2014 IEEE 17th International Conference on
Intelligent Transportation Systems (ITSC)
October 8-11, 2014. Qingdao, China

Traffic Condition Matrix Estimation via Weighted Spatio-Temporal Compressive Sensing for Unevenly-Distributed and Unreliable GPS Data

Ye Li[†§] Chen Tian[⋆†] Fan Zhang[†] Chengzhong Xu[†§]
[†]Shenzhen Institutes of Advanced Technology, Chinese Academy of Sciences, China
[⋆]Department of Electronics and Information, Huazhong University of Science and Technology, China
[§]Department of Electrical and Computer Engineering, Wayne State University, MI, USA
[†]{li.ye, zhangfan, cz.xu}@siat.ac.cn, [⋆]tianchen@hust.edu.cn

Abstract— **Traffic condition monitoring is important to nowadays metropolitan. A recent trend is to exploit the prevalence of Global Positioning System (GPS) embedded in public vehicles. The collected data forms a two dimensional traffic condition matrix (TCM), *i.e.*, time slot and road segment. The problem is that the TCM directly obtained from the probed data is incomplete. Traffic estimation can complete the TCM by filling the missing entries. We find that in practice it is challenging to reliably estimate a TCM. First, The distribution of probed data is uneven among road segments. Second, most entries of probed data are unreliable since they are the average of only a few reports. Our approach is Weighted Spatio-Temporal Compressive Sensing. Demonstrated by extensive large scale computational experiments, the estimation error of our approach reduces to just half of the baseline approach.**

I. Introduction

Traffic condition monitoring is important to nowadays metropolitan. The ultimate goal is to determine the traffic condition of *every* road segment at *every* time. Road infrastructure planning, traffic management, road engineering and personal route planning all can be benefit from accurate traffic monitoring. Traditional approaches, such as inductive loop detectors [1] and video cameras [2], are unscalable. Due to their large infrastructure deployment and operational cost, their coverage is limited to main roads only.

A recent trend of traffic monitoring is to exploit the prevalence of Global Positioning System (GPS) embedded in public vehicles (*i.e.*,taxis and buses) [3], [4], [5], [6], [7]. Driving along a road, each vehicle periodically sends its location and condition updates via a cellular data connection to a data center. The probed data collected in a fixed timeslot length (*e.g.*, 15 minutes) are aggregated; usually the average value of speed updates is used as the metric of the traffic condition for a road segment at that timeslot [8]. Let each timeslot serves as a row and each road segment serves as a column, the collected probed data forms a two dimensional traffic condition matrix (TCM). The advantages of using these public vehicles are: (1) large coverage of metropolitan roads given their operational nature, and (2) low deployment/operational cost given that a GPS receiver is already a standard equipment in most public vehicles.

The problem is that the TCM directly obtained from the probed data is incomplete for any reasonable timeslot length value (*i.e.*, minutes instead of hours or days). There are two reasons for value vacancies. First, probe vehicles drive according to their own wills; it should not be expected that there would have at least one taxi in each New York road segment at 5 AM. Second, there are serious impediments to reliable large scale TCM data collection: GPS signals can be blocked by urban skyscrapers or tunnels; wireless connection could be lost; data centers could fail *etc.*

Traffic estimation can complete the TCM by filling the missing entries. Leveraging the presence of certain types of structures and redundancy in collected data, algorithms such as KNN [9] and Compressive Sensing [10], [11] could interpolate the matrix.

A compressive-sensing based estimation algorithm, by Zhu *et al.* [6], has been proposed recently. We apply this simple baseline approach to our dataset, which consists of realtime updates from over 14,000 taxis traveling 6,217 road segments in Shenzhen, China. Initially, it seems that good results can be obtained. However, after looking deep into the resulted matrix, we find that in practice it is very challenging to reliably estimate a TCM.

First, The distribution of probed data is extremely uneven among road segments. Throughout a day, some main roads always have condition updates (*i.e.*, busy segments); while many auxiliary roads are always close to 0 update (*i.e.*, idle segments). It is intuitive that the missing values of busy segments, if any, could be easily interpolated. While given the already overly sparse samples, it should be extremely hard to accurately estimate missing values of idle segments.

There are two important implications: (1) idle segments should be the focus of traffic estimation, given they are more likely to generate large errors; (2) average error over all segments is a misleading metric. The baseline approach claims to achieve a 20% average error even when 80% entries are missing. However we find that with their approach, the error of busy segments are very low (as expected), while the error of idle segments are far more higher. After taking average, this defect could be easily ignored.

Second, most entries of probed data are unreliable. Either with just 1 or with 100 updates in a timeslot, the entry would be considered valid. It is clear that the reliability difference between 1 and 100 samples is huge. Ruey *et al.* proved that at least 10 samples per timeslot are needed to get a reliable

978-1-4799-6077-4/14/$31.00 ©2014 IEEE

speed measurement [8]. However, this bar is too high for majority of the probed data. As shown in Section III, even with 30 minutes timeslot, only 50% entries are reliable; this number reduces to 38% with 15 minutes timeslot.

There are also two important implications: (1) unreliable probed data could be a poison to the quality of traffic estimation, given these values are used when interpolating related empty entries; (2) there is no ground truth for error evaluation of many entries. Many times we find that the estimated value looks intuitively “more reasonable” compared with the probed value, which usually is averaged from just several updates. The consequence is that the estimation error values of such entries might be unnecessarily exaggerated.

The target of this paper is a practical solution for urban traffic estimation via unevenly-distributed and unreliable probed vehicle data. The contributions of this paper include:

- By analyzing a large dataset of real probed data from over 14,000 taxis in a metropolitan with 6,217 road segments, we proved that the distribution of probed value are uneven in both spatial and temporal dimensions; we also proved that most entries of probed data are unreliable (Section III).
- We fully analyze Zhu’s baseline approach; it has been shown that due to unevenly-distributed data, idle segments are more likely to generate large errors hence average error is a misleading metric; we also demonstrate that unreliable probed vehicle data could degrade the quality of estimation. (Section IV).
- Our approach is Weighted Spatio-Temporal Compressive Sensing. We exploit the hidden spatio-temporal relationship in the TCM to improve the estimation quality of idle segments. We use weighted average between historical and probed data together to improve the reliability of the input values, hence reduce the impact of poisonous data to the interpolation process (Section V).
- Demonstrated by extensive experiments, the estimation error of our approach for idle segments is less than 40% of the baseline approach; the overall estimation error reduces to around half (Section VI).

II. Back Ground

A. *Traffic Condition Matrix Estimation*

A Traffic Condition Matrix (TCM) is a non-negative matrix X that describes the speed of traffic (*e.g.*, kilometer per hour) per road segment per timeslot. A road segment is between two neighboring road intersections. In practice the speed is typically measured over some timeslot length, and the entry value reported is an average. The length of a timeslot is determined by road condition dynamics, normally in minutes. The TCM can be thought of as a two-dimensional array $X \in \mathbb{R}^m \times \mathbb{R}^n$ (where there are m timeslots and n segments present). The columns of X represent the road segments at different times, while the rows represent the time evolution of the matrix.

Let $D \in \mathbb{R}^m \times \mathbb{R}^n$ contains the direct available probed values and M is a $m \times n$ matrix given by

$$M(i,j) = \begin{cases} 0, \text{if } D(i,j) \text{ is missing} \\ 1, \text{otherwise} \end{cases} \tag{1}$$

and $.*$ denotes an element-wise product (*i.e.*, $A = B.*C$ means $A(i,j) = B(i,j)C(i,j)$). Then there is a set of linear constraints on the TCM

$$M.*X = D \tag{2}$$

.

The purpose of traffic estimation is seeking an estimated matrix $\widehat{X}$ that satisfies the conditions imposed by Equation 2. Normally, there are not enough information to unambiguously determine $\widehat{X}$; TCM estimation is an under-constrained linear-inverse problem.

The objective is to minimize the estimation error. The error can be measured by using the Normalized Mean Absolute Error (NMAE) metric

$$NMAE = \frac{\sum_{i \in m} \sum_{j \in n} |X(i,j) - \widehat{X}(i,j)|}{\sum_{i \in m} \sum_{j \in n} |X(i,j)|}. \tag{3}$$

Note that NMAE could be larger than 1, if the error value is large enough.

B. *Probed Dataset*

Our probed dataset consists of realtime updates from over 14,000 taxis traveling 6,217 road segments in Shenzhen, China. Besides longitude, latitude and time-stamp readings observed by the GPS receiver, the embedded device also captures speed and heading measurements. Each taxi updates 30 seconds in average, together there are around 4 GB data per day and over 100 GB per month.

The road segment paradigm of Shenzhen is directional. The reported location and heading information of each update is matched against the library to find a matching segment. Finally, an entry is obtained by averaging the updates within a specific timeslot. Usually, we take one full day as a TCM.

For such a large dataset, the process should be parallel. We set up a Hadoop cluster over 12 nodes with 192 cores and 192 GB memory in total. By developing several MapReduce paradigms, the preprocessing of one month data costs just a few hours.

C. *Interpolation Algorithms*

There are a number of approaches that have been proposed for matrix completion recently.

Seasonal ARIMA (SARIMA): SARIMA is a straightforward time series analysis model for univariate traffic condition data [12]. The foundation of this method rests on the Wold decomposition theorem and the observations that discrete interval traffic condition data implies a hidden periodical characteristics in temporal dimension. Hence, historical traffic data for a specific road can be used to estimate missing values both in the past and the future.

TABLE I

TCM INTEGRITY AND RELIABILITY

	5 minutes	15 mins	30 mins	60 mins
Updates>0	56.33%	70.78%	78.41%	84.66%
Updates≥ 10	19.59%	38.09%	50.00%	61.44%

K-nearest neighbors algorithm (KNN): KNN is a classic local interpolation algorithm using information of nearest neighbours to interpolate [9]. For a missing data, KNN searches for its K-nearest neighbours and calculates estimation by a weighted average. KNN intends to utilize correlations between near neighbours, for example temporal and spatial correlation in TCM.

Compressive Sensing: Compressive Sensing is a relatively new signal processing technique for data compression and signal reconstruction [10]. The main idea is that if the objects we observed are compressible in a fixed structure, then we can reconstruct and recover these observations with only a small number of measurements. In TCM, spatial correlation between roads and temporal periodic characteristics of traffic flow can be used as prior knowledge for interpolating missing values.

III. PROBED DATASET ANALYSIS

We demonstrate the distribution characteristics of our TCM by analyzing the probed data of 24 hours on July 09, 2013. Let *integrity* denotes the fraction of timeslots for a road segment that there exists at least one vehicle update. Four granularities (*i.e.*, 5 minutes, 15 minutes, 30 minutes and 60 minutes) are analyzed; the default granularity is set to 15 minutes.

A. Sparsity of TCM

Table I first row shows the whole matrix integrity under four time granularities. It is intuitive that with the growth of timeslot length, the integrity increases too. With the typical 15 minutes granularity, over 70% entries have update values. While with 60 minutes, nearly 85% entries have values. To sum up, our TCM is incomplete, but not extremely sparse.

Zhu's measured TCM is more sparse than ours: with 2,000 probe vehicles running on 5,812 road segments and update at a 15 minutes timeslot granularity, the TCM integrity is less than 25% [6]. The reason is that the number of our probe vehicles is more than 7 times of their system, while the number of covered segments is comparable. The advantage of our dataset is that more data could be used in the evaluation to demonstrate the performance of approaches.

B. Unevenly-Distributed Data

First we study the uneven distribution from a spatial perspective. Figure 1(a) shows the CDFs of all roads under different timeslot granularities. Even with 60 minutes granularity, there are still around 10% segments whose integrity is less than 20%; it implies that some auxiliary roads are extremely idle. With a typical 15 minutes granularity, half of all segments have an integrity larger than 90%: these roads are not necessary busy; it just suggests that there are enough number of taxis probing the roads.

TABLE II

GROUND TRUTH OF ENTRIES

timeslot	*1*	*2*	*3*	*4*	*5*
Update Count	11	5	4	3	12
X value	42	37	13	22	37
$\widehat{X}$ value	41	33	29	39	33

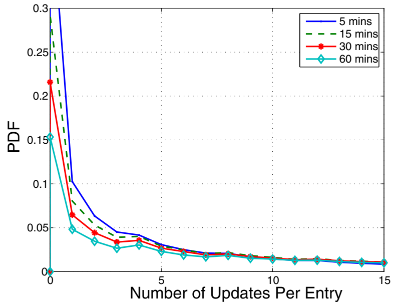

Fig. 2. Number of Updates per Entry

Next we study the uneven distribution from a temporary perspective. Shown in Figure 1(b), with 15 minutes granularity, there is no timeslot whose integrity is less than 50%. And for around 10% timeslots, their integrities are larger than 75%. It is also interesting that even with 5 minutes granularity, there is no timeslot whose integrity less than 30%. Obviously that the increase in the number of probe vehicles significantly reduce the level of sparsity.

Finally we demonstrate spatio-temporal integrity difference together in Figure 1(c). Two sets of segments are shown with 5 and 15 minutes granularity: one is busy set, another is idle set. The average integrity of all segments is also presented. Clearly, the traffic trough is around 5 AM. It is also clear that busy segments rarely have missing entries. Even at the trough with 5 minutes granularity, their average integrity value is above 90%. As a comparison, idle segment is very sparse throughout the day: mostly the integrity is just around 20%.

C. Reliability of Probed Data

In this part we analyze the reliability of those probed data. Intuitively speaking, the reliability difference between 1 and 100 updates should be huge. We present an update log of a typical segment in Table II. In 5 consecutive timeslots, the segment get valid updates: the updates count and the averaged value are shown in the first and second row respectively. Apparently, the values of timeslot 3 looks suspicious compared with its neighboring entries; it might because that timeslot 3 is derived from only 4 updates.

Ruey *et al.* proved that at least 10 samples per timeslot are needed to get a reliable speed measurement [8]. However, this bar is too high for majority of the probed data. Shown in the second row of Table I, even with 30 minutes timeslot, only 50% values are reliable; this number reduces to 38% with 15 minutes timeslot.

To make it more clear, we plot the PDF of the numbers of updates under different timeslot granularities in Figure 2. Only entries with updates number between 0 and 15 are plot-

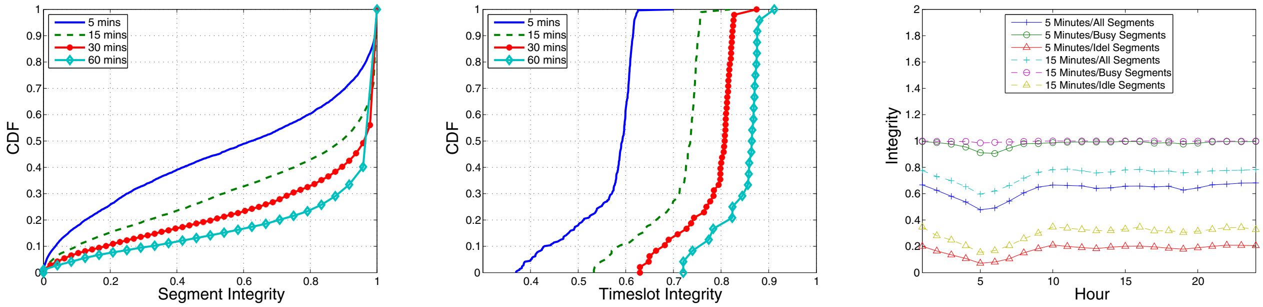

Fig. 1. (a) CDF of integrity over roads (b) CDF of integrity over timeslots (c) spatial-temporal integrity

ted. With the increase of numbers, the probability decreases. Many entries have a value between 2 and 4. The line slope becomes stable after number 10; we do observe some entries have over 100 updates.

IV. Lessons Learned

A. Baseline Algorithm

We present an approach proposed by Zhu *et al.* [6] as the baseline algorithm for analysis. By analyzing the probed data in Shanghai, they proved the hidden low rank structures of TCM.

The idea is that the estimated matrix $\widehat{X}$ should be reasonably close to the measurement D; it should have a low rank since TCM is compressible. Thus this algorithm solves the following minimization problem:

$$\begin{aligned} &\text{minimize } \operatorname{rank}(\widehat{X}) \\ &\text{subject to } M.*\widehat{X} = D. \end{aligned} \tag{4}$$

Rank minimization problem has a non-convex objective and it is difficult to solve. They use a SVD-like factorization to rewrite as $\widehat{X} = U\Sigma V^T = LR^T$. Thus Equation (4) turns into a new form of:

$$\begin{aligned} &\text{minimize } ||L||_F^2 + ||R^T||_F^2 \\ &\text{subject to } M.*(LR^T) = D. \end{aligned} \tag{5}$$

And the solution is to just find matrix L and R that minimize the summation of their Frobenius norms

$$\text{minimize } ||M(LR^T) - D||_F^2 + \lambda(||L||_F^2 + ||R||_F^2). \tag{6}$$

Here regularization parameter λ is a tunable tradeoff variable between an estimated precision and the goal of achieving low rank.

For evaluation, they choose 221 road segments in Shanghai downtown area, whose TCM is almost full; let's denote this directly observed matrix as D_{orig}. The evaluation procedure is: by randomly drop part of the probed values from D_{orig}, a new partial TCM D_{part} is get. Let $D = D_{part}$, they get the estimated $\widehat{X}$ by solve formulation (6). Let $D_{dropped} = D_{orig} - D_{part}$, the entries in $D_{dropped}$ are treated as the ground truth for evaluation purpose: the estimation quality is measured by the average error of all entries in $D_{dropped}$ compared with their interpolated values in $\widehat{X}$. We call this method "dropped for evaluation". Their result seems promising: even when the total integrity of the matrix is reduced to 20%, the average estimation error is no more than 20%.

We apply the baseline approach to the same dataset of 24 hours on July 09, 2013; the time granularity is 15 mins. The resulted NMAE is similar: it is around 23% when when the TCM integrity is reduced to 20%. However, after looking deep into the resulted matrix, we find that in practice it is very challenging to reliably estimate a TCM.

B. Impact of Unevenly-distributed Data

As mentioned above, idle segments should be the focus of traffic estimation. Shown in Figure 3(a), the error distribution of the baseline approach is uneven: the error of some roads are extremely high, some are extremely low, while others are in between.

We classify segments based on their whole day integrity to 20 categories in a 0.05 granularity. Each NMAE value of a segment category is plotted in Figure 3(b). The real fact is that the error of busy segments are very low, while the error of idle segments are far more higher. After taking average, this defect could be easily ignored.

People may wonder how could the baseline approach achieves the 20% average error. It is still because of the unevenly-distributed data: most entries belong to those high integrity roads, hence most values randomly dropped for later comparison belong to high integrity roads. This situation is demonstrated in Figure 3(c): x-axis is the index of $D_{dropped}$ entries sorted by their segment integrity; over 60% entries are from roads whose integrity higher than 90%; these entries have low errors; although there are extremely high errors in the range of 0 to 30% integrity, the significantly outnumbered high integrity entries successfully push the average error to a low value such as 20%.

To sum up, average error is a misleading metric. We define a new metric named Integrity-Categorized Normalized Mean Absolute Error (IC-NMAE): the mean of the NMAE results of multiple segment integrity categories.

C. Impact of Unreliable Probed Value

First of all, unreliable probed data could be a poison to the quality of traffic estimation. In traffic estimation, input data are used to interpolate related empty entries; if they are unreliable, the results are unreliable. A naive solution to improve data reliability could be a threshold of the number of updates: only average speed entries generated from larger

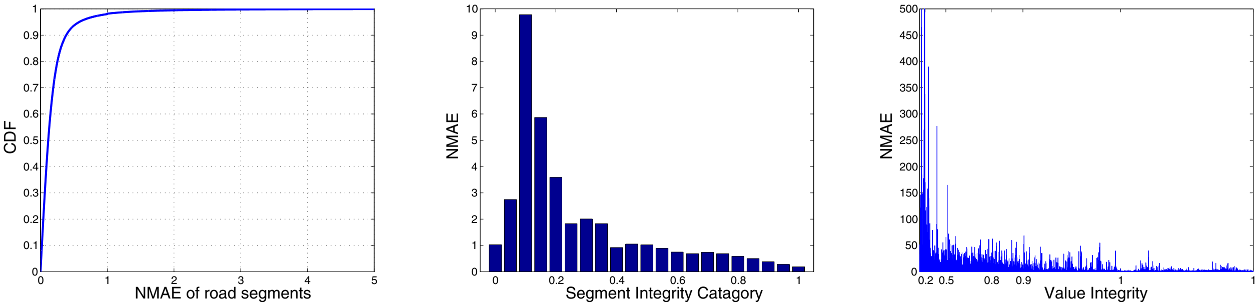

Fig. 3. (a) NMAE value CDF (b) NMAE over Integrity Category (c) NMAE over values

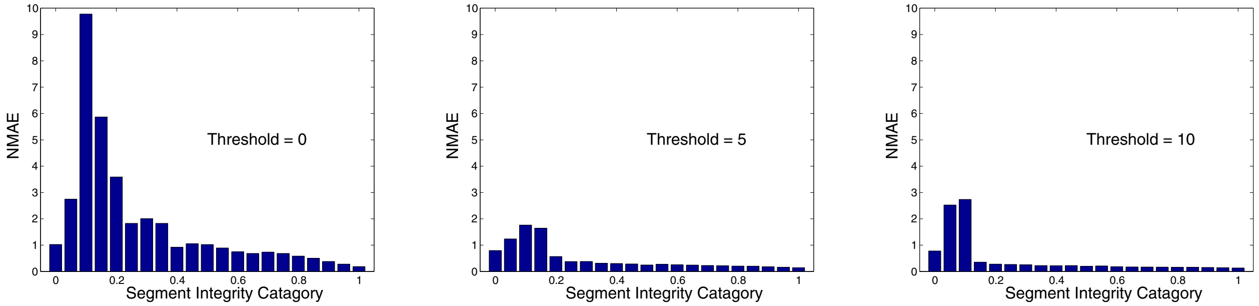

Fig. 4. Estimate Error with (a) Threshold 0 (b) Threshold 5 (c) Threshold 10

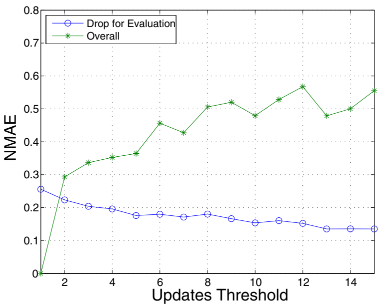

Fig. 5. Errors increases together with threshold

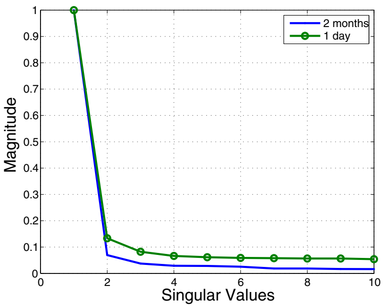

Fig. 6. Magnitude of Singular Values

enough updates are admitted to the estimation algorithm. We repeat the baseline approach to our dataset with different threshold settings: 0, 5 and 10. The NMAE values of segment integrity categories are presented in Figure 4: the higher the threshold, the lower the error.

The trick here is that: the higher the threshold, the more reliable the remained values, hence the more reliable the value in $D_{dropped}$ for evaluation. That is the main reason for good results in Figure 4. while from a global point of view, the higher the threshold, the more probed data dropped. Those drooped values, although not that reliable, still contains useful information; remove them completely out of the process actually hurts the whole system. Shown in Figure 5, the overall error actually increases together with the threshold.

The second implication is that, there is no ground truth for error evaluation of many entries. Many times we found that the estimated value is intuitively more correct compared with the removed probed value, which usually is averaged from just several updates. Shown in Table II third row, after an evaluation, the $\widehat{X}$ value of timeslot 3 looks more reasonable than that in D. The consequence is that the estimation error values of such entries might be unnecessarily exaggerated.

This conjecture is consistent with the implementation gap between Zhu's dataset and our dataset. Note that their evaluation chooses road segments in Shanghai downtown area; their entries in $D_{dropped}$, for evaluation, intuitively should be more reasonable than ours.

V. Weighted Spatio-Temporal Compressive Sensing

A. Low Rank Analysis

To analyze the low rank characteristics of our TCM, we present the magnitude of TCM singular value in Figure 6. There are two dataset, one is the TCM of July, 9, 2013, the other is a TCM averaged from the historical data of the last two months. The advantage of the latter one is complete (100% integrity), and we denote it as $\overrightarrow{X}$.

The results from both one day data and historical $\overrightarrow{X}$ confirm that major energy concentrated in the first several components. In the following approach, we take rank 2 as the low rank approximation of X.

B. Spatial-Temporal Factors to Help Idle Segments

Besides the global low-rank structure, we know that there are additional spatio-temporal relationships between TCM rows and columns. As mentioned above, given the already sparse samples, it is extremely hard to accurately estimate the missing values of idle segments. We introduce the spatio-temporal factors to help the estimation of idle segments, similar to a previous work in Internet traffic estimation [11].

Let S be the spatial constraint matrix. We need to find which columns (segments) are close to each other. We choose S based on the similarity between columns. For each column of $\overrightarrow{X}$, the K most similar columns are chosen; a linear regression finds the set of weights $\omega(k)$; then we set $S(i,i) = 1$ and $S(i, j_k) = -\omega(k)$. The spatial approximation is to minimize $||(LR^T)S^T||_F^2$.

Let T be the temporal constraint matrix. We let $T = Toeplitz(0,1,-1)$, which is a Toeplitz matrix with central diagonal given by ones, and the first upper diagonal given by negative ones. It reflect the intuition that adjacent points in time domain are often similar. The temporal approximation is to minimize $||T(LR^T)||_F^2$.

With all these factors, we solve the following

$$\begin{aligned} \text{minimize } & ||M(LR^T) - D||_F^2 + \lambda(||L||_F^2 + ||R||_F^2) \\ & + \lambda_s ||(LR^T)S^T||_F^2 + \lambda_t ||T(LR^T)||_F^2. \end{aligned} \tag{7}$$

λ_s and λ_t are magnitude adjusting parameters. We choose to use $\lambda_s = 0.1\sqrt{\lambda}$ and $\lambda_t = \sqrt{\lambda}$ respectively.

C. Weighted Average to Alleviate Impact from Unreliable Values

To remove the impact of poisonous data to the interpolation process, we use weighted combination between historical $\overrightarrow{X}$ and probed data D together to improve the reliability of the input values.

Let the new weighted TCM be D'. We take 10 updates as a threshold and trust the values above that; otherwise, a weighted average replaces the original observed entry. Let D_{num} denote the number of updates per entry in D, the algorithm is Note that all empty entries are replaced by values from $\overrightarrow{X}$; D' is a complete TCM.

Algorithm 1: Weighted Average

```
if $D_{num}(i,j) \geq 10$ then
    $D'(i,j) = D(i,j)$;
else
    $D'(i,j) =$
    $((10 - D_{num}(i,j)) * \overrightarrow{X}(i,j) + D_{num}(i,j) * D(i,j))/10$
end if
```

D. Overall Process

Our Weighted Spatio-Temporal Compressive Sensing process takes the following steps

1) *Preliminary Interpolation* We use Algorithm 1 to make an initial estimation matrix D' from D and $\overrightarrow{X}$;

Algorithm 2: Restore Algorithm

```
if $D_{num}(i,j) \geq 0$ then
    $\widehat{X}(i,j) = D(i,j)$;
end if
```

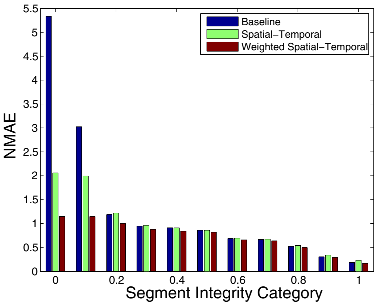

Fig. 7. Estimate error for each Integrity Category

2) *Estimation* We let D' replace D in Formulation (7), and starts the estimation process to get $\widehat{X}$.
3) *Restore* After estimation, the replaced value are restored according to Algorithm 2.

VI. Evaluation

A. Settings

The main dataset we use is still July 09, 2013. We also evaluate performance in other dates and the results are similar. The three approaches been evaluated are *Baseline* (Zhu's compressive sensing), *Spatio-Temporal* (only spatio-temporal factors) and *Weighted Spatio-Temporal* (spatio-temporal factors and weighted input). After training, we set λ to 0.001, 0.01 and 0.1 for 15 minutes, 30 minutes and 60 minutes respectively. The granularity of segment integrity increases to 0.1 for better presentation.

We borrow the idea of "dropped for evaluation". To avoid the same pitfall of biased evaluation (as that in the baseline approach), *all* segments are used in our experiments. Still denote the directly observed matrix as D_{orig}. The evaluation procedure is: by randomly drop part of the probed values from D_{orig}, a new partial TCM D_{part} is get. Let $D = D_{part}$, here *Weighted Spatio-Temporal* will follow the process defined in Section V; there is no change to the other two algorithms. After experiments, let $D_{dropped} = D_{orig} - D_{part}$, the estimation quality is measured by the average error of all entries in $D_{dropped}$ compared with their interpolated values in $\widehat{X}$.

B. Initial Comparison

Figure 7 shows a comparison of algorithms for matrix integrity 20% at time granularity 15 minutes. Compared with *Baseline*, *Spatio-Temporal* significantly reduces the error for low integrity categories. With the introduce of weighted input adjustment, the estimation error is further reduced at every category.

To make it more clear, we present the IC-NMAE Metric is Table III. As mentioned above, most exceptional results

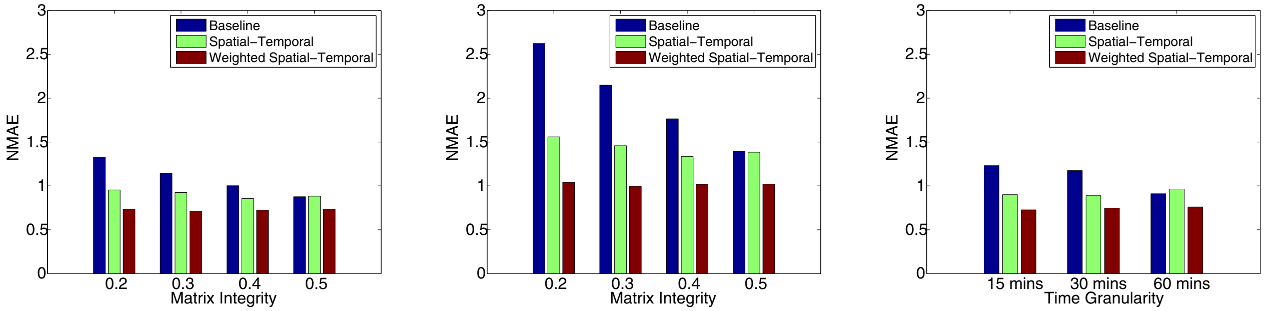

Fig. 8. Estimate Error vs. Integrity (a) All (b) Low and vs. (c) Time granularity

TABLE III

IC-NMAE Metric

Integrity Category	High	Low	All
Baseline	0.59	2.62	1.33
Spatio-Temporal	0.61	1.56	0.95
Weighted Spatio-Temporal	0.56	1.04	0.73

TABLE IV

IC-NMAE Metric for hypothetic ground truth

Integrity Category	High	Low	All
Baseline	0.20	1.42	0.65
Spatio-Temporal	0.24	0.78	0.44
Weighted Spatio-Temporal	0.17	0.28	0.20

appear in range 0 to 0.3 integrity. We further divide segment integrity categories to High (0.3-1) and Low (0-0.3) groups. It is interesting that *Spatio-Temporal* actually increase errors a little bit for the High group; we haven't figure out the reason and will leave this to future work. However, its errors for Low group is less than 60% of that of *Baseline*. *Weighted Spatio-Temporal* is even superior: its errors for Low group is less than 40% of that of *Baseline*. Even for High group, *Weighted Spatio-Temporal* successfully reduces errors by more than 5%. Overall, our approach reduces errors to nearly half of Zhu's baseline approach (0.73 v.s. 1.33).

People may notice that the average error of Low group is still as high as 1.04 even for *Weighted Spatio-Temporal*: there are a little portion of exceptional error entries, which push the average value higher; however, they could be easily filtered in the post-processing phase; we leave the reasoning of such exceptional entries to future work.

C. Impact of Integrity

In this part we evaluate the algorithm performance under different geiven TCM integrities. The timeslot is set to 15 minutes, and 4 matrix integrities are evaluated: 0.2, 0.3, 0.4 and 0.5.

The overall NMAE values of each algorithm are shown in Figure 8(a). With the increase of matrix integrity, the errors of *Baseline* and *Spatio-Temporal* decrease too; while *Weighted Spatio-Temporal* always maintain at low error level. The same is the Low group estimation quality shown in Figure 8(b). It is possible that our approach has hit a performance bottleneck: there are randomness in traffic condition.

D. Impact of Time Granularity

We also evaluate our approach under different time granularities: 15, 30 and 60 minutes. The increase in the timeslot length leads to more updates in a single timeslot, we expect the error would decrease.

Figure 8(c) presents the results. Consistent with our analysis, the performance of *Baseline* increases. While *Spatio-Temporal* and *Weighted Spatio-Temporal* maintain stable. It might because the temporal factor in these two approaches already considers the information from neighboring times.

E. Hypothetic Ground Truth

In this part we want to continue the discussion of "ground truth". As mentioned above, the estimation quality is measured by the error of all entries in $D_{dropped} = D_{orig} - D_{part}$ compared with their interpolated values in $\widehat{X}$. If the conjecture is correct, $D_{dropped}$ is not a reliable ground truth.

By applying Algorithm 1 to $D_{dropped}$, a "Hypothetic Ground Truth" $D'_{dropped}$ is got. We redo the experiments in Section VI-B against this "Hypothetic Ground Truth" and the results are shown in Table IV.

It is interesting that compared with Table III, every number is better. However we still consider this $D'_{dropped}$ as *Hypothetic*; we leave the dilemma of how to find a better ground truth for evaluation purpose to our future work.

VII. Related Work

A. Traffic Monitoring with Probe data

A growing number of vehicles and smartphones now embed GPS sensors. Probe data from such devices as well as WiFi and GSM sensors has become ubiquitous. It raises increasing interest in monitoring traffic condition through such location information.

Ferman *et al.* builds an architecture of real-time traffic monitoring system and they develop a simple analytical/statistical model to test the feasibility of this system [13]. They identify the difference between traffic monitoring on freeways and surface roads that the latter requires higher rate of penetration.

Yoon *et al.* propose a method that can characterize traffic patterns and can identify traffic states for a specific road segment [14]. Also they reveal underlying road condition is consistent from analysis on abundant data.

Fabritiis *et al.* present a large-scale working application using real-time floating car data to deliver real-time traffic speed information in Italy [15]. For online short-term traffic prediction, they design two algorithms based on Pattern Matching and Artificial Neural Network respectively, to utilize spatial and temporal average speed information in traffic forecasting.

Thiagarajan *et al.* propose VTrack [16] to measure and locate travel delay. Instead of GPS, they use less power-consumption sensors such as GSM and WiFi to estimate both user's trajectory and travel time. VTrack uses a hidden Markov model-based map matching scheme and travel time estimation method to identify probable routes.

These previous researches seldom have concerns on insufficient samples when comparing limited probe vehicles to immense urban road networks, while our work intends to provide an ubiquitously available traffic information with high coverage of urban cities.

B. Traffic Estimation via Sparse Data

Some studies have been devoted to methodologies of traffic estimation using sparse probe data.

Williams et al. [12] present the theoretical basis for modeling univariate traffic condition data for road segment and use Seasonal ARIMA process for time-series traffic estimation. They reveal traffic condition hides a consistent weekly pattern and they assume one-week lag is the first seasonal difference. However, they share few concerns on sparsity inside the traffic condition and they base their research on freeway traffic, while our works focus on sparse urban traffic.

Zhang et al. [11] introduce compressive sensing technique into internet traffic estimation which is a similar case to urban traffic condition estimations. They examine spatio-temporal structure in network traffic and propose a hybrid algorithm incorporating global spatio-temporal properties and local interpolation and perform experiments over various network traffic matrix with different characteristics.

Zhu et al. [7] adapt a compressive sensing approach to urban traffic estimation with probe vehicles. They reveal hidden structure under urban traffic probe data with principal component analysis (PCA) and singular value decomposition (SVD) and then recover the sparse traffic matrix with compressive sensing which is described as the baseline algorithm in this paper.

We proceed deeper data analysis than previous research and reveal uneven distribution and unreliability inside probe data and analyze the impact of them. Also we prove that average error as an evaluation on previous methodologies can be a misleading metric due to characteristics of urban road network. Further, we present our method which incorporating spatio-temporal information of urban traffic matrix and unbias historical information.

VIII. Conclusion

Our approach exploit the hidden spatio-temporal relationship in the TCM, use weighted average between historical and probed data together to improve the reliability of the input values. Demonstrated by extensive experiments, the estimation error of our approach is around half of a baseline approach.

Acknowledgment

The authors would like to thank anonymous reviewers for their valuable comments. This work is supported in part by "National Natural Science Foundation of China (No. 61100220, No. 61202107, No. 61202303)", and by the "Fundamental Research Funds for the Central Universities", and by NSF under grant CCF-1016966.

References

[1] B. Coifman, "Estimating travel times and vehicle trajectories on freeways using dual loop detectors," *Transportation Research Part A: Policy and Practice*, vol. 36, no. 4, pp. 351–364, 2002.

[2] M. Bramberger, J. Brunner, B. Rinner, and H. Schwabach, "Real-time video analysis on an embedded smart camera for traffic surveillance," in *Real-Time and Embedded Technology and Applications Symposium, 2004. Proceedings. RTAS 2004. 10th IEEE*. IEEE, 2004, pp. 174–181.

[3] A. I. Bejan and R. J. Gibbens, "Evaluation of velocity fields via sparse bus probe data in urban areas," in *Intelligent Transportation Systems (ITSC), 2011 14th International IEEE Conference on*. IEEE, 2011, pp. 746–753.

[4] R. Herring, A. Hofleitner, P. Abbeel, and A. Bayen, "Estimating arterial traffic conditions using sparse probe data," in *Intelligent Transportation Systems (ITSC), 2010 13th International IEEE Conference on*. IEEE, 2010, pp. 929–936.

[5] F. Zheng, H. Van Zuylen, L. Xia, and Y. Chen, "Investigating the feasibility of urban link travel time estimation based on probe vehicle data," in *International Conference on Transportation Engineering*, 2009, pp. 3387–3392.

[6] Y. Zhu, Z. Li, H. Zhu, M. Li, and Q. Zhang, "A compressive sensing approach to urban traffic estimation with probe vehicles," *IEEE Transactions on Mobile Computing*, vol. 99, no. PrePrints, p. 1, 2012.

[7] Z. Li, Y. Zhu, H. Zhu, and M. Li, "Compressive sensing approach to urban traffic sensing," in *Proceedings of the 2011 31st International Conference on Distributed Computing Systems*, 2011, pp. 889–898.

[8] R. Long Cheu, C. Xie, and D.-H. Lee, "Probe vehicle population and sample size for arterial speed estimation," *Computer-Aided Civil and Infrastructure Engineering*, vol. 17, no. 1, pp. 53–60, 2002.

[9] R. Bell, Y. Koren, and C. Volinsky, "Chasing $1,000,000: How we won the Netflix progress prize," *ASA Statistical and Computing Graphics Newsletter*, vol. 18, no. 2, pp. 4–12, 2007.

[10] D. L. Donoho, "Compressed sensing," *Information Theory, IEEE Transactions on*, vol. 52, no. 4, pp. 1289–1306, 2006.

[11] Y. Zhang, M. Roughan, W. Willinger, and L. Qiu, "Spatio-temporal compressive sensing and internet traffic matrices," in *ACM SIGCOMM Computer Communication Review*, vol. 39, no. 4. ACM, 2009, pp. 267–278.

[12] B. M. Williams and L. A. Hoel, "Modeling and forecasting vehicular traffic flow as a seasonal arima process: Theoretical basis and empirical results," *Journal of Transportation Engineering*, vol. 129, no. 6, pp. 664–672, 2003.

[13] M. A. Ferman, D. E. Blumenfeld, and X. Dai, "An analytical evaluation of a real-time traffic information system using probe vehicles," in *Intelligent Transportation Systems*, vol. 9, no. 1. Taylor & Francis, 2005, pp. 23–34.

[14] J. Yoon, B. Noble, and M. Liu, "Surface street traffic estimation," in *Proceedings of the 5th international conference on Mobile systems, applications and services*. ACM, 2007, pp. 220–232.

[15] C. De Fabritiis, R. Ragona, and G. Valenti, "Traffic estimation and prediction based on real time floating car data," in *Intelligent Transportation Systems, 2008. ITSC 2008. 11th International IEEE Conference on*. IEEE, 2008, pp. 197–203.

[16] A. Thiagarajan, L. Ravindranath, K. LaCurts, S. Madden, H. Balakrishnan, S. Toledo, and J. Eriksson, "Vtrack: accurate, energy-aware road traffic delay estimation using mobile phones," in *Proceedings of the 7th ACM Conference on Embedded Networked Sensor Systems*. ACM, 2009, pp. 85–98.