TRR
JOURNAL OF THE TRANSPORTATION RESEARCH BOARD

Research Article

Transportation Research Record
1–12

DOI: 10.1177/0361198120912424
journals.sagepub.com/home/trr
SAGE

Data Infrastructure for Connected Vehicle Applications

Xingmin Wang[1], Shengyin Shen[2], Debra Bezzina[2], James R. Sayer[2], Henry X. Liu[1,2], and Yiheng Feng[2]

Abstract

Ann Arbor Connected Vehicle Test Environment (AACVTE) is the world's largest operational, real-world deployment of connected vehicles (CVs) and connected infrastructure, with over 2,500 vehicles and 74 infrastructure sites, including intersections, midblocks, and highway ramps. The AACVTE generates a massive amount of data on a scale not seen in the traditional transportation systems, which provides a unique opportunity for developing a wide range of connected vehicle (CV) applications. This paper introduces a data infrastructure that processes the CV data and provides interfaces to support real-time or near real-time CV applications. There are three major components of the data infrastructure: data receiving, data preprocessing, and visualization including the performance measurements generation. The data processing algorithms include signal phasing and timing (SPaT) data compression, lane phase mapping identification, trajectory data map matching, and global positioning system (GPS) coordinates conversion. Simple performance measures are derived from the processed data, including the time–space diagram, vehicle delay, and observed queue length. Finally, a web-based interface is designed to visualize the data. A list of potential CV applications including traffic state estimation, traffic control, and safety, which can be built on this connected data infrastructure is discussed.

Connected vehicle (CV) technology can significantly improve safety, mobility, system efficiency and reduce fuel consumption and emissions, and has thus attracted worldwide attention. In the U.S., several large-scale real-world deployments of CV systems have been carried out in the past few years. The U.S. Department of Transportation (U.S. DOT) initiated the Safety Pilot Model Deployment (SPMD) project to validate the dedicated short-range communication (DSRC) technology for vehicle-to-vehicle (V2V) and vehicle-to-infrastructure (V2I) safety applications in 2011 (*1*). Later in 2016, U.S. DOT launched three more CV pilot programs in New York City, Tampa, Florida, and Wyoming as national efforts to deploy, test, and operationalize CV-based transportation systems (*2*). In addition to U.S. DOT, state and local transportation agencies and universities are also applying and testing CV technology for a wide range of applications (e.g., the City of Denver, Utah DOT, and Maricopa County, AZ).

CV systems generate a massive amount of data on a scale not seen in the traditional transportation systems, from both on-board units (OBUs) and road-side units (RSUs). The Society of Automotive Engineers (SAE) defines the formats and information contained in all CV data types in the SAE J2735_201603 standard. Different CV applications require subsets of data based on both time and spatial contexts. For example, traffic management applications may need network-wide, but less time-sensitive data, and safety-related applications require real-time, but localized data within a small region. As a result, a data infrastructure is needed to process the raw data and provide interfaces to a variety of CV applications.

The main objective of this paper is to develop such a data infrastructure for CV systems to support real-time or near real-time CV applications. The data infrastructure is built on the Ann Arbor Connected Vehicle Test Environment (AACVTE), which will be introduced in the next section. Three important CV data types are integrated: basic safety messages (BSMs) from connected vehicles (CVs), signal phasing and timing (SPaT) from signal controllers, as well as static map data to provide geofencing and map-matching functions. There are three major components of the data infrastructure: data

[1]Department of Civil and Environmental Engineering, University of Michigan, Ann Arbor, MI
[2]University of Michigan Transportation Research Institute, Ann Arbor, MI

Corresponding Author:
Yiheng Feng, yhfeng@umich.edu

processing, performance measurement generation, and visualization. A set of algorithms are developed in the data processing component to convert the raw data to processed data. The algorithms include SPaT data compression, lane phase mapping identification, trajectory data map matching, and global positioning system (GPS) coordinates conversion. Simple performance measures are derived from the processed data, including the time–space diagram, vehicle delay, and observed queue length. Finally, a web-based interface is designed to visualize the data. After introducing the data infrastructure, a list of potential CV applications that can be built on this infrastructure is discussed. This paper gives one implementation example detailing how to build a CV data infrastructure.

The rest of the paper is organized as follows. The next section gives a brief introduction to the AACVTE project and CV data. The following section describes the data infrastructure, including data processing algorithms, performance measures generation, and visualization interface. The final two sections discuss the potential CV applications and conclude the paper.

AACVTE Project

AACVTE is the world's largest operational, real-world deployment of connected vehicles and connected infrastructure, built on the existing Ann Arbor SPMD project. Over 2,500 vehicles (e.g., passenger vehicles, buses, and trucks) are equipped with CV devices including passenger vehicles, commercial trucks, and buses. The vehicles are equipped with one of the two types of DSRC on-board units: aftermarket safety device (ASD) or vehicle awareness device (VAD). The aftermath safety devices (ASDs) enable V2V communication for safety applications, including forward collision warning (FCW) and intersection movement assist (IMA), among others. The vehicle awareness devices (VADs) simply transmit the basic safety message (BSM) to seed the environment with connected vehicles to maximize interactions for optimal technology and application development. Owing to the number of different makes and models that comprise the AACVTE fleet, neither ASDs nor VADs are connected to the vehicle controller area network (CAN). In addition to the ASD or VAD, the vehicles are equipped with a global navigation satellite system (GNSS) antenna and a DSRC antenna. The GNSS antenna is mounted exterior to the vehicle to achieve the best GPS performance. The exact mounting location varies, depending on the vehicle type. Both devices have implemented untethered dead reckoning to further enhance the performance.

Furthermore, the SPMD infrastructure footprint has grown from 25 sites to over 70 infrastructure locations equipped with RSUs. Figure 1 shows the AACVTE deployment area and infrastructure sites. The sites include:

- 2 Curve speed warning sites (4 RSUs),
- 4 Pedestrian mid-block crosswalks (4 RSUs),
- 60 Intersections (5 at freeway entrance/exit ramps),
- 1 Roundabout,
- 5 staging/testing sites

The RSUs located at the 60 intersections are connected to the traffic signal controllers and broadcast SPaT messages and MAP messages to support infrastructure-related applications such as red light violation warning (RLVW) and curve speed warning (CSW). All vehicle and infrastructure communications follow national and international standards including SAE J2735, SAE J2945, IEEE 1609.2, IEEE 1609.3, and so forth, and, with the exception of the five staging/testing sites, utilize production security certificates to ensure security and privacy. Through the City's optical fiber network, both BSMs and SPaT data are forwarded to the data server located at the University of Michigan Transportation Research Institute (UMTRI) in real-time.

Data Infrastructure

The framework of the data infrastructure is shown in Figure 2. After receiving and decoding, the SPaT and BSM data are combined with map data for pre-processing. The pre-processing step generates event-based signal data (i.e., the timestamp at which the signal phase state changes and its duration) and matched vehicle trajectories, from which the time–space diagrams can be constructed.

Receiving the Data

The real-time CV data are forwarded from the infrastructure side (i.e., the road-side unit [RSU]) through the IPv6 network. The SPaT data are directly collected from the signal controllers, and the BSM data only contains trajectory segments if the vehicles are within the communication ranges of the RSUs. Although MAP messages are broadcast through the RSU, we use data from OpenStreetMap for map-matching algorithms, because the MAP data is static and does not change over time (*3*). An extraction tool is developed to extract vehicle movements at the intersection (e.g., eastbound left turn) from the original map data.

As the raw BSM data follows the SAE J2735 standard and are encoded with the UPER (i.e., unaligned packed encoding rule) rules, a message decoder is developed to

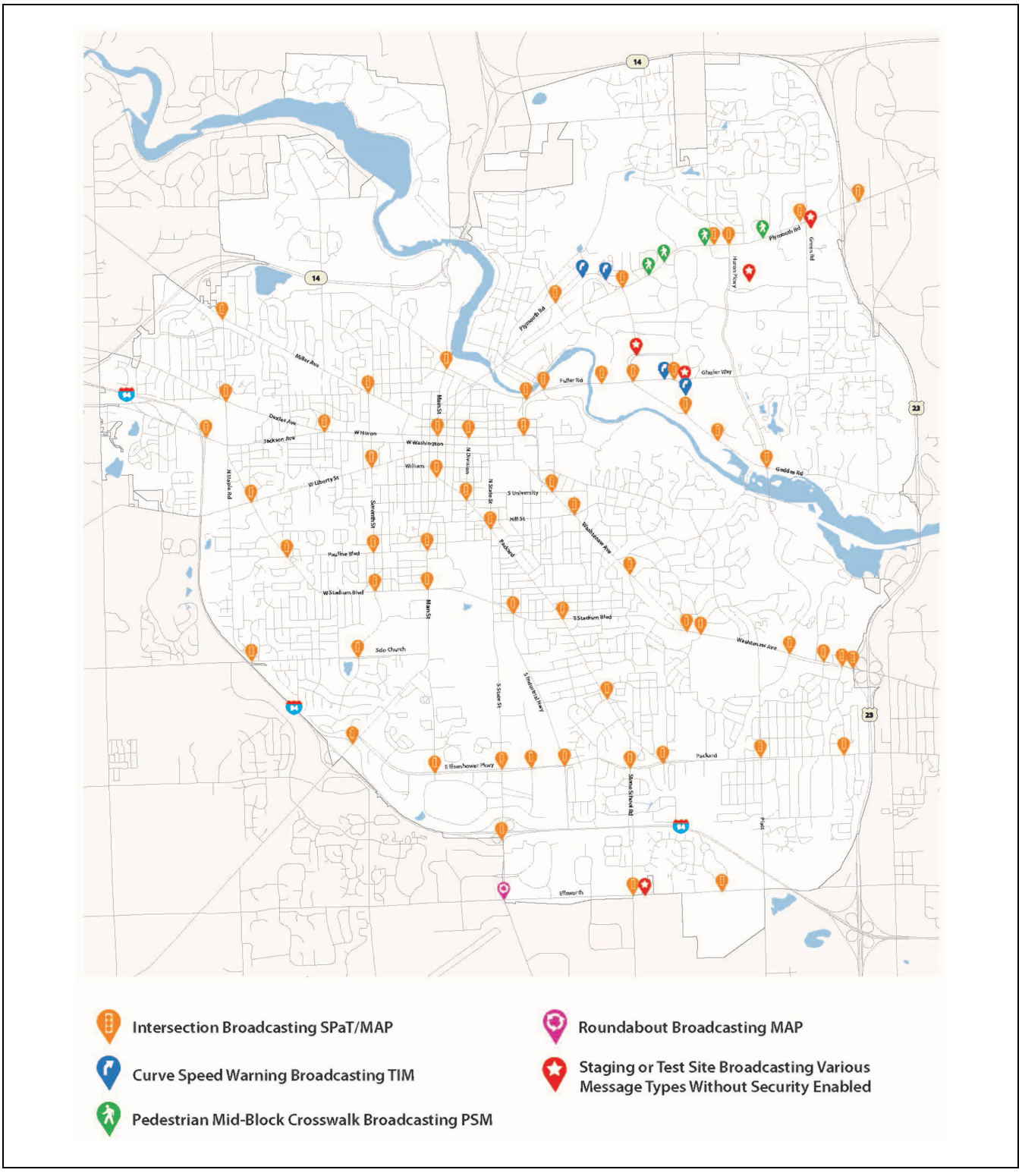

Figure 1. Ann Arbor connected vehicle test environment infrastructure deployment locations.

transfer the hex strings into C++ class objects for further processing. To verify whether the latency and frequency of the received BSM are within accepted ranges, timestamps of when the BSM/SPaT are received at the server and timestamps for when they are generated are compared. The result is shown in Table 1. The average

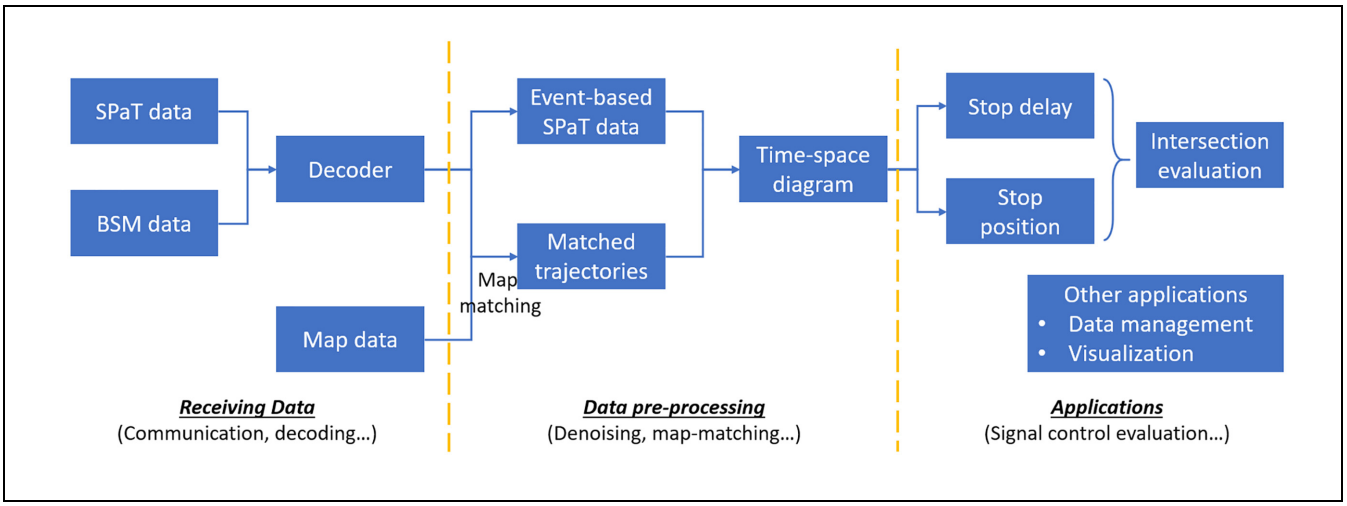

Figure 2. Framework of the connected vehicle data infrastructure.

Table 1. Frequency and Communication Delay of BSM/SPaT Data

Data type	BSM	SPaT
Resolution	0.1 s	0.1 s
Average communication delay	0.265 s	0.890 s
Standard deviation of delay	1.36 s	0.600 s

Note: BSM = basic safety message; SPaT = signal phasing and timing.

communication delays of both BSM and SPaT are within 1 s, and the frequency is 10 Hz. Note that this data infrastructure is not designed for safety-critical applications, which should directly rely on V2V communications. The delay presented in Table 1 is sufficiently low for most real-time infrastructure applications (e.g., adaptive signal control).

Data Pre-processing

Signal Phase Mapping Identification. The SPaT data sent from the signal controller includes a timestamp and the states of all signal phases at the intersection. However, the mapping relationships between the signal phases and vehicle movements (e.g., left-turn and through movement of four directions) are not known. Moreover, intersections in the real world can be very diverse, they have different numbers of approaches and the left-turn movements are not always protected, and so forth. Usually, the signal phase mapping information is obtained from field observations or directly from local transportation agencies. However, this method cannot dynamically update the mapping relationships if the configurations of traffic signals or the road geometry are updated. As a result, an automatic identification approach is proposed below.

As most vehicles do not pass the intersection when the signal is red, historical SPaT and trajectory data are aggregated to estimate the phase mapping relationships by minimizing the ratio of vehicles passing through the intersection during the red time. Let $\mathbf{s} = [s_1, s_2, \ldots, s_n]^T$, $\{s_i\} = \{1, 2, \ldots, n\}$ be the mapping relationship between the signal state and the real-world vehicle movement and $s_i = j$ represent the ith phase of the signal, corresponding to movement j. An optimization problem is formulated using aggregated historical data to automatically infer the phase mapping relationship. As the mapping relationships remain unchanged, we can aggregate as much data as needed. Usually, one hundred trajectories per movement is sufficient for an accurate estimation. Firstly, the map matching algorithm, introduced in the next section locates the vehicle trajectory to the corresponding movement and the time at which the vehicle passed the intersection is calculated. Then, the number of vehicles of each movement passing the intersection during the red time given the mapping relationship $\mathbf{s}$ is calculated. Let $N_i^{\text{red}}(s_i)$ be the number of vehicles that violate the traffic signal of phase i, then the optimization problem can be written as:

$$\mathbf{s} = \arg\min \sum_i N_i^{\text{red}}(s_i) \tag{1}$$

subject to $\{s_1, \ldots, s_n\} = \{1, 2, \ldots, n\}$. The results of the phase mapping can be verified by plotting the time–space diagram and checking the violations.

Trajectory Data Map-Matching. Map-matching is an important pre-processing step that connects BSM data with road geometry. The BSM data is collected by the RSUs

installed at intersections. It contains the vehicle identification (ID), timestamps, and the GPS coordinates of the vehicle that can be used to construct the trajectory.

A hidden Markov model (HMM) is applied to match the trajectory to the corresponding movement. This HMM map-matching model proposed by Newson and Krumm can find the most likely route of the trajectory considering both the distance of the trajectory to the route and the path feasibility (*4*). In our scenario, as the trajectory data are collected close to intersections, we only need to match the trajectory to its corresponding movement. Currently, the trajectories are not matched to specific lanes owing to the GPS accuracy.

Assume the trajectory of a vehicle is denoted by $\mathbf{x} = [x_1, x_2, \ldots, x_n]^T$ and each element of the vector is a GPS location with a latitude and longitude. Let $\mathbf{r}_i = [r_{i1}, r_{i2}, \ldots, r_{im}]^T$ be the GPS coordinates of the *ith* movement of the intersection given by the map data. $\mathcal{R}$ is the movement set of the intersection, so that $\mathbf{r}_i \in \mathcal{R}$. Then, the corresponding movement of the trajectory is obtained by solving the following optimization problem:

$$\mathbf{r}(x) = \arg\max_{\mathbf{r}_i \in \mathcal{R}} \prod_{q=1}^{n} \mathbb{P}(x_q|\mathbf{r}_i) \cdot \prod_{q=1}^{n-1} \mathbb{P}((x_q, x_{q+1})|\mathbf{r}_i) \quad (2)$$

where $\mathbb{P}(x_q|\mathbf{r}_i)$ is the probability that the GPS point x_q belongs to the movement $\mathbf{r}_i$ and $\mathbb{P}((x_q, x_{q+1})|\mathbf{r}_i)$ is the transfer probability of the trajectory point transferring from x_q to x_{q+1} given trajectory $\mathbf{x}$ belonging to the movement $\mathbf{r}_i$.

For $\mathbb{P}(x_q|\mathbf{r}_i)$, we assume the errors coming from the GPS receiver follow a Gaussian distribution with the probability:

$$\mathbb{P}(x_q|\mathbf{r}_i) = \frac{1}{\sqrt{2\pi}\sigma} \exp(-|d(x_q, \mathbf{r}_i)|^2/(2\sigma^2)) \quad \forall q \quad (3)$$

where $d(x_q, \mathbf{r}_i)$ is the distance between the point x_q and the movement $\mathbf{r}_i$.

The transfer probability aims at capturing the path feasibility (*4*). In our case, the path is always feasible because the "feasible" movements have already been extracted from the map. However, the headings of the vehicle trajectory and the roadway need to be checked. If only the distance between the trajectory and the movement is considered, we cannot distinguish, for example, the through movements of the opposite direction, as vehicles traveling in the opposite direction have a very close set of GPS coordinates, but in a reversed sequence. Therefore, the transfer probability is needed to match the direction or heading of the trajectory and the roadway (i.e., movement). The transfer probability is written as:

$$\mathbb{P}((x_q, x_{q+1})|\mathbf{r}_i) = \frac{1}{\beta} \exp(-\phi((x_q, x_{q+1}), \mathbf{r}_i)/\beta) \quad (4)$$

where $\phi((x_q, x_{q+1}), \mathbf{r}_i) \in [0, \pi]$ is the angle between the vector (x_q, x_{q+1}) and $\mathbf{r}_i$. Similar to the previously published literature, it is assumed that the transfer probability follows an exponential distribution as the historical data showed that the heading between the roads and the trajectories can be approximated by an exponential distribution (*4*). As $\mathbf{r}_i$ is composed of piece-wise line segments, the $\phi((x_q, x_{q+1}), \mathbf{r}_i)$ can be calculated as the angle between the vector (x_q, x_{q+1}) and the closest line segment of $\mathbf{r}_i$ to x_q.

By substituting Equations 3 and 4 into Equation 2 we can get the final equation for the trajectory map-match. The variance of the Gaussian distribution σ and the parameter of the exponential distribution β determines the weights of these two factors.

Figure 3 shows the result of map-matching of one day's data at the intersection of Plymouth Rd and Green Rd, Ann Arbor. The blue lines and blue dots are the geometric coordinates of the road and the red lines are matched trajectories. Each subplot is a phase or movement of the intersection. The number in the subtitle is the number of matched trajectories. There are still some outliers in the figure as the map-matching model can only match the trajectory to the most likely movement among all of the movements. The trajectory needs to be further checked before final use. Similar to the signal phase mapping, the result of map matching can be further verified with time–space diagrams.

Distance to the Intersection. After grouping the trajectory data to the corresponding movements, the original GPS coordinates are converted to the distance to the intersection. More accurately, 2-D GPS coordinates are converted to 1-D intrinsic coordinates and the closest point of the trajectory to the intersection is set as the zero point of distance.

Let $\mathbf{x} = [x_1, \ldots, x_n]^T$ be the GPS coordinates of a trajectory in which each element x_i is the GPS coordinates. y is the GPS coordinates of the intersection center from the map data. The first step is to find the closest point to the intersection as the zero point of the trajectory. The trajectory can be seen as a set of piece-wise line segments $\mathcal{X} = \{\hat{\mathbf{x}}_i = (x_i, x_{i+1})\}, i = 1, \ldots, n-1$, which is the linear interpolation of the trajectory. Let $d(y, \hat{\mathbf{x}}_i)$ be the distance between the point y and the line segment $\hat{\mathbf{x}}_i$ and x_i^* is the point belonging to $\hat{\mathbf{x}}_i$ and gives the minimum distance to the point y. Then, the zero point of the trajectory is given by:

$$i = \arg\min_{k=1,2\ldots,n-1} d(y, \hat{\mathbf{x}}_k) \quad x_i^* = \arg\min_{x \in \hat{\mathbf{x}}_i} d(x, y) \quad (5)$$

where x_i^* is the closest point from the intersection to the interpolated trajectory. Put this point to the original

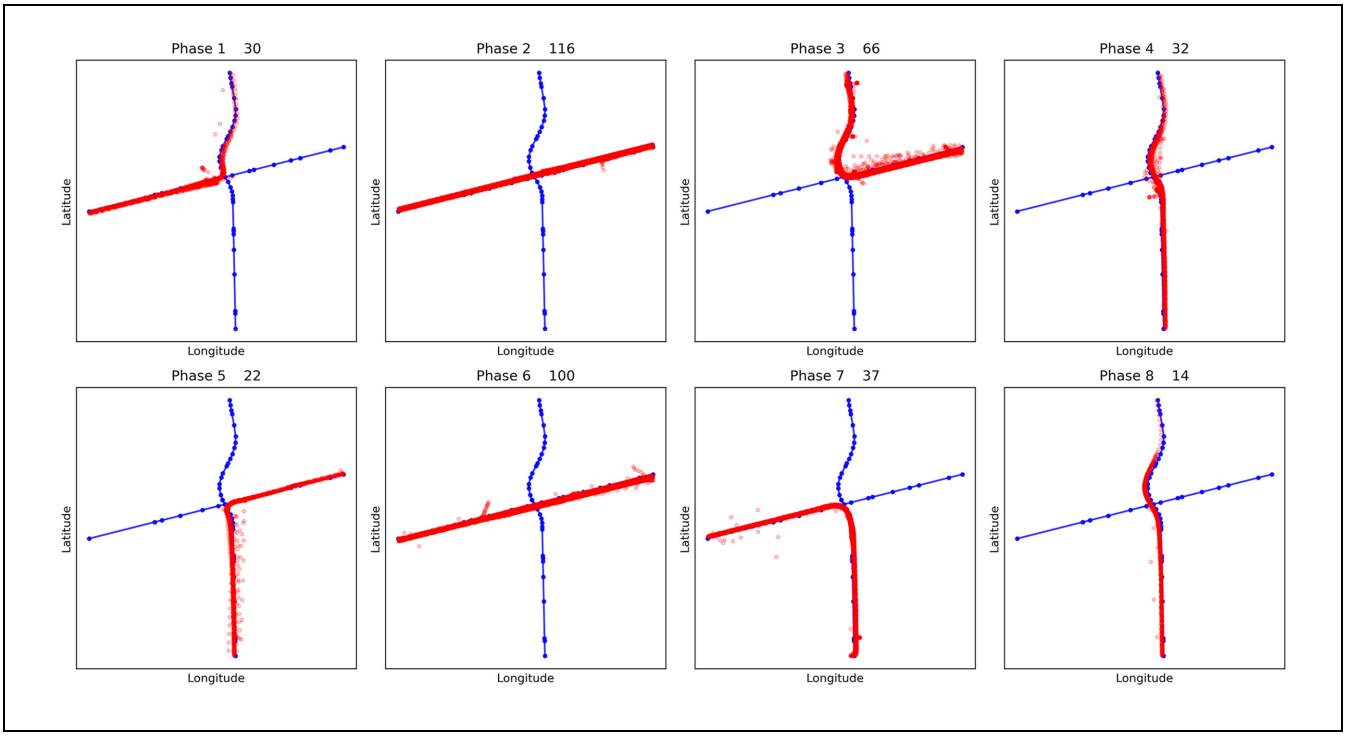

Figure 3. Trajectory map-matching results.

vector GPS coordinates of the intersection and a new vector with an additional point x_i^* is constructed:

$$\mathbf{x}' = [x_1, \ldots x_i, x_i^*, x_{i+1}, \ldots, x_n]^T \quad (6)$$

Then, the distance of each element x_k to the center of the intersection can be expressed by:

$$\tilde{x}_k = \sum_{m=1}^{k-1} d(x_m, x_{m+1}) - \sum_{m=1}^{i-1} d(x_m, x_{m+1}) - d(x_i, x_i^*) \quad (7)$$

where $d(x, y)$ gives the distance between two GPS coordinates. In this way, each of the original GPS coordinates to the distance to the intersection can be calculated. The negative value indicates that the trajectory is from the upstream of the intersection, and a positive value indicates downstream.

Time–Space Diagram Generation. The time–space diagram is generated by combining the SPaT data and the matched trajectory data. The time–space diagram provides intuitive information about the traffic state, such as whether the movement is over-saturated. Owing to the low penetration rate of CV, the trajectories from multiple cycles are aggregated to plot the time–space diagram. This is based on the assumption that the traffic state at the same time of day (TOD) is relatively stationary. Different times of day (TODs) are needed for the entire day to make the traffic state within one TOD relatively stable. Although the aggregated data cannot be applied to estimate the real-time traffic state, it can still be used to estimate long-term performance. Readers can refer to previously published studies for estimation of the different traffic measures, including the penetration rate, queue length, and volume using aggregated trajectory data (*5–7*).

The City of Ann Arbor has implemented both fixed-time and actuated/adaptive signal plans at a different intersection. For fixed-time intersections, to draw the time–space diagram, we only need to shift the trajectory by integer numbers of cycle lengths. For the adaptive or actuate signal timing plans with variant cycle lengths, we draw the time–space diagram by shifting the trajectory temporally to make all trajectories share the same cycle start time (e.g., green start). This can be easily achieved by subtracting the timestamps of the trajectory using the green starting time of the cycle in which the trajectory passes the intersection. Figure 4 shows the time–space diagram generated from all CV trajectories that passed the intersection of Maiden Lane and Fuller in Ann Arbor during the evening peak hour between 5:00 and 7:00 p.m. from 01/06/2020 to 01/10/2020 (five weekdays). Each sub-figure represents aggregated CV trajectories of each signal phase in all five days. It is assumed that traffic demands and patterns during the weekday morning peak hours are similar. Phases 2, 4, 6, and 8 are through

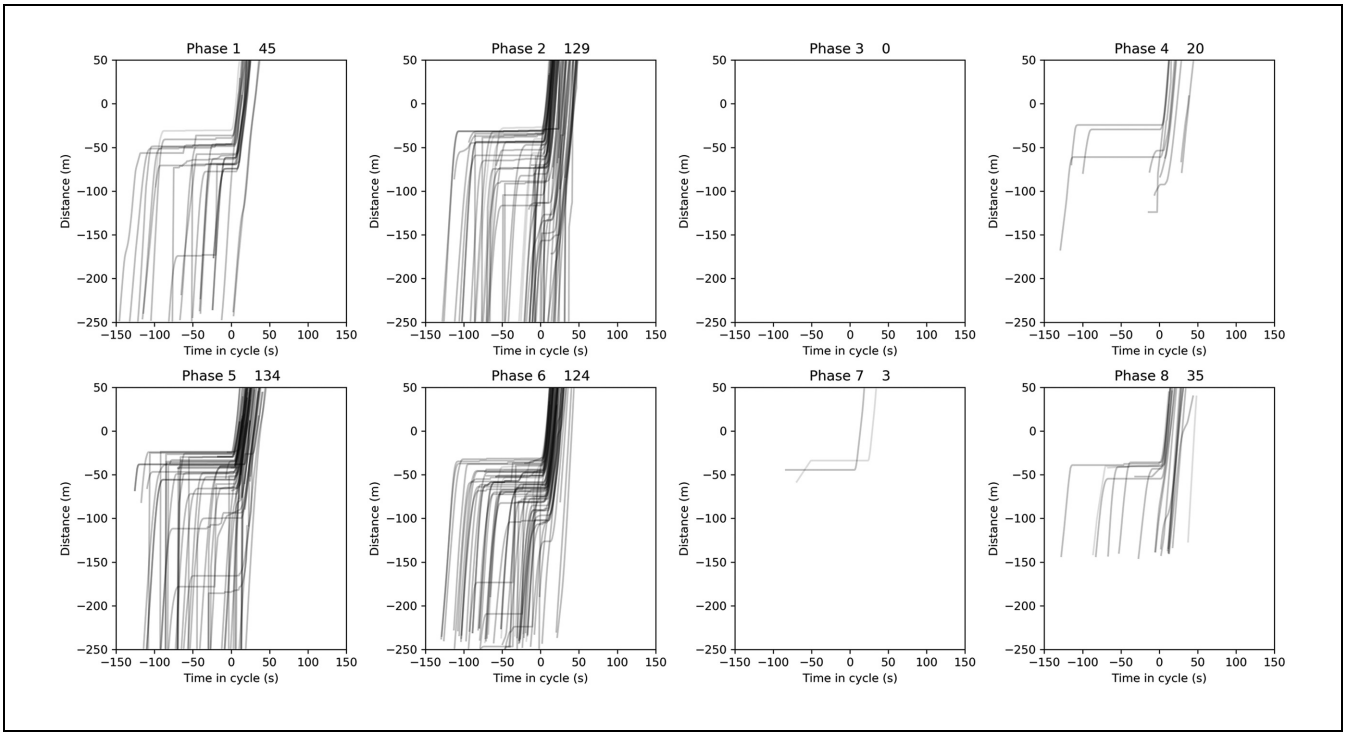

Figure 4. Time–space diagrams of Maiden Lane and Fuller intersection.

movements, and the others are left-turn movements. As the trajectory only shares the same green start time, only the departure shockwave has valid physical meaning. The departure shockwave shows the queue dissipation pattern and can also be applied to estimate the shockwave speed and saturation flow rate (*8*). In addition, we can also shift the trajectory to make them have the same red-start time so that the arrival patterns at the red signal can be obtained.

Performance Measurement and Visualization

Delay and Stop Position Measures. Two basic intersection performance measures are derived based on the time–space diagram: vehicle stop delay and stop position, to evaluate the performance of the traffic signal control. The delay scatters and the stop position scatters are shown in Figures 5 and 6. The data also comes from 01/06/2020 to 01/10/2020 at the intersection of Maiden Lane and Fuller in Ann Arbor. Each subplot is a phase/movement of the intersection and each blue dot stands for one vehicle. The x-axis is the time of day (0–24 h), and the y-axis is the delay in seconds and the stop position in meters correspondingly.

For the stop delay scatters, vehicle stop delays are calculated by summarizing the duration of time when the speed of the trajectory is less than the threshold (2 m/s). Compared with the total delay estimated by subtracting the actual travel time by free-flow travel time, the stop delay estimation method is more robust without knowing the accurate free-flow speed and can directly quantify the waiting time caused by the intersection. For the stop position scatters, the stop positions are estimated as the maximum distance to the intersection at which the vehicle comes to a complete stop. The stop distance is set to zero if the vehicle passes the intersection without stopping. It should be noted that the zero value of the y-axis is not exactly the location of the stop bar but the center of the intersection, therefore the stop positions (i.e., blue points) start from around 20 m, which is the distance from the stop bar to the intersection center. The stop positions reflect the queue lengths of the movement, which are the inputs for many traffic management applications.

Both the stop delay and stop position scatters can be used to measure the performance of the intersection. For example, if there is quite a proportion of vehicles waiting over the average cycle length, then the movement is oversaturated. They can also be used to evaluate the balance of green split among movements.

Data Visualization Interface. A web-based interface was designed and developed to manage the data and use it as a visualization tool. Figure 7 shows the home page of the website (http://aacvlive.umtri.umich.edu/home.html). The website displays real-time information, including the

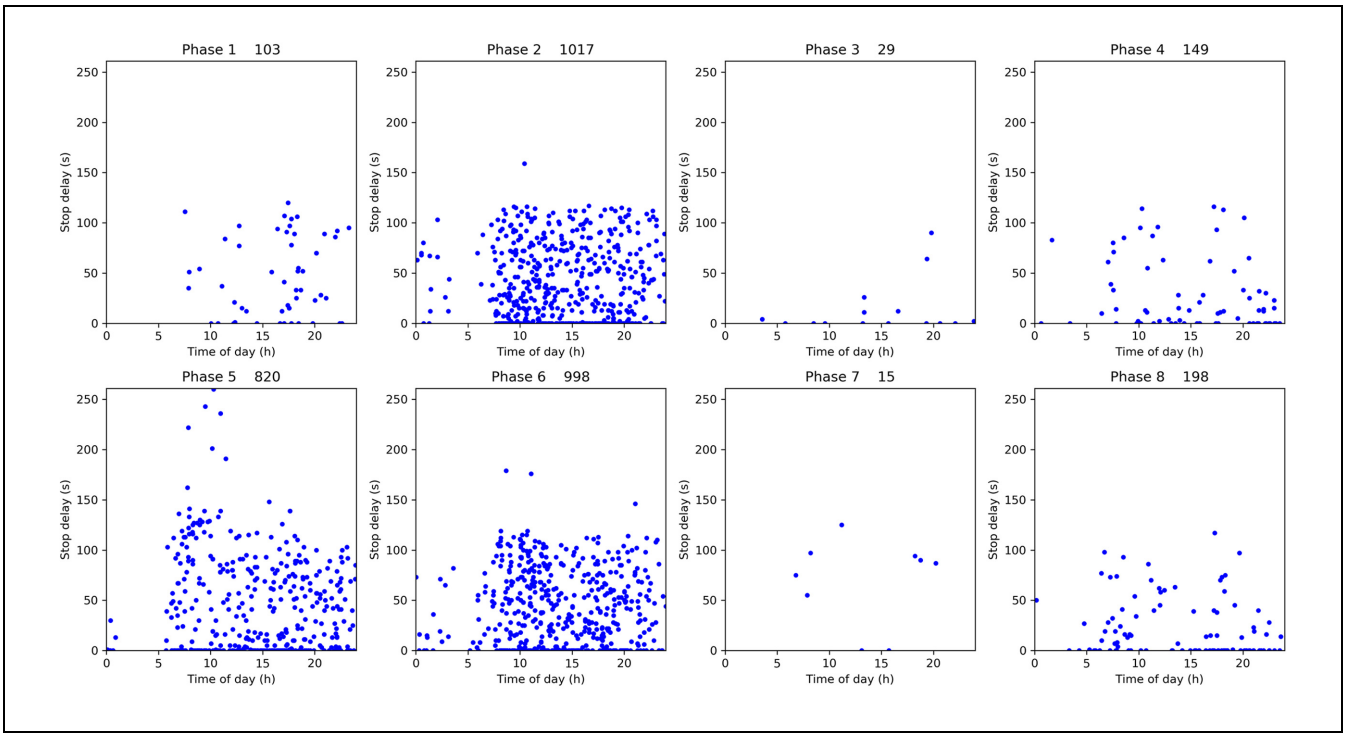

Figure 5. Stop delay scatters of Maiden Lane and Fuller intersection.

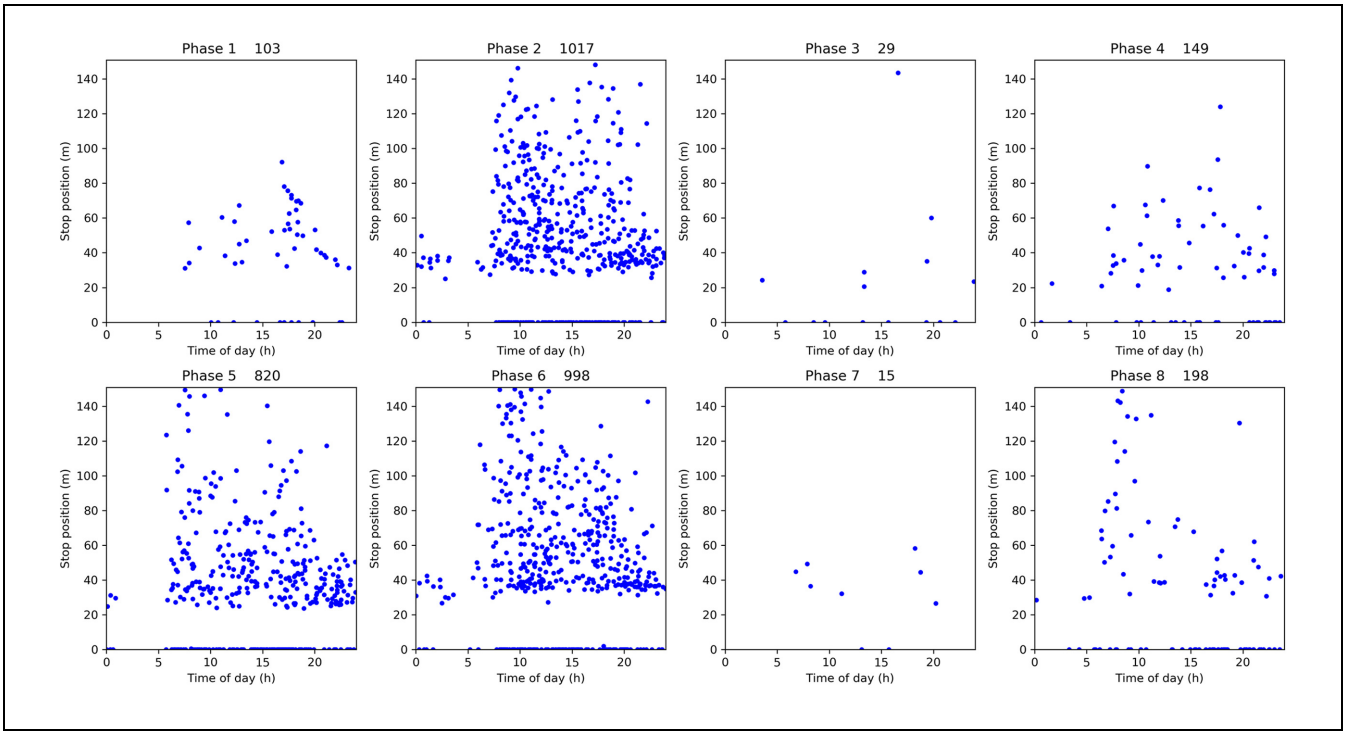

Figure 6. Stop position scatters of Maiden Lane and Fuller intersection.

CV trajectory data and traffic signal data. Each blue dot on the map represents a CV in the vicinity of an RSU. Each red mark represents one RSU at the intersection. Real-time signal states are shown on the right after clicking the intersection marks. Basic intersection statistics, including the number of vehicles, average stop delay, and

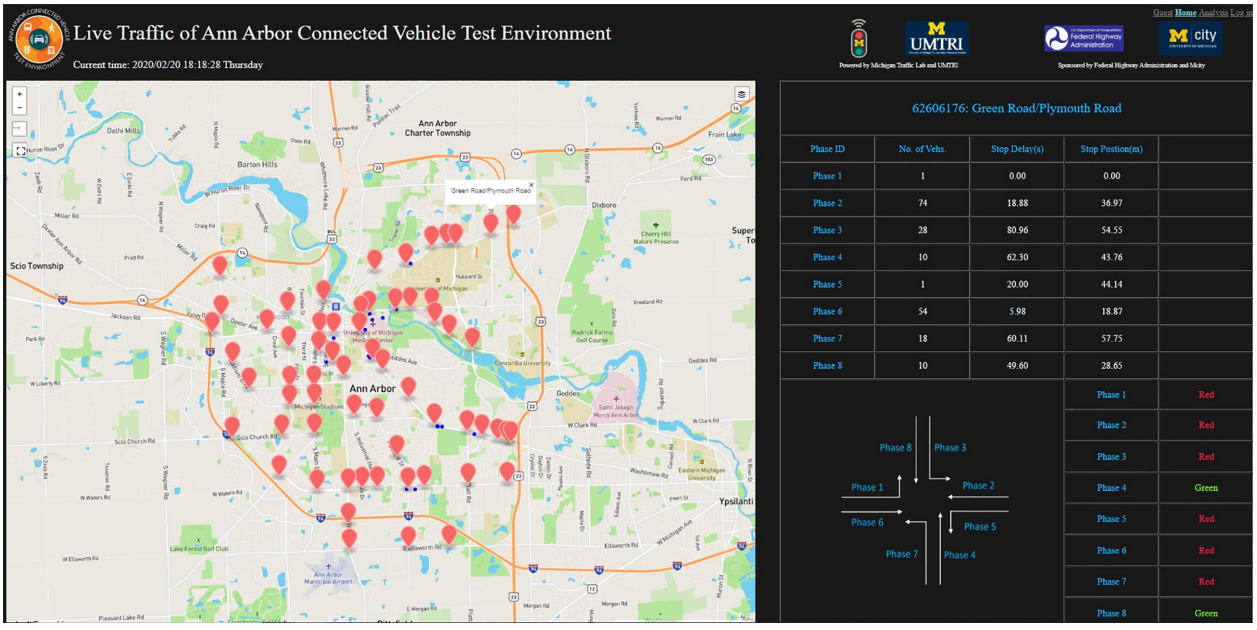

Figure 7. Ann Arbor connected vehicle test environment live traffic with basic statistics and diagrams.

average stop position, are summarized for each movement. The website can also plot the time–space diagram and delay diagram of each movement. This website is also embedded with a data query function through which the users can acquire historical data and diagrams. All of these functions make this website a powerful visualization and evaluation tool both for researches and traffic engineers at local transportation agencies.

Potential Applications

The main purpose of developing such a data infrastructure is to serve as a pre-step in developing a variety of vehicle trajectory based mobility and safety applications. For example, the generated time-space diagrams provide direct information on the intersection performance measures including vehicle delay, queue length, and numbers of stops, which can be used in traffic control applications. Meanwhile, by matching trajectories temporally and spatially, the interactions between different vehicles (e.g., time to collision) can be captured, which can be used as surrogate measures for safety applications. If the CV penetration rate is low, trajectory data need to be aggregated for a certain time period to provide sufficient information. If the CV penetration rate increases in the future, real-time applications will be supported. However, the design of the data infrastructure can remain unchanged. In the following, some representative applications that can be implemented based on the proposed data infrastructure are introduced.

Traffic State Estimation and Traffic Signal Control

Traffic state estimation provides necessary information for both real-time signal control and long-term traffic management. As a new data source, CV trajectory data can be used to estimate traffic states such as traffic volumes, queue lengths, or to directly construct the trajectory of unequipped vehicles. Traffic state estimation using CV data faces different challenges compared with traditional loop detector data. For loop detector data, the total count of vehicles at specific locations can be obtained. From CV data, however, it is difficult to obtain information from the entire population, but a low proportion of all vehicles, also called the penetration rate. One of the biggest challenges using CV data is how to estimate the traffic state with a low and unknown penetration rate.

Traffic state estimation using CV data has drawn significant attention over the years. For urban traffic networks, the estimation methods can be roughly divided into two categories, probability-based methods and shockwave-based methods. For the probability-based methods, the basic idea is to treat each CV as an event and use maximum likelihood methods to estimate the unknown parameters given the CV data. Comert and Cetin developed different models to estimate the cycle-by-cycle queue length at isolated intersections based on a known CV penetration rate and queue length distribution (*9*, *10*). However, the real-world penetration rate is unknown in most of the cases, Wong et al. proposed an

unbiased method to estimate the penetration rate and Zhao et al. estimated both the penetration rate and the queue length based on the assumption that CV data have the same queue lengths distribution as regular vehicles (*6*, *7*). In addition to the queue length estimation, Zheng and Liu used the EM algorithm to estimate the traffic volumes at a signalized intersection, which assumed that the vehicles' arrival followed a Poisson process (*5*). For the shockwave-based methods, the basic idea is to detect the shockwave in the time–space diagram and use the kinematic wave theory to estimate the traffic state. Cheng et al. used a classification method to detect the shockwave in the time–space diagram and used the shockwaves to estimate the cycle-by-cycle queue length (*11*). Ban and Hao used travel time to construct the shockwaves and estimate the queue lengths and the signal timing plan (*12*, *13*). Wang et al. used the RANSAC algorithm to detect the departure shockwave and estimated the shockwave speed and saturation flow rate at signalized intersections (*8*). Vasudevan et al. used sparse BSMs to predict the congestion state with a high temporal and spatial resolution (*14*).

There are also other models proposed to estimate the traffic state using the CV. Unlike the loop detector data, which is more convenient to use a Eulerian expression, the CV data can be directly used in the Lagrangian coordinates (*15*). Zheng et al. proposed a stochastic traffic model in Lagrangian coordinates and used the Kalman filter to construct the complete trajectories using a fusion of detector and CV data (*16*). To deal with the nonlinear dynamic model, Xie et al. proposed a generic trajectory reconstruction framework at a signalized intersection using a particle filter (*17*).

One direct application of traffic state estimation is traffic signal optimization, in which the objective function is usually formulated as one or more traffic states such as the queue length, delay, and travel time. Goodall et al. proposed a predictive microscopic simulation algorithm which used the position, heading, and speed from CVs to predict the traffic condition and optimize the traffic signals (*18*). Feng et al. used the location and speed of CV data and proposed a two-level optimization based on dynamic programming to allocate the green time (*19*). Feng et al. also proposed a model to estimate the delay and optimize the signal control in a low penetration rate CV environment (*20*).

Safety Applications

Leveraging the BSM data from CVs under different market penetration rates (MPRs), we can estimate the surrogate traffic safety performance from two perspectives (*21*): 1) individual vehicle dynamics based; and 2) inter-vehicle proximity based. The first set of safety measures may include vehicle speed, acceleration, jerk, and stop frequency, which can be used to infer the driver's states and his/her decision-making process. Mixed models can be built to identify driver behavior factors that are related to safety. The BSMs can be further fused with SPaT and MAP data to analyze the driver's behavior at locations of high interest. The derived information can be employed to estimate the likelihood of an individual vehicle's involvement in conflicts with other vehicles. For example, studies from Arvin and Kamrani quantified and used about 30 measures of driving volatility by using speed, longitudinal and lateral acceleration, and yaw-rate, extracted from BSMs at signalized intersections(*22*, *23*). These volatilities were then used to explain crash frequencies at intersections. It was found that erratic longitudinal/lateral movements increased the risk of crashes. Inter-vehicle safety measures such as time-to-collision (TTC), post-encroachment time (PET) can be imported to the analysis tools such as Pu et al. to estimate the occurrence of potential traffic conflicts (*24*). Both temporal and spatial correlations of CV trajectories can be explored to find the interactions between different CVs. A new surrogate safety measure (SSM) named time to collision with disturbance (TTCD) was proposed in Xie et al. for risk identification (*25*). The new measure can achieve a higher Pearson's correlation coefficient with rear-end crash rate than other traditional SSMs. All of the aforementioned safety studies utilized BSMs from the Safety Pilot database, which was collected in the SPMD project before AACVTE.

Conclusions

AACVTE is the world's largest operational CV system and is collecting massive CV data from vehicles and infrastructure sites. This paper developed a data infrastructure using this data to support both safety and mobility CV applications.

The data infrastructure mainly includes three parts: data receiving, data pre-processing, and visualization including the performance measurements. In the data pre-processing part, an optimization problem was formulated, which can automatically infer the phase mapping relationship combining BSMs and SPaT. The timestamp and GPS location were used to construct the vehicle trajectory from BSMs, and a HMM was used to match the trajectory to the road network. Then, the trajectory with GPS coordinates was converted to the distance to the intersection. Time–space diagrams were generated by combining trajectory data and SPaT data. For the performance measurements, stop delay and stop position scatters were generated. Both diagrams can be used to evaluate the performance of intersection level signal control. For the visualization part, a website was developed

as the data visualization and management interface. The website displays real-time CV trajectories, traffic signal information, and basic statistics. Users can also query historical data through this website.

This data infrastructure provides a solid foundation in developing further CV applications. Some of the applications are discussed, including traffic state estimation, traffic signal control, and safety measurement.

Acknowledgments

The authors would like to thank Federal Highway Administration (FHWA) at U.S. DOT and Mcity at the University of Michigan for financial support.

Author Contributions

The authors confirm contribution to the paper as follows: study conception and design: Debra Bezzina, James Sayer, Yiheng Feng, and Henry Liu; data collection: Xingmin Wang, Shengyin, and Yiheng Feng; analysis and interpretation of results: Xingmin Wang and Shengyin Shen; draft manuscript preparation: All authors. All authors reviewed the results and approved the final version of the manuscript.

Declaration of Conflicting Interests

The author(s) declared no potential conflicts of interest with respect to the research, authorship, and/or publication of this article.

Funding

The author(s) disclosed receipt of the following financial support for the research, authorship, and/or publication of this article: This research is partially supported by Federal Highway Administration (FHWA) at U.S. DOT, and Mcity at the University of Michigan.

References

1. Bezzina, D., and J. Sayer. *Safety Pilot Model Deployment: Test Conductor Team Report* (Report No DOT HS 812 171). National Highway Traffic Safety Administration, Washington, D.C., 2014.
2. U.S. Department of Transportation. Connected Vehicle Pilot Deployment Program. Technical Report. U.S. DOT, Washington, D. C., 2019. https://www.its.dot.gov/pilots.
3. OpenStreetMap. *Openstreetmap*. Technical Report. 2019. https://www.openstreetmap.org.
4. Newson, P., and J. Krumm. Hidden Markov Map Matching Through Noise and Sparseness. *Proc., 17th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems*. ACM, Seattle, Washington, 2009, pp. 336–343.
5. Zheng, J., and H. X. Liu. Estimating Traffic Volumes for Signalized Intersections using Connected Vehicle Data. *Transportation Research Part C: Emerging Technologies*, Vol. 79, 2017, pp. 347–362.
6. Zhao, Y., J. Zheng, W. Wong, X. Wang, Y. Meng, and H. X. Liu. Estimation of Queue Lengths, Probe Vehicle Penetration Rates, and Traffic Volumes at Signalized Intersections Using Probe Vehicle Trajectories. *Transportation Research Record: Journal of the Transportation Research Board*, 2019. 2673: 660–670.
7. Wong, W., S. Shen, Y. Zhao, and H. X. Liu. On the Estimation of Connected Vehicle Penetration Rate Based on Single-source Connected Vehicle Data. *Transportation Research Part B: Methodological*, Vol. 126, 2019, pp. 169–191.
8. Wang, X., J. Zheng, H. X. Liu, and W. Sun. Estimating Saturation Flow Rate for Signalized Intersection Using Trajectory Data. Presented at 98th Annual Meeting of the Transportation Research Board, Washington, D.C., 2019.
9. Comert, G., and M. Cetin. Queue Length Estimation from Probe Vehicle Location and the Impacts of Sample Size. *European Journal of Operational Research*, Vol. 197, No. 1, 2009, pp. 196–202.
10. Comert, G. Queue Length Estimation from Probe Vehicles at Isolated Intersections: Estimators for Primary Parameters. *European Journal of Operational Research*, Vol. 252, No. 2, 2016, pp. 502–521.
11. Cheng, Y., X. Qin, J. Jin, B. Ran, and J. Anderson. Cycle-by-cycle Queue Length Estimation for Signalized Intersections using Sampled Trajectory Data. *Transportation Research Record: Journal of the Transportation Research Board*, 2011. 2257: 87–94.
12. Ban, X. J., P. Hao, and Z. Sun. Real Time Queue Length Estimation for Signalized Intersections Using Travel Times from Mobile Sensors. *Transportation Research Part C: Emerging Technologies*, Vol. 19, No. 6, 2011, pp. 1133–1156.
13. Hao, P., X. Ban, K. P. Bennett, Q. Ji, and Z. Sun. Signal Timing Estimation Using Sample Intersection Travel Times. *IEEE Transactions on Intelligent Transportation Systems* Vol. 13, No. 2, 2012, pp. 792–804.
14. Vasudevan, M., D. Negron, M. Feltz, J. Mallette, and K. Wunderlich. Predicting Congestion States from Basic Safety Messages by Using Big-data Graph Analytics. *Transportation Research Record: Journal of the Transportation Research Board*, 2015. 2500: 59–66.
15. Feng, S., X. Wang, H. Sun, Y. Zhang, and L. Li. A Better Understanding of Long-range Temporal Dependence of Traffic Flow Time Series. *Physica A: Statistical Mechanics and its Applications*, Vol. 492, 2018, pp. 639–650.
16. Zheng, F., S. E. Jabari, H. X. Liu, and D. Lin. Traffic State Estimation Using Stochastic Lagrangian Dynamics. *Transportation Research Part B: Methodological* Vol. 115, 2018, pp. 143–165.
17. Xie, X., H. van Lint, and A. Verbraeck. A Generic Data Assimilation Framework for Vehicle Trajectory Reconstruction on Signalized Urban Arterials Using Particle Filters. *Transportation Research Part C: Emerging Technologies* Vol. 92, 2018, pp. 364–391.
18. Goodall, N. J., B. L. Smith, and B. Park. Traffic Signal Control with Connected Vehicles. *Transportation Research Record: Journal of the Transportation Research Board*, 2013. 2381: 65–72.

19. Feng, Y., K. L. Head, S. Khoshmagham, and M. Zamanipour. A Real-time Adaptive Signal Control in a Connected Vehicle Environment. *Transportation Research Part C: Emerging Technologies*, Vol. 55, 2015, pp. 460–473.
20. Feng, Y., J. Zheng, and H. X. Liu. Real-time Detector-free Adaptive Signal Control with Low Penetration of connected vehicles. *Transportation Research Record: Journal of the Transportation Research Board*, 2018. 2672: 35–44.
21. Gettman, D., and L. Head. Surrogate Safety Measures from Traffic Simulation Models. *Transportation Research Record: Journal of the Transportation Research Board*, 2003. 1840: 104–115.
22. Arvin, R., M. Kamrani, and A. J. Khattak. How Instantaneous Driving Behavior Contributes to Crashes at Intersections: Extracting Useful Information from Connected Vehicle Message Data. *Accident Analysis & Prevention*, Vol. 127, 2019, pp. 118–133.
23. Kamrani, M., R. Arvin, and A. J. Khattak. Extracting Useful Information from Basic Safety Message Data: An Empirical Study of Driving Volatility Measures and Crash Frequency at Intersections. *Transportation Research Record: Journal of the Transportation Research Board*, 2018. 2672: 290–301.
24. Pu, L., and R. Joshi. *Surrogate Safety Assessment Model (SSAM)–Software User Manual*. Technical report, Turner-Fairbank Highway Research Center, McLean, VA, 2008.
25. Xie, K., D. Yang, K. Ozbay, and H. Yang. Use of Real-world Connected Vehicle Data in Identifying High-risk Locations Based on a New Surrogate Safety Measure. *Accident Analysis & Prevention*, 2019; 125: 311–319.

The views presented in this paper are those of the authors alone.