>
<td>6.66</td>
<td>7.78</td>
</tr>
<tr>
<td>0.9</td>
<td>4.48</td>
<td>5.23</td>
<td>5.82</td>
<td>6.49</td>
<td>7.32</td>
</tr>
<tr>
<td>0.8</td>
<td>4.49</td>
<td>5.15</td>
<td>5.72</td>
<td>6.32</td>
<td>7.36</td>
</tr>
<tr>
<td>0.7</td>
<td>4.47</td>
<td>5.12</td>
<td>5.67</td>
<td>6.38</td>
<td>7.66</td>
</tr>
<tr>
<td>0.6</td>
<td>4.34</td>
<td>5.01</td>
<td>5.56</td>
<td>6.39</td>
<td>7.96</td>
</tr>
<tr>
<td>0.5</td>
<td>4.33</td>
<td>4.88</td>
<td>5.55</td>
<td>6.57</td>
<td>8.16</td>
</tr>
<tr>
<td>0.4</td>
<td>4.24</td>
<td>4.91</td>
<td>5.55</td>
<td>6.85</td>
<td>8.22</td>
</tr>
<tr>
<td>0.3</td>
<td>4.16</td>
<td>4.91</td>
<td>5.59</td>
<td>7.26</td>
<td>8.63</td>
</tr>
<tr>
<td>0.2</td>
<td>4.19</td>
<td>4.89</td>
<td>5.72</td>
<td>7.78</td>
<td>8.90</td>
</tr>
<tr>
<td>0.1</td>
<td>4.23</td>
<td>4.88</td>
<td>5.96</td>
<td>8.13</td>
<td>9.27</td>
</tr>
</tbody>
</table>
5.3 Stochastic Congestion Map

After demonstrating the effectiveness of the Schatten $p$-norm method, we move onto reconstructing speed matrices of the urban network with 1,189 primary links, 602 secondary links, and 662 tertiary links for the entire one month study period. Speed information of this type could be used by traffic managers to develop strategies for traffic control and transportation planning in addition to visualization of traffic conditions. The stochastic congestion map is calculated at the network level to visualize traffic dynamics.

For traffic managers and travelers, a stochastic network congestion map is intuitive and useful. Both Google Maps and Baidu Maps can show the short-term predictions of traffic conditions, which inform route planning and congestion avoidance. Similarly, we calculate the stochastic congestion map at the network level to monitor and visualize traffic dynamics. In contrast to the representation on Google Maps and Baidu Maps, our stochastic congestion map shows the road segments with a high probability of congestion during weekdays and weekends in the long term. The congestion probability is defined as

$$ p_i = \frac{N_{e(<v_c)}}{N_e}, $$

where $p_i$ is the probability of congestion of link $i$, $v_c$ is the critical speed, set as 20 km/h, $N_e$ is the number of entries at specified time intervals, $N_{e(<v_c)}$ is the number of entries if $v < v_c$ at specified time intervals. We graphically represent $p_i$ values categorized into five levels: (0, 0.2], (0.2, 0.4], (0.4, 0.6], (0.6, 0.8], and (0.8, 1.0].

Fig. 13 and Fig. 14 shows the probability of congestion of each link when the probability of congestion is more than 0.4. Fig. 13 presents the calculated stochastic congestion maps during peak hours (8:00-9:00 and 18:00-19:00) on the weekday in the clockwise (toward the northeast and southeast) and anti-clockwise (toward the northwest and southwest) directions. Fig. 14 presents the calculated stochastic congestion maps during peak hours (8:00-9:00 and 18:00-19:00) on the weekend in the clockwise and anti-clockwise directions. In both figures, the data represents the one-month speed matrices of primary, secondary, and tertiary links after data recovery. Each line represents a link, and the color represents the congestion probability of the link. To highlight the links with a higher probability of congestion, we only plot links whose congestion probability is higher than 0.4. Most of the congested links are located near intersections, indicating that delays at intersections account for a large proportion of the total delay. The results show that the location of congested links varies spatially between weekdays and weekends, time of day, and traffic directions.
Fig. 13. Stochastic congestion maps on the weekday. (a) AM peak, clockwise; (b) AM peak, anti-clockwise; (c) PM peak, clockwise; (d) PM peak, anti-clockwise.
6. CONCLUSIONS

The monitoring, estimation, and prediction of urban traffic conditions is vital to enable real-time traffic management. The emergence of new technologies and new mobility services enables the extensive collection of traffic data across large areas. To the best of our knowledge, this is one of the first studies to process a large amount of high-resolution ride-sourcing GPS trajectory data to estimate the large-scale network-wide traffic speed with a matrix completion approach in the era of shared mobility. In this paper, we employed an efficient cell-based map-matching method using 3-second resolution GPS traces as inputs. The map-matching method can distinguish different links when the cell is located at intersections, even though parallel links are located in the same cell. To overcome the common problem of missing data in both the spatial and temporal dimensions, we adopt the Schatten $p$-norm matrix completion algorithm to recover the missing data. This algorithm outperforms several benchmark methods across the range of missing data rates and it performs very well with less than 6 km/h of mean absolute errors (MAE) even at an observation data rate of 10%. For the case of data loss across continuous time periods, the accuracy of the algorithm is reduced. To monitor and visualize traffic dynamics, we calculate the stochastic congestion map at the network level, which shows the congestion probability of road segments with different grades.

The chosen cell size affects the accuracy of the cell-based map-matching method, which could be adjusted to improve accuracy further. Furthermore, the method cannot yet distinguish between different links at interchanges without altitude data. Moreover, while our results suggest that Schatten $p$-norm method is well suited to inferring the nature of complicated traffic patterns, the performance of the matrix completion method is dependent on the intrinsic nature of traffic data to some extent. Therefore, the structure can be further exploited with more abundant data sources. The constructed stochastic congestion map provides a good understanding of network-wide traffic
congestion patterns and supports the decision making of traffic management authorities. In the future, we plan to collect multiple datasets, such as loop detection data and video data, so that traffic speed shall be estimated based on the fusion of various data sources for greater accuracy.

ACKNOWLEDGMENTS

This research is financially supported by the National Key Research and Development Program of China (2018YFB1600900), Zhejiang Provincial Natural Science Foundation of China (LR17E080002), National Natural Science Foundation of China (71922019, 71771198), joint project of National Natural Science Foundation of China and Joint Programming Initiative Urban Europe (NSFC – JPI UE) (‘U-PASS’, 71961137005), Young Elite Scientists Sponsorship Program by CAST (2018QNRC001), and China Postdoctoral Science Foundation (2017M621935). The authors are grateful to DiDi Chuxing (www.xiaojukeji.com) for providing us with some sample data. Data source: DiDi Chuxing GAIA Initiative (https://gaia.didichuxing.com). This work was supported in part by the Zhejiang University/University of Illinois at Urbana-Champaign Institute, and was led by Principal Supervisors Xiqun Chen and Simon Hu.

REFERENCES


rank matrix completion with applications to traffic data imputation. Knowledge-Based Systems, 132, 249-262.


Tao, S., Manolopoulos, V., Rodriguez Duenas, S., & Rusu, A. (2012). Real-time urban traffic state


Tong, D., Merry, C. J., & Coifman, B. (2005). Traffic information deriving using GPS probe vehicle data integrated with GIS. *Center for Urban and Regional Analysis and Department of Geography*. The Ohio State University. Ohio, USA.


