

Use of Probe Data for Arterial Roadway Travel Time Estimation and Freeway Medium-term Travel Time Prediction



**Morgan State University
The Pennsylvania State University
University of Maryland
University of Virginia
Virginia Polytechnic Institute & State University
West Virginia University**

**The Pennsylvania State University
The Thomas D. Larson Pennsylvania Transportation Institute
Transportation Research Building ❖ University Park, PA 16802-4710
Phone: 814-865-1891 ❖ Fax: 814-863-3707
www.mautc.psu.edu**

Use of Probe Data for Arterial Roadway Travel Time Estimation and Freeway Medium-term Travel Time Prediction

Prepared by:

Hesham Rakha

Samuel Reynolds Pritchard Professor of Engineering

Hao Chen

Research Associate

Center for Sustainable Mobility, Virginia Tech Transportation Institute
3500 Transportation Research Plaza
Blacksburg, VA 24061

Ali Haghani

Professor

Xuechi Zhang

Graduate Research Assistant

Masound Hamedi

Assistant Research Scientist

Department of Civil and Environment Engineering, University of Maryland
1173 Glenn L. Martin Hall
College Park, MD 20742

December 2015

1. Report No. MAUTC-2013-05	2. Government Accession No.	3. Recipient's Catalog No.	
4. Title and Subtitle Use of Probe Data for Arterial Roadway Travel Time Estimation and Freeway Medium-term Travel Time Prediction		5. Report Date December 2015	
		6. Performing Organization Code	
7. Author(s) Hesham Rakha, Hao Chen, Ali Haghani, Xuechi Zhang and Masoud Hamedi		8. Performing Organization Report No.	
9. Performing Organization Name and Address Virginia Polytechnic Institute & State University Blacksburg, VA 24061 University of Maryland College Park, MD 20742		10. Work Unit No. (TRAIS)	
		11. Contract or Grant No. DRTR12-G-UTC03	
12. Sponsoring Agency Name and Address US Department of Transportation Research & Innovative Technology Admin UTC Program, RDT-30 1200 New Jersey Ave., SE Washington, DC 20590		13. Type of Report and Period Covered Final, - 7/1/2013 - 12/31/2014	
		14. Sponsoring Agency Code	
15. Supplementary Notes			
16. Abstract Urban traffic congestion is a problem that plagues many cities in the United States. One approach to alleviating this congestion is to provide drivers with better travel time information so that they can make better departure time and routing decisions. This research project focuses on two efforts: (1) validating the use of probe data to estimate arterial travel times and (2) validating and developing techniques to predict freeway travel times over a 2 to 4 hour window. With regards to the arterial travel time estimation the objective are two-fold. The first goal is a comprehensive validation of INRIX arterial data. In the validation the variability and reliability of arterial data in different corridors equipped with permanent and portable traffic sensors will be studied under different traffic conditions. The second goal is to develop a methodology for augmenting INRIX data with other data sources in order to improve the data quality. With regards to freeway travel time prediction the objectives are also two-fold. The first objective is to validate the INRIX travel time prediction algorithms along a number of corridors in the state of Virginia. The second objective is to enhance the prediction accuracy using various pattern recognition and machine learning techniques.			
17. Key Words Arterial travel time estimation, freeway travel time prediction, data fusion, probe data, particle filter, agent based modeling, dynamic template matching		18. Distribution Statement No restrictions. This document is available from the National Technical Information Service, Springfield, VA 22161	
19. Security Classif. (of this report) Unclassified	20. Security Classif. (of this page) Unclassified	21. No. of Pages 83	22. Price N/A

Disclaimer

The contents of this report reflect the views of the authors, who are responsible for the facts and the accuracy of the information presented herein. This document is disseminated under the sponsorship of the U.S. Department of Transportation's University Transportation Centers Program, in the interest of information exchange. The U.S. Government assumes no liability for the contents or use thereof.

Table of Contents

Study Background	1
Paper 1: Arterial travel time validation and augmentation with two independent data sources (published in <i>Transportation Research Record</i>).....	1
Paper 2: Real-time travel time prediction by particle filtering with a non-explicit state- transition model (published in <i>Transportation Research Part C: Emerging Technologies</i>).....	1
Paper 3: Multi-step Prediction of Experienced Travel Times using Agent-based Modeling (under consideration for publication).....	2
Paper 4: Predicting Freeway Travel Times using Dynamic Template Matching (under consideration for publication).....	2
APPENDIX A	4
APPENDIX B	19
APPENDIX C	41
APPENDIX D	60

STUDY BACKGROUND

Urban traffic congestion in the United States is a significant drain on productivity and the environment. One study estimates that urban drivers in the United States spend about 36 hours annually stuck in congestion and that this results in a waste of about 24 gallons of fuel. While in the past this congestion has been mitigated by expanding the roadway network, roadway infrastructure investments are significantly expensive and have been shown in some cases to actually exacerbate congestion. With the introduction of vehicle connectivity, vehicles will be able to share their experiences creating a wealth of traffic and travel time data. This wealth of vehicle probe data can be used to estimate existing traffic conditions and more importantly predict future conditions. Therefore, this research effort focuses on validating probe vehicle data for the estimation of arterial roadway travel times and the prediction of freeway travel times.

This report consists of four research efforts that have either been published in academic journals or is in consideration for publication. The remainder of this report provides a short description of each of these studies. The manuscript of each of these studies are included in the Appendix of this report.

Paper 1: Arterial travel time validation and augmentation with two independent data sources (published in *Transportation Research Record*)

Travel time data is a key input to Intelligent Transportation Systems (ITS) applications. Advancement in vehicle tracking and re-identification technologies and proliferation of location aware and connected devices has made network wide travel time data available to transportation management agencies. The trend started with data collection on freeways has been quickly extended to arterials. Although the freeway travel time data has been validated extensively in recent years, the quality of arterial travel time data is not well known. This paper presents a comprehensive validation scheme for arterial travel time data based on GPS probe and Bluetooth generated data as two independent sources. Since travel time on arterials is subject to a higher degree of variation compared to freeways mainly due to presence of signals, a new validation methodology based on coefficient of variation is introduced. Moreover, a Context Dependent (CD) based travel time fusion framework is developed to improve the reliability of travel time information by fusing data from multiple sources. The entire 2012 data on a busy arterial corridor in Maryland has been used to demonstrate the proposed comparison and augmentation model.

The manuscript of this study is attached in Appendix A.

Paper 2: Real-time travel time prediction by particle filtering with non-explicit state-transition model (published in *Transportation Research Part C: Emerging Technologies*)

The research presented in this paper develops a particle filter approach for the real-time short to medium-term travel time prediction using real-time and historical data. Given the challenges in defining the particle filter time update process, the proposed algorithm selects particles from a historical database and propagates particles using historical data sequences as opposed to using a state-transition model. A partial resampling strategy is then developed to address the degeneracy

problem by replacing invalid or low weighted particles with historical data that provide similar data sequences to real-time traffic measurements. As a result, each particle generates a predicted travel time with a corresponding weight that represents the level of confidence in the prediction. Consequently, the prediction can produce a distribution of travel times by aggregating all weighted particles. A 95-mile freeway stretch from Richmond to Virginia Beach along I-64 and I-264 is used to test the proposed algorithm. Both the absolute and relative prediction errors using the leave-one-out cross validation concept demonstrate that the proposed method produces the least deviation from ground truth travel times, compared to instantaneous travel times, two Kalman filter algorithms and a K nearest neighbor (k-NN) method. Moreover, the maximum prediction error for the proposed method is the least of all the algorithms and maintains a stable performance for all test days. The confidence boundaries of the predicted travel times demonstrate that the proposed approach provides good accuracy in predicting travel time reliability. Lastly, the fast computation time and online processing ensure the method can be used in real-time applications.

The manuscript of this study is attached in Appendix B.

Paper 3: Multi-step Prediction of Experienced Travel Times using Agent-based Modeling (under consideration for publication)

This paper develops an agent-based modeling approach to predict multi-step ahead experienced travel times using real-time and historical spatiotemporal traffic data. At the microscopic level, each agent represents an expert in the decision-making system. Each expert predicts the travel time for each time interval according to experiences from a historical dataset. A set of agent interactions is developed to preserve agents that correspond to traffic patterns similar to the real-time measurements and replace invalid agents or agents associated with negligible weights with new agents. Consequently, the aggregation of each agent's recommendation (predicted travel time with associated weight) provides a macroscopic level of output - the predicted travel time distribution. Probe vehicle data from a 95-mile freeway stretch along I-64 and I-264 is used to test different predictors. The results show that the agent-based modeling approach produces the least prediction error compared to other state-of-practice and state-of-art methods (instantaneous travel time, historical average and k-nearest neighbor), and maintains less than a 9% prediction error for trip departures up to 60 minutes into the future for a two-hour trip. Moreover, the confidence boundaries of the predicted travel times demonstrate that the proposed approach also provides high accuracy in predicting travel time confidence intervals. Finally, the proposed approach does not require offline training thus making it easily transferable to other locations and the fast computation allows the proposed approach to be implemented in real-time applications in Traffic Management Centers.

The manuscript of this study is attached in Appendix C.

Paper 4: Predicting Freeway Travel Times using Dynamic Template Matching (under consideration for publication)

The paper develops a predictive travel time algorithm using dynamic template matching to identify similar spatiotemporal trends in a historical dataset for use in prediction purposes. Unlike previous approaches, which use fixed template sizes, the proposed method uses a

dynamic template size that is updated each time interval based on the spatiotemporal shape of the congestion upstream of the bottleneck. In addition, the computational cost is reduced using a Fast Fourier Transform instead of Euclidean distance. Subsequently, the historical candidates that are similar to the current conditions are used to predict the experienced travel times. The proposed method is tested on a freeway stretch along I-64 using five-minute aggregated probe data provided by INRIX. The study demonstrates that the proposed method produces significantly better and more stable prediction results for prediction horizons up to 30 minutes into the future compared to instantaneous and fixed template matching methods. Furthermore, a comparison of the fixed-template and dynamic-template methods indicates that the dynamic template enhances the prediction accuracy at the shoulders of the congested periods. Finally, the proposed dynamic template matching approach has the flexibility of using an incremental historical dataset, which is demonstrated to further improve the prediction accuracy over the use of a fixed historical dataset.

The manuscript of this study is attached in Appendix D.

APPENDIX A

Arterial travel time validation and augmentation with two independent data sources

This article may be cited as: Xuechi Zhang, Masoud Hamed, Ali Haghani, "Arterial Travel Time Validation And Augmentation with Two Independent Data Sources", Accepted for Publication in *Transportation Research Record*, Journal of Transportation Research Board, 2015.

INTRODUCTION

Urban traffic congestion has become a common recurrent phenomenon in most metropolitan areas. Transportation agencies utilize Intelligent Transportation Systems (ITS) to provide the travelling public with reliable and real-time traffic information in order to improve mobility. Travel time is a vital component of such systems since it is a direct indicator of delay and is easily understood by general public. In recent years several vehicle tracking and identification technologies including Automatic License Plate Reader (ALPR), Electronic Toll Tag matching, Bluetooth and WiFi detectors have been successfully developed to measure and communicate travel time data [1, 2, 3, 4].

Recent advancements in vehicle tracking technologies along with dramatic increase in number of location aware and internet enabled mobile devices carried by travelers has created new possibilities for collecting and reporting travel time data in large scale. Private sector companies such as INRIX take advantage of these resources to provide real-time information both on arterials and freeways mainly by capturing, consolidating and filtering GPS tracks reported by such devices [5]. In addition, Bluetooth (BT) travel time collection technic is proven to be a success due to its low cost and high privacy protection properties [6, 7]. Quality of both probe and BT freeway data have been extensively validated and examined in recent years [8]. However the quality of arterial data is not well known. This is due to the nature of traffic in the arterials which is heavily impacted by intersections as well as signal timing scheme on a given corridor [9, 10]. Moreover, lower traffic volumes and larger variance in travel time introduces unique challenges to the arterial performance measurement compared to freeways [11]. Therefore, it is necessary to develop proper quality assessment methods for arterial travel time data considering its characteristics. Meanwhile, when travel time data is available from multiple sources, the possibility and usefulness of merging such data in order to increase reliability of travel time on arterials needs to be investigated. The following is a summary of major efforts to address the abovementioned issues.

In the first validation report prepared for the I-95 Vehicle Probe Project (VPP), data is divided into four classes (i.e. speed bins) based on the observed mean speed in each time interval, then verification is performed for each category [8]. With the same purpose, a Paired-t Method is proposed as an alternative approach to validate INRIX reported data with BT datasets, and this method is shown to be effective when there are very few ground truth observations [12]. Although a few arterial validation studies are conducted for the VPP based on the same methodology, majority of validation is focused on freeways. Data post-analysis and consolidation are the key component to provide users with more reliable data. Since more and more independent traffic related data sources have emerged recently, data fusion is becoming a popular approach to combine data in order to achieve higher accuracy and resolution. A comprehensive survey in terms of data fusion progress and challenges in Intelligent Transportation Systems is reported in Faouzi et al. [13]. Based on the characteristics of the fused data, data sources can be further classified into two categories, i.e. heterogeneous and parallel. From the perspective of heterogeneous fusion, Anusha et al. [9] fused location based flow data and sparse travel time data obtained from probe vehicles to determine the stream flow travel speed. In addition, fusion models with heterogeneous data from underground loop detector and GPS-equipped probe vehicles are also proposed for urban arterial corridors [14, 15]. When it comes to fusion techniques with parallel datasets, Soriguera et al. [15] took advantage of Context

Dependent (CD) based fusion operator [16], which is well adopted in the field of image processing, to generate fuzzed travel time in a conservative way. However, data reliability and consistency are not addressed in their work.

Based on the aforementioned work, contributions of this paper are twofold. First, a new coefficient-of-variation (CV) based travel time validation scheme is proposed to compare and validate the GPS probe data reported by INRIX against the BT travel time for arterials. In addition to time of day impact, different traffic conditions are considered in the analysis. Second, a Context Dependent (CD) based travel time fusion framework is developed by using data from INRIX and BT datasets to improve the reported data quality. Although the proposed fusion framework is examined on INRIX and BT as two independent data sources, it can be flexibly modified and extended to any other type of data.

VALIDATION MODEL

Spatial and Temporal Alignment of Validation Segment

Comparing and validating reported travel time of one data source by using data from another data source requires both spatial and temporal alignment of the validation segment. Travel time data provider companies that utilize probe technology usually report data on Traffic Message Channel (TMC) codes. A travel path consists of one or multiple consecutive TMC segments. For each time interval, an estimated average travel time data point is accompanied by a data confidence score for every TMC segment. On the other hand, location of sensors and configuration of segments is more flexible when using Bluetooth detectors [6, 7, 8]. In order to make data comparable between the sources, it is important to deploy Bluetooth sensors in line with the corresponding TMC segments as shown in figure 1.

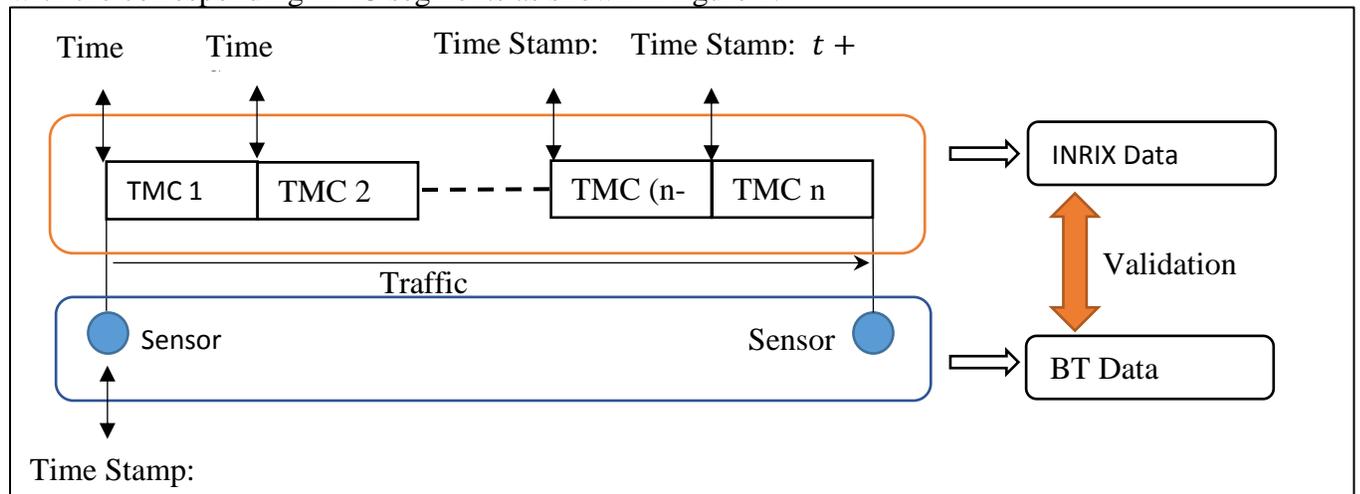


FIGURE 1 Spatial and Temporal Alignment of the Validation Segment.

General definition of path travel time at a particular time point t is the duration of time that a vehicle spends to get through the segment given the entrance time equals to t . Based on this definition, the measured travel time by BT is exactly the data of interest. This is not the case for the INRIX dataset since INRIX only provides travel time of each TMC segment, which is usually part of a multi-segment path. Hence, equivalent path travel time from INRIX must be calculated by consolidating data for the TMC segments in the path. A simple summation of travel time of the TMC's for the same time interval is not consistent with the real path travel time.

To obtain the real path travel time, a backtracking algorithm is used, which can be described by the following recursive equations.

$$F(k, t) = F(k - 1, t) + T[k, t + F(k - 1, t)], k = 2, 3, \dots, N \quad (1)$$

$$F(1, t) = T(1, t) \quad (2)$$

where, $T(k, t)$ is the reported travel time of the k^{th} TMC at time point t , and $F(k, t)$ is the real path travel time from the first TMC to the k^{th} at time point t . N is the total number of the aligned TMCs. Path travel time from the start of the first segment to the end of the last TMC segment can be obtained by calculating $F(N, t)$, given the starting time point t (or time interval t).

Coefficient of Variation (CV) based Validation

Data validation is the process of ensuring that the target data set meets certain quality measures when compared against ground truth. This section describes two general statistical validation methods that have been used for freeway data and proposes a new method that emphasizes on travel time variability as a main characteristic of arterial travel time.

Validation Method 1: Aggregate Mean Comparison

A typical way to compare one time series against another time series is a pairwise data point subtraction where mean absolute percentage difference (MAPD) is defined as an indicator to quantitatively measure the difference.

$$MAPD_S = \frac{1}{|S|} \sum_{\forall i \in S} \frac{|T_{INRIX}^i - T_{BLT}^i|}{T_{BLT}^i} \quad (3)$$

Where, T_{INRIX}^i and T_{BLT}^i are the reported travel time from INRIX and BT datasets at time point (or interval) i , respectively. $|S|$ is the size of the validation data set S . As the name suggests, MAPD yields an average of the absolute difference (i.e. difference) between the validated dataset and the ground truth dataset. Higher value of MAPD is an indicator of deviation from the ground-truth. MAPD is usually calculated separately for different categories of data (e.g. time of day, day of week or traffic condition).

Validation Method 2: t-test Based Comparison

Another effective way to validate the reported travel time by using independent detection samples is t-test method. BT dataset reports individual travel time of each valid detected vehicle within a specific time interval. Thus the hypothesis that *reported INRIX travel time is significantly different from the BT mean travel time* for any specific time interval with a valid set of BT observations can be statistically tested. The mean travel time confidence band is formed using the following equation.

$$B_{n,100(1-\alpha)}^i = \bar{T}_n^i \pm t_{n-1,1-\alpha/2} \cdot \sqrt{Var(T_n^i)} \quad (4)$$

$$Var(\bar{T}_n^i) = \frac{\sum_j (T_j^i - \bar{T}_n^i)^2}{n(n-1)} \quad (5)$$

Where, n is the number of samples within time interval i . T_j^i is the observed travel time of j^{th} sample in this time interval. \bar{T}_n^i is the sample mean, and $Var(\bar{T}_n^i)$ is the corresponding variance of sample mean. $t_{n-1,1-\alpha/2}$ is the student t-test value for degree $(n-1)$ and $100(1 - \alpha)$ percent confidence. If the reported INRIX travel time value T_{INRIX}^i is located in this confidence band, the hypothesis is rejected. In other words, it can be concluded that the difference between reported INRIX travel time and the BT mean travel time is not statistically significant for target

interval. The percentage of time intervals of category S that pass the test can be reported as the acceptance ratio for this category, which is calculated in equation (6). For any specified paired category (validation subsets), the higher the acceptance ratio is, the more similar the paired data is.

$$Accept_Ratio_S = 100\% \cdot \frac{1}{|S|} \cdot \sum_{i \in S} 1_{\{T_{INRIX}^i \in B_{n,100(1-\alpha)}^i\}} \quad (6)$$

Where, $1_{\{State(i)\}}$ is indicator function yielding 1 when statement $State(i)$ is true and 0 otherwise.

Validation Method 3: Travel Time Variation Categorization

Validation methods 1 and 2 have been widely used to validate freeway data. The measures have also been reported for several data categories based on “time of day”, “day of week” or “speed bins”. Vehicle probe data reported by INRIX consists of a single data point and does not show variability of travel time. However Bluetooth travel time data is generated by aggregating several travel time observations for each specific interval. This allows calculation of travel time variance in addition to simple average. Since freeway segments are not subject to major flow disruptions caused by intersections and traffic signals, travel time variance across time intervals is not significant and thus is not useful for categorizing data. On the other hand for an arterial corridor, travel time variability can be significant. Validation method 3 takes advantage of this characteristic to divide data into subsets formed by their degree of variation. Coefficient of variation (CV) is an effective indicator to quantify travel time variability based on the detected samples. CV is defined as the ratio of the standard deviation to the mean, and is considered a normalized measure of dispersion of a probability distribution. From the Bluetooth data, average \bar{T}_{BLT}^t and standard deviation σ_{BLT}^t for each time interval t is obtained by aggregating valid travel time samples of detected vehicles (Equation 7).

$$CV^t = \frac{\sigma_{BLT}^t}{\bar{T}_{BLT}^t} \quad (7)$$

Travel time reported for intervals with lower CV is considered more reliable and is an indication of a more stable traffic pattern. Consequently, the validation results with lower difference for intervals with smaller CV are more desirable. Therefore, we propose to use the CV indicator as a classification threshold to further construct and categorize the validation set S in order to describe validation results in a new format. Given the entire time series dataset for a particular time period (e.g. one month or a year), the validation set is divided based on the following set classification operator

$$S_{time_n, CV_m} = \{T_{INRIX}^i \text{ and } T_{BLT}^i | \forall i \in time_n \text{ and } CV^t \in CV_m\} \quad (8)$$

Where, S_{time_n, CV_m} is the target validation data set with “time of the day” and “day of the week” specified as $time_n$ and traffic variation limited within CV interval CV_m . Hence, validation of the INRIX travel time data compared to BT travel time data can be conducted in different scenarios with respect to different traffic variability states as well as different time of day. This method is applied to a case study and the results are discussed later in the paper.

DATA FUSION METHODOLOGY

This section describes a data fusion framework for blending GPS probe and Bluetooth generated travel time data for an arterial path. The objective of such approach is twofold. First, it can

increase temporal data coverage by benefiting from the complementation of multiple data sources. Secondly, by taking advantage of the data fusion logic, the accuracy of the estimated travel time will be enhanced.

Data Reliability Considerations

Data provided by different sources can be either parallel or heterogeneous. For instance, both INRIX and Bluetooth sources may provide average travel time of a particular time interval in parallel. Meanwhile, each data source comes with other data elements that describe travel time data. As for INRIX dataset, average travel time is always accompanied with another numerical indicator called confidence score which has a value of 10, 20 or 30. The higher the confidence score, the more reliable is the reported travel time. On the Bluetooth data set, in addition to average travel time other indicators such as number of samples and variance around mean can be calculated. One simplistic approach to data fusion is to calculate and report average travel time obtained from the two sources without considering other factors. However to increase the reliability and accuracy of data fusion engine, other valuable information such as confidence score and variation must be brought into the framework.

As mentioned before, the reported travel time of INRIX dataset is based on TMC. The corresponding confidence score takes value from $\{10, 20, 30\}$. When generating travel time of a particular path for a specific time stamp, a weighted average of confidence scores for the TMC codes across the path must be calculated. That is, for a studied path consisting of n consecutive TMC segments, the expected confidence score for time point t is calculated as,

$$conf_t = average\{c_t^1, c_{t+T_{1-2}^t}^2, c_{t+T_{1-3}^t}^3, \dots, c_{t+T_{1-n}^t}^n\} \quad (9)$$

Where, c_t^j is the reported confidence score of the j th TMC segment at time point t and belongs to $\{10, 20, 30\}$, and T_{1-j}^t denotes the travel time from the start of 1st TMC segment to the start of j th TMC segment at time point t . In other words, T_{1-j}^t is just the travel time of the path consisting of TMC segments, 1, 2, \dots , $(j-1)$ measured at time point t .

When it comes to the Bluetooth data, since Bluetooth detectors are located at both ends of the path, coefficient of variation (CV) as well as number of samples for each time interval can be calculate based on travel time samples belonging to the interval. Intuitively speaking, when the CV is high and the number of detections is relatively low, the corresponding measured travel time (the mean or the median travel time) might not be reliable. Instead, when the CV is low and the number of detections is relatively high, we have much more confidence in the measured travel time drawn from these detection samples.

Context Dependent Based Fusion Operator

Any data augmentation and fusion method can be classified as a specific type of fusion operator depending on its fusion behavior [16]. Bloch [16] proposed a classification of the operators in three classes and further showed that any specific operator fits in one of the classes. They are Context Independent Constant Behavior (CICB) Operators, Context Independent Variable Behavior (CIVB) Operators, and Context Dependent (CD) Operators. In this section, a CD fusion operator to fuse and augment the travel time with INRIX and Bluetooth datasets is proposed. In this operator not only the value itself plays an important role in the fusion process, but also data source reliability and data conflicts are taken into consideration.

As discussed previously, the average confidence score obtained from INRIX dataset can reveal some reliability information on the reported travel time. Thus we use it as an indicator to

quantitatively describe the reliability of the reported travel time value within a given time interval (or at a given time point). The context proposed here is a binary logic, where 1 means the reported data is reliable and 0 means unreliable.

$$R_{INRIX}^t = \begin{cases} 1, & \text{if } conf_t \geq \alpha \\ 0, & \text{if } conf_t < \alpha \end{cases} \quad (10)$$

Where R_{INRIX}^t is the INRIX data reliability indicator of the studied path at the time interval t . α is a user defined threshold and takes value within [10, 30].

Within a specific time interval, number of detections (or observations) is viewed as a significant indicator of the reliability when it comes to the Bluetooth data. Given all of the valid observations within time interval t , CV reflects the variation of the traffic state. Thus it can also be used as a proxy for travel time reliability. Higher number of observations is an indicator of a more reliable Bluetooth estimated travel time in a given time interval.

$$R_{BLT}^t = \begin{cases} 1, & \text{if } N_t \geq k(i) \\ 0, & \text{if } N_t < k(i) \end{cases} \quad (11)$$

Where, R_{BLT}^t is the BT data reliability indicator of the studied path at time interval t . N_t is the corresponding detection rate during that time interval. $k(i)$ is a segment-dependent criterion indicating the reliability of the detection data. Discussion related to choosing an appropriate value for $k(i)$ can be found in Haghani et al. (2010). The above binary logic is a basis for fusion and augmentation process when the final target data is generated from multiple independent sources. This is a main advantage of the CD operator since CICB and CIVB operators do not allow consideration of data sources reliability.

Another important issue considered in the proposed framework is conflict and consonance. In some scenarios, even though each independent source claims high reliability of their reported data, there might exist contradicting situations [16]. Hence, a specific fusion mechanism or logic must be developed to address the conflict issue between these so-called high reliable data. An effective way to quantify the conflict between the reported travel time of INRIX and Bluetooth within the same time bin is to investigate the mean distance of these two data points with consideration of CV. A binary logic to make a decision whether the reported INRIX travel time conflicts with the reported data from BT is developed (or whether the data from these two sources are consonant).

$$Consonant^t = \begin{cases} 1, & \text{if } T_{INRIX}^t \in B_{n,100(1-\alpha)}^t \\ 0, & \text{otherwise} \end{cases} \quad (12)$$

Where $Consonant^t$ is a binary indicator with value 1 meaning the reported travel time of INRIX for time interval t is consonant with that of BT detection data and 0 means otherwise. T_{INRIX}^t and T_{BLT}^t are the mean value of travel time from INRIX and BT, respectively. $B_{n,100(1-\alpha)}^t$ denotes the confidence band of mean time interval t . In other words, $Consonant^t$ equal to 1 means the reported value of travel time from INRIX is statistically captured by the BT dataset.

The proposed fusion operator is a context dependent operator defined in Bloch [16]. It is necessary to define all of the possible context combinations. Their definitions are

- NR-NR context:

$$R_{INRIX}^t = 0 \text{ and } R_{BLT}^t = 0$$

- NR-R context:

$$R_{INRIX}^t = 0 \text{ and } R_{BLT}^t = 1$$

- R-NR context:

$$R_{INRIX}^t = 1 \text{ and } R_{BLT}^t = 0$$

- R-R & C context:

$$R_{INRIX}^t = 1 \text{ and } R_{BLT}^t = 1 \text{ and } Consonant^t = 1$$

- R-R & NC context:

$$R_{INRIX}^t = 1 \text{ and } R_{BLT}^t = 1 \text{ and } Consonant^t = 0$$

The last two contexts take the ‘‘consonant’’ logic into consideration. In other words, when any of the multiple data providers shows an unreliable behavior, they are unlikely to be trusted in the current time interval. Therefore, whether the data is consonant with each other has a lower priority in comparison with their reliability.

First Level Fusion Operator

In the context of NR-NR (i.e. both data sources are unreliable), a cautious behavior is taken to fuse their reported travel time value. The cautious fusion behavior has the property that $\forall(x, y) \in R^2, \min(x, y) \leq F(x, y) \leq \max(x, y)$, where $F(x, y)$ is the fusion function with respect to datasets x and y . The function chosen here is simply the unweighted average function, which means that the same belief value is used on each dataset. Similarly, in the context of ‘‘R-R and NC’’, both data source are judged to be reliable while the data they provide conflict with each other, hence the unweighted average is chosen as the fusion function. In the context of ‘‘R-NR’’ and ‘‘NR-R’’ (i.e. one of the data sources is not reliable while the other one is reliable) the data from the reliable data source is chosen. The last context is the most desirable scenario, in which both the data sources are reliable and the data they provide is consonant with each other. Therefore, we can either choose the statistical BT mean or the average of BT and INRIX as the fusion output since the single INRIX data is well captured by the sampling group of BT detections. In some particular time intervals with a high travel time variance, although the reported INRIX data is statistically captured by the BT detection data, the mean difference of these two values can still be large. Thus the unweighted average is chosen as the final fusion operator in context ‘‘R-R and C’’. It is noted that this average value is still within the statistical mean confidence band. Finally, the first level context specific fusion operators are summarized in equation (11). In addition to the fused travel time, another set of fusion outputs, i.e. the belief of fusion, is introduced. This is a significant component of data fusion process that quantitatively indicates the belief of fusion results. As the name suggests, the higher the belief value is, the more trustable the fused data are. The belief is defined in three levels, i.e. with $Bel^t = 0$ meaning the fused result is not reliable, $Bel^t = 1$ meaning the fused result is plausible and $Bel^t = 2$ meaning the fused result is believable. The belief function is given in equation (12). For context ‘‘NR-NR’’, it is concluded that the fusion is not reliable, since neither source is reliable enough. In context ‘‘R-R and NC’’, although both sources are claimed to be reliable, but they are statistically different from each other, thus the fused results from this context is said to be plausible. On the contrary, when both sources are reliable and statistically consonant with each other, the fused result is claimed as believable. As for the other two contexts, i.e. only one data source is reliable, the fused result is plausible by choosing the data from the reliable data source.

$$\begin{aligned}
& T_{fused_1}^t = \\
& F(T_{INRIX}^t, T_{BLT}^t) \\
& = \begin{cases} \frac{T_{INRIX}^t + T_{BLT}^t}{2}, & \text{if } (NR - NR) \text{ or } (R - R \text{ and } NC) \text{ or } (R - R \text{ and } C) \\ \mathbf{1}_{\{R_{INRIX}^t=1\}} \cdot T_{INRIX}^t + \mathbf{1}_{\{R_{BLT}^t=1\}} \cdot T_{BLT}^t, & \text{if } (NR - R) \text{ or } (R - NR) \end{cases} \quad (13)
\end{aligned}$$

$$Bel^t = \begin{cases} 0, & \text{if } (NR - NR) \\ 1, & \text{if } (NR - R) \text{ or } (R - NR) \text{ or } (R - R \text{ and } NC) \\ 2, & \text{if } (R - R \text{ and } C) \end{cases} \quad (14)$$

Where $T_{fused_1}^t$ is the output from the fusion function $F(T_{INRIX}^t, T_{BLT}^t)$, and $\mathbf{1}_{\{statement\}}$ is the binary indicator function w.r.t. a specific statement.

Second Level Fusion Operator

At the first level of fusion process, the fused travel time is calculated in terms of each specific context, which is defined from the perspectives of data source reliability and data conflict. The fusion process is conducted in a conservative way by extracting and combining the information from both datasets (i.e. vertical fusion). In the second level, based on the fusion outputs from the first level, moving average method is chosen to further eliminate the error disturbance along the time line (i.e. horizontal fusion). The fusion operator is given by equation (15).

$$T_{fused_2}^t = \left\{ \frac{\sum_{i=t-(k-1)/2}^{i=t+(k-1)/2} T_{fused_1}^i}{k} \right\} \quad (15)$$

Where, k is a predefined moving distance and $T_{fused_1}^i$ is the 1st level fusion result at time interval (or time point) i.

CASE STUDY

This section presents a case study of the proposed travel time validation and augmentation model with travel time data collected on MD-355 in Maryland for entire year of 2012. A Satellite view of the target arterial segment is presented in **Error! Reference source not found.** The length of the arterial segment is approximately 2.8 miles, northbound from Country Club Rd to College Pkwy, where two Bluetooth detection sensors are deployed respectively. BT travel time data for this segment is collected and processed through these two detection devices with AVI settings. More than four million Bluetooth detections from both sensors were processed to generate more than 110 thousand travel time samples. The GPS probe travel time data for this segment is coming from seven TMC segments which are consecutively aligned on the studied arterial segment. The spatial and temporal alignment method mentioned earlier is used to generate the path travel time data. The travel time data reported by INRIX is reported in one minute intervals. This is not necessary the case for the BT dataset, meaning that Bluetooth samples are available only when at least one valid vehicle traverses the target segment.

Based on the work of Haghani et al (2010), a minimum of three samples per 5-minute is chosen as the reliability threshold, under which the estimated mean travel time is said to be reliable. It is noted that this threshold is not time dependent, since it suggests any 5-min time bin with less than three observations is either very lightly traveled or we just do not have enough samples from the statistical perspective. **Error! Reference source not found.** plots the CV distribution based on 5-minute time bins with respect to samples of weekdays and weekends from BT datasets. The plots indicate around 55% BT detected time bins having CV greater than 0.1, and nearly 25% time bins having CV larger than 0.2. These statistical values suggest travel time data has a high variance in the studied arterial segment. Therefore, further classifying the validation time bins based on their CV value in addition to the time of day is necessary as discussed in validation method 3.

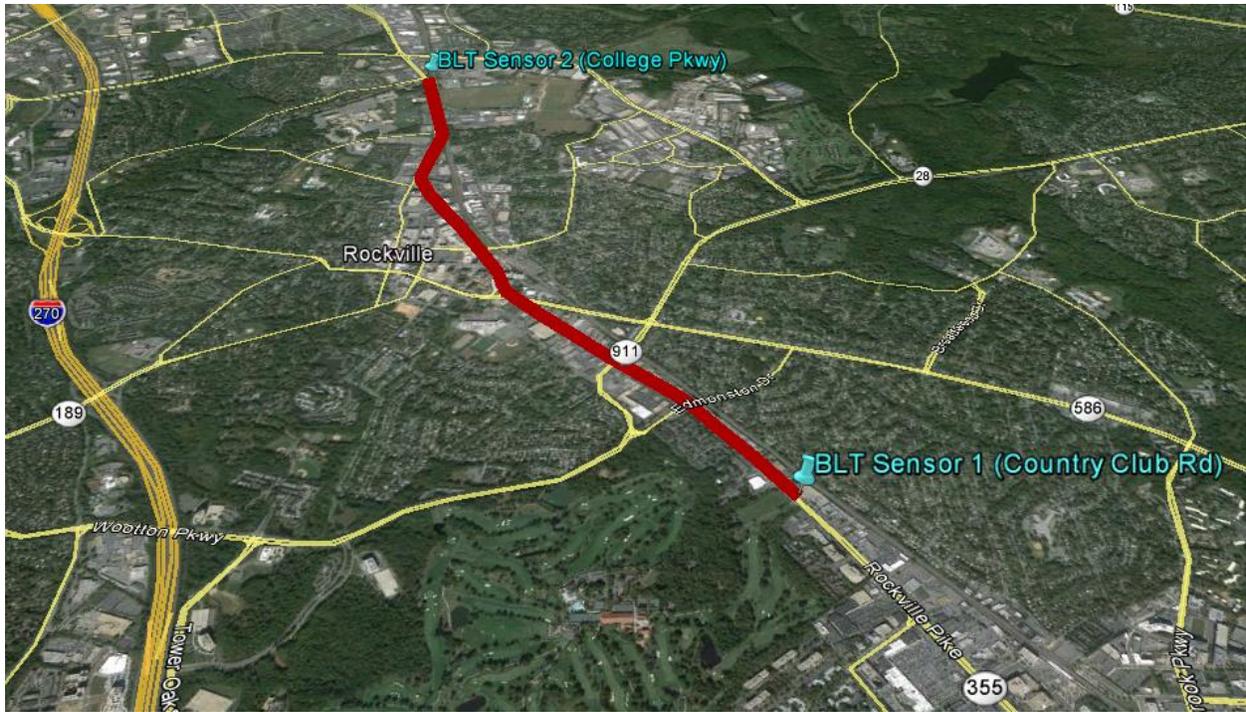


FIGURE 2 Satellite picture of MD-355, northbound from Country Club Rd to College Pkwy.

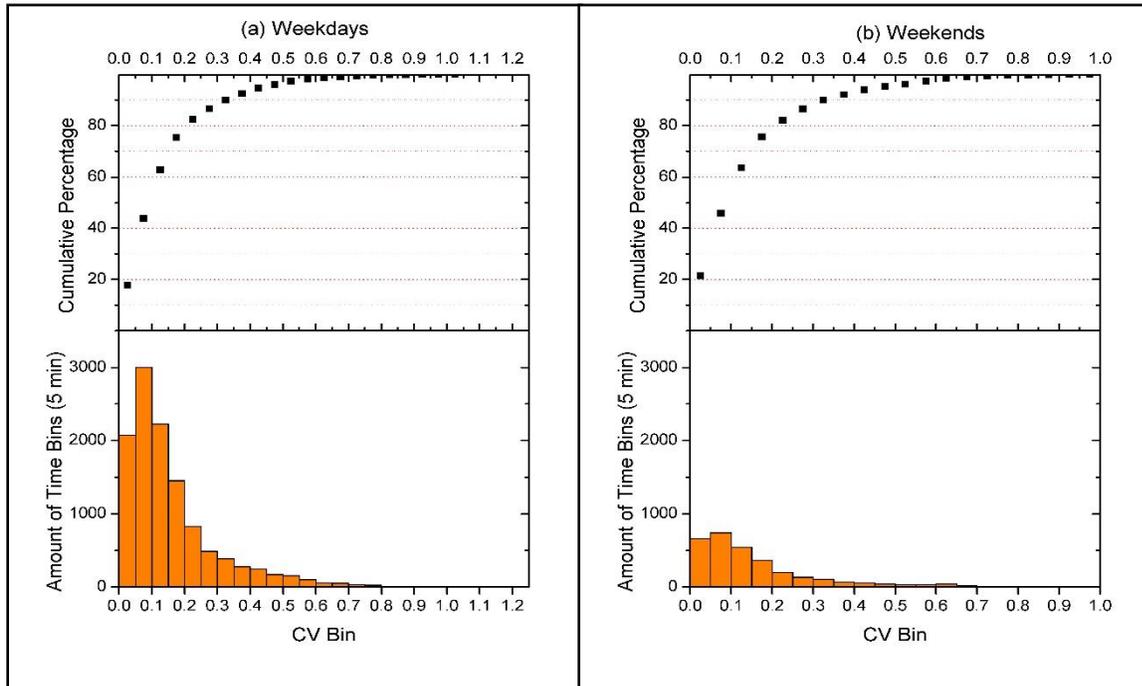


FIGURE 3 Histogram plots of CV distribution based on BT detection data.

TABLE 1 and TABLE 2 show the validation results between INRIX and BT travel time data with the proposed comparison method. Since the studied segment is located in the suburban area which yields different traffic patterns in weekdays and weekends, the validation work is further

separated with respect to weekdays and weekend. The sample time bins are classified according to time of day as well as their CV value obtained from the BT datasets. Since BT time bins with less than three observations are claimed to be unreliable, samples with one or two observations are not classified to any type of CV categories. Instead, validation results from these types of samples are listed out in the first column only for demonstration purpose. From the results yielding by validation method 1 (i.e. MAPD), the difference between INRIX data and BT data is always high in the higher CV area. For example, for verification samples belonging to 4:00-5:00 PM weekday, the MAPD of samples with CV belonging 0.4-0.5 is 35.4%, while this value is only 20.9% for the samples with CV belonging to 0.0-0.1 during this time of day. However, from the perspective of the second validation method (i.e. t-test with 95% confidence band), the acceptance ratio for the above mentioned samples with CV belonging to 0.4-0.5 is 100%. Instead this value is 95% for those with CV belonging to 0.0-0.1. The following key findings are listed according to the numerical results from *TABLE 1* and *TABLE 2*.

- For the studied segment, comparison and validation results vary for weekdays and weekends. For weekdays, the reported INRIX travel time data has a larger deviation from BT data for time period 5:00-8:00 PM, which is the peak hour period. For weekends, the difference is relatively high during the entire daytime, but lower than that of peak hour period of weekdays.
- MAPD method and t-test method display different deviation patterns under different CV categories. For time interval with high travel time variance, although the mean difference between INRIX and BT data is high, the reported travel time from INRIX are more likely to be the same with that of BT detection data.

The proposed data fusion model is also applied to the above arterial segment conceding INRIX and BT datasets of year 2012. The confidence band used to distinguish the “R-R&NC” and “R-R&C” contexts in equation (12) is set to be 95%. The reported path travel time confidence score of INRIX ranges from 20 to 30 due to the scale of the original TMC-based data confidence score. The score threshold to judge the reliability of INRIX data is arbitrarily chosen in this model. *FIGURE 4* plots the percentage of each fusion context among the overall fusion points under different settings of this reliability threshold. The most ideal fusion context is “R-R&C”, where both INRIX and BT data are reliable. Meanwhile, these two reported data are consonant with each other (i.e. INRIX data dropped within the 95% CI band of BT samples). *FIGURE 5* shows part of the fusion data series with the threshold set in a conservative manner (i.e. $\alpha = 25$). The numerical results from the fusion model reveal the following key conclusions.

- Regardless of the specific setting of the reliability threshold of INRIX data, the “R-R&NC” situations are approximately 1/3 of “R-R&C” context for the studied segment. From another perspective, around 75% data points of INRIX and BT are statistically the same given these two data sources behaving in a reliable manner.
- The fusion model performs in a conservative way when combining the data from these two independent data sources. The first-level fusion operator is able to statistically reject extreme data points by considering both the data reliability and difference.
- By applying the second-level fusion operator, the horizontal disturbance can be improved.

TABLE 1 CV based Comparison and Validation Results of Weekdays

Time of Day	Mean Absolute Percentage Difference (MAPD)							Amount of Verification Intervals							Acceptance Ratio (t-test)					
	obs=1 or 2	CV Category						obs=1 or 2	CV Category						CV Category					
		0-0.1	0.1-0.2	0.2-0.3	0.3-0.4	0.4-0.5	0.5+		0-0.1	0.1-0.2	0.2-0.3	0.3-0.4	0.4-0.5	0.5+	0-0.1	0.1-0.2	0.2-0.3	0.3-0.4	0.4-0.5	0.5+
0-1 AM	17.7%	10.9%	8.9%	N/A	N/A	N/A	N/A	566	6	10	0	0	0	0	100%	100%	N/A	N/A	N/A	N/A
1-2 AM	20.0%	0.1%	N/A	29.1%	N/A	N/A	N/A	257	1	0	1	0	0	0	100%	N/A	100%	N/A	N/A	N/A
2-3 AM	21.9%	N/A	1.5%	N/A	N/A	N/A	N/A	177	0	1	0	0	0	0	N/A	100%	N/A	N/A	N/A	N/A
3-4 AM	22.3%	N/A	N/A	10.4%	N/A	N/A	N/A	251	0	0	3	0	0	0	N/A	N/A	100%	N/A	N/A	N/A
4-5 AM	20.7%	N/A	N/A	N/A	N/A	N/A	N/A	272	0	0	0	0	0	0	N/A	N/A	N/A	N/A	N/A	N/A
5-6 AM	17.7%	21.8%	3.8%	10.6%	N/A	N/A	45.4%	355	1	5	4	0	0	1	0%	100%	100%	N/A	N/A	100%
6-7 AM	14.0%	9.1%	8.5%	13.0%	N/A	37.3%	N/A	802	8	5	3	0	1	0	75%	100%	100%	N/A	100%	N/A
7-8 AM	14.0%	10.7%	12.9%	16.3%	30.7%	48.4%	34.2%	1153	56	31	5	2	2	3	73%	100%	100%	100%	100%	100%
8-9 AM	15.1%	9.0%	13.0%	19.7%	22.0%	39.1%	43.9%	1380	115	95	29	6	6	9	82%	100%	100%	100%	100%	100%
9-10 AM	17.2%	13.4%	13.3%	18.1%	24.0%	26.7%	33.8%	1418	135	113	48	19	8	13	65%	100%	100%	100%	100%	100%
10-11 AM	17.9%	15.0%	14.4%	21.5%	32.3%	35.5%	41.7%	1451	196	150	57	26	28	25	53%	100%	98%	100%	100%	100%
11-12 AM	17.6%	14.7%	14.5%	18.8%	31.4%	35.6%	45.8%	1448	336	437	173	50	35	29	39	54%	99%	100%	100%	100%
0-1 PM	18.1%	13.8%	16.5%	25.3%	31.4%	41.4%	42.9%	1409	400	196	64	41	20	28	56%	98%	100%	100%	100%	100%
1-2 PM	18.0%	14.2%	17.3%	26.8%	32.4%	38.1%	43.0%	1308	466	272	77	51	31	25	60%	96%	100%	100%	100%	100%
2-3 PM	17.9%	15.2%	18.3%	24.7%	30.3%	39.8%	43.4%	1274	544	263	69	56	36	44	54%	97%	100%	100%	100%	100%
3-4 PM	18.7%	15.4%	13.2%	17.3%	26.5%	29.9%	40.0%	1151	336	437	160	77	64	82	57%	99%	100%	100%	100%	100%
4-5 PM	21.7%	20.9%	16.3%	22.8%	29.7%	35.4%	41.0%	969	518	346	158	79	50	57	38%	95%	99%	100%	100%	100%
5-6 PM	33.3%	34.2%	31.7%	39.6%	38.2%	45.0%	45.2%	950	755	429	159	69	20	27	11%	66%	97%	100%	100%	100%
6-7 PM	31.2%	34.7%	29.8%	33.2%	41.2%	44.4%	46.3%	1035	541	425	150	78	46	43	12%	65%	99%	100%	100%	100%
7-8 PM	27.4%	26.4%	26.5%	33.4%	39.1%	47.3%	46.6%	1371	311	215	82	48	27	29	17%	88%	100%	100%	100%	100%
8-9 PM	24.6%	23.0%	21.3%	26.6%	37.9%	44.2%	47.3%	1336	194	148	83	34	22	17	23%	95%	100%	100%	100%	100%
9-10 PM	22.6%	23.8%	17.5%	19.2%	34.1%	40.2%	38.7%	1342	113	76	69	24	10	13	26%	99%	100%	100%	100%	100%
10-11 PM	18.3%	15.1%	13.4%	16.7%	29.2%	25.1%	30.7%	1204	58	60	29	16	8	4	47%	98%	100%	100%	100%	100%
11-12 PM	16.5%	10.5%	12.7%	15.5%	24.7%	32.5%	12.1%	940	14	25	12	3	3	1	57%	100%	100%	100%	100%	100%

TABLE 2 CV based Comparison and Validation Results of Weekends

Time of Day	Mean Absolute Percentage Difference (MAPD)							Amount of Verification Intervals							Acceptance Ratio (t-test)					
	obs=1 or 2	CV Category						obs=1 or 2	CV Category						CV Category					
		0-0.1	0.1-0.2	0.2-0.3	0.3-0.4	0.4-0.5	0.5+		0-0.1	0.1-0.2	0.2-0.3	0.3-0.4	0.4-0.5	0.5+	0-0.1	0.1-0.2	0.2-0.3	0.3-0.4	0.4-0.5	0.5+
0-1 AM	15.3%	19.9%	7.8%	0.3%	N/A	N/A	N/A	360	11	12	1	0	0	0	36%	100%	100%	N/A	N/A	N/A
1-2 AM	16.8%	15.9%	9.6%	30.5%	N/A	N/A	N/A	191	2	1	1	0	0	0	50%	100%	100%	N/A	N/A	N/A
2-3 AM	21.3%	N/A	N/A	N/A	N/A	N/A	N/A	148	0	0	0	0	0	0	N/A	N/A	N/A	N/A	N/A	N/A
3-4 AM	19.4%	10.2%	N/A	N/A	N/A	N/A	N/A	109	1	0	0	0	0	0	0%	N/A	N/A	N/A	N/A	N/A
4-5 AM	20.1%	N/A	N/A	N/A	N/A	N/A	N/A	96	0	0	0	0	0	0	N/A	N/A	N/A	N/A	N/A	N/A
5-6 AM	16.8%	N/A	N/A	N/A	N/A	N/A	N/A	98	0	0	0	0	0	0	N/A	N/A	N/A	N/A	N/A	N/A
6-7 AM	20.1%	N/A	N/A	N/A	N/A	N/A	N/A	121	0	0	0	0	0	0	N/A	N/A	N/A	N/A	N/A	N/A
7-8 AM	17.9%	12.6%	3.7%	24.0%	N/A	53.5%	N/A	292	1	3	2	0	1	0	100%	100%	100%	N/A	100%	N/A
8-9 AM	16.7%	12.9%	11.8%	22.1%	22.9%	N/A	51.7%	367	14	15	4	3	0	1	50%	100%	100%	100%	N/A	100%
9-10 AM	19.8%	17.4%	14.5%	23.7%	16.2%	54.8%	45.2%	484	14	23	9	1	1	7	50%	96%	100%	100%	100%	100%
10-11 AM	21.6%	22.0%	18.6%	24.1%	39.2%	41.2%	48.7%	542	66	43	20	12	4	6	32%	100%	100%	100%	100%	100%
11-12 AM	25.1%	24.4%	26.3%	33.6%	37.7%	46.0%	53.6%	577	98	49	18	10	5	6	15%	90%	100%	100%	100%	100%
0-1 PM	26.2%	25.7%	27.2%	29.3%	42.1%	46.3%	50.5%	607	118	91	27	16	5	10	20%	85%	96%	100%	100%	100%
1-2 PM	28.7%	25.5%	29.4%	37.7%	40.4%	51.8%	49.5%	590	150	75	26	9	7	15	17%	85%	96%	100%	100%	100%
2-3 PM	26.8%	25.2%	28.4%	33.9%	39.8%	44.4%	47.4%	542	181	92	28	21	7	12	16%	88%	100%	100%	100%	100%
3-4 PM	28.2%	25.2%	27.6%	35.8%	44.5%	48.0%	53.5%	561	176	92	27	22	18	22	15%	84%	100%	100%	100%	100%
4-5 PM	29.7%	27.2%	27.8%	31.1%	38.1%	52.2%	54.7%	559	148	110	36	20	12	23	15%	82%	100%	100%	100%	100%
5-6 PM	28.7%	26.5%	28.1%	35.6%	41.1%	46.9%	53.2%	569	141	81	24	14	11	20	13%	80%	100%	100%	100%	100%
6-7 PM	28.2%	26.6%	27.6%	28.6%	40.1%	41.3%	49.0%	573	115	66	39	16	8	11	15%	85%	100%	100%	100%	100%
7-8 PM	26.3%	25.4%	25.2%	23.6%	39.9%	57.3%	49.9%	567	84	49	25	11	3	9	17%	94%	100%	100%	100%	100%
8-9 PM	23.6%	25.7%	19.1%	19.5%	34.2%	33.4%	39.5%	562	42	48	27	11	6	3	12%	100%	100%	100%	100%	100%
9-10 PM	21.6%	24.0%	21.0%	20.3%	28.4%	38.2%	25.2%	555	27	35	10	3	1	1	19%	97%	100%	100%	100%	100%
10-11 PM	17.2%	7.6%	11.1%	12.1%	34.0%	44.0%	37.1%	458	7	21	6	4	2	1	86%	95%	100%	100%	100%	100%
11-12 PM	16.1%	15.2%	7.9%	20.3%	0.0%	23.0%	45.0%	375	7	6	4	0	2	1	71%	100%	100%	N/A	100%	100%

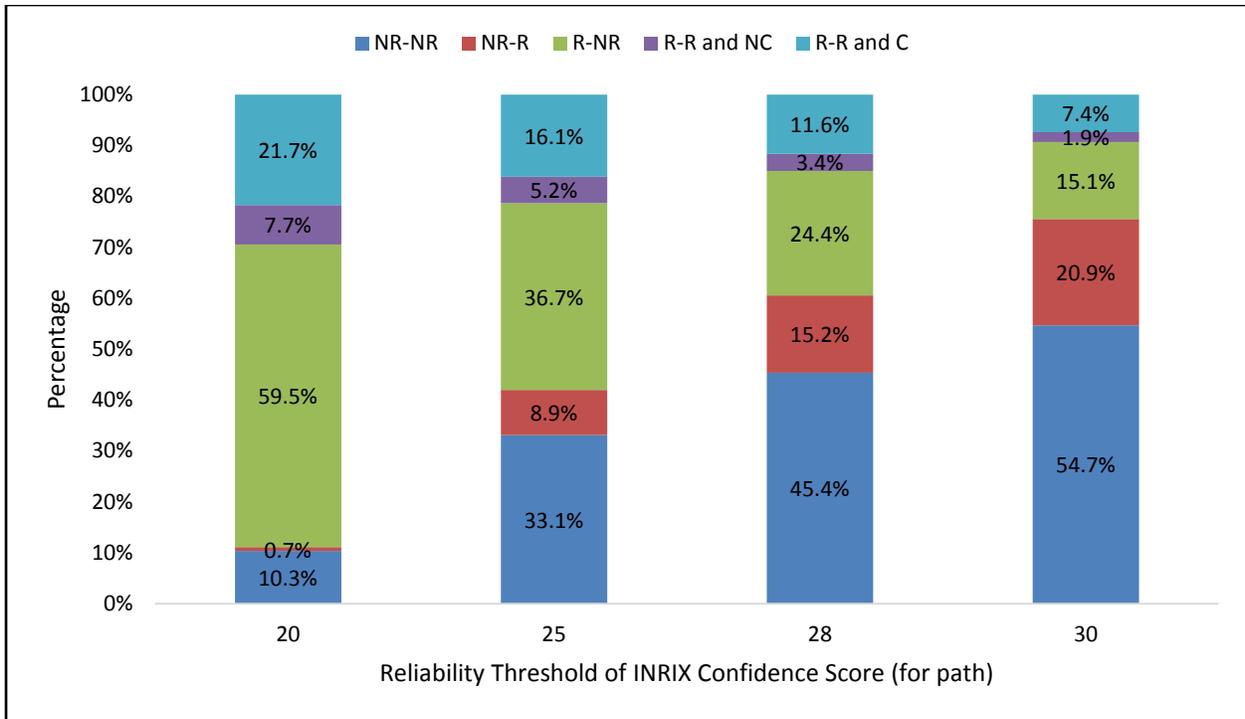


FIGURE 4 Percentage of Each Fusion Context with Different INRIX Data Reliability Threshold *a*.

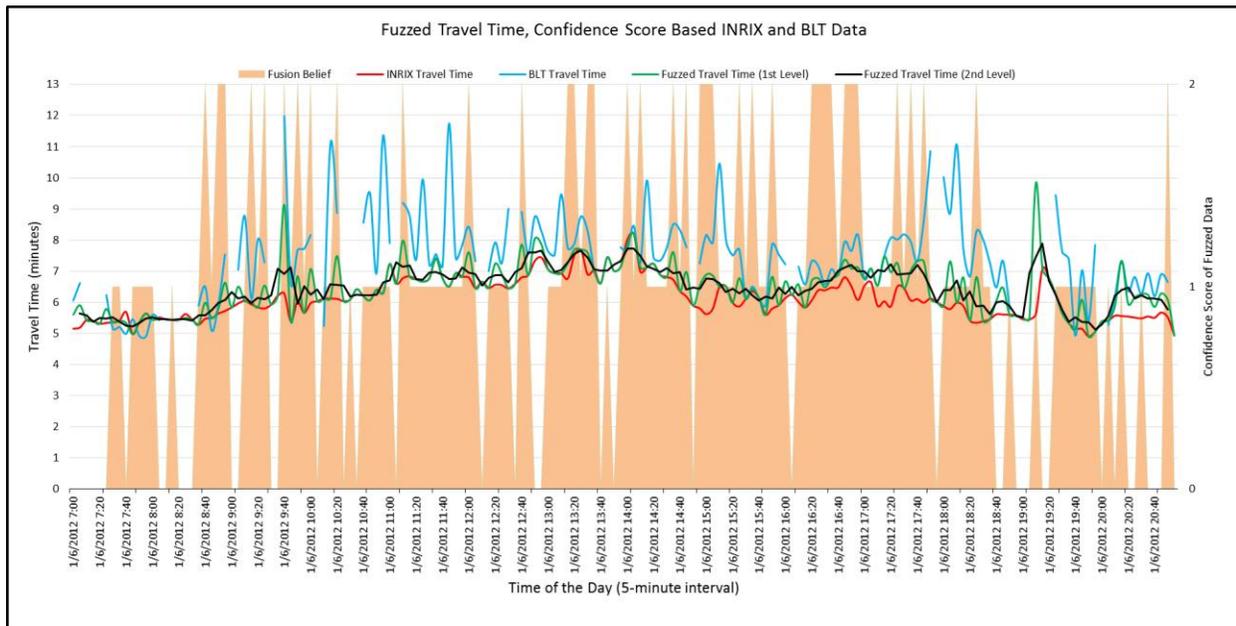


FIGURE 5 Part Comparison of Fused Travel Time and Fusion Belief against INRIX and Bluetooth Travel Time (Daytime of 01/06/2012).

SUMMARY AND CONCLUSIONS

This paper presented a new validation scheme for comparing travel time data from two independent data sources with an emphasis on arterial applications. By using the validation methods based on CV categories, the independent time series data can be comprehensively compared. In addition, a Context Dependent (CD) based travel time fusion framework is

developed to blend data from INRIX and BT datasets in order to improve the data quality. The fusion model takes advantage of a fusion belief system corresponding to each fused data point to declare the reliability of the fused data. The proposed model can be flexibly applied to scenarios with other independent data sources. The fused data, with a higher data quality can be used in various applications such as travel time prediction and travel time reliability evaluation. Both validation and fusion methodologies were applied on a case study and the results are reported.

ACKNOWLEDGEMENTS

This work was supported by Maryland State Highway Administration SPR Research Project SP309B4F. The Bluetooth Sensors used in this study were designed and manufactured in the Center for Advanced Transportation Technology of the University of Maryland, College Park. INRIX travel time data was retrieved from the Regional Integrated Transportation Information System (RITIS).

REFERENCES

1. Shawn M. Turner, William L. Eisele, Robert J. Benz, and Douglas J. Holdener, "Travel Time Data Collection Handbook," Federal Highway Administration, Report FHWA-PL-98-035, March 1998.
2. Tam, M. L., & Lam, W. H. (2008). Using Automatic Vehicle Identification Data for Travel Time Estimation in Hong Kong. *Transportmetrica*, 4(3), 179-194.
3. Liu, H. X., & Ma, W. (2009). A virtual vehicle probe model for time-dependent travel time estimation on signalized arterials. *Transportation Research Part C: Emerging Technologies*, 17(1), 11-26.
4. Kwong, K., Kavalier, R., Rajagopal, R., & Varaiya, P. (2009). Arterial travel time estimation based on vehicle re-identification using wireless magnetic sensors. *Transportation Research Part C: Emerging Technologies*, 17(6), 586-606.
5. Schrank, D., Lomax, T., & Turner, S. (2010). TTI's 2010 urban mobility report powered by INRIX traffic data. Texas Transportation Institute, Texas A&M University System.
6. Haghani, A., Hamed, M., Sadabadi, K. F., Young, S., & Tarnoff, P. (2010). Data collection of freeway travel time ground truth with bluetooth sensors. *Transportation Research Record: Journal of the Transportation Research Board*, 2160(1), 60-68.
7. Hamed, M., Haghani, A., & Sadabadi, F. (2009). Using Bluetooth Technology for Validating Vehicle Probe Data. In 16th ITS World Congress and Exhibition on Intelligent Transport Systems and Services.
8. Haghani, A., Hamed, M., & Sadabadi, K. F. (2009). I-95 Corridor Coalition Vehicle Probe Project: Validation of INRIX Data, September 2009.
9. Anusha, S. P., Anand, R. A., & Vanajakshi, L. (2012). Data Fusion Based Hybrid Approach for the Estimation of Urban Arterial Travel Time. *Journal of Applied Mathematics*, 2012.
10. Hofleitner, A., Herring, R., Abbeel, P., & Bayen, A. (2012). Learning the dynamics of arterial traffic from probe data using a dynamic Bayesian network. *Intelligent Transportation Systems, IEEE Transactions on*, 13(4), 1679-1693.
11. Herring, R., Hofleitner, A., Abbeel, P., & Bayen, A. (2010, September). Estimating arterial traffic conditions using sparse probe data. In *Intelligent Transportation Systems (ITSC), 2010 13th International IEEE Conference on* (pp. 929-936). IEEE.
12. Aliari, Y., & Haghani, A. (2012). Bluetooth sensor data and ground truth testing of reported travel times. *Transportation Research Record: Journal of the Transportation Research Board*, 2308(1), 167-172.
13. Faouzi, N. E. E., Leung, H., & Kurian, A. (2011). Data fusion in intelligent transportation systems: Progress and challenges—A survey. *Information Fusion*, 12(1), 4-10.
14. Kong, Q. J., Li, Z., Chen, Y., & Liu, Y. (2009). An approach to urban traffic state estimation by fusing multisource information. *Intelligent Transportation Systems, IEEE Transactions on*, 10(3), 499-511.
15. Soriguera, F., & Robusté, F. (2011). Highway travel time accurate measurement and short-term prediction using multiple data sources. *Transportmetrica*, 7(1), 85-109.(10)
16. Bloch, I. (1996). Information combination operators for data fusion: a comparative review with classification. *Systems, Man and Cybernetics, Part A: Systems and Humans, IEEE Transactions on*, 26(1), 52-67.

APPENDIX B

Real-time travel time prediction by particle filtering with a non-explicit state-transition model

This article may be cited as: Chen, H. and Rakha, H.A., 2014. Real-time travel time prediction using particle filtering with a non-explicit state-transition model. *Transportation Research Part C: Emerging Technologies*, 43, pp.112-126.

INTRODUCTION

Traffic prediction is an essential part of Advanced Traffic Management Systems (ATMSs) and Advanced Traveler Information Systems (ATISs). The Federal Highway Administration (FHWA) encourages all Traffic Management Centers (TMCs) to post travel times and incident information, which not only provide useful information to motorists but also assists them in making route choice decisions. Such information can assist drivers in making decisions to detour from congested highways, thus providing critical additional capacity and assisting in the management of congestion [1].

Many studies have been conducted to estimate or predict traffic states (e.g. flow, speed and density) [2-8] and travel times [9-14]. Among the existing methods, the use of macroscopic traffic models within a Bayesian filtering framework has gained popularity in recent years to address real-time estimation and prediction problems [3, 4, 6, 8, 11, 15].

There are two main advantages of using the combination of macroscopic traffic models with recursive Bayesian filters. First, within the time update process, the relationship between traffic state parameters over adjacent time intervals, also known as the state-transition model, is accurately characterized using macroscopic traffic models. Compared to previous studies that construct a relationship between predicted and previous traffic states (state-transition model) using data-driven methods [16, 17] or simply based on neighboring variables [12, 13], macroscopic traffic models provide an analytical solution to model the state-transition function that is explainable through physical principles of traffic stream movement and thus is not affected by noises in data-driven methods. More importantly, macroscopic traffic models can respond to non-recurrent or sudden changes in traffic conditions. For example, the model parameters including roadway capacity and free-flow-speed can be adjusted in response to a traffic incident or inclement weather conditions. The other advantage lies in the fact that both the measurement update (estimation) and time update (prediction) processes are included in the Bayesian filtering framework each time interval. The sequence of the two processes within a single time interval categorizes the problem as either data estimation or prediction. Once a new measurement is available, it is used to adjust the prior predicted value and obtain the estimation. Conversely, prediction is calculated by implementing the estimated value in the time update equation (state-transition model) [15]. Consequently, the Bayesian filtering framework can deal with both estimation and prediction problems.

For real-time travel time prediction problems, the state-transition models in previous studies are usually simply defined by the time series trends from near-past or historical travel times. In this way, the nonlinear transition function between adjacent travel times is divided into discrete linear functions. Consequently, Kalman filter based travel time prediction algorithms are proposed [12-14, 18]. Although those methods demonstrate better performance compared to other naive methods using real-time or historical data, the assumption of Gaussian noise in the Kalman filter may not always be consistent with field data [5]. Moreover, it should be noted that such Kalman filter methods are essentially based on the simplified linear state-transition function. However, the travel times of neighboring time intervals have a strong nonlinear relationship considering the nonlinear traffic stream behavior. It should be mentioned that Extended Kalman Filter (EKF) methods can deal with nonlinear problems using Taylor estimation and thus are widely used for traffic state estimation [8, 19]. EKF is a revised classical Kalman filter with the calculation of the Jacobian expression. However, sometimes it is very difficult to compute the Jacobian expression for nonlinear state-transition models. To overcome this problem, the Ensemble Kalman filter (EnKF) is proposed which enables the use of a fully nonlinear evolution equation while exploiting

the linear observation equation [6]. However, it cannot deal with the problem of nonlinear measurement.

Compared with previous mentioned filtering methods, the particle filter is a sequential Monte Carlo method with the advantages of addressing strong nonlinear dynamic problems and without the assumption of noise distributions [20]. Numerous studies have demonstrated the advantages of the use of particle filters over Kalman filters for various applications with nonlinear state-transition models [20, 21]. In the field of transportation, an Unscented Particle filter (UPF) was tested and demonstrated to outperform Kalman filter methods for traffic state estimation [5], and a particle filter method was demonstrated to produce half the prediction error for traffic speed estimation when compared to an EnKF algorithm [4]. Considering the aforementioned nonlinear traffic behavior, particle filters provide a better Bayesian filtering solution for travel time prediction.

However, another key problem still exists with the use of particle filter techniques to accurately model the travel time state transition function. Recently, several revised particle filter approaches have been developed to predict state values in other domains without specified state-transition models. For example, a variant of particle filtering algorithms is proposed to track the eye location of tropical cyclones using historical data. An explicit state update is not required in the approach since the prior distribution is predicted using historical trends [22]. In addition, a memory-based particle filter is proposed for tracking abrupt face changes under occlusions. This method can handle nonlinear, time-variant and non-Markov dynamics, which employs a random sampling from the history to generate prior distributions [23, 24]. Moreover, a chaotic modeling of nonlinear dynamical systems is proposed for prediction problems without the underlying dynamic models. Given an initial condition, the predictions of state variables are accomplished using kernel regression [25]. Consequently, the proposed approach uses these concepts for predicting experienced travel times.

Unlike previous travel time prediction studies, in this paper we develop multi-step predictions that estimate future travel time departures up to one hour later. Previous research has demonstrated that prediction accuracy typically deteriorates quickly using Artificial Neural Networks (ANNs) for multi-step predictions [26, 27]. Recently, a novel travel time prediction method was developed by considering temporal-spatial input dynamics in recurrent Neural Networks [28]. The test results using five different ANNs under various scenarios with or without incident data demonstrated the errors for 5-step-ahead prediction were nearly twice the errors for 1-step-ahead prediction. Moreover, off-line data training is needed in the ANN method, which makes it difficult to use ANNs to predict non-recurrent conditions or transfer to other locations. The proposed method can efficiently address these issues and produce higher accuracy for multi-step prediction.

In conclusion, the research presented here develops a particle filter approach for travel time prediction using real-time and historical data. Unlike previous studies that require an underlying physical model in modeling the state-transition function between predicted and previous travel times, the proposed particle filter uses historical trends to model the state-transition trend. The identified invalid or low weighted particles are removed and replaced in order to overcome the degeneracy problem in particle filters. Using the particle weights, the travel time reliability can be predicted by aggregating all the particles for each time interval. Probe data from Richmond to Virginia Beach are used to test and evaluate the performance of different prediction methods. The results indicate that the proposed method produces the least prediction error compared to the instantaneous method, two Kalman filters and a k -NN method for multi-step-ahead travel time

prediction. A sensitivity analysis is also conducted to explore the impacts of different model parameters on the prediction accuracy.

The remainder of this paper is organized as follows. Background information about Bayesian filtering and particle filtering techniques are provided in the next section. Subsequently, the framework of the proposed particle filter approach that does not use an explicit state-transition model is presented. This is followed by a description of the test data for the case study and a comparison of results using different prediction methods. The last section provides the conclusions of the paper and future research recommendations.

BACKGROUND

The new approach is developed from the concept of the Bayesian filter under the situation that only historical data are available instead of using an explicit state-transition model. Consequently, the theoretical background of the Bayesian filter and the general representation of a particle filter are initially introduced in this section.

When considering the problem of state tracking, the propagation of the state sequence using a state-transition model and the system update using measurement data are given by

$$x_t = f_t(x_{t-1}, \varphi_{t-1}) \quad (1)$$

$$z_t = h_t(x_t, \gamma_t) \quad (2)$$

where x_t and z_t represent the state variable and the data measurement at time interval t , respectively; φ_t and γ_t are time update and measurement update noises. Bayesian filters represent a general probabilistic approach to estimate the posterior probability density function (pdf) of a target state variable x_t at each discrete time interval t , using given past measurement data $z_{1:t} = \{z_1, z_2, \dots, z_t\}$. Specifically, the conditional density $p(x_t^{n+1}/y^{1:n})$ is recursively updated according to Eq. (3) and Eq. (4) as shown below [29]

$$p(x_t | z_{t-1}) = \int p(x_t | x_{t-1}) p(x_{t-1} | z_{t-1}) dx_{t-1} \quad (3)$$

$$p(x_t | z_{1:t}) = \frac{p(z_t | x_t) p(x_t | z_{1:t-1})}{p(z_t | z_{1:t-1})} \quad (4)$$

Where $p(x_t|x_{t-1})$ is the probability of system evolution, given by the time update process of Eq. (1); $p(z_t|x_t)$ is a likelihood function defined by the measurement update process of Eq. (2). However, the analytical solution of $p(x_t|z_{1:t})$ is difficult to calculate directly. In a particle filter approach, the posterior pdf of $p(x_t|z_{1:t})$ is represented by a set of random samples with corresponding weights. When the number of samples is large enough to approach infinity, these particles approximate the equivalent representation of the posterior pdf [20]. Suppose $x_t = \{x_t^{(i)}, w_t^{(i)}\}_{i=1}^N$ denotes a collection of particles, in which $x_t^{(i)}$ is the state value and $w_t^{(i)}$ is the corresponding weight of the i^{th} particle at time t . The posterior pdf can be approximated using Eq. (5), and the weights are updated using Eq. (6).

$$p(x_t | z_{1:t}) \approx \hat{\mathbf{a}} \sum_{i=1}^N w_t^{(i)} \delta(x_t - x_t^{(i)}) \quad (5)$$

$$w_t^{(i)} \propto w_{t-1}^{(i)} \times \frac{p(z_t | x_t^{(i)}) p(x_t^{(i)} | x_{t-1}^{(i)})}{q(x_t^{(i)} | x_{t-1}^{(i)}, z_t)} \quad (6)$$

Where $q(x_t^{(i)} | x_{t-1}^{(i)}, z_t)$ is the importance density, which is a known pdf chosen to generate the particles. If the importance density is chosen to be the same as the prior pdf $p(x_t^{(i)} | x_{t-1}^{(i)})$, then the weight update in Eq. (6) is simplified to Eq. (7) [20, 30]. Following this approximation, the

original integral calculation of Eq. (3) is transformed to an easier formulation of calculating the summation of particles with corresponding weights.

$$w_t^{(i)} \propto w_{t-1}^{(i)} \times p(z_t | x_t^{(i)}) = w_{t-1}^{(i)} \times p_e(z_t - h_t(x_t^{(i)})) \quad (7)$$

Table 3: Sampling importance resampling particle filter.

$\left[\left\{ x_t^{(i)} \right\}_{i=1}^N \right] = \text{SIR} \left[\left\{ x_{t-1}^{(i)} \right\}_{i=1}^N, z_t \right]$ <p><i>–Initialize particles</i></p> <p style="padding-left: 40px;"><i>Draw</i> $x_0^i \sim p(x_0), i \in [1:N]$</p> <p><i>Step 1: Time update</i></p> <p style="padding-left: 40px;"><i>Draw</i> $x_t^i \sim p(x_t x_{t-1}^i), i \in [1:N]$</p> <p><i>Step 2: Measurement update</i></p> <p style="padding-left: 40px;">$w_t^{(i)} \propto p(z_t x_t^{(i)}) = p_e(z_t - x_t^{(i)}), i \in [1:N]$</p> <p><i>Step 3: Resampling</i></p> <p style="padding-left: 40px;"><i>Initialize the cumulative density function</i> : $c^{(1)} = w_t^{(1)}$</p> <p style="padding-left: 40px;"><i>For</i> $i = 2:N$</p> <p style="padding-left: 80px;"><i>Construct the cumulative density function</i> : $c^{(i)} = c^{(i-1)} + w_t^{(i)}$</p> <p style="padding-left: 40px;"><i>End For</i></p> <p style="padding-left: 40px;"><i>Let</i> $i = 1$, <i>draw a starting point</i> : $u_1 \sim U(0, N^{-1})$</p> <p style="padding-left: 40px;"><i>For</i> $j = 1:N$</p> <p style="padding-left: 80px;">$u^{(j)} = u^{(1)} + N^{-1}(j-1)$</p> <p style="padding-left: 80px;">$i = i+1$, <i>when</i> $u^{(j)} > c^{(i)}$</p> <p style="padding-left: 80px;"><i>Assign sample</i> : $x_t^{(j)} = x_t^{(i)}$</p> <p style="padding-left: 40px;"><i>End For</i></p>

After the arrival of new measurement data, weights are updated considering the importance of corresponding particles. According to the calculation of likelihood $p(z_t | x_t^{(i)})$, the smaller error between a prediction and a measurement data results in the larger weight is assigned to the corresponding particle. In this way, the particle filter comprises recursive propagation of the weights and updates the state variable when new measurements are obtained. However, a common problem exists during weight updating in particle filter approaches, namely: the degeneracy problem. In this problem, the variance of weights can only increase over time, which results in all but one particle having negligible weights after several iterations. Although it is impossible to avoid the degeneracy problem, previous studies introduced resampling as an efficient alternative to reduce the effects of degeneracy [21]. The basic idea of resampling is to eliminate particles with small weights and to concentrate on particles with large weights. The sampling importance resampling (SIR) particle filter is described in Table 3, which is derived from the sequential importance sampling algorithm by choosing the importance density to be the transitional prior and by performing the resampling step at every time interval. This approach has the advantage that the importance density can be easily updated and the important weights are easily evaluated [20]. The concepts of the Bayesian filter and SIR particle filter are the bases to construct the proposed algorithm under the condition that the state-transition model $p(x_t | x_{t-1})$ is

not explicitly given.

METHODOLOGY

The implementation of the traditional Bayesian filter approach is typically challenging in travel time prediction, since the state-transition model, which characterizes the relationship between predicted and previous travel times is difficult to quantify. Consequently, a data-driven method based on historical data sampling is proposed in this paper to address this problem. The definition of the problem is presented first in this section, followed by a description of the proposed solution and the related problems.

Definitions and Denotations

The methodology in this paper attempts to predict experienced travel times using real-time and historical data. The experienced travel time is the actual, realized travel time that a vehicle could experience during a trip. Comparatively, the instantaneous travel time is the summation of section travel times at the same time interval. The instantaneous travel time is close to the experienced travel time when the roadway speed does not change significantly across time space during the entire trip, e.g. free-flow conditions. Nevertheless, instantaneous travel times may deviate substantially from the experienced travel time under transient states during which congestion is forming or dissipating during the trip [31].

Various traffic sensing technologies have been used to collect traffic data for use in computing travel times, including point to point travel time collection (license plate recognition systems, automatic vehicle identification systems, mobile, Bluetooth, probe vehicle, etc.) and station based traffic state measuring devices (loop detector, video camera, remote traffic microwave sensor, etc.). Private companies such as INRIX integrate different sources of measured data to provide section-based average speed or travel time, which can be used to construct traffic speed matrix over spatial and temporal and thus is used in this paper. The benefit of using temporal-spatial speed data is that travel time can be easily calculated afterward [32]. More importantly, such data provides the flexibility for scalable applications on traffic networks.

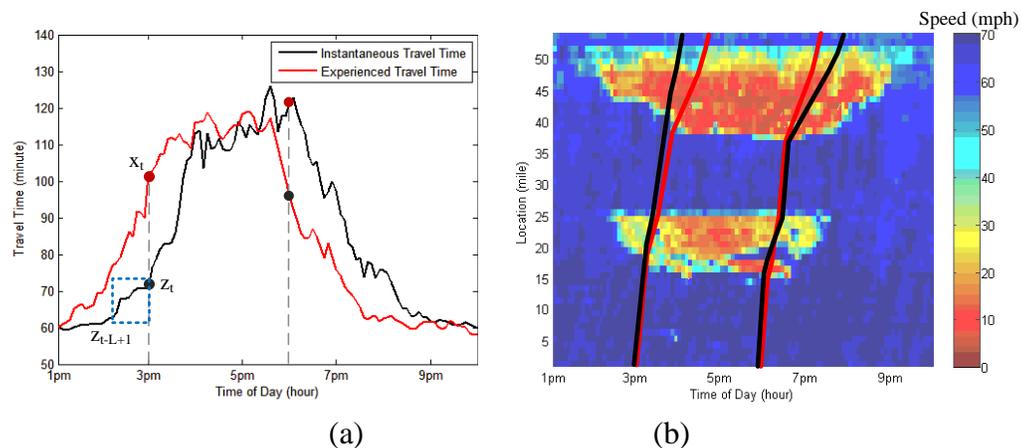


Figure 6: Representation of field data for May 19 2012 on I-66; (a) instantaneous and experienced travel times; (b) traffic speed contour.

In order to demonstrate the research problem, a contour plot of spatiotemporal traffic speed and the corresponding instantaneous and experienced travel times are illustrated on Figure 6. The speed contour is the five-minute aggregated INRIX probe data along I-66 between Richmond to Hampton Roads on May 19, 2012 from 1 PM to 10 PM, and traffic speed is represented by

different colors from red (congested) to blue (uncongested) in Figure 6(b). Based on the spatiotemporal speed data, trip trajectories can be plotted to calculate instantaneous and experienced travel times, which are denoted by black and red curves, respectively. For trip departure time of 3 PM, the instantaneous travel time (72 minutes) underestimates the experienced travel time (102 minutes) by 30 minutes. Conversely, the instantaneous travel time (123 minutes) on 6 PM overestimates the corresponding experienced travel time (96 minutes) by 27 minutes. The above examples of two trips demonstrate the discrepancy between instantaneous and experienced travel times. In order to develop a travel time prediction approach for real-time applications, the experienced travel time is required, since the instantaneous travel time is not a good indicator of the actual travel time, especially at the shoulders of the peak period. The experienced travel time cannot be measured until a traveler completes their trip and thus the experienced travel time departure for the previous time step (e.g. five minutes earlier) may not be available at the current time, especially for long trips.

Considering the above analysis, the state-transition and measurement update formulations defined in Eq. (1) and (2) are used in the proposed particle filter. It should be noted that both the state and measurement update equations are nonlinear functions in our application. The experienced travel time at time t is defined as the state variable x_t , and the state-transition function f_t or $p(x_t/x_{t-1})$ represents the nonlinear relationship from x_{t-1} to x_t . The instantaneous travel time sequence from short-past $t-L+1$ to current time t is defined as the measurement variable z_t , which also represents the traffic pattern at current time t and is highlighted by the blue rectangular box in Figure 6(a). Here, L denotes the length of the data sequence.

Since it is difficult to find an analytical solution for the state-transition model, historical data are used here to provide the pool of past information including traffic trends of instantaneous and experienced travel time sequences, which can be used to replace the state transition model. In the proposed method, assume historical data is denoted by Ω , the state variable x_t is approximated by a set of particles $\{x_t^{(i)}\}_{i=1}^N$ and each particle $x_t^{(i)}$ corresponds to the experienced travel time $\Omega_{exp}(d_t^{(i)}, j_t^{(i)})$ at time $j_t^{(i)}$ on the historical day $d_t^{(i)}$. Moreover, each particle $x_t^{(i)}$ is also associated with traffic pattern $y_t^{(i)}$ represented by the tail value $\Omega_{inst}(d_t^{(i)}, j_t^{(i)})$, which is used to match with the real-time traffic pattern and calculate the particle weight. Specifically, the real-time traffic pattern is used to update particles and calculate the weight of each particle based on the dissimilarity of two traffic patterns. The reason for using travel time sequences instead of a single values to represent traffic patterns is that more dynamic information are included in the data sequence and potentially can improve the accuracy of matching traffic patterns between real-time measurements and historical data. Here it should be noted that the nonlinear function h_t in the measurement update equation, which captures the relationship between experienced travel time and traffic pattern (instantaneous travel time sequence), can also be used to describe the correlation between $x_t^{(i)}$ and $y_t^{(i)}$.

(a) *Proposed Particle Filter Approach with non-Explicit State-transition Model*

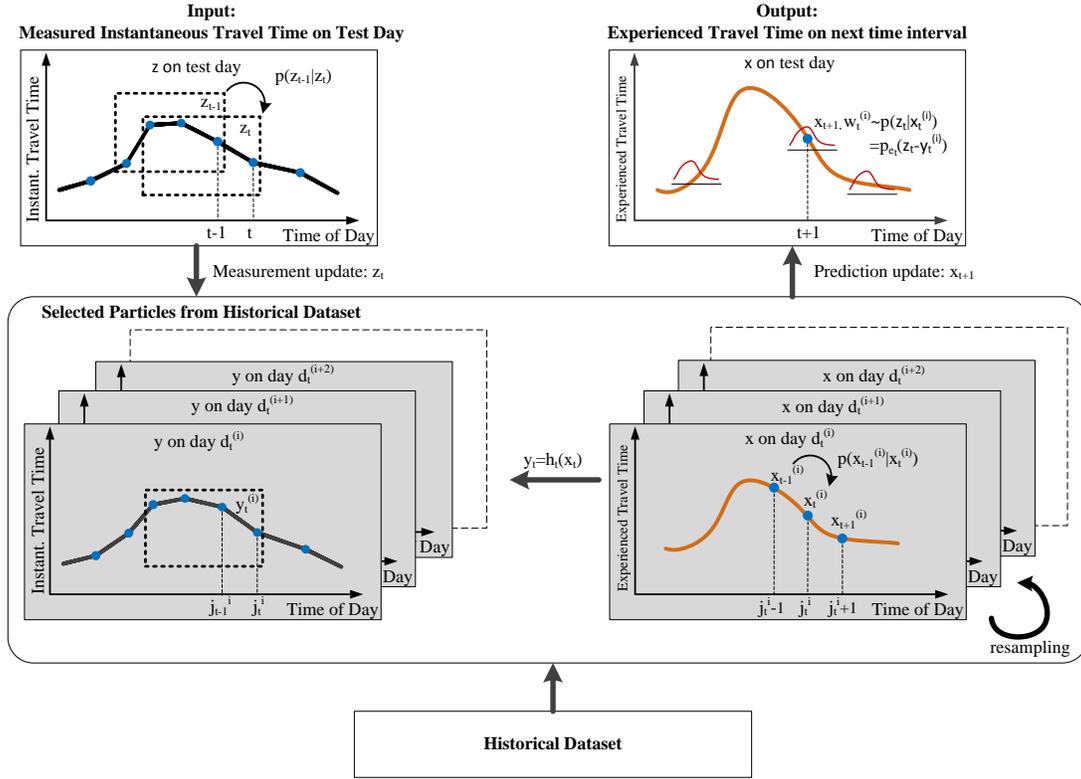


Figure 7: Demonstration of the proposed particle filter approach.

A graphical representation of the proposed approach of non-explicit state-transition particle filter (NSPF) is demonstrated in Figure 7. The input data are the measured instantaneous travel times for each time interval, the update of measurement data from z_{t-1} to z_t is conducted by shifting the data sequence window one time step forward. Each particle can be recognized as a data sequence of instantaneous travel times and a data sequence of experienced travel times on the same historical day. The time update of the particle filter from $x_{t-1}^{(i)}$ to $x_t^{(i)}$ is accomplished by shifting one step ahead along the data sequence of experienced travel time. For each particle, the corresponding traffic pattern $y_t^{(i)}$ can be derived according to the relationship with $x_t^{(i)}$ represented by $y_t = h_t(x_t)$. At the same time, the associated weight $w_t^{(i)}$ can be calculated as the likelihood $p(z_t|x_t^{(i)})$, which can be accomplished by comparing the dissimilarity between real-time and historical traffic pattern as $p_{e_t}(z_t - y_t^{(i)})$. In this study, the likelihood function p_{e_t} is chosen as a normal distribution $N(0,1)$ [30]. Consequently, the distribution of experienced travel time on the next time interval $t+1$ can be predicted as $\{x_{t+1}^{(i)}, w_t^{(i)}\}_{i=1}^N$. For multi-step prediction with prediction horizon $t+p$, the propagation along experienced travel time sequence on historical day can be iteratively conducted but the same weight updated by the current measurement is maintained for each particle. So the experienced travel time on $t+p$ can be predicted as $\{x_{t+p}^{(i)}, w_t^{(i)}\}_{i=1}^N$.

Considering the application of multi-step travel time prediction, the similar structure as SIR particle filter method is revised to develop the proposed algorithm, which includes the steps of initialization, time update process, measurement update process, resampling and travel time prediction. The rest of this section describes the details of each step in the proposed particle filter method and the pseudo code of these steps is presented in Table 4.

Initialization

Since there is no explicit state-transition model in the proposed method, the initial travel time values and the corresponding traffic patterns should be assigned to all the particles. For each particle $x_0^{(i)}$, the initialization process is accomplished by randomly selecting the day index $d_0^{(i)}$ and the time index $j_0^{(i)}$ on that day from the historical data set. Therefore, the corresponding experienced travel time and traffic pattern are $\Omega_{exp}(d_0^{(i)}, j_0^{(i)})$ and $\Omega_{inst}(d_0^{(i)}, j_0^{(i)})$ respectively.

Time Update Process

Comparing to the SIR particle filter described in Table 3, each particle in the proposed algorithm propagates along a historical experienced travel time sequence as opposed to using a state-transition model. For a particle $x_{t-1}^{(i)}$ at time $t-1$, the time update $p(x_t|x_{t-1})$ is conducted by shifting the time index $j_{t-1}^{(i)}$ one step ahead and keeping the same day index. A follow up process attempts to identify valid particles that provide a sufficient time interval buffer considering the prediction horizon p . Consequently, the invalid particles are selected if the corresponding experienced travel time sequences cannot provide prediction output by shifting the time index by p . Here, the last time interval of the historical experienced travel time sequence on day $d_t^{(i)}$ is denoted by $Hd_t^{(i)}$. Unlike the time update process using a continuous state-transition model, the prediction in the proposed algorithm considers the boundary of data sequence for each historical day. Although the end of one day is connected with the beginning of the following day, the option of moving to the subsequent data sequence is replaced by the alternative of resampling. The reason to use this alternative lies in the consideration that the travel time sequence for the next day may not be available in the historical data set. In this way, the valid particles with respect to prediction horizon p are identified as collection Ψ_t for the t^{th} time interval.

Measurement Update Process

The measurement update process attempts to calculate the weights of all valid particles in Ψ_t . For each valid particle $x_t^{(i)}$, the weight is calculated by the likelihood function p_{e_t} with input of corresponding traffic pattern $y_t^{(i)}$ and real-time traffic pattern z_t . Thereafter, all the valid particles are sorted according to the associated weight values in descending order. The top N_{th} particles are maintained and the remaining particles are resampled in the next step. In this way, all the particles are divided into two groups. The first group includes the particles with large weights. The second group includes the invalid particles that cannot provide prediction values (exceed data sequence boundary) or the particles with negligible weights. The second group of particles will be re-selected in the next process so that new particles with similar traffic patterns to the current time interval can be selected.

Resampling

It should be noted that the problem of degeneracy in the SIR particle filter also exists in the proposed method. A resampling procedure is proposed to tackle this problem. The traditional threshold-based resampling strategies include residual, stratified and systematic resampling. These methods are used to eliminate samples with low importance weights and multiply samples with high importance weights [20]. The procedure proposed in this study is used for the same purpose. To save on computation time, a partial resampling method is developed instead of the threshold based resampling strategies [33], as will be discussed in further detail.

In the resampling process, the remaining $N-N_{th}$ particles in the second group of the measurement update process are resampled from the historical data set. During the resampling process only the

historical data that have traffic patterns similar to the real-time measurements are selected to increase the efficiency of particle propagation. Consequently, a process is developed to calculate the maximum similarity between the traffic patterns from each historical day and the current day/time interval. The maximum similarity of each historical day is transferred to a probability using a likelihood function. In this way, the day with closer similarity to the current traffic pattern has a larger probability to be selected in the resampling process.

Travel Time Prediction

In the proposed algorithm, the multi-step prediction is conducted by iteratively shifting the time step along the corresponding experienced travel time sequence of each particle. At the same time, each particle maintains the same weight until a new measurement is obtained. Consequently, the aggregated results from all the particles can provide the travel time distribution prediction instead of a single expected value. Moreover, the average prediction result can also be calculated as the weighted average travel time of each particle.

Table 4: Multi-step travel time prediction by proposed particle filter approach (NSPF).

$\left[\left\{ x_t^{(i)} \right\}_{i=1}^N \right] = NSPF \left[\left\{ x_{t-1}^{(i)} \right\}_{i=1}^N, z_t, \Omega \right]$ <p><i>-Initialize particles</i> $x_0 : \left\{ x_0^{(i)} \mid x_0^{(i)} = \Omega_{\text{exp}} \left(d_0^{(i)}, j_0^{(i)} \right), i \in [1: N] \right\}$</p> <p style="padding-left: 40px;">$d_0^{(i)} = \text{randomly select a day from } [1, 2, \dots, D], j_0^{(i)} = \text{randomly select a time index at day } d_0^{(i)}, i \in [1: N]$</p> <p>Step 1: Time update</p> <p style="padding-left: 40px;"><i>Propagate the particles by drawing</i> $x_t^{(i)} \sim p \left(x_t^{(i)} \mid x_{t-1}^{(i)} \right)$</p> <p style="padding-left: 40px;">$d_t^{(i)} = d_{t-1}^{(i)}, j_t^i = j_{t-1}^i + 1, i \in [1: N]$</p> <p style="padding-left: 40px;"><i>Identify valid particles with respect to prediction horizon p</i></p> <p style="padding-left: 40px;">$\Psi_t = \left\{ i \mid j_t^{(i)} \leq H_{d_t^{(i)}} - p, i \in [1: N] \right\}$</p> <p>Step 2: Measurement update</p> <p style="padding-left: 40px;">$w_t^{(i)} \propto p \left(z_t \mid x_t^{(i)} \right) = p_e \left(z_t - y_t^{(i)} \right), i \in \Psi_t$</p> <p style="padding-left: 40px;"><i>Select</i> N_{th} <i>number of particles with least weight values</i></p> <p style="padding-left: 40px;"><i>For</i> $j = 1: N_{th}$</p> <p style="padding-left: 80px;">$x_t^{(j)} = x_t^{(i)}, w_t^{(j)} = w_t^{(i)}, \text{ when } i = \arg \max_{i \in \Psi_t} w_t^{(i)}, \Psi_t = \Psi_t - \{i\}$</p> <p style="padding-left: 40px;"><i>End For</i></p> <p>Step 3: Resampling</p> <p style="padding-left: 40px;"><i>For</i> $j = N_{th} + 1: N$</p> <p style="padding-left: 80px;"><i>Calculate the probability of selecting each historical day</i> λ_t^n</p> <p style="padding-left: 80px;">$\lambda_t^n = p_e \left(z_t - \Omega_{inst} \left(n, \sigma_t^n \right) \right), \text{ when } \sigma_t^n = \arg \min_{k \in [L, H_n - p]} \left(z_t - \Omega_{inst} \left(n, k \right) \right), n \in [1: D]$</p> <p style="padding-left: 80px;">$d_t^{(j)} = \text{randomly select a day from } [1, 2, \dots, D] \text{ according to the probability } [\lambda_t^1, \lambda_t^2, \dots, \lambda_t^D]$</p> <p style="padding-left: 80px;">$x_t^{(j)} = \Omega_{\text{exp}} \left(d_t^{(j)}, \sigma_t^{d_t^{(j)}} \right), w_t^{(j)} = \lambda_t^{d_t^{(j)}}$</p> <p style="padding-left: 40px;"><i>End For</i></p> <p>Step 4: Prediction</p> <p style="padding-left: 40px;"><i>Draw</i> $x_{t+p}^{(i)} \sim p \left(x_{t+p}^{(i)} \mid x_t^{(i)} \right), i \in [1: N]$</p> <p style="padding-left: 40px;">$\bar{x}_{t+p} = \frac{\sum_{i=1}^N w_t^{(i)} \cdot x_{t+p}^{(i)}}{\sum_{i=1}^N w_t^{(i)}}$</p>
--

CASE STUDY

To test the performance of the proposed travel time predictor, an empirical study is conducted in this section. The test environment is introduced, and then different prediction methods are implemented on the same test data set. Finally, the testing results and discussion are presented.

Test Environment Setup

A freeway stretch from Richmond to Virginia Beach (95 miles long) connected by I-64 and I-264 is selected as the test site in this study. This test site usually experiences high traffic volumes and serious congestion during the summer season, since Virginia Beach is a famous resort location, and the selected freeway stretch serves as the main route heading to the beaches. Consequently, efficient and accurate travel time prediction is needed for travelers in planning their trips and reducing traffic congestion around the area. The evaluation of travel time prediction on the test site is conducted based on probe data from INRIX. The data provided by INRIX are mainly collected by global positioning system (GPS)-equipped vehicles and supplemented with traditional road sensor data, as well as mobile devices and other sources [34]. The probe data on the test site covers 96 freeway segments with a total length of 95 miles. The average segment length is 0.65 miles long, and the length of each segment is unevenly divided in the raw data from 0.1 to 6.36 miles. The location of the study site and the deployment of segments are presented in Figure 8. The raw data provide the average speed for each segment and are collected at one-minute intervals. In this study, the raw data are aggregated at five minute intervals to reduce the noise in calculating the travel times. The aggregated daily traffic data between 2 and 8 p.m. from May 16, 2012 to September 15, 2012 are considered in this study to test the algorithm since the most congested periods are observed during this time frame. Consequently, the prediction performances using different methods are investigated during the peak periods. For each selected day, the instantaneous travel times are calculated for each time interval by assuming the segment speed does not change over time. Given the length of each section of roadway and the corresponding average speed for each time interval, the instantaneous travel time is calculated based on the aggregation of segment travel times at a specific time interval. Conversely, the experienced travel time is calculated from the ground truth data by considering the change of segment speed over time. In other words the speed profiles are piecewise constant speed values and the trip trajectory is a combination of diagonal curves over time and spaces [32].

In total, the travel time data for 123 days are included in this study. The test is conducted using the leave-one-out cross-validation method to provide a consistent validation test. During testing, the different methods are run on each day and the remaining days (122 days) serve as a historical data set. Finally, the average performance across the 123 test days is used to compare the prediction accuracy of the different methods. Leave-one-out cross-validation is a classic model validation technique for assessing how the results of a statistical analysis will generalize to an independent data set. Considering the fact that we only have four months of data on the selected freeway stretch, leave-one-out is an ideal option to quantify the prediction accuracy on this limited data. This method has also been used in many application fields including traffic prediction problems [35].

Several parameters in the proposed particle filter approach are pre-defined for the test. The data sequence length is chosen to be 6 periods (30 minutes), which entails the use of instantaneous travel times over half an hour as the input data sequence. The number of particles N is selected to be 200. The resampling threshold is set at 80% of the total number of particles, so N_{th} is 160. The impact of various parameters on the algorithm performance is quantified later in the paper.

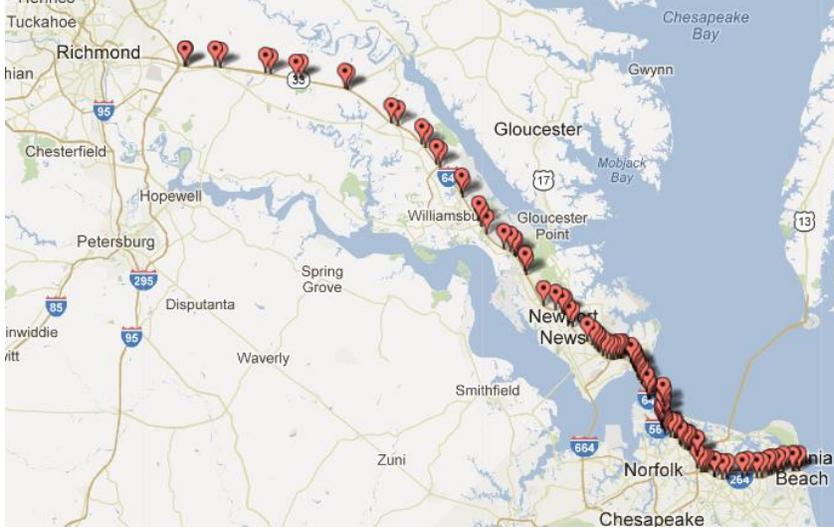


Figure 8: The study site from Richmond to Virginia Beach (source: Google Map).

Comparison of Algorithms and Performance Indices

To better evaluate the performance of the proposed predictor, several different methods are also considered on the same data set. The selected methods include state-of-the-practice instantaneous travel times, two types of state-of-the-art Kalman filter methods and a k -NN method. A detailed description of these methods is provided in this section.

The instantaneous travel time method is the easiest alternative to predict future travel times by assuming the current traffic speed along all the segments remains constant until the completion of the trip. This method is currently used by the Virginia Department of Transportation (VDOT) to display travel time information on variable message signs. Consequently, instantaneous travel times are considered the state-of-practice and used to quantify the tradeoff between simplicity and prediction accuracy.

As mentioned in the literature review, Kalman filters have been widely used in previous studies for real-time travel time predictions [12-14, 18]. In these studies, the problem of travel time prediction is modeled as a linear system as

$$\begin{cases} x_t = \Phi_{t-1} \cdot x_{t-1} + u_{t-1} \\ z_t = x_t + v_t \end{cases} \quad (8)$$

where state variable x_t is the predicted travel time at time t , and Φ_{t-1} is the state transition function to propagate travel times from time $t-1$ to time t ; z_t denotes the travel time measurement; u_{t-1} and v_t are system noises and are assumed to follow the standard normal distribution $N(0,1)$ in this study. Consequently, the travel time in the previous time interval is needed to calculate the predicted travel time. As far as the experienced travel time is concerned, Kalman filter methods cannot be used in real-time applications because previous experienced travel times are delayed significantly when the trip durations are long. Here, both of state and measurement variables are experienced travel times and we assume that the aforementioned problem can be ignored so that Kalman filter methods can be tested and compared with other predictors on the same data set.

According to previous studies using Kalman filter methods, the state-transition function is the key element that requires specification. Consequently, two types of Kalman filter methods are used in this study using two different methods to define the state-transition function. The transition function Φ_{t-1} in the first Kalman filter method (KF1) is defined as the ratio of

measurement values from time interval $t-1$ and $t-2$ using Eq. (12). This alternative is based on the assumption that traffic trends in the short past time periods will continue to propagate into the near future. Conversely, the state-transition function of the second Kalman filter method (KF2) is calculated using Eq. (13) as the average measurement at time t and $t-1$ using aggregated historical data on the same day-of-the-week as the testing day. This definition assumes that data trends from the same time intervals on historical days are consistent with the current day. For multi-step prediction considering a prediction horizon $t+p$, the state-transition function maintains a constant value of Φ_t and the time update equation is iteratively used for p times to calculate the prediction output x_{t+p} . It should be noted that the estimation value corrected by measurement z_t is used as the prediction output for a prediction horizon of zero.

$$KF1: \Phi_{t-1} = z_{t-1}/z_{t-2} \quad (9)$$

$$KF2: \Phi_{t-1} = \overline{z_t}/\overline{z_{t-1}} \quad (10)$$

The k -NN is another effective method, which is widely used to predict travel times for real-time applications [36, 37]. In order to conduct an objective comparison between the k -NN and the proposed method, the same instantaneous travel time sequence input z_t across L time intervals is used in the k -NN method. Thereafter, m numbers of similar data sequences with tail time $\{h_1, h_2, \dots, h_m\}$ can be selected as the candidates from the historical data set Ω as given in Eq. (14). For each candidate with index i , a weight w_i is calculated using the average Euclidean distance from data sequences for the current time z_t and historical time h_i . Moreover, the corresponding experienced travel time departure at h_{i+p} on the selected candidate day i can be obtained from the historical data set. Consequently, the experienced travel time x_{t+p} on the current day can be predicted as the weighted average of the travel times from all candidates using Eq. (15).

$$\begin{aligned} H_c &= \{h_1, h_2, \dots, h_m\} \\ \text{For } i &= 1:m \\ h_i &= \arg \min_{h \in \Omega} |z_t - h|, \Omega = \Omega - h_i \end{aligned} \quad (11)$$

End For

$$w_i = |z_t - h_i| / \sum_{i=1}^m |z_t - h_i| \quad (12)$$

$$x_{t+p} = \sum_{i=1}^m x_{h_i+p} \cdot w_i$$

Different combinations of parameters were tested and the optimum set of parameters was identified as $L = 5$ and $m = 20$, which corresponds to the least prediction error [36]. These parameters are used to test the k -NN method using the same data set and to serve as a comparison with other methods. It should be noted that the k -NN method is different from the proposed approach even if the number of candidates in the k -NN method is equal to the number of particles. The reason lies in the fact that there is no data propagation process in the k -NN algorithm, and all candidates are blindly selected from each time interval based on its similarity measure (shortest Euclidean distance) to the current travel time sequence.

To assess the different methods, the performance criteria are specified using both the absolute and relative prediction errors. The Mean Absolute Error (MAE) is the average absolute difference between the predicted travel time and ground truth using Eq. (16). The corresponding Mean Absolute Percentage Error (MAPE) is the average absolute percentage change between the predicted and the true values relative to the true value as demonstrated in Eq. (17).

$$MAE = (\hat{a} |TT - TT|) / N_{TT} \quad (13)$$

$$MAPE = (\sum |TT - \widehat{TT}| / \widehat{TT}) / N_{TT} \quad (14)$$

where TT is the predicted travel time, \widehat{TT} denotes the ground truth value of experienced travel time, and N_{TT} is the total number of predicted travel times.

Test Results

The average absolute and relative prediction errors produced by the five methods are summarized in Table 5. As demonstrated in the table, the least prediction errors are produced by the proposed non-explicit state transition particle filter approach. Among all the methods, KF1 provides the worst performance, with a significant degradation in performance with an increase in the prediction horizon. The precondition of this method is that the short past travel time relationship continues into the near future. However, this assumption is less valid as the prediction horizon increases. KF2 produces marginally higher prediction errors compared to the instantaneous method for prediction horizons between 0 to 20 minutes and then produces slightly lower errors for longer prediction horizons. Such results demonstrate that the simple average values of measured travel times on the same day of week from previous weeks do not capture the change in experienced travel times. The k -NN method outperforms the instantaneous travel time method. The absolute error produced by the k -NN predictor increases from 10.69 to 14.79 minutes (a 38% increase) when the prediction horizon increases from 0 to 60 minutes. The results demonstrate that the average absolute error for the proposed predictor only increases from 8.26 to 10.54 minutes (an increase of 27%) when predicting the experienced travel time for departures from the current time to one hour later. Alternatively, the prediction error produced by the instantaneous travel time method increases at a much higher rate from 11.54 to 19.25 minutes, which is an increase of 67%. In conclusion the results demonstrate that the proposed algorithm outperforms the state-of-the-practice and state-of-the-art methods, especially for longer prediction horizons.

Table 5: Prediction results by different methods for various prediction horizons.

		Prediction Horizon (min)						
		0	10	20	30	40	50	60
Instantaneous	MAE (min)	11.54	12.97	14.37	15.74	17.04	18.21	19.25
	MAPE (%)	10.67	12.09	13.44	14.79	16.08	17.23	18.25
KF1	MAE (min)	12.45	14.16	16.34	19.01	21.61	24.09	26.40
	MAPE (%)	11.48	13.10	15.18	17.78	20.36	22.86	25.14
KF2	MAE (min)	12.22	13.29	14.39	15.53	16.45	17.18	17.89
	MAPE (%)	11.32	12.38	13.45	14.58	15.49	16.24	16.94
k -NN	MAE (min)	10.69	11.46	12.20	12.92	13.64	14.26	14.79
	MAPE (%)	9.38	10.09	10.75	11.38	12.01	12.52	12.97
NSPF	MAE (min)	8.26	8.81	9.27	9.65	10.03	10.30	10.54
	MAPE (%)	7.32	7.80	8.19	8.53	8.85	9.08	9.29

The relative absolute errors associated with the five methods are presented in Figure 9. The figure clearly demonstrates a significant degradation in the prediction accuracy with time for the KF1, KF2, and instantaneous methods. Alternatively, the k -NN and the proposed methods produce consistent errors over the one hour prediction horizon. Specifically, the relative absolute

error produced by the k -NN method increases from 9.38% to 12.97% (a 38% increase), which is greater than the error associated with the NSPF method (a 27% increase). It should be noted that the relative errors produced by the proposed NSPF approach are always below 10% for the different prediction horizons within one hour, which indicates the prediction performance of the proposed method is much more reliable compared to the other four methods.

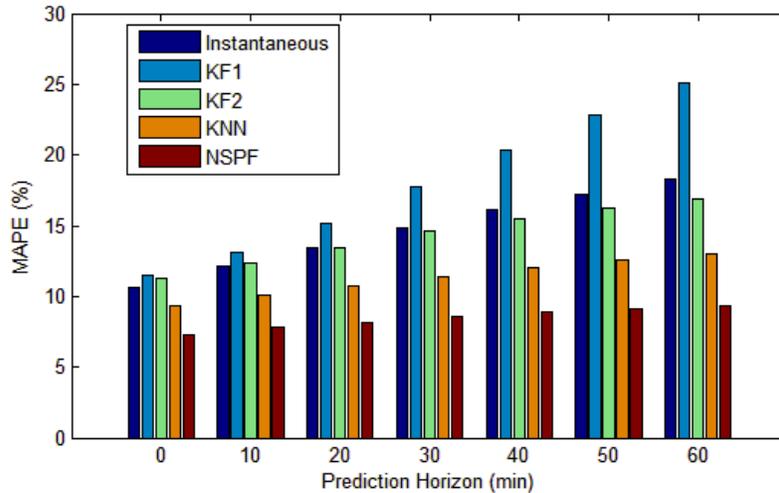
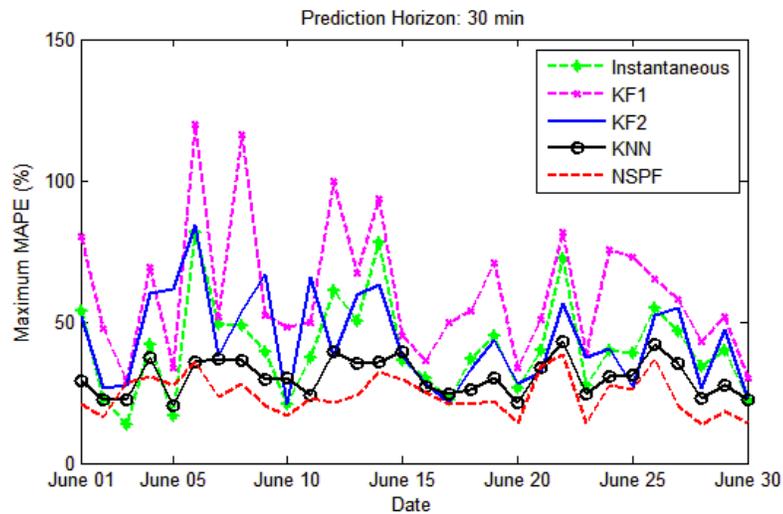


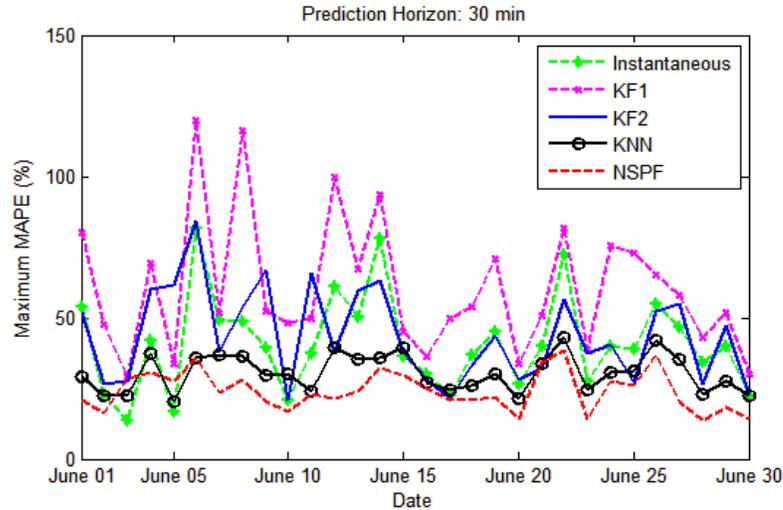
Figure 9: MAPE by different methods for various prediction horizons.

To investigate the maximum deviation between prediction results and ground truth data, the maximum MAPE produced by the five methods on different testing days in June 2012 are



selected and presented in

Figure 10. These results are for a prediction horizon of 30 minutes. The worst result is generated by the KF1 method (i.e., maximum MAPE ranging from 29.3% to 120%), followed by the KF2 and instantaneous methods (i.e., maximum MAPEs ranging from 20.9% to 84.3% and 13.6% to 81.5%, respectively). The maximum MAPE produced by the k -NN method varied between 20.4% and 43.2%, which is significantly better than the previous three methods. More importantly, the best performance corresponds to the proposed NSPF method with a maximum MAPE ranging from 13.5% and 38.2%. The results clearly demonstrate that the proposed predictor still produces the best performance even under the worst case conditions.



(b)

Figure 10: Maximum MAPE by different methods on June 2012.

Considering the prediction errors produced by the two Kalman filter methods are similar or worse than the instantaneous travel time estimates, these methods will not be considered in any further evaluations. The travel time curves for the remaining three methods are compared with the ground truth data for June 21, 23 and 29, as illustrated in Figure 11. These days include different traffic conditions on a weekday, weekend and Friday afternoon, respectively. Both the instantaneous and the k -NN method predictions experience a temporal lag relative to the ground truth data, especially during the formation and dissipation of congestion. Specifically, the instantaneous travel time method significantly underestimates the ground truth when congestion is forming, and overestimates the ground truth travel time when congestion is dissipating. Comparatively, the proposed method improves the prediction performance when congestion is forming and dissipating but still lags in some instances. For example, the red curve generated by the proposed method lags and oscillates between 2 to 4 p.m. on June 21, 2012. Similarly, the proposed method lags during the congestion period around 3 p.m. on June 29, 2012.

The NSPF not only predicts the expected travel time but can predict the travel time distribution. The 95% and 5% confidence intervals of the predicted travel times are calculated as the upper and lower boundaries in Figure 11. The gray shadow area between the boundaries covers most of the ground truth data temporal variation, which demonstrates that the proposed approach provides a good accuracy to predict travel time reliability. It should be noted that the researchers did not specify whether the historical data set comprised weekday or weekend data. This is one of the advantages of the proposed method. That is, if the test day is a weekend, the similar traffic pattern in historical weekends is automatically selected as a particle associated with larger weights, which contributes more significantly to the prediction.

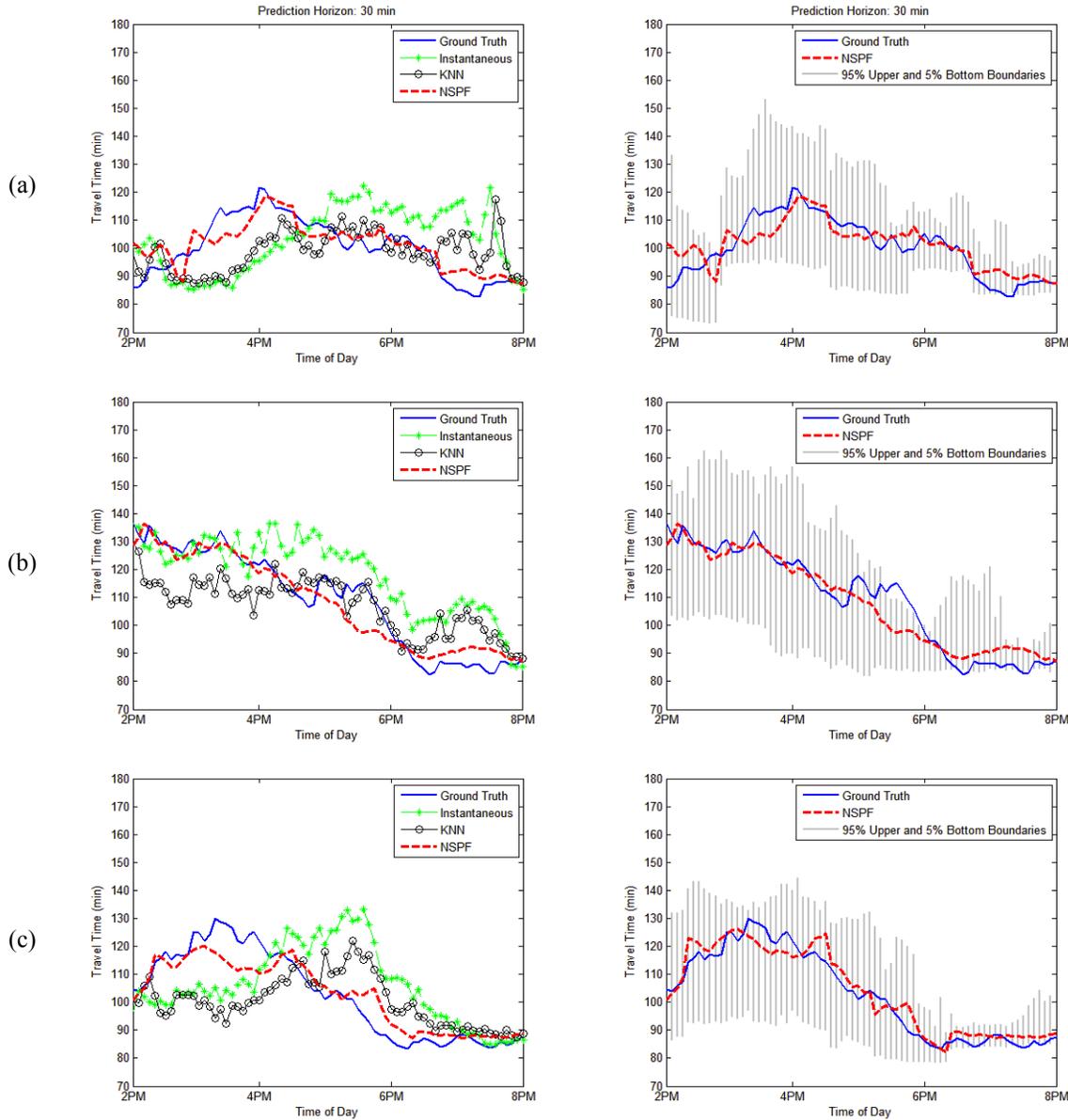


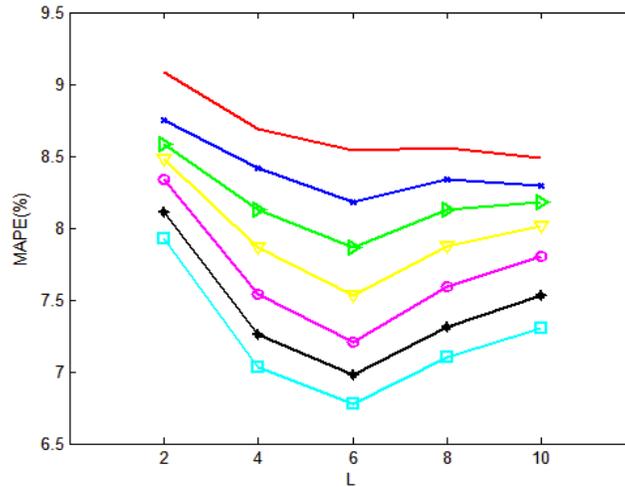
Figure 11: Travel time prediction results by three methods and the NSPF confidence boundaries on (a) June 21, 2012 (Thursday); (b) June 23, 2012 (Saturday); (c) June 29, 2012 (Friday).

Another advantage of the proposed NSPF approach is the fast computation time. This computational efficiency allows the model to be implemented in real-time in a TMC. The testing of the NSPF travel time predictor was performed on a personal computer with Intel dual core CPU, 2.40 GHz and 4GB of random-access memory within the MATLAB 2012b environment. Under the scenario of setting $L=6$, $N_{th}=160$ and $N=200$, the total average computation time for one day between 2 and 8 p.m. was 5.76 seconds. Consequently, the calculation of a single prediction only requires 0.08 seconds. Clearly, the computational performance of the proposed predictor meets the requirements for real-time applications. It's also worth noting that the INRIX probe data have a national coverage on most of the roadways in the United States and no off-line training is

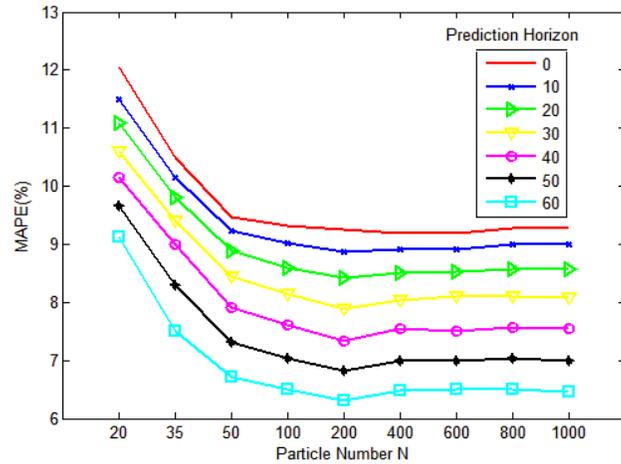
needed. Therefore, the proposed method can be implemented at any location with a similar data set.

Sensitivity Analysis

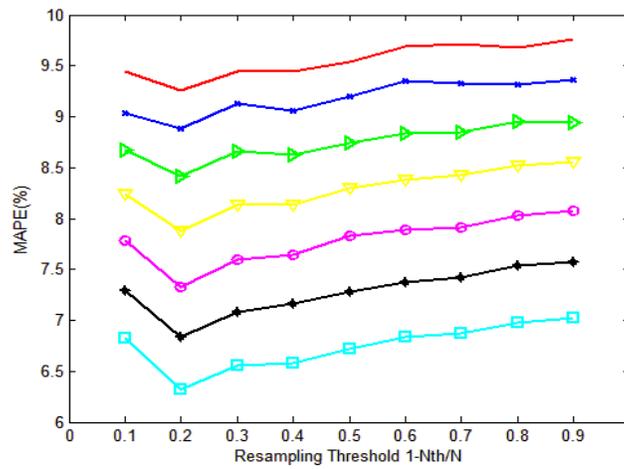
A sensitivity analysis is conducted to quantify the impact of the three parameters: L and N , and resampling ratio threshold $I-N_{th}/N$ on the prediction accuracy of the proposed method. The parameter L determines the matching window width in the particle propagation and re-selection processes, so a larger value of L results in a wider matching window and vice versa. Here, the values vary from 2 to 10 and are used to calculate the average MAPE for different prediction horizons, as presented in Figure 12 (a). The figure demonstrates that the minimum prediction error is obtained when L equals 6. Consequently, the best matching window is a half-hour of traffic speed data along all the freeway segments. Moreover, different particle numbers are also investigated to calculate the relative errors, as shown in Figure 12 (b). Generally, the prediction error decreases with larger particle numbers. However, the error reaches the minimum value when the particle number is 200, and then the prediction error increases slightly with a particle number greater than 200. Consequently, a particle number of 200 was used in this case study. Lastly, different values of resampling thresholds from 10% to 90% are tested, and the corresponding prediction errors with different prediction horizons are presented in Figure 12 (c). The optimum resampling threshold is reached when 20% of the total particles are resampled during each time interval. The same analysis can be conducted on different sites or roadway compositions to find the optimum model parameters.



(a)



(b)



(c)

Figure 12: Sensitivity analysis.

The prediction results in the case study demonstrate that the proposed NSPF method can produce highly accurate and reliable multi-step-ahead predicted travel times, based upon the comparison with the four state-of-the-art and state-of-the-practice methods under various traffic conditions.

CONCLUSIONS

This paper develops a new particle filter approach for the real-time application of multi-step travel time prediction using real-time and historical data set. Unlike previous studies that require an underlying physical model in modeling the state-transition function between predicted and previous travel times, the proposed particle filter uses historical trends to model the state-transition trend. A partial resampling strategy is then developed to address the degeneracy problem by replacing invalid or low weighted particles with historical data that provide similar data sequences to real-time traffic measurements. In this way, each particle can generate a travel time prediction value and a corresponding weight reflected by the similarity of the traffic patterns

between each particle and the real-time traffic measurement. Consequently, the prediction can produce a distribution of travel times by aggregating all weighted particles.

The probe data on the selected freeway stretch from Richmond to Virginia Beach along I-64 and I-264 are used to investigate the performance of different prediction approaches. Considering the fact that only four months of data are used on the selected freeway stretch, the leave-one-out cross validation method is an ideal option to quantify the prediction accuracy on this limited data. The MAE and MAPE of prediction results demonstrate the proposed method produces the least deviation from ground truth travel times, compared to instantaneous travel time, two Kalman filter algorithms and k -NN method. Besides, the maximum daily prediction errors on June 2012 indicate the proposed NSPF method outperforms other methods by maintaining a stable performance for all test days. Moreover, the proposed approach provides good accuracy in predicting travel time reliability. Lastly, the fast computation speed and online processing ensure the proposed NSPF can be used in real-time applications.

The proposed predictor has only been used to predict freeway travel times. Nevertheless, the essential proposed particle filter method does not require certain types of data sources and can be applied for nonlinear data tracking problems in other application fields. The proposed approach is also flexible in addressing data prediction problems in other application fields and can potentially produce a comparatively high accuracy if enough historical data are provided. The implementation of the proposed predictor into arterial travel time prediction will be considered in the future. Moreover, rather than using a data sequence as the input, future research will consider the use of spatiotemporal traffic information (e.g. speed matrix) to predict travel times.

ACKNOWLEDGEMENTS

This research effort was funded partially by Virginia Department of Transportation (VDOT) and partially by the Mid-Atlantic Universities Transportation Center (MAUTC). The authors also appreciate the input from Catherine McGhee, Ralph L. Jones and Philomena Lockwood from VDOT.

REFERENCES

- [1] J. Chu. (2011). *Travel Time Messages on Dynamic Message Signs*. Available: <http://ops.fhwa.dot.gov/travelinfo/dms/signs.htm>
- [2] T. Pan, A. Sumalee, R. Zhong, and N. Indra-payoong, "Short-Term Traffic State Prediction Based on Temporal-Spatial Correlation," *IEEE TRANSACTIONS ON INTELLIGENT TRANSPORTATION SYSTEMS*, vol. 14, pp. 1242-1254, 2013.
- [3] L. Mihaylova, A. Hegyi, A. Gning, and R. K. Boel, "Parallelized Particle and Gaussian Sum Particle Filters for Large-Scale Freeway Traffic Systems," *IEEE Transactions on Intelligent Transportation Systems*, vol. 13, pp. 36-48, 2012.
- [4] H. Chen, H. A. Rakha, and S. A. Sadek, "Real-time Freeway Traffic State Prediction: A Particle Filter Approach," in *14th International IEEE Conference on Intelligent Transportation Systems*, Washington, DC, USA, 2011, pp. 626-631.
- [5] Z. Ye and Y. Zhang, "Speed estimation from single loop data using an unscented particle filter," *Computer - Aided Civil and Infrastructure Engineering*, vol. 25, pp. 494-503, 2010.
- [6] D. B. Work, O.-P. Tossavainen, S. Blandin, A. M. Bayen, T. Iwuchukwu, and K. Tracton, "An ensemble Kalman filtering approach to highway traffic estimation using GPS enabled mobile devices," presented at the 47th IEEE Conference on Decision and

- Control, 2008.
- [7] L. Mihaylova, R. Boel, and A. Hegyi, "Freeway Traffic Estimation within Particle Filtering Framework," *Automatica*, vol. 43, pp. 290-300, 2007.
 - [8] Y. Wang and M. Papageorgiou, "Real-Time Freeway Traffic State Estimation Based on Extended Kalman Filter: A General Approach," *Transportation Research Part B*, vol. 39, pp. 141-167, 2005.
 - [9] M. Yildirimoglu and N. Geroliminis, "Experienced Travel Time Prediction for Congested Freeways," *Transportation Research Part B: Methodological*, vol. 53, pp. 45-63, 2013.
 - [10] H. Rakha, H. Chen, A. Haghani, and K. Farokhi, "Assessment of Data Quality Needs for use in Transportation Applications," MAUTC-2011-01, 2013.
 - [11] X. Fei, C.-C. Lu, and K. Liu, "A Bayesian Dynamic Linear Model Approach for Real-time Short-term Freeway Travel Time Prediction," *Transportation Research Part C: Emerging Technologies*, vol. 19, pp. 1306-1318, 2011.
 - [12] J.-S. Yang, "Travel Time Prediction Using the GPS Test Vehicle and Kalman Filtering Techniques," in *Proceedings of the 2005 American Control Conference*, 2005, pp. 2128-2133.
 - [13] S. I.-J. Chien and C. M. Kuchipudi, "Dynamic Travel Time Prediction with Real-Time and Historic Data " *Transportation Research Record: Journal of the Transportation Research Board*, vol. 129, pp. 608-616, 2003.
 - [14] M. Chen and S. I. J. Chien, "Dynamic Freeway Travel-Time Prediction with Probe Vehicle Data: Link Based Versus Path Based," *Transportation Research Record: Journal of the Transportation Research Board*, vol. 1768, pp. 157-161, 2001.
 - [15] H. Chen, H. A. Rakha, S. A. Sadek, and B. J. Katz, "A Particle Filter Approach for Real-time Freeway Traffic State Prediction," in *Transportation Research Board 91st Annual Meeting*, Washington D.C., 2012.
 - [16] J. Xia, M. Chen, and W. Huang, "A Multistep Corridor Travel-Time Prediction Method Using Presence-Type Vehicle Detector Data," *Journal of Intelligent Transportation Systems: Technology, Planning, and Operations*, vol. 15, pp. 104-113, 2011.
 - [17] C. P. I. van Hinsbergen, J. W. C. van Lint, and H. J. van Zuylen, "Bayesian combination of travel time prediction models," *Transportation Research Record: Journal of the Transportation Research Board*, vol. 2064, pp. 73-80, 2008.
 - [18] C. Nanthawichit, T. Nakatsuji, and H. Suzuki, "Application of Probe-vehicle Data for Real-time Traffic-state Estimation and Short-term Travel-time Prediction on a Freeway," *Transportation Research Record: Journal of the Transportation Research Board*, pp. 49-59, 2003.
 - [19] Y. Wang, M. Papageorgiou, and A. Messmer, "Real-Time Freeway Traffic State Estimation Based on Extended Kalman Filter: Adaptive Capabilities and Real Data Testing," *Transportation Research Part A*, vol. 42, pp. 1340-1358, 2008.
 - [20] B. Ristic, S. Arulampalam, and N. Gordon, *Beyond the Kalman Filter: Particle Filters for Tracking Applications*. Boston, MA, 2004.
 - [21] M. S. Arulampalam, S. Maskell, N. Gordon, and T. Clapp, "A Tutorial on Particle Filters for Online Nonlinear/Non-Gaussian Bayesian Tracking," *IEEE Transactions on Signal Processing*, vol. 50, pp. 174-188, 2002.
 - [22] A. Panangadan and A. Talukder, "A Variant of Particle Filtering Using Historic Datasets for Tracking Complex Geospatial Phenomena," presented at the 18th SIGSPATIAL International Conference on Advances in Geographic Information Systems, San Jose,

- California, USA, 2010.
- [23] D. Mikami, K. Otsuka, and J. Yamato, "Memory-based Particle Filter for Face Pose Tracking Robust under Complex Dynamics," in *Computer Vision and Pattern Recognition*, 2009.
 - [24] D. Mikami, K. Otsuka, and J. Yamato, "Memory-based Particle Filter for Tracking Objects with Large Variation in Pose and Appearance," in *The 11th European Conference on Computer Vision*, 2010, pp. 215-228.
 - [25] A. Basharat and M. Shah, "Time Series Prediction by Chaotic Modeling of Nonlinear Dynamical Systems," in *12th International Conference on Computer Vision*, 2009, pp. 1941-1948.
 - [26] A. G. Parlos, O. T. Rais, and A. F. Atiya, "Multi-step-ahead Prediction using Dynamic Recurrent Neural Networks," *Neural Networks*, vol. 13, pp. 765-786, 2000.
 - [27] R. Boné and M. Crucianu, "Multi-step-ahead Prediction with Neural Networks: A Review," *9emes rencontres internationales: Approches Connexionnistes en Sciences*, vol. 2, pp. 97-106, 2002.
 - [28] X. Zeng and Y. Zhang, "Development of Recurrent Neural Network Considering Temporal - Spatial Input Dynamics for Freeway Travel Time Modeling," *Computer - Aided Civil and Infrastructure Engineering*, 2013.
 - [29] F. Gustafsson, F. Gunnarsson, N. Bergman, U. Forssell, J. Jansson, R. Karlsson, *et al.*, "Particle filters for positioning, navigation, and tracking," *Signal Processing, IEEE Transactions on*, vol. 50, pp. 425-437, 2002.
 - [30] A. Doucet, N. de Freitas, and N. Gordon, "An introduction to sequential Monte Carlo methods," *Sequential Monte Carlo methods in practice*, pp. 3-14, 2001.
 - [31] H. Chen and H. A. Rakha, "Prediction of Dynamic Freeway Travel Times based on Vehicle Trajectory Construction," in *15th International IEEE Conference on Intelligent Transportation Systems*, Anchorage, AK, 2012, pp. 576 - 581.
 - [32] J. W. C. van Lint and N. J. van der Zijpp, "Improving A Travel Time Estimation Algorithm by Using Dual Loop Detectors," *Transportation Research Record: Journal of the Transportation Research Board*, vol. 1855, pp. 41-48, 2003.
 - [33] M. Bolić, P. M. Djurić, and S. Hong, "Resampling Algorithms for Particle Filters: A Computational Complexity Perspective," *EURASIP Journal on Applied Signal Processing*, vol. 1, pp. 2267-2277, 2004.
 - [34] INRIX. (2012). <http://www.inrix.com/trafficinformation.asp>. Available: <http://www.inrix.com/trafficinformation.asp>
 - [35] J. Kwon, B. Coifman, and P. Bickel, "Day-to-day Travel-time Trends and Travel-time Prediction from Loop-detector Data," *Transportation Research Record: Journal of the Transportation Research Board*, vol. 1717, pp. 120-129, 2000.
 - [36] W. Qiao, A. Haghani, and M. Hamed, "Short Term Travel Time Prediction Considering the Weather Impact," in *Transportation Research Board 91st Annual Meeting*, Washington D.C., 2012.
 - [37] B. I. Bustillos and Y.-C. Chiu, "Real-Time Freeway-Experienced Travel Time Prediction Using N-Curve and k Nearest Neighbor Methods," *Transportation Research Record: Journal of the Transportation Research Board*, vol. 2243, pp. 127-137, 2011.

APPENDIX C

Multi-step Prediction of Experienced Travel Times using Agent-based Modeling

This article is currently under review and may be cited as a working version: Chen, H. and Rakha, H.A., Multi-step Prediction of Experienced Travel Times using Agent-based Modeling. Working paper.

Introduction

Tackling congestion (both recurrent and non-recurrent) has proven to be a challenge for highway agencies. Adding capacity in response to congestion is becoming less of an option for these agencies due to a combination of financial, environmental, and social issues. Therefore, the main focus has been on improving the performance of existing facilities through continuous monitoring and dissemination of traffic information. The minimum that can be accomplished is to inform the public or, specifically, the potential users of what they should expect on the roadways before and during their trips. Additionally, this information can be applied to provide alternatives to users so that they may make informed decisions about their trips. This is the essence of Advanced Traveler Information System (ATIS) applications such as 511 that have been implemented nationwide. In many states, relevant traffic information is also posted on variable message signs (VMSs) that are strategically positioned along highways. Consequently, there is a need to provide predicted travel times to road users for better planning their trips and choosing their route of travel, further reducing congestion.

Various traffic sensing technologies have been used to collect traffic data for use in computing travel times, including point to point travel time collection (e.g. license plate recognition systems, automatic vehicle identification systems, mobile, Bluetooth, probe vehicle, etc.) and station based traffic state measuring devices (e.g. loop detector, video camera, remote traffic microwave sensor, etc.). Private companies such as INRIX integrate different sources of measured data to provide section-based traffic speed or travel time, which can be used to construct traffic speed matrix over spatial and temporal and thus is used in this paper. The benefit of using temporal-spatial speed data is that travel time can be easily estimated afterward [1]. More importantly, such data provides the flexibility for scalable applications on traffic networks. By providing section-based traffic state data, generally there are two approaches to compute travel time depending on the trip experience, which are instantaneous and experienced travel time [2, 3].

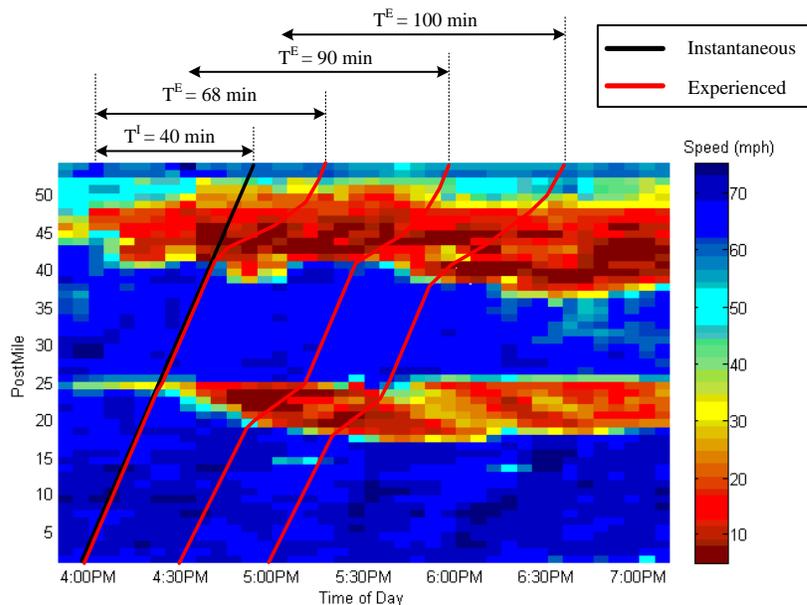


FIGURE 13 Spatiotemporal traffic speed map and trip trajectories on I-66 during June 22 2013.

Previous research has demonstrated that prediction accuracy typically deteriorates quickly with the increase in prediction horizon [4]. In order to demonstrate the discrepancy between instantaneous and experienced travel times, especially the errors of using instantaneous information for multi-step prediction of experienced travel time, a spatiotemporal traffic speed data provided by INRIX is presented in FIGURE 13. The traffic data was collected along I-64 from Richmond to Norfolk during afternoon peak hours on June 22 2013. The trip trajectories are plotted on the contour speed map. According to the black trajectory, the instantaneous travel time is calculated as 40 minutes for time interval at 4 p.m. Although the traffic on the selected route is uncongested at 4 p.m., two bottlenecks rapidly form afterward. Consequently, the instantaneous travel time at 4 p.m. underestimates the experienced travel time by 28 minutes, 50 minutes and 60 minutes for the prediction horizon of 0 minutes, 30 minutes and 60 minutes, respectively. These results demonstrate that the instantaneous travel time may not be a good predictor of experienced travel time, especially for multi-step prediction.

During the past decades, many studies have been conducted attempting to predict travel times. According to the manner of modeling, these methods can be classified into parametric methods (e.g. linear regression models [5, 6], Kalman filter methods [7-9], Auto-Regressive Integrated Moving Average (ARIMA) models [10-12]) and non-parametric methods (e.g. K-Nearest Neighbor (k -NN) [13-15], artificial neural network (ANN) models [16-18] and support vector regression (SVR) methods [19, 20]). These techniques are implemented through direct or indirect procedures to predict travel times using different types of state variables [21]. Travel time is directly used as the state variable in parametric or non-parametric methods to predict travel times. Indirect procedures are performed using other variables (such as traffic speed, density, flow, occupancy, etc.) as the state variable to predict the future traffic speed over space and time, and then travel times can be calculated based on the spatiotemporal speed map [1]. This paper attempts to predict experienced travel times for departures at current or future time intervals. For real-time application, instantaneous travel time can be obtained as the summation of section travel times at every time interval. Nevertheless, experienced travel time can only be obtained after the completion of the trip, because the spatiotemporal evolution of speed should be considered. In this case, the experienced travel time for the previous time interval usually is not available for predicting travel time in the next interval, especially for long trips. Consequently, many existing methods cannot work well for predicting experienced travel times [22].

Other than real-time information, historical data provide a pool of experienced traffic patterns that can be used to predict travel times. ANN methods are widely used to generate the predictor by training on a large historical dataset. However, the same problem exists, namely the prediction accuracy deteriorates rapidly for multi-step predictions [23, 24]. Considering the stochastic nature of traffic behavior, it is very difficult to predict travel time for multi-step time horizons accurately. For instance, a time-delayed state-space neural network (TDSSNN) approach was recently developed and demonstrated to outperform other popular ANN methods for travel time prediction. However, the prediction error for the proposed TDSSNN method on incident-free data increased from 5.4% to 15.1% for a prediction horizon of 5 minutes to 25 minutes [25]. In addition, there are several other deficiencies for ANNs, such as high computational costs for data the training process, a lack of the flexibility to deal with non-recurrent traffic patterns, and difficulty to implement on large-scale traffic networks or different sites. Consequently, there is a need to develop a robust method for multi-step prediction of experienced travel time, yet is still easily transferable to other sites without the need for a data training process.

Considering the aforementioned problems, the concept of agent-based modeling is used in this paper to address travel time prediction problems. Agent-based modeling has been widely used for problems of decision making, complex social system and etc. [26]. The advantages lie on the feature that each agent can behave as an individual expert decision system, so that individual agent has the ability to analyze data input and produce its own decision output by constructing rules. More importantly, different groups of agents can cooperate to model complex social systems. In the past decades, agent-based modeling has been successfully applied to various transportation problems, because of the flexibility and computational advantages of modeling complex transportation systems [27]. Although the direct application to predict travel time using agent-based modeling has not been developed yet, several similar applications have already been attempted to deal with time series prediction problems in other application fields. For instance, a group of individual cooperating agents are used to simulate different components of the stock trading process and tested to provide accurate prediction for stock buying/selling decisions [28]. Similar approaches have been developed to predict the evolution of market shares for electric vehicles [29] and the price change of the US wholesale power market [30]. These examples demonstrate the agent-based modeling methods can efficiently and accurately solve time-series prediction problems in complex systems. It should be noted that the state-transition of travel times over neighboring time intervals also has strong nonlinear trends as the aforementioned problems in other fields. In addition, a set of guidelines to use agent-based models for data forecasting problems are developed in [31], and the related problems of building a predictor using agent-based model for different categories of forecasting problems have been discussed. Consequently, there is a need to explore the potential of using the concept of agent-based model for travel time prediction.

In this paper, an agent-based modeling approach is developed to predict experienced travel times using real-time and historical traffic data. At the microscopic level, each agent acts as an expert and a set of agent interactions are developed to produce a recommendation for future experienced travel time with a measurement of recommendation confidence. Consequently, the aggregation of each agent's recommendation (predicted travel time with associated weight) provides a macroscopic level of output – a predicted travel time distribution. The INRIX probe data from Richmond to Virginia Beach along I-64 and I-264 in 2012 are used to test the performance of the proposed method. The results show that the agent-based modeling approach produces the least prediction error compared with other state-of-practice and state-of-art methods (instantaneous travel time, historical average and k-nearest neighbor), and maintains less than 9% error for future trip departures from the current time to 60 minutes later.

The remainder of this paper is organized as follows. The framework of the proposed agent-based modeling method is provided together with descriptions of the microscopic agent interaction rules. This is followed by an implementation on a selected test site and a comparison with other predictors to estimate experienced travel times considering different prediction horizons (0~60 minutes). The last section includes the summary conclusions of the proposed method and recommendations for future research.

Agent-based Model

The concept of correlating real-time and historical traffic measurement data using an agent-based model to predict travel time and the details of agent interactions are described in this section.

In this research, we assume that the traffic speed data for each time interval is updated along all roadway segments from the trip origin to destination. In this way, the daily traffic measurement data can be represented as a matrix, in which each cell is an average speed for the corresponding time interval and roadway segment. Here, different colors are used to represent the speed value - the dark blue denotes free flow speed and the bright red corresponds to congestion. Therefore, the traffic data matrix is demonstrated as a color map. At the same time, the experienced travel time can be computed by providing the spatiotemporal traffic speed map [1]. Consequently, a traffic speed map and an experienced travel time curve are included for each day.

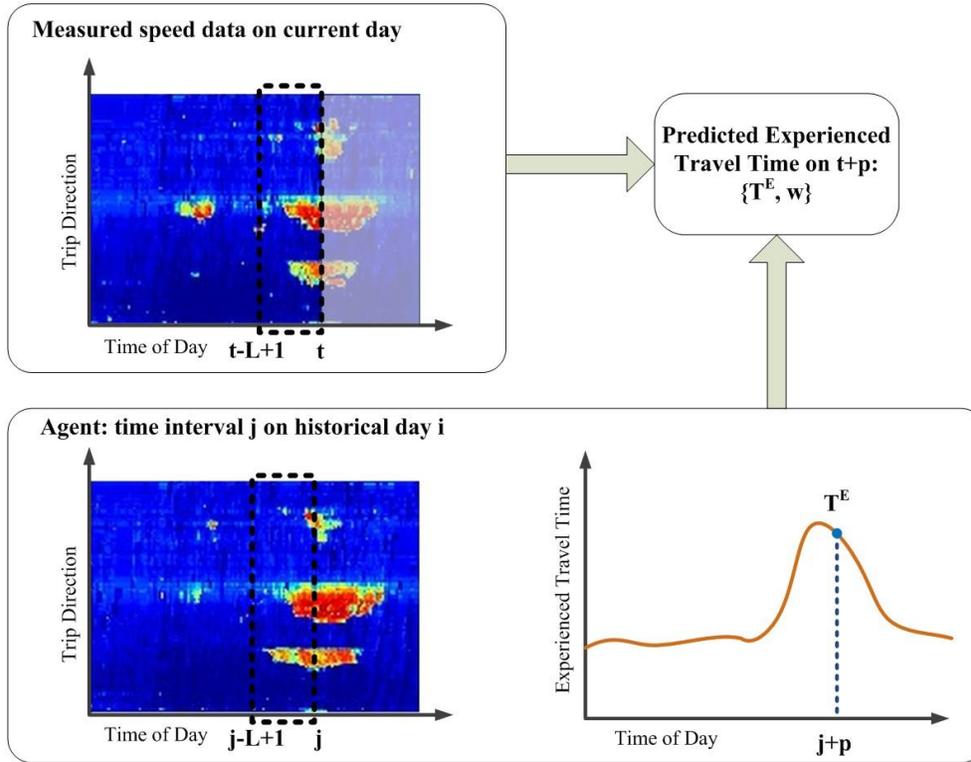


FIGURE 14: The illustration of travel time prediction by a single agent.

An illustration of the agent-based modeling approach is presented in FIGURE 14. Each agent corresponds to a specific time interval on a historical day. In this example, i and j are the day and time interval indices of the sample agent. Assume the current traffic pattern for the testing day is denoted by the speed matrix from time $t-L+1$ to t across all the segments. Thus the agent is used to provide a prediction of experienced travel time at $t+p$. The prediction result includes a value of travel time T^E and a corresponding weight value w . The former value is obtained by finding the experienced travel time at time interval $j+p$ on historical day i . The latter value is calculated by comparing the dissimilarity between two matrixes relative to the current day and the historical day, represented by dotted rectangle windows on FIGURE 14. The details of the framework of the proposed agent-based modeling approach and the interactions between agents are described below.

The Framework of Agent-based Modeling Approach

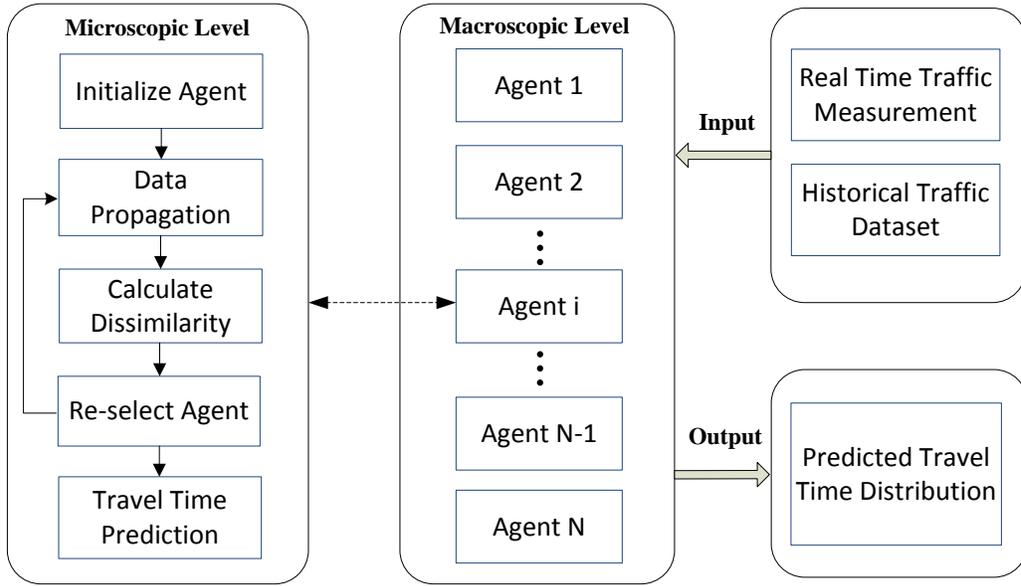


FIGURE 15: The framework of agent-based model.

From a traditional expert system perspective, each expert makes a recommendation based on its own experience of the target problem. The proposed agent-based model adopts the same logic in order to predict travel times using real-time and historical traffic data. Each agent represents an expert, who is responsible for providing a travel-time prediction estimate at each time interval. The framework of the proposed agent-based model is presented in Figure 7. At a microscopic level, each agent interacts individually according to the real-time and historical traffic status. Different interaction rules are constructed in order to simulate the process of choosing and updating individual experts according to its performance (similarity to real-time traffic information) for each time interval. The aggregation of each agent's recommendation (predicted travel time with associated weight) provides a macroscopic level of output – predicted travel time distribution.

Assume the current time is t , the available measurement data $u_{N_{seg} \times L}$ is the speed matrix from short past $t-L+1$ to t along all the freeway segments (total segment number is N_{seg}). Here, the speed matrix is denoted by the tail time as variable z_t , which also represents the real-time traffic status and is updated every time interval. The real-time traffic status and historical data will be used to conduct a data mining process to predict the experienced travel time T_{t+p}^E which departs at time $t+p$. Each agent represents an expert who can provide a prediction estimate based on the experience of a specific historical day. Consequently, the i^{th} agent, denoted by $x_t^{(i)}$, corresponds to a day index $d_t^{(i)}$ from the historical dataset Ω and a time index $j_t^{(i)}$ on that day. The corresponding speed matrix from time interval $d_t^{(i)}-L+1$ to $d_t^{(i)}$ along N_{seg} segments can be obtained as $\Omega(d_t^{(i)}, j_t^{(i)})$. The difference between two speed matrices from real-time measurement z_t and historical experience $\Omega(d_t^{(i)}, j_t^{(i)})$ corresponding to the i^{th} agent can be used to calculate the confidence level of an agent, denoted by weight $w_t^{(i)}$. In addition, the experienced travel time can be calculated in the historical dataset, so that the i^{th} agent can produce a recommendation of travel time $T^E(d_t^{(i)}, j_t^{(i)}+p)$ which departs at time $j_t^{(i)}+p$ for historical day $d_t^{(i)}$. Finally, the prediction output is denoted by the integration of each agent's recommendation with the corresponding

travel time $T^E(d_t^i, j_t^i + p)$ and weight $w_t^{(i)}$. The details of agent interaction rules are presented as blow.

Initialize Agent

At beginning, each agent should be assigned an initial correspondence on the historical dataset. The index of day $d_0^{(i)}$ is randomly selected from the historical dataset Ω (a total of D days) and then the index of time interval $j_0^{(i)}$ is randomly selected for that day. The initial agent set x_0 can be represented as

$$x_0 : \{x_0^{(i)} \mid x_0^{(i)} = \Omega(d_0^{(i)}, j_0^{(i)}), i \in [1: N]\} \quad (15)$$

Data Propagation

In order to match with the new incoming measurement data, the corresponding speed matrix for each agent needs to propagate along the time domain. This process is conducted by maintaining the same day index and increasing the time index by an additional time interval as

$$d_t^{(i)} = d_{t-1}^{(i)}, j_t^{(i)} = j_{t-1}^{(i)} + 1, i \in [1: N] \quad (16)$$

In the proposed algorithm, each daily traffic data is considered as a separate dataset from the adjacent days, even though the end of one day is followed by the start of next day. The reason for using each daily traffic dataset separately is based on two considerations. First, the adjacent day's traffic data may not be available in the historical dataset. Second, the measured traffic data may not provide full coverage for 24 hours. For instance, it is possible that the traffic data is only measured during the day time or peak hours. Consequently, a process to identify valid agents is developed to examine if the data propagation reaches the boundary of the same day. Here, the last time interval of the historical traffic speed matrix on day $d_t^{(i)}$ is denoted by $H_{d_t^{(i)}}$. Considering the prediction horizon p , the collection of the valid agent is identified as

$$\Psi_t = \{i \mid j_t^{(i)} \leq H_{d_t^{(i)}} - p, i \in [1: N]\} \quad (17)$$

Calculate Dissimilarity Measure

This process aims to calculate the weight of each valid agent and then find the top N_{th} number of agents associated with the largest weight values. The average absolute error between the speed matrices for the current and historical times is computed using Equation (4) to represent the dissimilarity $s_t^{(i)}$ between the current traffic status z_t and each valid agent $x_t^{(i)}$. The small value of dissimilarity represents the data matrices are more similar to each other. Here, a likelihood function which follows a Gaussian distribution $N(\mu, \sigma^2)$ is used to transfer the value of dissimilarity into weight $w_t^{(i)}$ as Equation (5).

$$s_t^{(i)} = |z_t - x_t^{(i)}| / (L \times N_{seg}), i \in \Psi_t \quad (18)$$

$$w_t^{(i)} = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left\{-\frac{(s_t^{(i)} - \mu)^2}{2\sigma^2}\right\}, i \in \Psi_t \quad (19)$$

Thereafter, the value of the weight for each valid agent is sorted in descending order and the top N_{th} number of agents with large weight values are maintained to use in the next iteration.

This process is described in Equation (6), in which the index of preserved agents is denoted by j and $j = 1:N_{th}$. In this way, the agents are divided into two groups. The first group includes the agents with large weights. The second group includes the invalid agents that cannot provide prediction values (exceed data boundary) or the agents with negligible weights. The second group of agents will be re-selected in the next process so that new agents with similar traffic status to the current time interval can be selected.

$$x_t^{(j)} = x_t^{(i)}, w_t^{(j)} = w_t^{(i)}, \text{ when } i = \arg \max_{i \in \Psi_t} w_t^{(i)}, \Psi_t = \Psi_t - \{i\} \quad (20)$$

Re-select Agent

Considering the top N_{th} numbers of valid agents with large weights are maintained but the rest $N-N_{th}$ number of agents are disregarded, a re-selection algorithm is developed here to fill the gap with agents associated with similar traffic patterns to the current traffic speed matrix. Here, each historical day can be selected to represent the new agent. Therefore, the probability to select each historical day is calculated. Firstly, the index of time interval which corresponds to the traffic speed matrix with minimum dissimilarity to the current traffic status is computed as σ_t^n for each historical day n by Equation (7). Thereafter, the dissimilarity between current traffic status z_t and the historical speed matrix $\Omega(n, \sigma_t^n)$ is calculated and then the same likelihood function as Equation (19) is used to obtain the selection probability λ_t^n of historical day n by Equation (8).

$$\sigma_t^n = \arg \min_{k \in [L, H_n - p]} (z_t - \Omega(n, k)), n \in [1: D] \quad (21)$$

$$\lambda_t^n = p_{e_t} (z_t - \Omega(n, \sigma_t^n)), n \in [1: D] \quad (22)$$

After the above calculation, the remaining $N-N_{th}$ number of agents (the index is denoted by $j = N_{th}+1:N$) can be re-selected according to the probability of λ_t^n , which represents the re-selection probability of historical day n under the condition of current time t . Therefore, the corresponding traffic speed matrix and weight can be located according to Equation (9).

$$d_t^{(j)} = \left[n \mid P(n|t) = \lambda_t^n \right], n \in [1: D]$$

$$x_t^{(j)} = \Omega(d_t^{(j)}, \sigma_t^{d_t^{(j)}}), w_t^{(j)} = \lambda_t^{d_t^{(j)}} \quad (23)$$

Travel Time Prediction

Finally, the total N number of agents are located for time interval t . Since each agent corresponds to a historical day with a certain time index, the i^{th} agent can produce a recommendation of travel time $T^E(d_t^{(i)}, j_t^{(i)}+p)$ which departs on time $j_t^{(i)}+p$ at historical day $d_t^{(i)}$. Therefore, the predicted travel time distribution can be obtained by aggregating the recommendations from all agents as Equation (10). And the average predicted value is calculated by the weighted average travel time using Equation (11).

$$T_{t+p}^E = \left\{ T^E(d_t^{(i)}, j_{t+p}^{(i)}), w_t^{(i)} \right\}, i \in [1: N] \quad (24)$$

$$\overline{T_{t+p}^E} = \frac{\sum_{i=1}^N w_t^{(i)} T^E(d_t^{(i)}, j_{t+p}^{(i)})}{\sum_{i=1}^N w_t^{(i)}}, i \in [1:N] \quad (25)$$

Case study

Test Environment

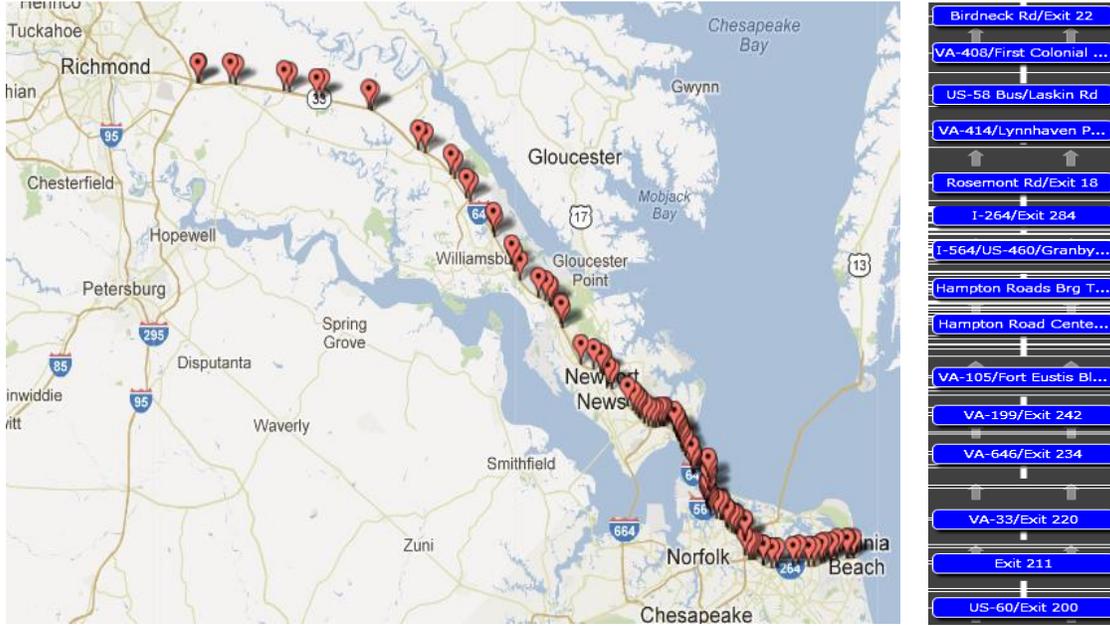


FIGURE 16: The selected freeway stretch from Richmond to Virginia Beach.

A freeway stretch from Richmond to Virginia Beach (95 miles long) connected by I-64 and I-264 is selected as the test site in this study. This test site usually experiences high traffic volumes and serious congestion during the summer season, since Virginia Beach is a famous tourist destination and the selected freeway stretch serves as the main route heading to the beaches. The evaluation of travel time prediction on the test site is conducted based on probe data from INRIX. The data provided by INRIX are mainly collected by GPS-equipped vehicles and supplemented with traditional road sensor data, as well as mobile devices and other sources [32]. The probe data on the test site covers 96 freeway segments with a total length of 95 miles. The average segment length is 0.65 miles long, and the length of each segment is unevenly divided in the raw data from 0.1 to 6.36 miles. The location of the study site and the deployment of segments are presented in FIGURE 16. The raw data provides the average speed for each segment and is collected at one-minute interval. In this study, the raw data are aggregated at five-minute intervals in order to reduce the noise in calculating the travel times [25, 33]. The aggregated traffic data from May 16, 2012 to September 15, 2012 and the corresponding afternoon time periods between 2 p.m. and 8 p.m. for each day are considered in this study to test the algorithm, since most congested periods are observed during this time frame. For each selected day, the instantaneous travel times are calculated for each time interval by assuming the segment speed does not change over time. Given the length of each section of roadway and the corresponding average speed for each time interval, the instantaneous travel time is calculated based on the aggregation of segment travel times at a specific time interval. Conversely, the experienced

travel time is calculated from the ground truth data by considering the change of segment speed over time. In other words the speed profiles are piecewise constant speed values and the trip trajectory is a combination of diagonal curves over time and spaces [1].

Given the fact that the limited summer data (totally 123 days) on the selected freeway stretch are used in this study, the leave-one-out cross-validation method is considered to quantify the prediction accuracy. Leave-one-out cross-validation is a classic model validation technique for assessing how the results of a statistical analysis will generalize to an independent data set. This method has also been used in many application fields including traffic prediction problems [34]. It should be noted that the data of previous two weeks are required for historical average method. Therefore, the leave-one-out testing starts from May 30 to September 15, 2012 (totally 109 days). The testing is conducted by using four different prediction methods for each day and the remaining 122 days serve as the historical dataset. Finally, the average performance over the 109 test days is used to compare the prediction accuracy of different methods. The parameters in the proposed method are pre-defined for the test. The width of the matching window to measure the dissimilarity between historical and real-time traffic status is chosen to be 6 time intervals (30 minutes), which entails the traffic speed matrix along all freeway segments over half an hour being used as an input variable. The number of agents N is selected to be 100. The re-selection threshold N_{th} is 80. The likelihood function to calculate the agent weight factor follows a Gaussian distribution $N(0,2)$. It should be noted that a sensitivity analysis is conducted in the case study to quantify the impacts of various parameters on the algorithm performance.

Comparison Methods and Performance Indices

In order to evaluate the performance of the proposed predictor, three other prediction methods are also tested on the same dataset. The instantaneous travel time method is the easiest alternative to predict future travel times by assuming the current traffic speed along all the segments will remain constant until the completion of the trip as Equation (12). This method is currently used by the Virginia Department of Transportation (VDOT) to display travel time information on variable message signs. Therefore, instantaneous travel times are considered as the state-of-practice method and used to quantify the tradeoff between simplicity and prediction accuracy.

$$T_{t+p}^E = T_t^I \quad (26)$$

Historical average data is a common indicator for recurrent traffic conditions and can also be used for travel time prediction. A previous study demonstrates that simple historical average of long term periods (e.g. one year) has large variations, and is not a good predictor for experienced travel times [22]. Therefore, the historical average from the past two weeks is used as an alternative. Since the current study only includes the summer season dataset from May 16 to September 15, 2012 and the variations of travel times between different seasons are non-existent, the historical average is anticipated to provide better prediction accuracy when compared to instantaneous methods. It should be noted that the traffic pattern on Monday and Friday are usually different from other weekdays [35], therefore we divide daily traffic data into four groups - Monday, Friday, weekday (Tuesday, Wednesday and Thursday), weekend (Saturday and Sunday). The average experienced travel times at the same time interval from the same group of days in the previous two weeks, which is denoted by \bar{T}_t , are used as the historical average to predict travel times on the current day using Equation (13).

$$T_{t+p}^E = \bar{T}_{t+p}^E \quad (27)$$

The k -NN is a widely used state-of-art method for real-time travel time prediction problems [13-15]. In order to conduct an objective comparison between k -NN and the proposed approach, the same spatial-temporal traffic speed matrix $u_{N_{seg} \times L}$ from short past to current time along all the freeway segments is used for k -NN. Thereafter, m numbers of similar data matrix with tail time $\{h_1, h_2, \dots, h_m\}$ can be selected as the candidates from the historical dataset Ω as given in Equation (11). For each candidate with index i , a weight w_i is calculated using the normalized matching error. Moreover, the corresponding experienced travel time departure at h_{i+p} on the selected candidate day i can be obtained from the historical data set. Consequently, the experienced travel time T_{t+p}^E on the current day can be predicted as the weighted average of the travel times from all candidates using Equation (12).

$$\begin{aligned} H_c &= \{h_1, h_2, \dots, h_m\} \\ \text{For } i &= 1:m \\ h_i &= \arg \min_{h \in \Omega} |z_t - h|, \Omega = \Omega - h_i \\ \text{End For} \end{aligned} \quad (28)$$

$$\begin{aligned} w_i &= |z_t - h_i| / \sum_{i=1}^m |z_t - h_i| \\ T_{t+p}^E &= \sum_{i=1}^m T_{h_i+p}^E \cdot w_i \end{aligned} \quad (29)$$

Different combinations of parameters were tested and the optimum parameters were selected as $L = 6$ and $m = 20$, which corresponds to the least prediction error [14]. These parameters are used to test the k -NN method using the same data set and to serve as a comparison with other methods. It should be noted that the k -NN method is different from the proposed approach even if the number of candidates in the k -NN method is equal to the number of particles. The reason lies in the fact that there is no data propagation process in the k -NN algorithm, and all candidates are blindly selected from each time interval based on its similarity measure (shortest Euclidean distance) to the current travel time sequence.

Both relative and absolute prediction errors are used to evaluate the performance of predictors. The absolute error is denoted by the mean absolute error (MAE) using Equation (13), which represents the average absolute deviations between the predicted and the ground truth values. The corresponding relative error is represented by the mean absolute percentage error (MAPE) of Equation (14), which denotes the absolute proportional deviations between the predicted and the ground truth values.

$$MAE = \frac{1}{I' J} \mathring{\mathbf{a}} \mathring{\mathbf{a}} \left| y_i^j - \mathring{y}_i^j \right| \quad (30)$$

$$MAPE = \frac{100}{I' J} \sum_{j=1}^J \sum_{i=1}^I \frac{|y_i^j - \hat{y}_i^j|}{y_i^j} \quad (31)$$

where J is the total number of days in the testing dataset (109 days in our case study); I is the total number of time intervals in one day (i.e., 72 intervals occurring every five minutes between 2 p.m. and 8 p.m.); and y_i^j and \hat{y}_i^j denote the ground truth and the predicted value, of the experienced travel time for the i^{th} time interval on the j^{th} day in the testing dataset.

Test Results

TABLE 6: Prediction Results by Different Methods

		Prediction Horizon (min)						
		0	10	20	30	40	50	60
Historical Average	MAE (min)	13.01	13.01	13.01	13.01	13.01	13.01	13.01
	MAPE (%)	11.46	11.46	11.46	11.46	11.46	11.46	11.46
Instantaneous	MAE (min)	11.52	13.06	14.40	15.78	17.10	18.28	19.38
	MAPE (%)	10.64	12.12	13.47	14.85	16.12	17.29	18.31
k -NN	MAE (min)	10.48	11.12	12.10	12.84	13.62	14.24	15.06
	MAPE (%)	9.24	9.95	10.68	11.31	11.98	12.61	13.18
ABM	MAE (min)	7.69	7.92	8.14	8.33	8.62	8.97	9.49
	MAPE (%)	6.75	6.98	7.21	7.53	7.86	8.18	8.57

The prediction results for four methods are summarized in Table 5. The least prediction errors are located at the bottom of the table using the proposed agent-based method (ABM). The relative absolute errors for the four predictors are presented in FIGURE 17. The figure clearly demonstrates a significant degradation in the prediction accuracy for longer prediction horizons for the instantaneous and historical average methods. By comparing the historical average method with the k -NN method, the latter method outperforms the former for short-term prediction (prediction horizons from 0 to 30 minutes) and historical average method outperforms the k -NN method for longer prediction horizons ranging from 40 to 60 minutes. Moreover, the MAPE by the k -NN method increases from 9.24% to 13.18% (43% increase), which is higher than the error associated with the proposed method from 6.75% to 8.57% (25% increase). More importantly, the relative errors produced by the proposed ABM approach is always below 9% for the different prediction horizons within one hour, which indicates the prediction performance of the proposed method is much more reliable compared to the other three methods. Based on the observations of daily prediction results, the historical average and instantaneous methods produce large variations in performance compared to the k -NN and ABM methods. The historical average method can accurately predict travel times in recurrent days, but the performances for non-recurrent days are very low. The instantaneous method works well for uncongested days, but produces large errors for congested days. Comparatively, both of k -NN and ABM methods generate reliable performance on different days, however the ABM method outperforms the k -NN method especially for long prediction horizons (e.g. 60 minutes).

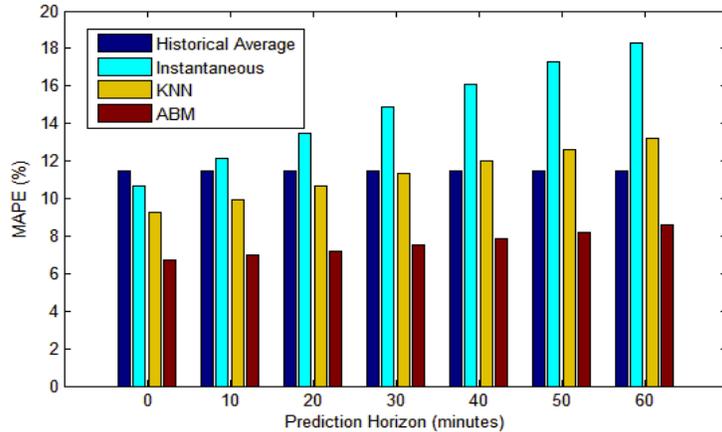


FIGURE 17: MAPEs using four predictors under various prediction horizons.

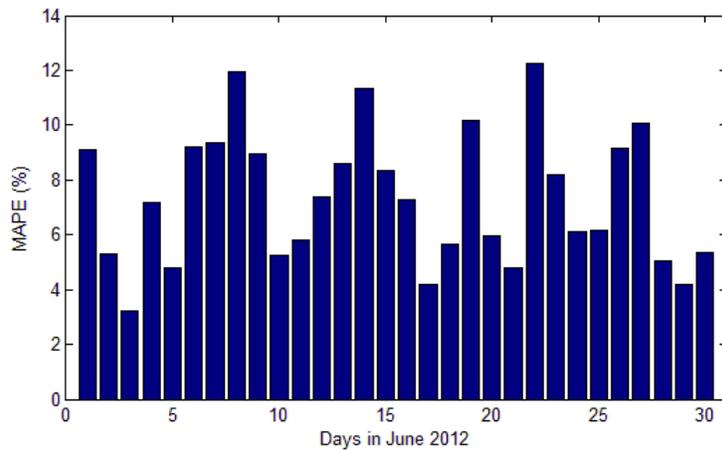


FIGURE 18: Daily prediction errors by ABM on June 2012.

The daily average prediction errors by the ABM method on June 2012 are presented in **FIGURE 18**. Generally, no obvious trends are observed in the 30 days-worth of results. Specifically, occasionally the algorithm performs better on weekdays and occasionally it performs better on weekends. Maybe this is caused by the fact that special conditions such as inclement weather or incidents are not filtered from our dataset. In order to further analyze the prediction errors for different methods, the predicted travel time curves for the four methods are compared with the ground truth data for two sample weekdays, as illustrated in Figure 11. Both the instantaneous and the k -NN method experience some time lag relative to the ground truth data, especially during the onset and dissipation of congestion. Specifically, the instantaneous method highly underestimates the travel time when congestion is forming, and overestimates the travel time when congestion is dissipating. It should be noted that the historical average method overestimates the congestion for June 21 between 3 and 5 p.m., and underestimates the ground truth data when congestion is dissipating. According to ground truth data on June 21, the congestion forms between 2 and 4 p.m. and dissipates between 4 and 8 p.m., and the corresponding average MAPEs by the ABM are 3.8% and 5.6%, respectively. On the other hand, the prediction errors using the historical average method are even worse by highly

overestimating the travel times on June 29. However, the proposed method improves the prediction performance by producing 3.3% and 4.5% of errors during congestion forming (2 to 3:30 p.m.) and dissipation (3:30 to 8 p.m.) periods. Moreover, the proposed method provides confidence intervals of predicted travel times. The 5th and 95th percentile of the predicted travel times are selected as the upper and lower boundaries in Figure 11. The gray shadow area between the boundaries covers most of the ground truth data temporal variation, which demonstrates that the proposed approach provides good accuracy in predicting travel time confidence intervals. There is a trade-off between the width of the confidence intervals and the accuracy of prediction results. Consequently, the use of a narrow confidence interval does not represent higher prediction accuracy, since the ground truth curve has less chance to be included within the confidence interval. More tests of different upper and bottom percentiles are needed to find the optimum choice of confidence intervals with narrow coverage but also providing dependable predictions to cover most of the ground truth curve.

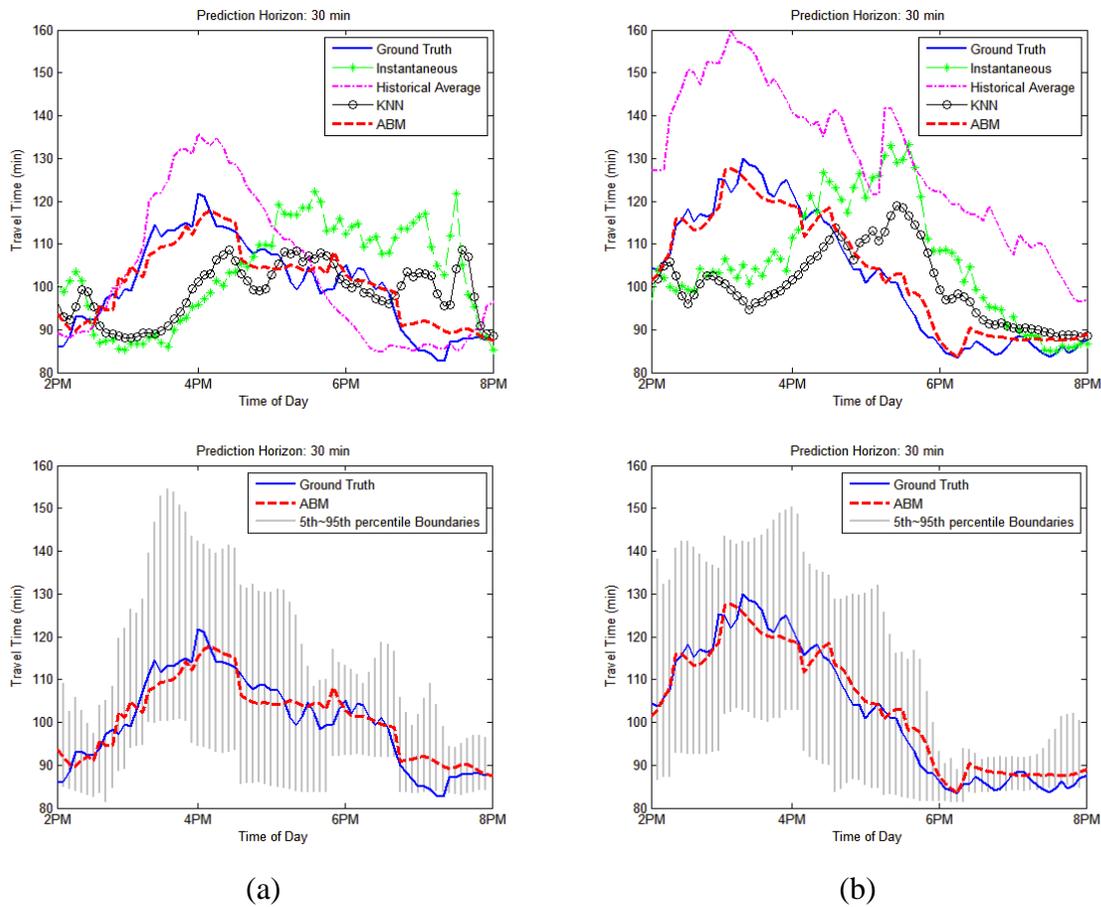


FIGURE 19: Travel time prediction results by four methods and the confidence intervals by ABM on (a) June 21, 2012 (Thursday); (b) June 29, 2012 (Friday).

During the case study tests, the days with and without traffic incidents are mixed together in both the historical and test datasets. In order to demonstrate the performance of predictors when incidents occur, a special day with an incident is selected, and the travel time prediction results using the four methods are illustrated in **FIGURE 20**. An incident occurs at the location directly

upstream of the Hampton Roads Bridge Tunnel between 6:30 and 7:30 p.m. on June 11, 2012. The proposed ABM method outperforms the other three methods by producing the least errors to ground truth data as shown in **FIGURE 20** (a). When the incident occurs, the multi-step predicted travel times from prediction horizon 0 to 60 minutes using the four methods are presented in **FIGURE 20** (b). The traffic congestion builds up quickly at 6:30 p.m., thus the ground truth travel time also increases fast. Apart from the ABM method, all the other methods cannot capture the growing trend and the predicted travel times deviate from the ground truth data. Although there is a slight delay, the ABM method can still predict the congestion-forming trend caused by the incident and thus it produces much higher prediction accuracy than other methods.

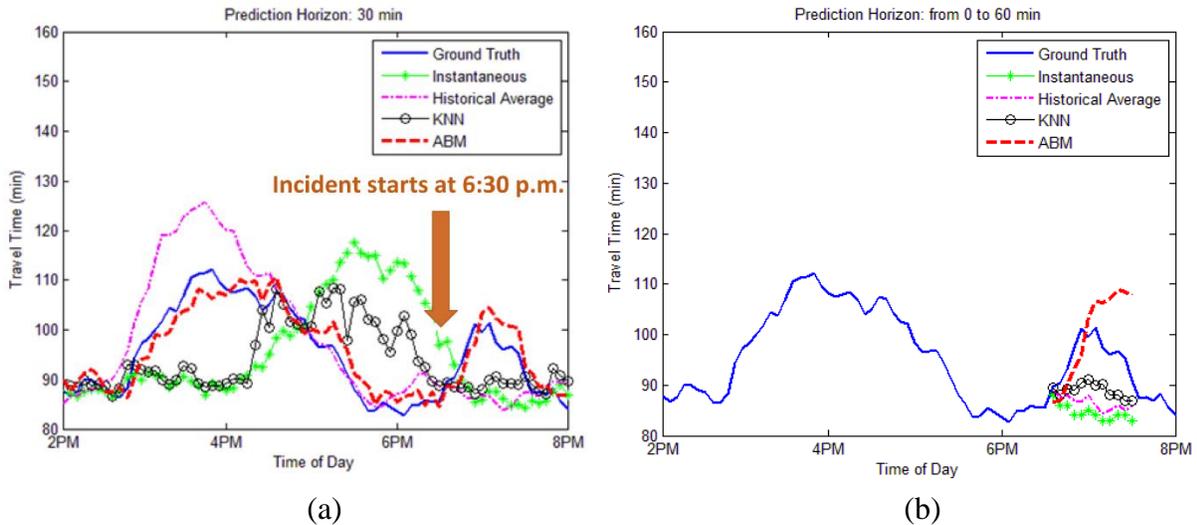


FIGURE 20: Illustration of travel time prediction when incident occurs on June 11, 2012; (a) Prediction results by four methods; (b) Prediction up to 60 minutes by four methods.

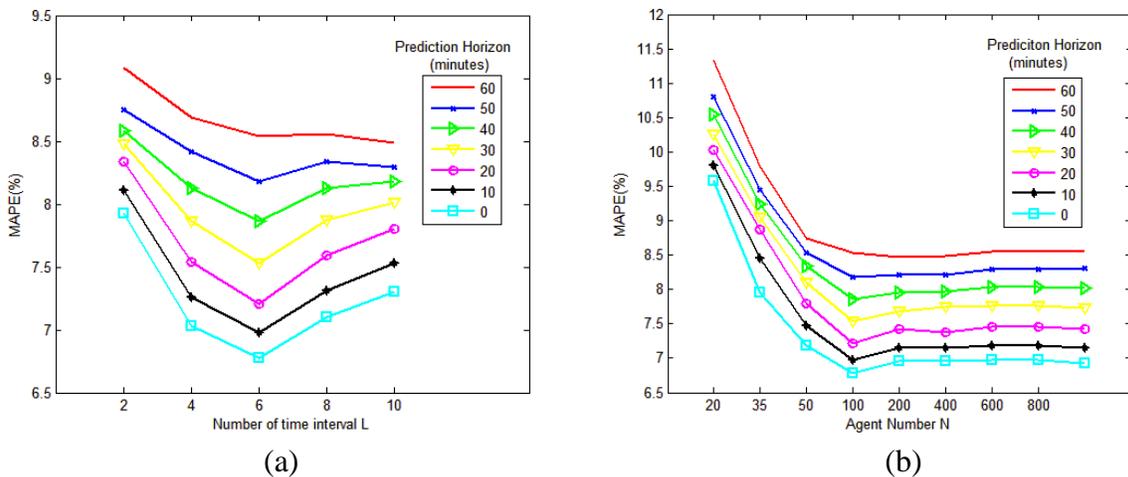


FIGURE 21: Sensitivity analysis.

Sensitivity Analysis

A sensitivity analysis is conducted to quantify the impact of parameter L and agent number N on the prediction accuracy of the proposed agent-based method. The parameter L determines the

matching window width in the agent propagation and re-selection processes, so a larger value of L results in a wider matching window and vice versa. Here, the agent number maintains the same value of 100 as in the previous testing and the values of L vary from 2 to 10 time intervals to calculate the average MAPE for different prediction horizons, as presented on **FIGURE 21** (a). Although the least MAPE for 60 minutes prediction horizon corresponds to the L value of 10, the optimum L value is selected by considering the best performance for prediction horizon from 0 to 60 minutes. The minimum errors for prediction horizon from 0 to 50 can be obtained when L equal to 6, so 6 time intervals is selected as the optimum value of L . Consequently, the best matching window is a half-hour of traffic speed data along all the freeway segments. Moreover, under the condition that L value is 6, different agent numbers are also investigated to calculate the relative errors as shown on **FIGURE 21** (b). Generally, the prediction error decreases with larger agent number. However, the error reaches the minimum value when agent number is 100, and then prediction error slightly increases with agent number greater than 100. Therefore, the agent number of 100 is the best choice to predict travel times in the case study. The same analysis can be conducted on different sites or roadway compositions to find the optimum model parameters.

Computation Speed

In order to investigate the potential to use the proposed method for real-time application, the computation speed of the algorithm on the cast study needs to be calculated. The testing of the ABM travel time predictor was performed on a personal computer with Intel dual core CPU, 2.40 GHz and 4GB of random-access memory within the MATLAB 2012b environment. It should be noted that the procedure of re-selecting agents in ABM method is very time consuming since many iterations are needed to calculate the probabilities of selecting each historical day. Considering matrix computation is very fast in MATLAB software, the iterations in Equation (21) to compare the dissimilarity between current traffic pattern and historical traffic pattern on each time interval was coded by matrix calculation to expedite computation speed. Under the scenario of setting $L = 6$, $N_{th} = 80$ and $N = 100$, the total average computation time for one day between 2 and 8 p.m. was 6.20 seconds. Totally 72 time intervals are included in this time period, so the calculation of a single prediction by every 5-minute only requires 0.086 seconds. Therefore, the computational efficiency allows the agent-based model approach to be implemented in real-time in Traffic Management Center (TMC).

Conclusions

In this paper, an agent-based modeling approach is proposed to conduct multi-step experienced travel time predictions. Although agent-based models have been widely used in various transportation problems, the algorithm developed in this paper is the first attempt to use the concept of agent-based modeling to predict travel times. Similar to the traditional expert decision systems, each agent represents an expert and can provide a recommendation of future experienced travel time. At the same time, a set of agent interaction rules are developed to update agent for future time intervals and also calculate the weight of each agent, which indicate the confidence level of the predicted travel time value. In this way, the average predicted travel time and the corresponding confidence boundaries can be calculated as the output of the algorithm. A 95-mile freeway stretch from Richmond to Virginia Beach along I-64 and I-264 is used to test the performance of the proposed method. The test results indicate that the proposed ABM method produces the least absolute and relative errors to predict experienced travel times when

compared to instantaneous travel time, the historical average and k -NN methods, and maintains less than a 9% prediction error for trip departures up to 60 minutes later. Moreover, the confidence boundaries of the predicted travel times indicate that the proposed approach also provides high accuracy in predicting travel time confidence intervals. In addition, sensitivity analysis is conducted to investigate the impact of different model parameters to the prediction accuracy. Lastly, the computation time of the proposed algorithm is tested and the result demonstrates the ABM method can provide accurate and efficient travel time prediction for real-time applications.

Although probe data from the private sector are used in the case study, the proposed method is data source independent as long as the spatiotemporal speed measurements are available. For instance, loop detector data, Bluetooth or cell phone data can also be used in the proposed method. Considering the proposed predictor provides more than 90% accuracy in predicting travel times with departures up to 60 minutes into the future, the proposed agent-based prediction algorithm can be extended to make a recommendation on the optimum departure time in addition to providing the expected travel time. In the future, we could also categorize historical days by different conditions such as weather, incident, holiday/special event, etc. So that the algorithm will use the sub-dataset with the same category to the test day to improve prediction accuracy.

Acknowledgements

This research effort was funded partially by Virginia Department of Transportation (VDOT) and partially by the Mid-Atlantic Universities Transportation Center (MAUTC). The authors also acknowledge Catherine McGhee, Ralph L. Jones and Philomena Lockwood for their assistance and feedback.

References

- [1] J. W. C. van Lint and N. J. van der Zijpp, "Improving A Travel Time Estimation Algorithm by Using Dual Loop Detectors," *Transportation Research Record: Journal of the Transportation Research Board*, vol. 1855, pp. 41-48, 2003.
- [2] H. Tu, "Monitoring Travel Time Reliability on Freeways," Ph.D., Department of Transport and Planning, Technische Universiteit Delft, 2008.
- [3] P.-E. Mazaré, O.-P. Tossavainen, A. Bayen, and D. B. Work, "Trade-offs between Inductive Loops and GPS Vehicles for Travel Time Estimation: A Mobile Century Case Study," in *Transportation Research Board 91st Annual Meeting*, Washington, D.C., 2012.
- [4] H. Chen, H. A. Rakha, S. A. Sadek, and B. J. Katz, "A Particle Filter Approach for Real-time Freeway Traffic State Prediction," in *Transportation Research Board 91st Annual Meeting*, Washington D.C., 2012.
- [5] J. Rice and E. Van Zwet, "A Simple and Effective Method for Predicting Travel Times on Freeways," *IEEE Transactions on Intelligent Transportation Systems*, vol. 5, pp. 200-207, 2004.
- [6] X. Zhang and J. A. Rice, "Short-term Travel Time Prediction," *Transportation Research Part C: Emerging Technologies*, vol. 11, pp. 187-210, 2003.
- [7] C. Nanthawichit, T. Nakatsuji, and H. Suzuki, "Application of Probe-vehicle Data for Real-time Traffic-state Estimation and Short-term Travel-time Prediction on a Freeway," *Transportation Research Record: Journal of the Transportation Research Board*, pp. 49-59, 2003.

- [8] S. I.-J. Chien and C. M. Kuchipudi, "Dynamic Travel Time Prediction with Real-Time and Historic Data " *Transportation Research Record: Journal of the Transportation Research Board*, vol. 129, pp. 608-616, 2003.
- [9] M. Chen and S. I. J. Chien, "Dynamic Freeway Travel-Time Prediction with Probe Vehicle Data: Link Based Versus Path Based," *Transportation Research Record: Journal of the Transportation Research Board*, vol. 1768, pp. 157-161, 2001.
- [10] J. Xia, M. Chen, and W. Huang, "A Multistep Corridor Travel-Time Prediction Method Using Presence-Type Vehicle Detector Data," *Journal of Intelligent Transportation Systems: Technology, Planning, and Operations*, vol. 15, pp. 104-113, 2011.
- [11] A. Guin, "Travel Time Prediction Using a Seasonal Autoregressive Integrated Moving Average Time Series Model," in *IEEE Conference on Intelligent Transportation Systems*, 2006, pp. 493-498.
- [12] D. Billings and J.-S. Yang, "Application of the ARIMA Models to Urban Roadway Travel Time Prediction - A Case Study," in *IEEE Conference on Systems, Man and Cybernetics*, 2006, pp. 2529-2534.
- [13] J. Myung, D.-K. Kim, S.-Y. Kho, and C.-H. Park, "Travel Time Prediction Using k Nearest Neighbor Method with Combined Data from Vehicle Detector System and Automatic Toll Collection System," *Transportation Research Record: Journal of the Transportation Research Board*, vol. 2256, pp. 51-59, 2011.
- [14] W. Qiao, A. Haghani, and M. Hamed, "Short Term Travel Time Prediction Considering the Weather Impact," in *Transportation Research Board 91st Annual Meeting*, Washington D.C., 2012.
- [15] B. I. Bustillos and Y.-C. Chiu, "Real-Time Freeway-Experienced Travel Time Prediction Using N-Curve and k Nearest Neighbor Methods," *Transportation Research Record: Journal of the Transportation Research Board*, vol. 2243, pp. 127-137, 2011.
- [16] J. W. C. v. Lint, S. P. Hoogendoorn, and H. J. v. Zuylen, "Accurate Freeway Travel Time Prediction with State-space Neural Networks Under Missing Data," *Transportation Research Part C: Emerging Technologies*, vol. 13, pp. 347-369, 2005.
- [17] D. Park and L. R. Rilett, "Forecasting Multiple-period Freeway Link Travel Times Using Modular Neural Networks " *Transportation Research Record: Journal of the Transportation Research Board*, vol. 1617, pp. 163-170, 1998.
- [18] J. W. C. van Lint, "Reliable real-time framework for short-term freeway travel time prediction," *Journal of transportation engineering*, vol. 132, pp. 921-932, 2006.
- [19] C.-H. Wu, J.-M. Ho, and D. T. Lee, "Travel-time Prediction with Support Vector Regression," *IEEE Transactions on Intelligent Transportation Systems*, vol. 5, pp. 276-281, 2004.
- [20] L. Vanajakshi and L. R. Rilett, "Support Vector Machine Technique for the Short Term Prediction of Travel Time," in *IEEE Proceedings of Intelligent Vehicles Symposium*, 2007, pp. 600-605.
- [21] H. Chen and H. A. Rakha, "Forecasting Freeway Dynamic Travel Times by Constructing Trip Trajectories," in *Transportation Research Board 92nd Annual Meeting* Washington D.C., 2013.
- [22] M. Yildirimoglu and N. Geroliminis, "Experienced Travel Time Prediction for Congested Freeways," *Transportation Research Part B: Methodological*, vol. 53, pp. 45-63, 2013.
- [23] A. G. Parlos, O. T. Rais, and A. F. Atiya, "Multi-step-ahead Prediction using Dynamic Recurrent Neural Networks," *Neural Networks*, vol. 13, pp. 765-786, 2000.

- [24] R. Boné and M. Crucianu, "Multi-step-ahead Prediction with Neural Networks: A Review," *9emes rencontres internationales: Approches Connexionnistes en Sciences*, vol. 2, pp. 97-106, 2002.
- [25] X. Zeng and Y. Zhang, "Development of Recurrent Neural Network Considering Temporal - Spatial Input Dynamics for Freeway Travel Time Modeling," *Computer - Aided Civil and Infrastructure Engineering*, vol. 28, pp. 359-371, 2013.
- [26] T. Bui and J. Lee, "An Agent-based Framework for Building Decision Support Systems," *Decision Support Systems*, vol. 25, pp. 225-237, 1999.
- [27] L. Zhang and D. Levinson, "Agent-based Approach to Travel Demand Modeling: Exploratory Analysis," *Transportation Research Record: Journal of the Transportation Research Board*, vol. 1898, pp. 28-36, 2004.
- [28] Y. Luo, K. Liu, and D. N. Davis, "A Multi-agent Decision Support System for Stock Trading," *IEEE Network*, vol. 16, pp. 20-27, 2002.
- [29] E. Shafiei, H. Thorkelsson, E. I. Ásgeirsson, B. Davidsdottir, M. Raberto, and H. Stefansson, "An Agent-based Modeling Approach to Predict the Evolution of Market Share of Electric Vehicles: A Case Study from Iceland," *Technological Forecasting and Social Change*, vol. 79, pp. 1638-1653, 2012.
- [30] T. Sueyoshi and G. R. Tadiparthi, "An Agent-based Decision Support System for Wholesale Electricity Market," *Decision Support Systems*, vol. 44, pp. 425-446, 2008.
- [31] S. Hassan, J. Arroyo, J. M. Galán, and J. Pavón, "Asking the Oracle: Introducing Forecasting Principles into Agent-Based Modelling," *Journal of Artificial Societies and Social Simulation*, vol. 16, p. 13, 2013.
- [32] INRIX. (2012). <http://www.inrix.com/trafficinformation.asp>. Available: <http://www.inrix.com/trafficinformation.asp>
- [33] D. Park, L. R. Rilett, B. J. Gajewski, C. H. Spiegelman, and C. Choi, "Identifying Optimal Data Aggregation Interval Sizes for Link and Corridor Travel Time Estimation and Forecasting," *Transportation*, vol. 36, pp. 77-95, 2009.
- [34] J. Kwon, B. Coifman, and P. Bickel, "Day-to-day Travel-time Trends and Travel-time Prediction from Loop-detector Data," *Transportation Research Record: Journal of the Transportation Research Board*, vol. 1717, pp. 120-129, 2000.
- [35] H. A. Rakha and M. Van Aerde, "Statistical Analysis of Day-to-day Variations in Real-time Traffic Flow Data," *Transportation Research Record: Journal of the Transportation Research Board*, vol. 1510, pp. 26-34, 1995.

APPENDIX D

Predicting Freeway Travel Times using Dynamic Template Matching

This article is currently under review and may be cited as a working version: Chen, H. and Rakha, H.A., Predicting Freeway Travel Times using Dynamic Template Matching. Working paper.

INTRODUCTION

Congestion has proven to be a serious problem across urban areas in the United States. In 2007, it cost highway users 4.2 billion extra hours of sitting in traffic and an extra 2.8 billion gallons of fuel. This all translated into an additional \$87.2 billion in congestion costs for road users in 2007, which showed a 50% increase in cost compared to data from the previous decade. Even though the recent economic downturn is said to have marginally eased the congestion problem nationwide, new evidence shows an uptrend in traffic and consequently congestion [1].

Travel-time information is an essential part of Advanced Traveler Information Systems (ATISs) and Advanced Traffic Management Systems (ATMSs). A key component of these systems is the prediction of travel times. From the perspective of travelers such information may assist in making better route choice and departure time decisions. For transportation agencies these data provide criteria with which to better manage and control traffic to reduce congestion. Tackling congestion (both recurrent and non-recurrent) has proven to be a challenge for highway agencies. Adding capacity in response to congestion is becoming less of an option for these agencies due to a combination of financial, environmental, and social issues. Therefore, the main focus has been on improving the performance of existing facilities through continuous monitoring and dissemination of traffic information. The minimum that can be accomplished is to inform the public or, specifically, the potential users of what they should expect on the roadways before and during their trips. Additionally, this information can be applied to provide alternatives to users so that they may make informed decisions about their trips. This is the essence of Advanced Traveler Information System (ATIS) applications such as 511 that have been implemented nationwide. In many states relevant traffic information is also posted on variable message signs (VMSs) that are strategically positioned along highways. Consequently, there is a need to provide predicted travel times to road users for better planning their trips and choosing their route of travel, further reducing congestion.

Various traffic sensing technologies have been used to collect traffic data for use in computing travel times, including point to point travel time collection (license plate recognition systems, automatic vehicle identification systems, mobile, Bluetooth, probe vehicle, etc.) and station based traffic state measuring devices (loop detector, video camera, remote traffic microwave sensor, etc.). Private companies, such as INRIX, integrate different sources of measured data to provide section-based traffic state data (speed, average travel time), which is used in our study to develop algorithms for predicting travel times. The benefit of using section-based traffic state data is that travel time can be easily calculated from traffic state data. More importantly, the section-based data provides the flexibility for scalable applications on traffic networks.

By providing section-based traffic state data, there are two approaches to compute travel time depending on the trip experience [2, 3]. Experienced travel time is the actual, realized travel time that a vehicle could experience during a trip. If a vehicle leaves its origin at the current time, the roadway speed will not only change across space but also across time during the entire trip. Consequently, experienced travel time can be obtained by using a prediction algorithm to compute the speed evolution in future time steps. Instantaneous travel time is the other approach available to compute travel times without the consideration of speed evolution across time. It is usually computed using the current speed along the entire roadway; in other words the speed field is assumed to remain constant in time. The instantaneous travel time is close to the experienced travel time when the roadway speed does not change significantly across time space during the trip. However, this approach may deviate substantially from the actual, experienced

travel time under transient states during which congestion is forming or dissipating during a trip [4].

Some attempts have been conducted using macroscopic traffic modeling to predict short-term traffic states, however the accuracy degrades rapidly with the increase in the prediction time span [5, 6]. It should be noted that traffic state in the near future usually cannot provide enough information to cover the entire trip, especially for long trips. For instance, in the case of a 100-mile trip, departures at the current time would still be traveling one hour in the future even under free-flow traffic conditions. For this case, the traffic state for the following one hour or more should be predicted in order to compute experienced travel times. An alternative to solving this problem is to use historical data. The historical dataset provides a pool of past experienced traffic patterns which can be used to predict future traffic states. The key issue is determining the similar historical traffic patterns to match with the changeable real-time traffic information.

This study develops a dynamic template matching method to predict experienced travel times over multiple prediction horizons. Instead of using a fixed template size, as is done in other studies, the template size is dynamically updated each time interval based on the spatiotemporal shape of the congestion formed upstream of the bottleneck. In addition, a Fast Fourier Transform (FFT) is used to reduce the computation cost in the template matching process. The selected historical candidates that are similar to current conditions are used to predict the experienced travel times. A freeway stretch on I-64 is selected to test the proposed algorithm using five-minute aggregated traffic data provided by INRIX. The travel time prediction results demonstrate that the proposed method produces higher prediction accuracies compared to instantaneous travel times and fixed template matching methods.

The remainder of this paper is organized as follows. A literature review of previous travel time prediction methods is provided. Subsequently, the proposed methodology of using current and historical traffic status to predict experienced travel times is presented. This is followed by a description of the test data for the case study and the comparison results of using proposed approach for prediction. The last section provides the summary conclusions of this study and some research recommendations for future research.

LITERATURE REVIEW

During the past decades, many studies have been conducted to predict travel times. Some of the reviews of different methods can be found in earlier publications [7-10]. According to the manner of modeling, those methods can be classified into time series models including Kalman filter [11, 12], Auto-Regressive Integrated Moving Average (ARIMA) models [12-14] and data-driven methods, such as artificial neural networks [9, 15-17], support vector regression (SVR) [18, 19] and K-Nearest Neighbor (k -NN) [8, 20, 21] models. These techniques are implemented through direct and indirect procedures to predict travel times using different types of state variables. Travel time is directly used as the state variable in model-based or data-driven methods to predict travel times. Indirect procedures are performed by using other variables (such as traffic speed, density, flow, occupancy, etc.) as the state variable to predict traffic status, and then future travel time can be calculated based on the transition to predicted traffic status.

Time series models construct the time series relationship of travel time or traffic state, and then current and/or past traffic data are used in the constructed models to predict travel times in the next time interval [22]. A Kalman Filter (KF) is a popular method for data estimation and tracking, in which time update and measurement update processes are included. A time series equation is used to predict state variables and then state values are corrected according to the new measurement data. The main advantage of a KF is that the recursive framework ensures

traffic data is efficiently updated only using data from previous states and not the entire history [5]. Kalman filters were proposed to predict travel times using Global Positioning System (GPS) information and probe vehicle data [12, 23]. The state transient parameter in the time series equation is defined from average historical data to calculate future travel times. The similar idea was used in the Bayesian dynamic linear model for real-time short-term travel time prediction [11]. The system noise can be adjusted for unforeseen events (incidents, accidents or bad weather) and integrated into the recursive Bayesian filter framework to quantify random variations on travel times. The experiment results based on loop detector data from a segment of I-66 demonstrates the proposed method produces higher prediction accuracy under both recurrent and non-recurrent traffic conditions. However, in these methods a problem exists in that the travel time in the previous time interval is needed to calculate the future travel time. For real-time applications, the travel time is usually greater than the time interval step size. Hence, the actual travel time from the previous time interval is not available to apply in the algorithms used to predict travel times for the next time interval.

A seasonal ARIMA model was proposed to quantify the seasonal recurrent pattern of traffic conditions (occupancy) [13, 14]. Moreover, an embedded adaptive Kalman filter was developed in order to update the occupancy estimate in real-time using new traffic volume measurements. Consequently, multi-step look ahead occupancy information are estimated to obtain a data matrix representing the temporal-spatial traffic condition for the future trip. Since travel time cannot be directly computed through traffic conditions (occupancy), future traffic speed can be calculated using occupancy data by assuming an average vehicle length and using a constant conversion factor known as the *g*-factor in the literature. Consequently, dynamic freeway corridor travel times are predicted with the consideration of traffic state evolution along the corridor. However, this approach may be difficult to implement since the described recurrent pattern of traffic conditions may not be found everywhere.

Data-driven methods usually predict travel times using a large amount of historical traffic data. Time series models are not specified in the data-driven methods, considering the complex stochasticity of traffic systems. Neural networks can be trained from historical data to discover hidden dependencies which can be used for predicting future states. A space neural network (SSNN) method was proposed to predict freeway travel times for missing data [9]. The missing data problem was tackled by simple imputation schemes, such as exponential forecasts and spatial interpolation. Travel time was the direct state variable used for the training process and the experiment results demonstrated the SSNN methods produced accurate travel time predictions on inductive loop detector data. Supported vector machine (SVM) is a successor to ANNs, which has greater generalization ability and is superior to the empirical risk minimization principle as adopted in ANNs [19]. The application of SVM to time series forecasting is called SVR. The SVR predictor was demonstrated to perform well for travel time prediction. The point to point travel time is usually used as the input to ANNs and SVRs. However, both methods require long training processes and are nontransferable to other sites [8].

The *k*-NN method can be used to find several candidate sequences from historical data, by matching with current to short past data sequences. Travel time and occupancy sequences were used to predict travel times using the *k*-NN method with combined data from vehicle detectors and automatic toll collection systems [8]. The occupancy was used since travel time sequence was collected for the recent past time intervals. The results from the case study demonstrated the improvement of prediction accuracy by combining two types of sequences for the matching process. Moreover, a *k*-NN method was proposed by selecting candidates through

the Euclidean distance and data trend measures to predict freeway travel times under different weather conditions [20]. Unlike ANNs and SVRs, *k*-NN methods are easy to implement at different sites without data training required. However, the prediction accuracy needs to be further improved and the computation cost is huge for a large historical dataset.

In summary, existing methods are either insufficient or have limitations for the real time approach of predicting experienced travel times multi-step into the future. Comparing to the previous methods, the proposed dynamic template matching approach fully explores the correlation between real time and historical traffic patterns, and produces efficiently and accurately multi-step prediction of travel times for online application. Instead of using travel time sequence as input in previous studies, the proposed method uses temporal-spatial traffic data given that more dynamic traffic information are included. More importantly, the traffic flow theories – congestion identification and shockwave propagation are used in the proposed algorithm to improve the performance of template matching. Based on the proposed framework, more advanced pattern matching algorithms can be used to further improve the efficiency or accuracy of travel time prediction.

METHODOLOGY

The Travel Time Prediction Framework

The proposed algorithm comprises three stages: identify current traffic status, match similar traffic patterns from historical data, and predict travel times. The framework for the three stages is demonstrated in Figure 22. The current traffic status is initially selected to represent the traffic status of all freeway sections from short-past to the current time interval. The traffic status in this case is a matrix across temporal and spatial axes. Thereafter, the historical traffic speed data with the same dimension to current traffic status is selected as a candidate. Based on the similarity to the current speed matrix, several candidates are extracted to represent the historical recurrent traffic patterns that are similar to the current status. Finally, the subsequence experienced travel times of those candidates are aggregated to represent the travel time distributions in the future.

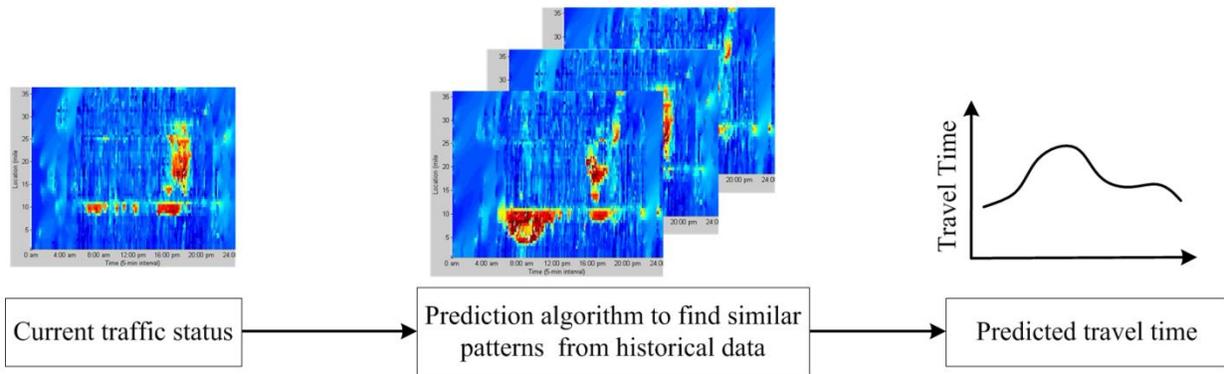


Figure 22: Framework of Proposed Travel Time Prediction Algorithm.

In order to accurately match traffic patterns between real time and historical data, one of the most frequently used pattern recognition techniques - template matching is revised and applied in this paper. Template matching is usually used in computer vision problems for finding small parts of an image (matrix) which match a template image matrix [24]. Comparing to other pattern recognition techniques, for instance image feature selection [25, 26] and artificial neural

networks [9, 15-17], template matching is a simple but powerful algorithm and very suitable to deal with online pattern recognition problems since the offline training process is not needed.

Suppose T is the current day and C denotes the current time interval. Considering the application of matching traffic patterns in this paper, the current traffic status $x(C,T)$ in the first stage of the proposed framework is the template matrix for time period $[T-L+1, T-L+2, \dots, T]$, and the matrix $x(h_i)$ representing the spatial-temporal traffic status on each historical day h_i is the image to be matched with template. Here, L represents the width of template across time intervals. Apparently the width of historical matrix $x(h_i)$ is greater than template of real time traffic status $x(C,T)$, given that the template only includes the traffic data in a short time period and the historical matrix covers the entire time periods on that day. If t is a selected time index in historical day h_i , a matrix $x(h_i, t)$ for time period $[t-L+1, t-L+2, \dots, t]$ can be matched with template. It should be noted that the traffic status for each time interval is a vector that covers the entire roadway segments (totally N segments), therefore the template is a matrix with dimension N by L . Given that the value of N is constant for a selected roadway stretch, the question is how to define the value of L in order to produce the best template matching result.

Different from the previous studies to match traffic pattern by a fixed window width, a dynamic template, which is updated in real time according to the identified congestion and bottleneck shape, is proposed in this paper to optimize the template size. Moreover, instead of finding the minimum Euclidean distance, a fast Fourier transform (FFT) - based method is used in the second stage of the proposed framework in order to save the computation time of template matching. Eventually within the final stage, the selected historical candidates by template matching can be aggregated to provide the multiple-step prediction of travel times on the current day. The details of methodologies for the three stages of proposed framework are presented as below.

Updating Dynamic Template

In this paper, the dynamic template is updated in real time according to the traffic flow fundamentals (e.g. congestion and bottleneck identification). The identified congested roadway segments are used to track the change in traffic conditions, so that the template size is dynamically computed to reflect the activation and propagation of shockwaves. Thereafter, a more accurate template matching and travel time prediction can be achieved based on the propagation of shockwaves in the future time intervals. The bottleneck identification technique is part of our proposed algorithm to update the dynamic template size. Considering the simple assumption and easy of implementation, a mixture model congestion identification algorithm which is developed by the Center for Sustainable Mobility at the Virginia Tech Transportation institute is used in this paper [27]. However, it should be noted that the proposed dynamic template matching method is not constrained to a specific technique, other similar approaches that deal with online congestion or bottleneck identification can also be considered as an alternative.

The mixture model congestion identification algorithm assumes the traffic speed across roadway road segments have an underlying fundamental diagram trend with randomness associated with the data. The variability of speed is substantial in congested traffic condition. Due to this random nature of traffic speed, stochastic models are the best choice for modeling the distribution of speed. Stochastic models have been proven to be really good tools in travel time reliability modeling [28, 29]. Assume the traffic condition is either congested or uncongested, a mixture distribution where each component corresponds to a specific traffic condition is used to model multistate traffic conditions. This algorithm does not require any pre-defined parameters

making it easier to implement compared to the state-of-the-art Chen's algorithm [30]. By using the traffic speed measurement over the spatiotemporal domain, this algorithm fits two lognormal distributions as demonstrated in Equation (1).

$$f(u | l, m_1, m_2, s_1, s_2) = l \frac{1}{\sqrt{2\pi} s_1} e^{-\frac{(\ln u - m_1)^2}{2s_1^2}} + (1 - l) \frac{1}{\sqrt{2\pi} s_2} e^{-\frac{(\ln u - m_2)^2}{2s_2^2}}. \quad (32)$$

where (μ_1, σ_1) and (μ_2, σ_2) are the mean and standard deviation of the first and second component distributions and λ is the mixture parameter. Thereafter, the threshold to identify the traffic condition can be calculated by locating the 0.001 quantile of the fitted uncongested distribution. Lastly, in order to filter noise in the results, a morphological operation is used to remove the spatiotemporal local uncongested regions identified in the congested area. The current criterion is set at four cells.

The congestion identification algorithm produces a spatiotemporal binary matrix of traffic state, in which the value of zero and one represent the uncongested and congested conditions, respectively. For real-time application, the online congestion identification is conducted for every time interval. A default value of template size is assumed if the real-time traffic status is uncongested. Here, the default value is selected based on the optimum window width for fixed template matching approach. The details of how to find the default template size will be demonstrated on the case study later on. Once the current traffic is identified within a bottleneck, then the corresponding template width is computed as the time difference between bottleneck activation and current time interval. In this way, the dynamic traffic information of bottleneck activation and propagation are included in the template window. It should be pointed out that multiple bottlenecks of different locations are considered as correlated to each other. So the congested traffic conditions along the entire roadway stretch are integrated together to come up the strategy of finding dynamic template. Specifically, a projection map is conducted by the summation of binary matrix along vertical direction. Therefore, the activation of each bottleneck can be identified from the projection map, and then the dynamic template can be updated by the time intervals from current congested time back to the bottleneck activation point. In this way, the dynamic template keeps updating in response to the real time traffic condition and bottleneck propagation.

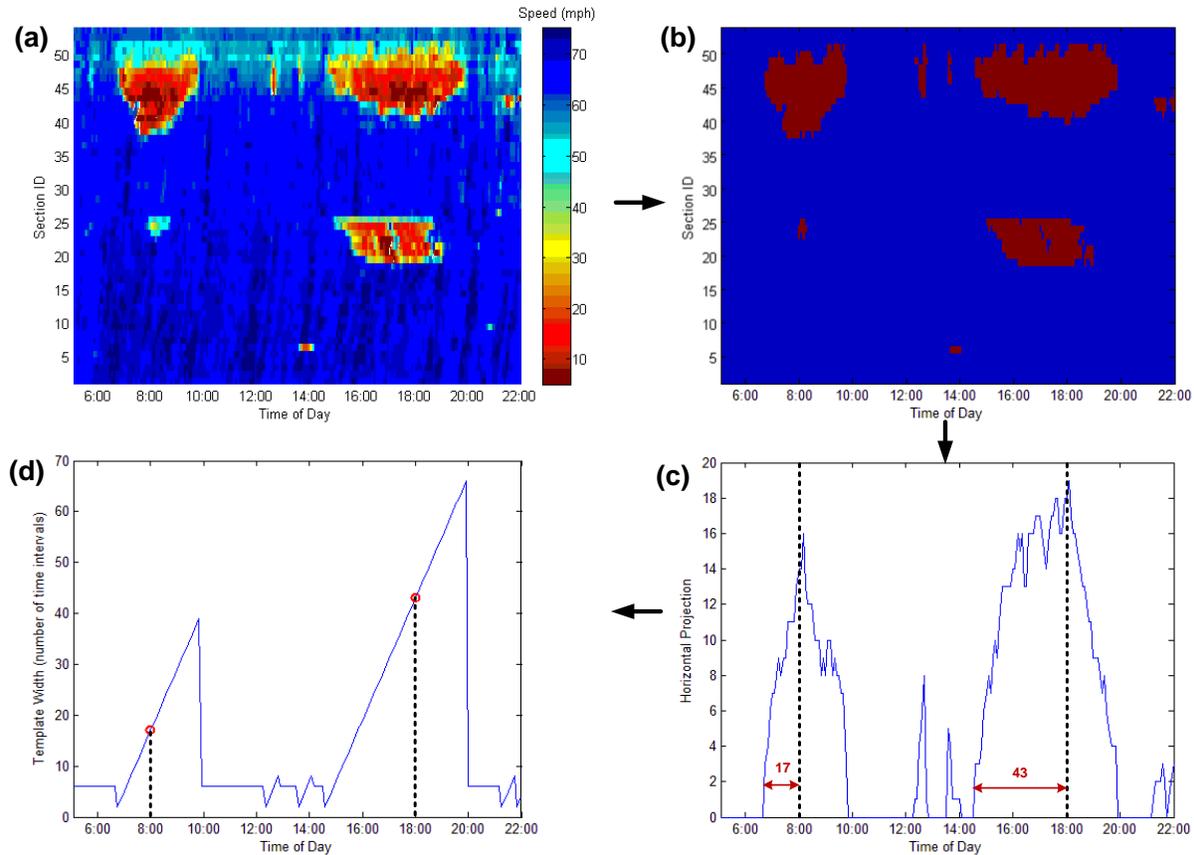


Figure 23: Calculate dynamic template width by congestion identification; (a) speed contour; (b) congestion identification result; (c) horizontal projection of congestion map; (d) dynamic template width.

An example of calculating the dynamic template width is demonstrated in **Figure 23** by using the 5-minute aggregated probe data along I-64 between Richmond and Hampton Roads on June 19, 2012 from 5:00 AM to 22:00 PM. The contour of speed measurement over spatial and temporal is presented in **Figure 23** (a) by different scale of colors from blue to red. The binary matrix is obtained as shown in **Figure 23** (b) using the described mixture distribution algorithm. Thereafter, the horizontal projection of the binary matrix is computed in **Figure 23** (c). The dynamic template is updated using the horizontal projection map. For instance, assume the current time is 8:00 AM, the corresponding template width is calculated as 17 time intervals since the current bottleneck begins on 6:35 AM. For the scenario on 18:00 PM, the same procedure can be used to calculate the template width as 43 time intervals since the associated bottleneck starts at 14:25 PM. Therefore, the width of dynamic template is calculated for each time interval in **Figure 23** (d).

Matching Traffic Patterns

In this section, template matching technique is conducted to match the template representing real time traffic pattern with historical dataset in order to select the most similar candidates. Euclidean distance is the most widely used criterion to calculate the similarity of template matching result [24]. It should be noted that matching current and historical traffic data by Euclidean distance has also been frequently used to predict travel time or traffic flow in recent years [8, 31, 32]. In these studies, the average Euclidean distance between the current traffic data and the data with same dimension from historical days is calculated to represent a similarity measure by Equation (2).

$$d(C, h_i) = \sum |x(C, T) - x(h_i, t)|. \quad (33)$$

where $M(c, L)$ and $M(h, L)$ represent the traffic data of the current and historical time intervals, respectively; $d(c, h)$ is the summation of absolute error between the template and matching matrix in each cell.

However, the main disadvantage of the above method is the high computational cost [24]. It should be noted that the proposed template matching algorithm will be used on a large historical dataset, since a large set of different traffic patterns is helpful to produce higher prediction accuracy. In order to fill the requirement of real-time computation, a fast template matching approach is needed to avoid the delay in the system.

The idea of using the convolution theorem was proposed to solve this problem and proved to be an efficiently computation alternative and very easy to implement [24, 33]. In this paper, the FFT-based convolution method is used for fast matching template of traffic pattern. In this approach, the previous objective of finding the least Euclidean distance is replaced by searching the maximum cross correlation represented by Equation (3). Mathematically, the convolution theorem states that the Fourier transform of a convolution is the pointwise product of Fourier transforms. Therefore, the convolution between the current traffic pattern and target matrix of the historical day can be calculated as Equation (4).

$$\text{conv}(C, h_i) = \sum x(C, T) \times x(h_i, t). \quad (34)$$

$$\mathcal{F}^{-1} \{ \mathcal{F} \{x(C, T)\} \times \mathcal{F} \{x(h_i)\} \}. \quad (35)$$

where \mathcal{F} is the Fourier transform, and \mathcal{F}^{-1} is the reverse function of Fourier transform; $x(h_i)$ represents the entire measurements of traffic status over spatial and temporal for i^{th} historical day. In the previous studies of using Euclidean distance to match template with one historical day, the template matching process needs multiple iterations by shifting the template window along the time period of the entire day. However, only one computation by Equation (35) is enough to complete the template matching in the FFT-based convolution method. The output of Equation (35) is a set of similarities represented the matching result between template and the data matrix in historical day h_i . Therefore, the best matching can be located on time interval t_i , which corresponds to the maximum similarity denoted by $d_{\max}(C, h_i)$.

The calculation of Equation (35) is iteratively conducted between the real time template and each historical day. Therefore, several candidates are selected according to the descending order of the similarity measure. Suppose the maximum number of candidates is denoted by K , the set of candidates H_c is selected as

$$\begin{aligned}
H_c &= \{h_1, h_2, \dots, h_K\} \\
\text{where } h_i &: \hat{g}_i, d_{\max}(C, h_i) \Big|_{t_i} = \max F^{-1} \{F\{x(C, T)\} \times F\{x(h_i)\}\} \\
h_1 &= \arg \max d_{\max}(C, h_i) \\
d_{\max}(C, h_i) &\leq d_{\max}(C, h_{i+1})
\end{aligned} \tag{36}$$

where h_i is the i^{th} selected candidate from historical dataset. The selected candidates represent the best matching to the current traffic status and will be used to calculate future travel times.

Travel Time Prediction

The future experienced travel times on the current day can be calculated based on the selected historical candidates. Considering the stochastic nature of a traffic system, the travel time prediction problem can be recognized as a time series prediction for nonlinear dynamic (chaotic) systems [34, 35]. The future traffic state for the current day can be predicted by the linear combination of subsequent traffic state of each candidate from the historical dataset, and the corresponding weight is defined as the normalized similarity measure. The predicted traffic state starting from time interval $c+p$ is obtained as

$$M(C, T+p) = \sum_{i=1}^K w(h_i) \times M(h_i, t_i+p) \tag{37}$$

$$w(h_i) = \frac{d_{\max}(C, h_i)}{\sum_{i=1}^K d_{\max}(C, h_i)} \tag{38}$$

where $M(h_i+p)$ represents the subsequent traffic state for i^{th} candidate starting from departure time t_i+p till the end of trip; and $w(h_i)$ denotes the weight of i^{th} candidate data.

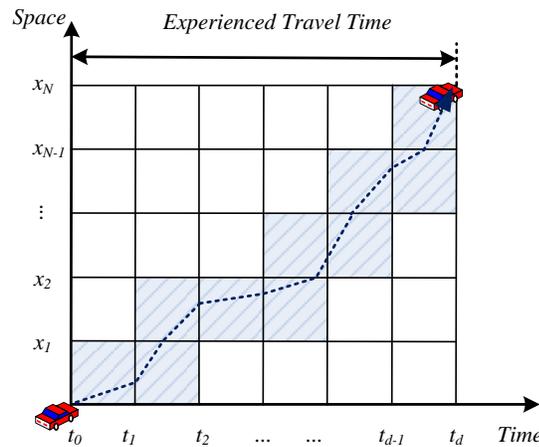


Figure 24: The calculation of experienced travel time.

Other than the predicted traffic state, the experienced travel time can also be predicted based on the subsequent experienced travel time of each candidate. Experienced travel time is the actual, realized travel time that a vehicle could experience during a trip. If a vehicle leaves a

trip origin at the current time, the roadway speed will not only change across space but also across time during the entire trip. Therefore, the traffic state evolution over space and time is considered in our approach to calculate experienced travel times. The speed values of shaded cells are used to compute experienced travel times. In this paper, the traffic state is assumed to be homogenous within each cell. Therefore the trajectory slope, which represents the traffic speed, is a constant value in each cell. Assume the trip starts from time interval t_n . In this way, once the vehicle enters a new cell, the trajectory within this cell can be drawn as the straight dotted line in **Figure 24** with the slope value equal to the traffic stream speed. Finally, the experienced travel time can be calculated when the trajectory reaches the downstream boundary of the last roadway section (destination). In this way, the subsequent experienced travel time of each candidate can be obtained and the corresponding weight (recurrent probability) is defined by the similarity measure of Equation (38). Therefore, the travel time distribution of the future trip departures on $T+p$ can be represented as

$$tt^{\text{exp}}(C, T+p) = \{tt^{\text{exp}}(h_i, t_i + p), w(h_i) | i = 1, L, K\}. \quad (39)$$

where $TT(c+p)$ represents the experienced travel time starting from time interval $c+p$; and $TT(h_i+p)$ denotes the subsequent travel time of i^{th} candidate according to the calculation in **Figure 24**. The travel time prediction result can also be calculated as the average value using Equation (9).

$$\bar{tt}^{\text{exp}}(C, T+p) = \hat{\mathbf{a}} \sum_{i=1}^K w(h_i) \times tt^{\text{exp}}(h_i, t_i + p). \quad (40)$$

CASE STUDY

Test Environment

The case study is conducted based on privately developed INRIX traffic data, which is mainly collected by GPS equipped vehicles. The collected probe data is supplemented by traditional road sensors, as well as mobile devices and other sources [36]. Since heavy traffic volumes are usually observed along I-64 heading to Virginia Beach during summer seasons and weekends, efficient and accurate travel time prediction can be helpful to travelers in planning their trips and reducing traffic congestion around the area. The INRIX data on the main segments along I-64 are used to construct the travel database in our study, which covers the major congested areas on I-64 from Richmond to Virginia Beach. The layout of the selected freeway stretch is presented in Figure 25, which includes 54 segments with the total length 67 miles. The average freeway segment length is 1.2 miles and the length of individual segment is unevenly distributed ranging from 0.1 to 6.4 miles. The average speed or travel time for each roadway segment are provided in the raw data, which are collected by every one-minute interval. In order to reduce the stochastic noises and measurement errors in the raw data, the raw speed information of each segment is aggregated by five-minute. Therefore, the traffic speed over spatial (upstream to downstream) and temporal (from 0:00 AM to 23:55 PM) can be obtained for each day, which is a data matrix with dimension 54 by 288. It should be noted that the full coverage of historical traffic speed data is required in this study. However, the problem of missing data is very common in the field measurements and thus must be addressed. Many traffic state estimation methods were proposed in order to obtain full coverage traffic state data by solving the mentioned problem [37, 38]. In the following sections, the traffic status is the full coverage traffic data after the process of data estimation. A detailed description of state estimation methods is beyond the scope of this paper and thus is not discussed further in this paper.

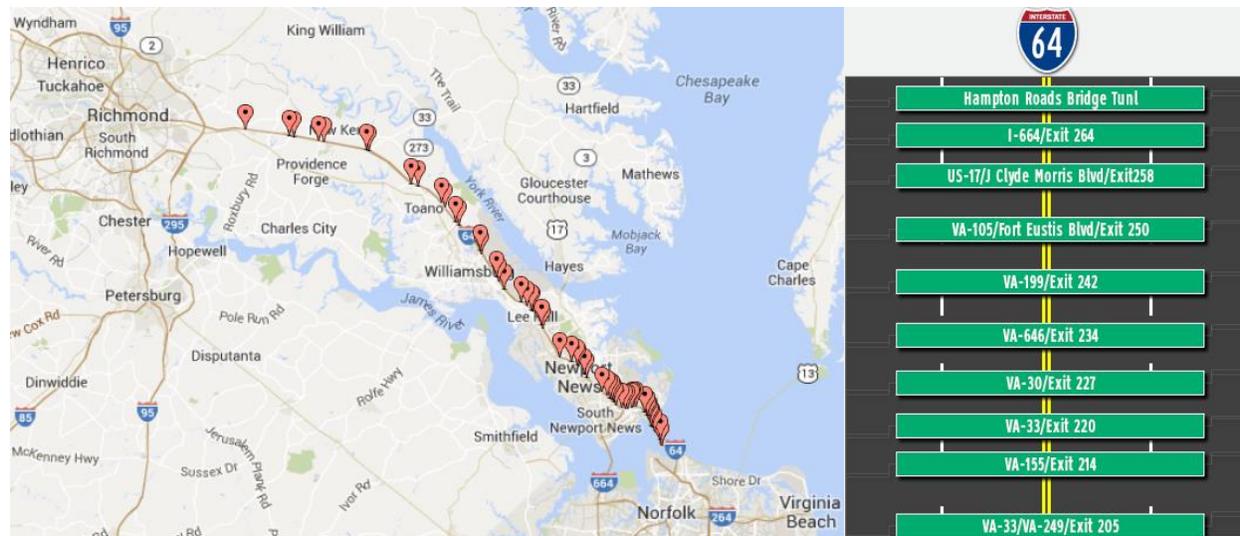


Figure 25: Layout of the selected freeway stretch.

Given the daily speed matrix, the instantaneous and experienced travel times can be calculated afterward. The instantaneous travel time is the summation of travel times for each roadway segment on the same time interval, in which the speed is constant over time. However, the experienced travel time is computed by considering the speed evolution across time. In this case, the speed profiles are piecewise constant values and the trip trajectory is a combination of diagonal curves over time and space [39]. In the following tests, time period between 5:00 AM to 22:00 PM is considered as the test period for each day since most of the congestions are covered in this period. Moreover, the ground truth travel time is represented by the experienced travel time, and the predicted travel times by different predictors are compared with the ground truth data to evaluate the prediction performance. Considering that the selected freeway stretch has heavy congestion during summer holiday season due to the high volume of traffic heading towards Virginia Beach, the traffic data from April to July, 2012 are used as the historical dataset and the subsequent traffic data in August and September, 2012 are employed as the test dataset to evaluate predictors.

Four prediction methods are tested on the selected freeway stretch. Firstly, the real time speed on each freeway segment is assumed to be constant on the future trip in the naïve idea to predict travel time. Therefore, the instantaneous travel time represents the easiest predictor to compare with other complicated methods. In order to explore the benefit of using dynamic template, a template matching by fixed window size is also included in this study and denoted as Method 1, in which a fixed template width is used as opposing to the dynamic template width in the proposed method. For the purpose of simplicity, the proposed dynamic template matching algorithm is denoted as Method 2. It should be pointed out that the pool of historical data for Method 1 and 2 only include the traffic data from April to July, 2012, which means the same historical dataset is used for different test days. However, the historical dataset keep updating to include all the previous days before the test day in Method 3, in which the dynamic template matching is used by an incremental historical dataset. For instance, the traffic data from April 1st to August 31st, 2012 are used as the historical dataset if the test day is September 1st, 2012. By comparing Method 2 and 3, we can explore the benefit of using incremental historical dataset for field implementation. It should be noted that the proposed dynamic template matching method

has the flexibility to use the updated historical dataset, comparing to other methods only use constant historical data pool, e.g. artificial neural networks [9, 15-17] and SVR [18, 19]. Considering the real world application, this is very important given that the characteristics of driver or traffic flow may change over time. For instance, an additional truck lane is added on the existing freeway and then the historical traffic data before this change may not provide a good pool of past traffic experiences to predict future traffic patterns.

Both relative and absolute prediction errors are used to evaluate the performance of predictors. The absolute error is denoted by the mean absolute error (MAE) using Equation (10), which represents the average absolute deviations between the predicted and the ground truth values. The corresponding relative error is represented by the mean absolute percentage error (MAPE) of Equation (11), which denotes the absolute proportional deviations between the predicted and the ground truth values.

$$MAE = \frac{1}{I' J} \sum_{j=1}^J \sum_{i=1}^{I'} |y_i^j - \hat{y}_i^j|. \quad (41)$$

$$MAPE = \frac{100}{I' J} \sum_{j=1}^J \sum_{i=1}^{I'} \frac{|y_i^j - \hat{y}_i^j|}{y_i^j}. \quad (42)$$

where J is the total number of days; I' is the total number of time intervals in one day (i.e., 204 intervals occurring every five minutes between 5:00 am and 22:00 pm); and y_i^j and \hat{y}_i^j denote the ground truth and the predicted value, respectively, of the experienced travel time for the i^{th} time interval on the j^{th} day in the test dataset.

Calculate the Threshold for Congestion Identification

In order to use the proposed dynamic template matching method, the threshold in the mixture model approach to identify congestion needs to be calculated firstly for the historical dataset. The daily speed matrices from April to July, 2012 are used to generate the histogram of speed values. Here, the histogram of speed value is normalized so that the fitted distribution can be plotted on top of it. Fitting the histogram by the mixture log-normal distribution as Equation (32), the parameters λ , μ_1 , σ_1 , μ_2 , σ_2 are estimated as 0.08, 24.2, 11.0, 65.1, 4.9. The two distribution curves are shown in **Figure 26** by red color to represent uncongested traffic and green color to represent congested traffic. Therefore, the threshold is selected by the speed value corresponding to 0.001 percentile of the uncongested traffic distribution, which is computed as 47.8 mph according to the cumulative distribution function (cdf) of its fitted distribution.

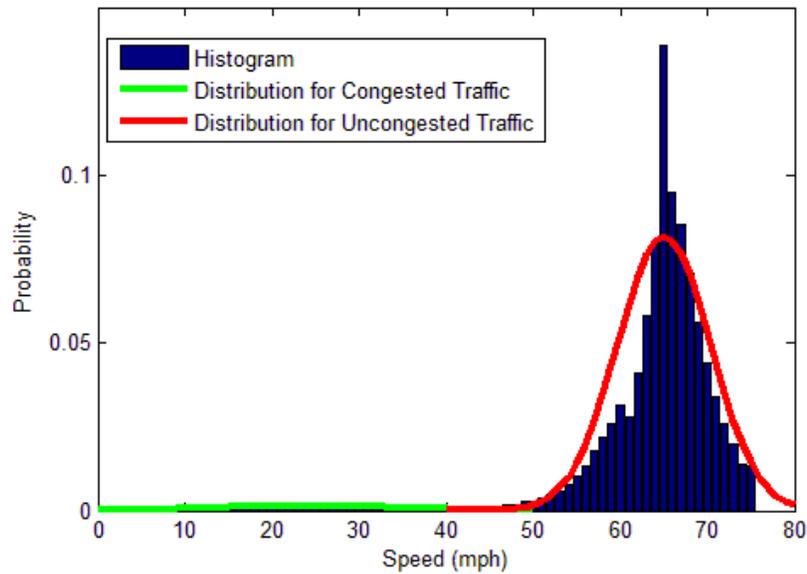


Figure 26: Calculate the threshold for congestion identification.

Find the Optimum Window Width for Fixed Template Method

As aforementioned, a default value of template width is assumed in the dynamic template matching approach if the real-time traffic status is uncongested. The default template width is selected in this study by finding the optimum window width for fixed template method. Given that a fixed window width L is used to match the real time traffic pattern with historical data and eventually the selected K number of candidates are used to predict travel times, the template matching with fixed window size is essentially a k nearest neighbor (k -NN) method in the previous studies [31, 32, 40, 41]. These studies demonstrate the performance of k -NN approach depends on the parameters of candidate number K and window width L . Therefore, the variation of prediction accuracy by using the fixed template method with different K and L values is investigated, which is used as the criterion to select parameters for both fixed and dynamic template matching methods.

The test result on the selected I-64 dataset indicates the range of K between 4 and 14 produces a prediction result with very little variation for the fixed-window template matching method. This phenomenon that number of candidate within a certain range has very little impact on the matching result has also been observed in the similar work in [31]. Hence, the value of K is assumed to be constantly as 10 in the following tests for both fixed template matching and the proposed dynamic template matching methods.

The impacts of using template width between 5 to 50 minutes for various prediction horizons are presented in **Figure 27**. Generally the minimum MAPE can be obtained when template width of 30 minutes is used for various prediction horizon between 0 to 30 minutes, the only exception is that the MAPE associated with template width of 35 minutes is a slightly less than the template width of 30 minutes for prediction horizon of 20 minutes. Hence, the template width of 30 minutes is selected as the optimum window width for fixed template method, and it's also used as the default template width in the dynamic template matching approach when the real-time traffic condition is uncongested.

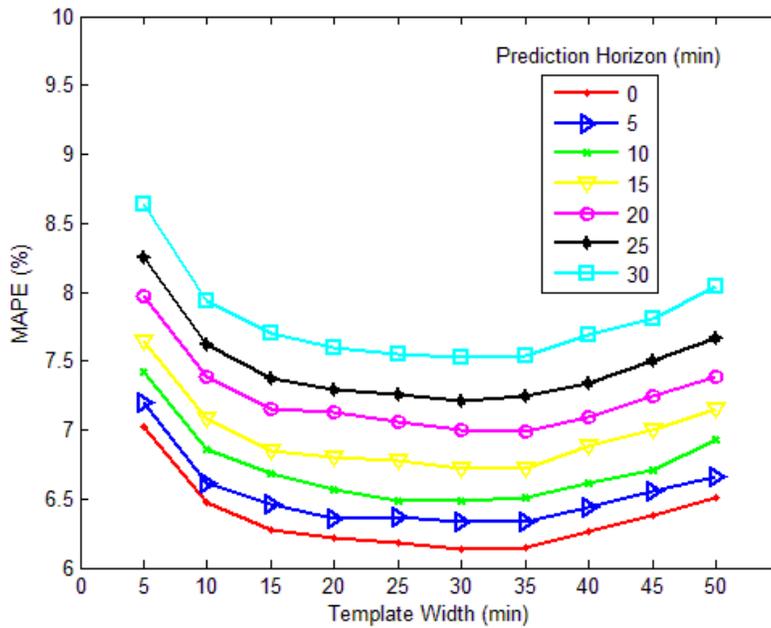


Figure 27: Impacts of template width on prediction accuracy

Test Results

Table 7 presents the MAE and MAPE values by using four methods for prediction horizon between 0 to 30 minutes. The instantaneous travel time produces the worst prediction results especially for long prediction horizons, in which the MAPEs increase from 10.81% to 16.19% for prediction horizon of 0 to 30 minutes. Compared to instantaneous travel time, the prediction accuracy by using Method II is greatly improved, and the range of relative errors is between 6.13% to 7.52% for departure time up to 30 minutes in the future. Moreover, the dynamic template in Method II further improves the prediction performance of template matching method as opposing to a fixed template size in Method I. The MAPEs is ranging between 5.56% to 6.63% when prediction horizon increases from 0 to 30 minutes. Instead of the constant historical dataset in Method II, the incremental historical dataset in Method III is helpful to the proposed dynamic template matching method to produce less prediction errors, in which the MAPEs increase from 5.21% to 6.28% for prediction horizon of 0 to 30 minutes. This improvement is reasonable considering that the incremental historical dataset has more traffic patterns especially from the recent past days, which is beneficial to produce more accurate template matching result.

Table 7: Prediction results by four methods for prediction horizon between 0 to 30 min.

		Prediction Horizon (min)						
		0	5	10	15	20	25	30
Instantaneous	MAE (min)	8.33	8.89	9.46	9.83	10.37	10.81	11.43
	MAPE (%)	10.81	11.64	12.42	13.15	13.63	14.74	16.19

Method 1	MAE (min)	4.62	4.74	4.91	5.15	5.29	5.45	5.68
	MAPE (%)	6.13	6.31	6.48	6.72	6.97	7.21	7.52
Method 2	MAE (min)	4.23	4.34	4.45	4.53	4.67	4.85	5.03
	MAPE (%)	5.56	5.64	5.81	6.02	6.19	6.37	6.63
Method 3	MAE (min)	3.92	3.98	4.06	4.24	4.38	4.48	4.71
	MAPE (%)	5.21	5.29	5.42	5.61	5.73	5.96	6.28

Method 1: fixed template matching

Method 2: dynamic template matching

Method 3: dynamic template matching with incremental historical dataset

Rather than evaluating the average performance of each predictor, the prediction accuracies for different traffic conditions are also investigated in this study. Given that it's very easy to make predictions for uncongested traffic condition, the prediction accuracy under congested time periods is more valuable since travelers would need the predicted travel time to assist them under congested traffic. Therefore, the congested periods during August 2012 are selected to test the performance by using the fixed and dynamic template matching methods as opposing to instantaneous travel time.

The congested traffic condition can be defined as the speed under 80% of the free flow speed, which is usually used a typical value for the speed at capacity on freeways. Therefore, the congested period at least 30 minutes long is identified in this study when the corresponding travel times are higher than 1.25 times of free flow travel time. Generally, up to four congestion periods can be extracted during a day, which are morning, noon, afternoon and evening congested periods. Considering 31 days in August 2012, a total of 50 congested periods can be identified. The relative error of MAPE is calculated to assess the prediction accuracies between the predicted travel times and ground truth values during congested periods. It should be noted the index of congestion period is obtained by sorting the corresponding MAPE produced by the instantaneous travel time method in an ascending order. **Figure 28** demonstrates the MAPE values using instantaneous travel time, Method 1 and Method 2 on each congested period for prediction horizon of 15 minutes. Out of the total 50 congestion periods, the template matching methods (Method 1 and 2) produce fewer errors than instantaneous method for 43 periods. For the left side of congestion periods (index from 1 to 22), the dynamic template in Method 2 doesn't provide an apparently improvement of prediction performance compared to the fixed template in Method 1. However, Method 2 produces fewer errors than Method 1 for the right side of congestion periods (index from 23 to 50). Given that the right side of congestion periods corresponds to the days with serious traffic congestion, dynamic template works better than fixed template for highly congested traffic conditions.

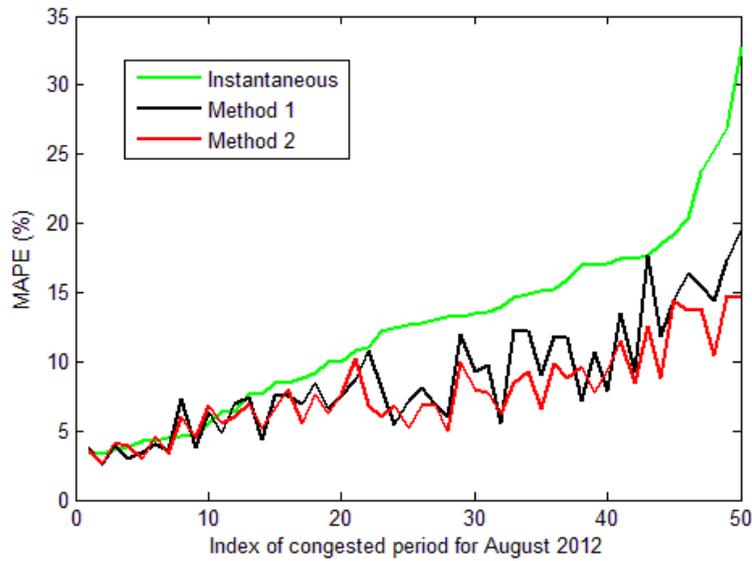


Figure 28: MAPE of congested periods by three methods for 15 min prediction horizon.

Figure 29 demonstrates the different results by using fixed and dynamic template matching methods on a sample day. The contour of speed matrix over space and time on August 18, 2012 is used as the test data. Assume the current time is 16:00 pm, the ground truth travel time when departures on 16:15 pm (prediction horizon of 15 minutes) can be calculated as 123 minutes by drawing the green trip trajectory on the speed contour as demonstrated in **Figure 24**. By using fixed window matching method, a template width ranging from 15:30 pm to 16:00 pm is extracted to match with historical dataset and the best matching result is located between 18:00 pm to 18:30 pm on July 19, 2012. Hence, the predicted travel time by fixed template matching method is 67 minutes, which underestimates the ground truth value by 56 minutes (MAPE of 45.5%). Alternatively, a much wider template width ranging between 11:30 am and 16:00 pm is used in the dynamic template matching method in order to cover the activation time of bottleneck. A similar traffic pattern with severe bottleneck to the dynamic template is selected between 12:00 pm to 18:30 pm on May 25, 2012, and the predicted travel time of 125 minutes only overestimates the ground truth value by 2 minutes (MAPE of 1.6%). This example clearly demonstrates that the dynamic template helps to find historical candidates with similar traffic pattern to the test day during congested traffic condition.

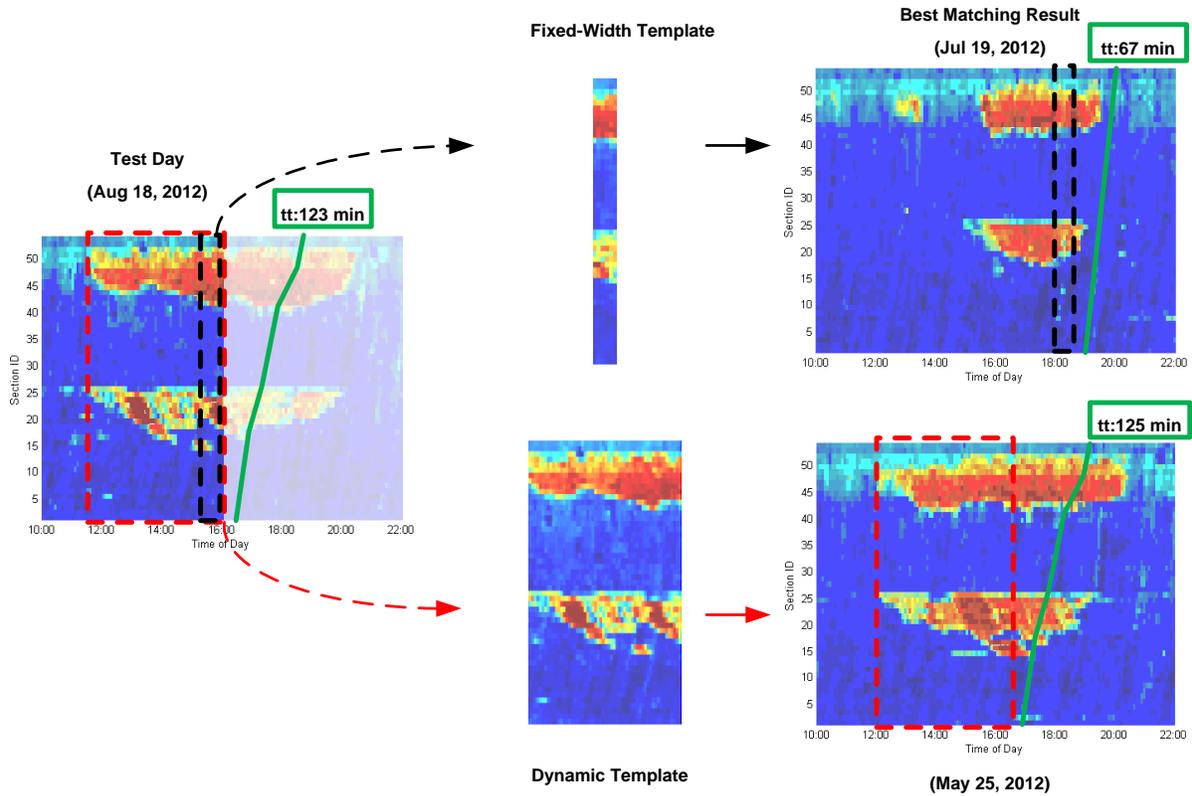


Figure 29: Illustration of fixed template width method vs. dynamic template matching.

The travel time curves produced by instantaneous travel time, fixed template matching, dynamic template matching method for a prediction horizon of 15 minutes, are compared with the ground truth data for a typical weekday on August 15, 2012 and a typical weekend on August 18, 2012 in **Figure 30**. The similar trends can be found on the results of two sample days. The instantaneous travel time experiences a temporal lag to the ground truth data, especially at the shoulders of the peak. Specifically, the instantaneous travel time highly underestimates the ground truth values when congestion is forming, and overestimates the ground truth travel time when congestion is dissipating. Comparatively, the fixed template method improves this problem and the predicted travel times have closer fit to the ground truth curve. It should be noted that Method 1 and 2 produce the same prediction results during uncongested traffic condition, even during the time period of congestion forming, since the dynamic template uses the same window width to the fixed template in these conditions. However, the proposed dynamic template matching method produces further improvements under congestion sustaining and dissipating periods, especially for the days with long congestion periods. The reason lies on the fact that the fundamental traffic flow theories help to find the bottleneck activation time and then the similar days with long congestion periods or severe bottlenecks can be accurately selected to improve the prediction accuracy.

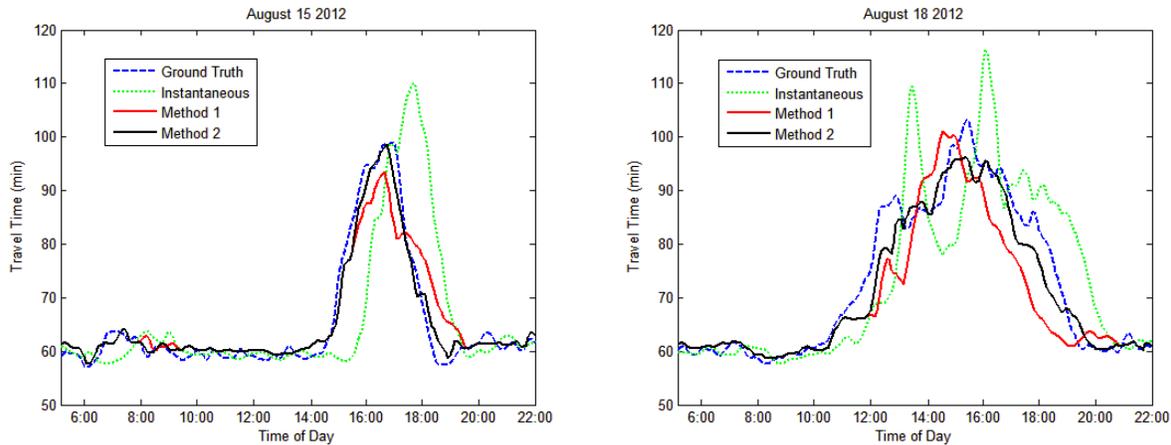


Figure 30: Travel time prediction results by three methods for 15 min prediction horizon by dynamic template matching on (a) August 15, 2012 (Wednesday); (b) August 18, 2012 (Saturday).

CONCLUSIONS AND RECOMMENDATIONS FOR FUTURE RESEARCH

This study develops a travel time prediction algorithm by matching traffic patterns from historical data to current real-time conditions. Instead of previous studies use a fixed window size in template matching techniques, the proposed method uses a dynamic template which is updated each time interval according to the real time traffic condition and the spatiotemporal shape of the congestion upstream of the bottleneck shape. Moreover, a fast Fourier transform based method is used in the template matching to reduce the computational costs for the purpose of real time application. Finally, the selected historical candidates which are similar to the current traffic conditions are used to predict the experienced travel times.

The probe data on a selected freeway stretch along I-64 from April to September, 2012 is used to test the performances of different predictors. The test results demonstrate the proposed dynamic template matching method produces the least prediction errors for prediction horizons up to 30 minutes into the future. Furthermore, the comparison results indicate the dynamic template enhances the prediction accuracy of template matching method at the shoulders of congestion periods, instead of the fixed template size. The proposed dynamic template matching also has the flexibility of using an incremental historical dataset and the test results show the prediction errors are further reduced instead of using a fixed historical dataset.

The proposed algorithm employed during this study provides a framework to use template matching technique to correlate real time and historical traffic data to predict experienced travel times. More advanced pattern recognition techniques can be considered to enhance the prediction accuracy or save the computation cost for future research.

ACKNOWLEDGEMENTS

This research effort was funded partially by Virginia Department of Transportation (VDOT) and partially by the Mid-Atlantic University Transportation Center (MAUTC). The authors also

appreciate Ralph L. Jones and Philomena Lockwood from the Virginia Department of Transportation for their assistance and feedback.

REFERENCE

- [1] David Schrank and Tim Lomax, "2007 Urban Mobility Report," Texas Transportation Institute 2007.
- [2] Huizhao Tu, "Monitoring Travel Time Reliability on Freeways," Ph.D., Department of Transport and Planning, Technische Universiteit Delft, 2008.
- [3] Pierre-Emmanuel Mazaré, Olli-Pekka Tossavainen, Alexandre Bayen, and Daniel B Work, "Trade-offs between Inductive Loops and GPS Vehicles for Travel Time Estimation: A Mobile Century Case Study," in *Transportation Research Board 91st Annual Meeting*, Washington, D.C., 2012.
- [4] Hao Chen and Hesham A. Rakha, "Prediction of Dynamic Freeway Travel Times based on Vehicle Trajectory Construction," in *15th International IEEE Conference on Intelligent Transportation Systems*, Anchorage, AK, 2012, pp. 576 - 581.
- [5] Hao Chen, Hesham A. Rakha, Shereef A. Sadek, and Bryan J. Katz, "A Particle Filter Approach for Real-time Freeway Traffic State Prediction," in *Transportation Research Board 91st Annual Meeting*, Washington D.C., 2012.
- [6] Hao Chen, Hesham A. Rakha, and Shereef A. Sadek, "Real-time Freeway Traffic State Prediction: A Particle Filter Approach," in *14th International IEEE Conference on Intelligent Transportation Systems*, Washington, DC, USA, 2011, pp. 626-631.
- [7] Lili Du, Srinivas Peeta, and Yong Hoon Kim, "An Adaptive Information Fusion Model to Predict the Short-term Link Travel Time Distribution in Dynamic Traffic Networks," *Transportation Research Part B: Methodological*, vol. 46, pp. 235-252, 2012.
- [8] Jiwon Myung, Dong-Kyu Kim, Seung-Young Kho, and Chang-Ho Park, "Travel Time Prediction Using k Nearest Neighbor Method with Combined Data from Vehicle Detector System and Automatic Toll Collection System," *Transportation Research Record: Journal of the Transportation Research Board*, vol. 2256, pp. 51-59, 2011.
- [9] J.W.C. van Lint, S.P. Hoogendoorn, and H.J. van Zuylen, "Accurate Freeway Travel Time Prediction with State-space Neural Networks Under Missing Data," *Transportation Research Part C: Emerging Technologies*, vol. 13, pp. 347-369, 2005.
- [10] Eleni I. Vlahogianni, John C. Golias, and Matthew G. Karlaftis, "Short-term Traffic Forecasting: Overview of Objectives and Methods," *Transport Reviews*, vol. 24, pp. 533-557, 2004.
- [11] Xiang Fei, Chung-Cheng Lu, and Ke Liu, "A Bayesian Dynamic Linear Model Approach for Real-time Short-term Freeway Travel Time Prediction," *Transportation Research Part C: Emerging Technologies*, vol. 19, pp. 1306-1318, 2011.
- [12] Jiann-Shiou Yang, "Travel Time Prediction Using the GPS Test Vehicle and Kalman Filtering Techniques," in *Proceedings of the 2005 American Control Conference*, 2005, pp. 2128-2133.
- [13] Jingxin Xia, Mei Chen, and Wei Huang, "A Multistep Corridor Travel-Time Prediction Method Using Presence-Type Vehicle Detector Data," *Journal of Intelligent Transportation Systems: Technology, Planning, and Operations*, vol. 15, pp. 104-113, 2011.
- [14] Jingxin Xia and Mei Chen, "Dynamic Freeway Corridor Travel Time Prediction Using Single Inductive Loop Detector Data," in *Transportation Research Board 88th Annual Meeting*, Washington D.C., 2009.

- [15] Xiaosi Zeng and Yunlong Zhang, "Development of Recurrent Neural Network Considering Temporal - Spatial Input Dynamics for Freeway Travel Time Modeling," *Computer - Aided Civil and Infrastructure Engineering*, vol. 28, pp. 359-371, 2013.
- [16] Yunlong Zhang and Hancheng Ge, "Freeway Travel Time Prediction Using Takagi - Sugeno - Kang Fuzzy Neural Network," *Computer - Aided Civil and Infrastructure Engineering*, 2013.
- [17] C.P.I.J. van Hinsbergen, A. Hegyi, J.W.C. van Lint, and H.J. van Zuylen, "Bayesian Neural Networks for the Prediction of Stochastic Travel Times in Urban Networks," *IET Intelligent Transport Systems*, vol. 5, pp. 259-265, 2011.
- [18] Lelitha Vanajakshi and Laurence R. Rilett, "Support Vector Machine Technique for the Short Term Prediction of Travel Time," in *IEEE Intelligent Vehicles Symposium*, Turkey, 2007, pp. 600-605.
- [19] Chun-Hsin Wu, Jan-Ming Ho, and D. T. Lee, "Travel-time Prediction with Support Vector Regression," *IEEE Transactions on Intelligent Transportation Systems*, vol. 5, pp. 276-281, 2004.
- [20] Wenxin Qiao, Ali Haghani, and Masoud Hamedi, "Short Term Travel Time Prediction Considering the Weather Impact," in *Transportation Research Board 91st Annual Meeting*, Washington D.C., 2012.
- [21] Brenda I. Bustillos and Yi-Chang Chiu, "Real-Time Freeway-Experienced Travel Time Prediction Using N-Curve and k Nearest Neighbor Methods," *Transportation Research Record: Journal of the Transportation Research Board*, vol. 2243, pp. 127-137, 2011.
- [22] Menglong Yang, Yiguang Liu, and Zhisheng You, "The Reliability of Travel Time Forecasting," *IEEE Trans. Intell. Transport. Syst.*, vol. 11, pp. 162-171, 2010.
- [23] Chumchoke Nanthawichit, Takashi Nakatsuji, and Hironori Suzuki, "Application of Probe-vehicle Data for Real-time Traffic-state Estimation and Short-term Travel-time Prediction on a Freeway," *Transportation Research Record: Journal of the Transportation Research Board*, pp. 49-59, 2003.
- [24] Roberto Brunelli, *Template matching techniques in computer vision: theory and practice*: John Wiley & Sons, 2009.
- [25] M.A. Turk and A.P. Pentland, "Face recognition using eigenfaces," in *Computer Vision and Pattern Recognition*, 1991, pp. 586-591.
- [26] T. Ahonen, A. Hadid, and M. Pietikainen, "Face Description with Local Binary Patterns: Application to Face Recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 28, pp. 2037-2041, 2006.
- [27] Mohammed Elhenawy and Hesham Rakha, "Title," unpublished|.
- [28] Feng Guo, Hesham Rakha, and Sangjun Park, "Multistate Model for Travel Time Reliability," *Transportation Research Record: Journal of the Transportation Research Board* 2010.
- [29] Feng Guo, Qing Li, and Hesham Rakha, "Multistate Travel Time Reliability Models with Skewed Component Distributions," *Transportation Research Record: Journal of the Transportation Research Board*, vol. 2315, pp. 47-53, 12/01/ 2012.
- [30] C. Chen, A. Skabardonis, and P. Varaiya, "Systematic identification of freeway bottlenecks," *Freeway Operations and Traffic Signal Systems 2004*, pp. 46-52, 2004.
- [31] H Chang, Y Lee, B Yoon, and S Baek, "Dynamic near-term traffic flow prediction: system-oriented approach based on past experiences," *IET Intelligent Transport Systems*, vol. 6, pp. 292-305, 2012.

- [32] Wenxin Qiao, Ali Haghani, and Masoud Hamed, "A nonparametric model for short-term travel time prediction using bluetooth data," *Journal of Intelligent Transportation Systems*, vol. 17, pp. 165-175, 2013.
- [33] JPI Lewis, "Fast template matching," in *Vision Interface*, 1995, pp. 15-19.
- [34] Arslan Basharat and Mubarak Shah, "Time Series Prediction by Chaotic Modeling of Nonlinear Dynamical Systems," in *12th International Conference on Computer Vision*, 2009, pp. 1941-1948.
- [35] T. Ikeguchi and K. Aihara, "Prediction of Chaotic Time Series with Noise," *IEICE Transaction on Fundamentals*, vol. E78, pp. 1291-1298, 1995.
- [36] INRIX. (2012). <http://www.inrix.com/trafficinformation.asp>. Available: <http://www.inrix.com/trafficinformation.asp>
- [37] Yibing Wang and Markos Papageorgiou, "Real-Time Freeway Traffic State Estimation Based on Extended Kalman Filter: A General Approach," *Transportation Research Part B*, vol. 39, pp. 141-167, 2005.
- [38] Yibing Wang, Markos Papageorgiou, and Albert Messmer, "Real-Time Freeway Traffic State Estimation Based on Extended Kalman Filter: Adaptive Capabilities and Real Data Testing," *Transportation Research Part A*, vol. 42, pp. 1340-1358, 2008.
- [39] J. W. C. van Lint and N. J. van der Zijpp, "Improving A Travel Time Estimation Algorithm by Using Dual Loop Detectors," *Transportation Research Record: Journal of the Transportation Research Board*, vol. 1855, pp. 41-48, 2003.
- [40] Zuduo Zheng and Dongcai Su, "Short-term traffic volume forecasting: A k-nearest neighbor approach enhanced by constrained linearly sewing principle component algorithm," *Transportation Research Part C: Emerging Technologies*, 2014.
- [41] S Lim and C Lee, "Data fusion algorithm improves travel time predictions," *Intelligent Transport Systems, IET*, vol. 5, pp. 302-309, 2011.