Data-Driven Travel Time Prediction from Latent Structures using Multiple Data Sources

Wan Li
University of Washington
Seattle, WS, 98125, USA
wan5@uw.edu

Choudhury Siddique
University of Washington
Seattle, WS, 98125, USA
siddic@uw.edu

Xuegang (Jeff) Ban*
University of Washington
Seattle, WS, 98125, USA
banx@uw.edu

ABSTRACT
Travel time prediction is extremely useful for both traffic management and traveler information services. Short-term travel time prediction is often conducted based on single data sources, e.g., loop detector data, and corrupted by noises, which may significantly influence the prediction accuracy. This paper investigates the application of Partial Least Square (PLS)-based prediction algorithm and dynamic mode decomposition (DMD) in short-term travel time prediction using various travel time estimation results from multiple data sources. In particular, twelve travel time estimates are computed for a road segment using loop detector data and GPS data. All these estimates are highly correlated. The PLS-based prediction algorithm predicts the current and near future (i.e., short term) travel times based on these estimates by accounting for their correlations. The proposed method is compared with another prediction algorithm, DMD, which only requires single estimates of travel times. Travel time data collected from a road segment on I-5 in Seattle are used for testing and comparison in this paper. The results show that the two methods are capable of extracting the dominant and relevant information from noisy data sets and reducing the complex data set to a lower dimension for efficient and effective prediction. The results also show that the PLS-based method that uses multiple data sources outperforms DMD that uses a single data source.

KEYWORDS
Travel time prediction, partial least squared, dynamic mode decomposition, loop detector data, GPS data, and data fusion.

INTRODUCTION
Reliable travel time information is important in transportation planning, operation, and management in urban areas where high population density is integrated with increasing traffic demand. Travel time in a spatial-temporal scope provides valuable information to transportation managers for efficient traffic control on major arterials. It is also critical to support advanced traveler information system (ATIS) and advanced traffic management systems (ATMS). Travelers are able to choose better departure times and make better road choices based on accurate and reliable travel time information.

Here we first distinguish travel time estimation and travel time prediction. Travel time estimation uses data collected during a trip to reconstruct the travel time of the trip that occurred in the past [1]. We call the travel time estimation results the travel time estimates in this paper. In order to obtain accurate travel time estimation, extensive research efforts have been made in recent years. Turner et al. [2] presented a comprehensive overview of travel time estimation methods and discussed the pros and cons of each method given the characteristics of the original data from different sources. There are two types of method to estimate travel times: instantaneous and experienced travel times. Instantaneous travel times are estimated based on the current prevailing traffic conditions, with the implicit assumption that such conditions would not change until the trip is completed. They can be calculated by summing up the travel time of each segment along the path at the departure time. The experienced travel times are those that actually experienced by vehicles as they travel on the route, which are affected by the real traffic conditions along the route. Xiao et al. [3] examined the differences between instantaneous and experienced travel time estimation based on detector data versus AVI data. They suggested that there is no significant difference under uncongested conditions, but a 10%-20% difference during congested condition for both the detector data and AVI data. These differences may be affected by the queue forming and dissipating speeds, route lengths, and locations of the congestion. Travel time estimation strongly rely on the characteristics and quality of the available data. Different traffic sensors will provide different traffic information. Mori et al. [4] classified the data sources into three categories: point detectors, interval detectors, and different combining data sources. Most studied in the literature applied one data source for travel time estimation. However, recent studies started to investigate various data fusion techniques in travel time estimation to increase the estimation reliability and reduce the sensing cost [5-6].

Travel time prediction is different from estimation in terms of the objectives and modeling techniques. The objective of travel time prediction is to forecast the travel time of traversing a road segment that will start in the present or future time [4]. For this purpose, it requires traffic data in real-time and in the past. Oh et al. [7]...
categorized data-driven travel time prediction methods into parametric (e.g., linear regression and time series) and non-parametric approaches (e.g., artificial intelligence and pattern searching) based on their underlying mechanisms and theoretical principles. It is believed that the parametric approaches can outperform the nonparametric approaches in term of their solid mathematical foundations [8]. However, the coefficients estimated from parametric approaches for certain road segments are often site-specific which may not be applicable to other segments [9-11]. Nonparametric approaches have advantages on less expensive computation effort and easy process on nonlinear data. For example, neural networks based methods have been applied in many studies due to their high level of accuracies, which are however considered as non-intuitive because of the unrevealed characteristics of hidden layers [7]. Many different types of neural networks were investigated from regular multilayer feedforward neural networks [12] to more complex generalized regression networks [13] and recurrent neural networks. Ma et al. [14] applied a deep Restricted Boltzmann Machine and Recurrent Neural Network architecture to predict traffic congestion evolution based on GPS data from taxis. Their accuracy can reach as high as 88%. Notice that travel time estimation and prediction discussed in this paper focus on travel times of a certain road segment at a certain time. They reflect the general traffic conditions of the segment at the specific time, rather than the traffic conditions experienced by individual travelers.

Most existing travel time prediction methods required large amount of data while solely focusing on single estimates of travel times from a single data source. As multiple estimates of travel times from different data sources (e.g., GPS and loop detector) are becoming increasingly available, they could be used for more accurate and reliable travel time prediction. This study aims to combine multiple estimates of travel times (i.e., travel time estimation results) in a partial least square (PLS) based prediction algorithm to predict travel times in short term. Similar algorithm was first developed in Coogan et al. [15] to predict traffic flow to determine the optimal traffic signal plans. In their study, the general long term (i.e., day-to-day) traffic trends were predicted based on the day-to-day traffic variations while ignoring the traffic disturbance in short term. If used for short term prediction, their method could lead to relatively large prediction errors. The PLS-based prediction method we propose here are tractable, fast, and reliable for real-time travel time prediction. First, we compute twelve estimates of travel times for a corridor with different methods using the two data sources, GPS data and loop detector data. These estimates are highly correlated. Then we conduct a short-term prediction of the “ground truth” travel times using the PLS-based prediction algorithm that can account for those correlations and extract the dominant and relevant information from the noisy travel time estimates. Finally, the prediction results are compared with another short-term prediction algorithm, the dynamic mode decomposition (DMD) method, which only requires single estimates of travel times from a single data source.

TRAVEL TIME ESTIMATION METHODS

It is necessary to understand how to apply different types of data to obtain travel time estimates. This study uses two types of data sources: loop detector data and GPS data. The data contain information of segment-wise average speeds at each time interval. By using such data, we estimate both instantaneous and experienced travel times. This study aims at combining different estimates of travel time in PLS-based and DMD-based prediction algorithms to improve prediction accuracy. We note that there are other data sources and other methods that can be used to compute such travel time estimates. The PLS and DMD methods presented here may be applied to those data sources or methods with proper modifications.

Instantaneous travel time estimation

Instantaneous travel time is estimated for each segment, which is determined by the detector location or the GPS recorded location. Let $tt_{\text{inst}}^{d}(t)$ denote the instantaneous travel time on day $d$ during time interval $t$.

$$tt_{\text{inst}}^{d}(t) = \sum_{i=1}^{l} \frac{t_{i}}{r_{i}(t)}$$

(1)

Here $l$ is the total number of segments and $t_{i}$ is the length of the segment $i$. The average speed on road segment $i$ at time $t$, $\sigma_{i}(t)$ is estimated using the speed recorded at the boundaries of road segment $i$.

$$\sigma_{i}(t) = \frac{v_{i}(t)+v_{i+1}(t)}{2}$$

(2)

Figure 1 Configuration for instantaneous travel time estimation

This method assumes that the traffic conditions keep unchanged over the entire time period. Under the congested traffic conditions, multiple stop-and-go conditions could occur at the road segment $i$ which however may not be recorded from neither detector data nor GPS data. As a result, the average speed in Eq. (2) may be overestimated under congested situations, leading to an underestimated travel time. However, this method can provide reasonably accurate estimations if the adjacent loop detectors or GPS recorded locations are close enough (less than 500 m) and the updating intervals of recorded data are less than 5 min [6].

Experienced travel time estimation

The instantaneous travel time assume the prevailing traffic conditions keep constant until the trip is completed. However, this may not be true especially under congested traffic conditions. The experienced travel time is estimated by accounting for the dynamic traffic conditions over time. Instead of using the average speed of each segment $\sigma_{i}(t)$, at the same time $t$, we used “real time” speed
\( \sigma_i(t + \tau_i) \) in travel time estimation, where \( \tau_i \) is the travel time from the origin of the corridor to the start of the segment \( i \).

\[
\tau_i^d(t) = \sum_{i=1}^{I} \frac{1}{\sigma_i(t + \tau_i)}
\]

In Figure 2, an example is given to illustrate the difference between instantaneous travel time and experienced travel time. Suppose we need to estimate travel time at 9:00 AM for a corridor containing four segments. Instantaneous travel time at 9:00 AM is estimated by summing up all segment-wise travel time (\( \tau_t \sim \tau_d \)) at 9:00 AM, resulting in 4 min. It ignores the dynamic traffic conditions and assumes the traffic states keep unchanged in the future 4 min. However, experience travel time accounts for the traffic changes over time. Suppose an incident occurred at segment 3 (\( i=3 \)) at 9:02 AM, causing the travel time at segment 3 increased to 5 min at 9:02 AM and segment 4 increasing to 2 min at 9:07 AM. The experienced travel time at 9:00 AM is estimated as 9 min in this case, much larger than the instantaneous travel time and more accurate. When the segment lengths and update intervals are small enough, the experienced travel time may be approximated as the "ground truth" travel time.

![Figure 2 Comparisons between instantaneous travel time and experienced travel time](image)

For travel time estimation at each time interval, we not only consider the current time interval \( t \), but also the previous time interval \( t-1 \), and the next time interval \( t+1 \). For example, when we estimate the instantaneous travel time at 8:00 AM, we consider the instantaneous travel time at 7:55 AM and 8:05 AM (if the unit time interval is 5 min) as two additional travel time estimates for better describing the network condition at 8:00 AM. As a result, we have 12 estimates of travel time (\( m=1,2, ..., 12 \)) at each time interval \( t \), as shown in Figure 3. In particular, we use the experienced travel time estimates calculated from GPS data for the current time interval as the "ground truth" travel times.

![Figure 3 Travel time estimates at time interval t](image)

**TRAVEL TIME PREDICTION**

In this section, the PLS-based travel time prediction algorithm is discussed first, and the DMD model is then introduced. As mentioned before, noisy data often make it hard to identify the patterns of travel times and affect the prediction accuracy. PLS-based model and DMD are all capable of decomposing and denoising the travel time estimates. PLS-based prediction methods combine twelve travel time estimates while DMD only uses one estimate in short-term travel time prediction.

**Prediction method based on PLS**

We apply the PLS-based prediction algorithm to predict the "ground truth" travel times using all available travel time estimates. The PLS algorithm constructs components as a linear combination of the original predictors (the 12 travel time estimates) while considering the correlations between predictors and response variables. In this study, the response variable is the ground-truth travel time, i.e., experienced travel time at current time interval from GPS data (\( m=11 \) in Figure 3). All other estimates are considered as predictor variables. PLS is especially useful for data that contain correlated predictor variables [16-18], which is also the case for this study since different travel time estimates are highly correlated as we will show later.

We aggregate the travel time estimates into the vector format. Vector \( \tau^d_m \in \mathbb{R}^T \) denotes the \( m \)-th travel time estimates at day \( d \). Denote \( D \) the total number of days, \( T \) the total number of time interval per day and \( M \) the total number of travel time estimates (\( M=12 \) in this study). \( \tau^d_m \in \mathbb{R}^{TM} \) is the vector of travel time estimates for all time intervals for day \( d \). Eq. (6) is the aggregated travel time estimate matrix.

\[
\tau^d_m = [\tau^d_1, \tau^d_2, ..., \tau^d_M]^T, \quad d = 1, ..., D, \quad m = 1, ..., M
\]

\[
\tau^d = [\tau^d_1, \tau^d_2, ..., \tau^d_M]^T, \quad d = 1, ..., D
\]

\[
TT = \begin{bmatrix}
(\tau^1)^T \\
(\tau^2)^T \\
\vdots \\
(\tau^D)^T
\end{bmatrix} \in \mathbb{R}^{D \times (TM)}
\]

where \( [\cdot]^T \) denotes vector transpose.

In order to learn the parameters from historical data and predict the current and future travel times, we divide the travel time estimates into two parts based on their time intervals. Denote \( T_p (T_p < T) \) the current time. The travel time data from all 11 estimates (exclude \( m = 11 \) in Figure 3) before time \( T_p \) are considered as predictor...
variables. We use travel time estimates up to \( T_p \) to predict the ground-truth value of travel times at \( t > T_p \). It is assumed that the variation of the 11 travel time estimates before \( T_p \) correlates with the ground truth (\( m = 11 \)) after \( T_p \). In practice, it is often more efficient to use the travel time estimates from the recent past of \( T_p \) (e.g., 2 hours) instead of all the estimates from the starting of the day; furthermore, it is often more accurate if the prediction is done for a short term say a few minutes. In the following, we describe the PLS method in general which can be readily modified to restrict the past data or the prediction horizon to any pre-defined time window; see the numerical section for more details.

In PLS, we aim at finding a collection of predictor components \( p^1, p^2, ... p^N \) with each \( p^i \in \mathbb{R}^{T_pM} \); predicted components \( q^1, q^2, ... q^M \) with each \( q^i \in \mathbb{R}^{(T-T_p)M} \); and a vector of weights \( \omega(d) \in \mathbb{R}^N \) with

\[
\omega(d) = [w_1(d) \quad w_2(d) \quad ... \quad w_N(d)]^T
\]

such that the travel time estimates before \( T_p \), \( tt_b \), and after \( T_p \), \( tt_a \), can be express as:

\[
\begin{align*}
\hat{tt}_b &= \overline{tt}_b + \sum_{i=1}^N w_i(d) p^i \quad (8) \\
\hat{tt}_a &= \overline{tt}_a + \sum_{i=1}^N w_i(d) q^i \quad (9)
\end{align*}
\]

where \( \overline{tt}_b \) and \( \overline{tt}_a \) are the average travel time estimates before and after \( T_p \).

After the predictor and predicted component estimated from PLS using historical data, we can predict the future ground truth travel times for a sample day after \( T_p \) using the calculated historical predictor component \( p^i \) and predicted component \( q^i \). Denote \( \hat{tt}_b^M \in \mathbb{R}^{T_pM} \) the \( M \) estimates of travel time before \( T_p \) on the sample day. The weights is estimated as:

\[
\hat{\omega} = ((\hat{tt}_b^M - \overline{tt}_b)(P^T)^\dagger)^T (10)
\]

where \( P \) is the vector of predicted component and \((P^T)^\dagger \) is the Moore-Penrose pseudoinverse of \( P^T \). Then the predicted ground truth value of travel time after \( T_p \) can be estimated:

\[
\hat{tt}_a = \hat{\omega}^T Q^T + \overline{tt}_a \quad \hat{tt}_a \in \mathbb{R}^{(T-T_p)M} (11)
\]

where \( Q^T \) is the vector of predicted components. \( P^T \) and \( Q^T \) are estimated from PLS.

In order to achieve a higher prediction accuracy, a prediction method based on PLS is developed, as shown in Figure 4. Given multiple data sources, i.e., loop detector data and GPS data, different estimates of travel time are calculated first as shown in Figure 3. The variable \( T_p \) denotes the current time interval and \( dt \) denotes the prediction interval, e.g., 5 min. Every time when \( T_p \) move forward, the predictors and ground-truth value need to be updated in PLS to generate predictor components, prediction components, and the corresponding weights.

**DMD prediction**

PLS-based prediction method is capable of conducting a real-time short-term prediction given a large amount of historical data from multiple sources. Through estimating the correlated predictor components and prediction components from historical data, the prediction is conducted for a certain day. In order to evaluate the performance of the PLS-based algorithm, we apply another prediction technique, the dynamic mode decomposition (DMD) method for short-term prediction. DMD is developed to understand, control or predict a complex system without knowing the underlying governing equations that drive the systems [19]. It has been widely used in the fields of fluid mechanics, atmospheric science, and nonlinear valves communities for data-driven learning and discovery [20, 21]. Compared to the PLS-based method that requires multiple travel time estimates, DMD only uses one estimate and decomposes it into a lower dimensional representation to produce a function that describe the dynamic processes contained in the data sequence.

![Figure 4 A prediction method based on PLS](image)

For real time prediction, DMD only uses the experienced travel time \((m=11 \text{ in Figure 3})\) before the current time and predict the future ground-truth travel time for a sample day. The data can be grouped into two sets:

\[
X = [tt_1 \quad tt_2 \quad ... \quad tt_k] \quad X' = [tt_2 \quad tt_3 \quad ... \quad tt_k] (12)
\]

with \( tt_k \) an initial condition of a dynamic system \( \frac{dx}{dt} = AX \) and \( tt'_k \) is corresponding output after some prescribed evolution time \( \tau \). DMD computes the leading Eigen decomposition of the best-fit linear operator \( A \) where:

\[
A = X'X^\dagger (13)
\]

where \( ^\dagger \) denotes the Moore-Penrose pseudoinverse. \( A \) is known as the Koopman operator. Mathematically, the Koopman operator \( A \) is a linear, time-independent operator such that

\[
tt_{k+1} = A \ast tt_k (14)
\]
where, $k$ indicates the specific data collection time and $A$ is the linear operator that maps the data from time $t_k$ to $t_{k+1}$. The goal in the DMD algorithm is to optimally construct the matrix $A$ so that the predicted travel times (denoted as $\tilde{t}_k$) are close enough to the true solution $tt_k$.

The DMD algorithm proceeds as follows [22]:

First, take the singular value decomposition (SVD) of $X$:

$$ X = U \Sigma V^* $$

where $*$ denotes the conjugate transpose.

Second, the matrix $A$ can be obtained by using the pseudoinverse of $X$ via the Singular Value Decomposition (SVD):

$$ A = X^N U^* $$

where $*$ denotes the conjugate transpose. In practice, it is computationally more efficient to compute $\tilde{A}$ the $r \times r$ projection of the full matrix $A$. Here $r$ is the rank of the reduced SVD approximation of $X$.

$$ \tilde{A} = U^* A U = U^* X^N V^N $$

Next, the Eigen decomposition of $\tilde{A}$ is computed as:

$$ \tilde{A}W = WA $$

where columns of $W$ are eigenvectors and $A$ is a diagonal matrix containing the corresponding eigenvalues $\lambda_k$.

Finally, we may reconstruct the Eigen decomposition of $A$ from $W$ and $\tilde{A}$. In particular, the eigenvalues of $A$ are given by $\tilde{A}$ and the eigenvectors of $A$ are given by columns of $\Phi$:

$$ \Phi = X^N V^N W $$

With the low-rank approximations of both the eigenvalues and eigenvectors in hand, the projected predicted travel time can be constructed for all time in the future. The approximate travel time at all future times is given by

$$ \tilde{t}_k = \sum_{r=1}^{R} b_r \phi_r e^{\omega r} = \Phi \text{diag}(e^{\omega r}) b $$

where, $\omega_r = \ln(\lambda_r)/\Delta t$, $\lambda_k$ is the DMD eigenvalues. $b_k$ is the initial amplitude of each mode. $\Phi$ is the matrix whose columns are the DMD eigenvectors. $\text{diag}(e^{\omega r})$ is a diagonal matrix whose entries are the eigenvalues $e^{\omega r}$. and $b$ is the vector of the coefficients $b_r$. The core of DMD method can be thought as a combination of dimensionality reduction techniques with Fourier Transforms in time [24].

In order to apply DMD to travel time prediction, two key parameters: $M$ and $L$ need to be defined and optimized based on historical data. $M$ is the number of past minutes of travel time data taken. $L$ is the number of minutes in the future (prediction intervals). We use the ground truth travel times ($m=11$ in Figure 3) for each minute between 7:00 AM and 8:55 AM to find the optimum values of $M^*$ that provides the most accurate travel time predictions for a future time. In other words, for each minute in the future, an optimum length of the past sequence was calculated, which is then used to predict the travel time in that specific time of future. The algorithm of finding $M^*$ is present.

Algorithm to find the optimal length of past sequence ($M^*$)

Set current time interval $N = 120$ (e.g., for 9:00AM)
Set $X = \text{training data matrix}$ (e.g., 50 days * 210 min)
for each $L$ between time interval 1 and 90 (min):
    for each $M$ between time interval 20 and 120 (min):
        Setup the data matrix $X$ and $X'$
        $X = \tilde{X}(.,N-m;N-1)$; $X' = \tilde{X}(.,N-m+1;N)$
        Set $r = m-1$ (use full rank).
        Find the DMD modes $\Phi$ using $X, X'$ and $r$.
        Predict for a $L$ using $\Phi$.
        Find Prediction Error.
    end
Find the $M^*$ corresponding to minimum error

NUMERICAL EXPERIMENTS

In this study, we focus on a southbound ten-mile segment of Interstate 5 in Seattle, WA, as shown in Figure 5. This segment passes downtown Seattle, and experiences heavy congestion during peak hours. We have the access to a 3-month historical data set, for both loop detector data and GPS data. The loop detector data for those days were collected from the DRIVE Net website [23]. GPS data were provided by an industry partner. The loop data and GPS data cover the weekdays and weekends from 1/1/2012 to 3/31/2012 during morning peak hours from 7:00 AM to 10:30 AM.

Figure 5 Test site in Seattle, WA

Figure 6 shows the travel time distributions ($m=2$ in Figure 3) of the study segment for weekdays and weekends separately. Each line in the figure represents travel times over 24 h. Travel time on weekdays are generally higher than weekends. The study area contains heavy commute traffic. The morning and evening peak hours on weekdays can be clearly identified from the two peaks in Figure 6(a). On off-peak hours or weekend, the travel times are fluctuated around 9 min.
Figure 6 Travel time distributions for weekdays and weekends

The prediction model should be tested on weekdays and weekends separately because of the different traffic patterns. This section only shows the predictions results for weekdays since travel times on weekdays are more fluctuated than weekends and more complex to predict. In order to test the PLS-based travel time prediction algorithm, we choose $T_p$ as 9:00 AM. We use all travel time estimates from 7:00 AM to 8:55 AM to predict the ground-truth value of travel time of 9:00 – 10:30 AM on the study segment. Predictor components and predicted components in PLS are generated to explain the variance of the predictor and response variables. Figure 7 shows that 5 (out of 12) pairs of components can explain more than 95% of the variance. At a result, we adopt only 5 pairs of components and the corresponding weights for travel time prediction in order to reduce the original dataset to a lower dimension while maintaining the dominant information from the travel time data. By substituting the PLS components and weights into Eq.(8)-(9), the predicted travel time for a certain sample day can be estimated for each time interval.

Figure 7 Percent of variance explained by the PLS components

For the sake of comparison, DMD kept the training and testing datasets identical with the PLS algorithm. The travel times between 7:00 AM and 10:30 AM for first 50 weekdays were used in this case as training data. To get the best prediction of the travel time at a specific future time, we need the optimal length of past sequence $M^*$ estimated from training data. The training data matrix had 50 rows and 225 columns, where each row denotes one weekday, and each column represents travel time in each minute. We set the range of $M$ in between 20-120 min to explore the prediction interval $L$ from 1-30 min in the future starting from 9:00 AM. For the best possible prediction, we use the full rank of the data matrix. Figure 8 demonstrates $M^*$ for different future time ($L$) starting from 9:00 AM. It is evident that, the prediction error increases with the increase of $L$ in general. Moreover, the value of $M^*$ seems to be arbitrarily changing for different $L$. This implies that, for DMD, looking at longer past sequence of data does not necessarily translate into better prediction result. For a fare comparison with the PLS method, in this study, we use $L = 5$ minute for prediction using DMD. The optimal length of past sequence for this case is found to be 27 min. This means, according to the training data, looking at past 27 min will yield the best prediction for the travel time at 5 min into the future. The training error was around 26% when the prediction interval is 5 min. Using $L = 5$ minute and $M^* = 27$ minutes, we can predict the travel time between 9:05 AM and 10:30 AM for 15 weekdays.

Figure 8 Prediction Error for different future time using training data
Figure 9 shows the prediction results by the two methods for three testing days. Red star lines represent the ground truth travel times. The prediction for PLS and DMD are conducted in every 5 min starting from 9:00 AM. For each predication, travel time data up to the previous time interval are used. The results suggest that both the PLS-based and DMD based prediction algorithms predict reliable travel times that follow the similar trend as the ground truth value, no matter the ground truth travel time is stable (e.g., Test for day 11) or unstable (Test for day 3 and 14). In particular, prediction results from DMD have large variations, e.g., at 9:25 AM on test day 3 and 9:55 on test day 14. It is probably because that full rank travel time data are used in this experiment instead of low-rank approximation of the data. All variance have been accounted for in the prediction including the dominant information and some irrelevant characteristics. PLS results are much smoother probably due to two reasons. First, only 5 pairs of predictor and predicted components are applied, i.e., 95% of variance have been considered. Less significant information are excluded from the data. Second, multiple estimates of travel time from both data sources are used in prediction, which also lead to a higher predication accuracy. The prediction error from the PLS algorithm is 4.8%, which is lower than the DMD algorithm at 6.7%. DMD algorithm only uses the observations from the single estimate before the predicted time window (red star line from 7:00 – 9:00 AM) for the sample days.

Figure 9 PLS-based prediction results from sampled days

PLS-based prediction algorithms required historical data to predict the future travel time in a certain time period. Similar to Figure 8 for DMD, Figure 10 shows the prediction results for different prediction intervals from 5 min to 30 min (dt in Figure 4). Shorter prediction intervals require larger computational effort since the updating frequency of prediction is higher. It clearly shows that as the intervals increase, the prediction errors increase as well. In this study, the 5-min prediction intervals give the best prediction accuracy.
This study showed the first step to conduct short-term travel time prediction with multiple data sources. We will continue to collect more travel time data to refine the model and produce more accurate and reliable results. With more data, we also plan to apply the proposed algorithms to the network level emphasizing the time of day, day of week, and seasonal variations of travel times.

**REFERENCE**


**Figure 10** Prediction results from different prediction intervals

**CONCLUSION**

This paper investigated the PLS-based and DMD-based prediction methods in the application of travel time prediction using two data sources, loop detector data and GPS data. We first computed twelve travel time estimates using different methods. These estimates are all correlated (and can be mostly explained by only 5 components as shown in Figure 7). In order to apply different estimates in travel time prediction while accounting for the correlations, we developed a short-term PLS-based prediction method to predict the “ground truth” travel times. The PLS algorithm constructs components as a linear combination of the original predictors (estimates) while considering the correlations between the predictor and response variables. The prediction results from PLS outperform another prediction algorithm, the Dynamic Mode Decomposition (DMD) method that only uses single travel time estimates. It implies that the PLS-based prediction method, by combing various estimates from multiple data sources, can be a helpful tool for short-term travel time prediction. This is particularly useful under the current and future situations when data sources for travel time prediction are increasingly available.

This study showed the first step to conduct short-term travel time prediction with multiple data sources. We will continue to collect more travel time data to refine the model and produce more accurate and reliable results. With more data, we also plan to apply the proposed algorithms to the network level emphasizing the time of day, day of week, and seasonal variations of travel times.