Low-Dimensional Models for Compressed Sensing and Prediction of Large-Scale Traffic Data

The MIT Faculty has made this article openly available. Please share how this access benefits you. Your story matters.

<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>As Published</td>
<td><a href="http://dx.doi.org/10.1109/tits.2015.2411675">http://dx.doi.org/10.1109/tits.2015.2411675</a></td>
</tr>
<tr>
<td>Publisher</td>
<td>Institute of Electrical and Electronics Engineers (IEEE)</td>
</tr>
<tr>
<td>Version</td>
<td>Author's final manuscript</td>
</tr>
<tr>
<td>Accessed</td>
<td>Wed Mar 16 06:51:30 EDT 2016</td>
</tr>
<tr>
<td>Citable Link</td>
<td><a href="http://hdl.handle.net/1721.1/100719">http://hdl.handle.net/1721.1/100719</a></td>
</tr>
<tr>
<td>Terms of Use</td>
<td>Creative Commons Attribution-Noncommercial-Share Alike</td>
</tr>
<tr>
<td>Detailed Terms</td>
<td><a href="http://creativecommons.org/licenses/by-nc-sa/4.0/">http://creativecommons.org/licenses/by-nc-sa/4.0/</a></td>
</tr>
</tbody>
</table>
Low-dimensional Models for Compressed Sensing and Prediction of Large-Scale Traffic Data

Nikola Mitrovic, Student Member, IEEE, Muhammad Tayyab Asif, Student Member, IEEE,
Justin Dauwels, Senior Member, IEEE and Patrick Jaillet

Abstract—Advanced sensing and surveillance technologies often collect traffic information with high temporal and spatial resolutions. The volume of the collected data severely limits the scalability of online traffic operations. To overcome this issue, we propose a low-dimensional network representation where only a subset of road segments is explicitly monitored. Traffic information for the subset of roads is then used to estimate and predict conditions of the entire network. Numerical results show that such approach provides 10 times faster prediction at a loss of performance of 3% and 1% for 5 and 30 minutes prediction horizons, respectively.

Index Terms—Low-dimensional models, traffic prediction.

I. INTRODUCTION

Intelligent Transportation Systems (ITS) collect real-time traffic information from various sources such as probe vehicles, smartphone devices and infrastructure based traffic sensors. With advancements in sensor technology, traffic data (e.g., volume and speed) can be recorded on a large scale and with high temporal resolution. Recorded data is frequently used for historical analysis and traffic management operations such as network monitoring, transportation planning and congestion avoidance applications [1]. These applications heavily rely on fast and accurate assessment of current (estimation) and future (prediction) network states.

To model the road network, existing studies explicitly address each road segment in that network. For large traffic networks and online applications such an approach may not be feasible. To overcome this problem we focus on low-dimensional network models where only a subset of road segments needs to be explicitly monitored.

In this study, we use column-based (CX) matrix decomposition to express the original network in terms of a small subnetwork. We refer to the small subnetwork as the compressed state of the original network. We learn the relationship between compressed and original (uncompressed) network by analyzing the recorded data in offline manner. In this way, we can represent the traffic network as a product of two low-rank matrices: (i) the subnetwork data and (ii) the corresponding relationship matrix. We refer to this as CX compression scheme. The CX compression scheme is a stepping stone to compressed sensing and compressed prediction applications.

In the case of compressed sensing we aim to infer the present state of the entire network from the current traffic state of the subnetwork [2]. We use the training data (collected offline) to infer the relationship matrix. To assess the network state we multiply the data from the subnetwork, obtained from the testing set, with the relationship matrix, inferred from the training data set. Our underlying assumption is that traffic variables often vary smoothly across the traffic network [3].

In the matter of compressed prediction we apply the CX-based method to infer the future state of the network. First, we explicitly predict traffic state for the subnetwork using traditional prediction algorithms. Then we multiply the predicted data of the subnetwork with the relationship matrix, inferred from the training data set. Similarly to compressed sensing we rely on the observation that traffic conditions tend to follow distinct patterns and traffic parameters often vary smoothly [3], [4].

For our analysis, we consider the city-scale traffic network in Singapore, comprising 17,967 road segments. The numerical results show that the proposed methods can infer the current and future states of the network, while substantially improving the processing speed of the underlying modeling algorithm. The reduction in computational time is proportional to the compression ratio, i.e., the ratio of the number of links in the subnetwork and the total number of links.

The paper is structured as follows. In Section II we briefly review relevant literature. In Section III we introduce the column based (CX) matrix decomposition method. In Section IV we present three applications of the CX matrix decomposition methods in the realm of traffic modeling: compression, compressed sensing, and compressed prediction. In Section V we describe the traffic data set analyzed in this paper. In Section VI we provide and discuss results for our experiments. In Section VII we summarize our contributions and suggest topics for future research.

II. RELATED WORK

In this paper we propose a novel low-dimensional network model to improve the scalability of estimation and prediction operations in ITS. Low-dimensional representation of large traffic data sets is traditionally obtained by Principal Component Analysis (PCA) [3], [5], [6]. PCA provides an
effective low-dimensional representation in terms of latent variables and corresponding basis vectors. However, these latent variables are hard to interpret. Moreover, even if we obtain the basis vectors from historical data, we still need to collect data from all sensors during online operation. Due to this reason, PCA is often used for offline operations such as compression and data preprocessing [6].

Simulation (model) and data driven approaches are traditionally used to perform traffic estimation and prediction [7], [8]. Simulation approaches can be used for traffic management operations at various levels of network granularity [7], [9], [10]. For large areas, macroscopic and mesoscopic simulation tools (e.g., DYNAMIT) have been adopted to build custom models, relying on historical speed-density link relationships for that specific network [10]. In recent years, large volumes of collected data have served the relationship matrix \( \mathbf{X} \in \mathbb{R}^{c \times n} \), where \( c \) and \( n \) are the number of columns and rows (time instances) of that matrix \([16]\). Since the column based (CX) method has recently found applications in many fields such as text processing, finance and biology \([13]\)–[15]; it uses only a subset of the columns to reconstruct the entire data matrix. In our previous study, we applied the column and row (CUR) based method to impute a matrix of traffic data from a few columns (links) and few rows (time instances) of that matrix \([16]\). Since the CUR method occasionally requires traffic data for the entire network, it cannot be applied for compressed prediction. The CX-based method instead does not have this requirement. In the following, we will briefly review the CX-based method.

**Definition 1**: Let \( \mathbf{A} \in \mathbb{R}^{m \times n} \) be a given matrix. Let \( \mathbf{C} \in \mathbb{R}^{m \times c} \) be a matrix consisting of \( c \) columns of the matrix \( \mathbf{A} \). The column-based (CX) matrix approximation \( \hat{\mathbf{A}} \) of \( \mathbf{A} \) is defined as \( \hat{\mathbf{A}} = \mathbf{C} \mathbf{X} \), where \( \mathbf{X} \in \mathbb{R}^{c \times n} \) is a matrix that expresses every column of \( \mathbf{A} \) in terms of the basis provided by the columns of \( \mathbf{C} \) \([17]\).

**A. Column selection**

In order to select the best subset of columns, for a given size \( c \), one needs to test all possible combinations. However, the computational complexity of this brute-force approach is \( O(n^c) \) \([13]\). Due to this complexity, testing all possible choices of \( c \) columns is typically not practical. To alleviate this problem, several randomized algorithms have been proposed \([17]\), \([18]\). In our numerical experiments, the SVD sampling method yields the best reconstruction accuracy \([16]\). The SVD sampling algorithm assigns higher selection probability to the road segments with larger traffic speed variations \([16]\). This algorithm calculates the Euclidean norm of top \( k \) right singular vectors of matrix \( \mathbf{A} \in \mathbb{R}^{m \times n} \) to assign a score \( E_{a_i} \) to each column \([17]\). This score \( (E_{a_i}) \) is then converted into a probability \( P_{a_i} \) and further used to sample the columns:

\[
P_{a_i} = \frac{1}{k} E_{a_i} = \frac{1}{k} \sum_{j=1}^{k} v_{ij}^2 \quad \forall \ i = 1, \ldots, n,
\]

where \( v_{ij} \) is the \( i \)-th coordinate of \( j \)-th right singular vector.

**B. Relationship matrix**

For the sampled column matrix \( \mathbf{C} \in \mathbb{R}^{m \times c} \), we compute the relationship matrix \( \mathbf{X} \in \mathbb{R}^{c \times n} \), which will allow us to represent the columns of matrix \( \mathbf{A} \in \mathbb{R}^{m \times n} \) in terms of columns of the matrix \( \mathbf{C} \) \([17]\). The matrix \( \mathbf{X} \) can be regarded as an extrapolation matrix that maps the subnetwork associated with \( \mathbf{C} \) to the entire network represented by \( \mathbf{A} \). For given matrices \( \mathbf{C} \) and \( \mathbf{A} \), we compute the matrix \( \mathbf{X} \) as \( \mathbf{X} = \mathbf{C}^T \mathbf{A} \), where \( \mathbf{C}^+ \) is Moore-Penrose pseudo-inverse of matrix \( \mathbf{C} \) \([19]\).

**IV. CX-BASED METHOD FOR TRAFFIC APPLICATIONS**

In this section, we discuss how CX based method can be used to perform compression, compressed sensing, and compressed prediction of traffic data. For this purpose, we consider the traffic data in the form of a matrix \( \mathbf{A} \in \mathbb{R}^{m \times n} \) where the columns of the matrix \( \{a_i\}_{i=1}^{n} \) contain traffic data from different roads \( \{s_i\}_{i=1}^{p} \). Rows represent time instances \( \{t_i\}_{i=1}^{m} \) at which the traffic data is recorded. Each matrix cell \( (a_{ij}) \) shows the numerical value of an observed traffic variable (e.g., speed, volume) at location \( s_j \) during the interval of time \( (t_i - T, t_i) \) where \( T \) is the sampling period (e.g., 5 or 15 minutes). Therefore, the \( i \)-th row vector \( \mathbf{a}_i = [z(s_1, t_i) \ldots z(s_n, t_i)] \) of \( \mathbf{A} \) contains the traffic state for the entire network at a particular time \( t_i \). Similarly, the \( j \)-th column vector \( \mathbf{a}_j = [z(s_j, t_1) \ldots z(s_j, t_m)]^T \) of \( \mathbf{A} \) contains the observed condition at location \( s_j \) during the entire recording period. Hence, we can write traffic data matrix as \( \mathbf{A} = [\mathbf{a}_1 \ldots \mathbf{a}_n] \). For the sake of simplicity, we use subscripts \( h \), \( p \) and \( f \) in the rest of the paper to denote historical, present and future values, respectively.

**A. Compression**

Suppose that the matrix \( \mathbf{C}_h \) contains the observed traffic states of the \( c \) specific locations in the network, such that \( \{c_1, \ldots, c_c\} \subseteq \{a_1, \ldots, a_n\} \). Then, we can approximate the data matrix \( \mathbf{A}_h \) as \( \hat{\mathbf{A}}_h = \mathbf{C}_h \mathbf{X}_h \), where the matrix \( \mathbf{X}_h \) contains the relationships between the traffic condition at different locations in the network. Hence, instead of storing the large matrix \( \mathbf{A}_h \), we store the two smaller matrices \( \mathbf{C}_h \) and \( \mathbf{X}_h \). The compression ratio (CR) of such low-dimensional approximation is given by:

\[
\text{CR}_h = \frac{mn}{mc + cn}.
\]

Column based (CX) compression scheme leads to simple network representation. Although such compression scheme does not outperform PCA, still it could be useful for online traffic monitoring operations \([16]\). In the following we discuss two attractive applications of CX-based compressed representation, namely compressed sensing and compressed prediction.
B. Compressed sensing

So far, we have assumed that the matrix $X_h$ is stored together with $C_h$ leading to the compression of matrix $A_h$. In this scenario, the matrix $X_h$ is computed for a given data matrix $A_h$ and a column matrix $C_h$. Alternatively, one may precompute a matrix $X_h$ and re-use the same matrix to infer $A$ for any given $C$. Although we still need data from all the links to precompute $X_h$, this operation can easily be performed offline. Hence, during online operations, the system would only require data from a small number of sensors. We refer to this scenario as compressed sensing. It is noteworthy that low-dimensional PCA models can not be used for compressed sensing since PCA requires data from all sensors for both offline and online operations.

The underlying assumption of the proposed method is that the traffic conditions are stationary, so that a fixed matrix $X$ allows us to accurately reconstruct the original data matrix $A$ from $C$ [3], [4]. Therefore, we can estimate the present network state ($\hat{\alpha}_p$) as $\hat{\alpha}_p = c_p^T X_h$ $\forall i = k, \ldots, m$, where $\hat{\alpha}_p \in \mathbb{R}^{1 \times n}$ is a row vector that represents the current state of the entire network for test data ($i = k, \ldots, m$). Row vector $c_p \in \mathbb{R}^{1 \times c}$ contains the information about current traffic conditions at $c$ specific locations in the network. Matrix $X_h$ is the relationship matrix, learned from a training data set. We define the compression ratio for compressed sensing as $\rho$.

Large traffic networks contain a diverse set of road segments. We want to explore whether homogeneous subnetworks can improve the overall performances of compressed sensing. We divide the traffic network into $s$ mutually exclusive subnetworks such that $\alpha_p = [\alpha_1 \ldots \alpha_s]$ where $\alpha_i \in \mathbb{R}^{1 \times n}$ $\forall i = 1, \ldots, s$. Then, we perform compressed sensing for each subnetwork separately. At last, we merge the results of the clustered subnetworks to infer the traffic state of the entire network. Although different choices of temporal and/or spatial clustering can be applied, we consider simple clustering based on different road categories in this study.

The overall performance of the proposed compressed sensing method is sensitive to the “compressibility” of the network and “non-stationarity” in the traffic data. For compression, we represent the traffic data as a product of two low-rank matrices, i.e., the subnetwork data matrix and the most appropriate relationship matrix. As the compression is lossy, we expect the reconstructed matrix $\hat{A}_h$ to be different from the original matrix $A_h$. The issue of non-stationarity is due to the fact that matrix $X_h$ is inferred from training (historical) data instead of the current data. The matrix $A_p (A_p = [\alpha_1^k \ldots \alpha_p^m]^T)$ is not available, and the goal is to infer that matrix by extrapolating the matrix $C_p (C_p = [c_1^k \ldots c_p^m]^T)$ according to the CX decomposition. Obviously, the matrix $X_p$ cannot be extracted from the current data $A_p$, since the matrix $A_p$ is not available. Instead we determine $X_h$ from training data set. Since traffic is not perfectly stationary, this approximation will induce an additional reconstruction error. We refer to it as the error due to non-stationarity of traffic spatial relationships. To quantify this error, let us call $B = C_h X_h$, the reconstruction of the data matrix $A_p$, assuming the latter is available to compute the CX decomposition. The reconstruction $\hat{A}_p (\hat{A}_p = [\hat{\alpha}_1^k \ldots \hat{\alpha}_p^m]^T)$ in the scenario of compressed sensing is less accurate, since we need to replace $X_h$ (determined from the test data matrix $A_p$) by $X_h$ (determined from training data matrix $A_h$). The mean squared error (MSE) incurred for compressed sensing can be written as:

$$\frac{1}{m} \| \hat{A}_p - \hat{A}_p \|^2_F = \frac{1}{m} \| (A_p - C_p X_p) - (C_p X_h - C_p X_h) \|^2_F,$$

$$= \frac{1}{m} \| (A_p - B) - (\hat{A}_p - B) \|^2_F,$$

$$= \frac{1}{m} \left( \sum_{i = k}^{m} \sum_{j = f}^{n} (a_{ij} - b_{ij})^2 + \sum_{i = k}^{m} \sum_{j = f}^{n} (\hat{a}_{ij} - b_{ij})^2 - 2 \sum_{i = k}^{m} \sum_{j = f}^{n} (a_{ij} - b_{ij})(\hat{a}_{ij} - b_{ij}) \right),$$

where $r = (m - k + 1)$ represents the number of time instances in test data matrix $A_p$. The first component of the error corresponds to the compressibility of the network and the second component is due to the non-stationarity of spatial patterns within the network (see (5)). The third component of the error refers to the correlations between aforementioned error components (see (5)). To make this interpretation more explicit, we rewrite (5) as:

$$\text{MSE}_{est} = \text{MSE}_{com} + \text{MSE}_{ns} - 2 \xi_{est},$$

where $\xi_{est}$ is correlation coefficient between compressibility and non-stationarity. We will analyze the behavior of these errors for different compression ratios in Section VI.

C. Compressed prediction

In the previous section, we inferred the condition of the entire traffic network by observing traffic conditions at a small subset of links. Here we will extend this approach to prediction; we aim to predict the state of the entire traffic network from the predicted state of a small subset of links. We recall that low-dimensional models generated by PCA can not be utilized for this task since PCA requires information for all links in the network. Instead, we use state-of-the-art algorithm to predict the traffic speed only for a selected subset of locations. Then, we utilize the proposed method to extrapolate the predictions to the rest of the network using the precomputed relationship matrix. This can be written as $\hat{\alpha}_p = \hat{\alpha}_h X_h$, $\forall i = k, \ldots, m$, where $\hat{\alpha}_h \in \mathbb{R}^{1 \times n}$ is the row vector containing the predicted values of the traffic variable at the selected locations and $i$th time instance, $\hat{\alpha}_h \in \mathbb{R}^{1 \times n}$ contains the predictions for all locations at $i$th time instance, and $X_h$ is the relationship matrix. If the predictions $\hat{\alpha}_h$ would be identical to the true values $\alpha_h$, then the problem boils down to compressed sensing, which we discussed in the previous section. In practice, however, the predictions have some inaccuracies. Therefore, we can write $\hat{\alpha}_p = \hat{\alpha}_h + \Delta \alpha$ where $\Delta \alpha$ represents the prediction error for the subnetwork at time $i$. Furthermore, let $D = C_h X_h$ be the estimated network profile, during the entire observational period, without any prediction error in $C_f (C_f = [\hat{\alpha}_f^1 \ldots \hat{\alpha}_f^m]^T)$. Then, the MSE between predicted $\hat{A}_f$ ($\hat{A}_f = [\hat{\alpha}_f^1 \ldots \hat{\alpha}_f^m]^T$) and true future values $A_f (A_f = [\alpha_f^1 \ldots \alpha_f^m]^T)$ can be written as:

$$\frac{1}{m} \| A_f - \hat{A}_f \|^2_F = \frac{1}{m} \| (A_f - C_f X_h) - (C_f X_h - C_f X_h) \|^2_F,$$

$$= \frac{1}{m} \| (A_f - D) - (\hat{A}_f - D) \|^2_F,$$

$$= \frac{1}{m} \left( \sum_{i = k}^{m} \sum_{j = f}^{n} (a_{ij} - d_{ij})^2 + \sum_{i = k}^{m} \sum_{j = f}^{n} (\hat{a}_{ij} - d_{ij})^2 - 2 \sum_{i = k}^{m} \sum_{j = f}^{n} (a_{ij} - d_{ij})(\hat{a}_{ij} - d_{ij}) \right),$$

where the first component of the error corresponds to the non-stationarity of spatial patterns within the network and the
second component is due to inaccurate predictions. The third component of the error shows the correlations between these two error components. We refer to (9) as MSE for compressed prediction. We rewrite (9) in more explicit form:

\[
\text{MSE}_{\text{total}} = \text{MSE}_{\text{est}} + \text{MSE}_{\text{pred}} - 2\hat{\xi}_{\text{pred}},
\]

where \(\hat{\xi}_{\text{pred}}\) refers to correlation between non-stationarity and predictability. By substituting (6) in (10), we obtain:

\[
\text{MSE}_{\text{total}} = \text{MSE}_{\text{com}} + \text{MSE}_{\text{ns}} + \text{MSE}_{\text{pred}} - 2\hat{\xi}_{\text{est}} - 2\hat{\xi}_{\text{pred}}.
\]

Hence, the total error of compressed prediction can be decomposed into four error components: (i) error due to compression; (ii) error due to changes in spatial relationships (non-stationarity); (iii) error due to inaccurate predictions; (iv) correlations among the previous error components.

Compressed prediction provides significant reduction in computational complexity by explicitly predicting the traffic variables for only a small subset of road segments in the network. Compressed prediction involves two computations: (i) prediction of the traffic conditions at representative locations in the network and (ii) extrapolation of the predicted values to the entire network. In the former, the computational complexity depends on the underlying prediction algorithms, and is proportional to the number of locations \(c\) in the subnetwork. The second step (extrapolation) requires a single matrix-vector multiplication with complexity \(O(cn)\). In practice, the predictions at each link in the subnetwork are computationally complex. By contrast, the extrapolation can be executed much faster. Therefore, by performing prediction only for a small subnetwork, the computational complexity can be drastically reduced.

V. EXPERIMENTAL SETUP

We consider the nationwide traffic network in Singapore which contains diverse types of roads (see Fig.1). The variable of interest is the average traffic speed, i.e., the average speed of all vehicles which traverse a link during the given sampling interval of 5 minutes. The data set contains the average speed at each link of the transportation network for a period of three months (August - October 2011). We selected 17967 links which had less than 5% of missing values. We performed imputation by applying the Low Dimensional CP Weighted OPTimization (LDCP-WOPT) imputation method as it is able to deal with the large data set [20], [21].

We represent the data set in the form of a matrix as explained in Section IV. For compressed sensing and compressed prediction, we need training data to: (i) determine the subnetwork of \(c\) links, corresponding to the matrix \(C\) (see

Fig. 1: Left: City-scale network of Singapore with 17,967 road segments of different categories, from freeways to local feeders. Right: Corresponding input data matrix.
Table I: Overlap (%) among the \( k \) links with the highest selection probability (calculated by the SVD sampling method) in the three months of data.

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Oct.</td>
<td>100</td>
<td>98.38</td>
<td>92.65</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Sep.</td>
<td>93.88</td>
<td>100</td>
<td>92.82</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Aug.</td>
<td>100</td>
<td>93.16</td>
<td>91.67</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PRD (%)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table II: MSE of the proposed method for application of compressed sensing and for different compression ratios.

<table>
<thead>
<tr>
<th></th>
<th>2</th>
<th>4</th>
<th>6</th>
<th>8</th>
<th>10</th>
</tr>
</thead>
<tbody>
<tr>
<td>MSE (Comp)</td>
<td>0.01</td>
<td>2.18</td>
<td>7.27</td>
<td>11.95</td>
<td>15.93</td>
</tr>
<tr>
<td>MSE (Ns)</td>
<td>19.77</td>
<td>29.07</td>
<td>31.20</td>
<td>29.23</td>
<td>27.35</td>
</tr>
<tr>
<td>( \xi ) (Corr)</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>Total MSE</td>
<td>19.78</td>
<td>31.25</td>
<td>38.47</td>
<td>41.18</td>
<td>43.28</td>
</tr>
</tbody>
</table>

We now investigate the case of compressed sensing. We aim to reconstruct the average speed at each link in the entire network by collecting data from a small subset of roads. The relationship matrix \( X_k \) is determined from the training set (data from Aug-Sep, 2011), and the reconstruction error is assessed on the test set (data from October, 2011). Fig. 2b shows the reconstruction accuracy of the proposed method for three different approaches: In the first approach we select the subset of road segments according to SVD sampling scheme (see solid line in Fig. 2b). In the second approach we cluster the network according to the category of the road. For each cluster, we select the subset of the road segments using SVD sampling scheme. Then, we perform compressed sensing for each cluster separately (see dashed line in Fig. 2b). Intuitively, the difference between the second and the third approach shows the gain obtained by network clustering. Fig. 2b indicates that applying the compressed sensing method to different road categories leads to better estimation performance for the entire network. As expected, the reconstruction accuracy of all three approaches increases with the size of subnetwork. Let us now investigate the error of compressed sensing in more details. In our analysis, we consider the subset of links as defined in SVD scheme without clustering.

The overall compressed sensing (estimation) error is caused by information loss due to compression of traffic data and changes in traffic behavior between training and testing periods. Table II shows the MSE of the individual error components, the correlation between the two errors components and the total MSE, for different compression ratios. As it can be seen from Table II the non-stationarity of the traffic data is the main contributor to the estimation error. As expected, the error associated with the compressibility of traffic data increases with the compression ratio. Furthermore, we decompose the MSE of compressed prediction into estimation and prediction components. Table III shows the contribution of these two error components as well as the correlation between them for 5 minute prediction horizon. As it can be seen from Table III, the estimation error increases with the compression ratio. This increase in estimation error is mainly due to non-stationarity of the error component (see Table II). Table III shows that the prediction error tends to be dominant for smaller compression ratio, i.e., when significant portion of the network is explicitly predicted. From Table III, we can also see that there is some correlation between the two error components.

The proposed approach of compressed prediction provides substantial reduction in computational complexity by explicitly predicting the variables at a small representative set, followed
Computational complexity is approximately proportional to the number of road segments in C. Consequently, the reduction in computational complexity is approximately proportional to the compression ratio (see Table IV).

### VII. Conclusions

In this paper we utilized column based (CX) low-dimensional models to enhance the scalability of compressed sensing and compressed prediction. We decomposed the compressed prediction error into several components and investigated the relationship between them. Our numerical results show that the proposed method significantly reduces the computational cost at the expense of a negligible increase in prediction error.

In future work, we will explore whether other column selection techniques can lead to better performance of compressed prediction. Also, we will investigate how compressed prediction can be applied in conjunction with routing, in order to optimize routes taking future traffic conditions into account.

### VIII. Acknowledgment

Justin Dauwels wishes to thank Andrzej Cichocki for helpful discussions.

### References


