DEVELOPMENT OF A CLOUD-BASED TRAFFIC DIAGNOSIS AND MANAGEMENT LABORATORY BASED ON HIGH-COVERAGE ALPR

by

Weijie Tan

A Dissertation Submitted in

Partial Fulfillment of the

Requirements for the Degree of

Doctor of Philosophy

in Engineering

at

The University of Wisconsin-Milwaukee

May 2021

ABSTRACT

DEVELOPMENT OF A CLOUD-BASED TRAFFIC DIAGNOSIS AND MANAGEMENT LABORATORY BASED ON HIGH-COVERAGE ALPR

by

Weijie Tan

The University of Wisconsin-Milwaukee, 2021
Under the Supervision of Professor Yue Liu

As the traffic volume in urban areas continues to increase, traffic congestion results in longer travel time, lower reliability, and larger energy consumption. Traffic diagnosis and management strategies are generally considered the most cost-efficient approaches to reduce traffic congestion. Automated License Plate Reader (ALPR) is one of the most valuable data sources for traffic diagnosis and management. It outperforms many traditional methods of traffic data collection in terms of cost and accuracy. Many cities have deployed their ALRP systems in the past two decades. In those cities, traffic networks are highly covered by ALPRs. ALPRs produce a tremendous amount of data of moving vehicles at every moment. The information and insights hidden within the data can be a fortune for comprehensive traffic diagnosis and management. To fully take the advantage of ALPR data, more scalable and efficient data management, analysis algorithms, and visualization tools are required to process large-volume ALPRs data within reasonable time and budgets. An extensive body of research about ALPR data exists but mainly focuses on small-scale use of ALPRs on estimations of traffic flow characteristics. This dissertation plans to develop a traffic diagnosis and management laboratory utilizing high-coverage and large-scale ALPRs data.

The online laboratory consists of modules for traffic condition reconstruction, diagnosis-oriented visual analysis, and traffic control and management decision support. This research contributes to (1) Designing a comprehensive and extensible framework for traffic diagnosis and management with ALPR data; (2) Developing highly scalable and efficient algorithms for traffic condition reconstruction; (3) Implementing and designing advanced visualization tools for traffic condition monitoring and diagnosis; and (4) Exploring the capability of utilizing ALPR data to improve the performance of data-driven traffic control and management strategies.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

ALPR	Automatic License Plate Reader
AP	Access Point
API	Application Programming Interface
ATCS	Adaptive Traffic Control System
BFS	Breath-first search
CSV	Comma-separated Values
DFS	Depth First Search
EC2	Elastic Compute Cloud
FIFO	First-In, First-Out
GCE	Google Compute Engine
GIS	Geographic Information System
GPS	Global Positioning System
HCM	Highway Capacity Manual
HMM	Hidden Markov Model
IaaS	Infrastructure as a Service
ILD	Indictive Loop Detector
ITS	Intelligent Transportation System

LPU	Link Profile Unit
MAC	Media Access Control
MLE	Maximum Likelihood Estimation
OD	Origin-Destination
OPAC	Optimization Policies for Adaptive Control
PaaS	Platform as a Service
RHODES	Real-Time Hierarchical Optimized Distributed and Effective System
SaaS	Software as a Service
SCATS	Sydney Coordinated Adaptive Traffic System
SCOOT	Split Cycle Offset Optimization Technique
SHA	Secure Hash Algorithm
SMS	Space-Mean Speed
TMS	Time-Mean Speed
TOPSIS	Technique for Order of Preference by Similarity to Ideal Solution
XML	Extensible Markup Language

ACKNOWLEDGEMENTS

I would like to express my deepest appreciation to my academic supervisor, Prof. Yue Liu for his unwavering support and guidance during my Ph.D. years. His knowledge and experience have encouraged me in my academic research and daily life. I also appreciate the opportunities he gave me to acquire knowledge and build skills.

I would like to extend my sincere thanks to my committee members, Prof. Jie Yu, Prof Xiao Qin, Proj. Zeyun Yu and Jun Zhang for their helpful advice on my dissertation.

Thanks to my colleges Dr. Yun Yuan, Xinyu Liu, Dahai Han, Zihao jin, and Xiangyong Luo for their friendly help.

I also would like to express my thanks to my family for the support and encouragement.

Chapter 1: Introduction

1.1 Background

In cities around the world, traffic congestion has been a growing issue. It results in longer travel time, lower reliability, and larger energy consumption. As the traffic volume in urban areas continues to increase, it probably will remain a problem or even get worse. Many efforts have been made on relieving traffic congestion. To meet continuously growing travel demand, extra roadways have been built to increase the capacity of traffic networks. Meanwhile, traffic diagnosis and management strategies are applied to optimize the use of existing traffic networks. Compared with building more and more roadways, traffic diagnosis and management strategies are generally considered more cost-efficient approaches to reduce traffic congestion. However, the effectiveness of traffic diagnosis and management strategies heavily relies on data. Comprehensive data guide on where and when traffic signal redesign is needed and how to improve existing plans, while fragmented data may result in misjudgment of traffic conditions and misleading traffic management strategies. Data are consumed in the entire process of traffic diagnosis and management, including but limited to traffic monitoring, bottleneck identification, traffic signal evaluation, and back testing.

Most existing data collection methods, such as surveys, inductive loop detectors, Bluetooth, Wi-Fi cannot provide continuous and high-resolution data at a low cost. Most early research on traffic diagnosis is based on data from manual surveys and floating vehicles. Manual surveys and floating vehicles are so labor-intensive that they are mainly used as a sampling method to get representative samples for short periods and do not suffice comprehensive traffic diagnosis in the long run.

Compared with manual surveys and floating vehicle methods, inductive loop detectors, Bluetooth probes, and Wi-Fi probes are more cost-efficient. However, they suffer from their drawbacks. For example, data collected from inductive loop detectors have high sampling rates but do not include unique identifications for vehicles. Data from Bluetooth or Wi-Fi probes have MAC addresses as unique identifiers for vehicles, but sampling rates are relatively low.

Automated License Plate Readers (ALPRs) can capture the license plate numbers of vehicles when vehicles drive through specific areas. ALPRs record license plate numbers of vehicles with time, and ALPR IDs. Some advanced algorithms enable the ALRP system to capture even more detailed characteristics of vehicles, such as license plate colors, travel speeds, and turnings at intersections. Just like MAC addresses of Bluetooth and Wi-Fi, the license plate numbers in records work as unique identifications. Most importantly, compared with Bluetooth and Wi-Fi sensors, the sampling rate of ALPRs is much higher. So, ALPRs have improved data collection and analysis by boost the sample size and reliability.

Figure 1 illustrates a common setup of the ALPR system at an intersection. Each ALPR camera is set up on a traffic light and monitors one approach or several lanes of an approach of the intersection, recording vehicle license plates, arriving times, and other information when vehicles come into the view of ALPRs. The number of ALPRs to fully cover an intersection may vary because intersections have different numbers of legs or even lanes. More complicated intersections will need more ALPRs.

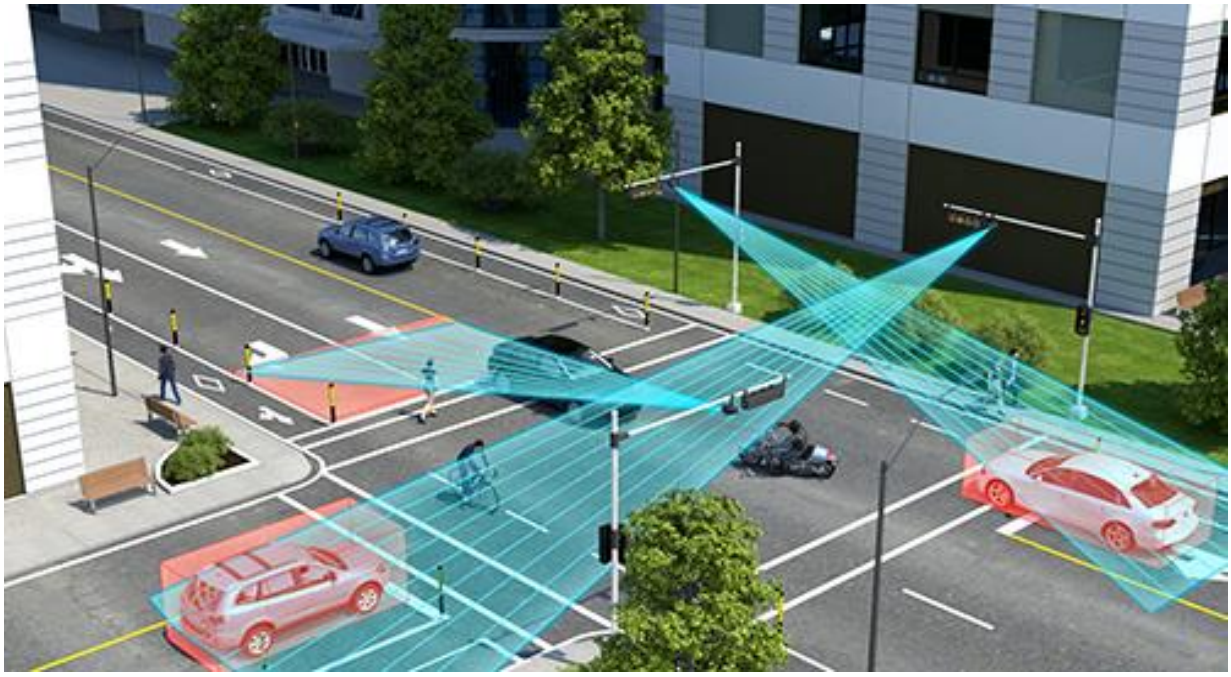


Figure 1 Automated license plate reader at an intersection

In summary, the comparison among generally used technologies for traffic monitoring and ALPRs is synthesized in Table 1. Most of the existing data collection methods are based on sampling, which means information of only a small fraction of vehicles in the traffic stream is being captured. The accuracy of the estimation of traffic characteristics highly depends on the sampling rates. Data collected from manual surveys are generally considered the ground truth of traffic flow. Therefore, it has the highest sampling rate among all methods. However, the data collection process of manual surveys is not automated and does not meet the requirements of real-time traffic diagnosis. Except for manual surveys, inductive loop detectors have the highest sampling rate. However, the aggregated information collected from inductive loop detectors does not suffice comprehensive traffic monitor and diagnosis. ALPRs have the highest sampling rate among technologies that keep records with unique identifications. The main drawback of ALPRs is that the data collected contain information about drivers and vehicles, which may raise privacy concerns. To address this issue, license plate numbers are usually encrypted before data processing. In this dissertation, all license

plate numbers are encrypted using SHA1, which is a commonly used encryption algorism. People may also have concerns about the cost of ALPRs. Since they are set up on traffic lights, the overall cost of setup and maintenance is much lower than loop detectors, which are buried a few inches below the road surface. Besides, in many cities, these ALPRs have already been set up for law enforcement. There is no reason not to utilize these valuable data for traffic monitoring and management.

Table 1 Comparison of sensor technologies

Technology	Unique Identifier	Detection Target	Sample Rate[1]
ALPR	License Plate Number	Vehicle License Plate	High
Manual Survey	N/A	Vehicle	High
Inductive Loop	N/A	Vehicle	High
Bluetooth	MAC Address	Electronic Device	Low
Wi-Fi	MAC Address	Electronic Device	Low

Overall, low cost per unit of data, continuous observations, automated data collection, relatively high sampling rate, and no disruption of traffic are major advantages of the ALPRs. On the other

1 Data from manual Survey, inductive loop, and radar detector are generally considered as the ground truth of traffic. So, the sample rate of these methodologies is considered as high in Table 1. Even tough research show the accuracy and reliability of estimation of travel time based on Bluetooth data (Sharifi et al. 2011; Haghani et al. 2010; Erkan and Hastemoglu 2016), the average travel time sample rate of Bluetooth is lower than 5% (Sharifi et al. 2011; Haghani et al. 2010). For privacy reason, new smart phones start to use randomly created MAC address, which leads to lower effective sampling rates in the future. On the other hand, based on our research, the average sampling rate of ALPRs is above 60% (corresponding travel time sample rate is about 35%) The sample rate is expected to be higher with better image processing and recognition technology. So, the sample rates of Bluetooth and Wi-Fi technologies are listed as low, while the sample rate of ALPR is considered as high.

hand, high initial cost, telecommunication requirements, and privacy issues are drawbacks of this technology.

1.2 Research objective

The primary objective of this dissertation is to design a laboratory utilizing ALPR data for traffic diagnosis and management. Traffic diagnosis and management is a continuous work, consisting of traffic monitoring and evaluation, bottleneck identification, traffic signal redesign, re-evaluation, and fine-tuning. ALPR system is one of the most promising data sources that can provide high-resolution data to support the entire procedure. Because of its overwhelming advantages, an extensive body of related literature exists. However, research mainly focused on small-scale cases on estimations of traffic flow characteristics. The rapidly increasing coverage and scale of ALPRs in cities put urgent needs for next-gen scalable and efficient algorithms. Besides, limited studies illustrate and validate the overall capability of ALPR data in the life cycle of traffic diagnosis and management, especially at the network level. Finally, there is no guidance about how the information extracted from ALPR data will better support traffic management. To fill these gaps, this research plans to develop a traffic diagnosis and management laboratory utilizing high-coverage and large-scale ALPRs data. The online laboratory consists of modules for traffic condition reconstruction, diagnosis-oriented visual analysis, and traffic control and management decision support. This research contributes to:

(1) developing a traffic diagnosis and management laboratory based on high-coverage ALPR data;

(2) designing scalable data-driven methodologies to measure and estimate traffic flow characteristics, operational performance, and traffic demand;

(3) designing and implementing advanced visualization tools for traffic monitoring and diagnosis at intersection, link, and network levels; and

(4) building ALPR-based traffic control and management decision support system;

1.3 Key research issues and primary research tasks

To accomplish the research objectives, this proposal has divided the research efforts into the following tasks:

1. Performing a comprehensive literature review of related research, including traffic condition reconstruction, transportation visual analysis, data-driven traffic control and management, and cloud computing in transportation.

2. Developing data-driven algorithms for reconstruction of traffic flow characteristics, estimation of operational performance, and reconstruction of traffic demand.

3. Developing visual analysis tools based on traffic condition reconstruction for traffic monitoring and diagnosis.

4. Developing methodologies and guidelines for traffic control and management decision support.

5. Collecting data from real-world case studies and validating proposed methodologies.

Figure 2 illustrates the entire research framework.

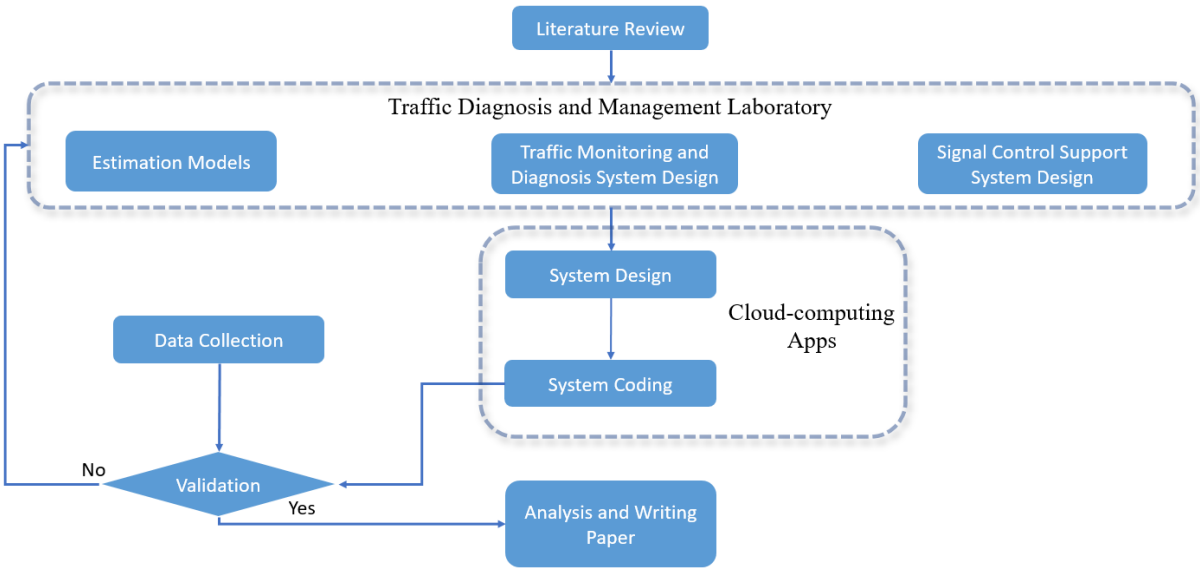


Figure 2 The overall dissertation research framework

1.4 Thesis outline

Based on the research objective, this dissertation is organized into seven chapters, shown as follows:

- Chapter 1 discusses the existing problems in the state of practice and motivation of this research concerning high-coverage and large-scale ALPRs system.

- Chapter 2 presents a comprehensive review of relevant research, including traffic condition reconstruction, transportation visual analysis, data-driven traffic control and management, and cloud computing in transportation. The review focuses on identifying the advantages and limitations of existing research.

- Chapter 3 designs the system framework for the cloud-based traffic diagnosis and management laboratory.

- Chapter 4 develops highly scalable and efficient algorithms for traffic condition reconstruction in aspects of reconstruction of traffic flow characteristics, Estimation of operational performance, and reconstruction of traffic demand.

- Chapter 5 develops visual analysis tools based on the results from traffic condition reconstruction for traffic monitoring and diagnosis.

- Chapter 6 develops methodologies and guidelines for traffic control and management decision support using ALPR data

- Chapter 7 summarizes the contributions of this dissertation and indicates future research directions.

Chapter 2: Literature review

2.1 Introduction

This chapter presents a comprehensive literature review of research in traffic condition reconstruction, transportation visual analysis, data-driven traffic control and management, and cloud computing in transportation. In each section, a review of existing studies is presented, followed by a brief summary of the limitations in the existing studies. The expected contributions are presented at the end of this chapter.

2.2 Traffic condition reconstruction

Traffic condition reconstruction is the procedure of extracting and analyzing traffic condition measurements from data. The ability to measure traffic conditions is crucial to traffic control and management. The remaining sections present the existing studies related to traffic condition reconstruction in five different aspects:

- *Traffic Sensor Technologies*: The innovation of sensor technologies provides a variety of data sources for traffic condition reconstruction. Section 2.2.1 summarizes widely used sensor technologies and compares the difference among them to show the strength and weaknesses of each technology. The summary also attempts to answer the question: From the perspective of sensor technology, in which aspects ALPR can perform better estimation for traffic condition reconstruction?

- *Data Cleaning Methods*: Data cleaning is a common step in the data-driven approach for traffic condition reconstruction. It improves the data quality, which in term boosts the accuracy of estimations. Section 2.2.2 presents a review of data cleaning problems and

corresponding treatments. At the end of section 2.2.2, a discussion about how these technologies fit in the proposed traffic diagnosis and management laboratory is presented.

- Speed Estimation: Travel speed estimation is one of the aspects that ALPR outperforms other data sources. As one of the fundamental traffic flow characteristics, estimation of traffic speed reflects traffic conditions in the perspective of traffic flow movement. Section 2.2.3 emphasizes the review of comparing the travel speed estimation using ALPR and other two widely used data sources: ILD and GPS. It also summarizes the deficiencies of existing travel speed estimation methodologies using ALPR.

- Delay Estimation: In urban traffic networks, control delay may account for a substantial part of total travel time. Measurements of intersection delay is another approach to measure traffic conditions. Section 2.2.4 summarizes the current approaches to delay estimation.

- Demand Analysis: Travel demand analysis provides an approach to understanding traffic conditions from the demand side. Section 2.2.5 presents a review of studies on travel demand analysis, especially origin-destination estimation.

2.2.1 Traffic sensor technologies

This section summarizes the major road sensor technologies for monitoring traffic flow. The recent development and innovations in sensor technologies enable monitoring traffic flow using different data sources. The main sensor technologies can be categorized into three groups: point detectors, continuous vehicle tracking, and point detectors with vehicle identification (Mori et al. 2015). Table 2 summarized the main sensor technologies used for traffic flow characteristics analysis and their strength and weakness (Haghani et al. 2010; Sharifi et al. 2011; Mazaré et al. 2012; Porter et al. 2013; Mori et al. 2015; Friesen and McLeod 2015; Nellore and Hancke 2016;

Jain, Saini, and Mittal 2019; APPIAH, QUAYSON, and OPOKU 2020). It can be found that compared with other sensor technologies, the three main features of ALPR include:

- Capable of identifying vehicles
- Relatively high sampling rate among technologies that can identify vehicles
- Fixed detection area and clear detection boundary.

Such features enable the ALPR to outperform other data sources with respect to the following aspects of traffic condition reconstruction:

- Speed estimation: More specifically, ALPR is a good sensor technology for estimating space-mean speed. In urban areas, measurement of travel time requires the technologies to identify vehicles. Among these technologies, ALPR has the highest sampling rate, on which analytics can perform the more reliable estimation. Besides, the detection area of ALPR is fixed and has a relatively clear boundary, which GPS and Wi-Fi probes do not have. GPS data is updated based on a fixed frequency. Matching the travel time from GPS and target road segments require extra algorithm and methodologies. Wi-Fi probes have a very broad detection area and can not specify the location of vehicles.
- Delay estimation: With the ability to capture the travel time of individual vehicles, ALPR can recognize how individual vehicles are affected by traffic control. At the same time, the high sample rate of ALPR ensures the delay of individual vehicles reflects the delay of most vehicles.
- Demand estimation: GPS is widely used for origin-destination estimation because it provides complete trajectories of vehicles. The only limitation is that the sampling rate of

GPS data is relatively low. On the contrary, the ALPR has a higher sampling rate than GPS data, but the trajectories built from ALPR are often incomplete. ALPR and GPS may cover the weakness of each other to perform a better estimation on the origin-destination matrix. There are many studies on origin-destination estimation based on other sensor technologies. More discussions on origin-destination estimation using advanced methodologies are discussed in section 2.2.5.

Table 2 Traffic sensor technologies

Sensor Technology	Description	Category	Strength	Weakness
Inductive Loop Detector (ILD)	ILDs are buried a few inches below the road surface and detect vehicles passing over them.	Point detectors	- Directly obtain traffic volume, occupancy, and headway - High sampling rate and the best accuracy for traffic count - Mature technology - Do not require frequent maintenance	- Require double inductive loops or advanced methodologies to measure instantaneous travel speed - Require multiple inductive loops to measure the space-mean speed - When maintenance is needed, lane needs to be closed
Ultrasonic sensors	Ultrasonic sensors are road-side detectors. They detect vehicles by emitting ultrasonic waves and receiving reflected waves.	Point detectors	- Directly obtain traffic volume, occupancy, and headway - Low cost - High sampling rate - Classification of vehicles based on the size and height	- Performance is affected by temperature change and air turbulence - Require two ultrasonic sensors set up near to each other or advanced methodologies to measure instantaneous travel speed - Require multiple ultrasonic sensors to measure the space-mean speed
Bluetooth probes	Bluetooth probes are road-side detectors and detect vehicles by monitoring the inquires made by digital devices with Bluetooth transceivers.	Point detectors with vehicles identifications	- Directly obtain density and flow - Identify vehicles with MAC addresses - Low cost - Easy to maintain	- The detection area is round, and no clear boundary exists. - The detection target is digital devices instead of vehicles - Relatively low sample rate
Wi-Fi probes	Wi-Fi probes are road-side detectors and detect vehicles by investigating	Point detectors with vehicles identifications	- Directly obtain density and flow - Identify vehicles with MAC addresses	- The detection area is a broad round and no clear boundary exists.

	the Wi-Fi request and response between digital devices and Accessing Point (AP).		- Low cost - Easy to maintain	- The detection target is digital devices instead of vehicles - Relatively low sample rate
Global Positioning System (GPS)	The Global Positioning System (GPS) is a positioning system based on satellites. It can record the trajectories of vehicles equipped with GPS trackers.	Continuous vehicle tracking	- High-frequency position information of vehicles - Directly obtain travel speed of individual vehicles - Directly obtain vehicles trajectories	- Require drivers voluntarily provide their private GPS data - Privacy concerns
(Automatic License Plate Reader) ALPR	ALPRs are cameras capable of recognizing license plate numbers using image processing technologies.	Point detectors with vehicles identifications	- Directly obtain traffic volume, headway, and instantaneous speed. - Identify vehicles with license plate numbers - Monitor multiples lanes - Measure travel time of individual vehicles by pairing license plate numbers captured at different ALPR - When the coverage is high, trajectories of vehicles can be reconstructed to reflect the real travel demand	- The sampling rate is affected by external factors, such as weather, temperature, and light condition. - Frequent maintenance, including lens cleaning - Can not obtain occupancy - Privacy concerns

2.2.2 Data cleaning methods

Data cleaning detects and removes errors and inconsistencies from data to improve the quality of data (Rahm and Do 2000). With the increasing demand for data integration and data transformation, the need for data cleaning continuously increases. Besides, the appearance of tremendous tools in the field of data science simplifies the process of analyzing data using statistical methodologies and machine learning algorithms. The focus of the majority of data analytics or researchers shifted from implementing methodologies to better utilizing the existing data and available tools. Data cleaning, data preprocessing and model tuning becomes more and more important for reliable data analysis. A survey shows about 60% of data analytics spend more than 60% of their time in data mining projects on data cleaning and preparation ("Data Preparation Part in Data Mining Projects" 2003).

Rahm and Do (2000) summarized the data quality problems using Figure 3. According to the number of data sources, they classify the data quality problems into two classes: single-souces problems and multi-source problems. Each class consists of scheme-level problems and instance-level problems. For single-source problems, they highlighted the importance of well-designed schema and integrity constraints. Significant research efforts have been made on designing and maintaining schema to protect the integrity of databases. Halpin and Proper (1995) discussed the early technologies in the theory of schema transformation and optimation. Galhardas et al. (2000) proposed a framework to models the data cleaning process of single-source problems for four main types of data errors: (1) the absence of universal keys for entities (2) the use of different data formats (3) the existence of data errors, and (4) the inconsistencies in data. Furthermore, they proposed an SQL extension to specify the data cleaning process of each type of error. Phipps and Davis (2002) designed algorithms for automatic conceptual schema development and evaluation

to the tremendously growing need on creating data warehouses. Curino et al. (2010, 2013) presented the PRISM/PRISM++ system and technologies to automate the process of migrating databases and rewriting legacy systems.

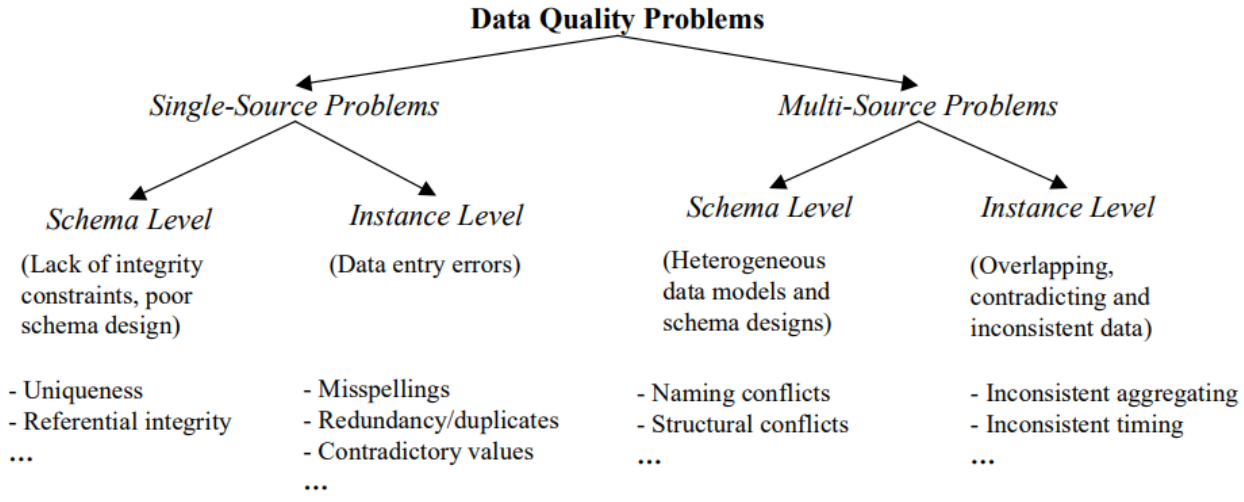


Figure 3 Classification of data quality problems in data sources (Rahm and Do 2000)

Instance-level problems are the problems invisible at the schema level. Instance-level problems include but not limited to duplicated records, missing values, misspelling, and outliers. Monge (2000) discussed the necessary properties of matching algorithms for a successful duplicate detection system. Low, Lee and Ling (2001) developed a generic knowledge-based framework for data duplicate elimination. Acock (2005) discussed the types of missing values and summarized traditional approaches to working with missing values: listwise deletion, pairwise deletion, indicator variable, and substitution. Kwak and Kim (2017) presented a comprehensive review on methodologies for outlier detection and treatment.

All the data quality problem of single-source systems happens to individual data sources in multi-source systems as well. Besides, when integrating multiple data sources, the conflicts and inconsistency among different data sources need to be resolved. One of the main problems in multi-

source systems is schema integration. A comparative analysis of 12 early methodologies for schema integration is performed by Batini, Lenzerini, and Navathe (1986). They summarize the comparison by highlighting the missing aspects of early methodologies: processing specifications, behavioral specification, and schema mapping. Doan, Domingos, and Levy (2000) found that manually constructing mappings between schemas in large-scale datasets is both labor-intensive and error-prone. To solve this problem, they designed a system that automatically finds such mappings. Madria, Passi, and Bhowmick (2008) noticed the different schema implemented in web XML data and needs for integration. He proposed a schema integration and querying methodology for XML data. The other common problem in multi-source systems is how to merge common entities among multiple data sources. Even if different data sources shared the same schema, entities can be represented differently in different data sources. Hernández and Stolfo (1998) developed a scalable methodology for searching duplicate information entries about the same entities and perform merging. Another scalable algorithm is designed by Ananthakrishna, Chaudhuri, and Ganti (2002) to eliminate fizzy duplicates. Bilenko and Mooney (2003) proposed a new methodology using a trainable measure of textual similarity to find duplicates. The improvements in duplicate detection accuracy were verified through case studies on a range of datasets.

Data cleaning is not all about methodologies and technologies. den Broeck et al. (2005) highlighted the importance of treating data cleaning as a planned and systematic process. They suggested that the processing should a repeated cycle of three stages: screening, diagnosing and editing. Because it is very hard to decide if a data point is collected with an error immediately, a continuously repeated process can help improve the quality of data. They suggest a soft cutoff for error screening. Instead of immediately categorizing data points into normal data samples and outliers,

a soft cutoff and continuous diagnosis preserve a middle area for suspicious data to be further verified.

There are enormous needs for data cleaning in the field of transportation because the data are often massive and unstructured. However, the related studies on data cleaning about traffic data are relatively limited and the majority of them are about performing outlier detection on sensor data. For example, Robinson and Polak (2006) proposed a data cleaning treatment for ILD data based on the relationship between flows measured on adjacent detectors. Bachechi, Rollo, and Po (2020) implemented a fast data cleaning process to remove traffic sensor errors. Dion and Rakha (2006) proposed a filter algorithm to detect outliers in ALPR samples to achieve higher accuracy on travel time estimation.

There are mainly three missing aspects in the studies on transportation data cleaning :

- Schema Integration: To fully utilize the traffic sensor data, analytics usually integrated sensors with geographic data and signal plan data. Besides, as the coverage of new sensor technologies raises, the number of data sources and size of data volume is continuously increasing. There is an urgent need for schema integration among different data sources.

- Solutions to merge \ purge problem: There are many common entities among traffic senor data, geographic data, and signal plan data. However, the information of these common entities may not be consistent among all data sources. A common case is that the names of intersections are expressed in different ways in different data sources.

- Systematic and continuous data cleaning process: A continuous data screening, diagnosis, and editing process helps to understand the quality of existing data and improve the consistency of the data system. Besides, profiles about entities need to be built during this

continuous process and help to improve the performance of the system. Errors may happen to any type of traffic sensor. By recording sampling rates and detection default rates result learned from data cleaning, the system can perform better estimation and identify malfunctioning sensors.

2.2.3 Speed estimation

Travel speed is one of the fundamental traffic flow characteristics. Travel speed is generally calculated based on travel time. Travel time is the time difference for a vehicle to travel from a place to another over a specified period. Time-mean speed and space-mean speed are two measurements of the speed of traffic. Time-mean speed focuses on instantaneous speed on a spot, while space-mean speed focuses on average travel time over a specified distance. The remaining of this section reviews the existing studies on travel speed estimation with different sensor technologies.

Inductive loops are widely used as traffic detectors (Klein, Mills, and Gibson 2006). But it is not designed to originally designed to measure travel speed. Single inductive loops can calculate the instantaneous speed of vehicles only if the length of vehicles is given (Mikhalkin, PAYNE, and Isaksen 1972). The spot speed of each vehicle can be expressed as:

$$v = \frac{L}{t}$$

where L is the sum of the effective length of the vehicle and the length of the sensor. In the Travel Time Data Collection Handbook (Turner et al. 1998), a constant vehicle length is assumed and estimation of spot speeds are expressed as:

$$speed = \frac{volume}{occupancy * g}$$

where g is the speed correction factor, which depends on the length of the vehicle and the detector. The assumptions behind this formula are the fundamental equation of traffic flow and a linear relationship between occupancy (F. L. Hall and Persaud 1989). F. L. Hall and Persaud (1989) also concluded that the results of this model were biased because the assumptions did not hold under general circumstances. Instead of assuming a constant vehicle length, Dailey (1999) extended the estimation by considering vehicle length as random variables. The key to the estimation process is a statistical algorithm.

However, these two methods may not fit different traffic conditions and need recalibration before implementations. Another approach of estimating spot speed using single loops utilizes the shape of vehicle waveform from inductive loops. C. Sun and Ritchie (1999) found a linear relationship between speed and slew rate of vehicle waveform and propose a linear regression approach to estimate spot speed with single inductive loops:

$$speed_i = \alpha + \beta slew_i + \varepsilon_i$$

Oh, Ritchie, and Oh (2002) extended the linear regression model by including vehicle grouping methods and a log transformation of duration as one of the independent variables.

Compared with single inductive, double inductive can directly calculate the spot speed of vehicles. Double inductive loops consist of two single inductive loops. These two single inductive loops are generally close to each other. With double inductive loops, travel time between two inductive loops can be measured, which in term can be used to calculate the spot speed of vehicles.

$$v = \frac{d}{t_2 - t_1}$$

Where

v = spot speed

d = distance between two single inductive loops

t_1 = starting point of occupancy of the first single inductive loop

t_2 = starting point of occupancy of the second single inductive loop

To improve accuracy, the equation can be transformed to utilize both the starting point and ending point of occupancy. These spot speeds of vehicles can be simply aggregated to measure the time-mean speed of vehicles near double inductive loops. The time-mean speed only focuses on instantaneous speed on a point. Better measurements of performance on road sections are space-mean speed and average travel time. There is a fundamental transformation between time-mean speed and space-mean speed (Wardrop 1952):

$$\overline{v_t} = \overline{v_s} + \frac{\sigma_s^2}{\overline{v_s}}$$

where

$\overline{v_t}$ is the time-mean speed;

$\overline{v_s}$ is the space-mean speed;

$\sigma^2{}_s$ is the variance of space distribution.

However, σ_s can not be achieved from inductive loop data. Han et al. (2010) presented a novel formulation to estimate SMS using TMS and traffic states.

Another approach of estimating SMS or average travel time is derived from traffic flow theories. Based on traffic flow theory, Nam and Drew (1996) derived a stochastic process model to estimate total travel time. The derivation of this stochastic process model is based on two assumptions: Vehicles follow the first-in, first-out (FIFO) queuing principle and conservation of vehicles. Lots of traffic flow theory-based models have been proposed to fit different traffic flow conditions (Coifman 2001; J.-S. Oh, Jayakrishnan, and Recker 2002; Vanajakshi, Williams, and Rilett 2009; Long, Gao, and Szeto 2011; Celikoglu 2013).

Unlike data from point detectors, GPS produces information about the time and location of vehicles over their trips, which can be directly used to calculate the travel time of vehicles along with their trips. Because the location information produced by GPS is longitude and latitude, spatial models, GIS systems, and map matching algorithms are crucial to project the travel time of GPS information to the travel time of road sections. Quiroga and Bullock (1998a, 1998b) verified the feasibility of utilizing GPS and GIS data for travel time estimation in the perspective of spatial models, sample size requirements, and detailed implementation procedures.

To improve the accuracy of travel time estimations, more advanced spatial models and map matching algorithms are developed (Greenfeld 2002; Marchal, Hackney, and Axhausen 2005; Quddus, Noland, and Ochieng 2006; Taylor et al. 2006). Most of the early GPS-based research highly rely on high-quality data, which are produced by specialist data providers (Sanaullah, Quddus, and Enoch 2016). With more and more vehicles are equipped with GPS devices, studies started to design more robust methodologies to utilized those sparse and noisy GPS vehicle data

(Hunter et al. 2009; Mazloumi, Currie, and Rose 2010; Sanaullah, Quddus, and Enoch 2016). The GPS data of private vehicles are under privacy protection. To avoid privacy concerns, the GPS data for traffic analysis are mainly collected from probe vehicles or a specific group of vehicles, such as taxis or buses.

Similar to GPS data, ALPR can identify vehicles. By matching the license plate number of different records from sensors, the travel time between sensors can be directly calculated. Besides, most ALPR sensors are set up at intersections. The travel time of vehicles between ALPR sensors measures travel time between intersections. Using ALPR data for travel time estimation almost avoids all the difficulties discussed in the above two sections. However, to have a robust and accurate estimation, a few challenges need to be addressed:

(1.) Due to detection error and coverage issues, the ALPR records of vehicles may not always construct the complete trajectory of vehicles. Even though the travel time of vehicles between intersections can be directly calculated, the exact trajectories of vehicles between these intersections are not determined.

(2.) Vehicles may stop off-street between intersections. The records of these vehicles produce outliers in the travel time estimation.

(3.) The performance of ALPR systems depends on a few external factors, such as weather, lightness (Du et al. 2012), and temperature. Rondon (2014) performed a comprehensive analysis on how weather and temperature affect the performance of ALPR systems.

Among these three problems, the second is the most critical one because it will directly make the estimation biased to outliers while the other two only slightly affect the sampling rate for travel time estimation. In recent years, a few studies made efforts on exploring methodologies to filter

the outliers. These studies can be categorized into two groups: algorithm-based methods and model-based methods.

Algorithm-based methods focus on providing efficient and applicable algorithms of estimations. Three outlier detection methods are compared to better estimate travel time (Clark, Grant-Muller, and Chen 2002). These three methods are straightforward to understand and easy to implement. There are also new filtering algorithms proposed for detecting outliers, which are proved to be helpful when sampling rates are low (Dion and Rakha 2006). Zhi-Peng et al. (2008) applied an improved adaptive smoothing method to achieve reliable short-term estimations and forecasts of travel time.

Compared with algorithm-based methods, the strength of model-based methods is their capability to explain traffic phenomena. Mo, Li, and Zhan (2017) proposed a vehicle speed profile estimation model, which includes a systematic License Plate Recognition (LPR) data-mending method and a customized car-following model. Kazagli and Koutsopoulos (2013) assumed that vehicles detected can be divided into two groups: one presenting normal movements through the network and the other representing vehicles that stop for any reason, which results in a mixture Gaussian model to fit the travel time data. Fei et al. (2013) built up a Bayesian dynamic linear model to estimate and predict real-time travel time.

To summarizes, compared with other sensor technologies, ALPR has two advantages in estimating travel time. 1) The travel time of individual vehicles between fixed points can be directly calculated, and 2) The sample rate of ALPR is relatively high. The main problem that remains to be solved to perform accurate and reliable estimation of travel time using ALPR data is how to filter out outliers in the data samples. In this dissertation, a new data-driven algorithm will be

developed for link travel time estimation. The results of the proposed algorithm are insensitive to outliers.

2.2.4 Intersection delay estimation

Intersection delay estimation is considered one of the most important measures of traffic conditions at intersections. It is widely used for the evaluation of intersection performance and many traffic control methodologies target minimizing the overall delay of traffic. More Specifically, intersection control delay is the increase in travel time relative to uncontrolled conditions (Highway Capacity Manual 2010). Several related terms are:

- Deceleration delay: the extra travel time during the deceleration process before vehicles stop before the stop line.
- Stop delay: the extra travel time when vehicles stop before the stop line.
- Acceleration delay: the extra travel time during the acceleration process after vehicles pass the controlled intersection.
- Approach delay: the extra travel time before entering the controlled intersection
- Approach delay: the extra travel time after entering the intersection and before reaching the free-flow speed

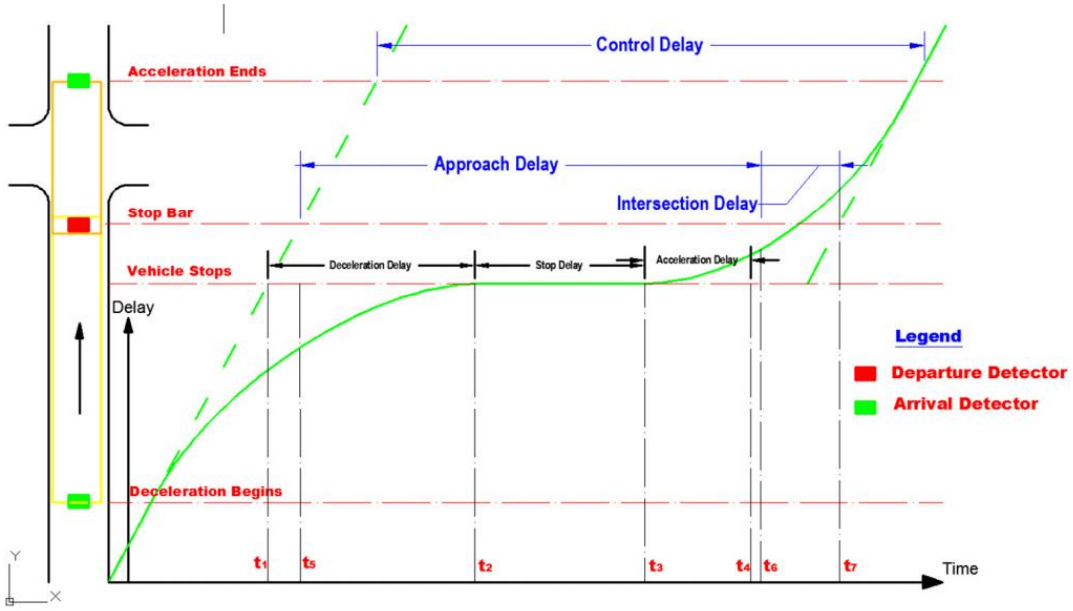


Figure 4 Delays at signalized intersections (Shatnawi, Yi, and Khliefat 2018)

Figure 4 illustrates the relationship among these delays. In the last decades, tremendous studies made efforts to estimate overall control delay or other delays at signalized intersections. These studies can be generally categorized into two groups:

- *Theoretical Delay Estimation Model*: The theoretical delay is derived based on assumed or learned arrival patterns of vehicles. Most of these models can measure delays based on traffic flow characteristics, which makes them easy to be implemented. However, because assumptions of vehicle arrivals are made, it does not fit in different, such as oversaturation conditions. A comprehensive review of the theoretical delay estimation model is done by Cheng et al. (2016). The main points are summarized in 3.

Table 3 Development of theoretical delay estimation models (Cheng et al. 2016)

Stage	Time	Research Point	Representative achievements	Advantage	Disadvantage
1	1920s–1970s	Deterministic Arrival	HRB (1928), Clayton (1941), Wardrop (1952), Newell (1965), May and Keller (1967)	Easy to calculate	Failed to consider stochastic arrival
		Binomial Arrival	Beckmann, McGuire, and Winsten (1956), Newell (1960)	Probability model and avoid infinite delay	Unsuitable in platoon arrivals
		Poisson Arrival	Adams (1936), Webster (1958), Little (1961), Darroch (1964), Similar to real traffic Miller (1969), Robertson (1969), Ohno (1978)	Similar to read traffic arrivals	Overestimation for high saturation
		General Arrival	Miller (1963), Newell (1965), Cronje (1983), Hutchinson (1972)	Improve accuracy	More parameters were needed
		Time dependent Model	Catling (1977), Kimber and Hollis (1979), Akc¸elik (1980, 1981,1988), Teply et al. (1984), HCM (1985, 1994, 2000), Burrow (1989), Brilon and Wu (1990), Rouphail and Akcelik (1992)	Stochastic model for under and over saturated conditions	–
2	1970s–2000s	Model Modification	Akcelik (1988), Fambro and Rouphail (1997), Akgungor and Bullen (1999)	Improve the accuracy of time-dependent models	–
		PFs	Reilly et al. (1982), Chang, Messer, and Fambro (1987), Courage et al. (1988), Rouphail (1989), Olszewski (1990), Prevedouros and Koga (1996), Kim and Benekohal (2005)	Consider progression effects and improve accuracy	Under or overestimation for specific arrival types
		New methods for arrival estimation	Benekohal and El-Zohairy (2001), El-Zohairy and Benekohal (2001), Strong, Rouphail, and Courage (2006), Strong and Rouphail (2006), Showers (2002), Kyte et al. (2008)	Consider different arrival patterns and improve accuracy	–
3	2000 onwards	Delay for queue spillback and blocking conditions	Ahmed and Abu-Lebdeh (2005), Ahmed (2006), Ahmed,Abu-Lebdeh, and Al-Omari (2012), Xu et al. (2010),Kikuchi et al. (2004), Zhang and Tong (2008), Wang and Benekohal (2007), Yin et al. (2011)	Consider delay for special traffic conditions	Inaccurate estimation results
		Artificial intelligence techniques used for delay estimation	Qiao et al. (2002), Murat and Baskan (2006), Murat (2006), Gokdag et al. (2007), Hasiloglu et al. (2014)	Improve accuracy for oversaturated conditions	No specific estimation formula

- *Data-driven Delay Estimation Methodologies*: Compared with theoretical delay estimation models, data-driven delay estimation methodologies do not have an assumption on the arrival pattern of vehicles. Quiroga and Bullock (1999) proposed a new methodology to measure control delay at signalized intersections using GPS data. The methodology determines the main components of delay based on distance-time, speed-time, and acceleration-time diagrams of vehicle movements captured from GPS data. Colyar and Rouphail (2003) proposed a methodology to measure control data using second-by-second speed data of sampled vehicles. These data are collected from portable, on-road vehicle data measurement devices, which can collect second-by-second vehicle speed. Ko, Hunter, and Guensler (2007) proposed another approach to identifying control delay components based on vehicle speed and acceleration profiles. Speed profiles are used to identifying stopped delay and acceleration profiles are used to identify deceleration and acceleration delay. Abbas et al. (2013) presented two models to calculate traffic control delay based on Bluetooth and GPS probe vehicle data. The models found a relationship between the number of Bluetooth observation samples and the time duration between detections. Based on the discovered relationship, control delay is estimated.

To summarize, the majority of studies of intersection delay estimation are based on the theoretical delay estimation model. These models usually formulate the intersection delay as a function of measurable traffic flow characteristics. This feature makes them easy to implement. However, these models are built upon different assumptions of traffic arrival. Most of them only fit in specific circumstances. On the contrary, data-driven delay estimation methodologies directly identify the delay of individual vehicles to estimate the intersection delay. The main data source used in these delay estimation methodologies is GPS data. One critical problem of utilizing GPS for intersection

delay estimation is the low sampling rate. These methodologies are performed on probe vehicles. It is unsure that these probe vehicles can represent the entire traffic flow. In this dissertation, a new approach will be proposed to estimate stopped delay based on ALPR data.

2.2.5 Demand analysis

Traffic demand, which is usually quantified with the origin-destination matrix, is crucial to develop traffic management strategies and signal control plans. Traditionally, the origin-destination matrix is estimated with the four-step travel model. The four-step model relies on travel surveys to estimate trip generation. Hendrickson and McNeil (1984) and Yang et al. (1992) proposed classic bi-level optimization models for OD estimation based.

In recent decades, researchers also explored the capability of GPS to recognize traffic demand. Several studies replace the traditional traffic demand survey with GPS data to recognize the travel demands of residents. Wargelin et al. (2012) performed the first GPS-only household travel survey and verify the feasibility of utilizing GPS data to estimate travel demand. In the survey, household members were asked to carry personal GPS devices. Directly achieving personal tracing data raised a few privacy concerns. Ge and Fukuda (2016) used aggregated GPS data from digital devices to protecting the privacy of individuals. Besides, GPS data are also utilized to recognize the traffic demand of trucks (Bernardin Jr, Trevino, and Short 2015; Zanjani et al. 2015; Gingerich, Maoh, and Anderson 2016).

However, conducting surveys is labor-intensive. Besides, it is questionable whether the data samples can represent the overall travel demand. ALPR data reflects the real travel demand of all vehicles. By reconstructing the complete trajectories of vehicles, the OD matrix can be estimated. Topological and statistical methodologies were proposed for trajectory reconstruction. Q. Wang,

Sun, and Sakai (2017) proposed a Hidden Markov Model (HMM) based system to rebuild to missing trajectories. H. Yu et al. (2018) designed a topological-based trajectory reconstruction algorithm. Depth First Search (DFS) is implemented to find trajectories patching set and the best trajectory is determined by the TOPSIS method. Rao et al. (2018) applied a particle filter to model the probability of different candidate trajectories.

One common drawback of these existing reconstruction-based methodologies is that they do fit in a large-scale network. The performance of these existing methodologies drops as the size of the traffic network increases. To address this deficiency, in this dissertation a scalable algorithm for trajectory reconstruction will be proposed.

2.3. Transportation visual analysis

Visualization transforms massive data into more intuitive graphs or maps. It helps users to capture the insights within data. Most data-driven traffic management systems have embedded visualization modules. The visual analysis tools within traffic management systems enable traffic network monitoring, bottleneck identification, and the decision-making process. The legacy problem of transportation visualization is how to represent the spatial and temporal information of traffic flow so that the spatial-temporal pattern of traffic flow can be revealed. Many studies in the field of visualization systematically summarized the existing visualization methodologies and their application in representing spatial and temporal information (Kucera 1992; Vasiliev 1997; Andrienko, Andrienko, and Gatalsky 2003). In recent years, the innovations in sensor technologies produced a large amount of data, which has challenged the traditional ways of visualization. First, traditional visual tools may not clearly represent the tremendous information on a large-scale map. Second, users are no longer satisfied with using the visual tool for presenting. Visual tools are

more considered as a way to explore the data behind. In the following two sections 2.3.1 and 2.3.2, the literature review focus on two aspects of visualization: large-scale data visualization and interactive visualization interfaces.

2.3.1 Large-scale data visualization

As the data volume increases, more and more information represented in the traditional visualization techniques may get lost or hidden in occlusions, which are also called visual clutters. This visual clutter in graphs or charts prevents users from getting the correct information. So, advanced methodologies are needed to visualize large-scale data. Visual abstraction and clustering are two widely used methodologies to reduce or eliminate visual clutters (Novotny 2004). Visual abstraction removes unnecessary information so that the complexity of the visual representation decreases and data of interest can be easily revealed. On the other hand, clustering group similar data points together and aggregate information to reduce the total amount of data samples.

In transportation visual analysis, visual clutter often happens when too many data samples are presented or the displayed traffic network becomes too complicated. One example is creating an origin-destination map over a complicated traffic network. The number of OD pairs increases rapidly as the number of vertices increases. The flow lines of OD pairs will start to overlay each other or even cover the entire map. Researchers have exploited different techniques to visualize information more clearly. Besides OD map, almost all dense directed and weighted flow graphs face the same challenge in visualization. They are powerful tools to depict latent patterns of flow data on maps. To solve the visual clutter problem in dense directed and weighted flow graphs, many algorithms are proposed. These algorithms can be categorized into three groups:

- *Region-based graphs*: group points of interest together and create regions to represent groups of points. By reducing the total number of entities within the graph, the complexity of the view decrease. Figure 5 is an example of regional-based graphs. Zeng et al. (2013) visualized the interchange pattern of the metro system using region-based graphs.

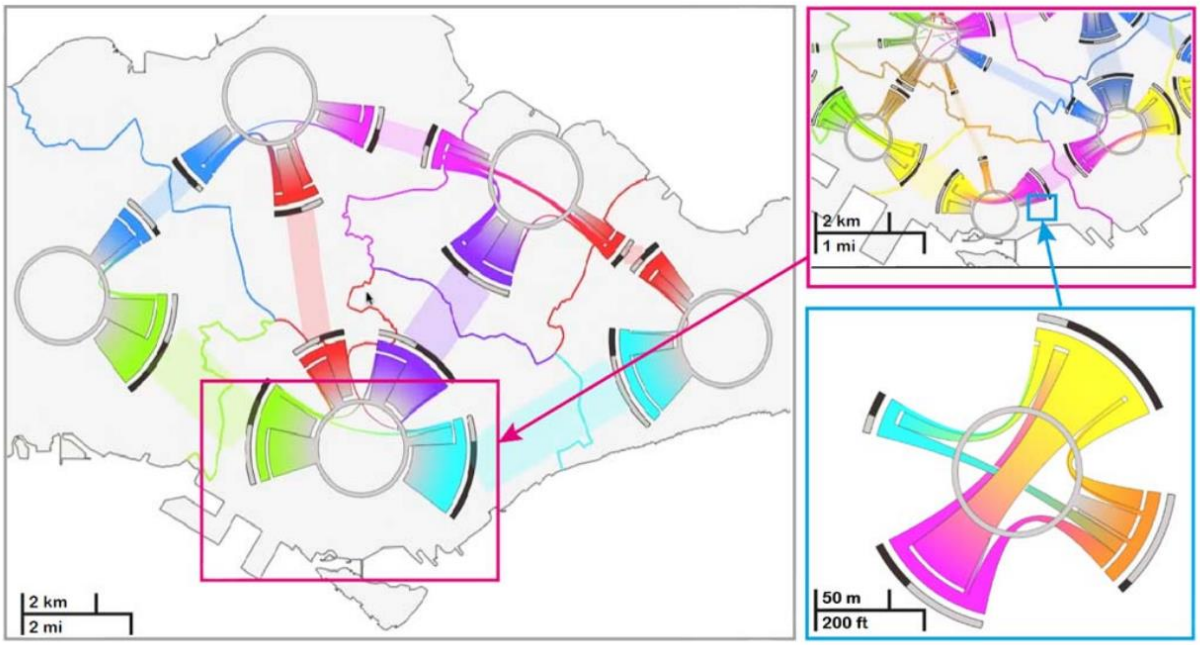


Figure 5 An example of regional-based graphs (Zeng et al. 2013)

- Edge bundling: group geographically close edges into bundles to reveal the overall pattern of the directed maps. Phan et al. (2005) first proposed the idea of edge bundling and presented a method to generated edge bundled flow maps using hierarchical clustering. Later, various methodologies were proposed. There are mainly three categories: cost-based edge bundling, geometry-based edge bundling, and image-based edge bundling. Zhou et al. (2013) summarized the different variations of edge bundling and discussed several implementations. Figure 6 a) is the example of a flow map before edge bundling. There are lots of overlapping between edges Figure 6 is the flow map after implementing the

skeleton-based edge bundling algorithm. The overlapped between edges is reduced and people understand the flow pattern more conveniently.

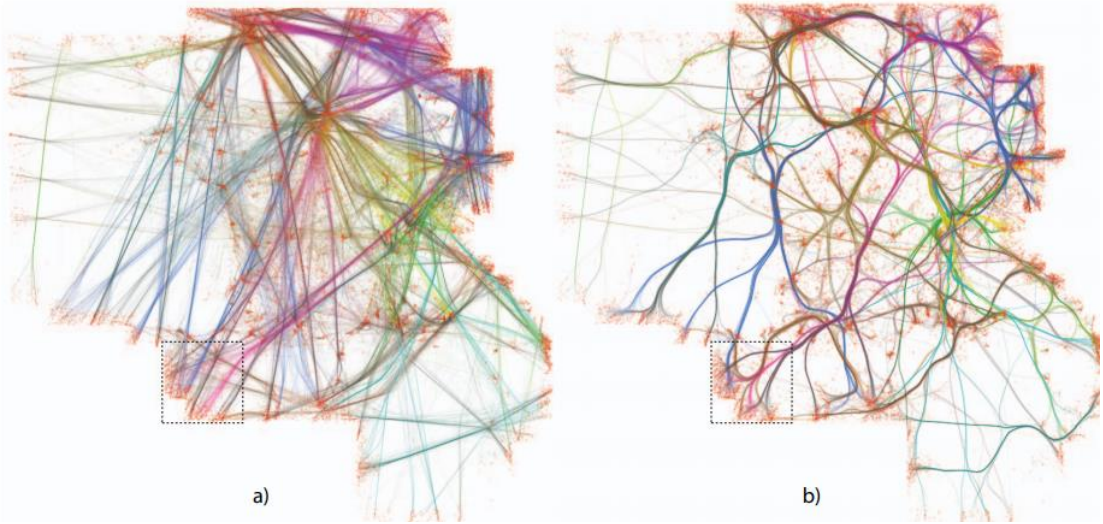


Figure 6 An example of skeleton-based edge bundling(Ersoy et al. 2011)

- *Clustered graph*: graphs with recursive clustering structures (Eades, Feng, and Lin 1996; Feng 1997). The clustered graph is one of the technologies to realize an interactive graph. Figure 7 illustrates the structure of a clustered graph. The clustered graph can be visualized at different levels. At the higher level, lower-level graphs are clustered into nodes so that the complexity of the higher-level graph is reduced. At the lower level, valuation is only performed on a subset of the network.

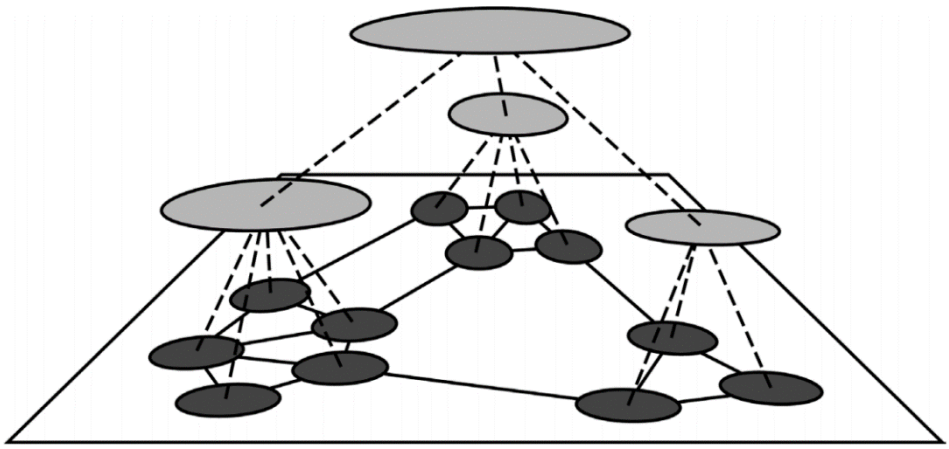


Figure 7 The structure of a clustered graph (Balzer and Deussen 2007)

Overall, monitoring large-scale traffic networks using large datasets necessitates advanced visualization tools to eliminate visual clutter problems. In the field of visualization, regional-based graph, edge bundling and clustered graph are the general solutions to visual clutters. Regional-based graphs and edge bundling methodology will be extended and implemented in the proposed research for visualizing origin-destination graphs.

2.3.2 Interactive visualization

Interactive visualization in intelligent Transportation (ITS) System allows users to explore data, test hypotheses, and evaluate redesigned signal plans. Traditional visualization methods do not suffice the need for traffic diagnosis and traffic management. Traffic diagnosis and management should include not only extracting information but also identifying problems, analyzing causes, and finding possible solutions. Besides, the characteristics of traffic conditions are closely associated. Each of these characteristics represents some aspects of traffic flow but will not tell the whole story. The visualization tools are required to query, integrate, transform and visualize multiple data sources based on users' needs. Last but not least, interactive visualization can avoid

visual clutter problems mentioned before. Recently, these needs have gained increasing attention in visual analytics.

G. Sun et al. (2014) designed a visualization system that can seamlessly embed temporal displays into a map for visualization of both the spatial and temporal attributes of the data. Flowstrates provide an interactive visualization approach to display spatial and temporal features of origin-destination flows in separate components (Boyandin et al. 2011). T-Watcher, an interactive visual analytics system, is designed for monitoring and analyzing complex traffic situations in big cities (Pu et al. 2013). Jiang et al. (2015) designed a system to assist users to interactively investigate O/D distributions over time. Yan et al. (2018) built up an interactive visual analytics system for users to explore movement patterns of bike-sharing from spatial and temporal attributes. Lu et al. (2016) present a visual analysis system to explore OD patterns based on taxi trajectories.

2.4. Data-driven traffic control and management

Traffic control and management have been widely applied to improve the efficiency and safety of existing traffic infrastructures. Since the invention of traffic control and management, the theories and systems have been continuously improved. In recent decades, these theories and systems have become more and more dependent on data. Compared with conventional traffic control management, data-driven traffic control and management are different in three aspects:

- More adaptive: Data-driven traffic control and management dynamically create and adjust the signal plan based on real-time data from sensors.

- More dependent on estimations and prediction: Data-driven traffic control and management directly rely on estimation and prediction of traffic conditions for optimization.
- Less dependent on the theoretical model: Theoretical models are commonly generalized for common use. Data-driven methodologies often recalibrate existing models or directly find patterns through data.

In terms of the range of applications, traffic control and management are generally performed at two levels: the intersection level and the arterial level. The studies in data-driven methodologies at intersection level and arterial level are summarized in 2.4.1 and 2.4.2 respectively.

2.4.1 The intersection level

At the intersection level, the general objective of traffic control strategies is to minimize the total delay of all vehicles or maximize the capacity of intersections. Isolated intersection control strategies achieve their goal by finding optimal split, and sequence. Conventional control strategies create fixed-time signal plans by minimizing delay, which is based on theoretical models and survey data (Allsop 1971, 1976; Yagar 1974; Stoffers 1967). The appearance of innovative sensing technologies gives birth to data-driven strategies.

One of the key limitations of conventional traffic control is that it pre-fixed signal plans do not fit with various traffic demand. Early data-driven methodologies focus on implement adaptive traffic control strategies to fit the fluctuation of traffic conditions. Miller proposed a traffic-responsive strategy that uses real-time information from vehicle detection loops to handle daily and weekly demand fluctuations in traffic flows (Miller 1963). The strategy continuously decides if the current phase should be truncated by evaluating the gain and loss on estimated delay. Gartner (1983)

proposed another traffic-response strategy called the Optimization Policies for Adaptive Control (OPAC). In OPAC, the decision-control problems are divided into smaller sub-problems to achieve better performance. Saiyed and Stewart (2004) investigated of results of pre-fixed and traffic-responsive signal controls strategies through a systematic simulation study and concludes that the traffic-responsive control strategies overperform the pre-fixed control strategies.

Many adaptive traffic control systems were developed based on adaptive traffic control strategies. Head, Mirchandani, and Sheppard (1992) proposed and developed the Real-time Hierarchical Optimized distributed and Effective System (RHODES). At the intersection level, RHODES optimizes the phase sequence and phase duration of each phase by minimizing measurements such as delay or stops (Head et al. 1998; Mirchandani and Head 2001). These measurements are calculated based on the real-time prediction about the traffic flow of links. RHODES predicts the traffic flow of links based on sensor data and parameter estimation, such as queue size, queue discharge rate, turning probability, and travel times. The prediction of traffic flow and estimation of parameters highly rely on real-time data captured from upstream sensors. Optimization Policies for Adaptive Control (OPAC) was first proposed in the 1980s and continuously developed (Gartner 1983; Gartner, Tarnoff, and Andrews 1991; Gartner, Pooran, and Andrews 2001, 2002). The local control level of OPAC implemented a rolling horizon strategy to utilize flow data estimated from traffic sensors. It uses both actual arrival data collected directly collected from traffic sensors and data obtained from a model, such as the moving average of past arrivals. Similar to RHODES, OPAC decides the sequence and phase splits by minimizing a performance function of delay or stops based on arrival date and estimations. Another adaptive traffic control system called the Split Cycle Offset Optimization Technique (SCOOT) is proposed in the 1990s (Hunt et al. 1981, 1982; Robertson and Bretherton 1991; Bretherton, Wood, and Bowen 1998). At the intersection level,

SCOOT uses measures of volume and occupancy to express traffic demand at traffic sensors. The measures also called Link Profile Units (LPUs), are utilized by a split optimizer to analyze the existing split timing. To summarize, at the intersection level, these adaptive traffic control systems estimate link traffic demand based on data from upstream sensors. Then the estimation of traffic demand determines the optimal signal plan. In this process, the performance of optimization directly relies on the estimation of traffic demand.

Another key limitation of the conventional control strategy is that the signal control procedure highly relies on a generalized model for the estimation of traffic demand and delay. The model builds a general relationship between traffic demand and traffic flow characteristics measured from sensors. However, a variety of extravel factors affect this relationship. These external factors include but are not limited to geometry design, driving behavior, and side street conditions. The generalized model cannot reflect the impact of these factors. In recent decades, many studies present data-driven models to solve this problem. Some of the studies proposed methodologies to recalibrate parameters in the theoretical model, which enables the theoretical model to better fit the real traffic conditions. Liu et al. (2015) proposed a data-driven linear decision rule approach for signal control. This approach consists of two stages. The first stage performs calibrations using historical data while the second stage performs optimization using real-time data. The others proposed models to directly recognize patterns through data. Zhang et al. (2018) proposed a data-driven model for traffic density prediction. The model is based on Hidden Markov Model and builds a dynamic state transition model between traffic volume and density. They also developed a signal control framework based on the traffic density prediction model. Numerical experiments are performed on 19 intersections and the results showed a decrease in traffic congestion under medium and high traffic demand. Abdulhai, Pringle, and Karakoulas (2003) presented a case study

utilizing reinforcement learning for adaptive isolated intersection signal control. The control system of the isolated intersection is the agent to learn a policy that maximizes the rewards in reinforcement learning. They concluded that promising results are achieved from implementing reinforcement learning to the isolated intersection control system.

Overall, one of the key components of intersection-level traffic control and management system is the estimation and prediction of link traffic flow. The accuracy of estimation and prediction of link traffic flow determined how the signal plan allocates green time to approaches. In this dissertation, the capability of estimating and predicting link flow rate using ALPR data will be developed and explored.

2.4.2 The arterial level

At the arterial level, signal control strategies optimized the phase difference (offset) between adjacent intersections to realized "green wave" along arterials. Several conventional models were proposed to maximize the bandwidth of "green wave (Little, Kelson, and Gartner 1981; Stamatiadis and Gartner 1996; Robertson 1969).

Compared with pre-fixed strategies, traffic-responsive strategies can achieve better performance by adjusting signal plans in real-time. SCOOT was designed by Robertson and his team as a coordinated traffic-responsive traffic control system (Hunt et al. 1982). SCOOT utilizes the real-time volume and occupancy data from sensors to build dynamic signal plans. Later, more traffic-responsive traffic control strategies were designed (Henry, Farges, and Tuffal 1984; Sen and Head 1997; Gartner, Pooran, and Andrews 2001). These dynamic adjustments benefit the most when the traffic demand fluctuates. On the other hand, traffic-responsive strategies require more real-time information to produce functional results. Most popular adaptive traffic control systems have

implemented arterial-level traffic control strategies. In RHODES, the arterial-level traffic control is based on REALBAND. REALBAND minimizes the total stops of vehicles in the platoon. Sensor data are used to recognize platoon in the traffic network. Based on the flow size and travel speed of the platoon, RHODES predicts the platoon flow at intersections. The optimal offsets are found to minimize the stop of estimated platoon flow. SCOOT implemented a platoon dispersion model to transform Link Profile Unites (LPU) into Cyclic Flow Profiles(CFP), which measure the platoon flow at the stop lines of intersections. Then the offset optimizer in SCOOT decide the offsets according to the CFPs. Similiaryly, OPAC utilize data from upstream sensors to identify platoon and predict the arrival of vehicles in the platoon. A platoon dispersion model predicts whether the arrival of an identified platoon will be uniform, random, or as a platoon.

A key assumption behind these coordinated signal control systems is that vehicles form platoons at the exit of upstream signalized intersection and arrive at the downstream intersection with particular arrival patterns. If the vehicle arrival patterns are robust and predictable, delays and stops of vehicles can be minimized by assigning optimized offsets to signals along an arterial or throughout a network. Platoon dispersion models are generally applied for validating the feasibility of coordinated control strategies. Platoon dispersion models explain the mechanism of the platoon dispersion process and predict the downstream flow profile.

Several platoon dispersion models have been developed in the literature. Pacey (1956) used kinematic theories to model the dispersion of vehicle platoons. This model is derived based on four strong assumptions:

- The vehicles in the platoon travel at a constant speed.
- The travel speeds of vehicles in the platoon follow a normal distribution.

- The platoon dispersion is caused by and only caused by the different travel speeds of vehicles in the platoon.
- Passing is free.

The downstream flow profile given by the model can be expressed as:

$$q_2(T)dT = \int q_1(t)g(T-t)dtdT$$

Where

$q_1(t)$ = the upstream traffic flow between t and t+dt

$q_2(t)$ = the downstream traffic flow between t and t+dt

$g(t)$ = the probability density function of the distribution of travel time

A more generally used expression is the discrete form of this equation:

$$q_2(j) = \sum_i q_2(i)g(j-i)$$

This equation represents the downstream flow in interval j as a weighted summation of upstream flows. The weights quantify the number of vehicles that can arrive at the downstream observation point after j-i time intervals.

Grace and Potts (1964) showed Pacey's model can be converted to a one-dimensional diffusion model. Another platoon dispersion model is introduced by Robertson in 1969 (Robertson 1969). Because it can be easily applied, Robertson's dispersion model is one of the most widely used platoon dispersion models. It has been applied in many traffic simulation and signal control applications such as TRANSYT (Robertson 1969), SCOOT (Hunt et al. 1981)s, INTEGRATION

(Rakha et al. 2000), and SATURN (M. Hall and Willumsen 1980). Robertson's platoon dispersion model can be expressed in a recursive equation:

$$q_2(i+t) = Fq_1(i) + (1-F)q_2(i+t-1)$$

Where

F = smoothing factor

t = minimal travel time between upstream and downstream observation points

By recursively expanding the second term in the equation, Seddon (1972) rewrite the equation as:

$$q_2(j) = \sum_{i=1}^{j-t} F(1-F)^{j-i-t}\, q_i(i)$$

In Robertson's platoon dispersion model, the smoothing factor is quantified with three parameters:

$$F = \frac{1}{1+\alpha\beta T_a}$$

Where

T_a = average travel time of vehicles

α = platoon dispersion factor

β = travel time factor, quantified as the ratio between minimal travel time and average travel time of vehicles: $\frac{T}{T_a}$

The parameters in Robertson's platoon dispersion model are generally considered site-specific and dependent on the geometry of the link and other factors such as grades, curvature, parking

conditions, traffic volume, etc. (Robertson 1969). The work performed by Guebert and Sparks (1990) shows that the parameter calibration of platoon dispersion models significantly influences traffic signal coordination plans.

Since Robertson's platoon dispersion model is developed, many studies have contributed to calibrating its parameters (Seddon 1972; Collins and Gower 1974; Lam, Sasaki, and Yamaoka 1977; El-Reedy and Ashworth 1978; Tarnoff and Parsonson 1981; McCoy et al. 1983). Early studies collected data from field investigation and calibrated parameters by finding values that best fit between actual and predicted downstream traffic flow patterns. Based on the results of the studies conducted, recommendation values of parameters are summarized under different roadway conditions.

Yu (2000) proposed three equations to calibrates the three parameters in Robertson's platoon dispersion model using the average travel time and standard deviation of travel time. The three equations can be expressed as:

$$\beta = \frac{1}{1+\alpha}$$

$$\alpha = \frac{\sqrt{1+4\sigma^2}-1}{2T_a+1-\sqrt{1+4\sigma^2}}$$

$$F = \frac{\sqrt{1+4\sigma^2}-1}{2\sigma^2}$$

Where

σ = the standard deviation of travel time

With the development of sensors and simulation technologies, researchers started to manually extract information from videos or produce artificial data from simulations (Guebert and Sparks 1990; L. Yu 1999; Paul et al. 2018; Farzaneh and Rakha 2006; Bie et al. 2013; Y. Wang et al. 2017; Zhao, Rilett, and Tufuor 2017). The improved data collection process enabled researchers to achieve more information about the traffic flow and acquire more insights into how the process of platoon dispersion is affected by various factors.

To summarize, an accurate estimation/prediction of platoon flow rate is the key to improve the performance of arterial traffic control. Platoon dispersion models explain the mechanism of the platoon dispersion process and predict the downstream flow profile. If the vehicle arrival patterns are robust and predictable, delays and stops of vehicles can be minimized by assigning optimized offsets to signals along an arterial or throughout a network. Roberston's Platoon dispersion model is the most widely implemented. However, the parameters in the model are site-specific and dependent on the geometry of the link and other factors. Recalibration is required for different circumstances. Most of the calibration methodologies require dynamic traffic flow characteristics. Better recalibration of parameters in platoon dispersion model and better estimation of platoon flow using ALPR data may help to predict the vehicle's arrival. Besides, a survey shows video detection has emerged as a cost-efficient and reliable replacement for inductive loops (Stevanovic 2010). Although approximately 93% of the agencies use inductive loop detectors, almost half (43%) also use video detection. There is a potential of using these video detectors as ALPRs for better performance. In this dissertation, a new calibration procedure using ALPR data will be proposed, which can automatically calibrate the parameters of the platoon dispersion model using real-time ALPR data.

2.5 Cloud computing in Transportation

The term “cloud computing” was widely used after it was first introduced at an industry conference by Google CEO Eric Schmidt (Regalado 2011). Studies defined the word in a variety of ways. One definition of cloud computing is from Armbrust et al. (2010). They define cloud computing as both the application delivered as services over the Internet and the hardware and systems software in the data centers that provide those services. Another way to understanding cloud computing is to figure out the key features of cloud computing. Foster et al. (2008) highlighted the key points of cloud computing: 1) cloud computing is massively scalable 2) it is accessed as an abstract entity that delivers different services to end-users 3) It is driven by economies of scalable, and 4) it is dynamically configured and delivered on demand.

Generally, cloud computing has three different service models. Table 4 summarized the three types of cloud computing and listed a few examples of each type (Patidar, Rane, and Jain 2012).

Table 4 Types of cloud computing

Types of Cloud Computing	Description	Examples
Infrastructure as a Service (IaaS)	IaaS provides virtualized infrastructures, such as storage and processing power.	Amazon Elastic Compute Cloud (EC2), Google Compute Engine (GCE)
Platform as a Service (PaaS)	PaaS provides virtualized infrastructures and developing platforms over the Internet. Users can develop their applications over the platforms.	Windows Azure, Google App Engine
Software as a Service (SaaS)	SaaS provides software for end-users	Google Apps, Dropbox

Cloud computing is changing the way companies work. Since the innovation of cloud computing, many businesses have shifted their services to cloud computing or plan to do so. Harvard Business Review Analytic Services surveyed the state of cloud-driven transformation. In the survey, representatives from a variety of organizations are asked the state of cloud transformation and plan of transformation. 83% of representatives say that cloud computing is critical to the growth of their organizations. 69% of respondents say that 60% of the infrastructures of their organizations will be transformed to cloud in two years("The State of Cloud-Driven Transformation Keys to Accelerating and Capitalizing on Cloud Adoption" 2020). The driving factor behind this transformation is the advantages of cloud computing over self-owned local servers. Avram (2014) summarized the five key advantages of adopting cloud computing from a business perspective:

- Cloud computing lowers the cost of computing power. The cost of cloud computing are dynamically determined based on many computing power is being used. For applications that require intensive computations but do not constantly consume a lot of computing power, the dynamically calculated cost lowers their overall cost of computing power.

- Cloud computing provides immediate access to hardware resources remotely. With cloud computing, organizations can adjust the hardware of their server through the cloud computing platform at any time upgrades or downgrades are needed.

- Cloud computing lowers IT barriers to innovations. Cloud computing platforms are maintained by professionals in IT areas so that users can focus on the innovations themselves.

- Cloud computing enables originations to scale their services. Based on the demand of customers, the needs of computing power may be scaled. Cloud computing allows users to scale the services by adding/deleting instances of server nodes.
- Cloud computing makes new classes of applications possible. Two examples are (a) applications that respond in real-time (b) applications that require parallel batch processing.

Transporation applications are experiencing the same transformation. Bitam and Mellouk (2012) proposed a new cloud computing model called ITS-Cloud to improve transport outcomes. They noticed the improvement of outcomes with the load balancer in cloud computing. Jaworski Pawełand Edwards, Moore, and Burnham (2011) proposed a cloud computing-based urban traffic control system to increase road throughput and optimize traffic control. Li, Chen, and Wang (2011) reviewed the history of traffic control and management systems along with the computing technologies and concluded that cloud computing can help traffic management systems handle the large amounts of storage and computing powers required for traffic signal control and management. Trivedi, Deshmukh, and Shrivastava (2012) found that cloud computing helps to deal with the problems caused by the large-scale use of mobile agents. They proposed a prototype urban-traffic management system based on cloud computing. Nallaperuma et al. (2019) developed a machine learning platform for data-driven smart traffic management. It intended to integrate heterogeneous big data streams for real-time traffic analysis and adaptive traffic control. To summarize, cloud computing help transportation application in different aspects, such as better load balancing, performance improvement on large-scale problems and handy ways to manage storage and computing power.

The proposed system is chosen to be implemented on a cloud computing lab and the expected benefits of implementing the proposed system based on cloud computing are:

- *Easy switch to commercial cloud computing platforms*: Most organizations are in the stage of transformation to cloud computing. If any organizations plan to implement the proposed traffic diagnosis and management laboratory for real-world applications, the laboratory can be easily migrated to any commercial cloud computing platform.

- *Better scalability*: The proposed traffic diagnosis and management laboratory are deeply modularized. Users may assemble their system based on modules within the laboratory or modules built on their own. Besides, the required computing also depends on the data volume of users' applications. The scalability of cloud computing helps to meet the requirements of computing power in different circumstances.

- *User-friendly interface*: Cloud-based applications provide a consistent web interface for different digital devices and system platforms. Users can conveniently explore their idea in the proposed laboratory through the web interface.

2.6 Expected research contribution

In summary, this chapter presents a comprehensive review of existing studies on traffic condition reconstruction, transportation visual analysis, traffic control and management, and cloud computing in transportation. Limitations of existing studies and the potential contribution of this dissertation have been identified at the end of each section. The main expected research contributions are:

- Designing a comprehensive and extensible framework for traffic condition reconstruction, transportation visual analysis, and traffic control with ALPR data;
- Developing highly scalable and efficient algorithms for data cleaning, and traffic condition diagnosis;
- Implementing and designing advanced visualization tools for traffic condition monitoring and diagnosis; and
- Exploring the capability of utilizing ALPR data to improve the performance of data-driven traffic control and management.

Chapter 3: System framework

3.1 Overview

This chapter presents the overall framework of cloud-based traffic diagnosis and management laboratory. Figure 8 illustrates the framework and the data flow of the entire system. The system includes three different layers: data collection layer, data storage layer, and data processing layer. Components in lower layers have more interaction with hardware, while ones in upper layers have more interaction with users. It also shows the process of how raw data are transformed into information and how information is utilized in the traffic diagnosis and management laboratory.

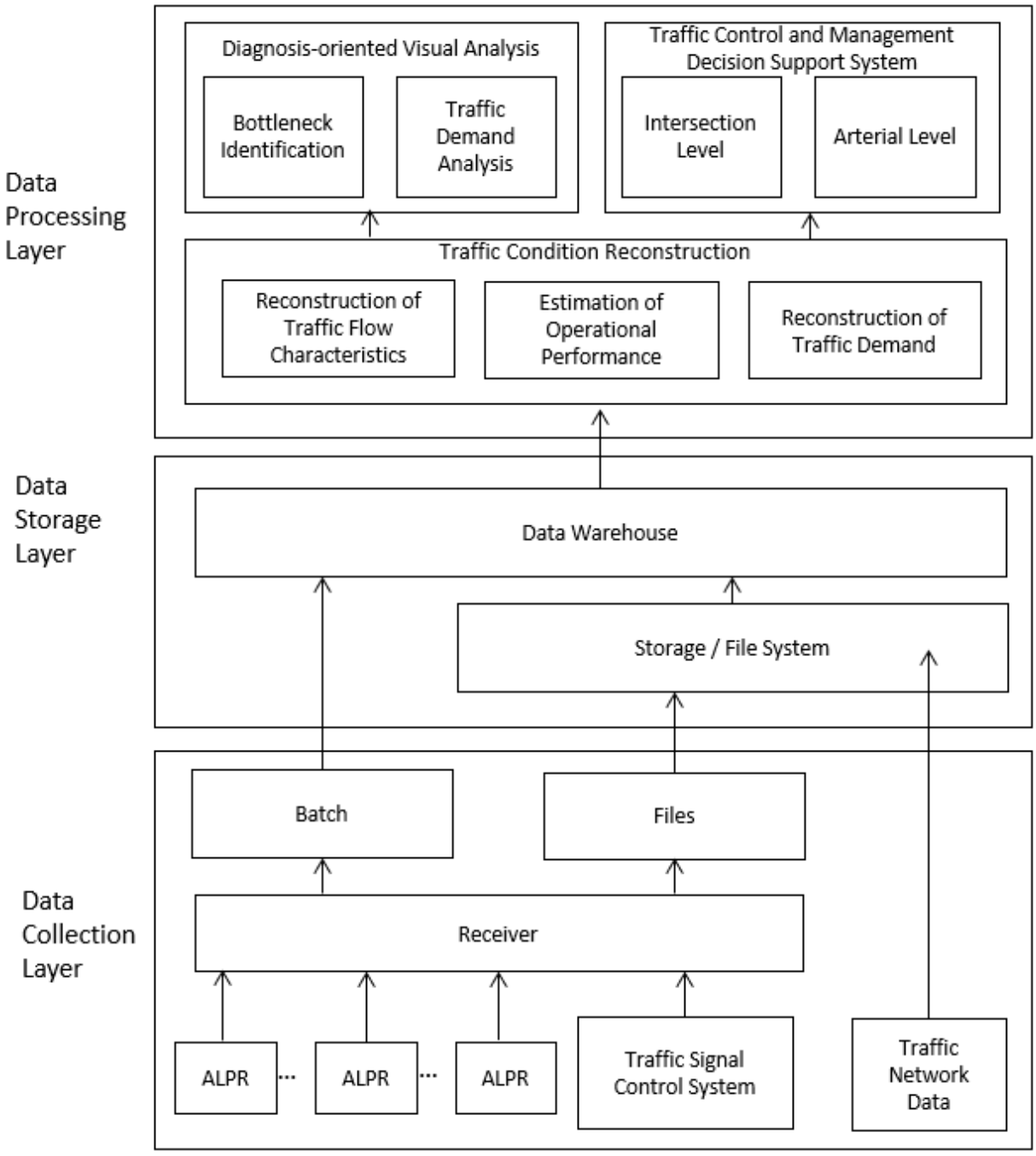


Figure 8 System framework

3.2 Data collection layer

The proposed system mainly consumes three types of data: ALPR data, traffic network data, and traffic signal data. For each data type, the collection processing, dataflow, and data standard will be discussed in 3.2.1, 3.2.2, and 3.2.3 respectively.

3.2.1 ALPR data

ALPRs are computer-controlled camera systems that collect license plate numbers, timestamps, and locations when vehicles come into specific areas. Some advanced ALPR systems can also capture other information of vehicles and their movements, such as lane ID, turning information, license plate color, and speed.

Based on how ALPR cameras are mounted, ALPRs can be divided into two groups: stationary ALPRs and mobile ALPRs. Stationary ALPRs are the ones mounted to a fixed location. In most cases, they are installed on traffic lights and monitoring in one direction. At one interstation, multiple ALPR cameras may be installed to monitor vehicles on multiple legs of intersections. On the other side, mobile ALPR cameras are generally attached to a car and used for law enforcement, such as detecting parking violations. Compared with mobile ALPR cameras, stationary ALPRs provide more stable and comprehensive information about traffic flow and all the discussions in this paper are about stationary ALPR.

ALPRs data are generated on the fly. Once a vehicle enters the view of ALPR cameras, one record is immediately created and uploaded to a local or central server. Generally, these records can be modeled in form of tables, in which each row represents one unique record, and each column represents one attribute, such as license plate number, time, or location. Relational databases or distributed file systems may be used to store and manage these tables. Even though the ALPRs

data are produced in real-time, when to produce these data could highly depend on the application. Traffic monitoring requires real-time data processing when they are uploads or in a near real-time manner, while OD estimation needs to wait for data to accumulate.

After ALPR data are collected from ALPR cameras, they will be sent to receivers. The main purpose of receivers is to receive data stream, preprocessing, and send out data to components in the upper layers. By having the receiver, the data stream can be better organized. ALPRs produce data in real-time, while not all the components in upper layers consume ALPR data in real-time. For example, origin-destination isis estimated on daily basis using daily ALPR data. In the proposed system, the receivers send out data in two different ways.

- When a record is received, the receiver will put it into a buffer (mini-batch). All the records within the buffer will be sent to the traffic monitoring module after a specified time interval. The time interval can be set up in a way that fits the volatility pattern of traffic flow conditions and data velocity. If traffic flow conditions fluctuate frequently and data come into the buffer very fast, a shorter time interval may help to reflect the changes in flow condition and avoid too much data stay in a buffer.

- Meanwhile, the receiver will also save the record into a local file system. These local files will be sent to the traffic control and management module every day.

Based on data volume and budget, the receiver can be implemented with a personal computer, a powerful workstation, or a cloud computing service. Figure 9 presents a receiver implemented with multiple servers. In ideal cases, receivers should be designed as a group of servers with a load balancer. In some cases, the ALPR system consists of large sets of ALPR cameras deployed over the traffic network. The ALPR system may produce a huge amount of data in a very short period.

The load balancer balances the workload of every server. The overall computing power of this group of machines is designed to be higher than the demand for processing data streams. Even if one or two servers are stopped, the entire system will still be functional. There is also a tradeoff between reliability and cost. Extra computing power helps to increase the reliability of the entire system but will also increase the cost.

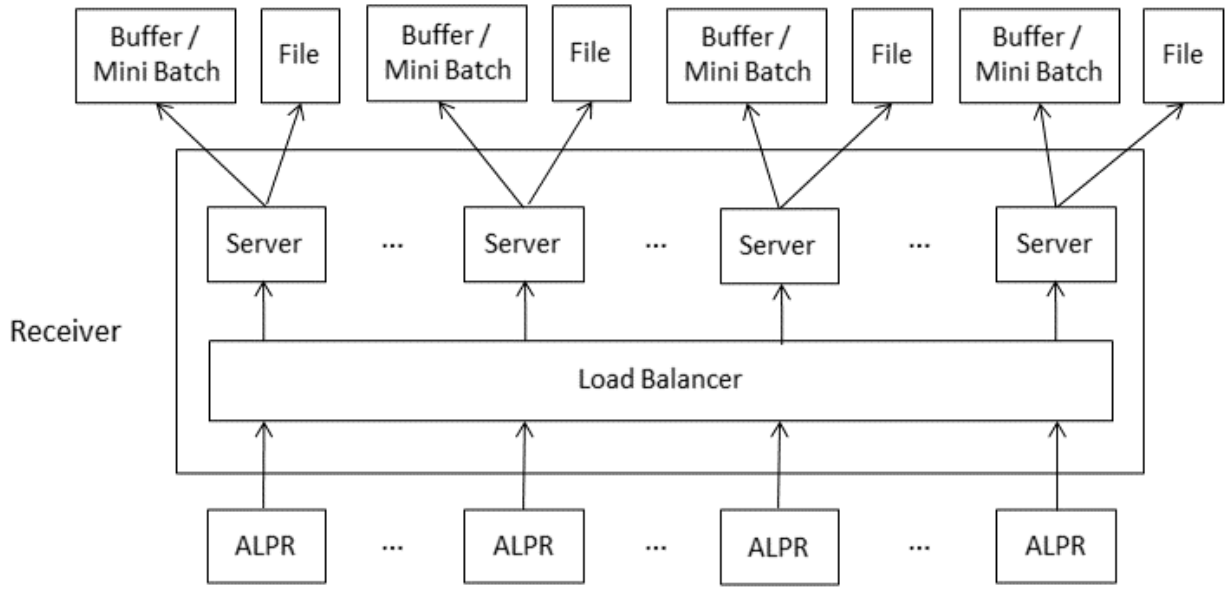


Figure 9 Receiver

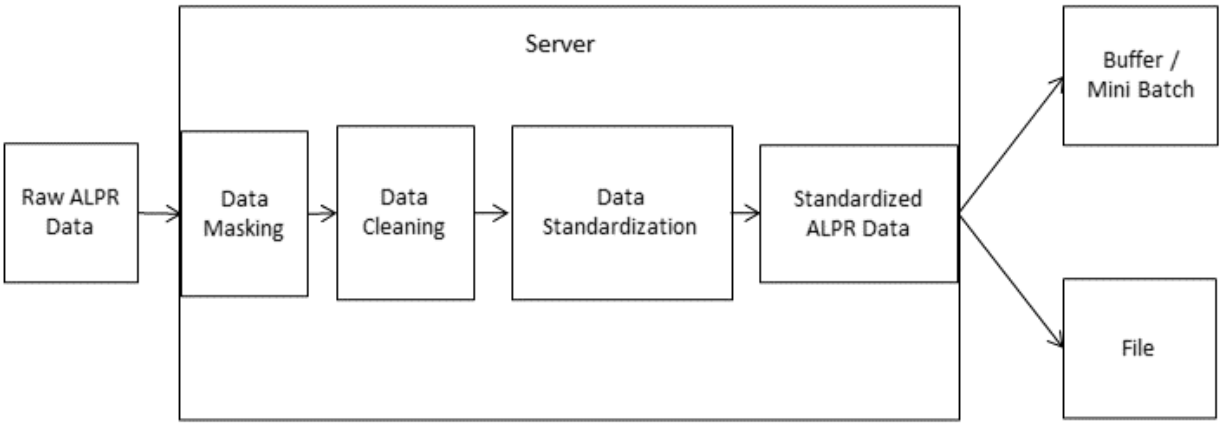


Figure 10 Data flow within a server of receiver

Figure 10 illustrates the data flow within a server of the receiver. ALPR data are masked, cleaned, and standardized before being sent out to the buffer and local files. Because the data set contains real license plate number, sensitive information in the data set have been masked to avoid privacy concerns. The data masking used in the system is a nonreversible transformation. So even if someone knows how these data are encrypted, the original data remain unknown. Data masking not only encrypts the data but may also improve the performance of the system by using numbers instead of a long string. Most operations have better performance on numbers than long string, such as matching. The algorithm of data masking may also be designed to boost the preformation of specific operations if those operations are dominant in the entire system. In this paper, all personally identifiable information is masked to avoid privacy issues and several other attributes, such as turning directions, the color of the license plate are masked for better performance. In this study, SHA1 is applied for data masking.

Data standardization is the process of transforming data into a uniform format in which data can be utilized more efficiently. Meanwhile, it is also the process of creating consistent definitions on the data model. The data model describes different perspectives of the data:

- Elements: Elements specifies the real-world entities included in the model and how they are related. ALPR data consists of three entities: ALPR camera, vehicle, and ALPR record. Record is the entity that connects ALPR and vehicle.

- Attributes: Attributes determine the properties stored to describe the elements in the model. The attributes of the ALPR camera include ALPR ID, location, related entity. ALPR ID is the unique identifier for an ALPR camera in the ALPR system. Location and related entity clarify the geographical information of the ALRP camera, which help to match

geographical information in traffic network data. The attributes of the vehicle include license plate number, license plate color, make, model, color. The attributes of the ALPR record include time, ALPR camera ID, vehicle license plate number, vehicle speed, turning direction, and lane number.

- Data Type: Data type specifies how the program will treat the data. It depends on how the program intends to use these data. In other words, the data type is decided by operations, such as mathematical calculation, matching, comparing, and ordering. Data types are crucial to maintaining the consistency of the data because each data type has its format. Besides, data types determine the data domain, which defines a set of values that is the attributes may contain. As a typical unique identifier, license plate numbers are strings that have already been standardized. However, regarding privacy concerns about misusing ALPR data, license plate numbers become integers after been encrypted in the process of data masking. Time stamps or locations may not be in standard formats for analysis. A standard format for time should be easily recognized by major database software and ready for cost-efficient data processing. Besides, locations may be decomposed into their components, which may include intersection name, leg, and lane. To achieve higher time efficiency, an artificial ID may be created for each intersection.

The following tables present the general standard of ALPR data. The attributes are divided into two categories: Basic and Advance. Basic attributes realize the core functionality of ALPR and all ALPR system should have these basic attributes. They record the time, the location, and unique identifier of vehicles. Because of the improvement of the camera and image identification algorithm, the ALPR system can provide more detailed information about vehicles such as the color of license plates. They are not the key to understanding the spatial-temporal distribution of

vehicles but may still provide insights if used properly. For example, the color of license plates may tell the type of vehicle, which can be utilized to calculate the heavy vehicle ratio. This discussion of the proposed system mainly focuses on the basic attributes since the proposed system is designed for general ALPR systems. Advanced will be briefly discussed if it can be used to improve the efficiency or accuracy of the system.

Table 5 Data format of ALPR camera attributes

Attributes	Data Type	Description	Example
ALPR_ID	int	The unique identifier for the ALPR camera.	3120392
LOCATION	a tuple of two floats	The license plate number of the recorded vehicle	(31.2990, 120.5853)
RELATED_ INTERSECTION	int	The id of intersection that is covered by the ALPR camera	1

Table 6: Data format of ALPR record attributes

Attributes	Data Type	Description	Example
ALPR_ID	int	The unique identifier for ALPRs	3120392
PLATE_NUM	int	The license plate number of the recorded vehicle	5239182
TIME_STAMP	datetime	Time when the vehicle gets recorded	2017-01-04 12:00:22.455000
DIR_INFO	int	The turning direction of the vehicle	1
LANE_ID	int	The unique identifier for the lanes	2
SPEED	int	Speed of the vehicle when it is recorded	1

Table 7 Data format of vehicle attributes

Attributes	Attributes	Description	Example

PLATE_NUM	int	The license plate number of the records vehicle	5239182
PLATE_COLOR	str	The license plate color of the records vehicle	blue
COLOR	str	Vehicle color	grey

The accuracy of the estimation depends on the quality of the data. Although in many cases, the quality of raw data is pre-determined or limited by budgets, data cleaning improves the quality of data before they are fed to the estimation models. There are no universal rules in the field of data preprocessing. Different datasets may require different methodology or transformation. However, a good data cleaning starts with understanding the nature of the datasets, such as data structure, data quality, possible errors, and outliers. The rest of this section will discuss a few issues that generally existed in ALPR data and corresponding solutions. This step aims to transform data into the desired format for data processing. Raw video surveillance data consist of records including license plate number information, time stamp, location name, speed, and direction.

After standardization, data are in formats ready for processing. Although image separation and recognition technology have been well-developed in recent years, mistakes are inevitable. Data cleaning and data recovery are fundamental steps to avoid errors, such as repeated records, missing records. The more errors are corrected, the more accurate estimations will be. Efficiency is also considered in the data cleaning procedure because traffic analysis and diagnosis need to react to traffic changes quickly. One type of mistake, defined as repeated records, is recognized by records appearing more than once. Fortunately, these repeated records are always adjacent to each other, which means that they can be found and fixed by going through all records only once. Another type of mistake is missing records caused by detection failures of ALPR or intersections uncovered

by ALPR. As mentioned before, the license plate number makes it possible to reconstruct individual vehicle trajectories. However, due to missing records, some vehicle trajectories are separated into multiple segments. It becomes easier to rebuild the entire trajectories with the topology of the traffic network. The topology of the network alone can reconstruct an individual trajectory only when no more than one missing record exists between two segmented trajectories of one vehicle. When two or more records are missing between the segmented trajectory of a vehicle, a model or algorithm is needed. In our proposed system, a statistical model is built up to reconstruct the entire trajectory and will be illustrated in the following section.

3.2.2 Traffic network data

Traffic network data provide topology of the traffic network and location information of ALPRs, intersections, and roads. Such network structures and location information can be achieved in different ways. Free traffic network data can be extracted from many different open-source geographical data projects, e.g. official websites of governments, OpenGIS platforms, and APIs such as Google Maps and OpenStreetMap. Most network data from these open-source projects are designed for general purposes and may not fit specific projects directly. Field survey can be a choice if the project requires high accuracy and have an adequate budget. Multiple data sources may be used for validation to improve accuracy and reliability. The structure and format of network data vary.

Most of the traffic network data downloaded from the Internet are stored in tree structure files, such as XML (Extensible Markup Language) files, which include both topology and location information. Two types of structures are generally used to store topology information. They are the adjacency list and adjacency matrix. In numerical analysis, the level of sparsity is measured by the density of zeros in a matrix. Since an intersection is always connected to its neighboring ones,

adjacency matrices of the traffic network will be of high sparsity, which results in unnecessary waste of storage. Therefore, adjacency lists are commonly used in storing traffic network information due to lower storage consumption. The proposed system standardized network data as table 2:

Table 8 Data format of intersection attributes

Attributes	Data type	Description	Example
INTERSECTION_ ID	int	The unique identifier for intersections	000001
INTERSECTION_ NAME	str	The name of intersections	
LATITUDE	float	The latitude of interstation's	31.299
LONGITUDE	float	The turning direction of the vehicle	120.583
SIGNAL_TYPE	int	The signal type of intersections	1
ADJANCE_LIST	List	The ID of intersections that are adjacent to this intersection	[000001, 000002, 00003]

Table 9 Data format of road attributes

Attributes	Data type	Description	Example
ROAD_ID	int	The unique identifier for roads	000001
ROAD_NAME	str	The name of roads	
START_NODE	int	The starting intersection ID of the road	000001
END_NODE	int	The ending intersection ID of the road	000002
ROAD_LENGTH	float	The length of the road	510
NUM_LANES	int	The number of lanes of the road	3

3.2.3 Traffic signal data

Traffic signal data are generally extracted directly from traffic control systems. The data structure of these data depends on the specific traffic control system. In the proposed system, the signal data is stored in four tables. The relationship diagram of signal control data is presented in Figure 11.

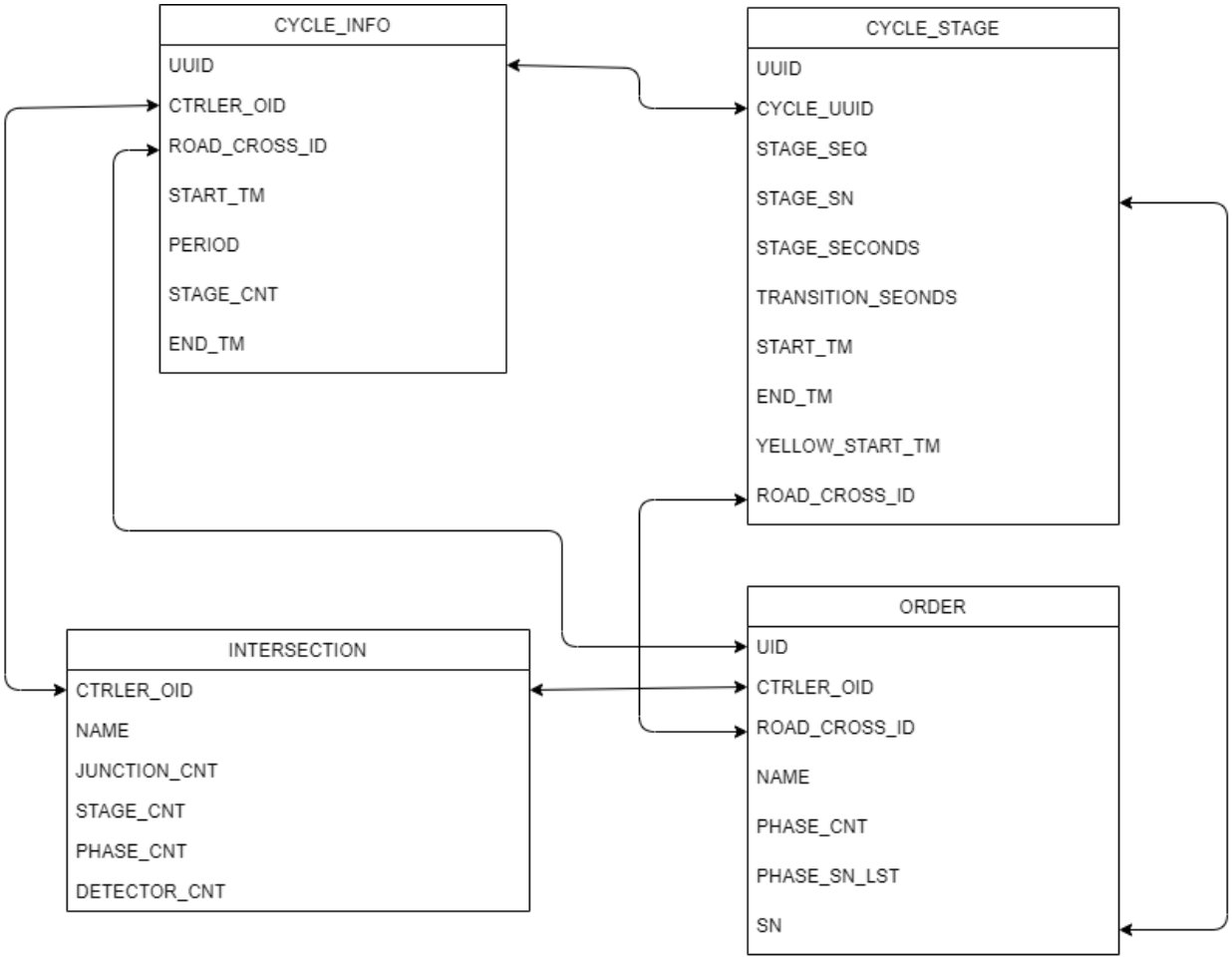


Figure 11 Relationship diagram of signal control data

Whether traffic signal data needs to go through the receiver depends on which type of traffic signal control system is used. If fixed signal control is used, traffic signal data will be directly stored in a file system. Otherwise, actuated signal control plan will be sent to a receiver.

3.3 Data storage layer

The data storage layer consists of a distributed file system and a data warehouse. The distributed file system maintains all the historical data in files. The proposed system uses Hadoop, a distributed file system, to store all the data including the ALPR data, traffic signal data, and traffic network data. Hadoop is one of the most popular open-source frameworks of distributed file systems. The file system is the foundation of the entire system. There are four reasons why a distributed file system is utilized in the system.

- Horizontal Scalability: It is easy to add additional nodes and devices to the distributed file system. The ALPR systems of many cities and regions start with covering a small area and continuously growing. The growing number of ALPR sensors produces a larger volume of data and leads to larger storage needs. A distributed file system provides the flexibility to add more storage and other devices easily.

- Cost-efficiency: Distributed file systems utilize large numbers of commodity hardware and fully take advantage of already available computing components.

- Reliability: Distributed systems are fault-tolerant as they can be made up of hundreds of nodes that work together. The system generally doesn't experience any disruptions if a single machine fails. The proposed system could be implemented for traffic monitoring, management, or even adaptive signal control, it is crucial to have a robust file system with fault tolerance.

- Performance: Distributed systems are extremely efficient because work- loads can be broken up and sent to multiple machines. Even though the entire traffic network is a complicated system and different parts of the network dynamically affect each other, most

of the traffic flow estimation can be done in parallel. In other words, the connections within the network are weak to some extent. Most nodes (intersections) in the traffic network are not directly connected. They are only connected to nodes that are geographically close to themselves. It also means estimating the movements of vehicles or in general, the characteristics of traffic flow within the sub-network is independent of estimating ones within other subnetworks. In the proposed system, distributed algorithms are designed for traffic flow characteristics estimation.

Data warehouse aims to integrate data flows from different sources so that the data are well-organized and can be directly fed to modules in the data processing layer for traffic diagnosis and management. The proposed system utilizes multiple data sources, such as ALPR data, signal control data, and traffic network data. Extract, Transform and Load (ETL) is designed to collect data from files and receivers to maintain the consistency of the data warehouse.

3.4 Data processing layer

The data processing layer consists of a traffic monitoring module and a traffic signal control and a management module. Traffic monitoring module estimate traffic flow characteristics in near real-time and reflect the changes of traffic flow condition with visual analysis tools. Once issues are recognized, the traffic signal control and management module will be triggered, and start looking for corresponding solutions. These components are the focus of this study and are discussed in chapter 4, 5, and 6.

Chapter 4: Traffic condition reconstruction

4.1 Introduction

Traffic condition reconstruction is the procedure of extract and analyzes traffic condition measurements from data. The ability to measure traffic conditions is crucial to traffic diagnosis and management. In the proposed diagnosis and management laboratory, traffic condition reconstruction is divided into three parts: traffic flow characteristics estimation, performance estimation, and traffic demand estimation. Traffic condition reconstruction is the cornerstone of the proposed laboratory. Bottleneck detection, travel demand analysis, driving pattern recognition, and traffic control support system all rely on the results of traffic condition reconstruction to accomplish their tasks. The accuracy of the traffic condition reconstruction will directly determine the effectiveness of the entire system. Meanwhile, traffic monitoring also has a real-time requirement for data processing. So, the trade-off between accuracy and time consumption is considered. This chapter is organized as follows:

- Section 4.2 lists the mathematical notations used in this chapter.
- Section 4.3 presents the measures of traffic flow characteristics and methodologies for traffic flow characteristics estimation using ALPR data. A clustering-based algorithm is proposed for link travel time estimation.
- Section 4.4 discusses the measures of intersection performance and methodologies for performance estimation.

- Section 4.5 illustrates the estimations of traffic demand on links, routes, and over networks. A scalable trajectory-reconstruction algorithm is proposed to rebuild the trajectories of vehicles using ALPR data.

- Section 4.6 performs a field application on the proposed methodologies.

4.2 Mathematical notation

V_i = vehicle i

I_i = intersection i

$L_{i,j}$ = link from I_i to intersection I_j

C_i = ALPR camera i

$C_{i,j} =$ ALPR camera $at\ I_j$ and its upstream intersection is I_i

$\boldsymbol{G}_{I_i}$ = set of green time at I_i

$\boldsymbol{S}_{I_i}$ = set of starting time of green intervals at I_i

$\boldsymbol{V}_{I_i}$ = set of vehicles detected by any ALPR camera at I_i

$\boldsymbol{V}_{C_i}$ = set of vehicles detected by C_i

$\boldsymbol{V}_{L_{i,j}}$ = set of vehicles on $L_{i,j}$

$\boldsymbol{D}_{I_i}$ = set of downstream ALPR of I_i

$\boldsymbol{C}_{I_i}$ = set of ALPR cameras that cover different approaches of I_i

α = sample rate of ALPR cameras

ADJ = adjacency list of traffic network

$ADJ(I_i) =$ the list of intersections adjacent to I_i

$N_i =$ sub − network i

$F_i = csv\ file\ i$

$R_i =\ ALPR\ record\ i$

$LPN(R_i) =$ license plate number $within\ R_c$

$TS(R_i) = timestamp\ within\ R_c$

$ALPR(R_i) = ALPR\ ID\ within\ R_c$

UI(ALPR) = upstream intersection of ALPR

DI(ALPR) = downstream intersection of ALPR

W = analysis time window

t_{V_k,C_x} = the timestamp when ALPR camera x detects vehicle k

$l_{L_{i,j}} =$ length of link ij

T = trajectory

L = likelihood

X = event

4.3 Reconstruction of traffic flow characteristics

4.3.1 Travel time

Most of ALPRs are set up in two different ways, which affect the detecting area and detecting behavior of ALPRs.

1. ALPRs are set up before stop lines and direct toward approaching. The vehicles will be detected when they are approaching the intersection. This is the general way to set up ALPRs for adaptive traffic control.
2. ALPRs are set up at the intersection and direct intersections. Vehicles are detected when they enter the intersection. This the classic way to set up ALPRs for law enforcement. Knowing when vehicles entering intersections and signal plan helps to identify if vehicles pass the stop line while the light is red.

According to how ALPRs are set up, the meaning behind the travel time of vehicles between ALPRs is quite different. It will also affect how to use ALPR data to estimate travel time and other related measurements. In this section and the following sections, which ALPR setup is used for estimation will be specified if the methodology does not fit both cases.

Figure 12 shows the locations of ALPRs and corresponding detecting areas for monitoring approaching vehicles. I_i and I_j are two adjacent intersections. The time-space diagram focuses on traffic flow moving from I_i to I_j. ALPR cameras C_x and C_y are set up at I_i and I_j to monitor approaching traffic. They are indicated by blue rectangles in the diagram. The signal plans of corresponding phases of two intersections are indicated by green and red horizontal bars. The trajectories of Vehicle 1 and Vehicle 2 represent two general trajectories cases in the traffic flow. In the diagram, the left solid blue line is the trajectory of Vehicle 1. The right solid blue curve is the trajectory of Vehicle 2. Vehicle 1 goes through I_i without a stop. Vehicle 2 hit red when it

approaches I_i. So, it decelerates and stops before the stop line. After the signal light turns green, it accelerates and moves toward intersection j.

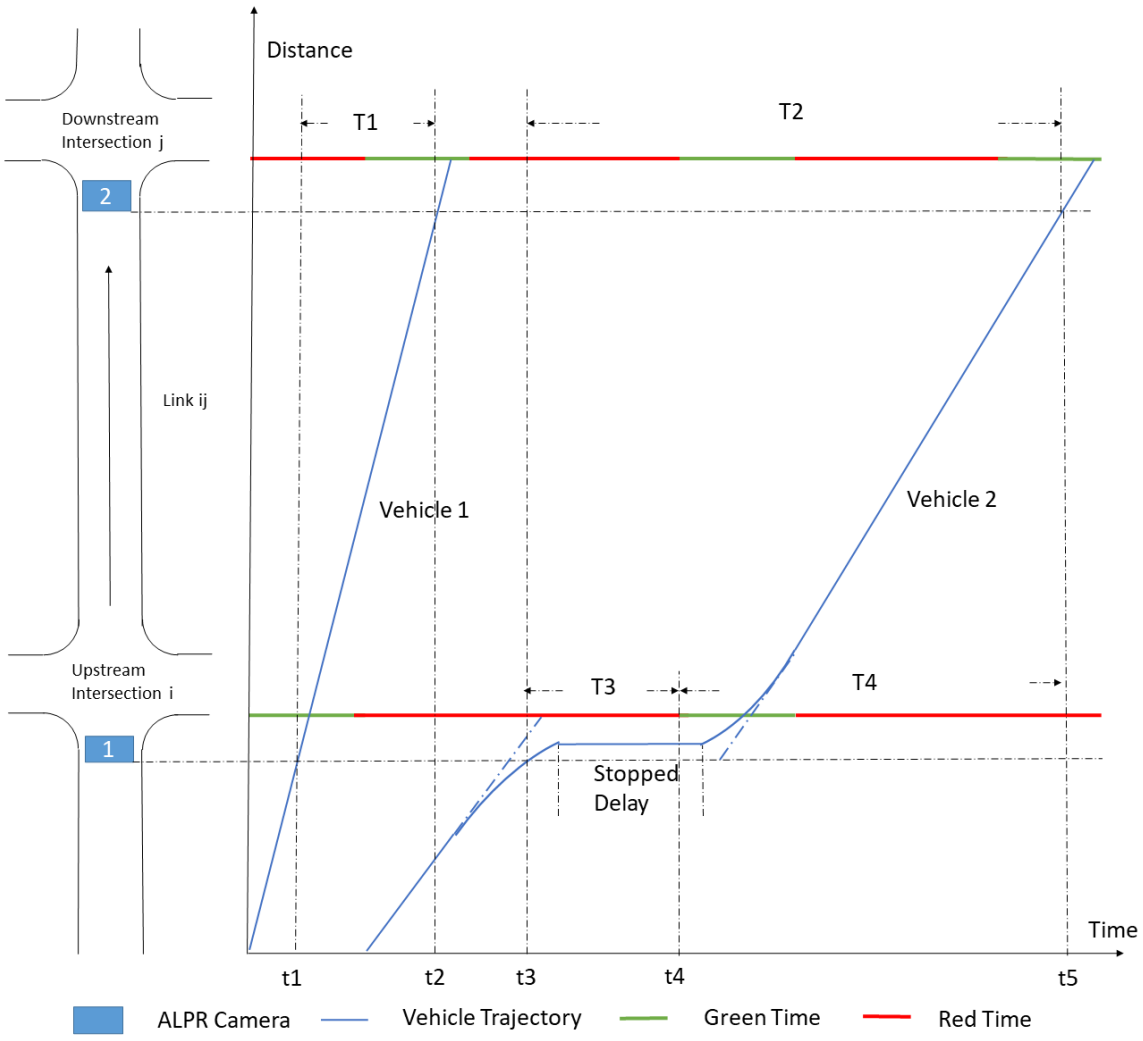


Figure 12 ALPR setup (Case 1)

Assuming no detection errors happen, two records will be produced for each trajectory. The records about Vehicle 1 contain timestamp t1, t2, and ALPR ID of C_x and C_y . The records of Vehicle 2 are associated with timestamp t3, t5, and ALPR ID of C_x and C_y . The time difference T1 and T2 are the measurements of travel time of Vehicle 1 and Vehicle 2 moving from C_x to C_y. Compared with the length of $L_{i,j}$, the distance between the intersection and detection area is

neglectable. So, the time difference between ALPR cameras is a good measurement of the travel time between intersections.

Besides, the travel time between two adjacent intersections consists of control delay at I_i and travel time on the $L_{i,j}$. Generally, the overall intersection delay includes slow deceleration delay, stopped delay, and acceleration delay. For vehicle 1, it is not affected by the signal control at intersection 1 and the control delay at I_i is 0. For vehicle 2, the time difference T3 between t3 and the start time of the next green t4 measure the stopped delay of Vehicle 2 at I_i. In the proposed system, it is also used a rough estimation of the control delay and T4 is used to estimate the travel time on $L_{i,j}$.

Based on the discussion above, the following travel time measures are derived.

- The travel time of V_k between intersection I_i and intersection I_j is measured using the travel time of V_k between the ALPR camera C_x at the upstream intersection I_i and the ALPR camera C_y at downstream intersection I_j:

$$TravelTime_{I_i,I_j,V_k} = t_{\mathrm{V_k,C_y}} - t_{\mathrm{V_k,C_x}}$$

- The control delay of V_k at I_i is measured using the stopped delay of V_k at I_i:

$$\mathrm{Delay}_{I_i,V_k} = \begin{cases} min_{t \in \boldsymbol{S}_{I_i}}(t | t > t_{\mathrm{V_k,C_x}}) - t_{\mathrm{V_k,C_x}} \; if \; t_{\mathrm{V_k,C_x}} \notin \boldsymbol{G}_{I_i} \\ 0 \; if \; t_{\mathrm{V_k,C_x}} \in \boldsymbol{G}_{I_i} \end{cases}$$

  $\boldsymbol{S}_{I_i}$ is the set of values consisting of the all starting point of green periods ofI_i. C_x is the ALPR camera setup at I_i that detected V_k. $min_{t \in \boldsymbol{S}_{I_i}}(t | t > t_{\mathrm{V_k,C_x}})$ is the starting point of the next green period.

- The travel time of vehicle k on $L_{i,j}$

$$TravelTime_{L_{i,j},V_k} = TravelTime_{I_x,I_y,V_k} - \mathrm{Delay}_{I_i,V_k}$$

ALPR x and ALPR y is the ALPR camera setup at intersection i and intersection j that detected vehicle k.

- The Average Travel time on $L_{i,j}$

  Vehicles may park off-street on $L_{i,j}$, while will produce abnormally long travel time on $L_{i,j}$. These outliers need to be identified before calculation or using estimators that are not sensitive to outliers, such as median or other percentiles. The proposed system implements median absolute deviation (MAD) to find outliers in the travel time data samples. Outliers are stored in TravelTimeOutliers for trajectory reconstruction.

$$AvgTravelTime_{L_{i,j}} = \frac{\sum_k TravelTime_{L_{i,j},V_k}}{\text{FlowRate}_{L_{i,j}}}$$

- The average travel time from I_i to I_j

$$AvgTravelTime_{I_i,I_j} = AvgTravelTime_{L_{i,j}} + AvgDelay_{I_i}$$

- The variance of travel time from I_i to I_j

$$VarTravelTime_{I_i,I_j} = \frac{\sum_k \left(TravelTime_{I_i,I_j,V_k} - AvgTravelTime_{I_i,I_j}\right)^2}{\text{FlowRate}_{L_{i,j}} - 1}$$

The discussion above assumes: (1) If vehicles are stopped by intersections, they will all stop within the detection area of ALPR cameras. (2) All stopped vehicles at intersections pass intersections within one cycle length. (3) No vehicles stopped on $L_{i,j}$. If the first assumption does not hold, the control delay will be underestimated while link travel time will be overestimated. But since ALPR can detect vehicles 35 meters away ("Guidelines for ANPR Camera Positioning & Set Up," n.d.). This assumption should hold in most circumstances. If the second assumption is violated, the delay

will be heavily underestimated and it often happens when the traffic demand is high. If the third assumption is violated, a vehicle stopped on the link will heavily affect the link travel time estimation. It will be biased because of these outliers.

Figure 13 shows the locations of ALPRs and corresponding detecting areas when ALPRs are set up facing the intersection instead of approaching vehicles. The main difference is that the travel time between ALPRs consists of link travel time and control delay of downstream intersections instead of delay of upstream intersections. Besides, because the arrival time of vehicles at downstream intersections is unknown. The measures used for the first setup of ALPRs can not be applied under this circumstance. Based on the observations, a clustering-based methodology is designed to estimate the link travel time. It also applies to the case when ALPRs are set up towards approaching vehicles. It helps to identify outliers and better estimate delay time when the intersection is oversaturated.

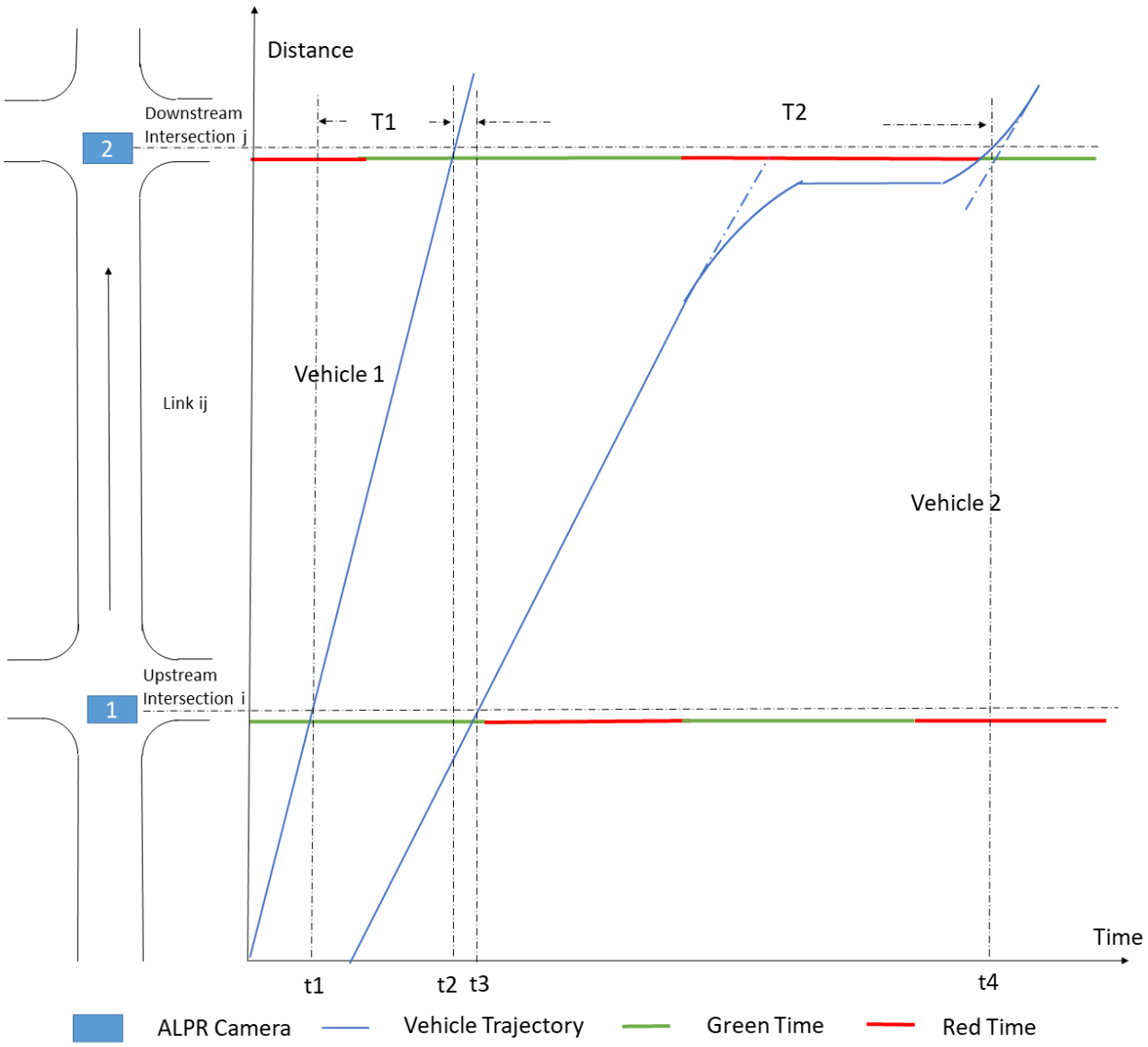


Figure 13 ALPR setup (Case 2)

This clustering-based approach is based on observations on enormous ALPR data. For each signal cycle of upstream intersection, the through traffic can mainly be divided into three groups:

- The vehicles that arrive at the downstream intersection when the signal is red. They stopped at the downstream intersection and formed a queue. The queue discharge when the signal turns green. As the queue discharge, the vehicles continuously enter the downstream intersection and get captured by the ALPRs. The headway is generally small when queue

discharge. So the time differences between these records are very small. Besides, because they are stopped by the downstream intersection, these vehicles experience high travel time.

- The vehicles that arrive at the downstream intersection when the signal is green and there is no existing queue. These vehicles entering the intersections immediately and get captured by ALPRs. The headway of these vehicles depends on the dispersion of the platoon and should be smaller than the headway when queue discharge.
- The vehicles that arrive at the downstream intersection when the signal is green. But the existing queue prevents the vehicles from entering the intersection during the green time. These vehicles will wait for the next green time. As the queue discharge during the next green time, these vehicles enter the intersection and are captured by ALPR. Similar to the first group of vehicles, The headway is expected to be small. And these vehicles experience long travel time.

So the key to estimating the link travel time is to recognize the first group of vehicles from the data. They are not affected by signal control. Their travel time can be used for link travel time estimation. Apart from the main groups, there are also some exceptions:

- The vehicles that temporarily stop at a street-side for a while and reenter the traffic. Depend on how much time they stopped at street sides, the travel time of these vehicles varies. If they reenter the traffic quickly, they could travel time will be similar to other vehicles. Otherwise, they may have travel time much longer than other vehicles. These vehicles should be treated as outliers for link travel time estimation and delay estimation.
- A small group of vehicles arrives at the downstream intersection when the light is red but gets captured immediately. This situation happens when they are too close to the stop line

or they stopped after the signal line. These samples tend to have a similar travel time as the vehicles in the first group.

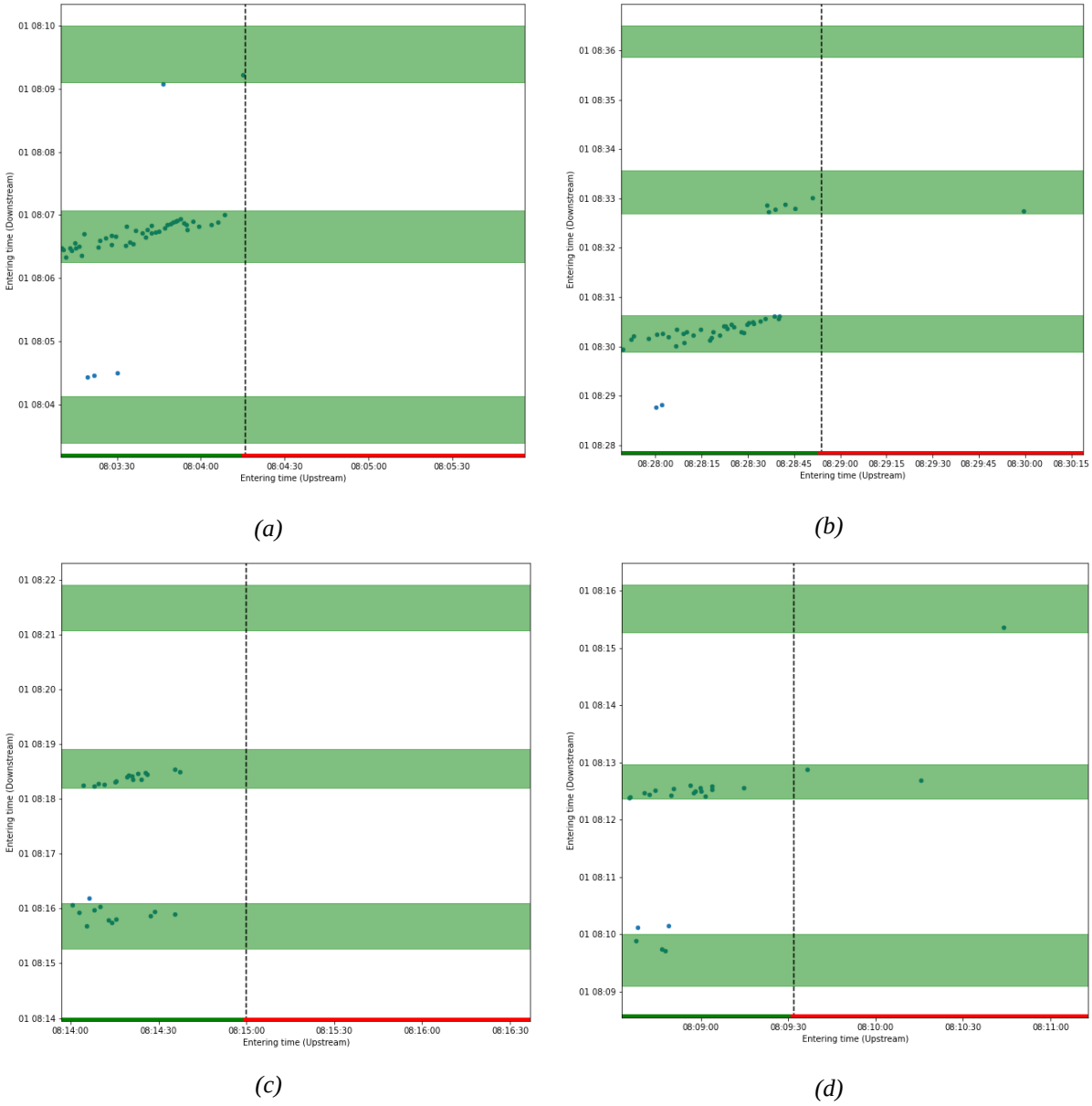


Figure 14 Upstream record time – downstream record time diagram

Figure 14 presents the records time of vehicles at upstream ALPRs against the records time of vehicles at downstream ALPRs. Each figure focus on the vehicles that enter the upstream intersection at one signal cycle. The horizontal green bar and the red bar at the x-axis represent the green time and red time for these vehicles at the upstream intersection. The green rectangle

represents the green time at the downstream intersection. In Figure 14and Figure 14b, vehicle records show a strong linear relationship in the diagram. The variance of recording time at downstream intersection keeps relatively stable. These records are close to each other along the y-axis. These features match the expectation of the queue discharging procedure. In Figure 14c and Figure 14d, a part of the vehicles entering the downstream intersection at the earliest green interval. Compared with the pattern in Figure 14 and Figure 14b, these data points are relatively sparse and no linearity is found. The rest of the vehicles entering the downstream intersection in the next green interval. The data points of these again show a similar pattern as data points in Figure 14 and Figure 14b. These patterns shown in these four figures can be easily recognized in most data samples.

Based on these observations, a clustering-based approach is designed. The proposed clustering-based algorithm for link travel time estimation is summarized in Figure 15. Given a time period, the data samples among different signal cycles are stacked to increase the number of data samples. The expectations are:

- Because the first group of vehicles is not affected by signal control. They will form a cluster even if they belong to different signal cycles. The variance along the y-axis is expected to be larger than the ones observed in the queue discharge procedure.
- The second and third groups of vehicles in each signal cycle will appear as one cluster. The cluster has a linear relationship.
- If two signal cycles have very similar offsets with the downstream intersection, the clusters of stopped vehicles in these two signal cycles will merge.

Based on this expectation, the number of clusters is initialized as one plus the number of signal cycles that do not have similar offsets with downstream intersections. The initialized number of clusters a few other close numbers are used to run the K-means clustering algorithm. Based on Silhouette Score, the best number of clusters is chosen. Then the cluster of vehicles with the smallest average record time is recognized as the cluster of vehicles that are not affected by the traffic signal. The average link travel time is calculated as:

$$AvgLinkTravelTime_L = \frac{\sum_{V_k \in S_n} t_{d,V_k} - t_{u,V_k}}{\sum_{V_k \in S_n} 1}$$

where

$t_{u,v}$ = the downstream record time of vehicle v

$t_{d,v}$ = the downstream record time of vehicle v

S_n = the cluster of vehicles that not affected by traffic control

Algorithm 1: Link travel time estimation

INPUT

- A time interval (T1, T2)
- ALPR records of upstream intersection UpstreamRecords
- ALPR records of downstream intersection DownstreamRecords
- Phases duration and starting points for through traffic at upstream intersection UpstreamPhaese
- Phases duration and starting points for through traffic at downstream intersection DownstreamPhaese

OUTPUT

Link travel time between upstream intersection and downstream intersection

ALGORITHM

$UpstreamDict \leftarrow initalize\ hashmap\ storing\ key\ vlaue\ pair\ of\ records\ (Veh : UpstreamRecord)\ from\ upstream\ ALPR\ sensors\ within\ time\ interval\ (T1, T2)$

$MatchedRecords \leftarrow initialize\ empty\ hashmap\ for\ storing\ matched\ reocrds$

$Traversal\ the\ records\ of\ downstream\ ALPRs\ and\ add\ matched\ records\ to\ MatchedRecords$

$ThruTimePair \leftarrow all\ Matched\ Records\ indicating\ no\ turns\ from\ upstream\ intersection\ to\ downstream\ intersection$

$Grouping\ the\ time\ pairs\ in\ TimePair\ by\ which\ signal\ cycle\ the\ vehicle\ enter\ the\ upstream\ intersection$

$For\ each\ t1,\ t2\ in\ ThruTimePair$

$$Do \begin{cases} tt1 \leftarrow t1 - starting\ time\ of\ the\ phase \\ tt2 \leftarrow t2 - starting\ time\ of\ the\ phase \end{cases}$$

$Group\ upstream\ signal\ cycles\ by\ their\ offests\ with\ downstream\ intesrection$

$n \leftarrow number\ of\ groups\ of\ upstream\ signal\ cycles$

$Perform\ Kmeans\ clustering\ with\ n+1\ clusters$

$Cluster \leftarrow\ the\ cluster\ with\ lowest\ mean\ on\ tt1\ and\ tt2$

$LinkTravelTime \leftarrow \bar{tt2} - \bar{tt1}$

$returnLinkTravelTime$

Figure 15 Link travel time estimation

4.3.2 Traffic volume

The number of records of ALPR cameras can be used as traffic counts over the detection area. To remove the bias between the number of records and real traffic volume, estimations of sampling

rates are required. The sample rate of ALPR depends not only on the video processing methodology but also on factors such as traffic conditions, lightness, and weather. Detailed discussion about how these factors affect sample rates is beyond the scope of this study. The following discussion of sampling rate focus on how to measure or estimate the sample rate of ALPRs. The data collection procedure of ALPR consists of two separate stages: detecting license plate and recognizing license plate number in the detected license plate. Two ratios are used to measure the sampling rates of these two stages:

- Detection Ratio (DR): the ratio between the Number of Detected License Plates (NDLP) and the real traffic volume over the detection area of the ALPR camera C_i.

$$DR_{C_i} = \frac{NDLP_{C_i}}{Traffic\ Volume_{C_i}}$$

- Recognition Ratio (RR): the ratio between the Number of Correctly Recognized License Plate (NCRLP) and the Number of Detected License Plates (NDLP).

$$RR_{C_i} = \frac{NCRLP_{C_i}}{NDLP_{C_i}}$$

The detection process is relatively more robust than the recognition process. (Roberts and Casanova 2012) concluded that the DR is high than the Recognition Ratio in most cases. They also surveyed DR and RR in the United Kingdom. The survey showed that the overall DR is 95% and RR is 90%. The theoretical sample rates are generally provided by the manufacturer of ALPR cameras. These theoretical sample rates can be directly used:

$$\widehat{DR} = DR_0$$

$$\widehat{\mathrm{RR}} = RR_0$$

However, since the sampling rates of ALPRs are affected by many factors, calibrating sampling helps to improve the performance of traffic condition reconstruction. The detection ratio can easy measured with extra data from manual traffic counting or sensors that can provide accurate traffic counts:

$$\widehat{DR_{C_i,t}} = \frac{NDLP_{C_i,t}}{Traffic\ Count_{C_i,t}}$$

The recognition ratio is more difficult to estimate because manually recording license plate numbers are too labor-intensive and there are no technologies that outperform ALPR on recognizing license plate numbers. This thesis proposes an applicable approach to estimate the lower bound and upper bound of recognition ratios ALPRs within high-coverage ALPR systems. This approach assumes:

- The ratios vary over time but are relatively stable over a short time period.
- At a given short time period, the recognition ratios of ALPRs monitoring the same intersections are the same, and RR of ALPRs monitoring adjacent intersections are similar.

Because most of the factors that affect the sampling rates are stable in short periods, these two assumptions should hold in most cases. First, an upper bound of the recognition ratio of each ALPR is achieved from records collected at each ALPR. Incorrectly recognized license plate numbers consisted of ones unrecognized and ones recognized as incorrect numbers. Unrecognized ones are generally expressed as missing values in records. The ratio between the Number of Recognized License Plates (NRLP) and the number of detected ones is an upper bound of the recognition ratio.

$$An\ Upper\ Bound\ of RR_{C_i,t} = \frac{NRLP_{C_i,t}}{NDLP_{C_i,t}}$$

To get a lower bound of R_{rd} for an ALPR C_i, one set of ALPRs is searched. The set $S_{Downstream}$ contains all the downstream ALPRs of the C_i. Then, according to recognized license plate numbers, the record captured by C_i are matched with records captured by ALPRs within $S_{Downstream}$.

When a license plate number is incorrectly recognized, there are tons of ways that it can be incorrect. Most of the time, the incorrectly recognized numbers do not belong to any vehicles. Besides, even if they belong to other vehicles, the possibility that these vehicles also appear in the records of downstream ALPR in the same period is very low. So the Number of Matched License Plates (NMLP) can be considered as the number of license plates that are correctly recognized by both ALPRs in S_1 and ALPRs in S_2. So, the relationship between RR, DR, NMLP, and Traffic Volume is:

$$Traffic\ Volume_{C_i,t} * DR_{C_i,t} * RR_{C_i,t} \geq NMLP_{C_i,t}$$

which provides a lower bound for RR:

$$A\ lower\ bound\ of\ RR_{C_i,t} = \frac{NMLP_{C_i,t}}{Traffic\ Volume_{C_i,t} * DR_{C_i,t}} = \frac{NMLP_{C_i,t}}{NDLP_{C_i,t}}$$

The estimator of RD is designed as the mean of the upper bound and lower bound of RD:

$$\widehat{RR_{C_i}} = \frac{1}{2}\left(\frac{NRLP_{C_i}}{NDLP_{C_i}} + \right)$$

This proposed approach provides a way to dynamically estimate the sampling rate of ALPRs. Based on the estimated sample rate, we can immediately detect malfunction of ALPRs when it happens. Besides, we can have a better estimation of traffic volumes.

Traffic volume on link L during time interval t:

$$Traffic\ Volume_{L,t} = \sum_{C_j \in L} \frac{NDLP_{C_{j,t}}}{DR_{C_{j,t}}}$$

Traffic volume of route R during time interval t:

$$Traffic\ Volume_{R,t} = \frac{N_R}{\prod_{L_j \in L} DR_{L_j,t} * RR_{L_j,t}}$$

N_R is the number of vehicles that have completely recorded trajectories over route R. This estimator only works when the number of links within the route is small. When the number of links is large, the number of samples is very small or even unachievable. For simplicity, RR and DR of links are used, which is calculated as the average of RR and DR over the links.

4.4 Estimation of operational performance

4.4.1 Delay

Intersection delay estimation is considered one of the most important measures of traffic conditions at intersections. It is widely used for the evaluation of intersection performance and many traffic control methodologies target minimizing the overall delay of traffic.

- Delay of V_k

$$\text{Delay}_{V_k} = \begin{cases} TravelTime_{V_k} - AvgLinkTravelTime_L if\ V_k \notin \boldsymbol{S_N} \\ 0\ if\ V_k \in S_N \end{cases}$$

- The average delay at I_i

$$AvgDelay_{I_i} = \frac{\sum_k \text{Delay}_{I_i,V_k}}{\sum_k 1_{I_i,V_k}}$$

- The variance of delay at I_i

$$VarDelay_{I_i} = \frac{\sum_k \left(Delay_{I_i,V_k} - AvgDelay_{I_i}\right)^2}{\sum_k 1_{I_i,V_k} - 1}$$

4.2.2 Number of stops

Number of stops at I_i

$$Stops_{I_i} = \sum_{V_k \in V_{C_x}} 1_{S_N}(k)$$

$$1_{S_N}(k) = \begin{cases} 1 \; if \;\; V_k \notin \boldsymbol{S_N} \\ 0 \; if \;\; V_k \in S_N \end{cases}$$

$1_{S_N}(k)$ is an indicator function indicating whether the vehicle V_k belong to the cluster of vehicles that are not affected by the signal control.

4.2.3 Headway

Average headway of vehicles cross

$$AvgHeadway = \frac{max_k\left(t_{\mathrm{V_k,C_x}}\right) - min_k\left(t_{\mathrm{V_k,C_x}}\right)}{\left(\sum_{V \in \boldsymbol{V}_{C_x}} 1\right) \; - \; 1}$$

Queue discharge headway of vehicles

$$QueueDischargeHeadway = \frac{max_k\left(t_{\mathrm{V_k,C_x}}\right) - min_k\left(t_{\mathrm{V_k,C_x}}\right)}{\left(\sum_{V \in \boldsymbol{V}_{C_x}} 1\right) \; - \; 1}, V_k \in Stopped\ Vehicles$$

4.2.4 Speed

Average travel speed of vehicle k on $L_{i,j}$

$$AvgVehSpeed_{L_{i,j},V_k} = \frac{TravelTime_{L_{i,j},V_k}}{l_{L_{i,j}}}$$

Time mean speed of vehicles on $L_{i,j}$

$$TimeMeanSpeed_{L_{ij}} = \frac{\sum_{V_k \in \left(\boldsymbol{V}_{C_x} \cap \boldsymbol{V}_{C_y}\right\}} \; AvgVehSpeed_{L_{i,j},V_k}}{\sum_{V_k \in \left(\boldsymbol{V}_{C_x} \cap \boldsymbol{V}_{C_y}\right\}} 1}$$

Space mean speed of vehicles on $L_{i,j}$

$$SpaceMeanSpeed_{Li,j} = \frac{l_{L_{i,j}} \sum_{V_k \in \left(v_{C_x} \cap v_{C_y}\right\}} 1}{\sum_{V_k \in \left(v_{C_x} \cap v_{C_y}\right\}} TravelTime_{L_{i,j},V_k}}$$

4.2.5 Saturation flow rate

$$s = \frac{3600}{QueueDischargeHeadway}$$

Where

s = saturation flow rate

4.2.6 Degree of saturation

$$X = \frac{v}{c} = \frac{CV}{sg}$$

Where

C = cycle length of the signal plan

g = effective green time

X = degree of saturation

V = traffic volume of the lane group

v = traffic demand of the lane group

c = capacity of the lane group

4.2.7 LOS

LOS of intersection i

$$LOS_{I_i} = \begin{cases} A & if\ AvgDelay_{I_i} \leq 10\ sec \\ B & if\ 10\ sec \leq AvgDelay_{I_i} \leq 20\ sec \\ C & if\ 20\ sec \leq AvgDelay_{I_i} \leq 35\ sec \\ D & if\ 35\ sec \leq AvgDelay_{I_i} \leq 55\ sec \\ E & if\ 55\ sec \leq AvgDelay_{I_i} \leq 80\ sec \\ F & if\ AvgDelay_{I_i} \geq 80\ sec \end{cases}$$

The LOS estimator uses the LOS criteria for signalized intersection in Highway Capacity Manual (HCM) 2021.

All the traffic flow characteristics estimation discussed above are very sensitive to changes in traffic conditions and needed to be estimated in real-time or near real-time for traffic monitoring and diagnosis purpose. Besides, they are also the foundations of more complex analysis. The system keeps the results of these estimations to build the profiles of infrastructures in the traffic network. These profiles will be used in two different ways. First, they will be visualized for traffic monitoring and diagnosis, which is the main topic in Chapter 5. Second, they are the foundation of more complicated analyses, such as origin-destination estimation, bottleneck identification, and so on.

4.5 Reconstruction of traffic demand

Different from the near real-time estimators discussed above, the origin-destination matrix is designed to be estimated on daily basis. The inputs, outputs, and high-level algorithm of origin-destination estimation are described in *Table 10*. Overall, this algorithm estimates the origin-destination matrix using the trajectories of vehicles. Due to coverage problems and detection errors, the trajectories of vehicles can be incomplete. It is necessary to restructure the complete trajectories of vehicles before calculating the origin-destination matrix. To improve the scalability, the algorithm also implements a divide-and-conquer approach to estimate the origin-destination matrix. Instead of directly construct the entire trajectory of each vehicle, the algorithm starts with

reconstructing partial trajectories of each vehicle over sub-networks and then reconstructs the entire trajectories with partial trajectories. The data processing jobs of different sub-networks are allocated to different processors or servers for better performance. The algorithm consists of five steps.

Table 10 Algorithm of origin-designation estimation

Algorithm of origin-destination estimation
INPUT - A dictionary that describing the ALPR coverage of each sub-network. - CSV files that contain daily ALPR records. Each CSV corresponds to one ALPR camera. - Adjacency lists of traffic network OUTPUT Origin-destination matrix ALGORITHM - Step 1: Reconstruct incomplete partial trajectories of vehicles over sub-networks - Step 2: Reconstruct complete partial trajectories of vehicles over sub-networks - Step 3: Reconstruct complete trajectories of vehicles over the network - Step 4: Calculate the origin-destination matrix based on vehicle trajectories

Algorithm 2: Origin-destination estimation

Step 1: Reconstruct incomplete partial trajectories of vehicles over sub networks

ALGORITHM

For each sub − network N_a

Do

- $VehSubTraj \leftarrow$ *Initialize empty hashmap*
- *For each csv file* F_b *that belong to ALPR camera with in* N_a
  - *Do*
    - *For each record* R_c *stored in* F_b
      - *Do*
        - *If the license plate number of* R_c *is not in* $VehSubTraj$:
          - *Do*
            - $SubTraj \leftarrow$ *Initialize empty double linked list*
            - *Store* $LPN(R_c) : Subtraj$ *into* $VehSubTraj$
        - $Node \leftarrow$ *Initialize node with* $TS(R_c)$ *and* $ALPR(R_c)$
        - *Insert* $Node$ *into* $VehSubTraj[R_c]$
- *for each* $LPN : SubTraj$ *pair in* $VehSubTraj$
  - *Sort* $VehSubTraj$ *in order of timestamp*

Figure 16 Origin-destination estimation: Step 1

Figure 16 illustrates the first step of origin-destination estimation in detail. The main purpose of the first step of origin-destination estimation is to get incomplete partial trajectories of vehicles over sub-networks. The license plate numbers of vehicles and their corresponding incomplete partial trajectories are stored within a hashmap structure VehSubTraj for quick access with license plate numbers. The algorithm traversal each record (R_c) for each sub-network (N_a). If a new license plate number is encountered, an empty double-linked list SubTraj will be created. The license plate number and SubTraj will be stored in VehSubTraj as key-value pair. Then a node containing the timestamp and location information of current records will be inserted to the tail of the double linked list. If the license plate number of the current record is already in VehSubTraj, a node containing the timestamp and location information of current records will be inserted to the tail of the corresponding double-linked list. After the traversal, all the records of each vehicle in a

sub-network are stored in a doubled-linked list. Then each doubled-linked list SubTraj is sorted in order of timestamp to get incomplete partial trajectories.

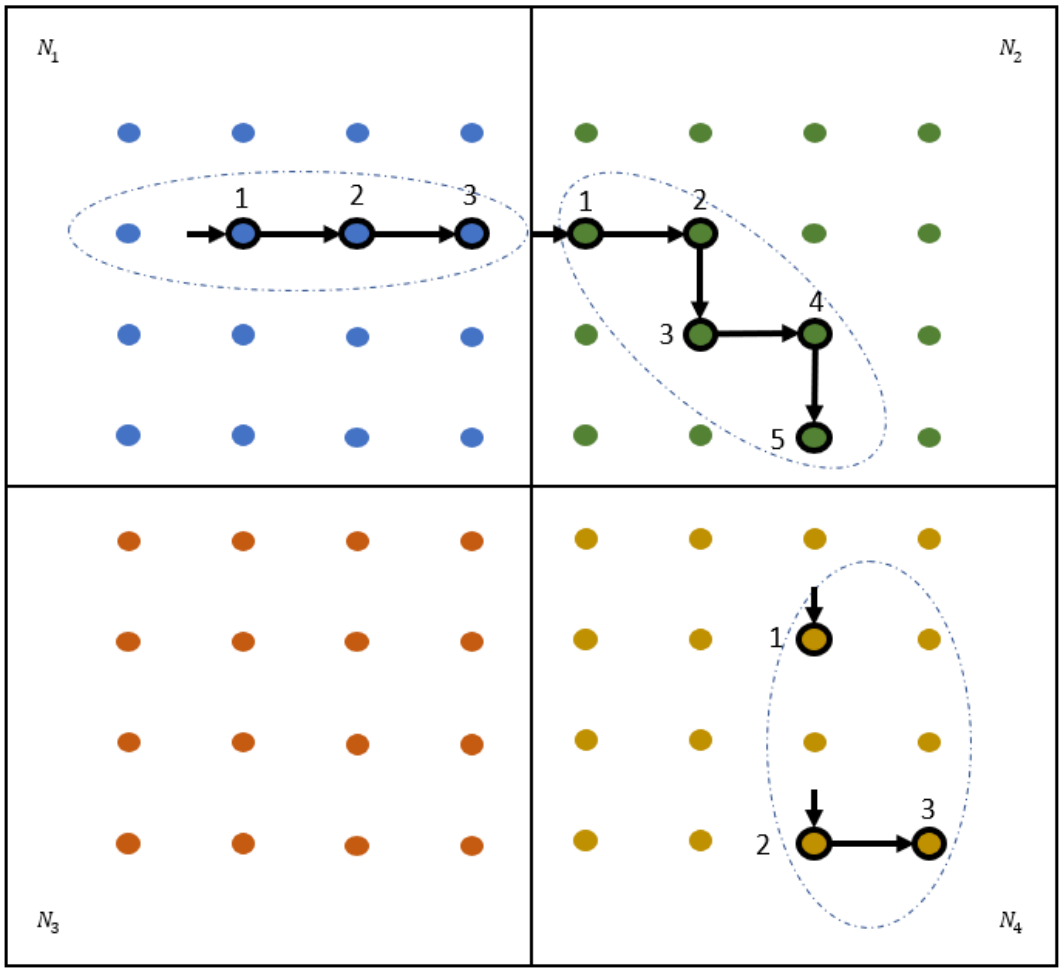


Figure 17 Incomplete partial trajectories of vehicles

Figure 17 illustrates an example of incomplete partial trajectories of one vehicle over four sub-networks N_1, N_2, N_3 and N_4. The dots with black outlines are the intersections at which the vehicle has been recorded by ALPR cameras. The arrows indicate at which approaches of the intersections the vehicle has been recorded. The numbers near these dots are the time order of those records. In this example, the partial trajectories over N_1 and N_2 are complete while the partial trajectory over sub-network N_4 is incomplete.

Algorithm 2: Origin-destination estimation

Step 2: Reconstruct complete partial trajectories of vehicles over sub-networks

ALGORITHM

```
For each sub − network N_a
    Do
        For each LPN, SubTraj in VehSubTraj
            Do
                For each Node in SubTraj
                    Do
                        NextNode ← Node.next
                        if UI(NextNode.ALPR) == DI(Node.ALPR)
                            continue
                        else
                            if UI(NextNode.ALPR) is in ADJ(DI(Node.ALPR))
                                I_i ← DI(Node.ALPR)
                                I_j ← UI(NextNode.ALPR)
                                I_k ← DI(NextNode.ALPR)
                                t_0 ← Node.timestamp
                                t_1 = AvgTravelTIme_{L_{i,j}}
                                Do
                                t_2 = AvgTravelTime_{L_{j,k}}
                                t = t_0 + t_1 * (t_1+t_2) / (NextNode.timestamp − Node.timestamp)
                                Node ← initializenodeNode.UpInt ← T_i
                                Node.DownInt ← L_j
                                Node.timestamp ← t
                                Insert Node into SubTraj
                            else
                                run Algorihm 2.1
```

Figure 18 Origin-destination estimation: Step 2

The second step in the algorithm of origin-destination estimation is illustrated in Figure 18. The main purpose of the second step is to fix the incompleteness issues of partial trajectories. The

algorithm traverses the key-value pair of VehSubTraj to fix the incomplete partial trajectory of each vehicle. For the SubTraj of each vehicle, the algorithm traversal each node and check if there is incompleteness in the trajectory. If the upstream intersection of the ALPR of the current node is the same as the upstream intersection of the ALPR of the next node, there is no missing trajectory between the current node and the next node, and no extra operations are needed. Otherwise, the trajectory between the current node and the next node needs to be reconstructed. *Figure* 19 illustrates how this process is realized with the double-linked list VehTraj.

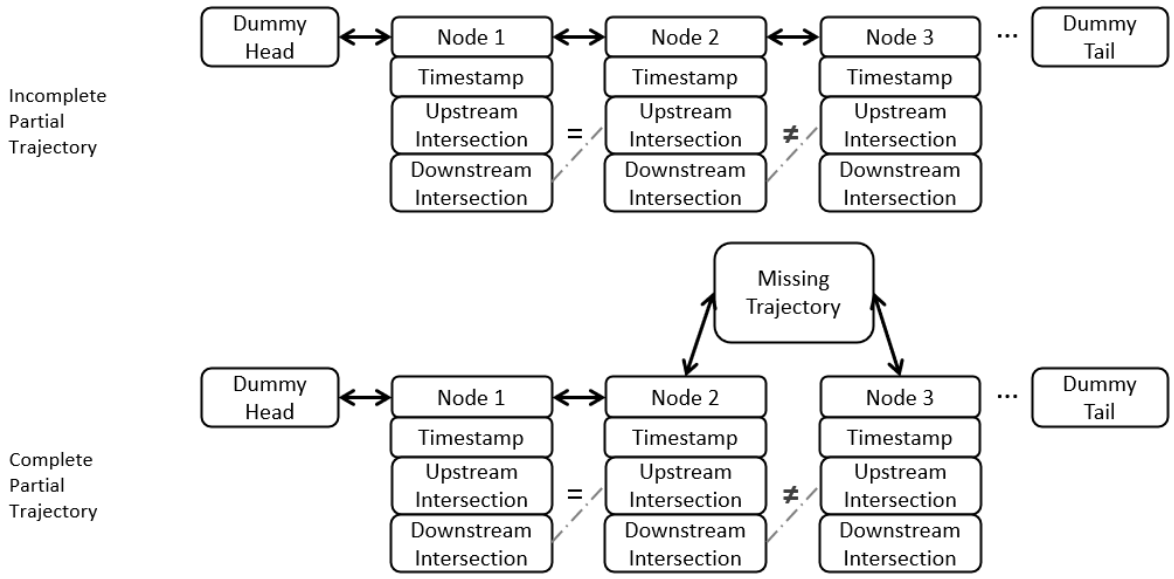


Figure 19 Reconstruct complete partial trajectory

Two methodologies are implemented in the system to reconstruct the missing trajectories. The first methodology is based on the topology of the traffic network and is intended to fix short missing trajectories in a fast manner. Since an ALPR camera generally monitors one approach of one intersection, the record of ALPR does not only specify the downstream intersection but also the upstream intersection of the vehicle trajectory. If the upstream intersection is exactly one of the intersections adjacent to the downstream intersection of the previous node, the system will fix the

trajectory by inserting the missing node. The timestamp of the node will be calculated based on the travel time estimation results. This methodology only works when there is only one node missing in the trajectory.

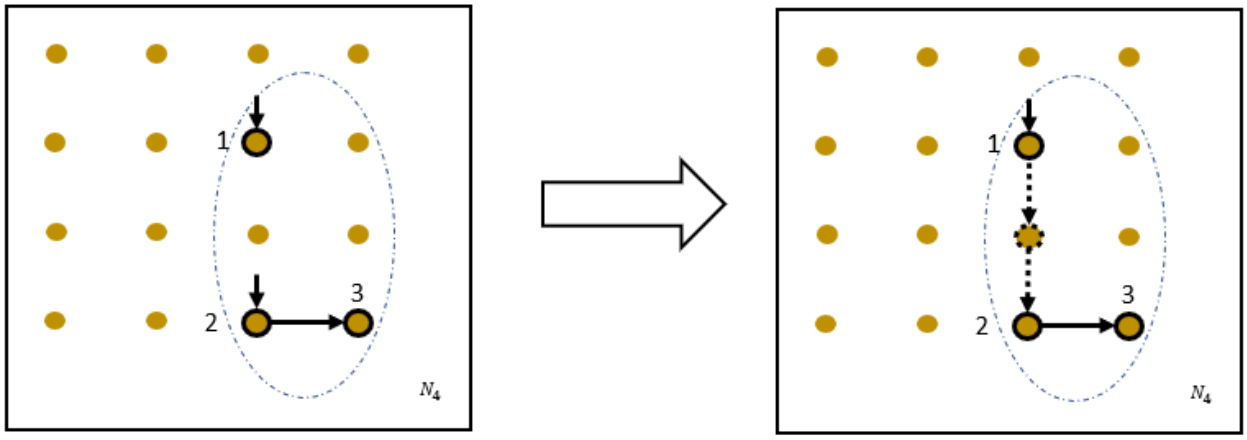


Figure 20 Reconstruct missing trajectory with topology-based methodology

Figure 20 continues the example in Figure 17. The partial trajectory of the vehicle in N_4 is incomplete before reconstruction, but it can be easily fixed with the topology-based methodology. The ALPR of the second node indicates the upstream intersection, which is highlighted in red in the figure. It is also adjacent to the downstream intersection for the first node. So, it meets the requirement of the topology-based methodology, and the fixed trajectory is shown on the right-hand side of Figure 20. The reconstructed part is specified by dashed arrows and cycles.

Algorithm 2.1 Trajectory reconstruction based on maximum likelihood estimation

INPUT

Pair of Nodes

ALGORITHM

$Node,\ NextNode \leftarrow Pair\ of\ Nodes$

$MaxLength$

$I_i \leftarrow Node.DownInt$

$t_i \leftarrow Node.timestamp$

$I_j \leftarrow NextNode.UpInt$

$t_{j+1} \leftarrow NextNode.timestamp$

$t_T \leftarrow t_{j+1} - t_i$

$MaxLikehood \leftarrow -inf$

$MaxLikehoodTraj \leftarrow None$

$T_{I_i,I_j} \leftarrow BFS\ stop\ when\ the\ length\ of\ tracejtory\ is\ MaxLength$

$For\ each\ T\ in\ T_{I_i,I_j}$

$$
Do\begin{cases}
\hat{\mu_{t|T}} \leftarrow AvgTravelTime_{I_j,I_{j+1}} \\
\hat{\sigma^2_{t|T}} \leftarrow VarTravelTime_{I_j,I_{j+1}} \\
L_2 \leftarrow 1 \\
Do\begin{cases}
for\ each\ I_k\ in\ T \\
\quad Do\begin{cases}
\hat{\mu_{t|T}} += AvgTravelTime_{I_k,I_{j+1}} \\
\hat{\sigma^2_{t|T}} += VarTravelTime_{I_j,I_{j+1}} \\
if\ L_{k.,k+1}\ is\ covered\ by\ the\ ALPR\ sytem: \\
\quad L_2 *= \hat{\alpha}
\end{cases}
\end{cases} \\
\mu = ln(\frac{\mu^2_{t|T}}{\sqrt{\mu^2_{t|T}+\sigma^2_{t|T}}}) \\
\sigma^2 = ln\left(1+\frac{\sigma^2_{t|T}}{\mu^2_{t|T}}\right) \\
L \leftarrow L_1 * L_2 if L > MaxLikehood \\
\quad Do\begin{cases}
MaxLikehood \leftarrow L \\
MaxLikehoodTraj \leftarrow T
\end{cases}
\end{cases}
$$

$Create\ Nodes\ for\ trajectory\ T\ and\ insert\ them\ between\ Node\ and\ NextNode$

Figure 21 Trajectory reconstruction based on MLE

The second methodology starts running if the first methodology fails to find the missing trajectory. The Pseudocode of Algorithm 1.1 is provided in Figure 21. It assumes the vehicle does not visit the same intersection twice in short periods and uses BFS to search alternative trajectories and find the best matching trajectory. In a high-coverage ALPR system, the best match will appear in the early stage of breadth-first traversal BFS. The BFS stops once the best match is found. During the BFS, the best-matching trajectory is the one that maximizes the likelihood.

Given there is a trajectory between I_i and I_j and the trajectory is $T = (I_{i+1}, \dots, I_{j-1})$, the likelihood function that this trajectory is not recorded by the ALPR system:

$$\mathrm{L}(T|X_1) = \prod_{k=i+1}^{k=j-1} (1-\alpha)^{1(k,k+1)}$$

$$1(k,k+1) = \begin{cases} 1 & If\ there\ is\ no\ ALPR\ camera\ on\ L_{k,k+1} \\ 0 & Otherwise \end{cases}$$

X_1: The trajectory leaves no records in the ALPR system

If the traffic from I_k to I_{k+1} is not monitored by any ALPR cameras, the record is missing simply because this approach of the intersection is not covered by the ALPR system. $1(k, k+1)$ equal to 0 and the probability that the vehicle is not recorded at I_{k+1} is 1. Otherwise, the missing records are caused by detection errors. In the formula, $1-\alpha$ is used to measure the probability of the detection errors that happens when the vehicle passes the ALPR camera. This measurement is reasonable if the following assumption stands:

1. The probability that a license plate is detected and recognized by a license plate reader is not related to vehicles, i.e. every vehicle is the same likely to be recorded by an automated

license plate reader when it passes through the coverage area of the automated license plate reader. This assumption may not be strictly satisfied in the real world. Some license plates may be too dirty to be detected and recognized by license plate readers. However, for most of the vehicles, this assumption is correct, and it tremendously simplifies the problem.

2. The probability that a license plate reader can detect and recognize vehicles is stable over time. This probability may be different between different license plate readers located in different areas due to the geometry of roads or intersections. However, the probability of each license plate reader is assumed to be achievable and deterministic. This assumption may be offended due to changes of day and night, different traffic conditions, or various weathers.

t_{C_i,C_j}

Meanwhile, the likelihood function for event X_2 (the total travel time of trajectory T is t_T):

$$L(T|X_2) = f(t|T)$$

$f(t|T)$ is the conditional density distribution of travel time t given the trajectory is T. Two approaches are designed to estimate the conditional density distribution of travel time. The first one assumes that the travel time follows a lognormal distribution. This is a strong assumption on the total travel time of trajectories. It only holds when the traffic is very light and the control delay can be neglected.

$$t|T \sim Lognormal(\mu, \sigma^2)$$

The expectation of t|T:

$$\mu_{t|T} = E(t|T) = E\left(\sum_{k=i}^{j-1} t_{k,k+1}|T\right) = \sum_{k=i}^{j-1} E\left(t_{k,k+1}\middle|T\right)$$

The estimator of expectation of t|T:

$$\widehat{\mu_{t|T}} = \sum_{k=i}^{j-1} \left(AvgTravelTime_{I_k, I_{k+1}} \right)$$

The variance of t|T: t_T

$$\sigma_{t|T}^2 = Var(\sum_{k=i}^{j-1} t_{k,k+1}|T)$$

Assume no correlation between average travel time on different links:

$$Var(\sum_{k=i}^{j-1} t_{k,k+1}|T) = \sum_{k=i}^{j-1} Var\left(t_{k,k+1}\middle|T\right)$$

$$\widehat{\sigma_{t|T}^2} = \sum_{k=i}^{j-1} \left(VarTravelTime_{I_k, I_{k+1}} \right)$$

Because $t|T \sim Lognormal(\mu, \sigma^2)$, the mean and variance of t|T follows:

$$\begin{cases} E(t|T) = e^{\left(\mu + \frac{\sigma^2}{2}\right)} \\ Var(t|T) = \left[e^{(\sigma)^2} - 1\right] e^{(2\mu + \sigma^2)} \end{cases}$$

Convert equations and get:

$$\begin{cases} \mu = ln(\dfrac{\mu_{t|T}^2}{\sqrt{\mu_{t|T}^2 + \sigma_{t|T}^2}}) \\ \sigma^2 = ln\left(1 + \dfrac{\sigma_{t|T}^2}{\mu_{t|T}^2}\right) \end{cases}$$

The estimated distribution of t|T using the estimations of $\mu_{t|T}$ and $\sigma_{t|T}^2$:

$$t|T \sim Lognormal(ln(\frac{\widehat{\mu_{t|T}}^2}{\sqrt{\widehat{\mu_{t|T}}^2 + \widehat{\sigma_{t|T}^2}}}), ln\left(1 + \frac{\widehat{\sigma_{t|T}^2}}{\widehat{\mu_{t|T}}^2}\right))$$

$$L(T|X_2) = f(t = t_T|T) = \frac{1}{t_T \sigma\sqrt{2\pi}} e^{\left(-\frac{(ln\, t_T - \mu)^2}{2\sigma^2}\right)}$$

The overall likelihood function of trajectory T:

$$L(T|X_1 \cap X_2) = L(T|X_1)\, L(T|X_2) = \frac{1}{t_T \sigma\sqrt{2\pi}} e^{\left(-\frac{(ln\, t_T - \mu)^2}{2\sigma^2}\right)} \prod\prod_{k=i+1}^{k=j-1} (1-\alpha)^{1(k,k+1)}$$

The Trajectory that maximizes the likelihood function will be chosen to rebuild the incomplete trajectory.

$$T* = argmin_T\, L(T|X_1 \cap X_2)$$

Subject to $T \in \mathbf{T_{I_i,I_j}}$

where $\mathbf{T_{I_i,I_j}}$ is the trajectory patching set for missing trajectory from I_i to I_j

Another approach estimates the probability density function with kernel density function. Figure 22 illustrates the third step in the origin-destination estimation. In this step, partial trajectories of vehicles are concatenated to rebuild the entire trajectories. Similar to step 1 and step 2, the partial trajectories of vehicles are grouped, sorted, and reconstructed. The methodologies used in the first and second step also apply in the third step with some modifications. In the third step, the nodes in the double-linked list are no longer about a single node but a partial trajectory. After the third step, the complete daily trajectories of all vehicles are rebuilt.

Algorithm 2: Origin-destination estimation

Step 3: Reconstruct complete trajectories of vehicles over the network

ALGORITHM

$VehTraj \leftarrow$ *initizalize empty hashmap*

for each subnetwork N_i *and its* $VehSubTraj$

- *For each Vehicle and its SubTraj in VehSubTraj*
  - *If the Vehicle is not in VehSubTraj :*
    - *Traj* $\leftarrow$ *Initialize empty double linked list*
    - *Store Vehicle : Traj into VehSubTraj*
  - *Insert Sub into Traj*
- *for each Vehicle : Traj pair in VehTraj*
  - *Sort VehSubTraj in order of timestamp*
  - *For each SubTraj in Traj*
    - $SubTrajLastNode \leftarrow SubTraj.tail.previous$
    - $NextSubTraj \leftarrow SubTraj.next$
    - $NextSubTrajFirstNode \leftarrow NextSubTraj.head.next$
    - *if* $UI(NextSubTrajFirstNode.ALPR) == DI(SubTrajLastNode.ALPR)$
      - *continue*
    - *else*
      - *if* $UI(NextSubTrajFirstNode.ALPR)$ *is in* $ADJ(DI(SubTrajLastNode.ALPR))$
        - $I_i \leftarrow DI(SubTrajLastNode.ALPR)$
        - $I_j \leftarrow UI(NextSubTrajFirstNode.ALPR)$
        - $I_k \leftarrow DI(NextSubTrajFirstNode.ALPR)$
        - $t_0 \leftarrow SubTrajLastNode.timestamp$
        - $t_1 = AvgTravelTIme_{L_{i,j}}$
        - $t_2 = AvgTravelTime_{L_{j,k}}$
        - $t = t_0 + t_1 * \frac{t_1+t_2}{NextSubTrajFirstNode.timestamp - SubTrajLastNode.timestamp}$
        - $Node \leftarrow initializenodeNode.UpInt \leftarrow T_i$
        - $Node.DownInt \leftarrow L_j$
        - $Node.timestamp \leftarrow t$
        - *Concatenate SubTraj, Node, and NextSubTraj*
      - *else*
        - $MissingTraj \leftarrow$ *run Algorihm* 1.1 $(SubTrajLastNode\,,\ NextSubTrajFirstNode)$
        - *Concatenate SubTraj, MissingTraj, and NextSubTraj*

Figure 22 Origin-destination estimation Step 3

Algorithm 2: Origin-destination estimation

Step 4: Calculate the origin-destination matrix based on vehicle trajectories

ALGORITHM

$ODMatrix \leftarrow$ *initialize empty matrix of shape* (n, n), *where* n *is number of intersections*

for each $Vehicle$ *and its* $Traj$ *in* $VehTraj$

Do

- $Origin \leftarrow Traj.head.next$
- $Destination \leftarrow Origin$
- *While* $Destination.next$ *is not* $Traj.tail$
  - *Do*
    - *if* $Destination.Next.timestamp$ *in* $TravelTimeOutliers[Vehicle]$
      - *Do*
        - $ODMatrix[Origin][Destination] += 1$
        - $Origin \leftarrow Destination.next$
        - $Destination \leftarrow Origin$
    - *else*
      - $Destination \leftarrow Destination.next$
- $ODMatrix[Origin][Destination] += 1$

return $ODMatrix$

Figure 23 Origin-destination estimation Step 4

The fourth step of the origin-destination estimation algorithm split daily trajectories of vehicles trajectories of trips based on the outliers found during travel time estimation. Then the origin-destination matrix is calculated by traversal all trajectories of trips and increment the corresponding element in the matrix. The pseudocode is presented in *Figure 23*.

By splitting the network into sub-networks, most of the operations in the origin-destination estimation algorithm are highly parallelizable. Different methodologies are implemented to reconstruct trajectories. The topologies-based method performs fast fixes and the MLE-based method handles more severe trajectory problems.

4.6 Case study

To better validate and illustrate the feasibility and capability of the proposed system, a real case is tested and presented at the end of Chapter 4, 5, and 6. Section 4.4.1 briefly describes the data the case study, data collection process and performs a descriptive analysis. Section 4.4.3 implements the proposed traffic flow estimation methodologies and algorithms on the real case and presents the results.

4.6.1 Description of case study

The studied data are collected from the ALPR system and the signal control system in Suzhou. Suzhou is a well-developed city in China with a total of more than 10 million resident population Its high population induces high traffic demands. The residents in Suzhou own more than 3 million vehicles. The ALPR system has more than five hundred ALPR cameras monitoring about one hundred and seventy intersections and ramps. The video ALPR system produces more than 4 million records per day.

The data analyzed in this study section covers all records from ALPR cameras around Suzhou Industrial Park, which has the highest coverage of ALPR cameras in Suzhou. The data spans from April 1^{st,} 2017 to April 2^{nd,} 2017. In this area, 47 out of 86 intersections are fully covered by the ALPR cameras. Figure 24 shows a map of Suzhou Industrial Park. Green dots in the figure represents the intersections covered by ALPR cameras, while red dots represent intersections uncovered. Samples of ALPR data and signal control data are presented in Appendix A and Appendix B. All sensitive fields in the samples are masked with SHA1. The ALPR system in Suzhou records not only the license plate number, timestamp, and ALPR ID but also the turning

directions, spot speed, license plate color, and turning direction. Besides, in most of the intersections covered by the ALPR system, each approach is monitored by at least one camera, which infers the upstream intersection of each record.

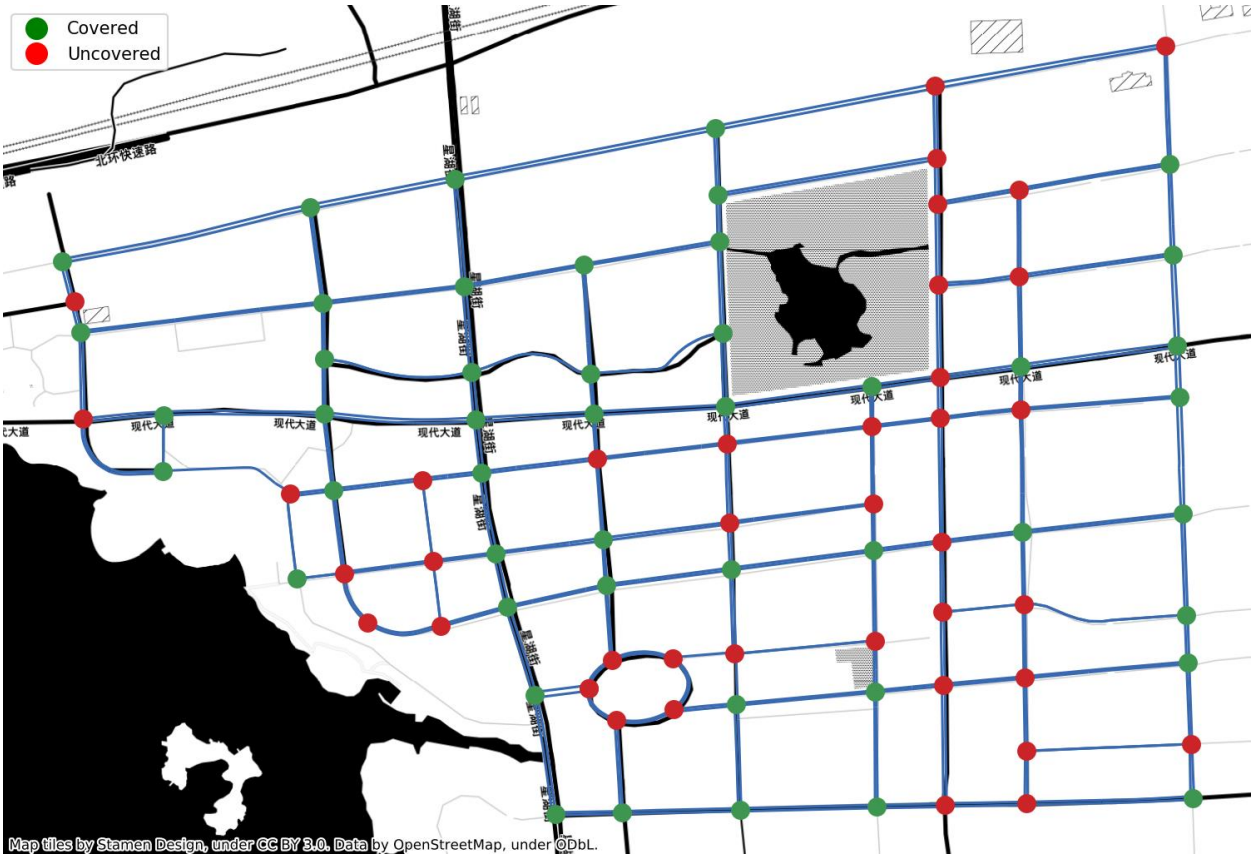


Figure 24 Coverage of ALPR system in the case study

The APRL system produced 642,676 records from 116,449 vehicles on Saturday, April 1st, 2017, and 529,170 records from 96,478 vehicles on Sunday, April 2nd, 2017. Figure 25 illustrates the number of records throughout the case study. The overall number of records on April 1st is high than April 2nd and has a two-peak pattern, which often only appears on weekdays.

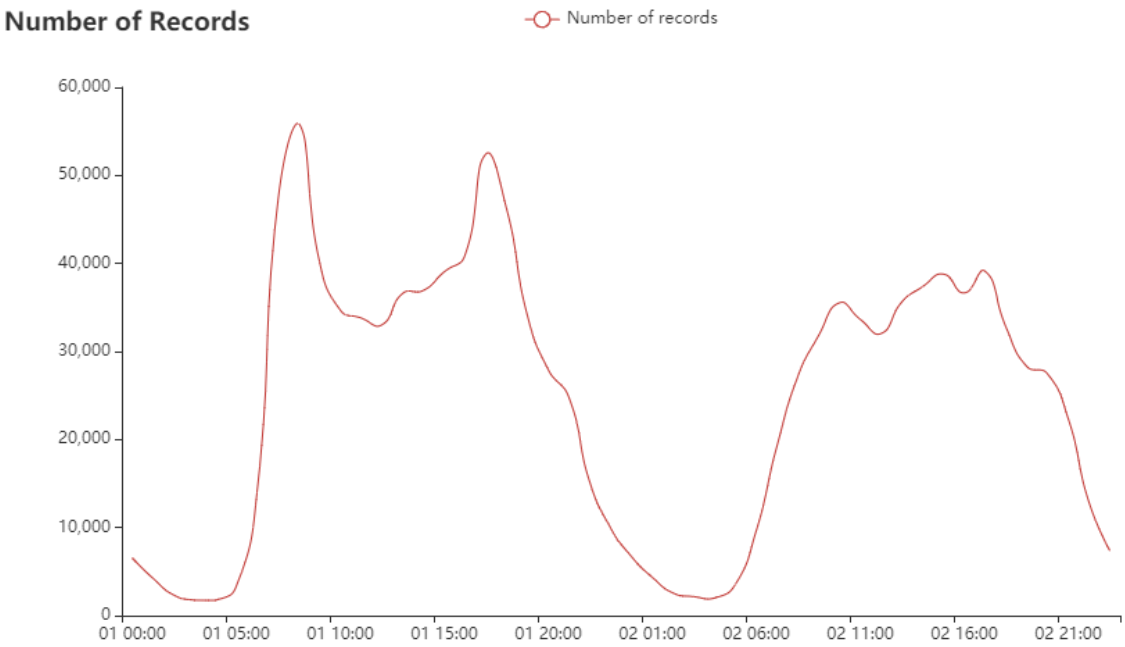


Figure 25 Number of records

4.6.2 Traffic flow characteristics estimation

A field application is performed to verify the feasibility of the proposed system. The results verify the feasibility of the proposed system and help potential users to better understand the framework and details in the system. This section presents the traffic flow characteristics estimation results of the field application. A detailed discussion on the results of intersection performance analysis and traffic demand estimation is presented with visualization tools at the end of Chapter 5.

Figure 26 The satellite photo of the intersection

Xiandai Avenue and Xinghu Street are the two main arterials in this traffic network. The field application takes the intersection of two main arterials as an example to present how the estimation methodologies can be implemented. Figure 26 is the satellite photo of the intersection. The intersection has four legs, and each leg has four lanes. Eight ALPRs are set up to monitor the traffic entering the intersection. Each ALPR covers two lanes of one leg. By implementing the proposed sampling rate estimation methodologies, the recognition ratio of ALPRs at this intersection can be estimated. Figure 27 presents the estimated recognition ratio of the ALPRs on the eastbound of the intersection. The daily average value of the recognition ratio is 88%. The recognition ratio range from 81 % to 92%, which is very close to the values in existing studies. The recognition ratio is

relatively lower during morning peak and evening peak. During peak hours, the increasing traffic volume may affect the ability of ALPR to capture the license plate numbers of all vehicles simultaneously. The higher traffic volume also increases the probability that the license plates of vehicles are covered by other vehicles. The figure also shows a huge drop before the morning peak. Considering the land use of the industrial park, there is a chance that the drop in recognition ratio is caused by the large traffic demand of trucks in the early morning. The license plates of trucks are harder to recognize than private vehicles because they are more likely to be covered by dirt. Besides, trucks may block the vision of ALPR and make it more difficult to identify and recognize the license plate of other vehicles. Figure 28 shows the heavy traffic ratio of traffic flow. The figure is created based on the plate color information in ALPR records. In China, the license plate color of heavy vehicles is yellow, which is different from private vehicles. The heavy traffic ratio has a huge peak in the early morning, which corresponds to the drop in recognization ratio.

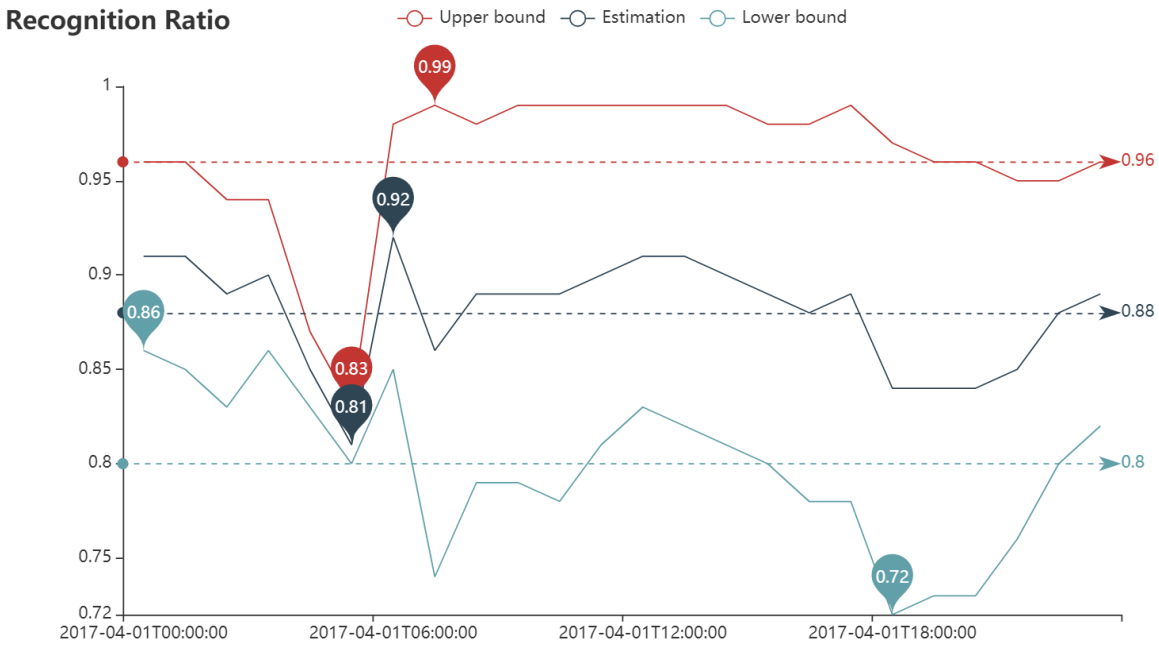


Figure 27 Estimation of recognition ratio

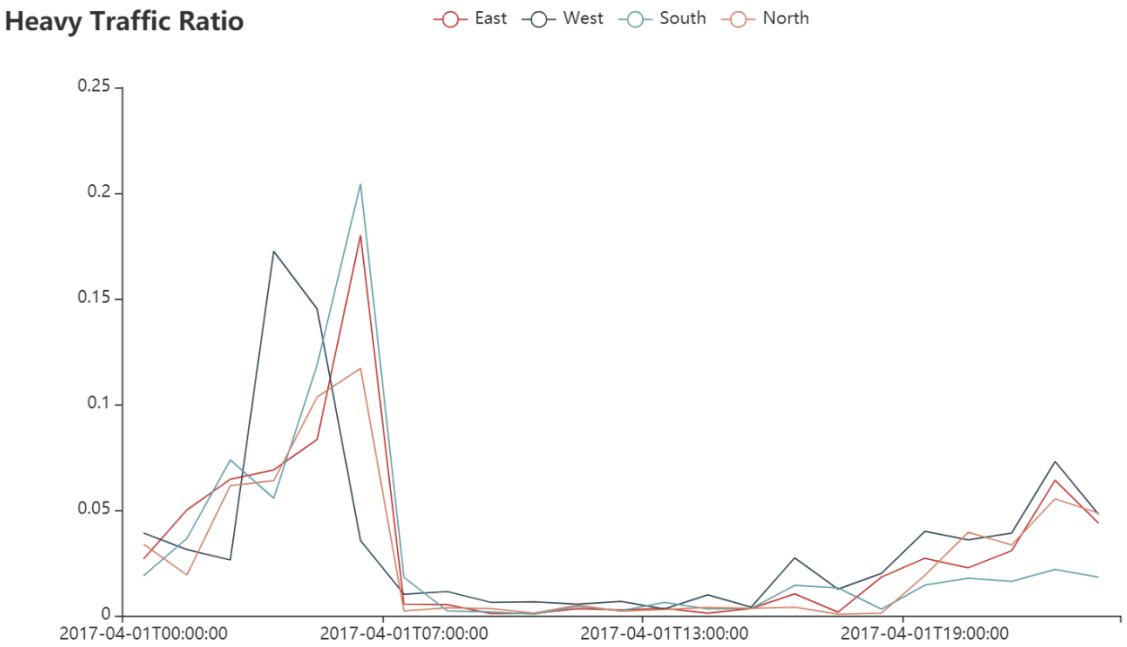


Figure 28 Heavy traffic ratio

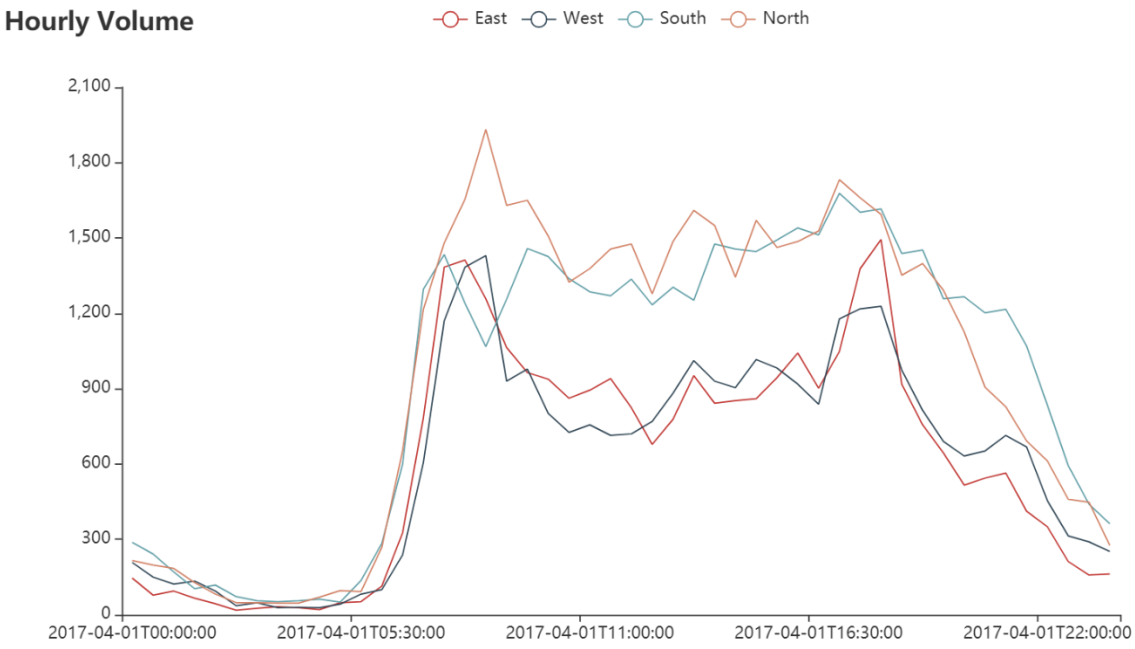


Figure 29 Hourly volume

Figure 29 illustrates how the traffic volume on the four legs of the intersection changes. As one of the busiest intersections in the traffic network, all four approaches experience high traffic demand from 6:00 AM to 10:00 PM. A morning peak appears at 7:00 AM and an evening peak appears at 6:30 PM. The hourly traffic volume of southbound and northbound of the intersection are generally higher than eastbound and westbound.

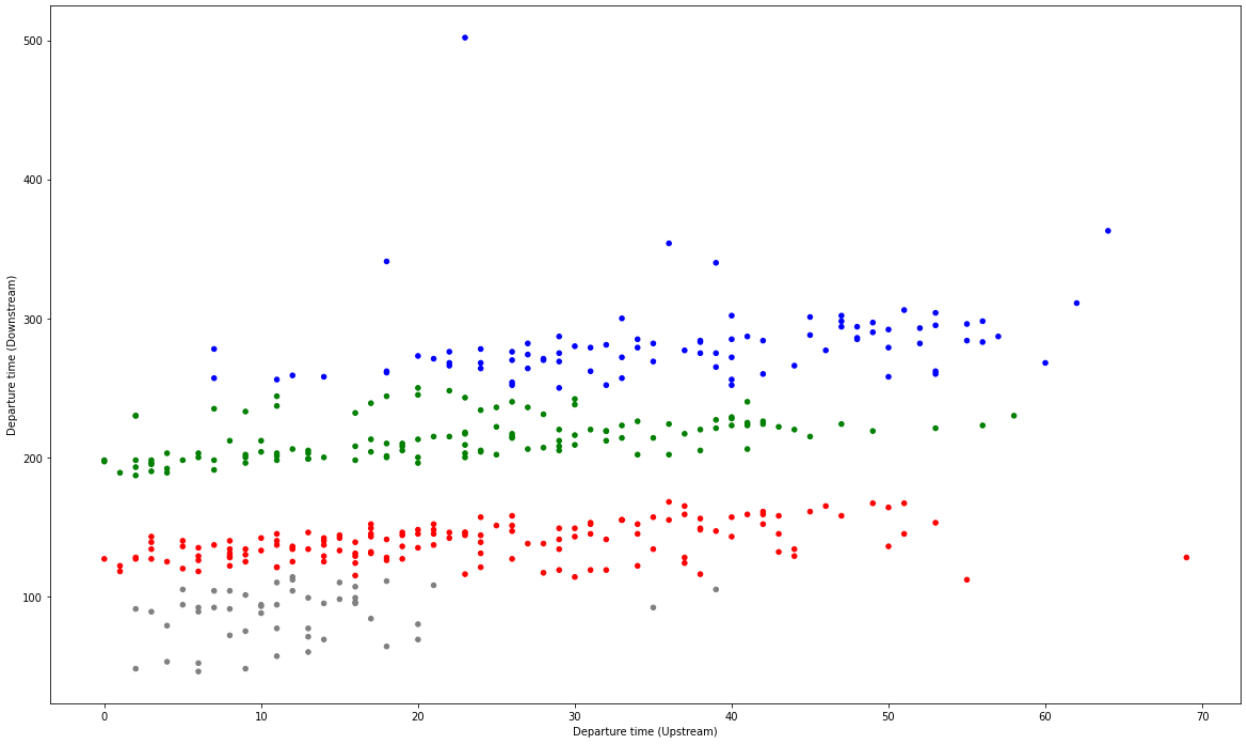


Figure 30 Clustering-based link travel time estimation

Figure 30 shows the clustering results of the proposed estimation algorithm for the eastbound of the intersection over 8:00 AM to 8:15 AM. Six signal cycles of the upstream intersection are within this time period. The offsets are 139 seconds, 13 seconds, 19 seconds, 33 seconds, 69 seconds, and 79 seconds. So the number of clusters is initialized as 4. The clustering algorithm successfully separates the clusters of data samples collected in the queue discharging process from the cluster of vehicles that are not affected by the signal control.

In this section, the results of traffic characteristics estimation on a field application is presented. To some extent, it verifies the feasibility and reliability of the proposed system. A detailed discussion of the results intersection performance analysis and traffic demand estimation is presented with visualization tools in Chapter 5.

Chapter 5: Diagnosis-oriented visual analysis

5.1 Introduction

This chapter illustrates the visual analytics tools embedded in the proposed system. The main purpose of the visual analytics tools is to transform the outputs of traffic condition reconstruction into intuitive insights. To ensure the extensibility of the system, the tools and modules within the system are deeply modularized. The visual analysis tools within the system work individually but also complement each other to reveal the overall spatial-temporal pattern in the data. According to the main purpose of the visual analysis tools, the visual analytics system consists of two separate modules:

- Bottleneck identification: Bottlenecks within the traffic networks are the road segments or intersections that have lower performance. The main purpose of the bottleneck identification module aims to monitor the performance of the entire traffic network, highlight the intersections or road segments that have lower performance, recognize the spatial-temporal pattern of the bottleneck and help to trace back the causing effects of the bottlenecks.
- Traffic Demand analysis: Traffic demand analysis aims to reveal the traffic demand distribution at intersections, along arterials, and over the entire traffic. Traffic demand analysis is the key to perform traffic demand management. The traffic demand is mainly measured by traffic volume on links, along routes, and over traffic networks.

5.2 Bottleneck identification

The bottlenecks are the road segments or intersections that have lower performance than others in the traffic network. Bottleneck identification is a critical component and function for traffic operators to detect problematic spots and segments in the road network for possible improvement. A bottleneck can be caused by various factors, such as incidents, high traffic demand, or bad signal plan design. By identifying the bottlenecks within the traffic network, incidents management, traffic demand management, or traffic signal redesign can be conducted to improve the performance of the intersection.

5.2.1 Traffic status map

The very first step in bottleneck identification is to monitor the traffic condition of road segments and intersections over the entire network. In the proposed system, multiple measurements and a variety of visual tools are designed to monitor traffic conditions and intersection performance. For road segments, the main measurement of performances used in the system is travel speed and travel time. The traffic status map is one of the most widely used visual tools to visualize the travel speed over the network. In a traffic status map, travel speeds are grouped into different levels and each level of travel speed is expressed with a specific color. By expressing links in different colors, the performance of links is highlighted. Users can identify if any road segments are congested or not. The simplicity is the power of the traffic status map. Because traffic status maps are straightforward and easy to implement, it is generally used by drivers and traffic operators. The proposed system provides a visual tool to create a real-time traffic status map using the results of traffic condition reconstruction.

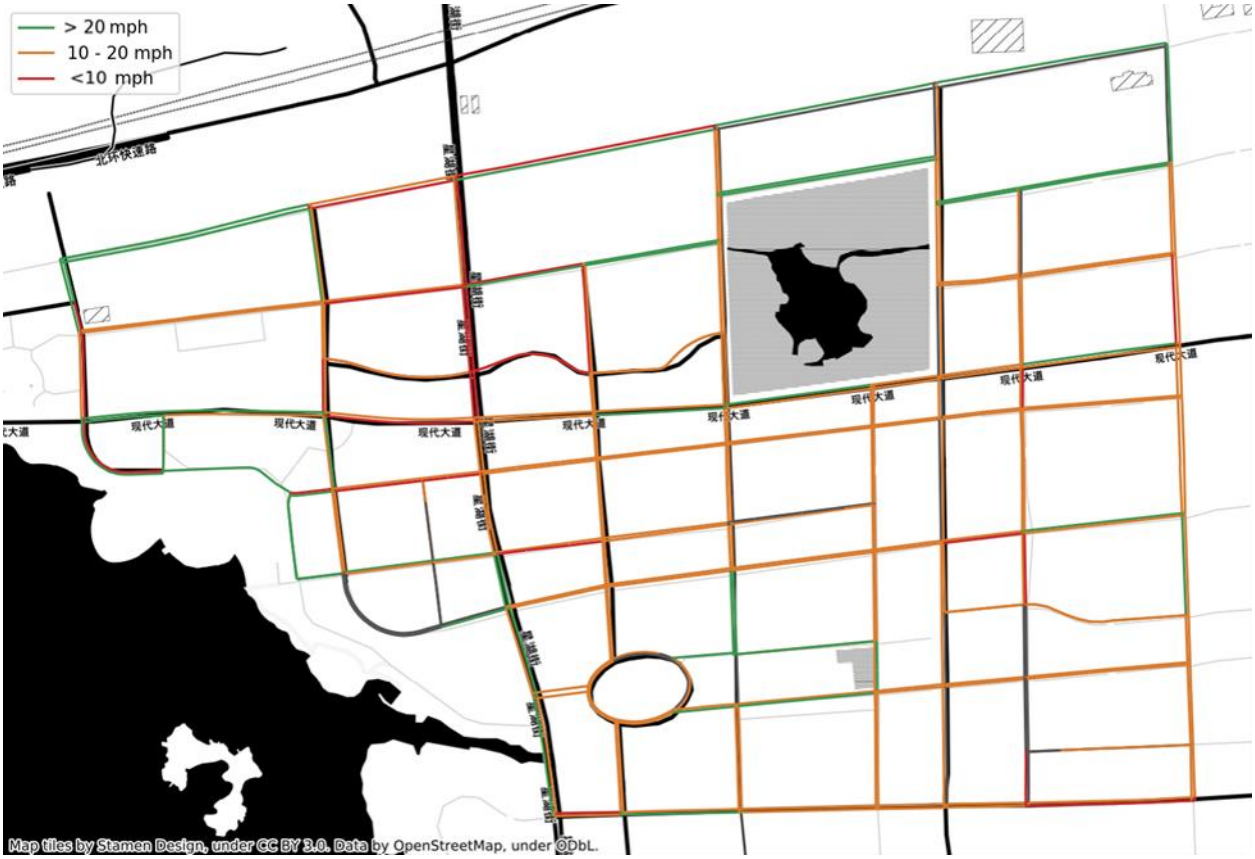


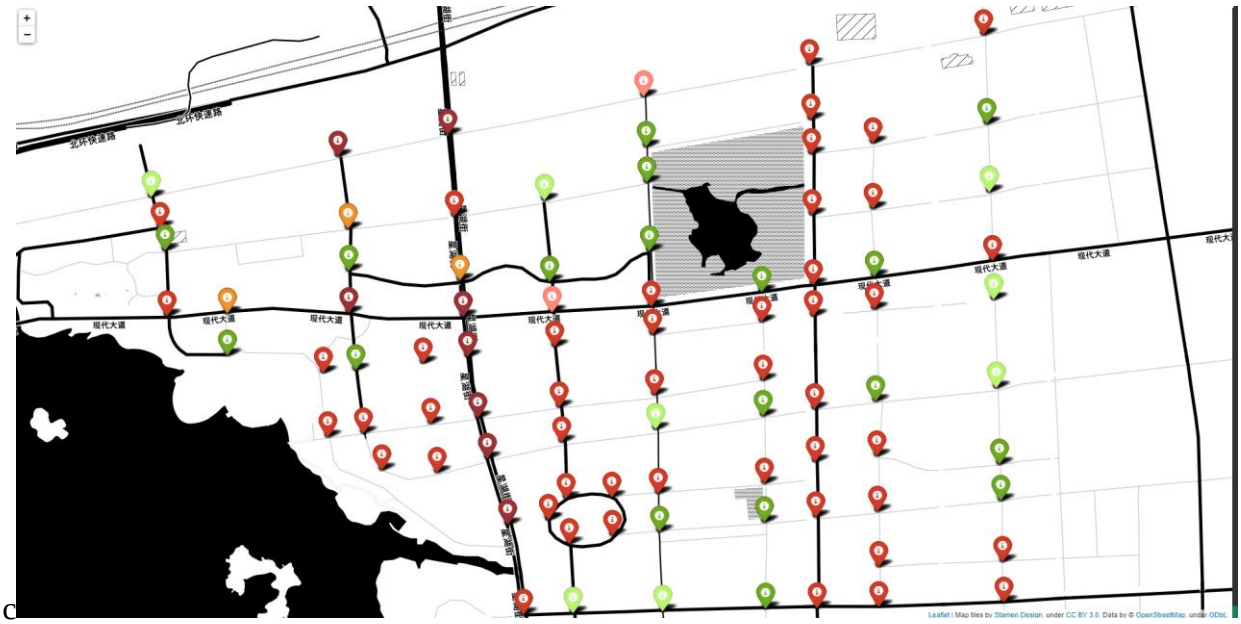
Figure 31 Traffic status map (travel speed)

Figure 31 shows a traffic status map with the data in the field application. The traffic speeds of road segments at the morning peak are presented in three different colors. The space-mean speeds of vehicles on the links are utilized to create this traffic status map. With links with lower performances highlighted in red, the bottlenecks can be easily recognized. The performance of road segments in the middle of the network is generally worse than others. To increase the flexibility of the proposed system, the measurement of traffic status can be customized. Instead of directly using measurements created during the traffic condition reconstruction, customized measurements can be calculated based on existing measurements. For instance, users may want to identify if the observed low performance is recurring or not. Users can customize a measurement using historical travel speed on the same segments with similar traffic demand to indicate how the

current travel speed is different from regular travel speed on this road segment with similar traffic demand. Similarly, a traffic status map for visualizing intersection performance is also implemented. For intersections, the measurement of performances is based on the delay or numbers of stops. Figure *32* shows an example of a traffic status map based on the level of service of intersections.

However, traffic states map also a few drawbacks:

- The traffic status map only illustrates the spatial pattern. By having the geospatial layout, the traffic status map focuses on illustrating the spatial distribution of the measurement. However, temporal information is omitted.
- The traffic status map presents categorized information, which prevents users from performing more comprehensive diagnoses.
- The traffic status map focuses on highlighting the travel speed information of individual traffic segments and ignoring the overall pattern on the traffic network.



C

Figure 32 Traffic status map (LOS)

5.2.2 Travel time contour map

Travel time contour map provides another way to identify potential bottlenecks with travel time data. Travel time contour map presents how far vehicles can reach in different directions with given time limits. Similar to the traffic status map, the travel time contour map presents the spatial pattern. But it focuses more on comparing the travel time and travel speed along with different directions instead of individual road segments. Each contour line shows the farthest place vehicles can reach given travel time. Besides, it provides more quantitative information about travel time and travel speed. Because the time distance between contour lines is pre-defined by users, the distance between contour lines along with a direction d can be regarded as a measure of travel speed the direction d:

$$TravelSpeed_d \propto \frac{1}{Distance_d}$$

Another difference is that in the travel time contour map the travel time includes both link travel time and intersection delay. If the distance between two contour lines is very narrow at a specific area, it indicates the average travel speed within this area is slow in the direction of leaving the central point of the contour map. There are many other ways to interpret the travel time contour map. Angles between segments of one contour line show the relative speed between different directions. Contour lines with different colors indicate the accessibility from the center. Even though the travel time contour map does not present temporal information, the diffidence between two contour maps is much easier to recognize than the difference between status maps.

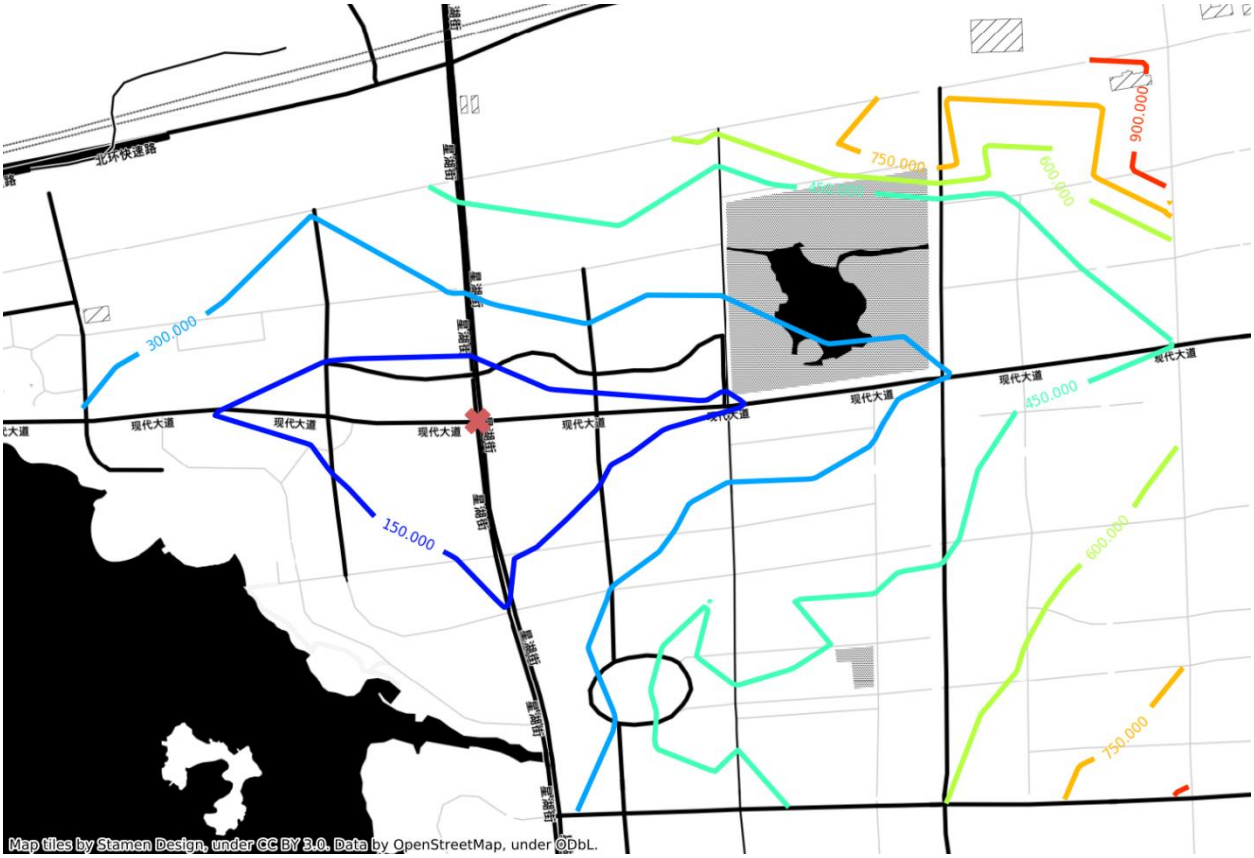


Figure 33 Travel time contour map

Figure *33* shows a travel time contour map created by the embedded visual analysis tools. When the user specifies the center and a time interval, the visual analysis tools can automatically update the travel time contour map. In Figure *33*, the center specified by users is highlighted with a red cross. It locates on two arterial roads in the traffic. Each contour line shows how far vehicles can reach from the center of the contour map. Smaller distances between contour lines on the north side of the center illustrate that vehicles are moving relatively slower when vehicles move toward the north side of the traffic network. Travel time contour map also has its limitations:

- Similar to the traffic status map, the travel time contour map only presents the spatial pattern.

- From the travel time contour map, users can not identify the exact road segment or intersection that is experiencing the lower performance.

5.2.3 Spatial-temporal heatmap

As the backbone of a traffic network, arterial roads carry heavy traffic and have significant influences on the entire network if congestion happens. The bottlenecks on the arterial are the keys to reduce congestion in traffic networks. A heatmap is implemented for bottleneck identification along arterials. Heatmap presents the magnitudes of measurement as different colors in two dimensions. The spatial-temporal heatmap has one axis expressing the time and the other representing the road segments or intersections along one direction of the arterial. In the heatmap, each ceil measures the travel time between adjacent intersections:

$$Ceil_{i,t} = TravelTime_{I_i,I_{i+1},t}$$

Where

$TravelTime_{I_i,I_{i+1},t}$ = the travel time between I_i and I_{i+1} during time period t

It reflects the distributions of bottlenecks over time and space. After congestion is formed at one intersection, it may propagate to adjacent road segments or intersections. Analyzing the temporal relationship of congestions helps to understand how congestions propagate and identify the key bottleneck on the arterial.

Figure 34 and Figure 35 illustrate an example of using the spatial-temporal heatmap to identify bottlenecks along an arterial. Xinhu Road is a north-south arterial in Suzhou. Figure *34* shows the layout of the arterial. Eight road segments along the arterial are included in the traffic network of the field application. Figure 35a and Figure 35b are heatmaps representing the travel speed of

northbound and southbound traffic along Xinhu Road. Overall, it is clear that the southbound experience heavier congestion over time and space. It can be also observed that both northbound and southbound traffic experience higher travel time during peak hours. The congestions of northbound appear at peak hours and are quickly relived after the peak hours. On the contrary, the congestion of southbound keep affecting the performance of several road segments of the southbound. Compared with other road segments, Suhong Road to Shenhu Road, Shenhu Road to Jiuhua Road, and Jinhua Road to Xiandai Ave are more heavily affected. To help traffic operators automatically find out bottleneck candidates, a threshold can be set to categorize each ceil within the heatmap classified as congested or not. In this case, 25 kilometers per hour is chosen:

$$Bottleneck_{i,t} = \begin{cases} 1 \; if \; Ceil_{i,t} < H \; km/h \\ 0 \; if \; Ceil_{i,t} \geq H \; km/h \end{cases}$$

Where

H = the threshold used to classify if a road segment is congested or not.

Figure 35c and Figure 35d show the results of processing. Black cells are the cells with traffic speeds lower than 25 kilometers per hour. This transformation may look redundant in this example but it could help users to locate potential bottlenecks quickly in a larger heatmap. Besides, it can also be used to measure the "bottleneck intensity" of arterials or intersections (Margiotta, Spiller, and Systematics 2009) The intensity index is calculated as the ratio between ceils recognized as congested and the total number of ceils. The intensity index of the bottleneck makes the conditions among intersections or arterials comparable.

Figure 34 Xinhu Road

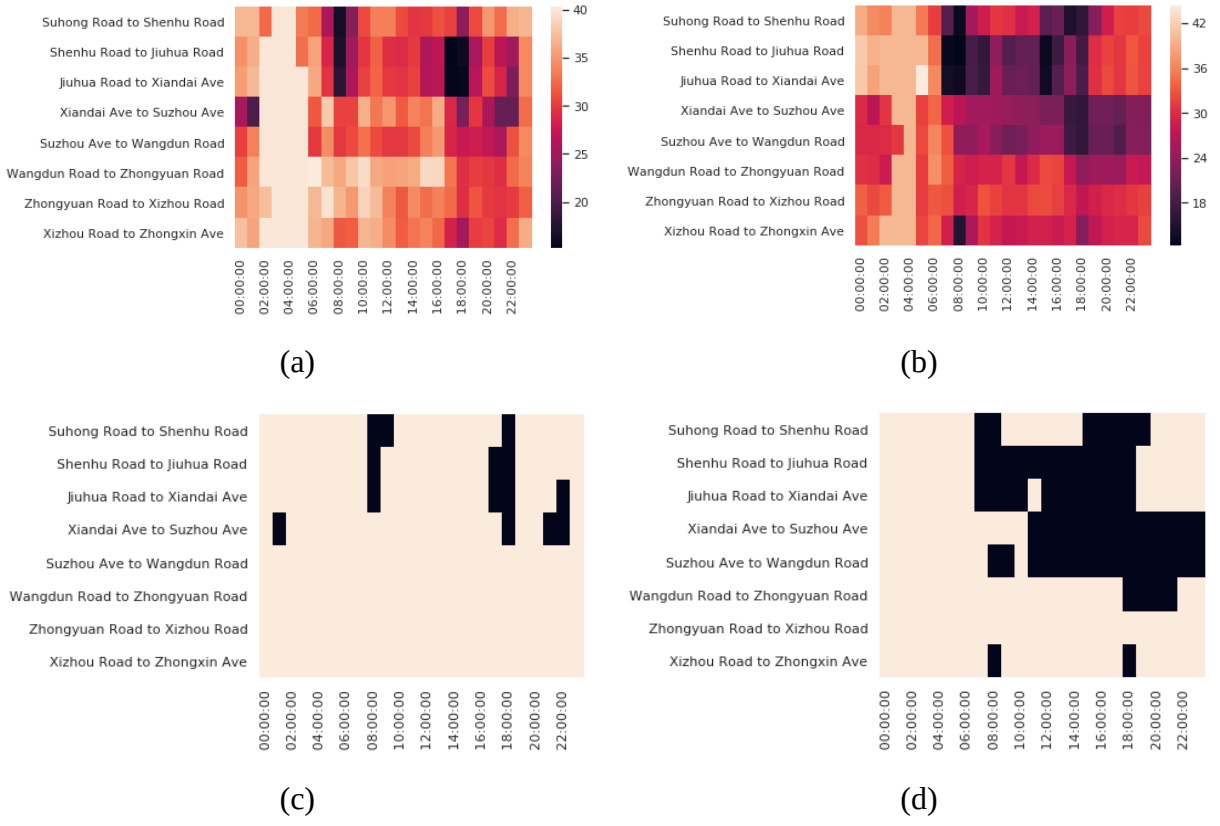


Figure 35 Spatial-temporal heatmap

5.2.4 Themeriver diagram

Themeriver diagram provides is an alternative to visualize the spatial-temporal pattern of bottleneck distribution. Themeriver diagram was first invented to keep track of the thematic variations of documents over time (Havre et al. 2002). The strength of the theme river diagram is to illustrate the total variation of measurement over time and simultaneously show how variations of components contribute to the total variation. The horizontal axis represents time and different components are shown as flows in different colors. The total variation and variations of components are revealed along the horizontal axis while the vertical height of components presents the contribution of components to the total variation. The proposed implement the themeriver diagram for revealing the main contributor of total delay along an arterial or at an intersection.

Figure 36 shows an example of using a themeriver diagram to visualize delay at an intersection over time. Four flows with different colors present the variation of delay of four approaches of the intersection. The largest total delay happened at around 18:00, and themeriver diagrams show the variation of delay of westbound contribution to half of the total delay during that time. As other tools discussed above, users can customize the diagrams depending on how they would like to explore the data. That is one of the main reasons why the system is deeply modulized. For example, after noticing that the increasing total delay on evening peak is mainly caused by variation of delay on westbound. Users may also check the traffic demand on each bound to see if the unbalanced increasing traffic demand on westbound leads to the tremendous increasing delay. Figure *37* shows the traffic demand of different approaches to the intersection with the themeriver diagram.

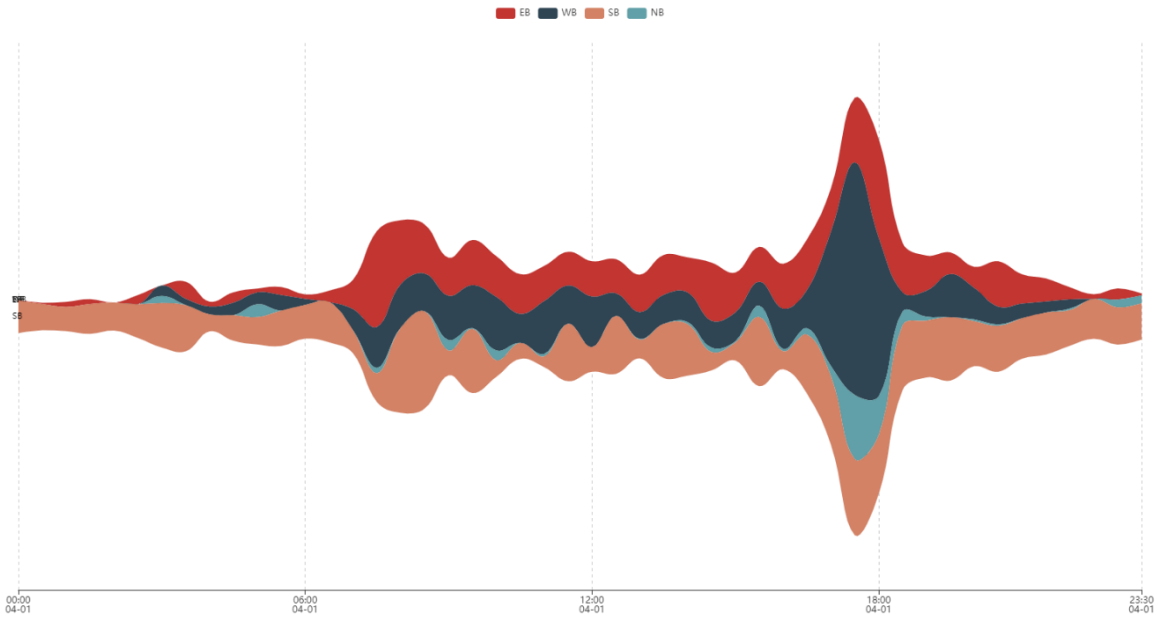


Figure 36 Themeriver diagram (delay)

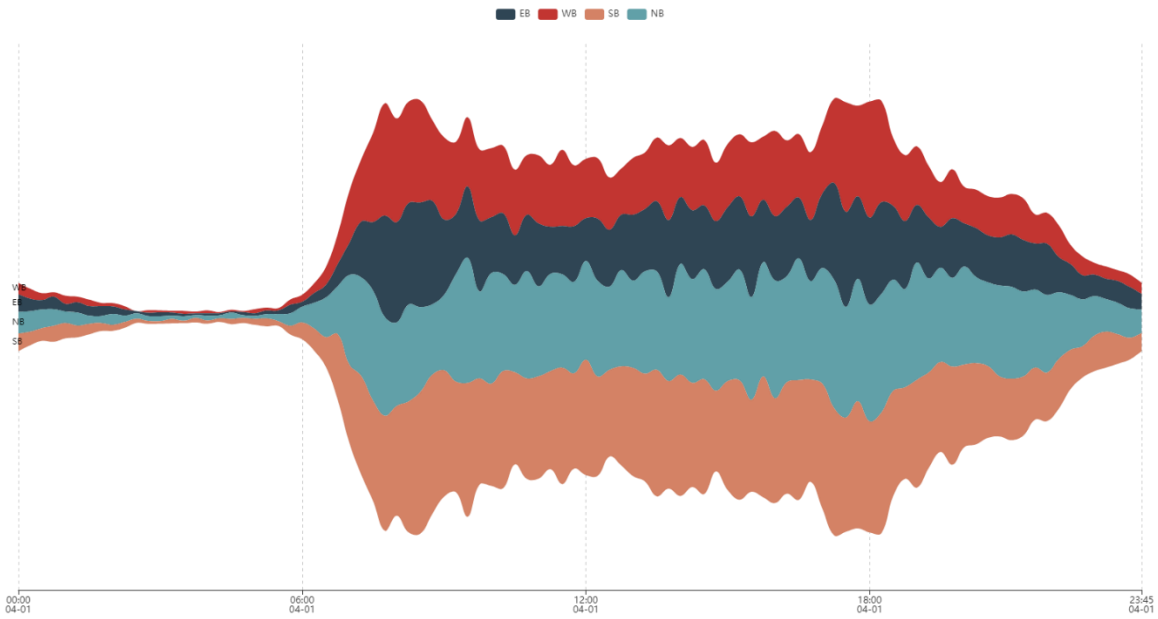


Figure 37 Themeriver diagram (traffic demand)

In the bottleneck identification module, a variety of visual analysis tools are implemented to help users to identify intersections or road segments that have lower performance, recognize the spatial-temporal pattern of the bottleneck, and trace back the causing effects of the bottlenecks.

Each tool has its only strength and limitations. To take advantage of the strength of these tools, the tools are modularized. Depending on the use case, users can decide how these tools are integrated.

5.3 Traffic demand analysis

Traffic demand management relieves bottlenecks by redistributing traffic demand over time and space. Traffic demand analysis is the key to perform traffic demand management. The traffic demand is mainly measured by traffic volume on links, along routes, and over traffic networks. Several tools discussed in the previous module can be applied to visualize the traffic volume on links. In this module, multiple visual analysis tools are implemented to visualize traffic demand along routes and over traffic networks.

5.3.1 Origin-destination flow map

Traffic demands over traffic networks are measured by origin-destination flow matrices. Origin-destination flow maps visualize the matrices to reveal the spatial pattern of the traffic demand. In the origin-destination flow map, traffic demands of origin-destination pairs are represented by directed weighted edges. The width of edges illustrates the magnitude of traffic demand:

$$WidthOfEdge \propto OD_{I_i,I_j}$$

The direction of traffic demands is usually represented by the color gradients of the edges. To make the edges between origin-destination pairs more distinguishable, most origin-destination flow maps plot the edges as curves instead of lines. A general implementation of the origin-destination flow map is embedded in the proposed system.

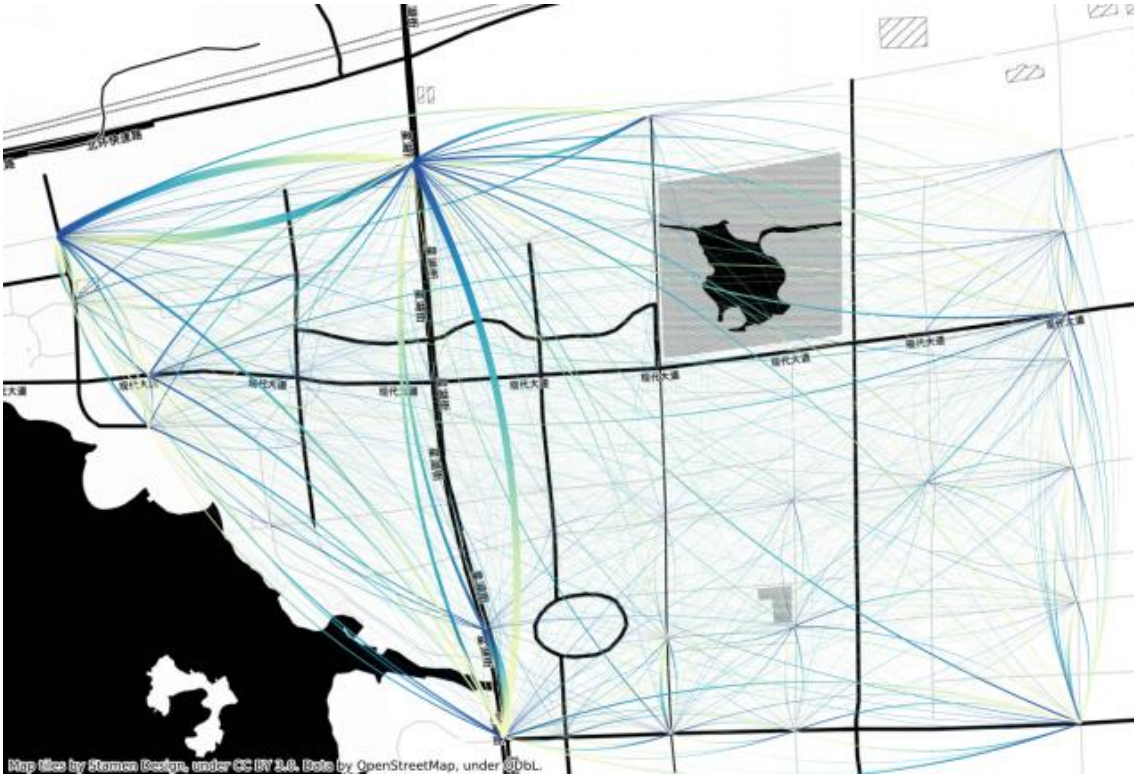


Figure 38 Origin-destination flow map

Figure *38* presents the daily origin-destination matrix of the traffic network in the field application. A few wider curves can be recognized on the map. These wider curves indicate origin-destination pairs with higher traffic demand. However, many curves in the map seem redundant. No high-level pattern can be recognized from these separately visualized traffic demands of origin-destination pairs. Besides, as the size of the traffic network increases, a large number of curves or edges on the origin-destination map will result in visual clutter. Several treatments of visual clutter are discussed in the literature review. Two visual analysis tools are implemented to visual clutter problems.

5.3.2 Edge-bundled origin-destination flow map

Many studies have proven that edge bundling reduces the visual clutter in the flow map (Holten and Van Wijk 2009; Selassie, Heller, and Heer 2011; Ersoy et al. 2011). By deforming edges

into curves and combines similar edges into bundles, edge bundling reduces the complexity of the flow map and reveals aggregated patterns in the flow map. In the proposed system, an edge-bundling origin-destination flow map is implemented. The implementation is based on forced-directed edge bundling (Holten and Van Wijk 2009). The main idea of forced-direct is shown in Figure 39. Edges are separated into segments. The division points on the edges are assumed to be attracted by springs forces F_s and electrostatic force F_e. The combined force on division point p_i is:

$$F_{P_i} = k_p \left(\left| ||p_{i-1} - p_i|| \right| + \left| ||p_i - p_{i+1}|| \right| + \sum_{Q \in E} \frac{1}{||p_i - q_i||} \right)$$

The combined force will bundle the edges close to each other and reduce the visual complexity of the map and reduce visual clutter.

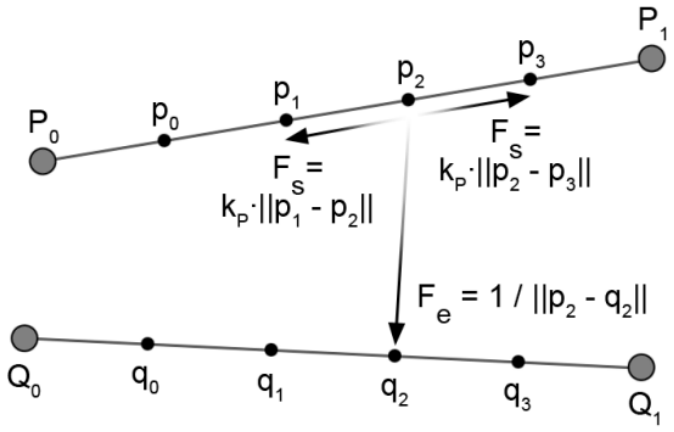


Figure 39 Force-directed edge bundling (Holten and Van Wijk 2009)

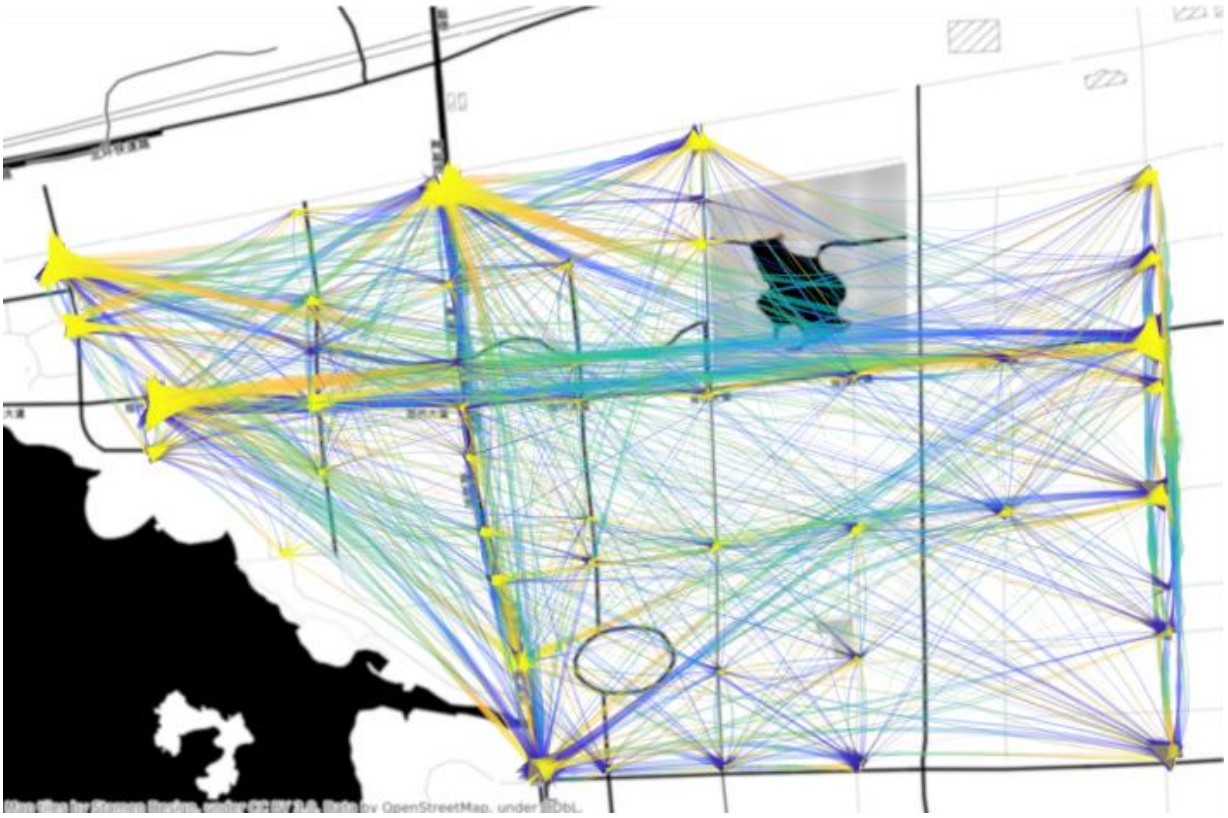


Figure 40 Edge-bundled origin-destination map

Figure *40* presents the daily origin-destination matrix of the traffic network in the field application with the edge-bundling origin-destination flow map. Compared with Figure *38*, it reveals more patterns of the traffic demand on the traffic network. The high traffic demand along two arterials is very obvious when the traffic demands are aggregated by the edge bunding technologies. Besides, on the east-south side of the traffic network, diagonal traffic demand patterns show up after tiny but dense edges in Figure *38* get bundled together.

5.3.3 Clustered origin-destination chord diagram

Chord Diagrams help to depict matrix, which makes it a good tool to show the O-D pattern of a network. In an O-D chord diagram, each arc represents the traffic volume originated at an intersection and each ribbon represents O-D pairs between two intersections. The degree measure

of the arc representing I_i quantify the ratio between the traffic volume originated at the intersection and total traffic demand over the traffic network:

$$DegreeOfArc_{I_i} = \frac{\sum_j OD_{I_i,I_j}}{\sum_i \sum_j OD_{I_i,I_j}}$$

The degree measure of the ribbon representing traffic demand from I_i to I_j quantify the ratio between the O-D pair flow and total traffic demand over the traffic network:

$$DegreeOfRibbon_{I_i} = \frac{OD_{I_i,I_j}}{\sum_i \sum_j OD_{I_i,I_j}}$$

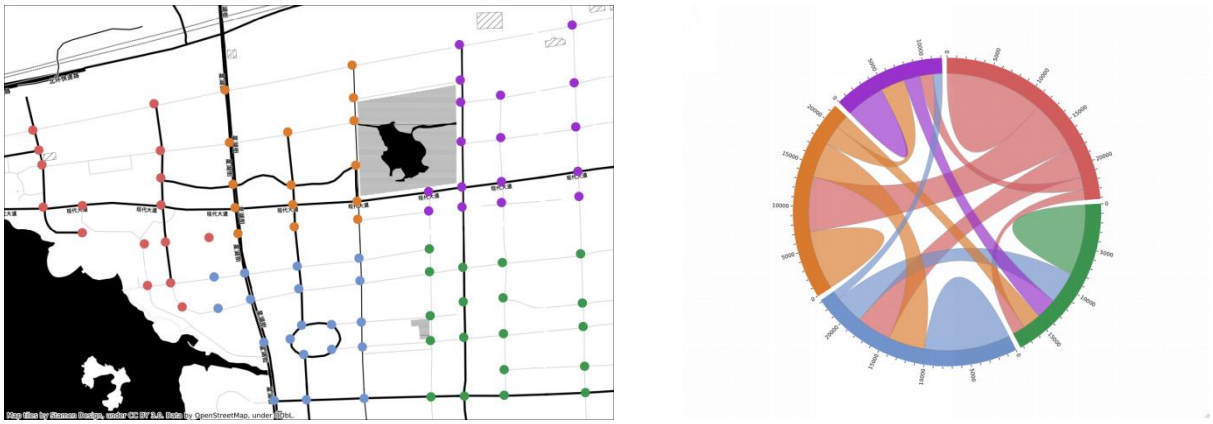
Like origin-destination maps, O-D chord diagrams lose track of high-level traffic patterns on large maps. As the size of the network increase, ribbons overlap each other. Besides, because the chord itself does not include geospatial information, it is hard to understand the geospatial distribution of the pattern appearing in a chord diagram. To solve these problems, the system integrates the chord diagram with clustering technology. Intersections in the traffic network are first grouped into regions. If users have a preference on how intersections should be grouped, users can predefine traffic analysis zones. Otherwise, the visual analysis tool clusters the intersections based on the distance and traffic flow between intersections. Then the origin-destination chord diagram visualizes the travel demand among regions. In the clustered origin-destination chord diagram, each arc represents the travel demand originated in a region. The degree measure of the arc representing R_i quantify the ratio between the traffic volume originated at the region and total traffic demand over the traffic network:

$$DegreeOfArc_{R_i} = \frac{\sum_j OD_{R_i,R_j}}{\sum_i \sum_j OD_{R_i,R_j}}$$

$$OD_{R_i,R_j} = \sum_{I_k \in R_i, I_l \in R_j} OD_{I_k,I_l}$$

The degree measure of the ribbon representing traffic demand from $R_i$to R_j quantify the ratio between the O-D pair flow and total traffic demand over the traffic network:

$$DegreeOfRibbon_{R_i} = \frac{OD_{R_i,R_j}}{\sum_i \sum_j OD_{R_i,R_j}}$$



(a) (b)

Figure 41 Clustered origin-destination chord diagram

Figure 41 present an example of using a chord diagram to visualize the traffic demand over the traffic network in the field application. The intersections are classified into six groups and the intersections in different groups are expressed with different colors. *Figure 41*b shows the corresponding chord diagram after grouping. From the chord diagram, it can be recognized that the main traffic demand comes from the group of intersections in red to the group of intersections in orange, which corresponds to the traffic demand from the upper left region of the network to the upper middle region of the network.

5.3.4 Backtracing diagram

Incidents are the main contributors to non-recurrent congestions. Evacuations caused by incidents require intensive efforts on transportation professionals to guide traffic. A gate control strategy is can be applied to improve the performance of these evacuations by guiding evacuation traffic to go through selected "gate" nodes or near the Protective Action Zone (PAZ) boundary with large access and throughput capacities (Bu et al. 2016). However, so far it is only a theoretical idea. Real-time traffic backtracing and O-D extraction are required to realize gate control strategy, but they are not available until the appearance of a video surveillance system and GPS. This proposed system implemented a backtracing diagram to trace back traffic using reconstructed trajectories of vehicles. It provides support for location critical gate and gate control boundaries. Besides, the integration of critical path sub-module and traffic backtracing sub-module enables to revision of the traffic from gates to target intersections. In the backtracing diagram, the area of circles quantify the traffic demand from intersections to the intersection I_d:

$$AreaOfCircle_{I_i} \propto OD_{I_i,I_d}$$

The proposed system can provide support to decide the location of critical gates and gate control boundaries. Figure *42* shows an example of the backtracing diagram. The red node in the diagram is the intersection with the critical intersection in the traffic network. So, it is chosen as an example of traffic backtracing. The size of blue circles represents the number of trips that stem from corresponding intersections. As shown, almost all of the trips start from two arteries, Xinhu Street and Xiandai Avenue. And three largest traffic demands come from the boundary of the network, which means a large number of traffic trips going through the intersection stem from places beyond this network in the field application. If traffic operator wants to utilize a gate control strategy, these results could help to better locate critical gates and define gate control boundaries.

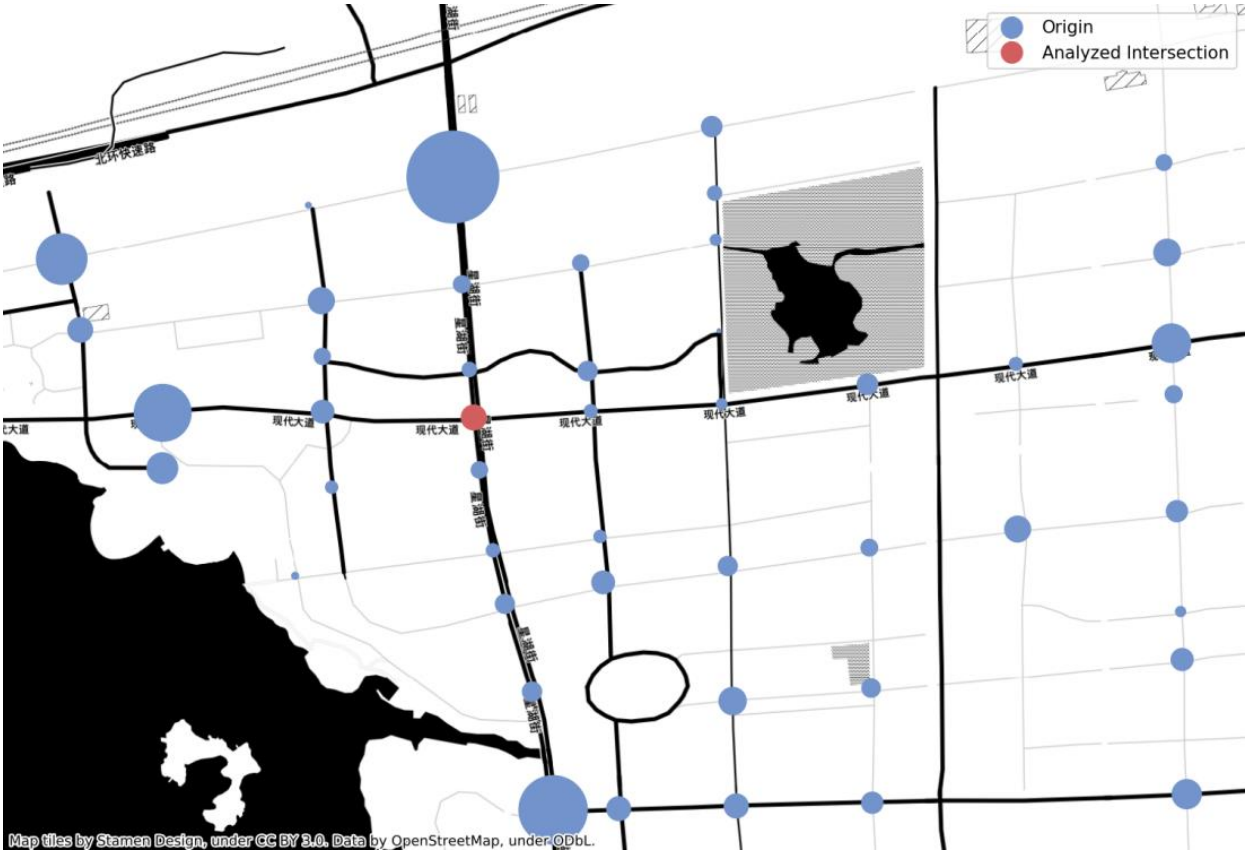


Figure 42 Backtracing diagram

5.3.5 Critical path diagram

To reduce delays and stoppages, many studies about how to synchronize signals between intersections. In most studies, the green bandwidth of arterial streets or paths is used as the objective of optimization to improve the efficiency of entire networks. These arterial streets or paths are considered as the key to network signal synchronization and they are also called “critical paths”. However, critical paths are pre-defined in these theories and implementations. Few studies discussed how to achieve these critical paths using real-world data. Critical paths can be recognized by the number of vehicles moving through these paths. A critical path diagram is implemented in the proposed system to visualize the findings of the critical paths in the traffic network.

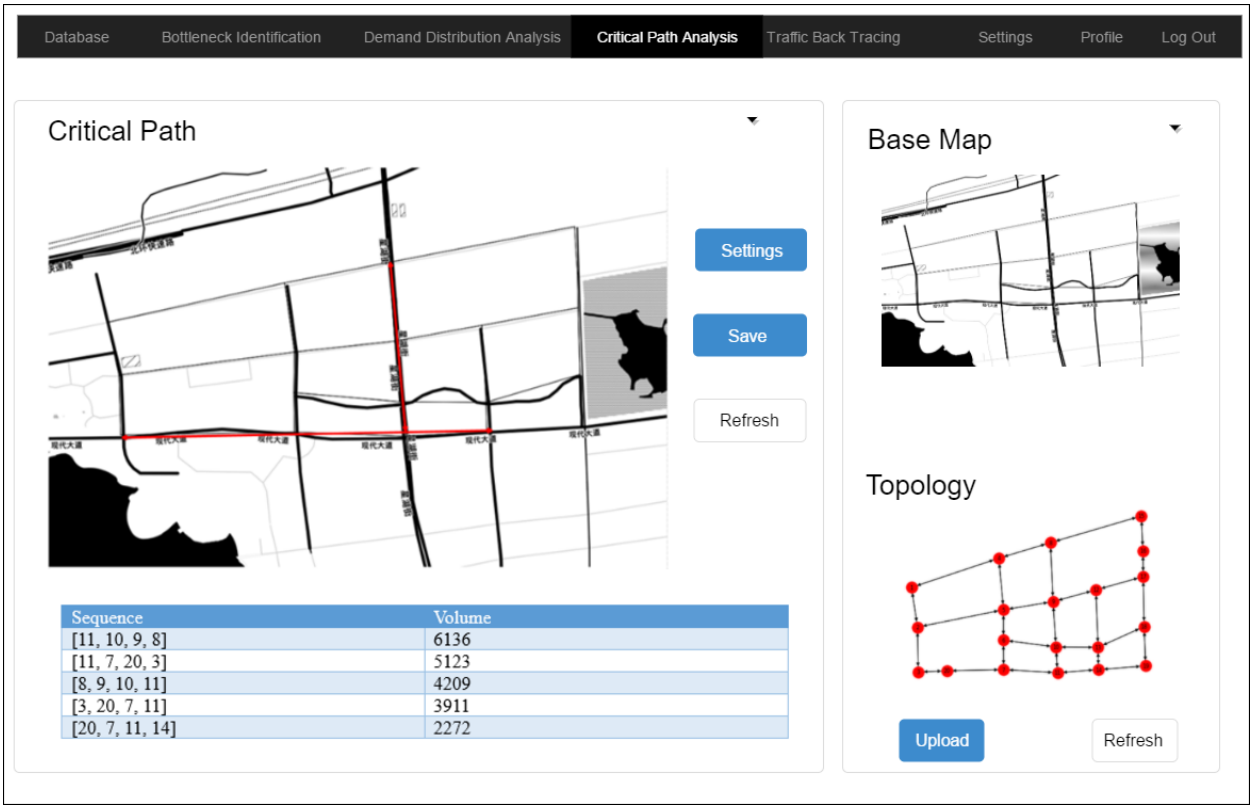


Figure 43 Critical path analysis

Figure *43* shows an example of the critical path diagram with field application data. This diagram helps users to find out the critical paths for synchronization. In this example, the critical path with length three is found and visualized. It can be found out that there are 258 paths with length three in this network. The top 5 critical paths with length three are shown in a table. By overlapping these sequences, the critical path with a higher length can be found and the result is shown on the map. If synchronization of signals is applied in this network, these paths should be mainly concerned. Besides, advanced settings are available in the proposed system to achieve critical paths in different time intervals. So, the users may design a time-varying signal synchronization strategy to better meet traffic demands in different time intervals.

Overall, a variety of visual analysis tools are embedded in the proposed system for traffic demand analysis. These tools enable users to reveal the temporal-spatial traffic demand patterns at different

levels. Besides, these tools are highly customizable and modularized. Users can interactively explore the data depending on their preference.

Chapter 6: Traffic control and management decision support system

6.1 Introduction

Traffic control and management have been widely applied to improve the efficiency and safety of existing traffic infrastructures. Since the invention of traffic control and management, the theories and systems have been continuously improved. In recent decades, these theories and systems have become more and more dependent on data. As an innovative traffic sensor technology, the ALPR outperforms other sensor technologies in many aspects. In the last two chapters, the capability of ALPR data on traffic condition reconstruction and traffic visual analysis is discussed.

This chapter illustrates how the ALPR-based traffic condition reconstruction can support the decision-making procedure of traffic control and management. Furthermore, a traffic control and management decision support system is implemented in the proposed traffic diagnosis and management laboratory. The system aims to provides modularized tools for decision supports. The decision support system mainly takes advantage of the results from ALPR-based traffic condition reconstruction to support traffic control and management decisions. On the other hand, because the system does not directly rely on ALPR data, it keeps the flexibility to work with other data sources. The system supports the decision-making of traffic control and management at two levels: intersection level and arterial level.

6.2 Intersection Level

At the intersection level, the system supports the decision making of traffic control and management in three aspects:

- Evaluation and reporting: The decision support system produces a comprehensive report for each intersection. The report including information about the layout design, traffic demand, signal plan, and performance evaluation.

- Time-of-day breakpoints: Based on the traffic demand variations, a clustering algorithm is designed to find time-of-day breakpoints for multiple signal plans.

- Phase split and cycle length optimization: For each signal plan of time-of-day signal control, the phase split and cycle length are optimized based on traffic demand estimation.

6.2.1 Evaluation and reporting

The first task of the decision support system is to help users to understand the current condition of the intersection. The reporting tool is implemented to produce a comprehensive report for each intersection. It fetches signal plan data and results from traffic condition reconstruction to create content in the report. To make the reporting tool more extensible, an Excel spreadsheet is designed as an alternative way for data inputs. The Excel spreadsheet mainly consists of three parts:

- Geometric design and traffic volume: This part mainly collects information about the geometric design of the intersection, such as the number of outbound lanes, turning movements. Besides, it also collects traffic volume information.

- Phases and timing: It contains the phase ID for each turning movement.

- Ring and barrier diagram: The diagram collects the phase sequence and phase duration of the signal plan.

Figure *44*, Figure *45*, and Figure *46* present examples of these three parts of the spreadsheet respectively.

Links	EB						
Link Name	EB Rd						
Link ID	2						
Link Orientation (Azimuth, degrees)	270						
Outbound Lanes (#)	2						
Link Length (m)	500						
Link Speed (km/h)	30						
Travel Time (s)	60						
Area Tyep (CBD?)	Yes						
Turning Movements		Left	Left-2	Through	Right	Right-2	
Lanes and Sharing	-	L1	L1.5	T2	R1.25	-	-
Turning Destination Link ID		3	5	1	4	6	
Traffic Volume (vph)		150	50	550	50	50	
Lane Width (m)	3.50	3.50	3.60	3.40	4.00	3.50	3.50
Grade (%)	0						
Storage Length (m)	-	-	50	-	50	-	-
Storage Lanes (#)	-	-	2	-	1	-	-
Taper Length (m)	-	-	25	-	25	-	-
Right-turn Channelized	-	-	-	-	-	-	None
Left Turn Waiting Area?	-	-	No	-	-	-	-
Through Waiting Area?	-	-	-	No	-	-	-
Peak Hour Factor	0.92	0.92	0.92	0.92	0.92	0.92	0.92
Growth Factor	1.00	1.00	1.00	1.00	1.00	1.00	1.00
Ideal Saturation Flow (vphpl)	1700	1700	1700	1800	1750	1750	1750
Conflicting Peds. (#/hr)	0	0	0	0	0	0	0
Conflicting Bikes (#/hr)	0	0	0	0	0	0	0
Heavy Vehicles (%)	2	2	2	2	2	2	2
Bus Blockages (#/hr)	15	0	0	0	0	0	0
Parking Maneuvers (#/hr)	16	17	18	0	0	0	0
Traffic From Mid-block (%)	0						
Pedestrian Flow (#/hr)	0						
Bike Turning Movements			Left	Through	Right		
Bike Flow (#/hr)	0	0	0	0	0	0	0
Adjusted Flow (vph)	0	163	54	598	54	54	0
Lane Group Flow (vph)	-	163	54	598	109	-	-
Lane Utilization Factor	-	1.00	1.00	0.95	1.00	-	-
Right-turn Factor	-	1.00	1.00	1.00	0.85	-	-
Left-turn Factor (Protected)	-	1.00	1.00	1.00	1.00	-	-
Left-turn Factor (Permissive)	-	1.00	1.00	1.00	1.00	-	-
Lane Width Factor	-	1.00	1.00	1.00	1.00	-	-
Grade Factor	-	1.00	1.00	1.00	1.00	-	-
Area Type Factor	-	0.90	0.90	0.90	0.90	-	-
Heavy Vehicles Factor	-	1.00	1.00	1.00	1.00	-	-
Bus Blockage Factor	-	1.00	1.00	1.00	1.00	-	-
Parking Factor	-	0.82	0.81	1.00	1.00	-	-
Lane Group Saturation Flow (vph)	-	1247	1239	3078	1339	-	-
Manually Adj. Lane Group Saturation Flow (vph)							
Manually Adj. Lane Group Flow (vph)							

Figure 44 Geometric design and traffic volume

Links	EB						
Link Name	EB Rd						
Turning Movements		Left	Left-2	Through	Right	Right-2	
Lanes And Sharing							
Phases		5	5	2	2	2	
Bike Turning Movements			Left		Right		
Bike Phases							
Pedestrian Crosswalk Located At:	Inbound Link			Median	Outbound Link		
Pedestrian Phases							
Pedestrian Green Time (s)							
Green Time (s)	-	19	19	48	48	48	-
Yellow Time (s)	-	4	4	4	4	4	-
All-Red Time (s)	-	2	2	2	2	2	-
Effective Green Time (s)	-	20	20	49	49	49	-
Total Lost Time (s)	3	3	3	3	3	3	3
g/C Ratio	-	0.11	0.11	0.26	0.26	0.26	-
Capacity (vph)	-	135	134	810	352	-	-
Delay (s)	-	227.77	86.53	68.69	57.22	-	-
v/s Ratio (volume/sat. flow)	-	0.13	0.04	0.19	0.08	-	-
Volume To Capacity Ratio (v/c)	-	1.21	0.41	0.74	0.31	-	-
LOS	-	F	F	E	E	-	-
Queue Length (vehs)	-	167	166	138	137	-	-
Approach Delay (s)	96.46						
Approach LOS	F						

Figure 45 Phases and timing

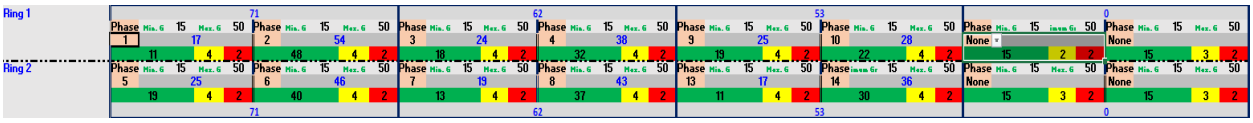


Figure 46 Ring-and-barrier diagram

In the report created by the reporting tool, four diagrams present the traffic flow, existing signal plan, degree of saturation, and level of service of the intersection. The diagram of traffic flow presents the traffic flow at the intersection. Figure *47* shows the layout of the traffic flow diagram. It presents the inbound and outbound traffic flow on each leg of the intersection with bars and arrows. Besides, it reflects the traffic flow on each turning movement with arcs. Figure *48* presents an example of the diagram of phases and timing. The diagram of phases and timing consists of two diagrams: a diagram of geometric design and a ring-and-barrier diagram. The diagram of

geometric design is developed to presents the geometric design of an intersection with no more than six legs. It shows the lanes, turning movements of each leg. The ring-and-barrier diagram presents the existing plan of the intersection. The phase IDs match the IDs in the diagram of geometric design. The diagram of DoS is build up the layout geometric design diagram. It highlights each lane with different colors, which correspond to its degree of saturation. Similar to the DoS diagram, the LOS diagram shows the LOS of each lane by expressing lanes in different colors. The red dot in the middle of the diagram presents the LOS of the entire intersection. Examples of the DoS diagram and LOS diagrams are presented in Figure *49* and Figure *50*.

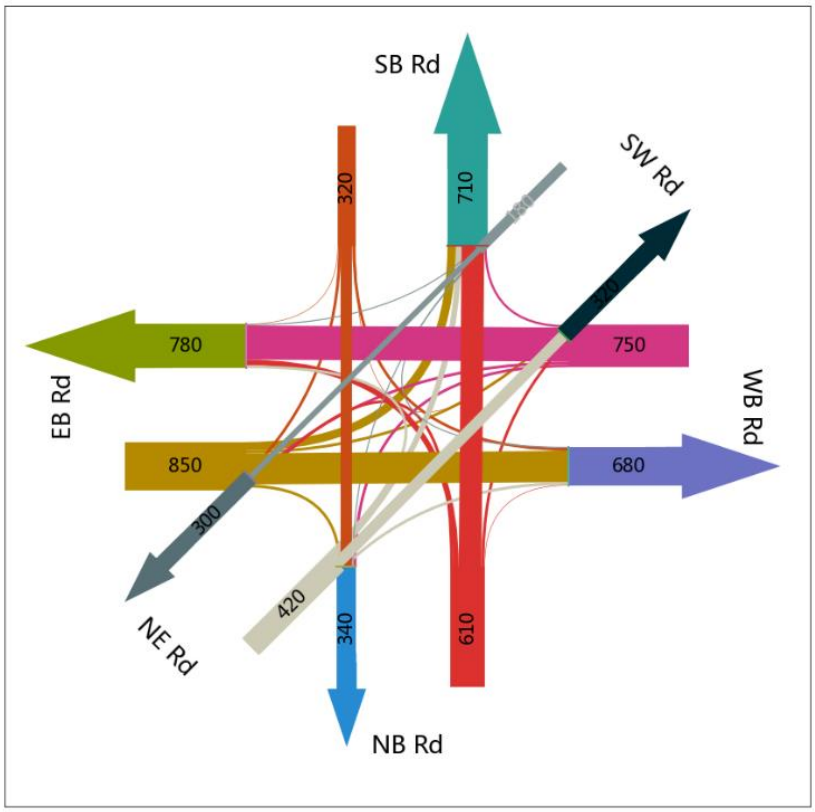


Figure 47 Diagram of traffic flow

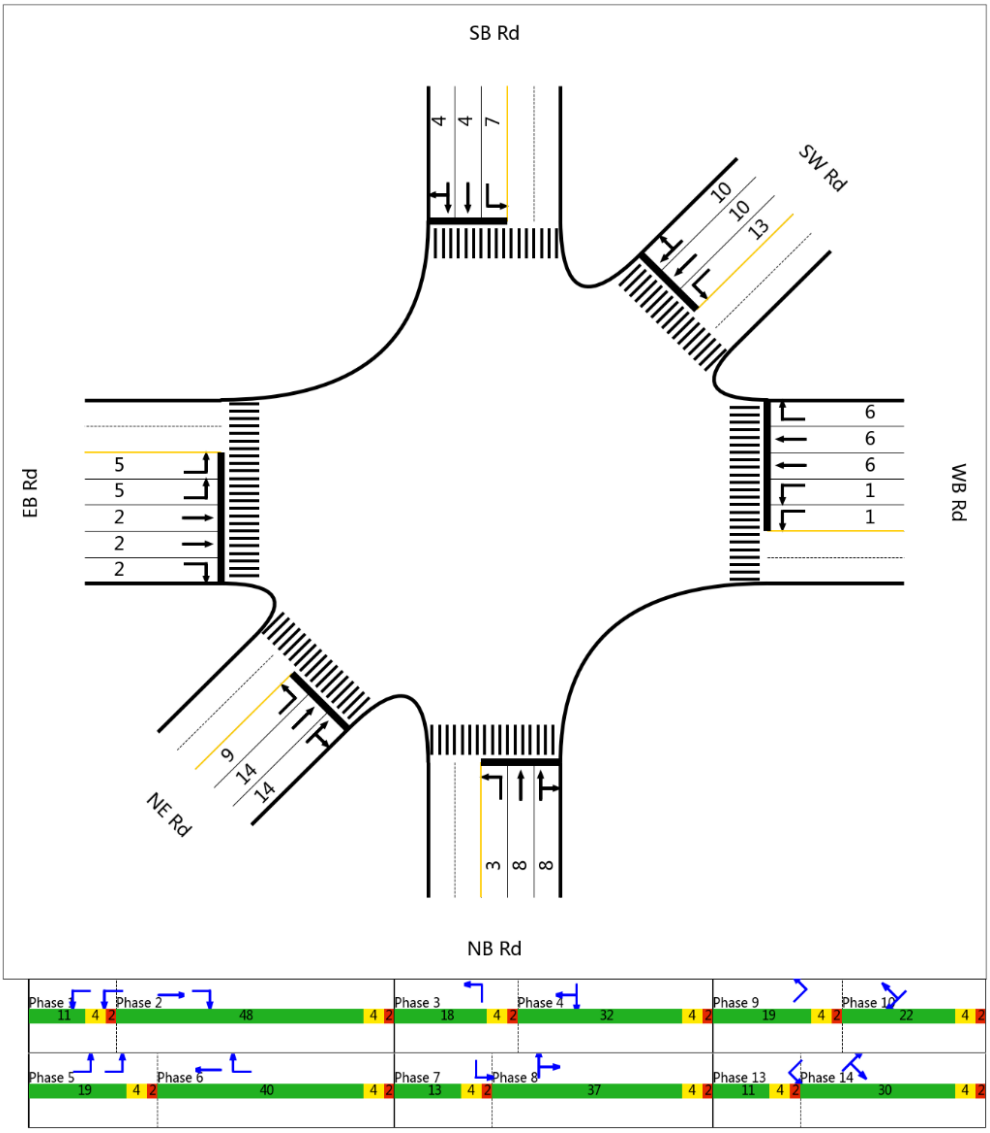


Figure 48 Diagram of phases and timing

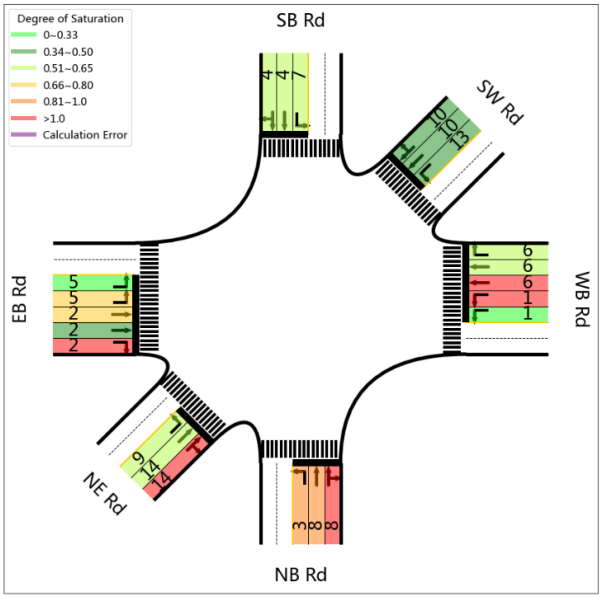


Figure 49 Diagram of DoS

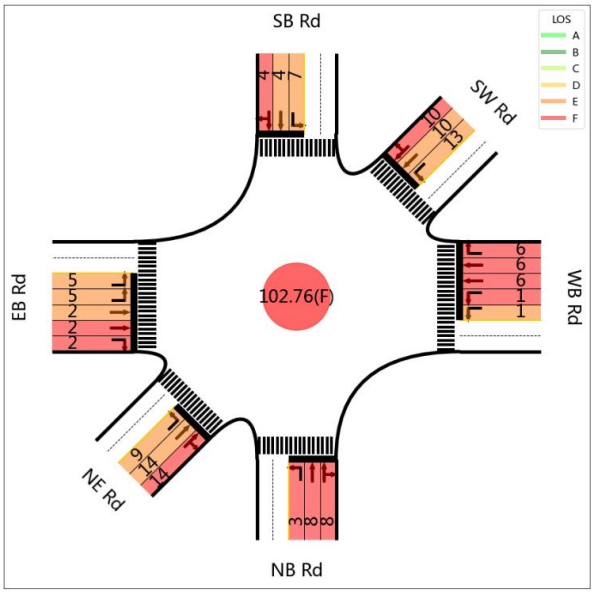


Figure 50 Diagram of LOS

6.2.2 Time-of-day breakpoints

Time-of-day signal control is a type of signal control strategy that does not require an adaptive traffic control system but can design multiple signal plans to match the daily variation of traffic demands. Time-of-day signal control separates a day into multiple groups of time intervals based on traffic demand and designs a signal plan for each group of time intervals. The idea behind this strategy is that the traffic demand in each interval or each group of time intervals is relatively stable. One signal plan can meet the relatively stable traffic demand within one group of time intervals. So, the key to time-of-day signal control is how to find the breakpoints that of time intervals so that the traffic demand within each group of time intervals is relatively stable.

In the decision support system, a clustering algorithm is implemented to decide the time-of-day breakpoints. The algorithm is summarized in Figure *51*. The main idea of the clustering algorithm to transform route volume estimated by ALPR data to phase volumes and find breakpoints based on variations of phase volumes. First, the traffic flow of turning movement is calculated using

route volume. Then, the traffic flow of turning movements is aggregated to get the traffic demand of phases in the signal plan. Clustering is performed to congregate the time interval with similar traffic demand of phases. The number of clusters depends on users' preferences. Otherwise, the algorithm chooses the number of clusters based on Sihouette analysis. Silhouette analysis provides a way to analyze how each data point has been clustered. It measures the distance from a data point to its cluster compared with its distance to other clusters.

Algorithm 3: Time-of-day breakpoint

INPUT

- ALPR records of upstream intersection: UpstreamRecords
- ALPR records of downstream intersections: DownstreamRecords
- Signal plan: Sig
- Number of clusters (Optional): n

OUTPUT

Time-of-day breakpoint

ALGORITHM

$N_R \leftarrow count\ the\ number\ of\ matched\ records\ for\ all\ possible\ routes$

$TrafficVolume_{R,\,t} \leftarrow \frac{N_R}{\prod_{L_j \in L} DR_{L_j,t} * R\ R_{L_j,t}}$

$TurnVol \leftarrow achieve\ turning\ movement\ volume\ from\ traffic\ Volume_{R,\,t}$

$TurnVolSig \leftarrow join\ TurnVol\ with\ Sig$

$TurnVol \leftarrow\ groupby\ phases\ and\ get\ phase\ traffic\ volume\ of\ phases$

$if\ n\ is\ specified$

$\quad Perform\ Kmeans\ clustering\ with\ n\ clusters$

$else$

$$Do \begin{cases} n \leftarrow conduct\ Silhouette\ analysis\ with\ different\ number\ of\ clusters \\ Perform\ Kmeans\ clustering\ with\ n\ clusters \end{cases}$$

$Sort\ time\ intervals\ by\ time$

$TimeOfDayBreakpoint \leftarrow find\ the\ breakpoint\ along\ time$

$Return\ TimeOfDayBreakpoint$

Figure 51 Time-of-day Breakpoints Algorithm

In the rest of this section, the procedure and results of the time-of-day breakpoint algorithm on a field application are presented. Figure *52* shows the traffic volumes in four phases of the intersection in field application. The daily pattern of the traffic demand is very typical. The overall traffic volume at the intersection has two peaks: a morning peak and an evening peak. Between these two peaks, the traffic demand is stable and very close to the traffic volume in peak hours. Besides, the weight of the traffic demand of each phase is relatively stable. The left turn volume is much smaller than the through traffic all over the day.

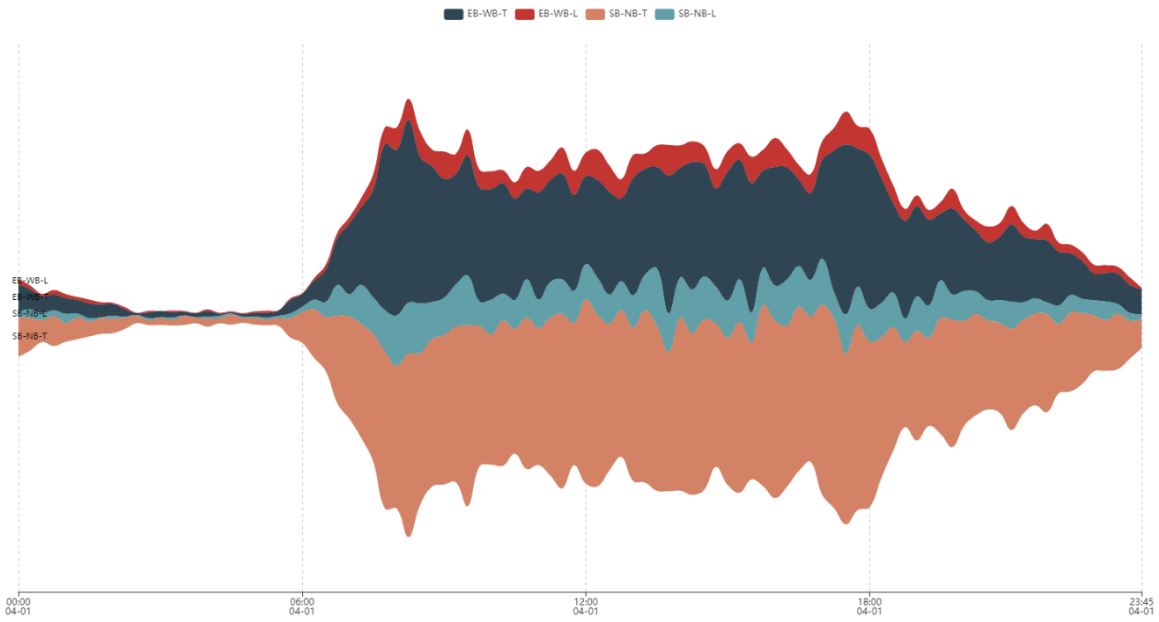
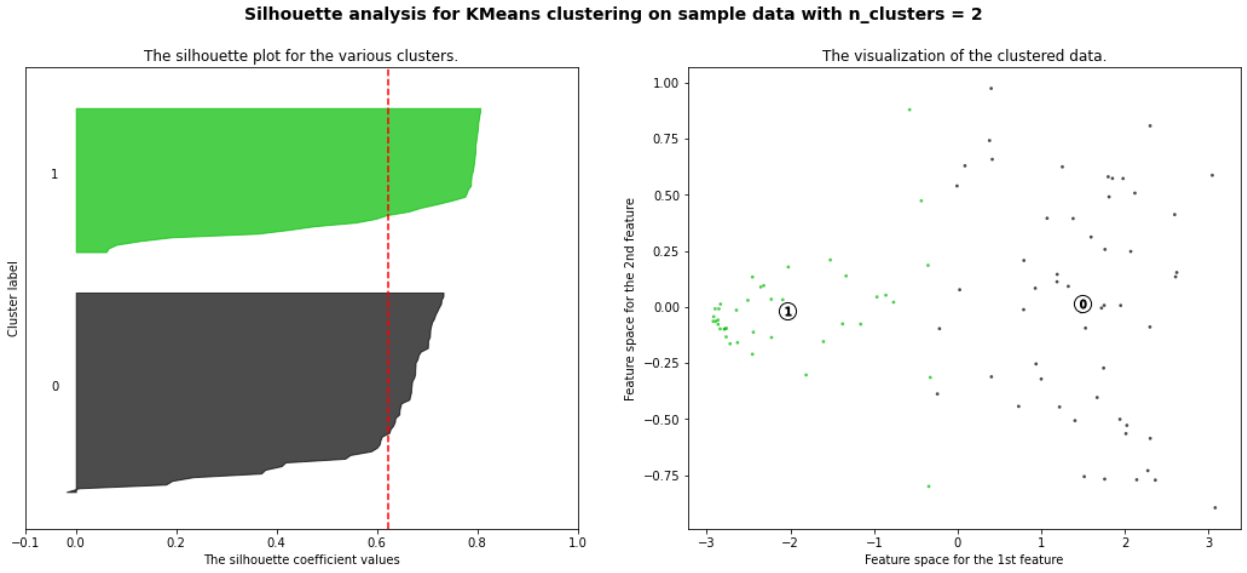


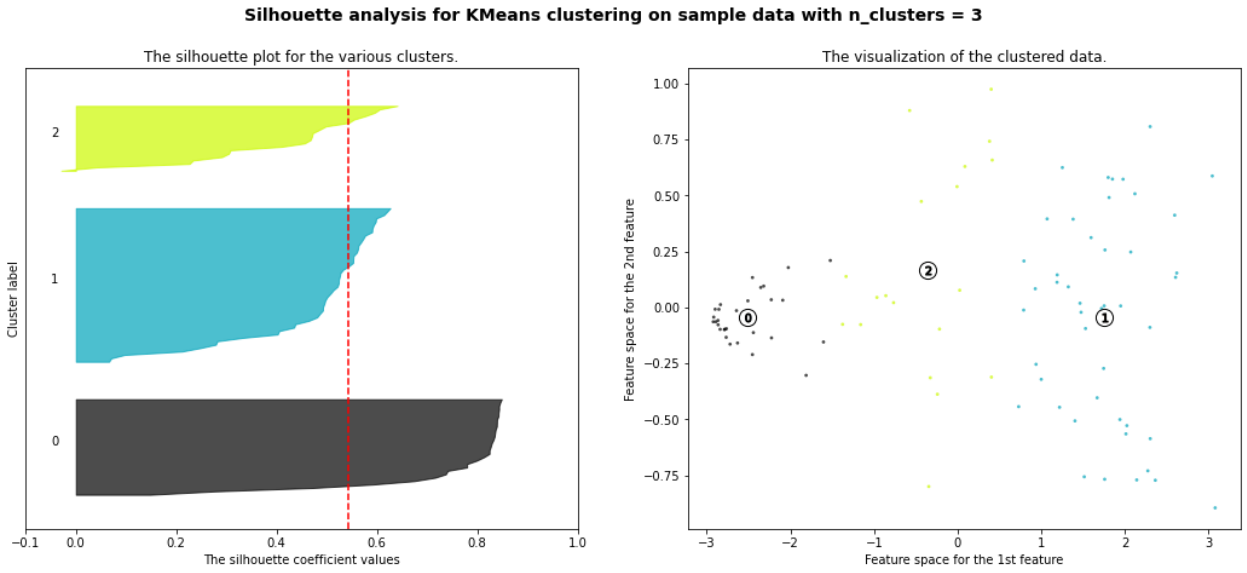
Figure 52 Phase volume

Based on the phase volume, the K-means clustering algorithm is applied to find the time-of-day breakpoints. The Silhouette analysis for different numbers of clusters is shown in *Figure 53*. The left side of the diagram shows the silhouette coefficient values of data points. The vertical red line shows the average silhouette coefficient value of all data points. The plot shows that two or three are better picks. The plot of silhouette coefficient with four clusters shows the coefficient of one

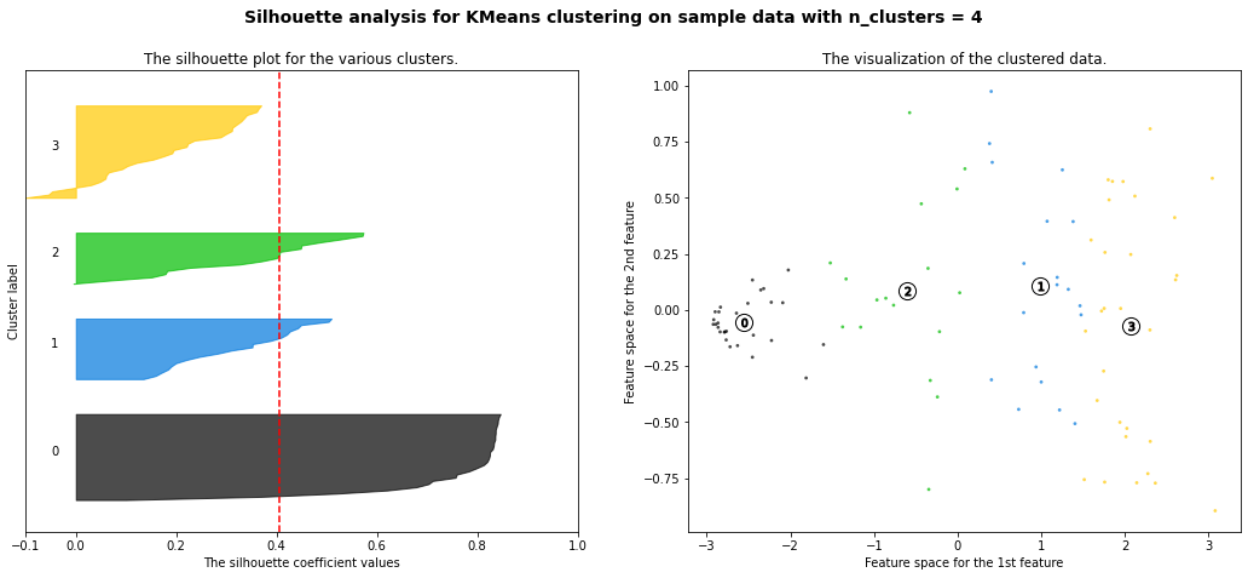
cluster is smaller than the average coefficients. Besides, a few data points in the fourth cluster have negative coefficients, which means the boundary of the two clusters is not clear. The time-of-day results of three clusters are presented in Figure *54*.



(a) n=2



(b) n=3



(c) n=4

Figure 53 Silhouette analysis

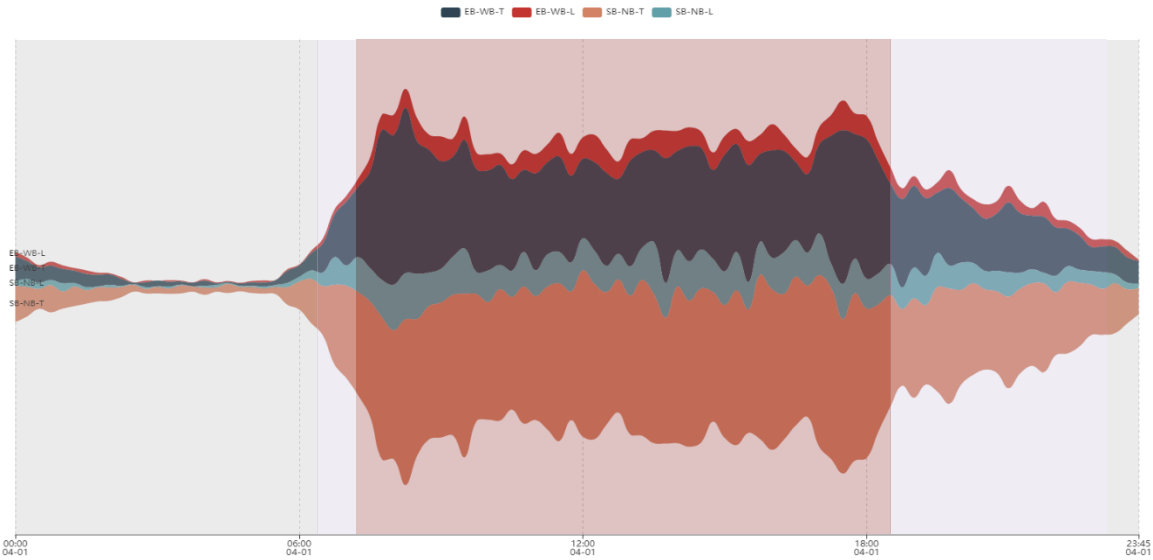


Figure 54 Time-of-day breakpoints

6.2.3 Phase split and cycle length optimization

After time-of-day breakpoints are specified, the decision support system optimizes phase split and cycle length of signal plans. The objective of the optimization is to minimize the total delay of intersections. The HCM 2010 control delay model for signalized intersections is used to model the

total delay of intersections. Studies show the control delay model with default values does not accurately measure the total delay in real-world cases. So, the decision support system calibrates the model before the optimization process starts, which separate the optimization into two steps:

- Model calibration: The HCM 2010 control delay model for signalized intersections formulize the delay as:

$$d = d_1 + d_2$$

$$d_1 = \frac{0.5C(1 - g/C)^2}{1 - min(1, X)g/C}$$

$$d_2 = 900T\left[(X - 1) + \sqrt{(X - 1)^2 + \frac{8kIX}{cT}}\right]$$

$$X = \frac{v}{c} = \frac{CV}{sg}$$

Where

C = cycle length of the signal plan

g = effective green time

X = degree of saturation

v = traffic volume of the lane group

c = capacity of the lane group

s = saturation flow rate

T = duration of the time period

k = incremental delay factor

In this step, the system fetches estimated saturation flow rates from traffic condition reconstruction. Besides, parameters in the delay model are calibrated with estimated delay by minimizing the difference between estimated delay and delay calculated based on the HCM delay model:

$$Min_{lk} \sum_{t} (d_t - d_t^e)^2$$

Where

d_t = delay calculated based on the HCM delay model

d_{t_e}= estimated delay based on ALPR data.

- Optimation: The default values in the HCM delay model are replaced with the estimated degree of saturation and calibrated parameters. Then the decision support system optimizes the cycle length and phase splits by minimizing total delay.

6.3 Arterial Level

At the arterial level, signal control strategies optimized the phase offsets between adjacent intersections to realize “green wave” along arterials. Most adaptive traffic control systems optimized the offsets based on predicted traffic demand, which in term prediction relies on platoon dispersion models. So, one of the keys to improving the performance of a coordinated signal plan is to increase the accuracy of the platoon dispersion model. Robertson’s platoon dispersion model is one of the most widely used platoon dispersion models. It expresses the downstream flow profile as:

$$q_2(j) = \sum_{i=1}^{j-t} F(1-F)^{j-i-t}\, q_1(i)$$

F = smoothing factor

t = minimal travel time between upstream and downstream observation points

In Robertson's platoon dispersion model, the smoothing factor is quantified with two factors:

$$F = \frac{1}{1 + \alpha\beta T_a}$$

Where

T_a = average travel time of vehicles

α = platoon dispersion factor

β = travel time factor, quantified as the ratio between minimal travel time and average travel time of vehicles: $\frac{T}{T_a}$

These parameters are generally considered site-specific and dependent on the geometry of the link and other factors such as grades, curvature, parking conditions, traffic volume, etc. (Robertson 1969). The work performed by Guebert and Sparks (1990) shows that the parameter calibration of platoon dispersion models significantly influences traffic signal coordination plans. So, at the arterial level, the decision support system mainly focuses on how to calibrate the travel time factor and the platoon dispersion factor for each road segment using ALPR data. In the proposed system, an applicable approach is designed. This approach dynamically calibrates the parameters for each road segment using ALPR data. The proposed calibration methodology using ALPR data is

illustrated in Section 6.3.1. In Section 6.3.2, a field application is performed to verify the feasibility of the proposed methodology.

6.3.1 Calibration of Robertson's Platoon Dispersion Model

The calibration methodology aims to calibrate the travel time factor β and the platoon dispersion factor α in Robertson's platoon dispersion model. Because these factors are considered site-specific and depend on traffic conditions. The calibration methodology is designed to calibrate parameters dynamically for each road segment. It calibration the travel time factor and the platoon dispersion factor in two steps. In the first step, the travel time factor is calibrated based on the minimal link travel time and average link travel time:

$$\hat{\beta} = \frac{\hat{T}}{\widehat{T_a}}$$

The minimal link travel time and average link travel time can be achieved from the traffic condition reconstruction module using ALPR data:

$$\widehat{T_a} = AvgLinkTravelTime_L = \frac{\sum_{v \in S_n} t_{d,v} - t_{u,v}}{\sum_{v \in S_n} 1}$$

$$\hat{T} = MinLinkTravelTime_L = Min_{v \in S_n}(t_{d,v} - t_{u,v})$$

where

$t_{u,v}$ = the downstream record time of vehicle v

$t_{d,v}$ = the downstream record time of vehicle v

S_n = the cluster of vehicles that not affected by traffic control

In the second step, the platoon dispersion factor is calibrated by minimizing the difference of predicted traffic flow profile and actual traffic flow profile detected from the downstream ALPR sensor. Different from most existing studies, vehicles between the upstream ALPR sensor and the downstream ALPR sensor may be affected by signal control. To utilize ALPR data for platoon dispersion calibration, the process of queue formation and queue discharge needs to be modeled. Figure *55* illustrates the procedure of platoon dispersion, queue formation, and queue discharge of a platoon between upstream ALPR and downstream ALPR. The flow profile captured by the upstream ALPR is the departure flow rate at the upstream intersection. As the platoon approaching the downstream intersection, the platoon disperse. Some vehicles arrive at the downstream intersection when the traffic light is green. They enter the downstream intersection and get captured by the downstream intersection. The rest of the vehicles stop before the downstream intersection and form a queue. The queue discharge when the light turns green again. The vehicles in the queue get captured by the downstream ALPR when they enter the downstream intersection.

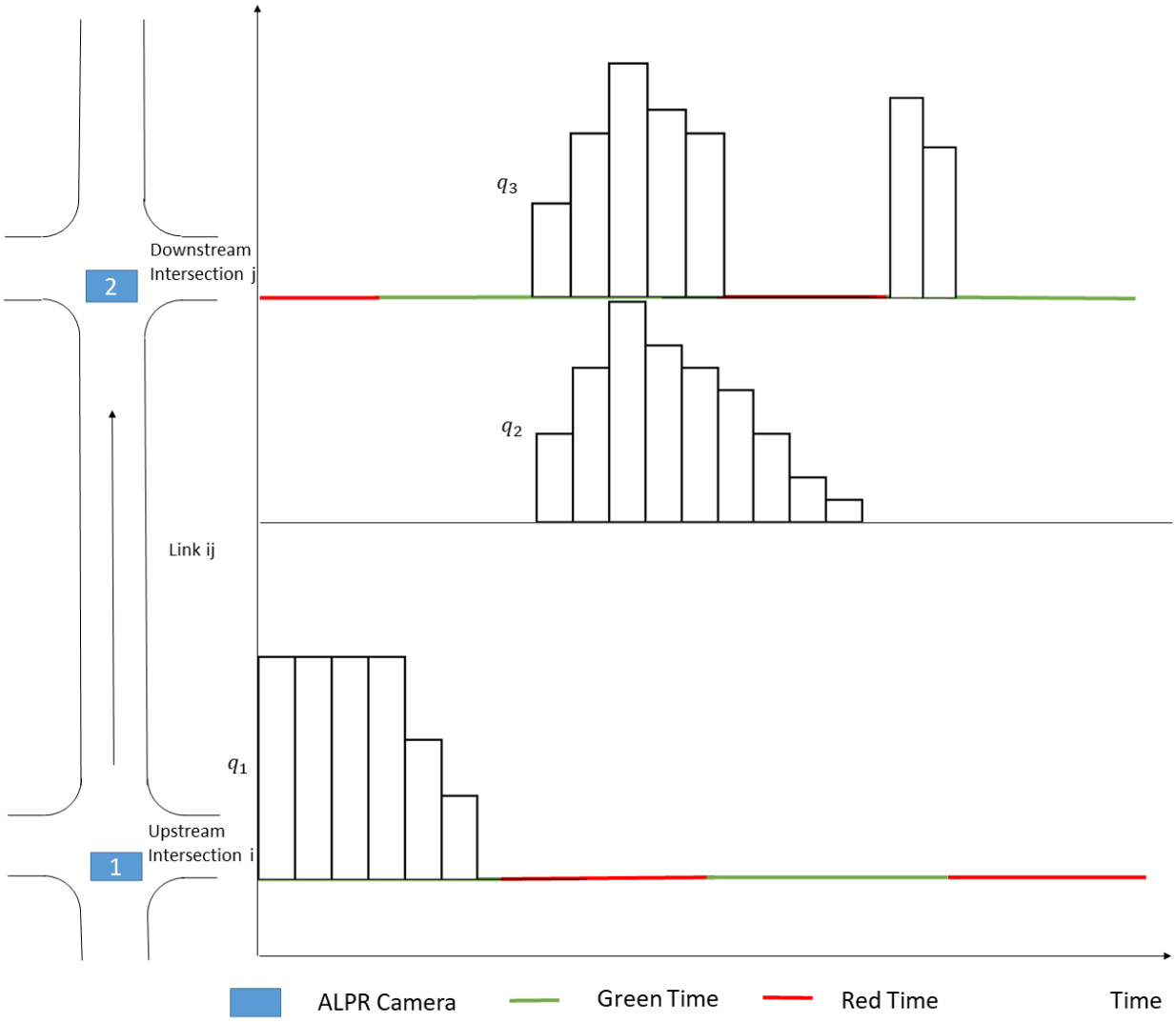


Figure 55 Platoon dispersion, queue formation, and queue discharge

So, the following model is built to model the process of platoon dispersion, queue formation, and queue discharge. The first equation is the recursive form of Robertson's platoon equation, which models the platoon dispersion. The second equation measure the length of the queue. It depends on the number of arrival vehicles, existing queue, and discharge of queue. For simplicity, the queue discharge rate is assumed to be a constant, which can be achieved from ALPR data The third equation estimates the vehicle's departure flow at downstream ALPR, which is also the estimation of detected flow rate by downstream ALPR.

$$q_2(j) = \sum_{i=1}^{j-t} F(1-F)^{j-i-t}\, q_1(i)$$

$$Q(j) = max\big(Q(j-1) - r_q * g(j) + q_2(j), 0\big)$$

$$q_3(j) = q_2(j) + Q(j) - Q(j-1)$$

Where

$q_1(i)$ = the traffic departure flow at the upstream intersection within i^{th} time interval

$q_2(j)$ = the traffic arrival flow at the downstream intersection within j^{th} time interval

$q_3(j)$ = the estimated traffic departure flow at the downstream intersection within j^{th} time interval

$g(t)$ = the probability density function of the distribution of travel time

$Q(j)$ = the queue length at the downstream intersection within j^{th} time interval

r_q = queue discharge rate

$g(j)$ = the green time within j^{th} time interval.

F = smoothing factor

T = minimal difference between upstream departure time and downstream arrival time

The queue discharge rate is estimated using queue discharge headway:

$$\widehat{R_q} = \frac{3600}{h_q}$$

Where

h_q = queue discharge headway

Then the estimation of the smoothing factor is achieved:

$$\hat{F} = argmin \sum_j [q_3(j) - q_3^d(j)]^2$$

Where

$q_3^d(j)$ = the detected traffic departure flow at the downstream intersection within j^{th} time interval

The equation between the smoothing factor, the travel time factor, and the platoon dispersion factor is reformulated to achieve the calibrated platoon dispersion factor:

$$\hat{\alpha} = \frac{\frac{1}{\hat{F}} - 1}{\hat{\beta}\widehat{T_a}}$$

6.3.2 Case study

A field application has been done for validating the proposed calibration methodology. In the field application, the platoon dispersion calibration methodology is performed on the eastbound traffic of five continuous road segments of Xiandai Ave. For each road segment, the parameters are calibrated for both peak hour traffic and off-peak hour traffic. The geographic layouts of the five road segments are shown in Figure 56. To verify the performance of calibration, the calibrated platoon dispersion model is applied to predict the traffic flow profile of the next half hour. The Mean Absolute Percentage Error (MAPE) is calculated to measure the accuracy of prediction. For comparison, the prediction accuracy of the platoon dispersion model with default is used as a benchmark. The results are shown in Table 11. The key findings are:

- Calibration of parameters of platoon dispersion model boosts the accuracy of traffic flow prediction using platoon dispersion model. In the field application, the average MAPE of prediction using calibrated platoon dispersion model is 10.91% lower than the one using the platoon dispersion model without calibration.

- Both the platoon dispersion factor and the travel time factor varies for different road segments and different periods. This finding highlights the importance of the dynamic calibration procedure for traffic demand prediction. Compared with platoon dispersion factors, travel time factors are more stable. For each road segment, the platoon dispersion factor is higher during the off-peak hour. Many existing studies have similar results. It is a sign that the calibration methodology can capture the impact of traffic conditions on platoon dispersion.

Figure 56 The satellite photo of Xiandai Ave

Table 11 Calibration results

Road Segment	Length (m)	Peak Hour / Off-peak Hour	Min Travel Time (s)	Avg Travel Time (s)	Calibrated			Default		
					α	β	MAPE	α	β	MAPE

Linglong Road - Guanfeng Road	363	Peak	23	34	0.09	0.68	25.46 %	0.50	0.80	37.64 %
		Off-peak	11	26	0.39	0.42	32.35 %	0.50	0.80	40.14 %
Guanfeng Road - Huachi Road	775	Peak	34	57	0.26	0.60	27.10 %	0.50	0.80	38.86 %
		Off-peak	27	54	0.33	0.50	34.57 %	0.50	0.80	41.46 %
Huachi Road - Xinghu Road	723	Peak	39	68	0.23	0.57	30.93 %	0.50	0.80	44.38 %
		Off-peak	35	63	0.26	0.56	26.14 %	0.50	0.80	35.23 %
Xinghu Road - Wansheng Road	578	Peak	28	49	0.11	0.57	28.63 %	0.50	0.80	47.29 %
		Off-peak	30	55	0.24	0.55	30.66 %	0.50	0.80	37.73 %
Wansheng Road - Nanshi Road	640	Peak	29	48	0.17	0.60	31.74 %	0.50	0.80	44.27 %
		Off-peak	34	62	0.22	0.55	32.85 %	0.50	0.80	42.48 %

To summarize, the proposed calibration methodology can perform dynamic parameters calibration of the platoon dispersion model using ALPR data. The field application validates the feasibility of the proposed methodology. The calibrated model has better performance on the prediction of traffic demand, which in term helps to boost the performance of coordinated traffic control strategies.

Chapter 7: Conclusion and future studies

This dissertation developed a traffic diagnosis and management laboratory based on high-coverage ALPR data. The proposed laboratory consists of traffic condition reconstruction, diagnosis-oriented visual analysis, and traffic control and management. This research further contributes to exploring the capability of ALPR data on traffic diagnosis and management. Data-driven methodologies are proposed to utilize ALPR data for traffic condition reconstruction, diagnosis-oriented visual analysis, and traffic control and management. Field applications are performed to validate the feasibility of the proposed laboratory and data-driven methodologies.

In traffic condition reconstruction, this dissertation designed methodologies and algorithms for the reconstruction of traffic flow characteristics, operational performance, and traffic demand. A new data-driven algorithm is developed to estimate the link travel time and control delay of vehicles using ALPR data. The methodology is designed based on observations on enormous ALPR data. It estimates link travel time and control delay by separating the group of vehicles that are not affected by the signal control from the others. Besides, an applicable approach is designed to dynamically estimate the recognition ratio of ALPRs. A better estimation of recognition ratio improves the accuracy of traffic volume estimation and helps to detect sensor malfunctions. Furthermore, a divide-and-conquer algorithm is developed to reconstruct vehicle trajectories. The algorithm is designed to fit large-scale and fight-coverage ALPR systems. Compared with existing methodologies, the algorithm has better performance on large-scale networks.

Diagnosis-oriented visual analysis tools are implemented for bottleneck identification and traffic demand analysis. These diagnosis-oriented visual analysis tools can transform the outputs of traffic condition reconstruction into intuitive insights. To ensure the extensibility of the system, the tools

and modules within the system are deeply modularized. The tools within the system work individually but also complement each other to reveal the overall spatial-temporal pattern in the data.

A traffic control and management decision support system is implemented in the proposed traffic diagnosis and management laboratory. The system supports the decision-making of traffic control and management at two levels: intersection level and arterial level. At the intersection level, the system produces comprehensive reports for intersections, locates time-of-day breakpoints, and optimize phase split and cycle length of signal plans. At the arterial level, the system dynamically calibrates the parameters of the platoon dispersion model, which in term helps to design coordinated signal plans.

Future research may focus on 1) the integration of ALPR data and other traffic sensor data to further boost the accuracy of traffic condition reconstruction. 2) a more interactive visual analysis system for traffic diagnosis, and 3) development of adaptive traffic control strategies based on ALPR data.

References

Abbas, Montasir, Lakshmi Rajasekhar, Asmita Gharat, and John Paul Dunning. 2013. "Microscopic Modeling of Control Delay at Signalized Intersections Based on Bluetooth Data." *Journal of Intelligent Transportation Systems* 17 (2): 110–22.

Acock, Alan C. 2005. "Working with Missing Values." *Journal of Marriage and Family* 67 (4): 1012–28.

Allsop, Richard E. 1971. "SIGSET: A Computer Program for Calculating Traffic Signal Settings." *Traffic Engineering \& Control*.

———. 1976. "SIGCAP: A Computer Program for Assessing the Traffic Capacity of Signal-Controlled Road Junctions." *Traffic Engineering \& Control* 17 (Analytic).

Ananthakrishna, Rohit, Surajit Chaudhuri, and Venkatesh Ganti. 2002. "Eliminating Fuzzy Duplicates in Data Warehouses." In *VLDB'02: Proceedings of the 28th International Conference on Very Large Databases*, 586–97.

Andrienko, Natalia, Gennady Andrienko, and Peter Gatalsky. 2003. "Exploratory Spatio-Temporal Visualization: An Analytical Review." *Journal of Visual Languages & Computing* 14 (6): 503–41.

APPIAH, OBED, EBENEZER QUAYSON, and ERIC OPOKU. 2020. "Ultrasonic Sensor Based Traffic Information Acquisition System; a Cheaper Alternative for ITS Application in Developing Countries." *Scientific African*, e00487.

Armbrust, Michael, Armando Fox, Rean Griffith, Anthony D Joseph, Randy Katz, Andy Konwinski, Gunho Lee, et al. 2010. "A View of Cloud Computing." *Communications of the ACM* 53 (4): 50–58.

Avram, Maricela-Georgiana. 2014. "Advantages and Challenges of Adopting Cloud Computing from an Enterprise Perspective." *Procedia Technology* 12: 529–34.

Bachechi, Chiara, Federica Rollo, and Laura Po. 2020. "Real-Time Data Cleaning in Traffic Sensor Networks." In *2020 IEEE/ACS 17th International Conference on Computer Systems and Applications (AICCSA)*, 1–8.

Balzer, Michael, and Oliver Deussen. 2007. "Level-of-Detail Visualization of Clustered Graph Layouts." In *2007 6th International Asia-Pacific Symposium on Visualization*, 133–40.

Batini, Carlo, Maurizio Lenzerini, and Shamkant B Navathe. 1986. "A Comparative Analysis of Methodologies for Database Schema Integration." *ACM Computing Surveys (CSUR)* 18 (4): 323–64.

Bernardin Jr, Vincent L, Steven Trevino, and Jeffery Short. 2015. "Expanding Truck GPS-Based Passive Origin-Destination Data in Iowa and Tennessee."

Bie, Yiming, Zhiyuan Liu, Dongfang Ma, and Dianhai Wang. 2013. "Calibration of Platoon Dispersion Parameter Considering the Impact of the Number of Lanes." *Journal of Transportation Engineering* 139 (2): 200–207.

Bilenko, Mikhail, and Raymond J Mooney. 2003. “Adaptive Duplicate Detection Using Learnable String Similarity Measures.” In *Proceedings of the Ninth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 39–48.

Bitam, Salim, and Abdelhamid Mellouk. 2012. “Its-Cloud: Cloud Computing for Intelligent Transportation System.” In *2012 IEEE Global Communications Conference (GLOBECOM)*, 2054–59.

Boyandin, Ilya, Enrico Bertini, Peter Bak, and Denis Lalanne. 2011. “Flowstrates: An Approach for Visual Exploration of Temporal Origin-Destination Data.” In *Computer Graphics Forum*, 30:971–80.

Bretherton, R D, K Wood, and G T Bowen. 1998. “SCOOT Version 4.”

Broeck, Jan den, Solveig Argeseanu Cunningham, Roger Eeckels, and Kobus Herbst. 2005. “Data Cleaning: Detecting, Diagnosing, and Editing Data Abnormalities.” *PLoS Med* 2 (10): e267.

Bu, Lei, Feng Wang, Xuesong Zhou, and Chuanzhong Yin. 2016. “Modeling Gate Control Strategy for Traffic Management in Emergency Evacuation.” *International Journal of Transportation* 4 (2): 31–46.

Celikoglu, Hilmi Berk. 2013. “Flow-Based Freeway Travel-Time Estimation: A Comparative Evaluation within Dynamic Path Loading.” *IEEE Transactions on Intelligent Transportation Systems* 14 (2): 772–81.

Cheng, Cheng, Yuchuan Du, Lijun Sun, and Yuxiong Ji. 2016. “Review on Theoretical Delay Estimation Model for Signalized Intersections.” *Transport Reviews* 36 (4): 479–99.

Clark, Stephen D, Susan Grant-Muller, and Haibo Chen. 2002. “Cleaning of Matched License Plate Data.” *Transportation Research Record* 1804 (1): 1–7.

Coifman, Benjamin. 2001. “Improved Velocity Estimation Using Single Loop Detectors.” *Transportation Research Part A: Policy and Practice* 35 (10): 863–80.

Collins, J F, and P Gower. 1974. *Dispersion of Traffic Platoons on A4 in Hounslow*. Traffic Management and Control Division, Traffic Engineering Unit, Transport~….

Colyar, James D, and Nagui M Rouphail. 2003. “Measured Distributions of Control Delay on Signalized Arterials.” *Transportation Research Record* 1852 (1): 1–9.

Curino, Carlo, Hyun Jin Moon, Alin Deutsch, and Carlo Zaniolo. 2010. “Update Rewriting and Integrity Constraint Maintenance in a Schema Evolution Support System: Prism++.” *Proceedings of the VLDB Endowment* 4 (2): 117–28.

———. 2013. “Automating the Database Schema Evolution Process.” *The VLDB Journal* 22 (1): 73–98.

Dailey, Daniel J. 1999. “A Statistical Algorithm for Estimating Speed from Single Loop Volume and Occupancy Measurements.” *Transportation Research Part B: Methodological* 33 (5): 313–22.

"Data Preparation Part in Data Mining Projects." 2003. 2003. https://www.kdnuggets.com/polls/2003/data_preparation.htm.

Dion, Francois, and Hesham Rakha. 2006. "Estimating Dynamic Roadway Travel Times Using Automatic Vehicle Identification Data for Low Sampling Rates." *Transportation Research Part B: Methodological* 40 (9): 745–66.

Doan, AnHai, Pedro M Domingos, and Alon Y Levy. 2000. "Learning Source Description for Data Integration." In *WebDB (Informal Proceedings)*, 81–86.

Du, Shan, Mahmoud Ibrahim, Mohamed Shehata, and Wael Badawy. 2012. "Automatic License Plate Recognition (ALPR): A State-of-the-Art Review." *IEEE Transactions on Circuits and Systems for Video Technology* 23 (2): 311–25.

Eades, Peter, Qing-Wen Feng, and Xuemin Lin. 1996. "Straight-Line Drawing Algorithms for Hierarchical Graphs and Clustered Graphs." In *International Symposium on Graph Drawing*, 113–28.

El-Reedy, T Y, and R Ashworth. 1978. "Platoon Dispersion along a Major Road in Sheffield." *Traffic Engineering \& Control* 19 (4).

Erkan, Ilker, and Hasan Hastemoglu. 2016. "Bluetooth as a Traffic Sensor for Stream Travel Time Estimation under Bogazici Bosporus Conditions in Turkey." *Journal of Modern Transportation* 24 (3): 207–14.

Ersoy, Ozan, Christophe Hurter, Fernando Paulovich, Gabriel Cantareiro, and Alex Telea. 2011. "Skeleton-Based Edge Bundling for Graph Visualization." *IEEE Transactions on Visualization and Computer Graphics* 17 (12): 2364–73.

Farzaneh, Mohamadreza, and Hesham Rakha. 2006. "Procedures for Calibrating TRANSYT Platoon Dispersion Model." *Journal of Transportation Engineering* 132 (7): 548–54.

Fei, Xiang, Yuzhou Zhang, Ke Liu, and Min Guo. 2013. "Bayesian Dynamic Linear Model with Switching for Real-Time Short-Term Freeway Travel Time Prediction with License Plate Recognition Data." *Journal of Transportation Engineering* 139 (11): 1058–67.

Feng, Qingwen. 1997. *Algorithms for Drawing Clustered Graphs*. Citeseer.

Foster, Ian, Yong Zhao, Ioan Raicu, and Shiyong Lu. 2008. "Cloud Computing and Grid Computing 360-Degree Compared." In *2008 Grid Computing Environments Workshop*, 1–10.

Friesen, Marcia R, and Robert D McLeod. 2015. "Bluetooth in Intelligent Transportation Systems: A Survey." *International Journal of Intelligent Transportation Systems Research* 13 (3): 143–53.

Galhardas, Helena, Daniela Florescu, Dennis Shasha, and Eric Simon. 2000. "Declaratively Cleaning Your Data Using AJAX." In *In Journees Bases de Donnees*.

Gartner, Nathan H. 1983. *OPAC: A Demand-Responsive Strategy for Traffic Signal Control*.

Gartner, Nathan H, Farhad J Pooran, and Christina M Andrews. 2001. "Implementation of the

OPAC Adaptive Control Strategy in a Traffic Signal Network." In *ITSC 2001. 2001 IEEE Intelligent Transportation Systems. Proceedings (Cat. No. 01TH8585)*, 195–200.

———. 2002. "Optimized Policies for Adaptive Control Strategy in Real-Time Traffic Adaptive Control Systems: Implementation and Field Testing." *Transportation Research Record* 1811 (1): 148–56.

Gartner, Nathan H, Philip J Tarnoff, and Christine M Andrews. 1991. "Evaluation of Optimized Policies for Adaptive Control Strategy." *Transportation Research Record*, no. 1324.

Ge, Qian, and Daisuke Fukuda. 2016. "Updating Origin--Destination Matrices with Aggregated Data of GPS Traces." *Transportation Research Part C: Emerging Technologies* 69: 291–312.

Gingerich, Kevin, Hanna Maoh, and William Anderson. 2016. "Characterization of International Origin--Destination Truck Movements Across Two Major US--Canadian Border Crossings." *Transportation Research Record* 2547 (1): 1–10.

Grace, Muriel J, and Renfrey B Potts. 1964. "A Theory of the Diffusion of Traffic Platoons." *Operations Research* 12 (2): 255–75.

Greenfeld, Joshua S. 2002. "Matching GPS Observations to Locations on a Digital Map." In *81th Annual Meeting of the Transportation Research Board*, 1:164–73.

Guebert, A A, and G Sparks. 1990. "Timing Plan Sensitivity to Changes in Platoon Dispersion Settings." In *TRAFFIC CONTROL METHODS. PROCEEDINGS OF THE FIFTH NG FOUNDATION CONFERENCE, SANTA BARBARA, CALIFORNI*.

"Guidelines for ANPR Camera Positioning & Set Up." n.d. https://static1.squarespace.com/static/5a62285ff9a61e755f2d9f9b/t/5a96e4e753450a754a7981cb/1519838439799/Camera+Set+up+Guideline.pdf.

Haghani, Ali, Masoud Hamedi, Kaveh Farokhi Sadabadi, Stanley Young, and Philip Tarnoff. 2010. "Data Collection of Freeway Travel Time Ground Truth with Bluetooth Sensors." *Transportation Research Record* 2160 (1): 60–68.

Hall, Fred L, and Bhagwant N Persaud. 1989. "Evaluation of Speed Estimates Made with Single-Detector Data from Freeway Traffic Management Systems." *Transportation Research Record*, no. 1232.

Hall, M, and L G Willumsen. 1980. "SATURN-a Simulation-Assignment Model for the Evaluation of Traffic Management Schemes." *Traffic Engineering \& Control* 21 (4).

Halpin, Terry A, and Henderik Alex Proper. 1995. "Database Schema Transformation and Optimization." In *International Conference on Conceptual Modeling*, 191–203.

Han, Jiang, John W Polak, Javier Barria, and Rajesh Krishnan. 2010. "On the Estimation of Space-Mean-Speed from Inductive Loop Detector Data." *Transportation Planning and Technology* 33 (1): 91–104.

Havre, Susan, Elizabeth Hetzler, Paul Whitney, and Lucy Nowell. 2002. "Themeriver:

Visualizing Thematic Changes in Large Document Collections." *IEEE Transactions on Visualization and Computer Graphics* 8 (1): 9–20.

Head, K Larry, Pitu B Mirchandanai, Steve Shelby, and others. 1998. *The RHODES Prototype: A Description and Some Results*. Transportation Research Board Washington DC, USA.

Head, K Larry, Pitu B Mirchandani, and Dennis Sheppard. 1992. *Hierarchical Framework for Real-Time Traffic Control*.

Hendrickson, Chris, and Sue McNeil. 1984. *Estimation of Origin-Destination Matrices with Constrained Regression*.

Henry, Jean-Jacques, Jean Loup Farges, and Jean Tuffal. 1984. "The PRODYN Real Time Traffic Algorithm." In *Control in Transportation Systems*, 305–10. Elsevier.

Hernández, Mauricio A, and Salvatore J Stolfo. 1998. "Real-World Data Is Dirty: Data Cleansing and the Merge/Purge Problem." *Data Mining and Knowledge Discovery* 2 (1): 9–37.

Holten, Danny, and Jarke J Van Wijk. 2009. "Force-Directed Edge Bundling for Graph Visualization." In *Computer Graphics Forum*, 28:983–90.

Hunt, P B, D I Robertson, R D Bretherton, and M Cr Royle. 1982. "The SCOOT On-Line Traffic Signal Optimisation Technique." *Traffic Engineering \& Control* 23 (4).

Hunt, P B, D I Robertson, R D Bretherton, and R I Winton. 1981. "SCOOT-a Traffic Responsive Method of Coordinating Signals."

Hunter, Timothy, Ryan Herring, Pieter Abbeel, and Alexandre Bayen. 2009. "Path and Travel Time Inference from GPS Probe Vehicle Data." *NIPS Analyzing Networks and Learning with Graphs* 12 (1): 2.

Jain, Neeraj Kumar, R K Saini, and Preeti Mittal. 2019. "A Review on Traffic Monitoring System Techniques." In *Soft Computing: Theories and Applications*, 569–77. Springer.

Jaworski Pawełand Edwards, Tim, Jonathan Moore, and Keith Burnham. 2011. "Cloud Computing Concept for Intelligent Transportation Systems." In *2011 14th International IEEE Conference on Intelligent Transportation Systems (ITSC)*, 391–936.

Jiang, Xiaorui, Chunyi Zheng, Ya Tian, and Ronghua Liang. 2015. "Large-Scale Taxi O/D Visual Analytics for Understanding Metropolitan Human Movement Patterns." *Journal of Visualization* 18 (2): 185–200.

Kazagli, Evanthia, and Haris N Koutsopoulos. 2013. "Estimation of Arterial Travel Time from Automatic Number Plate Recognition Data." *Transportation Research Record* 2391 (1): 22–31.

Klein, Lawrence A, Milton K Mills, and David R P Gibson. 2006. *Traffic Detector Handbook*. US Department of Transportation, Federal Highway Administration, Research~….

Ko, Joonho, Michael Hunter, and Randall Guensler. 2007. "Measuring Control Delay Using Second--by--Second GPS Speed Data." *Transportation Research Board*.

Kucera, Gail. 1992. *Time in Geographic Information Systems*. CRC Press.

Kwak, Sang Kyu, and Jong Hae Kim. 2017. "Statistical Data Preparation: Management of Missing Values and Outliers." *Korean Journal of Anesthesiology* 70 (4): 407.

Lam, Joseph K, Tsuna Sasaki, and Takeo Yamaoka. 1977. *Studies of a Platoon Dispersion Model and Its Practical Implication*. Institute of Systems Science Research.

Li, ZhenJiang, Cheng Chen, and Kai Wang. 2011. "Cloud Computing for Agent-Based Urban Transportation Systems." *IEEE Intelligent Systems* 26 (1): 73–79.

Little, John D C, Mark D Kelson, and Nathan H Gartner. 1981. "MAXBAND: A Versatile Program for Setting Signals on Arteries and Triangular Networks."

Liu, Hongcheng, Ke Han, Vikash Gayah, Terry Friesz, and Tao Yao. 2015. "Data-Driven Linear Decision Rule Approach for Distributionally Robust Optimization of on-Line Signal Control." *Transportation Research Procedia* 7: 536–55.

Long, Jiancheng, Ziyou Gao, and W Y Szeto. 2011. "Discretised Link Travel Time Models Based on Cumulative Flows: Formulations and Properties." *Transportation Research Part B: Methodological* 45 (1): 232–54.

Low, Wai Lup, Mong Li Lee, and Tok Wang Ling. 2001. "A Knowledge-Based Approach for Duplicate Elimination in Data Cleaning." *Information Systems* 26 (8): 585–606.

Lu, Min, Jie Liang, Zuchao Wang, and Xiaoru Yuan. 2016. "Exploring OD Patterns of Interested Region Based on Taxi Trajectories." *Journal of Visualization* 19 (4): 811–21.

Madria, Sanjay, Kalpdrum Passi, and Sourav Bhowmick. 2008. "An XML Schema Integration and Query Mechanism System." *Data & Knowledge Engineering* 65 (2): 266–303.

Manual, Highway Capacity. 2010. "HCM2010." *Transportation Research Board, National Research Council, Washington, DC*, 1207.

Marchal, Fabrice, Jeremy Hackney, and Kay W Axhausen. 2005. "Efficient Map Matching of Large Global Positioning System Data Sets: Tests on Speed-Monitoring Experiment in Zürich." *Transportation Research Record* 1935 (1): 93–100.

Margiotta, Richard A, Neil C Spiller, and Cambridge Systematics. 2009. "Recurring Traffic Bottlenecks: A Primer: Focus on Low-Cost Operational Improvements."

Mazaré, Pierre-Emmanuel, Olli-Pekka Tossavainen, Alexandre Bayen, and D Work. 2012. "Trade-Offs between Inductive Loops and GPS Probe Vehicles for Travel Time Estimation: A Mobile Century Case Study." In *Transportation Research Board 91st Annual Meeting (TRB'12)*. Vol. 349.

Mazloumi, Ehsan, Graham Currie, and Geoffrey Rose. 2010. "Using GPS Data to Gain Insight into Public Transport Travel Time Variability." *Journal of Transportation Engineering* 136 (7): 623–31.

McCoy, Patrick T, Elizabeth A Balderson, Richard T Hsueh, and Abbas K Mohaddes. 1983. "Calibration of TRANSYT Platoon Dispersion Model for Passenger Cars under Low-

Friction Traffic Flow Conditions." *Transportation Research Record* 905: 48–52.

Mikhalkin, Basil, HAROLD J PAYNE, and Leif Isaksen. 1972. "Estimation of Speed from Presence Detectors."

Miller, Alan J. 1963. "A COMPUTER CONTROL SYSTEM FOR TRAFFIC NETWORKS."

Mirchandani, Pitu, and Larry Head. 2001. "A Real-Time Traffic Signal Control System: Architecture, Algorithms, and Analysis." *Transportation Research Part C: Emerging Technologies* 9 (6): 415–32.

Mo, Baichuan, Ruimin Li, and Xianyuan Zhan. 2017. "Speed Profile Estimation Using License Plate Recognition Data." *Transportation Research Part C: Emerging Technologies* 82: 358–78.

Monge, Alvaro E. 2000. "Matching Algorithms within a Duplicate Detection System." *IEEE Data Eng. Bull.* 23 (4): 14–20.

Mori, Usue, Alexander Mendiburu, Maite Álvarez, and Jose A Lozano. 2015. "A Review of Travel Time Estimation and Forecasting for Advanced Traveller Information Systems." *Transportmetrica A: Transport Science* 11 (2): 119–57.

Nallaperuma, Dinithi, Rashmika Nawaratne, Tharindu Bandaragoda, Achini Adikari, Su Nguyen, Thimal Kempitiya, Daswin De Silva, Damminda Alahakoon, and Dakshan Pothuhera. 2019. "Online Incremental Machine Learning Platform for Big Data-Driven Smart Traffic Management." *IEEE Transactions on Intelligent Transportation Systems* 20 (12): 4679–90.

Nam, Do H, and Donald R Drew. 1996. "Traffic Dynamics: Method for Estimating Freeway Travel Times in Real Time from Flow Measurements." *Journal of Transportation Engineering* 122 (3): 185–91.

Nellore, Kapileswar, and Gerhard P Hancke. 2016. "A Survey on Urban Traffic Management System Using Wireless Sensor Networks." *Sensors* 16 (2): 157.

Novotny, Matej. 2004. "Visually Effective Information Visualization of Large Data." In *Proceedings of Central European Seminar on Computer Graphics (CESCG)*.

Oh, Jun-Seok, R Jayakrishnan, and Will Recker. 2002. "Section Travel Time Estimation from Point Detection Data."

Oh, Seri, Stephen G Ritchie, and Cheol Oh. 2002. "Real-Time Traffic Measurement from Single Loop Inductive Signatures." *Transportation Research Record* 1804 (1): 98–106.

Pacey, G M. 1956. "The Progress of a Bunch of Vehicles Released from a Traffic Signal." *Road Research Laboratory Note RN/2665/GMP* 1956 (1956): 36–50.

Patidar, Shyam, Dheeraj Rane, and Pritesh Jain. 2012. "A Survey Paper on Cloud Computing." In *2012 Second International Conference on Advanced Computing & Communication Technologies*, 394–98.

Paul, Bhaskar, Mayank Ramteke, Bhargab Maitra, and Sudeshna Mitra. 2018. "New Approach

for Calibrating Robertson's Platoon Dispersion Model." *Journal of Transportation Engineering, Part A: Systems* 144 (5): 4018014.

Phan, Doantam, Ling Xiao, Ron Yeh, and Pat Hanrahan. 2005. "Flow Map Layout." In *IEEE Symposium on Information Visualization, 2005. INFOVIS 2005.*, 219–24.

Phipps, Cassandra, and Karen C Davis. 2002. "Automating Data Warehouse Conceptual Schema Design and Evaluation." In *DMDW*, 2:23–32.

Porter, J David, David S Kim, Mario E Magaña, Panupat Poocharoen, and Carlos Antar Gutierrez Arriaga. 2013. "Antenna Characterization for Bluetooth-Based Travel Time Data Collection." *Journal of Intelligent Transportation Systems* 17 (2): 142–51.

Pu, Jiansu, Siyuan Liu, Ye Ding, Huamin Qu, and Lionel Ni. 2013. "T-Watcher: A New Visual Analytic System for Effective Traffic Surveillance." In *2013 IEEE 14th International Conference on Mobile Data Management*, 1:127–36.

Quddus, Mohammed A, Robert B Noland, and Washington Y Ochieng. 2006. "A High Accuracy Fuzzy Logic Based Map Matching Algorithm for Road Transport." *Journal of Intelligent Transportation Systems* 10 (3): 103–15.

Quiroga, Cesar A. 1997. "An Integrated GPS-GIS Methodology for Performing Travel Time Studies."

Quiroga, Cesar A, and Darcy Bullock. 1998a. "Determination of Sample Sizes for Travel Time Studies." *ITE Journal* 68 (8): 92–98.

———. 1998b. "Travel Time Studies with Global Positioning and Geographic Information Systems: An Integrated Methodology." *Transportation Research Part C: Emerging Technologies* 6 (1–2): 101–27.

———. 1999. "Measuring Control Delay at Signalized Intersections." *Journal of Transportation Engineering* 125 (4): 271–80.

Rahm, Erhard, and Hong Hai Do. 2000. "Data Cleaning: Problems and Current Approaches." *IEEE Data Eng. Bull.* 23 (4): 3–13.

Rakha, Hesham, Michel Van Aerde, Kyoungho Ahn, and Antonio Trani. 2000. "Requirements for Evaluating Traffic Signal Control Impacts on Energy and Emissions Based on Instantaneous Speed and Acceleration Measurements." *Transportation Research Record* 1738 (1): 56–67.

Rao, Wenming, Yao-Jan Wu, Jingxin Xia, Jishun Ou, and Robert Kluger. 2018. "Origin-Destination Pattern Estimation Based on Trajectory Reconstruction Using Automatic License Plate Recognition Data." *Transportation Research Part C: Emerging Technologies* 95: 29–46.

Regalado, Antonio. 2011. "Who Coined 'Cloud Computing.'" *Technology Review* 31.

Roberts, David J, and Meghann Casanova. 2012. "Automated License Plate Recognition Systems: Policy and Operational Guidance for Law Enforcement."

Robertson, Dennis I. 1969. “TRANSYT: A Traffic Network Study Tool.”

Robertson, Dennis I, and R David Bretherton. 1991. “Optimizing Networks of Traffic Signals in Real Time-the SCOOT Method.” *IEEE Transactions on Vehicular Technology* 40 (1): 11–15.

Robinson, Steve, and John W Polak. 2006. “Inductive Loop Detector Data Cleaning Treatments and Their Effect on Performance of Urban Link Travel Time Models.”

Rondon, Abraham. 2014. “The Impact of Weather Conditions on Urban Travel Speed Using ANPR Observations.”

Saiyed, S, and J A Stewart. 2004. “An Assessment of Pre-Timed, Actuated and Adaptive Signal Control Strategies for Unsaturated and Saturated Arterial Network.” *Transportation Research Board*.

Sanaullah, Irum, Mohammed Quddus, and Marcus Enoch. 2016. “Developing Travel Time Estimation Methods Using Sparse GPS Data.” *Journal of Intelligent Transportation Systems* 20 (6): 532–44.

Seddon, P A. 1972. “Another Look at Platoon Dispersion: 3. The Recurrence Relationship.” *Traffic Engineering and Control* 13 (10): 442–44.

Selassie, David, Brandon Heller, and Jeffrey Heer. 2011. “Divided Edge Bundling for Directional Network Data.” *IEEE Transactions on Visualization and Computer Graphics* 17 (12): 2354–63.

Sen, Suvrajeet, and K Larry Head. 1997. “Controlled Optimization of Phases at an Intersection.” *Transportation Science* 31 (1): 5–17.

Sharifi, Elham, Masoud Hamedi, Ali Haghani, and Hadi Sadrsadat. 2011. “Analysis of Vehicle Detection Rate for Bluetooth Traffic Sensors: A Case Study in Maryland and Delaware.” In *18th World Congress on on Intelligent Transport Systems*.

Shatnawi, Ibrahem, Ping Yi, and Ibrahim Khliefat. 2018. “Automated Intersection Delay Estimation Using the Input--Output Principle and Turning Movement Data.” *International Journal of Transportation Science and Technology* 7 (2): 137–50.

Stamatiadis, Chronis, and Nathan H Gartner. 1996. “MULTIBAND-96: A Program for Variable-Bandwidth Progression Optimization of Multiarterial Traffic Networks.” *Transportation Research Record* 1554 (1): 9–17.

Stevanovic, Aleksandar. 2010. *Adaptive Traffic Control Systems: Domestic and Foreign State of Practice*.

Stoffers, Karl E. 1967. “Scheduling of Traffic Lights-a New Approach.” *Transportation Research/UK/* 2 (3).

Sun, Carlos, and Stephen G Ritchie. 1999. “Individual Vehicle Speed Estimation Using Single Loop Inductive Waveforms.” *Journal of Transportation Engineering* 125 (6): 531–38.

Sun, Guodao, Yang Liu, Wenbin Wu, Ronghua Liang, and Huamin Qu. 2014. “Embedding

Temporal Display into Maps for Occlusion-Free Visualization of Spatio-Temporal Data." In *2014 IEEE Pacific Visualization Symposium*, 185–92.

Tarnoff, Philip J, and Peter S Parsonson. 1981. "Selecting Traffic Signal Control at Individual Intersections." *NCHRP Report*, no. 233.

Taylor, George, Chris Brunsdon, Jing Li, Andrew Olden, Dörte Steup, and Marylin Winter. 2006. "GPS Accuracy Estimation Using Map Matching Techniques: Applied to Vehicle Positioning and Odometer Calibration." *Computers, Environment and Urban Systems* 30 (6): 757–72.

"The State of Cloud-Driven Transformation Keys to Accelerating and Capitalizing on Cloud Adoption." 2020. https://hbr.org/resources/pdfs/comm/splunk/thestateofclouddriven.pdf.

Trivedi, Prashant, Kavita Deshmukh, and Manish Shrivastava. 2012. "Cloud Computing for Intelligent Transportation System." *International Journal of Soft Computing and Engineering (IJSCE)* 2 (3): 2231–2307.

Turner, Shawn M, William L Eisele, Robert J Benz, and Douglas J Holdener. 1998. "Travel Time Data Collection Handbook."

Vanajakshi, Lelitha D, Billy M Williams, and Laurence R Rilett. 2009. "Improved Flow-Based Travel Time Estimation Method from Point Detector Data for Freeways." *Journal of Transportation Engineering* 135 (1): 26–36.

Vasiliev, Irina. 1997. "Mapping Time." *Cartographica: The International Journal for Geographic Information and Geovisualization* 34 (2): 1–51.

Wang, Qin, Min Te Sun, and Kazuya Sakai. 2017. "Trajectory Data Cleansing Using HMM." *Proceedings of the International Conference on Parallel Processing Workshops*, 8–15. https://doi.org/10.1109/ICPPW.2017.15.

Wang, Youan, Xumei Chen, Lei Yu, and Yi Qi. 2017. "Calibration of the Platoon Dispersion Model by Considering the Impact of the Percentage of Buses at Signalized Intersections." *Transportation Research Record* 2647 (1): 93–99.

Wardrop, John Glen. 1952. "Road Paper. Some Theoretical Aspects of Road Traffic Research." *Proceedings of the Institution of Civil Engineers* 1 (3): 325–62.

Wargelin, Laurie, Peter Stopher, Jason Minser, Kevin Tierney, Mindy Rhindress, Sharon O'Connor, and others. 2012. "GPS-Based Household Interview Survey for the Cincinnati, Ohio Region."

Yagar, Sam. 1974. "Capacity of a Signalized Road Junction: Critique and Extensions." *Transportation Research* 8 (2): 137–47.

Yan, Yuyu, Yubo Tao, Jin Xu, Shuilin Ren, and Hai Lin. 2018. "Visual Analytics of Bike-Sharing Data Based on Tensor Factorization." *Journal of Visualization* 21 (3): 495–509.

Yang, Hai, Tsuna Sasaki, Yasunori Iida, and Yasuo Asakura. 1992. "Estimation of Origin-Destination Matrices from Link Traffic Counts on Congested Networks." *Transportation*

Research Part B: Methodological 26 (6): 417–34.

Yu, Haiyang, Shuai Yang, Zhihai Wu, and Xiaolei Ma. 2018. "Vehicle Trajectory Reconstruction from Automatic License Plate Reader Data." *International Journal of Distributed Sensor Networks* 14 (2). https://doi.org/10.1177/1550147718755637.

Yu, Lei. 1999. "Real-Time Calibration of Platoon Dispersion Model to Optimize the Coordinated Traffic Signal Timing in ATMS Networks."

———. 2000. "Calibration of Platoon Dispersion Parameters on the Basis of Link Travel Time Statistics." *Transportation Research Record* 1727 (1): 89–94.

Zanjani, Akbar Bakhshi, Abdul R Pinjari, Mohammadreza Kamali, Aayush Thakur, Jeffrey Short, Vidya Mysore, and S Frank Tabatabaee. 2015. "Estimation of Statewide Origin--Destination Truck Flows from Large Streams of GPS Data: Application for Florida Statewide Model." *Transportation Research Record* 2494 (1): 87–96.

Zeng, Wei, Chi-Wing Fu, Stefan Müller Arisona, and Huamin Qu. 2013. "Visualizing Interchange Patterns in Massive Movement Data." In *Computer Graphics Forum*, 32:271–80.

Zhang, Chen, Yugeng Xi, Dewei Li, and Yunwen Xu. 2018. "Data-Driven Model for Traffic Signal Control." In *2018 37th Chinese Control Conference (CCC)*, 7880–85.

Zhao, Li, Laurence R Rilett, and Ernest Tufuor. 2017. "Calibrating the Robertson's Platoon Dispersion Model on a Coordinated Corridor with Advance Warning Flashers." *Transportation Research Record* 2623 (1): 10–18.

Zhi-Peng, L I, Y U Hong, L I U Yun-Cai, and L I U Fu-Qiang. 2008. "An Improved Adaptive Exponential Smoothing Model for Short-Term Travel Time Forecasting of Urban Arterial Street." *Acta Automatica Sinica* 34 (11): 1404–9.

Zhou, Hong, Panpan Xu, Xiaoru Yuan, and Huamin Qu. 2013. "Edge Bundling in Information Visualization." *Tsinghua Science and Technology* 18 (2): 145–56.

APPENDICES

APPENDIX A: ALPR sample data

RECORD_ID,ALPR_NAME,LANE_ID,DIR_INFO,PLATE_NUM,PLATE_COLOR,TIME_STAMP,SPEED

147200,fb01ea083093fc03570df1c52f78256bce5ce10c,3,4,ec00b6793af7e16c7e9c46e7b29b687f2e8b6648,3,2017-04-01 06:25:53.630000,13

147201,17c904ee8de014318972d4756f1226f164a31cac,4,2,38724fa5e4b96e25e529e2b1138ee2c5e3fdceed,1,2017-04-01 06:25:50.333000,24

147202,f3df979c80e25ee6dad31b93fe18931dac8567b5,2,4,b020228218c3cda8986c6cafd03f9fe4dfd786f2,1,2017-04-01 06:25:50.015000,79

147203,c3764ed114ee370306a3a9faf2e8670b328b6ca5,3,3,1f97bd9a2e7aa495d88dab6fc3e43f47fd5b9d34,1,2017-04-01 06:27:21.020000,74

147204,c2bbb351cb76848f89f64ba54cceb65a9b81d828,2,2,761ac13fae6c4fe52c088b5d84c1b613bffd0b9a,1,2017-04-01 06:25:51.356000,69

147205,9a6289ae4e26a747a0a840c84875d3bc70aa36a7,4,3,2befa1c89c882dcf4eda9032b779e09d1a5ca7cb,1,2017-04-01 06:25:51.551000,75

147206,9a6289ae4e26a747a0a840c84875d3bc70aa36a7,3,3,736055f86cdcbbb414baa8f6eca9f23b2a12a475,1,2017-04-01 06:25:53.768000,72

147207,b5e3f3e0828805fb16f921878a2369322da56cb9,1,2,d8e6aa23021ca0c5552bee5e4f88ad58caccd673,1,2017-04-01 06:25:52.727000,0

147208,c3764ed114ee370306a3a9faf2e8670b328b6ca5,5,3,689bc9006ed60eeed1a60e533ff443193233e531,3,2017-04-01 06:25:54.383000,51

147209,c3764ed114ee370306a3a9faf2e8670b328b6ca5,2,3,0a0daaa793848e60239c2d152b8894a1b155ea72,1,2017-04-01 06:25:53.590000,82

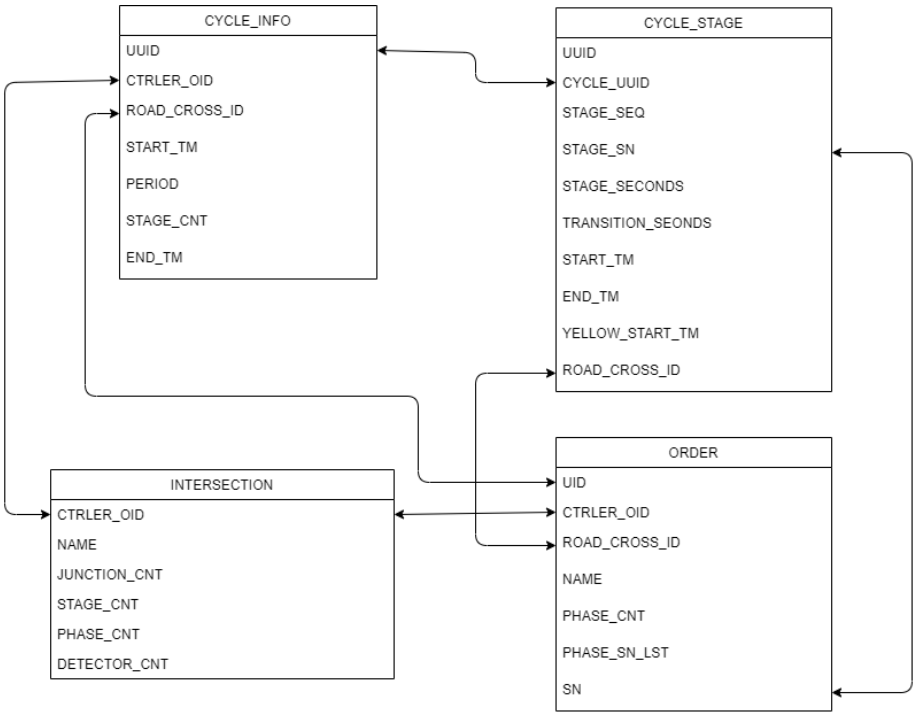
147210,aae70920bb737cb95c8fedcd5e04bb8b3f333c1e,2,4,8b36a555f01f472253f4b08430d77d5148fc3aa5,1,2017-04-01 06:26:31.206000,14

147211,dcc703573667332b4b1e64e75728a6fa631ac94f,3,2,b1bf2765f65566e03c41de49b49355676a145171,2,2017-04-01 06:26:31.921000,21

147212,18a6b809317a7c81dc2b9b5e8c8a37d8f58e00d5,1,1,4b6b4d3e133af9d001f45ceb4f7aeb44c44159a4,1,2017-04-01 06:26:31.025000,10

147213,aae70920bb737cb95c8fedcd5e04bb8b3f333c1e,2,4,e811795bf604ba06228fcc828bb7ee26aafd5e29,1,2017-04-01 06:26:33.000000,14

APPENDIX B: Signal control sample data



INTERSECTION,CTRLER_OID,NAME,JUNCTION_CNT,STAGE_CNT,PHASE_CNT,DET
ECTOR_CNT

0,011803,d7acc0555ba0feed6c233ef025f6e089c733401c,1,9,9,8

1,013F02,db629382e6771879f02cf50e4f2230355e51946f,1,9,12,0

2,011A02,4559c1df0465fdba41816a3b109537cdc2b19336,1,9,16,17

3,011C03,42487ba9e18392a6098136502b405d68873b8ca3,1,9,10,10

4,011E01,2a03b897a343c9783630482817e2865f1e4a2b7e,1,9,14,20

5,011E02,a6518275c934349566df2a7fef1ab90425919d8c,1,9,16,11

6,011D03,951f0ef8ef985465876e3061d2d1546535994923,1,9,10,14

7,011C01,00f3e4dbd2d874213f026a2d7d0e8a7e49188c7f,1,9,16,16

8,011C04,15ca78a38cfecd1bef6fa41778f7b86cbf2055fe,1,9,13,16

9,011B04,c572c9a23b8b3f62751573c68516d2417c99a777,1,9,11,17

10,011E03,c7e7e3545baed033e85dc43c3f58a4a769d2f553,1,9,7,9

ORDER

,UID,CTRLER_OID,ROAD_CROSS_ID,NAME,PHASE_CNT,PHASE_SN_LST,SN

0,03924CE493E648B0BE446FF2B2F0230B,011901,108330,东西左转,2,0104,2

1,6F85A4004F61402CA3A21D27458EEBE2,011901,108330,全红,0,,3

2,303524C9705040B183BE240922B9EE64,011901,108330,全红,0,,4

3,A132A4A49B2D4F1B835086A2B25D09F7,011901,108330,全红,0,,5

4,F06AAD2807284FD78DA93B532BF247A0,011901,108330,南北放行,4,00030709,6

5,ABF8FBBF505E4DB5BFF279B10ED31E5D,011901,108330,东放行,3,040506,7

6,4BEA4A17C62441C2BF2D70BA97FA5030,011901,108330,西放行,3,010208,8

7,1534CCB39BEB4E13B7A2D5BC3DADF9C9,011902,108530,s0,0,,0

8,05E551F98D6B42C29FFF20C3B862B1D2,011902,108530,全红,0,,1

9,9AF969939A5F4D81A421C86F1C08D1C1,011902,108530,全红,0,,2

10,591263187D174BC4B299625BC227E087,011902,108530,全红,0,,3

11,822B8F155D6640CCB98815C19233D54A,011902,108530,全红,0,,4

12,03777A8A4FE643C9BF42B0C5087AD0CE,011902,108530,东西放行,6,0304090A0C0E,5

13,015D202807F3471AB71B7CE922DAABFC,011902,108530,南北放行,6,000106070D0F,6

14,3977EBE7639F4AB6A5D631E56DC268E5,011902,108530,东放行,3,090A0C,7

15,89D9ECCD7E864F958AEA6AEDC7B07005,011902,108530,西放行,3,03040E,8

16,B02D92C5C50D433B8EF56DC5CA2024A4,011903,108560,s0,0,,0

17,7F225EC6726D4778B06C7315271395DE,011903,108560,全红,0,,1

18,E61C5865242647EDB16F5AB873F2EB4A,011903,108560,全红,0,,2

19,2F09C009A0F640DD8E9D4059F125C073,011903,108560,全红,0,,3

20,E39025BB39B64D0FB991ACD7B5D75111,011903,108560,全红,0,,4

21,CC84C1B438EC4CD58C90A68A16C4496E,011903,108560,东西放行,6,0304090A0C0E,5

22,B51D914B81844762A6AD4B579AE7D8FB,011903,108560,南北放行,6,000106070D0F,6

CYCLE_INFO

,UUID,CTRLER_OID,ROAD_CROSS_ID,START_TM,PERIOD,STAGE_CNT,END_TM

0,731CAC8EF177445F8DFA3E2E97C1A167,010401,108660,02-4 月 -17 09.16.21.000000 下午,102,4,02-4 月 -17 09.18.03.000000 下午

1,08AF62572A8B49AEBE1C3B6F96DA9F35,011F03,110631,02-4 月 -17 09.16.58.000000 下午,65,2,02-4 月 -17 09.18.03.000000 下午

2,0C578C45CBA0403C9C5B14795A671044,011404,103781,02-4 月 -17 09.16.20.000000 下午,103,3,02-4 月 -17 09.18.03.000000 下午

3,F6FECE22ED404E56A65CA9618F1C552B,012201,103691,02-4 月 -17 09.16.43.000000 下午,80,5,02-4 月 -17 09.18.03.000000 下午

4,A6CFEE6D7AF14C9BAF295FD66FCFB5E9,012002,110311,02-4 月 -17 09.16.53.000000 下午,70,2,02-4 月 -17 09.18.03.000000 下午

5,E68D9F0DDC1245C4B67EA59E19A1F4A2,012801,100430,02-4 月 -17 09.16.32.000000 下午,93,4,02-4 月 -17 09.18.05.000000 下午

6,EEE1F6AF716041D280567A41CA43C49D,010402,107500,02-4 月 -17 09.17.13.000000 下午,52,2,02-4 月 -17 09.18.05.000000 下午

7,A3279CBC9E6D4D65A583A3BB98287327,010B02,102440,02-4 月 -17 09.17.18.000000 下午,92,3,02-4 月 -17 09.18.50.000000 下午

8,BBD5487F58C946A59FFDE2C250F40134,012901,101600,02-4 月 -17 09.17.08.000000 下午,103,4,02-4 月 -17 09.18.51.000000 下午

9,C404FF60225F494C968EA2513FEDBEEB,012A01,101080,02-4 月 -17 09.17.16.000000 下午,95,4,02-4 月 -17 09.18.51.000000 下午

10,82412B097A5043AF8EBBCA5D9810AB8B,010402,107500,02-4 月 -17 09.18.05.000000 下午,48,2,02-4 月 -17 09.18.53.000000 下午

11,17975D6752E14479AD3AEEDD8758BB07,010E01,100420,02-4 月 -17 09.17.37.000000 下午,76,3,02-4 月 -17 09.18.53.000000 下午

12,B74A59E4C16C486694EA7F584C8A2EA1,010101,101240,02-4 月 -17 09.17.18.000000 下午,95,3,02-4 月 -17 09.18.53.000000 下午

13,8B60667AD76D41DCADA8612952A238E0,012702,103101,02-4 月 -17 09.17.21.000000 下午,92,4,02-4 月 -17 09.18.53.000000 下午

CYCLE_STAGE

,UUID,CYCLE_UUID,STAGE_SEQ,STAGE_SN,STAGE_SECONDS,TRANSITION_SECONDS,START_TM,END_TM,YELLOW_START_TM,ROAD_CROSS_ID

0,4C456D73BB3F4A25E0537408640AAB69,A132C2660C8B42419D46B4DE6E15AF31,3,1,12,9,03-4 月 -17 11.59.59.000000 下午,04-4 月 -17 12.00.20.000000 上午,04-4 月 -17 12.00.14.000000 上午,103220

1,4C456D73BB444A25E0537408640AAB69,B635B3BD62BA4340ACED16B6D977686F,2,5,17,4,03-4 月 -17 11.59.59.000000 下午,04-4 月 -17 12.00.20.000000 上午,04-4 月 -17 12.00.19.000000 上午,101560

2,4C456D73BB464A25E0537408640AAB69,999E921875E34350AEC4C23DFE93E4E6,2,6,15,7,03-4 月 -17 11.59.59.000000 下午,04-4 月 -17 12.00.21.000000 上午,04-4 月 -17 12.00.17.000000 上午,110030

3,4C456D73D8334A25E0537408640AAB69,18FC6CBD3FAD4634A96A11A14D18BF00,1,6,7,7,03-4 月 -17 11.59.59.000000 下午,04-4 月 -17 12.00.13.000000 上午,04-4 月 -17 12.00.09.000000 上午,105710

4,4C456D73EB694A25E0537408640AAB69,628D3B6751824DB8B7B7A47687980B9D,1,6,16,8,03-4 月 -17 11.59.59.000000 下午,04-4 月 -17 12.00.23.000000 上午,04-4 月 -17 12.00.18.000000 上午,108500

5,4C456D73A4B04A25E0537408640AAB69,BEEC05EB9E214835A8AAFB8EA6D3072B,2,6,6,7,03-4 月 -17 11.59.59.000000 下午,04-4 月 -17 12.00.12.000000 上午,04-4 月 -17 12.00.08.000000 上午,110020

6,4C4625B9B5015006E0537408640AB3EE,AF69B9F61EE64897BC1E9DEF5F96B66A,1,4,9,7,03-4 月 -17 11.59.58.000000 下午,04-4 月 -17 12.00.14.000000 上午,04-4 月 -17 12.00.10.000000 上午,110693

7,4C4625B9D02A5006E0537408640AB3EE,6AEE327967CA43A2B86958CA2FD3CCA7,1,6,60,7,03-4 月 -17 11.59.58.000000 下午,04-4 月 -17 12.01.05.000000 上午,04-4 月 -17 12.01.01.000000 上午,101350

8,4C4625B9DA1E5006E0537408640AB3EE,28754CE007D8402FA0E8A621AA4A3231,1,5,17,8,03-4 月 -17 11.59.58.000000 下午,04-4 月 -17 12.00.23.000000 上午,04-4 月 -17 12.00.18.000000 上午,101630

9,4C456D73EB5E4A25E0537408640AAB69,ADDBF1A259C8445DA7EA0DD804EDFAC1,2,6,35,8,03-4 月 -17 11.59.58.000000 下午,04-4 月 -17 12.00.41.000000 上午,04-4 月 -17 12.00.36.000000 上午,101380

CURRICULUM VITAE

Weijie Tan

Place of birth: Shanghai, China

Education

B.S., Shanghai University of Finance and Economics, P.R.C, June 2016

Major: Finance

Dissertation Title: Development of a cloud-based traffic diagnosis and management laboratory based on high-coverage ALPR

Publications

Yuan, Y., Yu, J., Han, D, Tan, W, Jin, Z. Strategic planning of evacuation in urban network with balanced signal control and cross elimination strategies. Journal of Transportation Engineering, Part A: Systems, 2018, 145(1), 04018080.

Han D., Yu J.*, Beimborn E., Jin Z., Tan W. Elements of successful universal student transit pass programs from planning to implementation: a benchmark study. The Transportation Research Board (TRB) 98th Annual Meeting, Washington, D.C., United States, 2019.01.13-01.17