ESTIMATE TRAVEL TIME RELIABILITY AND EMISSIONS FOR ACTIVE TRAFFIC AND DEMAND MANAGEMENT

by

Hao Lei

A dissertation submitted to the faculty of
The University of Utah
in partial fulfillment of the requirements for the degree of

Doctor of Philosophy

Department of Civil and Environmental Engineering

The University of Utah

December 2013

The University of Utah Graduate School

STATEMENT OF DISSERTATION APPROVAL

The dissertation of **Hao Lei**

has been approved by the following supervisory committee members:

Xuesong Zhou	, Chair	**6/13/2013** Date Approved
Peter Martin	, Member	Date Approved
Richard Porter	, Member	**6/13/2013** Date Approved
Suresh Venkatasubramanian	, Member	**6/13/2013** Date Approved
George List	, Member	Date Approved

and by **Chris Pantelides** , Chair/Dean of

the Department/College/School of **Civil and Environmental Engineering**

and by David B. Kieda, Dean of The Graduate School.

ABSTRACT

The Active Traffic and Demand Management (ATDM) initiative aims to integrate various management strategies and control measures so as to achieve the mobility, environment and sustainability goals. To support the active monitoring and management of real-world complex traffic conditions, the first objective of this dissertation is to develop a travel time reliability estimation and prediction methodology that can provide informed decisions for the management and operation agencies and travelers. A systematic modeling framework was developed to consider a corridor with multiple bottlenecks, and a series of close-form formulas was derived to quantify the travel time distribution under both stochastic demand and capacity, with possible on-ramp and off-ramp flow changes.

Traffic state estimation techniques are often used to guide operational management decisions, and accurate traffic estimates are critically needed in ATDM applications designed for reducing instability, volatility and emissions in the transportation system. By capturing the essential forward and backward wave propagation characteristics under possible random measurement errors, this dissertation proposes a unified representation with a simple but theoretically sound explanation for traffic observations under free-flow, congested and dynamic transient conditions. This study also presents a linear programming model to quantify the value of traffic measurements, in a heterogeneous data environment with fixed sensors, Bluetooth readers and GPS sensors.

It is important to design comprehensive traffic control measures that can systematically address deteriorating congestion and environmental issues. To better evaluate and assess the mobility and environmental benefits of the transportation improvement plans, this dissertation also discusses a cross-resolution modeling framework for integrating a microscopic emission model with the existing mesoscopic traffic simulation model. A simplified car-following model-based vehicle trajectory construction method is used to generate the high-resolution vehicle trajectory profiles and resulting emission output.

In addition, this dissertation discusses a number of important issues for a cloud computing-based software system implementation. A prototype of a reliability-based traveler information provision and dissemination system is developed to offer a rich set of travel reliability information for the general public and traffic management and planning organizations.

To my wife

LIST OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

ACKNOWLEDGEMENTS

I would like to express my sincere gratitude to my advisor, Dr. Xuesong Zhou, whom I respect and admire greatly. His intelligent guidance, continuous support and encouragement have been of the most importance to my research and the completion of this dissertation. I would also like to express my deep appreciation to Dr. Peter Martin and Dr. R.J. Porter for their support and invaluable insight. Special thanks are given to Dr. Suresh Venkatasubramanian and Dr. George List for their participation and constructive comments on my research.

I feel grateful as well to my dear colleagues and friends, Dr. Mingxin Li, Dr. Tao Xing, Jeffrey Taylor and others, who made my stay at the University of Utah a very pleasant and memorable one.

I would like to express my heartfelt gratitude to my parents for their endless support and sacrifices.

Finally this work is dedicated to my beloved wife, Ying, who gave me love, encouragement, patience and confidence.

CHAPTER 1

INTRODUCTION

1.1 Background

Many metropolitan areas are facing rising traffic congestion problems due to the continued growth of travel demand on congested urban freeway corridors. Meanwhile, the limited funding opportunities for agencies make it difficult to support sufficient road expansion and other capacity improvement projects. As a result, a number of Intelligent Transportation Systems (ITS) technologies, such as Advanced Transportation Management Systems (ATMS) and Advanced Traveler Information Systems (ATIS), are widely adopted to improve the operational performances of the existing facilities and customer satisfaction, and help reduce the number of accidents and environmental impacts.

To monitor and administer the operations of the facilities, a large number of traffic surveillance equipment is deployed by public transportation agencies on freeways and arterials to measure time-varying traffic network flow patterns. The traffic measurement data from this equipment are collected and extracted by ATMS and later utilized by Traffic Management Centers (TMC) to accordingly generate coordinated control strategies and further optimize the traffic operations. In order to maximize the and select sensor information gains from the limited resources on sensor installation, it is vital to

identify and select sensor investments by type and location. In addition, it is necessary to reduce the volume of the measurement data for more efficient data extraction and processing while still providing accurate enough traffic flow variability information. Thus, how to select the sensor spacing and reporting rate for travel time reliability monitoring creates a big challenge for the transportation agencies.

Using synthesized traffic state information from ATMS, ATIS provides pretrip and en-route trip planning information, route guidance and other advisory functions for travelers. Pretrip information enables travelers to select the route, departure time and mode that best fit their travel purposes, while en-route traveler information provides the travelers with traffic information, roadway conditions, transit information, as well as personalized route guidance information. Effective traveler information systems should provide timely, accurate and reliable predictive information to travelers.

Recently, the Active Traffic and Demand Management (ATDM) initiative aims to integrate Active Traffic Management (ATM), Travel Demand Management (TDM) and other ITS strategies for dynamic management of transportation systems. The Active Management concept of ATDM implies a proactive approach for dynamically managing and controlling demand and available capacity of transportation facilities. For example, ATDM uses real-time and historical traffic information to predict the traffic condition so that some specific situations, such as congestion, and the resultant impacts can be prevented. If the impacts have already occurred, ATDM dynamically reacts to these impacts through demand control strategies; that is, when congestion is detected, congestion warning and alternative route information are provided to divert the traffic demand away from the impacted corridors or areas to mitigate the congestion and the

negative environmental impacts. The successful implementation of ATDM uses a performance and reliability-driven philosophy to consider the full range of strategies. The ATDM performance outcomes extend from the traditional highway Level of Service (LOS) and other operational performance indices to other measures addressing mobility and environmental considerations.

1.2 Overview of Research Topics

Aiming at improving the end-to-end corridor travel time reliability estimation and prediction, and incorporating environmental factors into ATDM applications, this dissertation will discuss the following practically important and theoretically challenging questions: 1) travel time reliability estimation and prediction and 2) traffic state estimation and prediction. In addition, two practical applications, to integrate the microscopic emission models into the mesoscopic traffic assignment and simulation model and for reliability-based traveler information provision and dissemination, are also discussed.

1.2.1 Estimating Traffic Reliability Along Freeway Corridors

Travel time has long been regarded as one of the most important performance measures in transportation systems. Recently, significant attention has been devoted to evaluating and quantifying the reliability of travel times due to the influence of travel time variability on route, departure time, and mode choices. Operating agencies have also increased their efforts to monitor and improve the reliability of their systems through probe-based data collection, integrated corridor management and advanced traveler

information systems. In particular, vigorous data collection efforts have improved data quality and uncovered the root sources of travel time reliability. Corridor management strategies have been designed to balance the performance of freeway and arterial networks in response to congestion and nonrecurring events. In addition, advanced traveler information systems provide reliability-related information that enables travelers to achieve on-time arrival goals. There also appears to be a growing trend toward incorporating travel time reliability into traffic network design, analysis and management models.

Recently, significant attention has been devoted to evaluating and quantifying travel time variability due to its influences on travelers' mode, route and departure time choices. Operating and management agencies have also increased efforts for monitoring and improving the reliability of transportation systems through probe-based data collection, integrated corridor management and advanced traveler information provision. In particular, vigorous data collection efforts have been made to improve measurement quality and uncover the root sources of travel time unreliability. For instance, a wide range of corridor management strategies have been designed to balance traffic between freeway and arterial corridors in response to various nonrecurring traffic congestion sources. In addition, advanced traveler information provision systems have been enhanced to provide reliability-related information to enable travelers to meet their limited travel time budget constraints. There also appears to be a growing trend toward incorporating end-to-end trip travel time variability measures, and their related traveler behavior components, into traffic network analysis and management models.

Although noteworthy progress has been made in quantifying the variability in travel

times, a number of challenges still need to be addressed. One is how to estimate distributions of individual vehicle travel times for both recurring and nonrecurring congestion conditions, especially since both the demands and the capacity are stochastic. A framework for doing this is vital for both travelers and operating agencies (e.g., traffic management team) if they are to make informed decisions about actions that improve reliability. Past research often assumes that the demands are stochastic, but not the capacity. Additionally, merging and diverging locations cause significant disturbances. Hence, the effects of these locations need to be carefully examined.

Thus, the first objective of this dissertation is to develop a travel time reliability estimation methodology to support the active monitoring and management of the traffic conditions and further provide informed decisions for the management and operation agencies and travelers.

1.2.2 Traffic State Estimation Model

Traffic state estimation techniques are often used to guide operational management decisions, where the goal is to reduce instability and volatility in the transportation system by encouraging more efficient operation for better end-to-end trip travel time reliability and reduced total emissions. By closely monitoring and accurately estimating the state of the traffic system using heterogeneous data sources, it is possible to apply information provision and control actions in real time to best utilize the available highway capacity. In addition, to generate anticipatory and coordinated control and information supply strategies, intelligent traffic network management systems call for accurate and reliable estimation of time-varying traffic flow patterns. Essentially, any

application of real-time and data-driven traffic decision support systems involves estimation of traffic states, e.g., microscopic states for detailed emission analysis and path-based travel time for personalized route guidance.

The second objective of this dissertation is to investigate two challenging questions in the area of traffic flow estimation: how to reconstruct macroscopic freeway traffic states from heterogeneous measurements, and how to quantify the values of additional data.

1.2.3 Integrate Microscopic Emission Model into Mesoscopic Traffic Assignment and Simulation Model

The continuous growth of travel demands leads to increasing energy consumptions and greenhouse gas (GHG) emissions. Based on *Annual Energy Review 2011* by the U.S. EIA (Energy Information Administration, released in 2012), U.S. transportation petroleum consumption accounts for approximate 71% of total U.S. petroleum consumption, and more than 65% of the consumption is due to personal vehicles. Meanwhile, a dedicated CO2 emissions report – *CO2 Emissions From Fuel Combustion Highlights* (IEA, 2011) – has indicated that globally, 23% of CO2 emissions are attributed to vehicular emissions, although considerable differences are present between countries. According to U.S. EPA (Environmental Protection Agency), transportation is the largest end-use source of greenhouse gases (including direct emissions and emissions from electricity use) and accounts for 45% of the net increase in total U.S. greenhouse gas emissions from 1990-2010 (EPA, 2012).

In order to improve traffic operation and reduce greenhouse gas emissions and fuel consumption, the transportation planning and administration agencies have adapted a

number of new traffic operational policies and strategies: for example, strategic management policies to encourage public transit, carpool or nonmotorized transportation; integrated land use and transportation planning measures to shift travel modes; integrated corridor management with optimized traffic signal controls to mitigate congestion and improve air quality. Through the interplay between the change of the traveler demand and the traffic network conditions, these policies are expected to address the deteriorating congestion and environmental issues. This requires effective and efficient tools to evaluate these alternative polices.

However, the conventional four-step static traffic assignment models are not sensitive to this dynamic interaction, and thus are not efficient at evaluating the alternative policies. According to the *Special Report 288* by the National Research Council (2007), the traditional models lack the capabilities to identify the factors that impact the travelers' choice and behaviors in response to congestion and other system performance measurements. These models are also unable to generate accurate, disaggregate, estimates of time-dependent traffic conditions on target corridors.

Although new tools like Dynamic Traffic Assignment (DTA) models have been developed in the past years to address the aforementioned issues associated with the static traffic assignment models, many planners and engineers are concerned that these tools lack necessary representation details on the high-fidelity traffic dynamics. For a subarea study, one could simply extract vehicle path data from a (macroscopic/mesoscopic) DTA tool and feed the data into a microscopic simulation model to generate second-by-second vehicle speed and acceleration output for the microscopic emissions or mobility-related analysis. Nevertheless, limited efforts have been devoted to tightly interconnecting these

models into a unified system. Thus, in the third task of this dissertation, a systematic mechanism is presented to integrate the mesoscopic traffic assignment and simulation model with the microscopic emission model for more flexible and efficient evaluation of the traffic control measures and operating policy alternatives.

1.2.4 Reliability-based Traveler Information Provision and Dissemination

Under the existing loosely-distributed sensoring environment with heterogeneous data sources, transportation planning and management agencies have found a critical need for efficiently storing, processing and extracting network-level information to support the demanding traffic information requirements for informed decision making. Thanks to the advances of telecommunication and information technologies, the emerging practice of Cloud Computing provides a revolutionary solution platform to combine data archiving and information extraction processes. Thus, the last task of this research is to demonstrate the applicability of the cloud computing technique in traveler information provision and dissemination.

1.3 Organization of the Dissertation

This dissertation is organized as follows. Chapter 2 provides a comprehensive literature review for the related research tasks. Chapter 3 discusses the design of the corridor-level travel time and reliability quantification methodology. The traffic state estimation problem is investigated in Chapter 4. The first part of Chapter 5 describes a simplified approach for model emissions, based on MOVES model; and the second part details how to apply Newell's car-following model to generate second-by-second vehicle

operating parameters for the microscopic emission models. A demonstrative example implementation of the Cloud Computing-based travel time reliability information provision system is introduced in Chapter 6. Conclusions are drawn in Chapter 7.

CHAPTER 2

LITERATURE REVIEW

This chapter presents literature reviews for the topics related to travel time models, traffic state estimation models and emission models. In Section 2.1, deterministic link-based travel time models and the underlying distributions of capacity and demand elements are examined. This is followed by a discussion on the various modeling approaches for connecting travel time variability with its root sources. Section 2.2 provides a comprehensive review of the major traffic state estimation and prediction models. In Section 2.3, the macroscopic and microscopic emission models are compared.

2.1 Travel Time Models and Capacity/demand Element Distributions

Within the subject of analytical dynamic traffic network analysis, the "whole-link" model is widely adopted to describe link travel time evolution due to its simple description of traffic flow propagation through an analytical form. The link travel time function introduced by Friesz et al. (1993) defines the travel time $\tau(t)$ on a single link at a time *t* as a linear function of the number of vehicles $x(t)$ on the link at time *t*:

$$\tau(t) = a + bx(t) \tag{2.1}$$

where a and b are constants in the above general linear form. A nondecreasing and continuous function is defined to calculate the number of vehicles on the link based on the inflow and outflow rates, $u(t)$ and $v(t)$, at time t:

$$x(t) = x(0) + \int_0^t \left(u(s) - v(s)\right) ds \tag{2.2}$$

Meanwhile, some more general nonlinear travel time functions have been proposed as:

$$\tau(t) = f\left(x(t), u(t), v(t)\right) \tag{2.3}$$

A special case of this form, introduced by Ran et al. (1993), decomposes the link travel time as two different functions: g_1 accounts for flow-independent travel time and g_2 accounts for the queuing delay. A detailed mathematical representation is shown below:

$$\tau(t) = g_1\left[x(t), u(t)\right] + g_2\left[x(t), v(t)\right] \tag{2.4}$$

They later showed that, by assuming g_1 and g_2 are separable, i.e., $g_1 = g_{1a}\left[x(t)\right] + g_{1b}\left[u(t)\right]$ and $g_2 = g_{2a}\left[x(t)\right] + g_{2b}\left[v(t)\right]$, Eq. (2.4) can be rewritten as:

$$\tau(t) = \alpha + f\left(u(t)\right) + g\left(v(t)\right) + h\left(x(t)\right) \tag{2.5}$$

where α is the free flow travel time, and $f(\cdot)$, $g(\cdot)$ and $h(\cdot)$ correspond to the functions of link inflow rate, link outflow rate and the number of vehicles on the link, respectively.

Daganzo (1995) draws attention to problems with the general form in Eq. (2.3), indicating that either a rapid decline in the inflows $u(t)$ or a rapid increase in outflow $v(t)$ would lead to unrealistic travel time. Thus, he recommended omitting $u(t)$ and $v(t)$ from Eq. (2.3), reducing the link travel time to a function of the number of vehicles on the link, that is, $\tau(t) = f\left(x(t)\right)$.

Although the link travel time function models provide some degree of simplification on travel time analysis, there is one significant drawback. Traffic congestion usually occurs at some bottleneck, and queues are produced and often grow beyond the bottleneck, which is difficult for any travel time function to capture (Zhang and Nie, 2005).

In dynamic traffic assignment and other applications, the vertical queue or point-queue model (Daganzo, 1995) was widely adopted to describe bottleneck traffic dynamics (Zhang and Nie, 2005). In a queuing-based travel time model, it is important to capture the variations of queue discharge flow rates and incoming demand to a bottleneck.

Conventionally, freeway capacity is viewed as a constant value – the maximum discharge flow rate before failure (HCM, 2000). However, the capacities vary according to different external factors in real-life situations. Conceptually, capacity or discharge flow rate can be represented as the reciprocal of the average of vehicle headways. Over the past decades, many researchers have developed a number of headway models to describe its distribution. Representatives of these models include the exponential-distribution by Cowan (1975), and normal distribution, gamma-distribution and lognormal-distribution models by Greenberg (1966).

Incidents are one of the major contributing factors in capacity reductions, and the magnitude and duration of capacity reductions are directly related to the severity and duration of incidents (Guiliano, 1989; Kripalari and Scherer, 2007). In quantifying capacity reduction, the HCM 2000 provides guidance for estimating the remaining freeway capacity during incident conditions. Using over two years of data collected on freeways in the greater Los Angeles area, Golob et al. (1987) found that accident duration fit a lognormal distribution. By extending the research of Golob et al., Guiliano (1989) applied a lognormal distribution when analyzing incident duration for 512 incidents in Los Angeles.

It is commonly observed that travel demand fluctuates significantly within a day. During the morning and evening peak hours, surging demand may overwhelm a roadway's physical capacity and results in delays (FHWA, 2009). Waller and Ziliaskopoulos (2001), Chen et al. (2003) and Lam et al. (2008) have used the normal distribution for modeling travel demand variation. Other researchers have modeled travel demand using the Poisson distribution (Hazelton, 2001; Clark and Watling, 2005) and the uniform distribution (Ukkusuri et al. 2005).

2.1.1 Methods for Estimating Travel Time Variability

Substantial efforts have been devoted to travel time variability estimation over the last decade, producing several different approaches for estimating travel time variability. Statistical approaches (Richardson, 2003; Oh and Chung, 2006) have been widely adopted to quantify travel time variability from archived sensor data. In recent studies investigating the different sources of travel time variability, Kwon et al. (2010) proposed

a quantile regression model to quantify the 95th percentile travel time based on the congestion source variables, such as incidents and weather. In their multistate travel time reliability modeling framework, Guo et al. (2010) provided connections between the travel time distributions and the uncertainty associated with the traffic states, e.g., with incidents vs. without incidents. In addition, they (Park et al. 2010) show that a multi-mode model could lead to better representations of real-world observations compared to single-mode models (represented by mean and variance parameters).

A second approach uses numerical methods to characterize travel time distributions as a result of stochastic capacity and stochastic demand. Given a stochastic capacity probability distribution function (PDF), a Mellin transforms-based method was adopted by Lo and Tung (2003) to estimate the mean and variance of travel time distributions. Using a sensitivity analysis framework, Clark and Watling (2005) developed a computational procedure to construct a link travel time PDF under stochastic demand conditions. Given various sets of traffic flow assignment results, Ng and Waller (2010) introduced a fast Fourier transformation approach to approximate the travel time PDF from underlying stochastic capacity distributions. Although it can quantify the impacts of demand and capacity variation on the travel times, the steady-state travel time function-based approach is still unable to address the underlying time-dependent traffic dynamics.

In order to account for the inherent time-dependent traffic dynamics, some researchers have incorporated point-queue models into travel time variability estimation techniques. Assuming lognormal distributions on capacity and demand, Zhou et al. (2010) adopted a point-queue model and a cumulative count curve approach to quantify the day-to-day travel time variability. For single bottlenecks, the travel time variability is

analytically derived from the variation parameters in demand and capacity. The challenging issue in extending their model on a corridor-level analysis is how to quantify route-level travel time along several corridors where downstream and upstream traffic states are correlated. Using a dynamic traffic assignment simulator, Alibabai (2010) developed an algorithmic framework to investigate the properties of the path travel time function with respect to various path flow variables. While realistic simulation results require significant efforts in simulation/assignment model calibration, this approach is particularly suited for studying the effects of various uncertainty sources and assessing the benefits of traffic management strategies and traffic information systems.

2.2 Traffic State Estimation Methods

Substantial research efforts have been devoted to traffic state estimation and prediction using macroscopic traffic flow models. Typical macroscopic traffic flow models contain a fundamental diagram, a fluid conservation law and a speed-concentration equation. The most notable macroscopic model is the first-order model, developed by Lighthill-Whitham-Richards (Lighthill and Whitham, 1955; Richards, 1956), which describes traffic flow as a one-dimensional compressible fluid with partial differential equations (PDEs). Based on a triangular shaped flow-density relation, Newell's simplified kinematic wave (KW) model (2003), or the three-detector model, which has been systematically described by Daganzo (1997), considers the cumulative flow count at an intermediate location of a homogeneous freeway segment as a minimization function of the upstream and downstream cumulative arrival and departure counts. Similarly, the classic Cell Transmission Model (CTM) proposed by Daganzo

(1994) captures the transfer flow volume between cells as a minimum of sending and receiving flows.

Given a freeway corridor with multiple point detectors, different macroscopic traffic flow models are embedded into traffic state estimators or traffic analysis tools to capture traffic flow, density and queue lengths on each link segment of the corridor. Early studies such as Gazis and Szeto (1972) focus on traffic condition estimation on short road sections. Starting from the early 1980s, many studies such as Cremer and Papageorgiou (1981) adopted the computationally efficient Kalman filtering technique to estimate traffic state variables along a freeway corridor, where on-ramp flows, off-ramp flows and entering flows from the upstream section are typically assumed to be available.

Many on-line traffic estimation systems involving linear Kalman filtering find modeling difficulties regarding how to integrate traffic flow models as transition equations and how to capture the complex nonlinear traffic flow dynamics evolution such as queue spillbacks. Recognizing the nonlinearity in those models, some studies (e.g., Wang and Papageorgiou, 2005; Wang et al., 2007) adopt extended Kalman filtering formulations to linearize the underlying process and measurement equations. Based on the CTM and a mixture Kalman filter, Muñoz et al. (2003) and Sun et al. (2003) proposed a novel switching-mode model to describe traffic dynamics and transform the minimization equations in the CTM into a set of piecewise linear equations, and the probabilistic state space is approximated through a finite number of mode sample sequences. Mihaylova et al. (2007) developed a CTM-based second-order macroscopic traffic flow model and adopted a particle-filtering framework to avoid the nonlinearity.

To fully utilize information from emerging traffic probe data, a Lagrangian sensing

framework also received much attention recently. For example, Nanthawichit et al. (2003) integrated Payne's traffic flow model and Kalman filtering within a Lagrangian sensing framework, and Work et al. (2010) derived velocity-based partial differential equations to construct linear measurement equations for linking GPS local speed data with internal traffic state variables.

While significant progress has been made in estimating macroscopic traffic conditions from available heterogeneous data sources, there are still a number of theoretically challenging but practically important questions which need to be addressed.

First, in many existing link density-based estimation methods, the raw traffic observations (e.g., point occupancy and time-mean speed) from a point sensor are typically used to approximate the link-based states (e.g., space-mean speed or link-based density), and the exact location of a sensor, typically in the middle of a link, is not explicitly recognized. However, when the discretized link length is relatively long (e.g., 0.5 miles), the difference between point measurements and link-based states becomes significantly large, and this inconsistency must be systematically taken into account. AVI and AVL data are widely used in corridor-level traffic time collection applications (Sherali et al., 2006), but limited progress has been made in utilizing end-to-end travel time (especially across multiple links) to estimate traffic dynamics along the corridor in space and time. Thus, theoretically rigorous models are critically needed to correctly map both point measurements and end-to-end travel time samples to internal states of traffic estimators.

Second, although Kalman filtering (KF) has demonstrated its excellent computational efficiency through the recursive updating of traffic state estimates at each estimation

interval (typically 30 seconds or 1 minute), it has inherent difficulties in modeling several key issues related to congestion propagation. During congested conditions, each traveler takes a certain time to pass two nonconsecutive AVI reading stations, and the resulting travel time sample may cover multiple estimation intervals. In this case, a recursive KF estimator might fail to re-update lagged traffic states at the previous time intervals. On the other hand, attributing lagged measurements only to traffic states during the current estimation stage may potentially cause significant bias in estimation results. One can extend the dimension of the state variable vector to include all of the lagged state variables in the current estimation stage (Okutani and Stephanedes, 1984), but the resulting expanded state space could significantly increase the computational complexity. Another modeling issue is how to incorporate inequality constraints into a standard KF estimator, while various inequalities are needed to ensure the nonnegativity for all traffic state variables and enforce the jam density constraints on each link for capturing possible queue spillbacks. Although the minimization function in a CTM-based model can be easily expressed as a set of two inequality constraints (e.g., a linear programming formulation proposed by Ziliaskopoulos, 2000), many existing KF-oriented estimators have to deploy multiple transition equality and probabilistic sample representations to approximate the feasible spaces and detect the binding constraints.

Based on a representation of cumulative flow counts, Newell's KW model is another viable modeling alterative to address the aforementioned challenging issues, and it has been adopted by various studies (Hurdle and Son, 2000; Coifman, 2002; Mehran et al., 2011) to represent measurements from a point sensor in terms of cumulative flow counts at the boundaries of a homogenous segment. This cumulative flow count-based

characterization has also been shown by Ni (2007) to derive various traffic state estimates for ITS applications. Furthermore, Daganzo (2003, 2004, 2006) presented an extension to the variational formulation of kinematic waves, and showed the equivalence between Newell's KW model and Newell's simplified linear car-following model (Newell, 2002). Ni (2004) incorporated a merge and diverge model within Newell's KW model to address the network traffic state estimate problem in a freeway system. Recently, Deng and Zhou (2012) extended Newell's three-detector method to consider AVI, AVL and point sensor measurement noises on a single homogeneous link.

2.3 Emission Estimation Models

The regulatory emissions model, MOBILE6 (EPA, 2003), from the U.S. Environmental Protection Agency (US EPA) has been widely adopted to evaluate the environmental impacts of transportation projects. In this model, the estimates of emission rates are expressed as functions of average speed and based on limited numbers of driving cycles. As pointed out by Rakka et al. (2004), the driving cycle-based emission estimating approach adopted by MOBILE6 is not sensitive to change in vehicle acceleration behavior, especially under stop-and-go situations. This may cause MOBILE6 to generate identical emission estimates for trips with identical average speeds on the same roadway facility. However, a sample data set by Oak Ridge National Laboratory (Ahn et al., 2002) for one of the test vehicles clearly demonstrates the large nonlinear behavior in all pollutants as a function of the vehicle speed and acceleration. Furthermore, a number of researchers (Unal et al., 2003; Frey et al., 2001; Frey et al., 2003; Frey et al., 2008) have used Portable Emission Monitoring Systems (PEMS) to

collect second-by-second micro-scale real-world, on-road emission data. These data enable the quantification of variability in emissions measurements for representative real-world trips. Based on these data, these researchers have noticed that the standard driving cycles may not reflect the real-world traffic patterns under various traffic flow conditions with different traffic operating and improvement strategies. Due to this incapability for evaluation of the microscale impact of traffic dynamics, the application of MOBILE6 is limited in estimating large-scale emission impacts and poorly suited to estimate the emission-reduction benefits of traffic control measures and traffic improvement plans.

To keep up with new analysis needs, modeling approaches and data, the new EPA MOVES model has been formulated to address different geographic scales of emissions and energy consumption estimation from national, regional and local to project-level inventories. As part of the development of the conceptual basis for MOVES, researchers (Frey et al., 2002) developed a second-by-second database, incorporating data from PEMS and dynamometer data. This database was used to identify key explanatory variables and Vehicle Specific Power (VSP) was consistently identified as the most important explanatory variable (Jimènez, 1999; Huai et al., 2005; Zhai et al., 2008; Morris et al., 2012). VSP is a second-by-second function of vehicle speed, acceleration, and road grade that accounts for kinetic energy, rolling resistance, aerodynamic drag and gravity in combination with road grade. A VSP binning-based methodology was developed to estimate vehicle emissions (Frey et. al., 2002). Different bins account for different combinations of speed, acceleration and road grade.

The current version of the MOVES model contains a finer definition of VSP (operating) modes (EPA, 2007). For example, bin 0 and bin 1 include emissions during

idling and deceleration, respectively. Bins 11 to 16 characterize the emissions during low speeds (< 25 mph), bins 21 to 30 address the emissions for vehicle travel speed between 25 mph and 50 mph and bins 30 to 40 describe the emissions under high travel speed range (> 50 mph). Each bin is associated with a distinct emission rate calibrated from the collected PEMS or other field test data, for each specific vehicle type and vehicle age.

The major differences between macroscopic and microscopic emission models are summarized in Table 2.1. Thanks to this operating modes-based modeling approach, MOVES is able to account for different patterns of acceleration, cruising and deceleration. Nevertheless, a common challenge in using MOVES and other microscopic energy and emission models, as pointed out by Wang et al. (2011), is the lack of very detailed spatial and temporal resolution of vehicle activity data. Microscopic traffic simulation tools have been widely used to generate vehicle emissions estimates by evaluating driving speed and acceleration characteristics/profiles on a vehicle-by-vehicle and second-by-second basis. Although a high-fidelity traffic simulator is desirable for analyzing individual movement delays and facilities with complex geometric configurations, microscopic simulation typically requires a wide range of detailed geometric data and driving behavior parameters, which are difficult to calibrate and are computationally intensive. Another popular vehicle activity data collecting method is using probe vehicles equipped with GPS units to collect the vehicle activity data. The major drawback of this method is the collected data are only samples of the traffic state and may not reflect the complex dynamics of real-world traffic evolution, especially under low market penetration rates of GPS probes.

Table 2.1: Comparison Between Macroscopic and Microscopic Emission Models

	Macroscopic Emissions Models	Microscopic Emissions Models
Methodology	Average trip speed	Modal activity
Emission Estimation	Trip-based vehicle average speed	Distribute total activity into source and operating mode bins with second-by-second speed and acceleration
Spatial Scale	Large-scale network	Large scale as well as local and link level
Computational Efficiency	Fast	Slow
Accuracy	Outdated (more than 20 years old) driving cycle data; various degrees of under- and over-estimation of emissions	Includes much larger, finer and more renewed data, thus more accurate
Representative Models in US	MOBILE (EPA), EMFAC (State of California)	MOVES (EPA), CMEM (State of California)
Sensitive to Vehicle Behaviors	Not sensitive	Sensitive; emissions change with vehicle behavior

Lately, applying mesoscopic traffic assignment and simulation to generate detailed vehicle activities has drawn the attention of researchers and planners. Given a set of link-based mesoscopic traffic simulation results from DTA (e.g., using a time resolution of 6 seconds), one can use a postprocessing procedure to generate second-by-second detailed vehicle trajectories based on a simple Linear Car-Following model (LCF) proposed by Newell (1962, 2002). A recent paper by Daganzo (2006) proved that, by assuming a

triangular flow-density, vehicle trajectories constructed from a simplified kinematic wave model are equivalent to those generated by Newell's simple linear car-following model and two types of Cellular Automata (CA) models within a certain approximation range. In a calibration and validation study by Brockfeld et al. (2004) for a number of well-known car-following models, Newell's simplified LCF model showed reasonable performance with limited calibration efforts.

CHAPTER 3

TOWARDS PREDICTING CORRIDOR-LEVEL TRAVEL TIME DISTRIBUTION BASED ON STOCHASTIC FLOW AND CAPACITY VARIATIONS

This chapter aims to establish a point-queue-based end-to-end travel time simulation method on a corridor with multiple merges and diverges. A set of analytical equations is developed to calculate the number of queued vehicles ahead of the probe vehicle and further capture many important factors affecting end-to-end travel times: the prevailing congestion level, queue discharge rates at bottlenecks and flow rates associated with merges and diverges. Based on multiple random scenarios and a vector of arrival times, the experienced delay at each bottleneck along a corridor is recursively estimated to produce end-to-end travel time distributions. The remainder of this chapter is organized as follows. Section 3.1 describes the point-queue-based end-to-end travel time estimation framework with deterministic inputs, along with an illustrative example. The travel time distribution with stochastic inflow, outflow and discharge rates is estimated with Monte Carlo simulation in Section 3.2. The proposed travel time estimation model is verified with the NGSIM dataset and the results are reported in Section 3.3.

3.1 Computing Route-level Travel Times

3.1.1 Problem Statement

Consider a corridor with *M* bottlenecks, where each node in the node-link structure represents a bottleneck, and the road segments between consecutive bottlenecks are links with homogeneous capacity. Assume that node *0* is the starting point of the corridor, node *m* corresponds to bottleneck *m* and each link between bottlenecks is denoted as link (*m-1*, *m*), for $1 \le m \le M$. Link (*m-1*, *m*) is the same as link *m*. Figure 3.1 illustrates a node-link representation for a corridor with *M* bottlenecks. Possible merge or diverge nodes are connected to bottleneck *m* and are denoted as *m′* or *m″*, respectively, so that the on-ramp before node 1 is denoted as (1', 1), and the off-ramp before node 2 is denoted as (2, 2"). In other words, the merge and diverge links are directly connected to the bottleneck. If there are more than one inflows or outflows between two bottlenecks, one can further decompose the link between those inflow entrances to several segments and merge/diverge points so as to construct the above node-link representation.

For purposes of this analysis, the interest lies in how to estimate the travel time distribution for trips from point *0* to point *m* for a probe vehicle *z*, departing at time t_0 = 0. The aim is to estimate the distribution of the route-level (route) travel time, p_m^z , based on the following: (1) the number of vehicles $x_m(t_0)$ on each link *m* along the path at time

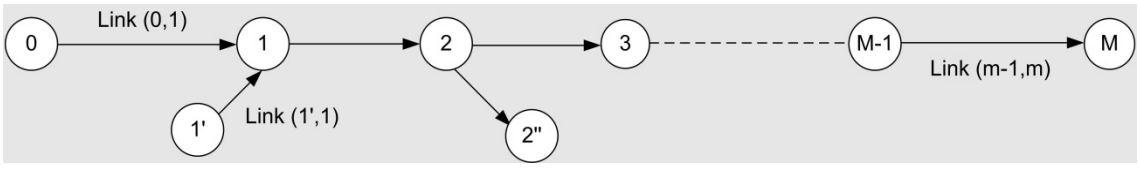


Figure 3.1: A Node-link Representation of a Corridor with *M* Bottlenecks

t_0, (2) the discharge flow rate for each bottleneck c_m and (3) the on-ramp or off-ramp flow rates f_m^{net}. The route-level travel time is defined to be the difference between the departure time at bottleneck *0* and the departure time at bottleneck *m* for probe vehicle *z*. Further, the departure time at bottleneck *m* is defined to be the time the probe vehicle leaves the queue at bottleneck *m*, which is the time when the number of vehicles in the queue before probe vehicle *z* at bottleneck *m* is 0.

The number of vehicles on each link is assumed to be observable from sensors, such as loop detectors, and the discharge flow rates and net flow rates on the on-ramps and off-ramps are assumed to be estimable from historical flow patterns or estimated based on prevailing traffic conditions (e.g., capacity reduction due to incidents).

The notation for the route-level travel time is described below.

Indices:

z: index for identifying a probe vehicle;

k: index for the simulation instance used in Monte Carlo simulation;

m: index for the bottlenecks and links along the corridor.

Inputs:

t_0: starting time, $t_0 = 0$;

M: number of bottlenecks along the corridor of interest;

$FFTT_m$: free-flow travel time over link (*m-1*, *m*);

c_m: queue discharge rate of bottleneck *m*;

f_m^{net}: net flow rate at a merge or diverge corresponding to bottleneck *m*, that is, from an on-ramp to the mainline segment or from the mainline to the off-ramp;

$x_m(t_0)$: number of vehicles on link (*m-1*, *m*) at time t_0;

$\mu_m(t)$: arrival rate of link *(m-1*, *m)* at time *t*;

$v_m(t)$: departure rate of link *(m-1*, *m)* at time *t*;

Variables to be calculated:

$\tau_m^z(t)$: travel time on link *m* for probe vehicle *z* entering the link at time *t*;

$\lambda_m(t)$: number of vehicles waiting at the bottleneck *m* at time *t*, that is, the number of queued vehicles behind bottleneck *m*;

w_m^z: waiting time in the vertical queue of bottleneck *m* for probe vehicle *z*;

t_m^z: arrival time for probe vehicle *z* at bottleneck *m*;

p_m^z: route-level path travel time from node *0* to bottleneck *m*.

3.1.2 Travel Time Calculation

In a point-queue model, a link can be considered as two segments: the free-flow segment and the queuing segment. A vehicle travels at free-flow speed on the free-flow segment until reaching the beginning of the queuing segment, where it joins the queue waiting to be discharged. A queue is only formed if the link demand exceeds the bottleneck capacity; that is, the link arrival rate exceeds the link departure rate.

To construct a numerically tractable model for calculating route-level travel times along a corridor with multiple bottlenecks, several important assumptions are made.

1) A point-queue model is adopted to calculate the delay on each link. On each link, a FIFO (First-In, First-Out) property is assumed to assure that any vehicles that enter the link before time *t* will exit the link before those entering

after time *t*.

2) The link traversal time is assumed to comprise a free-flow travel time and a queuing delay. The free-flow travel time is constant and flow-independent. The queuing delay is dependent on the number of vehicles in the queue when the probe vehicle arrives at the bottleneck $\lambda(t+FFTT)$ and the bottleneck queue discharge rate c_m. Thus, the link travel time is:

$$\tau_m(t) = FFTT_m + w(t+FFTT_m) = FFTT_m + \frac{\lambda_m(t+FFTT_m)}{c_m}$$

where $w(t+FFTT)$ is the queuing delay when vehicle *z* reaches the vertical queue at the bottleneck at time $t+FFTT$.

3) The merge or diverge location is coincident with the position of the vertical queue.

4) The bottleneck *m* remains congested cross the estimation horizon, which extends from the current time t_0 to the arrival time of the probe vehicle *z* at the bottleneck *m*, t_m^z. The corresponding queue discharge rates c_m and net flow rates f_m^{net} in the estimation horizon are also assumed to be constant.

The first two assumptions are widely used in queuing models. The third makes it easy to incorporate the flow rate from a merge/diverge without explicitly considering the driving distance and free-flow travel time from the merge/diverge point to the bottleneck *m*.

Eq. (3.2) considers the arrival time at the beginning of a link. By considering the arrival time at bottleneck *m* for vehicle *z*, t_m^z, the link traversal time can be rewritten as

$$\tau_m\left(t_m^z - FFTT_m\right) = FFTT_m + \frac{\lambda_m\left(t_m^z\right)}{c_m} \tag{3.1}$$

For a general queue with time-dependent arrival and departure rates, a continuous transition model can be used in Eq. (3.2) to update the number of vehicles in the queue at any given time *t*:

$$\frac{d\lambda_m(t)}{dt} = \mu_m(t - FFTT_m) - v_m(t) \tag{3.2}$$

The number of queued vehicles $\lambda_m\left(t_m^z\right)$ at time t_m^z on bottleneck *m* can be derived from Eq. (3.3), as shown in Eq. (3.4):

$$\begin{aligned} &\lambda_m\left(t_m^z\right) = \lambda_m\left(t_0\right) + \int_{t_0}^{t_m^z} \frac{d\lambda_m(t)}{dt} dt = \lambda_m\left(t_0\right) + \int_{t_0}^{t_m^z} \left[\mu_m(t - FFTT_m) - v_m(t)\right] dt \\ &= \lambda_m\left(t_0\right) + \int_{t_0}^{t_m^z} \mu_m(t - FFTT_m) dt - \int_{t_0}^{t_m^z} v_m(t) dt \end{aligned} \tag{3.3}$$

Since the fourth assumption has the bottleneck remain extant in for the entire estimation period, the departure rate is equal to the bottleneck capacity of $\int_{t_0}^{t_m^z} v_m(t) dt = c_m \times \left(t_m^z - t_0\right)$. The remaining challenge is to estimate the unknown queue length $\lambda_m\left(t_0\right)$ at time t_0, and calculate the complex integral of $\int_{t_0}^{t_m^z} \mu_m\left(t - FFTT_m\right) dt$.

To illustrate these ideas, consider the example in Figure 3.1, where m = 1. In this case, the number of vehicles $x_1\left(t_0\right)$ and the net flow rate f_1^{net} associated with bottleneck 1 are given. For the specific starting time t_0, a probe vehicle *z* enters the vertical queue of bottleneck 1 at time $t_1^z = t_0 + FFTT_1$, and the number of vehicles in the queue at time t_1^z is:

$$\lambda_1\left(t_1^z\right) = \lambda_1\left(t_0\right) + \int_{t0}^{t_1^z} \mu_1(t - FFTT_1)dt - c_1 \times (t_1^z - t_0) \qquad (3.4)$$

Now consider a simpler case without merge and diverge points, i.e., $f_1^{net} = 0$. Thanks to the first-in and first-out property, we can show that $\lambda_1\left(t_0\right) + \int_{t0}^{t_1^z} \mu_1(t - FFTT_1)dt = x_1\left(t_0\right)$. The left-hand side $\lambda_1\left(t_0\right) + \int_{t0}^{t_1^z} \mu_1(t - FFTT_1)dt$ is the total number of vehicles stored in both the free-flow segment and the queuing segment before the probe vehicle *z*. The right-hand side is the actual number of vehicles observed on the *physical* link. One can use Figure 3.2 to map or "rotate" some of the vehicles from the physical link (shaded) to the vertical stack queue, and the other vehicles on the physical link (not shaded) correspond to the vehicles that will arrive at the vertical queue between time t_0 and time $FFTT_1$ (that is, right before the probe vehicle). Notice that the length of the queue segment in the point-queue model is equal to zero and has unlimited storage capacity. Interested readers are referred to the paper by Hurdle and Son (2001) to examine the connection between physical queues and vertical stack queues.

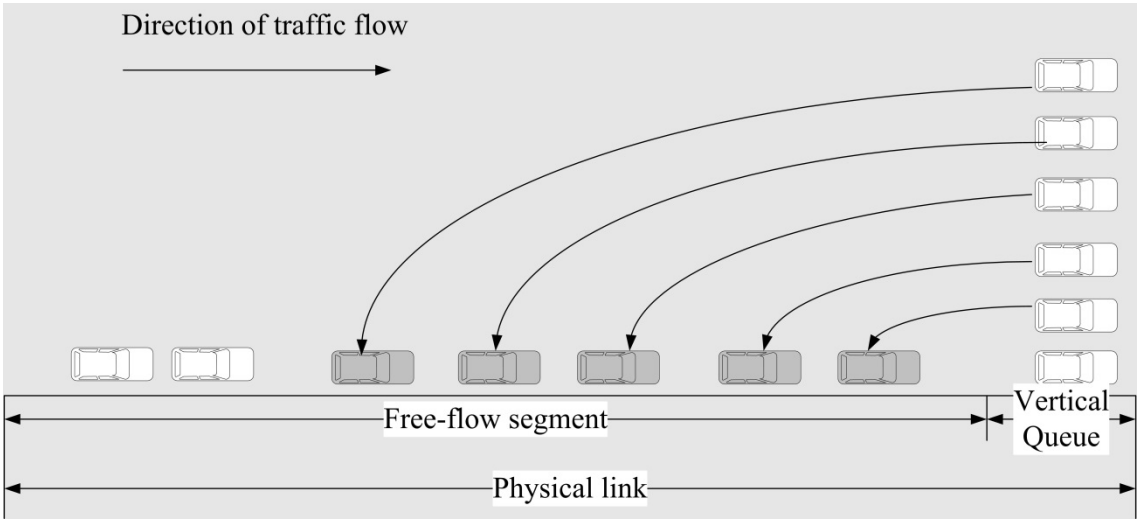


Figure 3.2: A Vertical Stack Queue

The individual components of Eq. (3.4) can be described visually using the cumulative vehicle count curves shown in Figure 3.3. Curve A is equivalent to the integral over the arrival rate, $A(t) = \int \mu_1(t)\,dt$, and the cumulative arrival curve at the vertical stack queue V is the cumulative arrival curve shifted by the free-flow travel time, $V(t) = A(t - FFTT)$, and thus $V(t) = \int \mu_1(t - FFTT_1)dt$. The cumulative departure curve D is equivalent to the integral over the departure rate, $D(t) = \int v_1(t)dt = c_1 \times (t_1^z - t_0)$. Substituting t with values of t_0 and t_1^z for $V(t)$, then Figure 3.3 shows that $V(t_1^z) - V(t_0) = \int_{t0}^{t_1^z} \mu_1(t - FFTT_1)dt$ and thus $\lambda_1(t_1^z) = \lambda_1(t_0) + \int \mu_1(t - FFTT_1)dt - c_1 \times (t_1^z - t_0)$ and $x_1(t_0) = \lambda_1(t_0) + \int_{t0}^{t_1^z} \mu_1(t - FFTT_1)dt$.

By further considering the net flow rate from the merge or diverge point connected to the bottleneck, we now have:

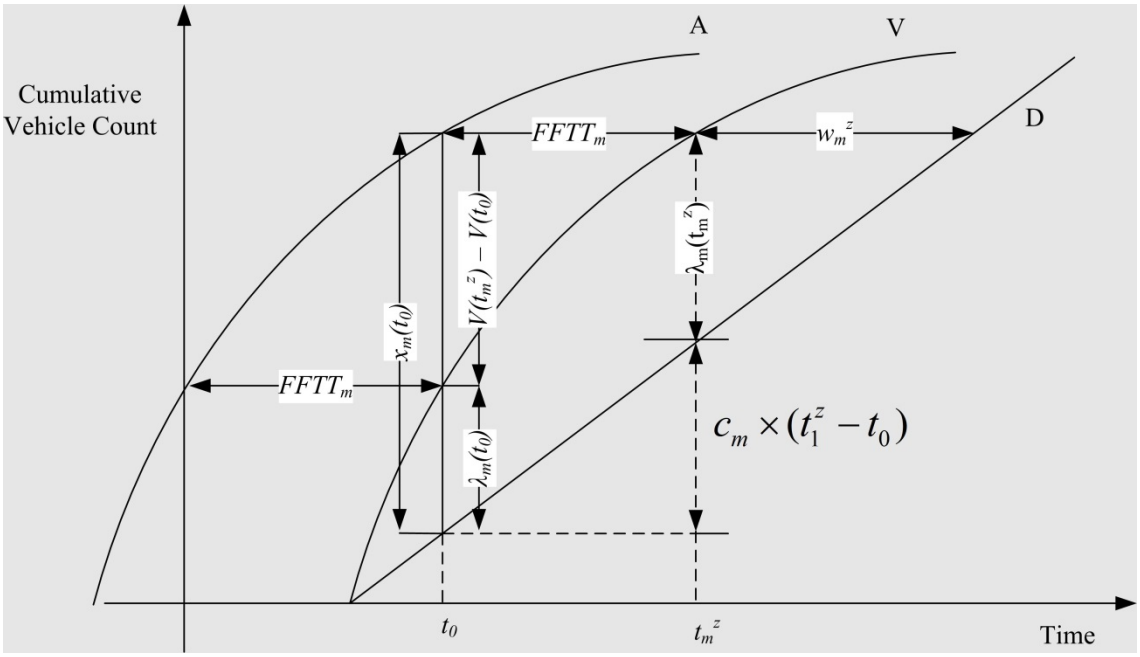


Figure 3.3: Visual Representation for Eq. (3.4)

$$\lambda_1\left(t_1^z\right)=\lambda_1\left(t_0\right)+\int_{t0}^{t_1^z}\left[\mu_1(t-FFTT_1)-c_1\right]dt=x_1\left(t_0\right)+f_1^{net}\times t_1^z-c_1\times t_1^z \quad (3.5)$$

Continuing to link 2 in Figure 3.1, the probe vehicle *z* will arrive at the queue of bottleneck 2 at t_2^z. Again, considering the FIFO assumption, the number of vehicles transferring from the first link to the second link before the probe vehicle *z* includes two terms, $\lambda_1\left(t_1^z\right)+c_1\times t_1^z$, which are the number of queued vehicles $\lambda_1\left(t_1^z\right)$ when the probe vehicle arrives at the first bottleneck at time t_1^z, and those vehicles $c_1\times t_1^z$ already entering the second link before time t_1^z. Following the derivation logic for Eq. (3.5), the number of vehicles waiting in the queue ahead of vehicle *z* when it arrives at the second bottleneck at time t_2^z is:

$$\lambda_2\left(t_2^z\right)=\lambda_1\left(t_1^z\right)+c_1\times t_1^z+x_2\left(t_0\right)+f_2^{net}\times t_2^z-c_2\times t_2^z \quad (3.6)$$

By substituting $\lambda_1\left(t_1^z\right)$ from Eq. (3.6), Eq. (3.7) reduces to:

$$\begin{aligned}\lambda_2\left(t_2^z\right)&=x_1\left(t_0\right)+f_1^{net}\times t_1^z-c_1\times t_1^z+c_1\times t_1^z+x_2\left(t_0\right)+f_2^{net}\times t_2^z-c_2\times t_2^z\\&=x_1\left(t_0\right)+f_1^{net}\times t_1^z+x_2\left(t_0\right)+f_2^{net}\times t_2^z-c_2\times t_2^z\end{aligned} \quad (3.7)$$

More generally, for bottleneck *m*:

1) The number of vehicles waiting at the vertical queue of bottleneck *m* at time t_m^z can be expressed as:

$$\lambda_m\left(t_m^z\right)=\sum_{i=1}^{m}x_i\left(t_0\right)+\sum_{i=1}^{m}\left(f_i^{net}\times t_i^z\right)-c_m\times t_m^z \quad (3.8)$$

2) The arrival time for the probe vehicle at bottleneck *m* is:

$$t_m^z = t_{m-1}^z + w_{m-1}^z + FFTT_m = t_{m-1}^z + \frac{\lambda_{m-1}\left(t_{m-1}^z\right)}{c_{m-1}} + FFTT_m \tag{3.9}$$

where $w_m^z = \frac{\lambda_m\left(t_m^z\right)}{c_m}$.

3) Finally, the route-level travel time from bottleneck 0 to bottleneck *m* is:

$$p_m^z = t_m^z + w_m^z = \sum_{i=1}^{m}\left[FFTT_i + w_i^z\right] \tag{3.10}$$

In a summary, given the number of vehicles on each link, the queue discharge rate and the net flow on each bottleneck, the route-level travel time for a vehicle can be calculated by applying Eqs. (3.9-3.11) iteratively for links 1 through *m*. In each iteration, one first applies Eq. 3.9 to obtain the number of queued vehicles at the bottleneck and then computes the queuing delay and update the route-level travel time up to the bottleneck of interest.

3.1.3 Illustrative Example

To demonstrate how to the model can be used to calculate the route-level travel time and capture the delay propagation along a corridor, a corridor with 3 bottlenecks (Figure 3.4) can be used. Bottleneck 1 is on the freeway, bottleneck 2 is associated with an on-ramp and bottleneck 3 is in conjunction with an off-ramp. The bottleneck discharge rates for those bottlenecks are 90, 90 and 60 vehicles/min, respectively. The initial numbers of vehicles on each link are 750, 600 and 650, respectively. The inflow rate for the on-ramp

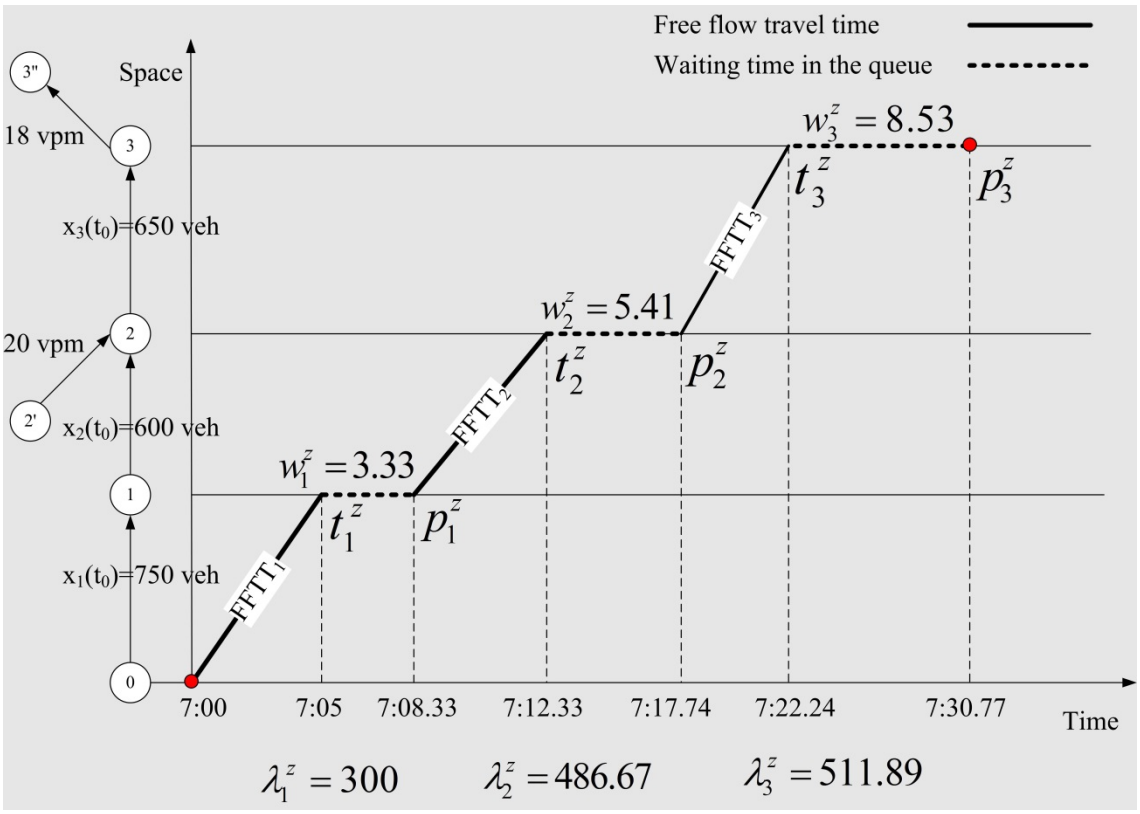


Figure 3.4: 3-bottleneck Example Corridor

at bottleneck 2 is 20 vehicles/min (vpm), which is equivalent to 1,200 vehicles per hour. The outflow rate for the off-ramp is 18 vehicles/min (vpm). The free-flow travel time over each link is 5, 4 and 4.5 minutes, respectively.

For the probe vehicle in Figure 3.4 (starting at time 7:00 AM), we now have the following calculation process for its route-level travel time.

1) Departing at 7:00, it takes 5 minutes (free-flow travel time) for this probe vehicle to reach the point-queue of bottleneck 1 at 7:05. At this time instance, the number of vehicles waiting in the queue is 750 – (5 min*90 veh/min) = 300 vehicles. With the discharge rate of 90 vehicles/min, this probe vehicle will spend w_1^z =

3.33 minutes waiting in the queue. Thus, the total travel time for this vehicle is 8.33 minutes at the end of this bottleneck.

2) The probe vehicle enters link 2 at 7:08.33, spends 4 minutes traveling through the free-flow segment and arrives at the vertical stack queue at t_2^z = 7:12.33. From 7:00 to 7:12.33, there have been 12.33 min*20 veh/min = 246.6 vehicles entering this bottleneck from the on-ramp. The number of vehicles waiting in the queue at this time is $\lambda_2\left(t_2^z\right)$ = (750 + 600) + (12.33*20) – (12.33*90) = 486.67. With the discharge rate of 90 vehicles/min, this vehicle leaves the queue w_2^z=5.41 minutes later. The departure time from the second bottleneck is 7:17.74.
3) Following the same calculation process, the number of vehicles waiting at the queue of bottleneck 3 is $\lambda_3\left(t_3^z\right)$ = (750 + 600 + 650) + (12.33*20) + (-18*22.24) – (60*22.24) = 511.89 vehicles and the waiting time in the queue is w_3^z = 8.53 minutes. This vehicle leaves bottleneck 3 at 7:30.77. The total route-level end-to-end travel time p_3^z is 30.77 minutes.

3.1.4 Travel Time Calculation Algorithm with Deterministic Inputs

The algorithm for calculating the route-level path travel time for vehicle *z* entering the corridor with *M* bottlenecks at time t_0 is summarized below.

Input: The specific starting time t_0, the number of vehicles on each link $x_m\left(t_0\right)$, the net flow rate on each bottleneck f_m^{net} and the bottleneck discharge rate c_m, at time t_0

Route-level Travel Time Calculation

For m = 1 to M

1. Calculate the arrival time at bottleneck *m*

$$t_m^z = t_{m-1}^z + w_{m-1}^z + FFTT_m \text{, where } t_0^z = 0,\ w_0^z = 0.$$

2. Use Eq. (3.9) to calculate the number of vehicles ahead of the probe vehicle *z* in the vertical stack queue of bottleneck *m*, $\lambda_m^z\left(t_m^z\right)$, when the probe vehicle *z* reaches the beginning of the queue at time t_m^z.
3. Use Eq. (3.10) to calculate the delay experienced by the probe vehicle on bottleneck *m*, w_m^z .
4. Use Eq. (3.11) to update the route-level end-to-end travel time over *m*, p_m^z

End For

Output: The route-level travel time p_M^z from bottleneck 1 to bottleneck *M*.

3.1.5 Discussion

To consider complex real-life conditions, the model must further use the following approximation methods for calculating the route-level path travel time along a corridor with multiple bottlenecks.

3.1.5.1 Approximating the Time-dependent Flow Rates with Average Flow Rates

In Eq. (3.4) and Eq. (3.5), we use the maximum bottleneck discharge rates to approximate the actual discharge rates. In reality, the rates (including the queue discharge flow rates and net flow rates from and to ramps) are highly dynamic and could fluctuate significantly even in a short time interval, as shown in Figure 3.5. In this situation, one

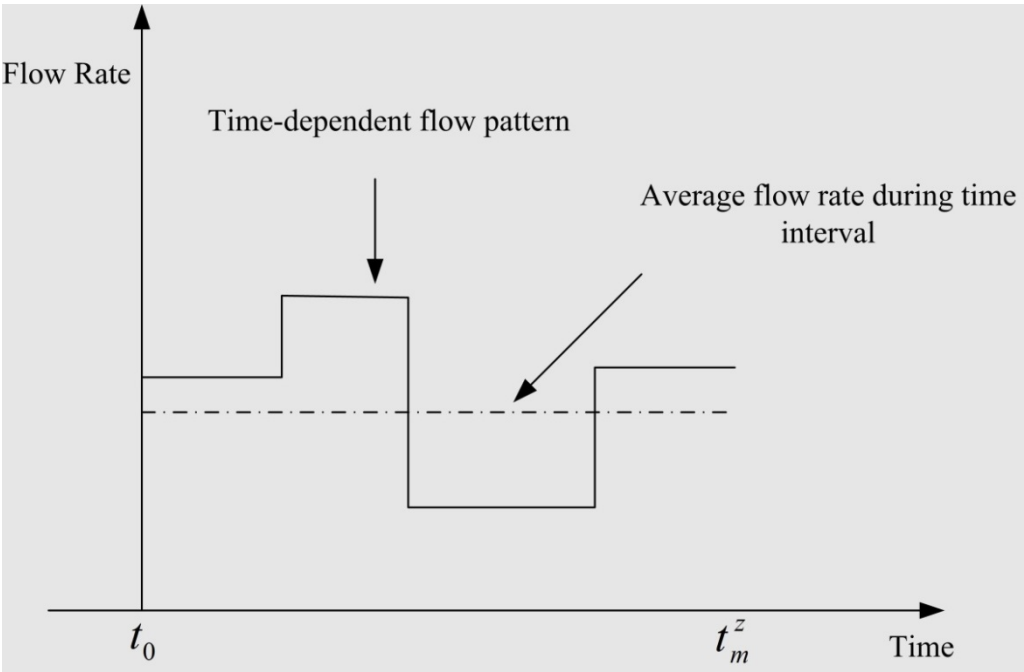


Figure 3.5: Time-dependent Flow Rate to Average Flow Rate

needs to use the average flow rate (i.e., the dashed line in Figure 3.5) during the interval from t_0 to the arrival time t_m^z to approximate the time-dependent volume. Although this approximation ignores traffic dynamics, in Eq. (3.9) it still gives a reasonable estimate about the total number of vehicles leaving or entering the bottleneck before the probe vehicle.

3.1.5.2 Considering Further Reduced Bottleneck Discharge Flow Rate Due to Queue Spillback

The proposed point-queue-based model captures the effects of queue spillback from a downstream bottleneck. Essentially, when a queue spillback occurs, the discharge capacity from the upstream bottleneck is then constrained by the discharge rates at the downstream bottleneck. The method detects spillback and then uses the reduced queue discharge rate to calculate the waiting time at the bottleneck with queue spillback.

As illustrated in Figure 3.6, the physical queue for bottleneck *m* spills back to bottleneck (*m-1*) between time t_1 and t_5 through backward waves. Interested readers are referred to the paper by Newell (1993) to learn more. Due to the queue spillback from bottleneck *m*, the actual discharge rate c'_{m-1} of bottleneck (*m-1*) between time t_1 and t_5 is constrained by the discharge rate of bottleneck *m*, c_m, rather than the original discharge rate c_{m-1}. For example, at time t_2 (where $t_2 > t_1$), a probe vehicle arrives at bottleneck (*m-1*); if the effect of queue spillback is not taken into account, this probe vehicle in the model will leave bottleneck (*m-1*) at time t_3 after waiting in the queue behind bottleneck (*m-1*), using the original, unaffected queue discharge rate c_{m-1}. With the reduced discharge rate $c_m < c_{m-1}$ at bottleneck (*m-1*), the actual waiting time for the probe vehicle will be longer with a departing time of $t_4 > t_3$.

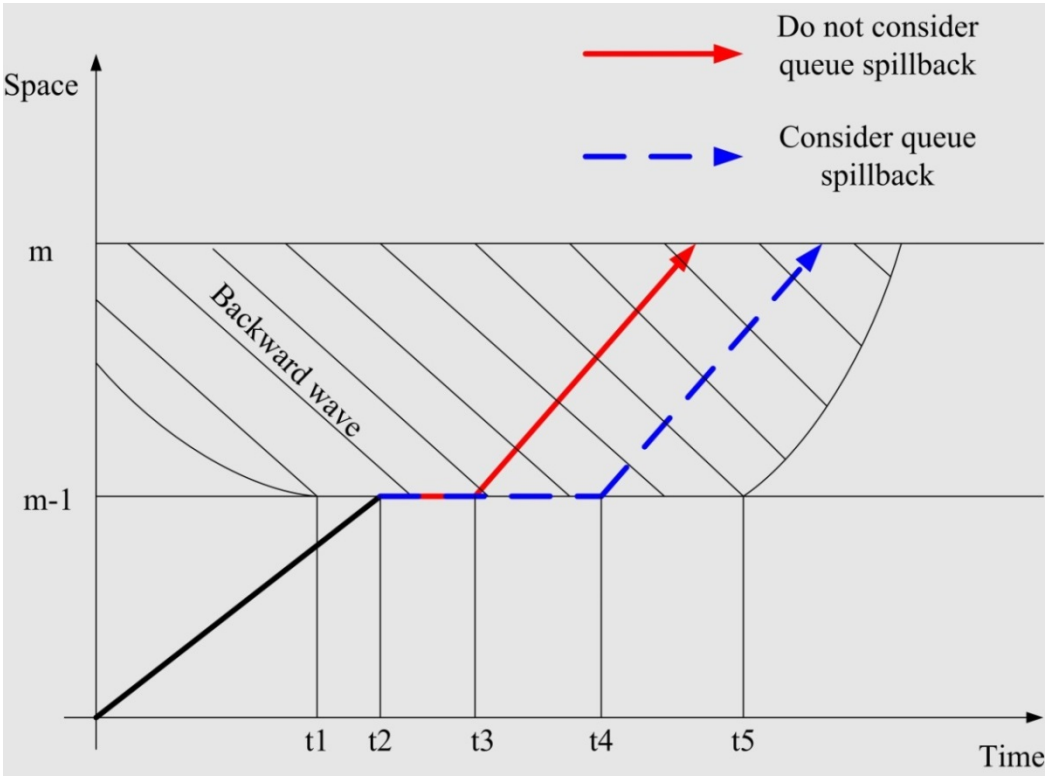


Figure 3.6: Queue Spillback

3.1.5.3 Calculating the Net Flow Rate for On-ramps

When estimating the net flow at a merge or diverge location, the flow rates in previous instances are assumed to be known and time-invariant. However, special attention must be paid to conditions where the mainline and the on-ramp are both congested. In such instances, (a) the number of vehicles that can enter the bottleneck from the on-ramp and (b) the number of vehicles that can enter from the upstream segment to the bottleneck are constrained by the mainline bottleneck discharge rate. In this case, the available bottleneck discharge rate should be allocated to the upstream segment and the on-ramp proportionally, according to certain rules (Zhang and Nie, 2005). One simple rule is to split the mainline discharge rates according to the number of lanes associated with each incoming approach.

3.1.5.4 Considering Vehicle Overtaking/passing

Lastly, the FIFO property assumed on each link rules out the possibility that a vehicle can overtake and pass another vehicle. Future research will consider the impact of this condition on route-level travel time estimation using this approach.

3.2 Methods for Calculating Route-level Travel Time Distributions

3.2.1 Assumptions

In the previous discussion, input parameters such as the net rates f_m^{net} at the merge and diverge points, and the bottleneck discharge rates c_m are assumed to be deterministic. In this section, we will further consider the variations or uncertainty in the input

parameters, especially in the following two applications: (1) day-to-day travel time variability estimation by considering flow variations at the same time period, and (2) real-time travel time reliability estimation, where the near-future traffic flows are estimated from different sources of data with various degrees of estimation uncertainty. Emphases are placed on how to calculate the route-level end-to-end travel time distribution based on the stochasticity of the random input parameters.

3.2.2 Important Observations on Path Travel Time

3.2.2.1 Simple Corridor without Merge and Diverge

Consider a simple two-bottleneck corridor with no on-ramp and off-ramp, that is, f_1^{net} and f_2^{net} are equal to 0. According to Eqs. (3.9-3.11), the route-level travel time to bottleneck 1 for probe vehicle z entering link 1 at time t_0 is:

$$\begin{aligned} p_1^z &= t_1^z + w_1^z = t_0 + FFTT_1 + \frac{x_1(t_0) - c_1 \times t_1^z}{c_1} \\ &= t_0 + FFTT_1 + \frac{x_1(t_0) - c_1 \times (t_0 + FFTT_1)}{c_1} \\ &= t_0 + FFTT_1 + \frac{x_1(t_0)}{c_1} - (t_0 + FFTT_1) = \frac{x_1(t_0)}{c_1} \end{aligned} \tag{3.11}$$

And the route-level end-to-end travel time to bottleneck 2 is:

$$\begin{aligned} p_2^z &= p_1^z + FFTT_2 + w_2^z = p_1^z + FFTT_2 + \frac{x_1(t_0) + x_2(t_0) - c_2 \times t_2^z}{c_2} \\ &= p_1^z + FFTT_2 + \frac{x_1(t_0) + x_2(t_0) - c_2 \times (p_1^z + FFTT_2)}{c_2} \\ &= p_1^z + FFTT_2 + \frac{x_1(t_0) + x_2(t_0)}{c_2} - (p_1^z + FFTT_2) = \frac{x_1(t_0) + x_2(t_0)}{c_2} \end{aligned} \tag{3.12}$$

By comparing $p_1^z = \frac{x_1(t_0)}{c_1}$ and $p_2^z = \frac{x_1(t_0) + x_2(t_0)}{c_2}$, we can make the following important observation: the proposed formula can correctly capture the correlations between the route-level end-to-end travel times p_1^z and p_2^z, as both values are dependent on the number of vehicles on link 1, $x_1(t_0)$. If $x_1(t_0)$ and $x_2(t_0)$ are assumed to be deterministic, the distributions of p_1^z and p_2^z are further dependent on the distribution of the bottleneck discharge rates, c_1 and c_2, respectively.

3.2.2.2 Simple Corridor with Merge and Diverge

If we further consider situations where a merge and diverge occur at both bottlenecks, the path travel time formulas can be expressed as follows.

$$p_1^z = t_1^z + w_1^z = \frac{x_1(t_0) + f_{net}^1 \times (t_0 + FFTT_1)}{c_1} \tag{3.13}$$

$$p_2^z = \frac{x_1(t_0) + f_1^{net} \times FFTT_1 + x_2(t_0) + f_2^{net} \times (p_1^z + FFTT_2)}{c_2} \tag{3.14}$$

The above equations introduce more complex dependencies for both p_1^z and p_2^z, and no additive formula or decomposed elements can be easily constructed to simplify these equations. This observation reinforces many previous research studies which indicate that development of the route-level end-to-end travel time distribution is extremely challenging.

3.2.3 Monte Carlo Simulation

Monte Carlo simulation is widely used to simulate the behavior of various physical and mathematical systems, especially for those problems with significant uncertainty in inputs. The model presented here uses Monte Carlo simulation to investigate the route-level end-to-end travel time distribution based on the proposed travel time calculation framework. In each simulation run, a realization of the random input parameters leads to a realization of the random path travel time outputs, which can be regarded as estimates of the true route-level end-to-end travel time variable. A sufficient number of simulations then provide a good representation of the travel time distributions under various traffic conditions and uncertainties.

The following procedure assumes all random variables are log-normally distributed, and calculates travel time distribution through K simulation runs.

Input:

The specific starting time t_0;

The distribution of the number of vehicles on each link $x_m(t_0)$, where $x_m(t_0) \sim LN\left(\mu_{x_m}, \sigma^2_{x_m}\right)$;

The distribution of the net flow rate on each bottleneck f_m^{net}, where $f_m^{net} \sim LN\left(\mu_{f_m^{net}}, \sigma^2_{f_m^{net}}\right)$;

The distribution of the bottleneck discharge rate c_m on each bottleneck, at time t_0 where $c_m \sim LN\left(\mu_{c_m}, \sigma^2_{c_m}\right)$;

Link free-flow travel time $FFTT_m$, assumed to be constant;

Number of simulations = K.

For k=1 to K,

For m=1 to M

1: Based on the underlying distribution parameters (μ and σ) of the individual inputs, generate a set of random samples for the following key variables: the number of vehicles on the link, the bottleneck discharge rate and net flow rates.

2: Call the algorithm introduced in Section 2.4 to calculate the estimated route-level travel time for simulation *k*: $p_m^z[k]$ from this set of random samples.

End For

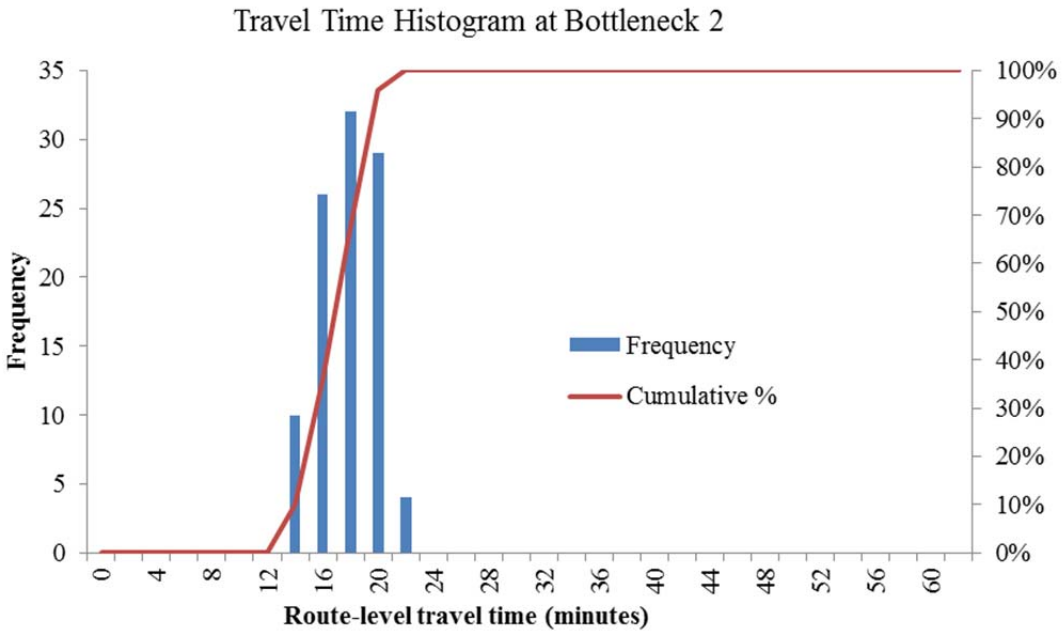
End For

Output: Calculate the histogram, mean and variance for the route-level end-to-end travel time from the results over *K* simulation runs.

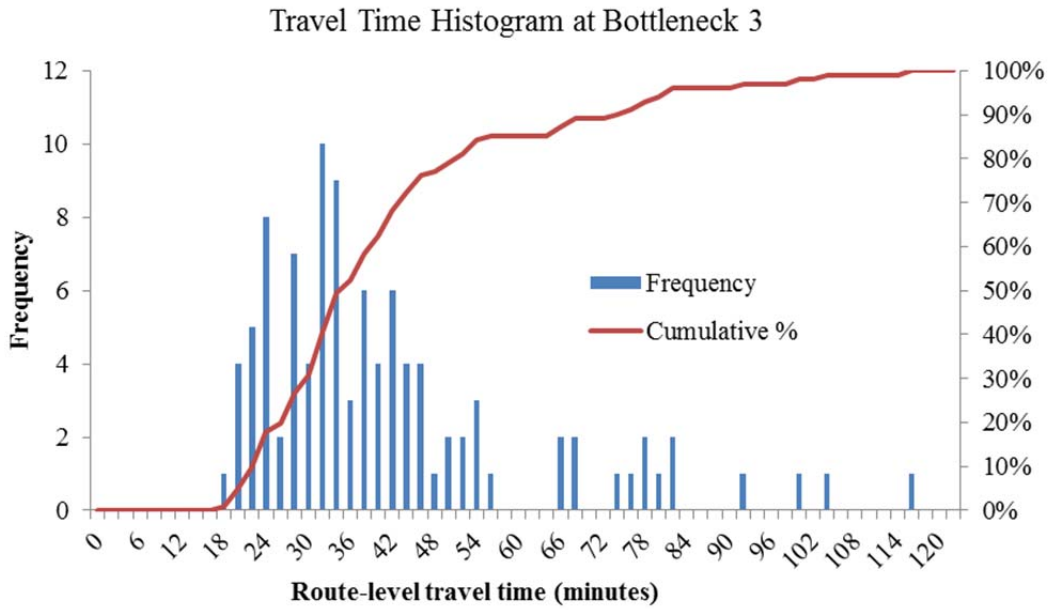
3.2.3.1 Monte Carlo Simulation

For the same example corridor in Section 3.2 (with three bottlenecks), Monte Carlo experiments were conducted to calculate the route-level travel time by assuming that the bottleneck discharge rates, inflow/outflow rates on ramps and existing number of vehicles on the link are all log-normal variables. K = 100 simulation runs were performed with different scenarios of stochastic input parameters.

Figures 3.7-(a) and (b) show the distributions of the simulated route-level end-to-end travel times p_2^z and p_3^z for probe vehicle z through bottleneck 2 and through bottleneck 3, respectively. Obviously, the mean travel time based on p_3^z is larger than that of p_2^z. In



(a)



(b)

Figure 3.7: Route-level End-to-end Travel Time Distribution

addition, a clear propagation of randomness can be observed, as p_3^z has higher variance than p_2^z. It should be noted that, by using different input distributions for flow discharge rates and the prevailing number of vehicles on the road, the resulting travel time distributions will vary. This demonstrates the advantage of the proposed model in recognizing the impact of capacity and congestion levels on travel time reliability.

3.2.3.2 Evaluating the Improvement in Reliability for Traffic Management Strategies

In the previous section, Monte Carlo simulation was used to demonstrate the application of the proposed traffic time estimation framework on the route-level end-to-end travel time distribution quantification. In this section, the calculation framework is further applied to evaluate the effectiveness of Advanced Traffic Management Strategies (ATMS).

To demonstrate the use of the proposed calculation framework, a 1-mile 4-lane freeway corridor with the average bottleneck discharge rate of 2000 vehicles per hour per lane is investigated. Before implementing ATMS, the probability of incidents on this corridor is 20% and 3 lanes are closed due to an incident if an incident occurs. After implementing ATMS (e.g., rapid incident response teams), the number of lanes closed due to incident is reduced to 2.

Figures 3.8(a-b) and Figures 3.8(c-d), respectively, show the simulated distribution of the bottleneck discharge flow rate and the calculated route-level end-to-end travel time, before and after the implementation of the ATMS. The first peak of Figures 3.8(a) and 3.8(c) represents the average capacity under incidents and the second peak represents the

capacity under normal conditions. By comparing Figures 3.8.b with 3.8.d, we can observe that the travel time distribution with ATMS has a smaller tail and less fluctuation. Figure 3.8.e further reveals the potential benefit of ATMS in improving the travel time reliability: the 95th percentile travel time is improved from 20 minutes (without ATMS) to 10 minutes (with the implementation of ATMS), while interestingly the median travel time has not changed significantly.

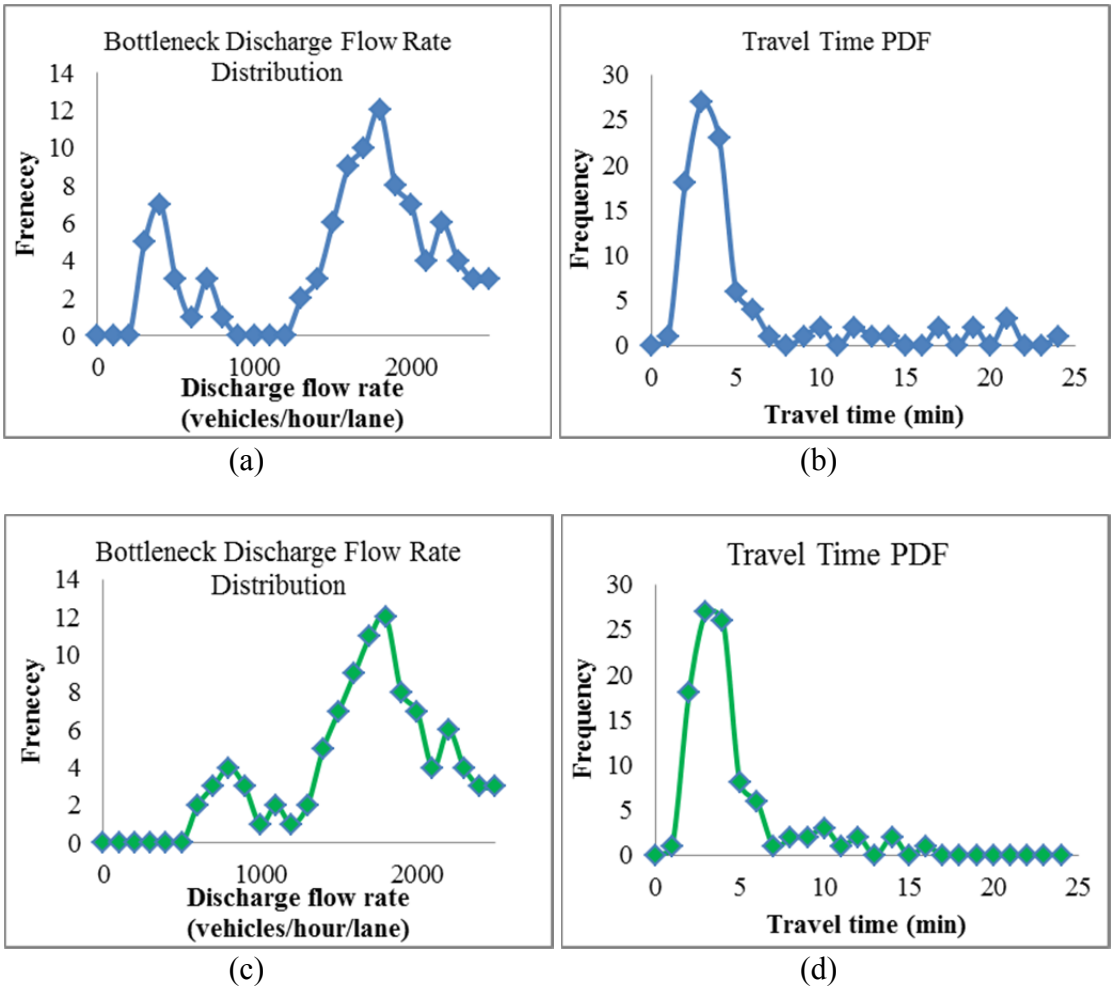
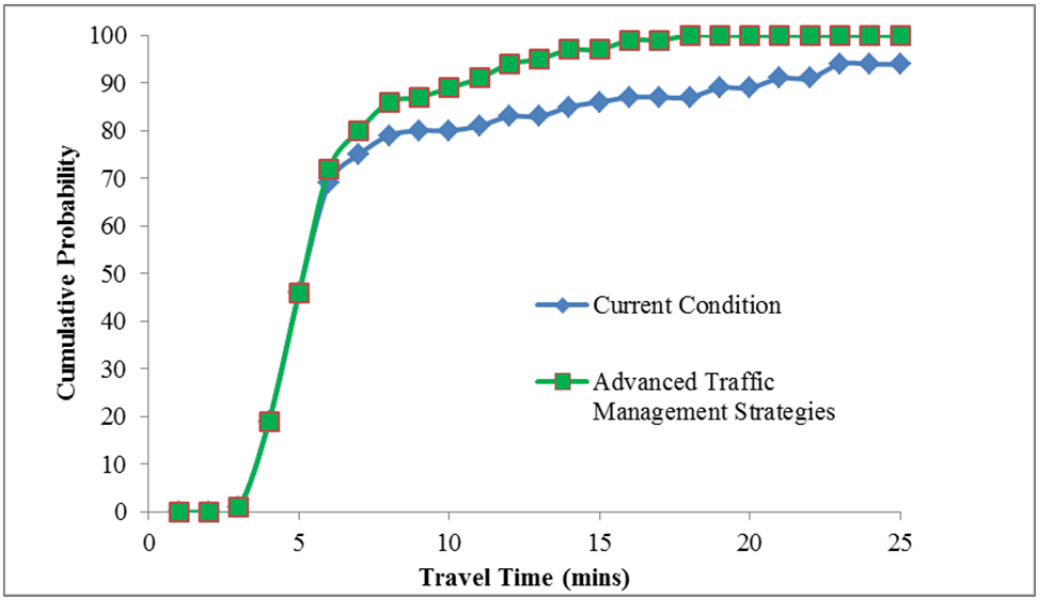


Figure 3.8: ATMS Evaluation. (a) Bottleneck Discharge Rate Distrution before ATMS, (b) Travel Time Distribution before ATMS, (c) Bottleneck Discharge Rate Distribution after ATMS, (d) Travel Time Distribution after ATMS, (e) Comparision of Travel Time CDF before and after ATMS.



(e)

Figure 3.8: Continued

3.3 Travel Time Estimation Model Validation Using NGSIM Data

This section uses vehicle trajectory data available from the NGSIM (Next Generation SIMulation) project (FHWA, 2006) as ground-truth data to verify the proposed methodology and examine the sources of estimation error.

3.3.1 Data Descriptions

The NGSIM vehicle trajectory data used in this study come from the I-80 dataset, which were collected by a video camera located at Emeryville, California. This data collection point is located adjacent to I-80, as shown in Figure 3.9. The site was approximately 1,650 feet in length, with an on-ramp at Powell Street (indicated in Figure 3.9 by the circle). The freeway segment covered in the dataset includes six lanes, numbered incrementally from the left-most lane (HOV lane). Video data are available for three time intervals: 4:00 p.m. to 4:15 p.m., 5:00 p.m. to 5:15 p.m. and 5:15 p.m. to 5:30 p.m., on April 13, 2005. Complete, transcribed vehicle trajectories are available with a time resolution of 0.1 seconds.

3.3.2 Data Extraction from NGSIM Dataset

The step-by-step data extraction procedure is described in detailed as follows:

i. To extract vehicle flow counts data from the NGSIM dataset, we first construct a node-link structure to represent the freeway segment in Figure 3.9. This stretch of freeway is divided into two links, as shown in Figure 3.10, with the on-ramp connected with node 1.

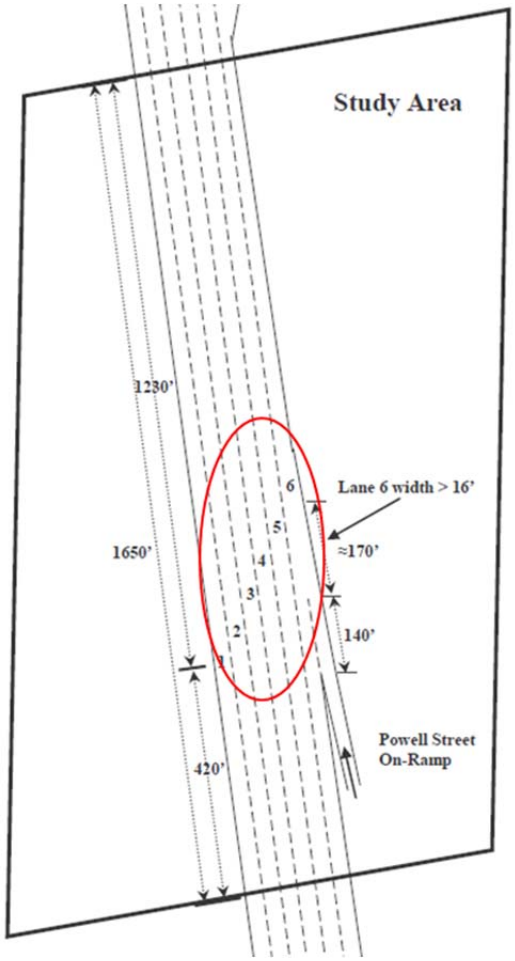


Figure 3.9: Schematic Illustration of NGSIM Study Area
Source: Adapted from Figure 1, NGSIM I-80 Data Analysis Summary Report (Cambridge Systematics, 2005)

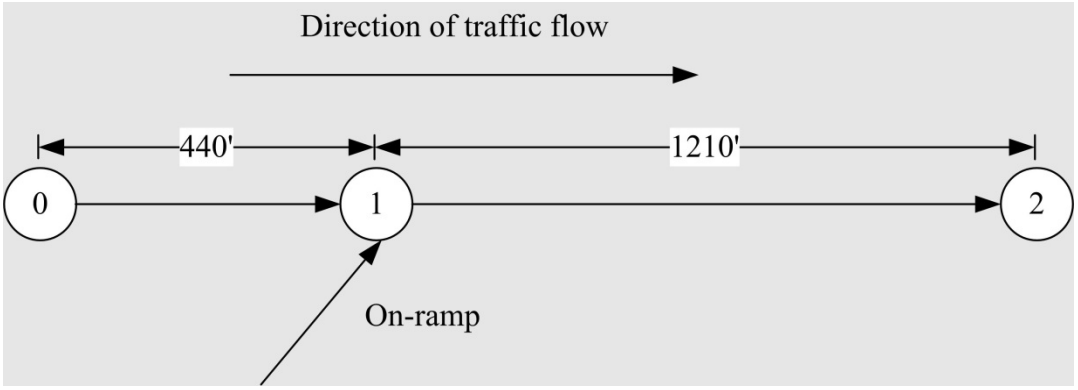


Figure 3.10: Node-link Representation of NGSIM Network

ii. In order to obtain the flow rate at the node/bottleneck, this study introduces a set of virtual detectors at node 1 and at node 2, respectively. Meanwhile, another virtual detector is placed on the on-ramp link so that inflow vehicles from the ramp are also counted. In addition, video cameras are assumed to be installed on both links to provide link snapshots (for probe vehicle data).

iii. The vehicle trajectory data are divided into 5-minute intervals for counting vehicles. An example of one 5-minute span of vehicle trajectories on one lane is shown in Figure 3.11 to illustrate how the vehicle counts are collected. As mentioned before, two sets of virtual detectors A and B are placed at nodes 1 and 2 (shown as triangles in Figure 3.11), and video cameras C and D are also installed on both link 1 and link 2. Vehicles are counted along the vertical line

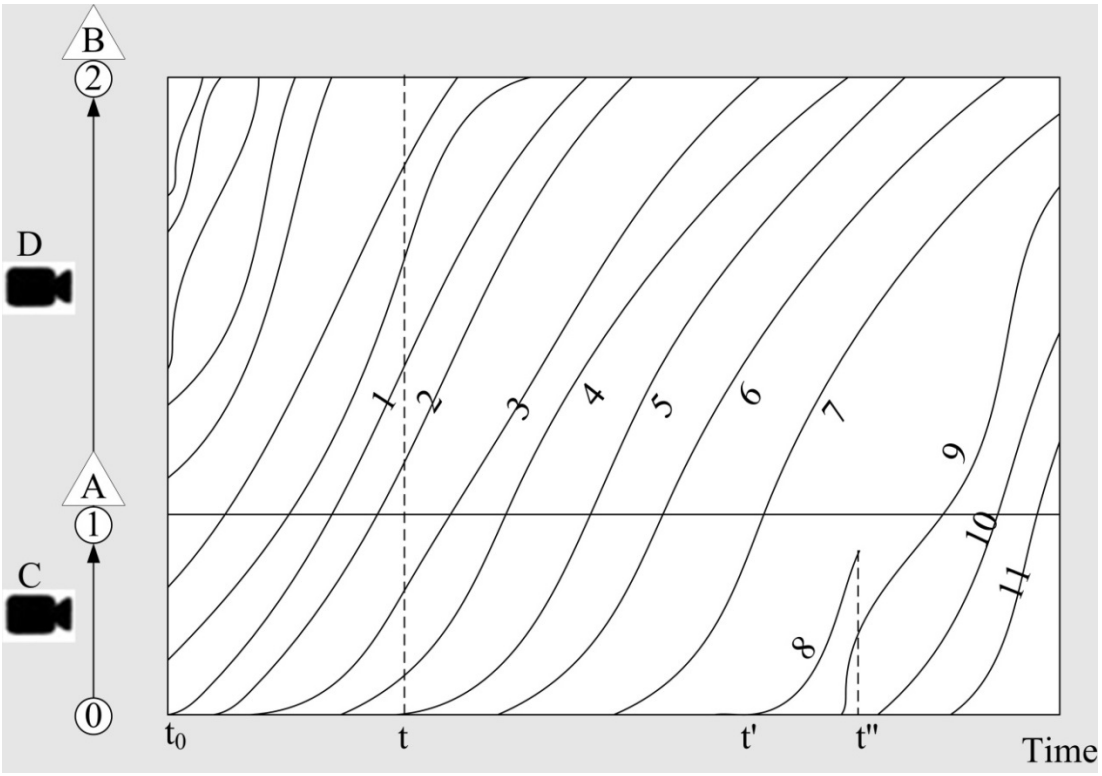


Figure 3.11: Vehicle Trajectories on a Lane

drawn at the given time t. At time $t_0 = 0$, probe vehicle $z = 0$ enters link 1. At this time step, two vehicles are observed on link 1 and five vehicles are observed on link 2 by video cameras C and D, that is, $x_1(t_0) = 2$ and $x_2(t_0) = 5$. Similarly, probe vehicle $z = 5$ enters link 1 at time t. At this moment, $x_1(t) = 2$ and $x_2(t) = 4$.

Probe vehicle $z = 8$ is worth mentioning, which enters link 1 at time t'. However, at time t'', this vehicle changes lanes. This can be seen in Figure 3.11 because there is an incomplete vehicle trajectory. During this 5-minute interval, 12 vehicles are counted by detector A, including two vehicles entering before probe vehicle z = 1, but excluding probe vehicle z = 8. Meanwhile, 13 vehicles are counted by detector B. This count includes seven vehicles before probe vehicle $z = 1$, but excludes probe vehicles 7-11, which have not yet departed from link 2.

Figure 3.12 illustrates the relationship between the actual trajectories extracted from the NGSIM dataset and the number of vehicles waiting in the "modeled" vertical queue. For vehicles 1 and 2, we plot the free-flow travel times (dashed lines), the waiting times (arrowed lines) and the experienced travel times (solid lines). Essentially, the experienced travel time = free-flow travel time + waiting time at the vertical queue, where the waiting time is determined by the number of vehicles in the queue and the capacity, as stated by Eq. (3.1).

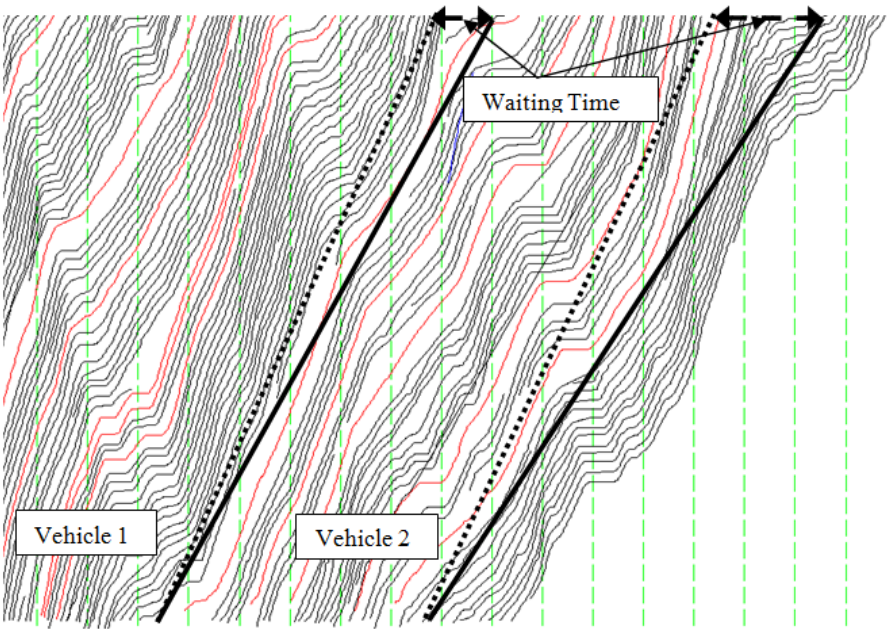


Figure 3.12: Waiting Times under Different Traffic Conditions

3.3.3 Model Validation

Two variants on the basic travel time distribution estimation approach presented here are investigated in this section. The first is lane-based; the second is link-based.

The examples require some additional notation to represent lane-specific parameters.

Z: number of probe vehicles;

n: index identifying a lane;

t_0^z: starting time for probe vehicle z;

n^z: starting lane number for probe vehicle z;

$x_m(t,n)$: number of vehicles on lane n at time t;

$\lambda_m^z\left(t_m^z,n\right)$: lane n specific number of vehicles behind bottleneck m;

$c_m(n)$: lane n specific discharge rate of bottleneck m;

$f_m^{net}(n)$: net flow rate from or to ramps by lane n;

$\theta(n)$: vehicle distribution rate from on-ramp to lane *n*;

$w_m^z(n)$: waiting time for probe vehicle *z* on bottleneck *m* on lane *n*;

$p_m^z(n)$: lane-based route-level end-to-end travel time for probe vehicle *z* through lane *n*;

The following procedure is used to calculate the lane-based travel time distribution.

For *z* = 1 to *Z* on the link

Obtain arrival time t_0^z and starting lane number n^z for each probe vehicle *z*.

Obtain the lane-based number of vehicles $x_m(t_0, \mathrm{n})$;

Obtain the lane-specific discharge rate $c_m(n)$;

Calculate net flow rate $f_m^{net}(n)$ from the on-ramp by applying $\theta(n) \times f_m^{net}$;

Calculate the number of vehicles behind bottleneck *m* $\lambda_m^z(t_m^z, n)$;

Calculate $w_m^z(n)$ based on $c_m(n)$ and $x_m(t_0, n)$;

Update the route-level end-to-end lane travel time $p_m^z(n)$.

End For

Output: Create the lane-based path travel time distribution based on $p_m^z(n)$

3.3.3.1 Lane-based Route-level End-to-end Travel Time Estimation

The distribution of the estimated route-level end-to-end travel times for each 5-minute interval, calculated over the 3 time periods with available data (4:00 p.m. to 4:15

p.m., 5:00 p.m. to 5:15 p.m. and 5:15 p.m. to 5:30 p.m.) are plotted in Figure 3.13 with the ground truth route-level end-to-end travel time obtained directly from the NGSIM data. As can be observed, the distribution of the estimated route-level end-to-end travel time is very close to that of the ground truth route-level end-to-end travel time. This demonstrates that our model is able to accurately estimate the route-level end-to-end travel time distribution.

3.3.3.2 Link-based Route-level End-to-end Travel Time Estimation

One common practice is to use link-based flow rates or density to estimate travel time reliability. We replace the lane-based variables in the previous approach with link-based variables $x_m(t_0)$ and c_m. That is, $x_m(t_0)$ is the existing number of vehicles on all the lanes on the link and c_m is the link discharge rate. The distribution of the estimated route-level travel time and true route-level end-to-end travel time for different time intervals are shown in Figure 3.14.

As it can be observed, the distribution of the estimated link-based route-level end-to-end travel times fails to capture the wide-spread distribution in the ground truth travel times. This is explained by the fact that link-based input variables would yield the same estimated route-level end-to-end travel times for those vehicles entering the link at the same time, regardless of their driving lanes.

In order to understand the extent and sources of the lane-by-lane travel time variation, we use the time period between 4:00 p.m. and 4:15 p.m. as an example. Figure 3.15 shows the lane discharge rate, the existing numbers of vehicles on the lane and the average true and estimated route-level end-to-end travel times for each lane for each 5-

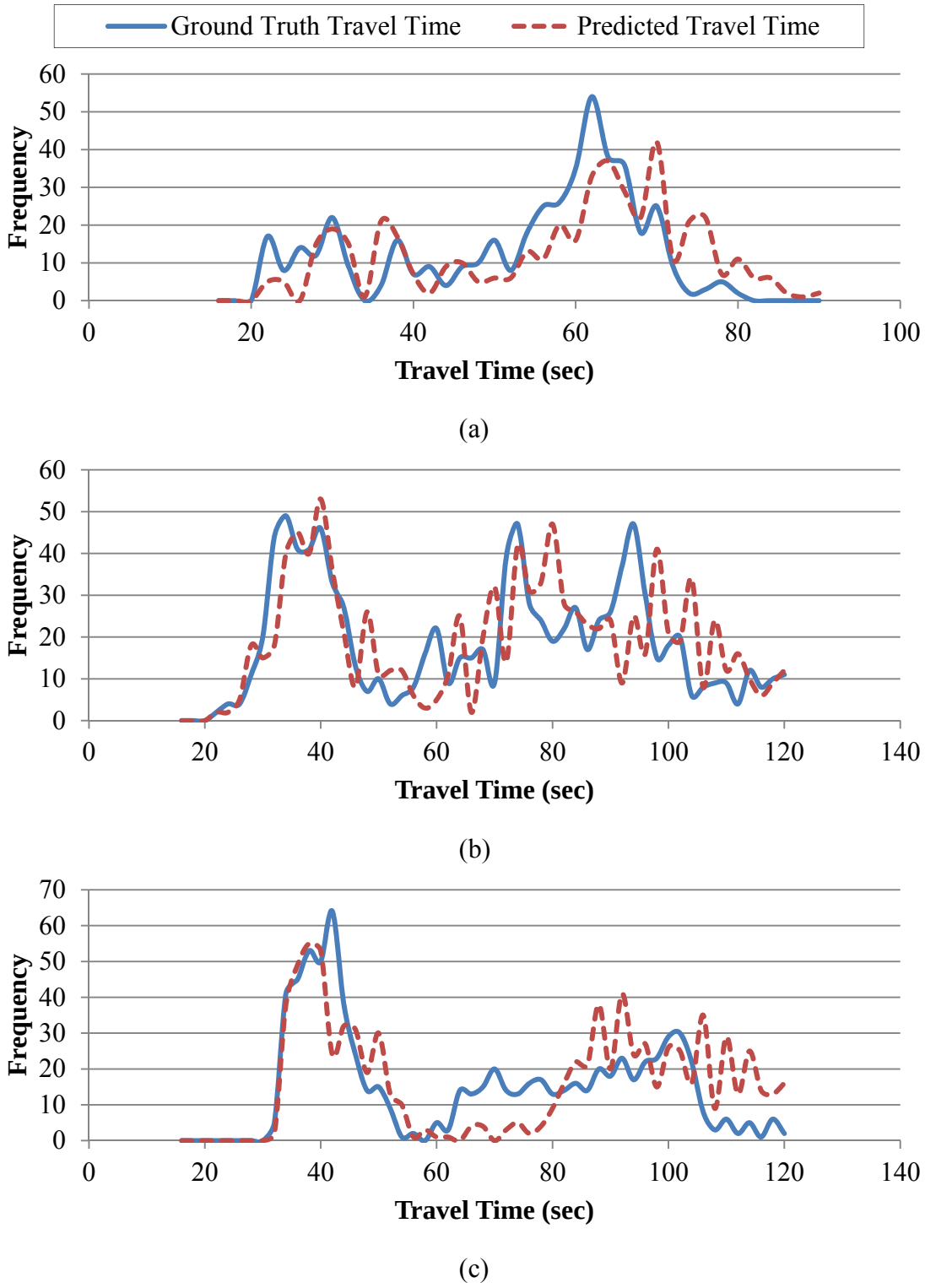


Figure 3.13: Lane-based Route-level Travel Time Distributions. (a) 04:00 p.m. – 04:15 p.m., (b) 05:00 p.m. – 05:15 p.m., (c) 05:15 p.m. – 05:30 p.m.

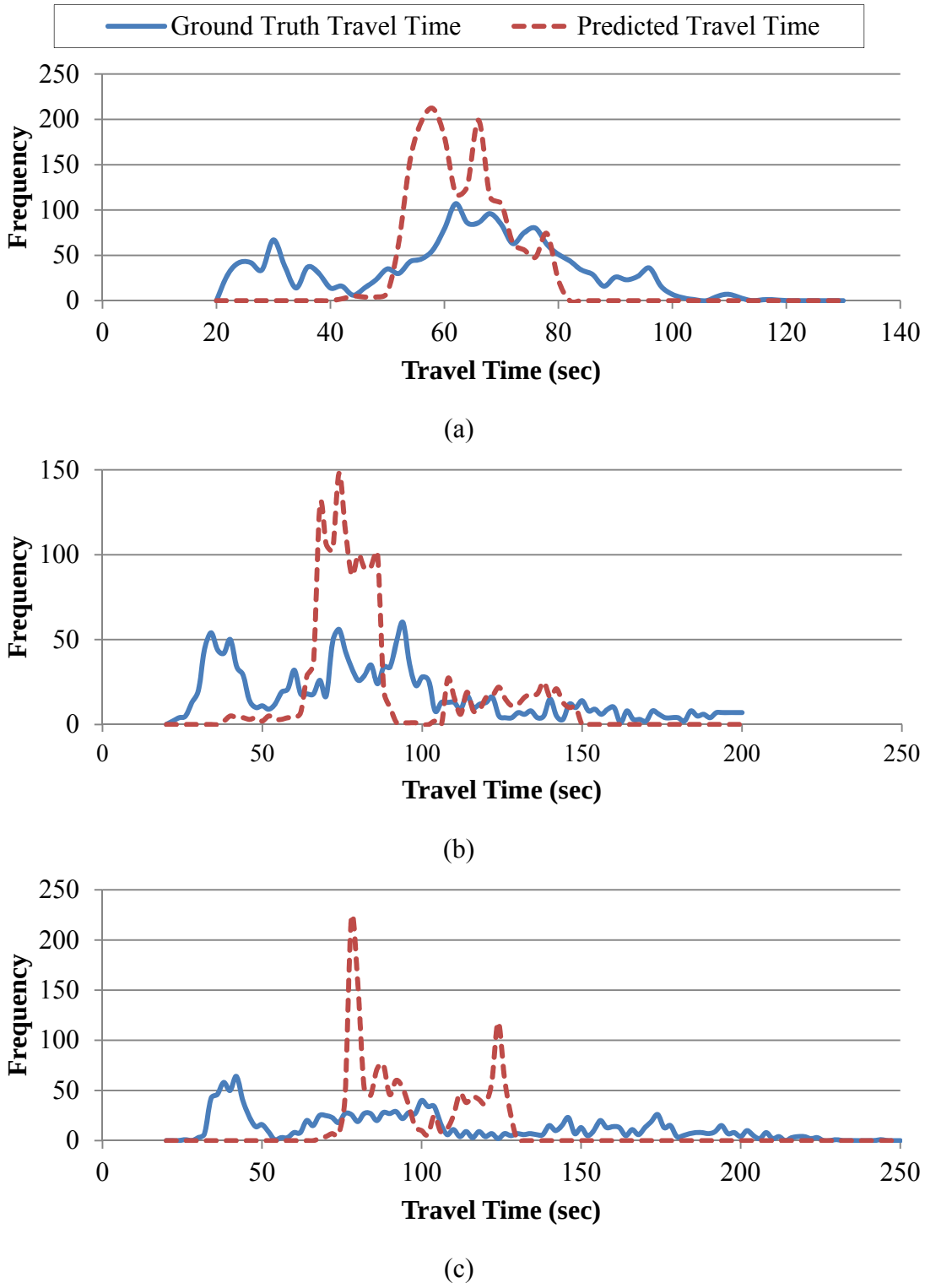


Figure 3.14: Link-based Route-level Travel Time Distributions. (a) 04:00 p.m. – 04:15 p.m., (b) 05:00 p.m. – 05:15 p.m., (c) 05:15 p.m. – 05:30 p.m.

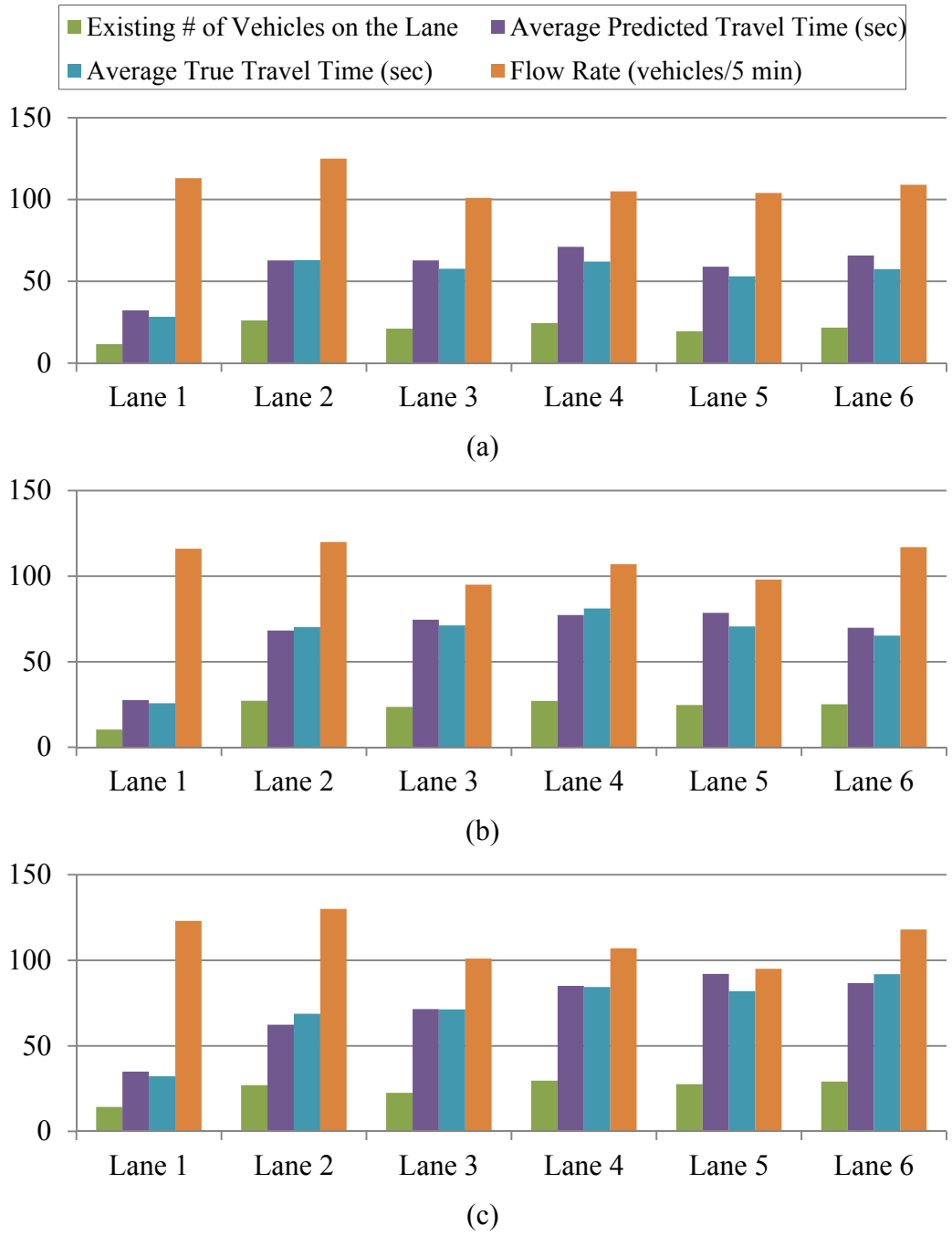


Figure 3.15: Lane-by-lane Travel Time Variability. (a) 04:00 p.m. – 04:05 p.m., (b) 04:05 p.m. – 04:10 p.m., (c) 04:10 p.m. – 04:15 p.m.

minute interval in the time period. The lane sequence is sorted by the true route-level end-to-end travel time.

Several observations can be made based on Figure 3.15. Lane 1 (HOV lane) has the lowest existing number of vehicles on the lane and has the lowest average route-level end-to-end travel time. The left-most lanes (lanes 1 and 2) usually have the highest discharge rates while lanes 3 and 4 usually have the lowest discharge rates. In most cases, lanes 3 and 4 have the highest average existing numbers of vehicles on the lane, as well as the highest average route-level travel times.

These observations imply that, due to the variation of the discharge rates and the number of vehicles on the lane, the route-level travel times also present strong lane-by-lane variations. As a result, we suggest using lane-based statistics to better quantify the travel time variability.

3.3.4 Estimation Error Sources

By comparing the estimated results with the NGSIM ground truth data, we can further uncover other possible sources of errors in the proposed travel time estimation model.

i. Aggregation errors: The link/lane discharge rates c_m and on-ramp flow rates used in the calculations are average flow rates over a certain time interval, e.g., 5-minute rates, while the existing number of vehicles on the link/lane $x_m(t_0)$ is an instantaneous value based on the entering time of a probe vehicle.

ii. Measurement errors: The number of vehicles on the lane observed by the video camera at time t_0 is assumed to be error-free. In fact, there are always

vehicle detection errors in NGSIM vehicle trajectory data associated with the underlying video recognition algorithm.

iii. Modeling errors associated with lane changing: Since the queue model incorporates the first-in-first-out principle, lane change behavior is not considered in the calculation. This will introduce two types of errors in the model:

a) The model may underestimate or overestimate the number of vehicles behind the bottleneck, $\lambda_m\left(t_m^z\right)$. For example, some vehicles will enter the lane (from the other lanes) before a probe vehicle reaches the bottleneck, or some vehicles originally counted in $x_m\left(t_0\right)$ on the current lane will leave to one of the adjacent lanes, corresponding to a lower value of $x_m\left(t_0\right)$.

b) When a probe vehicle changes lane from, for example, $n^z = 1$ to lane n', the discharge rate used in the calculation should be changed to the one associated with lane n'.

CHAPTER 4

TRAFFIC STATE ESTIMATION MODEL

To generate anticipatory and coordinated control and information supply strategies, intelligent traffic network management systems call for accurate and reliable estimation of time-varying traffic flow patterns. Essentially, any application of real-time and data-driven traffic decision support systems involves estimation of traffic states, e.g., microscopic states for detailed emission analysis and path-based travel time for personalized route guidance.

There are a number of surveillance techniques available for the traffic monitoring and management purposes, each with ability to collect and process real-time traffic data in specific types, including point, point-to-point and path measurements. In particular, point detectors, such as inductive loops, passive acoustic, passive infrared and microwave radar detectors, can directly detect flow, occupancy and time-mean speed at a single point or a short section (e.g., 20 ft). Automatic Vehicle Identification (AVI) techniques, such as Bluetooth readers, can capture the travel time of an individual vehicle traversing multiple but noncontiguous reader stations. In addition, Automatic Vehicle Location (AVL) technologies, such as Global Positioning System (GPS) and cellular telephone tracking, provide new possibilities for traffic monitoring to semicontinuously obtain detailed passing time and location information along individual vehicle trajectories, but how to

connect microscopic traffic state observations and the internal macroscopic traffic flow variables in a traffic estimator, such as link-based flow, density and speed, have remained as a challenging task.

In order to address the aforementioned challenge, this dissertation formulates an optimization problem. The objective function of this problem is to minimize the difference between measurements and estimated traffic states within a certain horizon length, and the constraints are building on traffic flow propagation, flow equivalent in merge and diverge, and time-dependent capacity.

This chapter is organized as follows. Section 4.1 presents the notation and the problem statement. Section 4.2 and 4.3 formulates the objective and inequality constraints of the optimization modeling approach, respectively. In Section 4.5, the measurements of middle point sensor and AVI data are introduced to the optimization modeling approach. In Section 4.6, the NGSIM dataset is used to verify the proposed estimation model. Section 4.7 summarizes the chapter.

4.1 Notation and Problem Statement

4.1.1 Notation

Before the detailed description of the proposed traffic estimation model, the notation and subscripts used in the mathematical formulation in this section are listed below:

t, t': Time index, $t = 0,1,\ldots,T$;

T: Length of estimation time horizon;

x: Length of estimation time horizon;

x_A: Location of upstream end of a link;

x_D : Location of downstream end of a link;

l : Length of a link;

m : Point sensor index, $m = \{1, \ldots, M\}$;

M : Number of point sensor locations;

p : AVI probe vehicle index, $p = \{1, \ldots P\}$;

P: Number of AVI probe vehicles;

q : GPS probe vehicle index, $q = \{1, \ldots Q\}$;

Q: Number of GPS probe vehicles;

$z(x,t)$: Observed cumulative flow count at position x and time t;

$n(x,t)$: Unknown cumulative flow count at position x and time t;

$q(x,t)$: Flow rate of position x and time t;

$k(x,t)$: Density of position x and time t;

k_j : Jam density, veh/mile;

w : Backward wave speed, mile/h;

v_f : Free-flow speed, mile/h;

$BWTT$: Backward wave travel time of a given link;

$FFTT$: Free-flow travel time of a given link.

4.1.2 Problem Statement

To calculate the emissions experienced by a vehicle entering a link at a specific time step, it is important to estimate the detailed vehicle engine operating parameters, i.e.,

instantaneous velocity and acceleration, when this vehicle moves along this link. In order to obtain these parameters, a macroscopic traffic flow model, namely Newell's simplified kinematic wave model, is adapted to estimate the dynamic traffic states, in terms of cumulative flow counts, in a link.

Without loss of generality, we consider a stretch of a freeway corridor with three links, numbered *a*, *b* and *c*, and an on-ramp and an off-ramp in between. As shown in Figure 4.1, four point detectors are located at both ends (i.e., *A* and *D*) of the corridor stretch, and entrance points at the on-ramp and off-ramp (B_o and C_f). Essentially, the traffic measurements at these four locations correspond to the space-time boundary conditions of the traffic estimation problem under consideration. For this closed network under consideration, the flow count or cumulative flow counts at each source and sink of the freeway corridor are assumed to be given. Where direct measurements at the boundary (e.g., ramps) are not available, one must resort to prior estimates of link counts (e.g., from historical flow counts) to ensure the estimation problem has a bounded solution space in the proposed least squares estimation model. A middle-point sensor (e.g., a loop detector) is placed at location x' of link *b*. In a more general form, we can

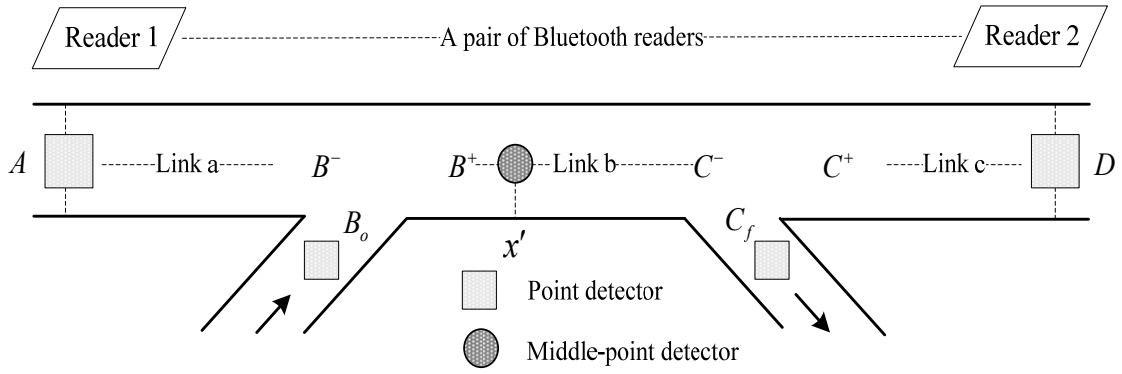


Figure 4.1: Layout of Freeway Corridor Segments

denote $m = \{1, \ldots, M\}$ as the point sensor index, and $z\left(x_m, t^{'}\right)$ represents the detected cumulative flow count at sensor *m* with location x_m and time instance *t*.

A pair of Bluetooth readers is installed at both ends (A and D) to record end-to-end travel time data with a certain sampling frequency (which is determined by the market penetration rate and identification rates of Bluetooth equipped drivers). We denote *h* as the AVI probe vehicle index, and there is a total of *H* probe vehicles identified from AVI sensors. For each probe vehicle, the Bluetooth readers located on the upstream end *A* and the downstream end *D* can directly capture its two timestamps t_A and t_D, respectively.

Conceptually, the traffic state estimation problem on a freeway corridor aims to reconstruct a complete picture of various traffic states, such as flow and density along the segment of interest, link-specific travel times, as well as queue length evolution profiles. In this problem, we are interested in estimating the cumulative flow counts $n(x,t)$ at any intermediate point *x* at time *t* on each link, which serve as a foundation to reconstruct the detailed vehicle trajectories.

Before constructing a complete optimization model with measurement equations, we first present the objective function and a series of constraints related to time-dependent flow propagation and flow redistribution around merges and diverges.

4.2 Objective Functions

The goal of the proposed estimation model is to minimize the absolute difference between the observed and unknown cumulative flow counts over an estimation horizon $t = 1, \ldots T$ and all point sensor locations, $m = 1, \ldots M$:

$$obj = \min \sum_{m=1}^{M} \sum_{t=0}^{T} \left| z(x_m, t) - n(x_m, t) \right| \tag{4.1}$$

Alternatively, we can also define an objective function that minimizes the squared difference between the observed and unknown flow volume at each time interval as follows:

$$obj = \min \sum_{m=1}^{M} \sum_{t=0}^{T-1} \left| z(x_m, t+1) - z(x_m, t) - n(x_m, t+1) + n(x_m, t) \right| \tag{4.2}$$

where $z(x_m, t+1) - z(x_m, t)$ and $n(x_m, t+1) - n(x_m, t)$ denote the observed and unknown flow counts between time intervals t and $t+1$, respectively.

4.3 Traffic Flow Balance and Flow Propagation Constraints

For a general traffic state estimation problem, its numerical solution to the PDEs is constrained by the characteristic wave of flux function. In the triangular flow-density relation, Newell's KW model represents the traffic state propagation of a link as a set of constraints by a forward wave and a backward wave.

As illustrated in Figure 4.2, the cumulative flow count at any intermediate location x of a homogeneous link can be described as a minimization function of a forward wave from upstream location x_A and a backward wave from downstream location x_D associated with a certain time lag, i.e.:

$$n(x,t) = \min \left(n(x_U, t - \frac{x - x_U}{v_f}), n(x_D, t - \frac{x_D - x}{w}) + k_j \times (x_D - x) \right) \tag{4.3}$$

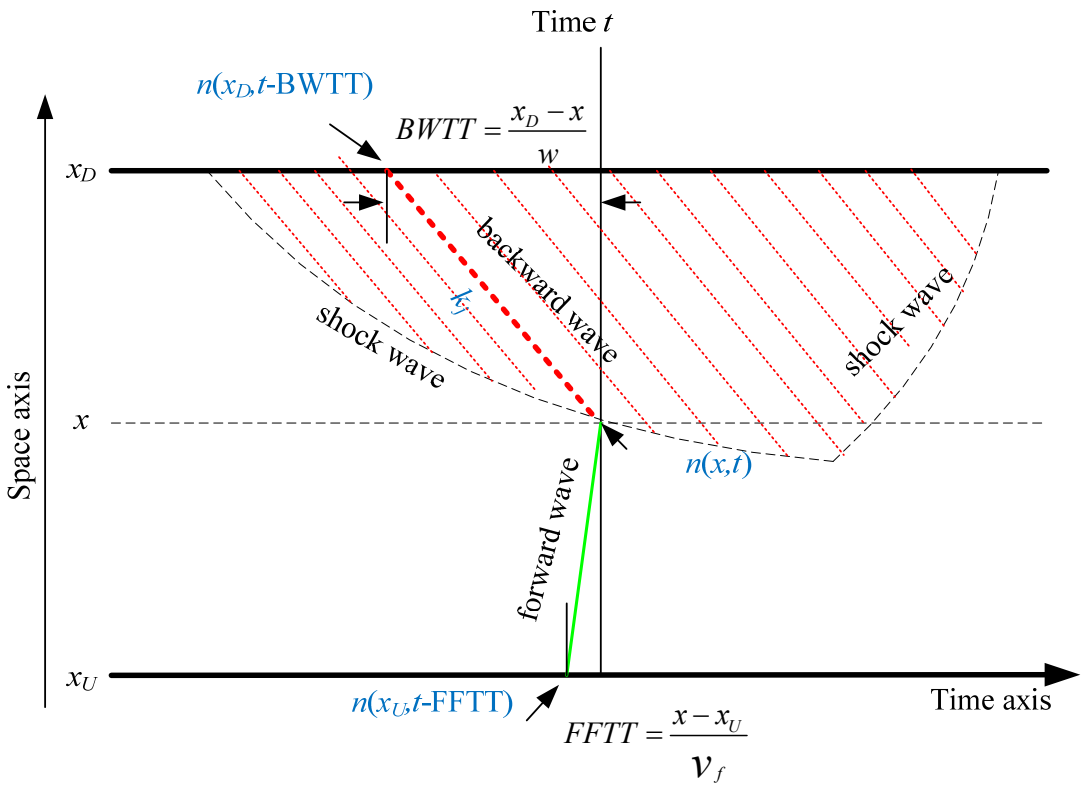


Figure 4.2: Illustration of Newell's Simplified Kinematic Wave Model on a homogeneous Freeway Segment, with Time Lags of Free-Flow Travel Time (FFTT) and Backward Wave Traversal Time (BWTT)

If we move location x extremely close to the upstream end, the cumulative arrival curve $n(x_U,t)$ at time t shall be less than or equal to the cumulative departure curve $n(x_D,t-BWTT)$ plus the maximum number of vehicles contained along the backward wave propagation line, i.e., $k_j \times l$, shown in constraint (4.4). Similarly, if location x is moved extremely close to the downstream end, inequality (4.3) reduces to inequality (4.5):

$$n\left(x_A,t\right) \le n\left(x_D,t-\frac{l}{w}\right)+k_j \times l \tag{4.4}$$

$$n\left(x_D,t\right) \le n\left(x_A,t-\frac{l}{v_f}\right) \tag{4.5}$$

$$n\left(B^{-},t\right)+n\left(B_o,t\right)=n\left(B^{+},t\right) \tag{4.6}$$

$$n\left(C^{-},t\right)-n\left(C_f,t\right)=n\left(C^{+},t\right) \tag{4.7}$$

Figure 4.3 illustrates scenarios with an on-ramp and an off-ramp. Following Ni's representation schema for Newell's KW model (Ni, 2004), we use superscripts "-" and "+" to indicate "slightly upstream of merge point" and "slightly downstream of merge point", respectively. Obviously, we have the following corresponding flow balance equations.

Furthermore, the maximum traffic flow at each location shall be constrained by the physical inflow/outflow discharge rates at that location, for example, $n(B^{-},t)-n(B^{-},t-1)\le cap(B^{-},t)$. That is, the cumulative flow count at that location shall be less than or equal to the maximum cumulative flow count for any time *t*:

$$n\left(B^{-},t\right)\le\sum_{t'=0}^{t}cap\left(B^{-},t'\right) \tag{4.8}$$

$$n\left(B_o,t\right)\le\sum_{t'=0}^{t}cap\left(B_o,t'\right) \tag{4.9}$$

$$n\left(B^{+},t\right)\le\sum_{t'=0}^{t}cap\left(B^{+},t'\right) \tag{4.10}$$

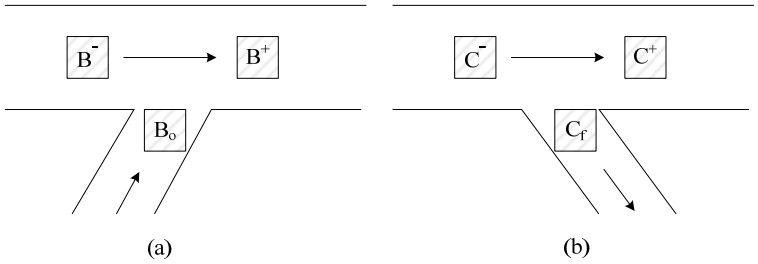


Figure 4.3: Flow Balance Constraints at a Merge and a Diverge

where $cap(x,t')$ denotes maximum unit vehicle count at location x at time interval t'. $cap(x,t)$ can be determined by various capacity allocation schemes that distribute the limited merge area capacity to flows on the mainline and on the on-ramp. For modeling simplicity, our model assumes the prespecified capacity on on-ramp link B_o as well as freeway merge points B^- and B^+, rather than endogenous variables that depend on the incoming flow rates. Interested readers are referred to a recent paper by Tampère et al. (2011) which describes a wide range of possible capacity allocation models for merges and diverges.

4.4 Measurements from Middle-point Sensors

Assuming a point detector is located at internal location x' of link b, as shown in Figure 4.1, we first examine vehicle count measurements from this detector. The objective function (4.3) can be extended to (4.11) by adding an additional component for the deviation between $z\left(x^{'},t\right)$ and $n\left(x^{'},t\right)$, where B_o and C_f correspond to on-ramp and off-ramp locations shown in Figure 4.1:

$$\begin{aligned}\min \quad obj = &\sum_{t=0}^{T}\left|z(x_A,t)-n(x_A,t)\right|+\left|z(x_D,t)-n(x_D,t)\right|+\sum_{t=0}^{T}\left|z(x',t)-n(x',t)\right| \\ &+\sum_{t=0}^{T}\left|z(B_o,t)-n(B_o,t)\right|+\left|z(C_f,t)-n(C_f,t)\right|\end{aligned} \tag{4.11}$$

To ensure the traffic flow propagation constraint (4.3) is met, the following two inequalities map the traffic state of location x' to the local boundary condition (e.g., B^+, C^-) of link b:

$$n(x',t) \le n\left(C^-, t - \frac{d(x',C^-)}{w} \right) + k_j \times d(x',C^-) \tag{4.12}$$

$$n(x',t) \le n\left(B^+, t - \frac{d(x',B^+)}{v_f} \right) \tag{4.13}$$

where $d(x',C^-)$ and $d(x',B^+)$ are the distances between x' and C^-, and x' and B^+, respectively.

4.5 Measurements from AVI/GPS Readers

Figure 4.4 illustrates how AVI or particularly Bluetooth readers detect the timestamp information of an individual probe vehicle. If the Bluetooth readers are located on the boundary of the freeway stretch, we can directly capture two time instances t_A and t_D when the vehicle passing the upstream end A and the downstream end D. If no AVI sensors are placed on points B and C, two important timestamps are not directly observed

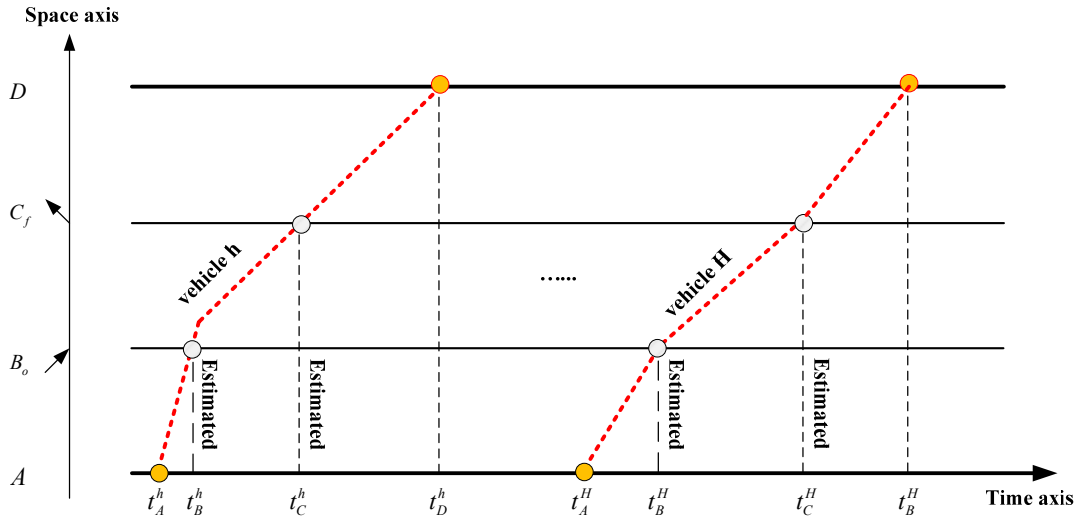


Figure 4.4: Travel Timestamp Measurements Available from AVI Probe Vehicles

and should be estimated: t_B at the on-ramp, and t_C at the off-ramp. The estimation of timestamps t_B and t_C requires some prior knowledge of the travel time on the corresponding segments, such as link *a* and link *b*.

Using a cumulative flow count-based representation and under a First-In-First-Out (FIFO) assumption, we can explicitly model the AVI measurements into constraints. Essentially, the cumulative flow count at downstream end *D* is contributed by two components: (i) the total inflow of upstream end *A* and (ii) the total net flow changes on ramps inside the corridor.

Let us consider a simple case, where the on-ramp inflow rate and off-ramp outflow rate are assumed to be the same. This implies that the net flow inside the freeway corridor equals to zero, and the cumulative flow count at upstream end *A* equals to that at downstream end *D* with a travel time lag $t_D - t_A$:

$$n\left(x_A, t_A\right) = n\left(x_D, t_D\right) + \varepsilon \tag{4.14}$$

where ε denotes the measurement error of AVI data.

In a general case where the on-ramp and off-ramp flow rates are different, and the total net flow before vehicles passing ramps shall be considered:

$$n\left(x_A, t_A\right) + n\left(x_{B_o}, t_B\right) - n\left(x_{C_f}, t_C\right) = n\left(x_D, t_D\right) + \varepsilon' \tag{4.15}$$

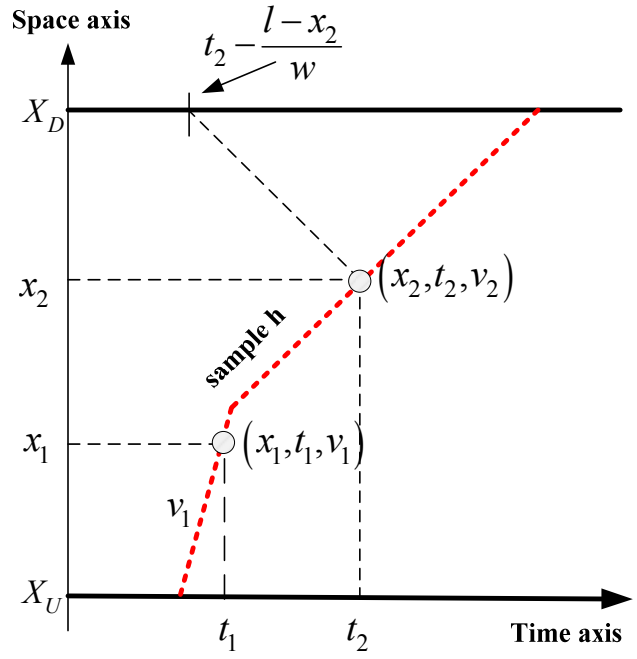
where $n(B_o, t_B)$ is the total inflow at on-ramp from *t=0 to* t_B, and $n(C_f, t_C)$ is the total outflow at off-ramp from $t = 0$ *to* t_C. ε' is the combined errors that include the estimation error for unknown intermediate time stamps t_B and t_C, as well as the measurement error

of Bluetooth sensors . Interested readers are referred to a paper by Malinovskiy et al. (2010) on the systematical investigation of Bluetooth-related travel time estimation error sources. If the information about the distribution or magnitude of the error term ε' is not available, we can utilize an ordinary least square framework to minimize the sum of the squares of the errors. Thus, the following term can be added to the existing objective function:

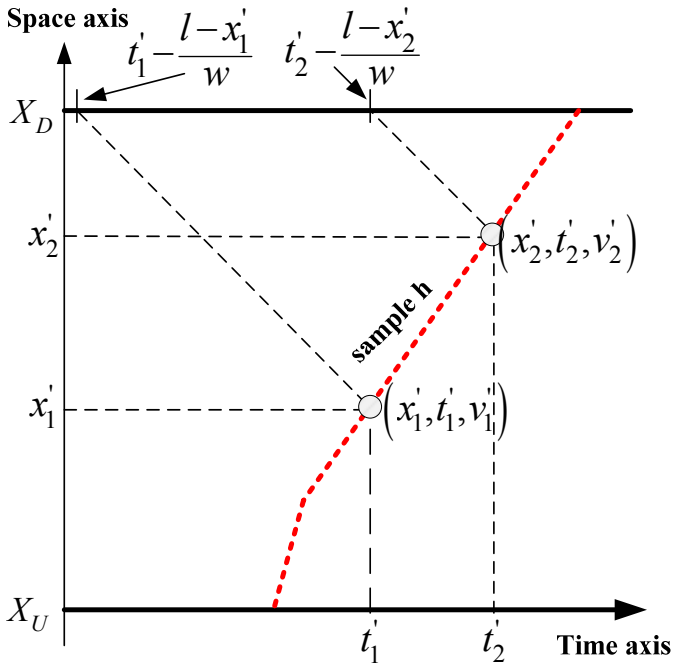
$$\sum_{h=1}^{H}\left|n\left(x_A,t_A^h\right)+n\left(x_{B_o},t_B^h\right)-n\left(x_{C_f},t_C^h\right)-n\left(x_D,t_D^h\right)\right| \tag{4.16}$$

Similarly, by explicitly representing the individual vehicle trajectory in terms of cumulative flow counts along the trajectory (Deng, Lei and Zhou, 2013), we can utilize the semicontinuous vehicle trajectory data from Global Position System (GPS) in the proposed optimization framework. Assuming a fixed sampling interval, samples can be captured for each probe vehicle along a link. For each sample *i*, the instantaneous travel speed v_i, the time stamp t_i and the location x_i are recorded. Figure 4.5 shows an example of two recorded GPS samples when a probe vehicle moves along a link.

As shown in Figure 4.5-(a), if v_1 at the first sample point is the free-flow speed while v_2 at the second sample point is not, it can be assumed that this probe vehicle travels at the free-flow speed until it meets the end of the queue propagating from the downstream and then travels at the reduced speed v_2. In this case, the departure time for this probe vehicle at the upstream boundary x_U is $t_1-\frac{x_1}{v_f}$. Thus, the cumulative arrival count number for this vehicle at x_U would be $z\left(x_A,t_1-\frac{x_1}{v_f}\right)$. Since the second sample of this



(a)



(b)

Figure 4.5: Measurements Available from GPS Probe Vehicles

vehicle indicates that this vehicle is inside the congested section of the link, according to Eq. (4.3), the associated downstream cumulative flow count would be $z\left(x_D, t_2 - \frac{l - x_2}{w}\right)$.

Meanwhile, the number of vehicles ahead of the probe vehicle at location x_2 can be derived by using the jam density and the segment length, that is, $\left(l - x_2\right) \times k_j$. With all this information, we can form a constraint that relates the probe vehicle's arrival count number to its departure count number. This constraint is shown as Eq. (4.17):

$$z\left(x_A, t_1 - \frac{x_1}{v_f}\right) = z\left(x_D, t_2 - \frac{l - x_2}{w}\right) + \left(l - x_2\right) \times k_j \tag{4.17}$$

If both observed speed $v_1^{'}$ at the first sample point and $v_2^{'}$ at the second sample point are not the free-flow speed (Figure 4.5-(b)), it can be assumed that both sample points are observed in the queuing/congested area of the link. In this case, according to Eq. (4.3), we can derive the corresponding departure flow count number at time $t_1^{'}$ and $t_2^{'}$. The derived departure flow count numbers corresponding to these two time stamp are $n\left(x_D, t_1^{'} - \frac{l - x_1^{'}}{w}\right)$ and $n\left(x_D, t_2^{'} - \frac{l - x_2^{'}}{w}\right)$, respectively. During this time period, $t_2^{'} - t_1^{'}$, the number of vehicles discharged is $\left(t_2^{'} - t_1^{'}\right) \times q_j$, where q_j is the queue discharge rate. This can be rewritten as $\frac{\left(x_2^{'} - x_1^{'}\right)}{w} \times q_j$ or $\left(x_2^{'} - x_1^{'}\right) \times k_j$. Thus, we have a new constraint:

$$n\left(x_D, t_2^{'} - \frac{l - x_2^{'}}{w}\right) = n\left(x_D, t_1^{'} - \frac{l - x_1^{'}}{w}\right) + \left(x_2^{'} - x_1^{'}\right) \times k_j \tag{4.18}$$

4.6 Optimization Model Implemented for Traffic Estimation Problem

The proposed optimization model for the traffic state estimation model is investigated in this section with the NGSIM dataset. The value of information by various sources of measurements is also evaluated.

4.6.1 Data Set

In this section, the proposed model is applied to a NGSIM vehicle trajectory data set to estimate traffic states on a freeway segment. The vehicle trajectory data with 0.1-s resolution are collected between 5:15 PM – 5:30 PM on April 13, 2005. The vehicle trajectory of lane 6 is shown in Figure 4.6, where an on-ramp is connected 440-ft upstream of the freeway segment. To meet the homogenous link requirement, we select 800-ft upstream of the freeway segment as the upstream sensor location, which leads to

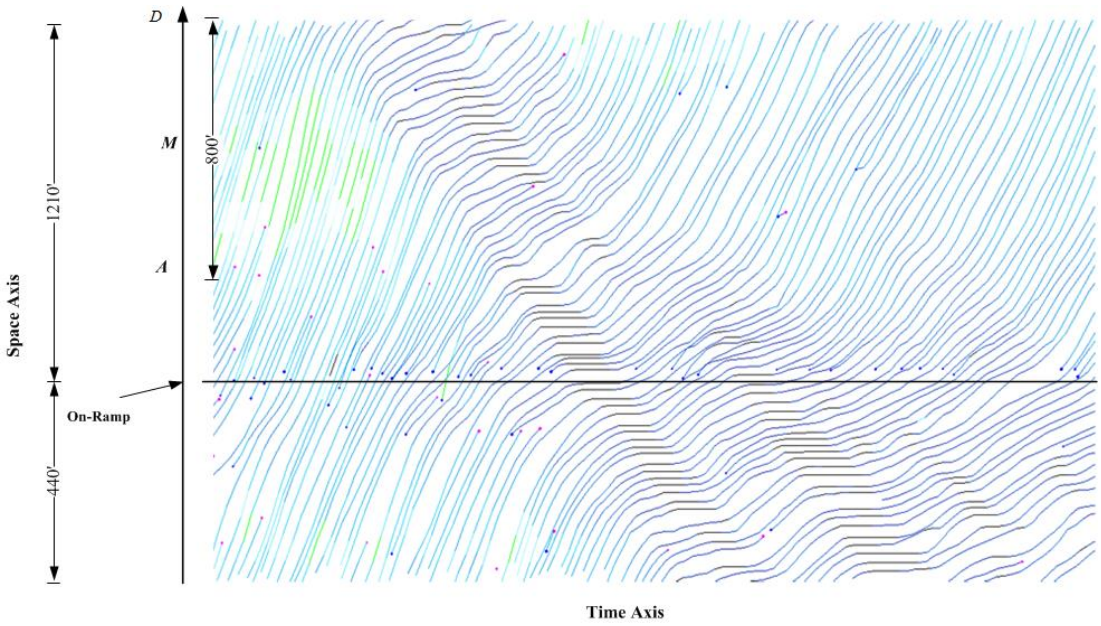


Figure 4.6: The Stretch of Northbound I-80 and the Vehicle Trajectory Data of Lane 6 at 5:15-5:30 PM

an 800-ft link between upstream and downstream sensors, marked as ‘A’ and ‘D’, respectively.

Figure 4.7 illustrates the detailed layout of the analysis time and space horizon. The first 5 minutes (5:15 – 5:20 PM) is the warm-up period to ensure complete vehicle trajectories are observed at *A*, *M* and *D*. The next 10 minutes (5:20 – 5:30 PM) is the estimation horizon.

This section mainly focuses on how to construct an optimization model to estimate traffic states on the homogenous link using different data sources. To apply the proposed traffic state estimation model in the real-world NGSIM data set, we make the following enhancements. (i) It does not require prior information on cumulative flow counts, and the objective function aims to minimize the absolute difference between the observed and unknown cumulative flow volume at each sampling time interval. (ii) We directly use two inequality constraints to incorporate the minimization function (4.3) that represents Newell’s three detector model. (iii) The new model also adds additional inequality constraints for nonnegative and maximum flow rates. The Microsoft Solver Foundation is

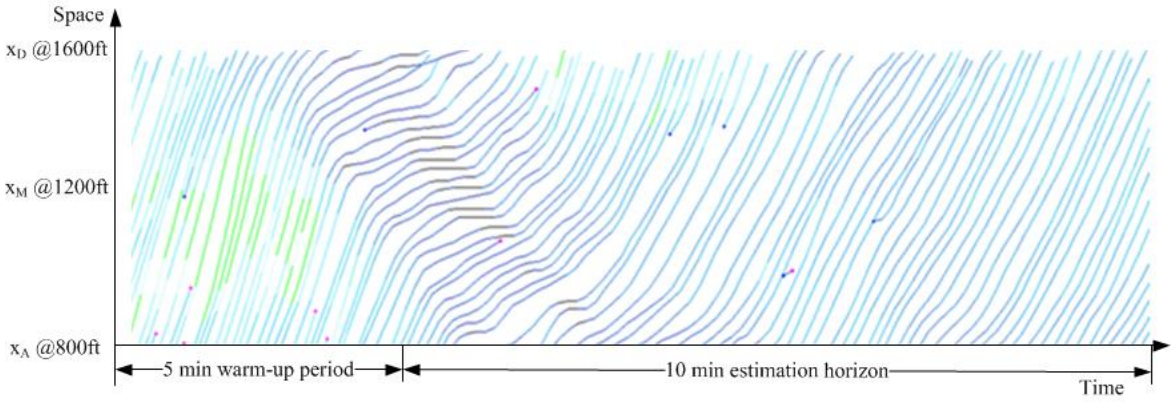


Figure 4.7: Layout of Analysis Time and Space Horizon

a C# optimization library that provides very flexible data and programming interfaces to represent time lags between variables (e.g., free-flow travel time and backward wave travel time in our case).

By locating virtual loop detectors at upstream location A and downstream location D, we first generate the corresponding cumulative flow count measurements at a 30-second sampling rate. To meet the FIFO constraints on homogeneous segments required by Newell's KW model, the raw vehicle position data from different lanes need to be aggregated to generate link-based flow counts. In particular, as shown in Figure 4.8, significantly higher average speed is observed in lane 1 (HOV lane) while the other 5 lanes share similar speed profiles. In addition, most vehicles traveling on lane 1 will remain in this lane. Thus, only the traffic flow counts from lanes 2-6 are aggregated as the link flow rates. We further consider three hypothetical data sources in the following experiments:

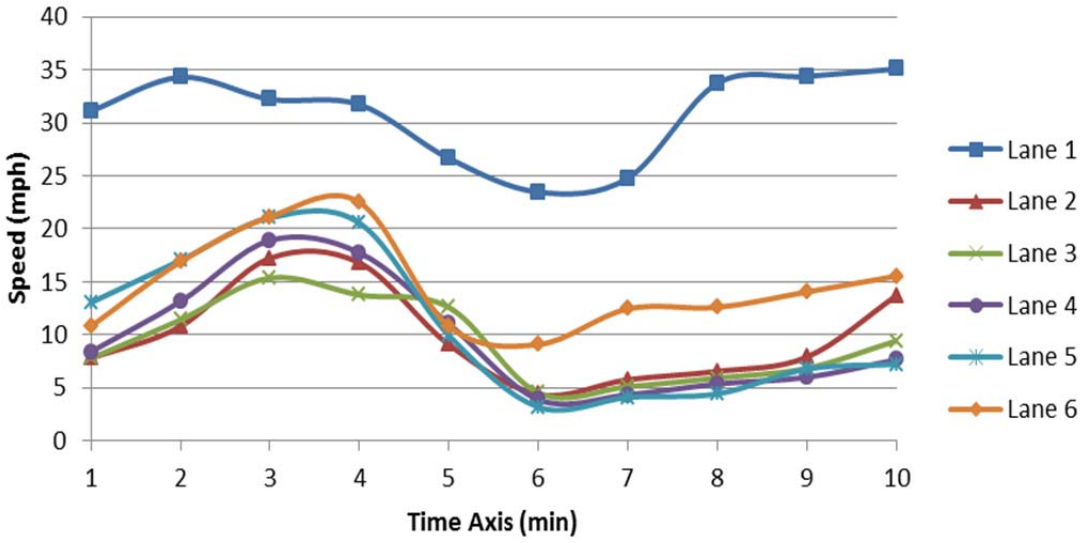


Figure 4.8: Lane-by-lane Time-dependent Speed on NGSIM Data Set

i. Middle sensor: A point detector is located at the middle-point of the link (marked as "*M*" in Figure 4.7).

ii. AVI data: Two AVI readers at locations A and D generate end-to-end travel time measurements, for a given market penetration rate. Due to complex lane changing behavior on the segment, the FIFO constraint may not be satisfied. In order to reduce sampling errors, the travel time obtained from the AVI data are aggregated at a fixed time interval, e.g., every 1 minute.

iii. GPS data: Assuming a fixed time sampling interval (e.g., 15 seconds), vehicle location samples can be collected after each probe vehicle passes the upstream boundary. Each data record *i* includes the time stamp t_i and the location x_i.

4.6.2 Test Cases

A number of test cases shown in Table 4.1 are used to investigate the impacts of the various sources of sensor data on the estimation performance in the real-word NGSIM data set. First, variables to be estimated are cumulative flow counts $n(x,t)$ at different locations and time-dependent jam density $k_{j\tau}$ at each time interval τ. The case-specific objective functions and constraints are described in the following section.

Table 4.1: Test Case Configuration

Case no	Short Name	Measurements	Objective Functions
1	base case	*x-min flow rates from the upstream and downstream sensors*	obj_{base}
2	with middle sensor	+ *x-min flow rates from middle sensor*	$obj_{base} + obj_{MS}$
3	with AVI data	+*AVI data from upstream to downstream*	$obj_{base} + obj_{AVI}$
4	with GPS	+ *GPS data*	$obj_{base} + obj_{GPS}$

4.6.2.1 Base Case Objective Function

The base case objective function that considers only upstream and downstream pointer sensors is:

$$obj_{base} = \sum_{t=0}^{T} \left| z\left(x_A - t\right) - n\left(x_A - t\right) \right| + \left| z\left(x_D - t\right) - n\left(x_D - t\right) \right| \qquad (4.19)$$

4.6.2.2 Objective Function Considering Middle-point Sensor

If the middle-pointer sensor data set at location M available, then the following objective function is added into Eq. (4.19):

$$obj_{MS} = \sum_{t=0}^{T} \left| z\left(x_M - t\right) - n\left(x_M - t\right) \right| \qquad (4.20)$$

4.6.2.3 AVI Measurement Data

As shown in Figure 4.9, we can directly capture two time instances t_A^p and t_D^p when vehicle p passes the upstream end A and the downstream end D. We further calculate aggregate segment travel time B_t for each departure time t, leading to the following objective function:

$$obj_{AVI} = \sum_{t=0}^{T} \left| n\left(x_A, t\right) - n\left(x_D - B_t\right) \right| \qquad (4.21)$$

4.6.2.4 GPS Measurement Data

Recall in Figure 4.8, all the lanes under consideration in this test have a low speed of below 25 miles per hour consistently throughout the analysis time horizon, so we only

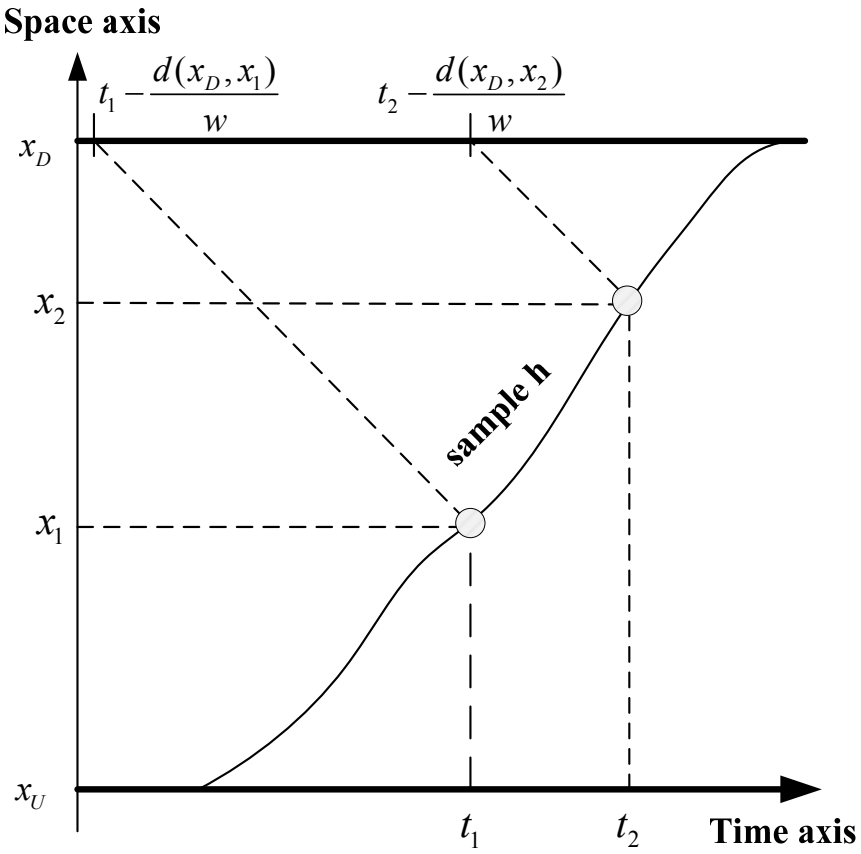


Figure 4.9: Measurements Available from GPS Probe Vehicles

consider how to include congested GPS samples in the optimization model.

Assume the GPS probe vehicle is traveling in the queuing/congested area of the link and its movement is constrained by the downstream traffic through backward wave. That is, the departure flow count numbers corresponding to these two time stamps t_1 and t_2 are $n\left(x_D, t_1 - \frac{d(x_D, x_1)}{w}\right)$ and $n\left(x_D, t_2 - \frac{d(x_D, x_2)}{w}\right)$, respectively, and we can relate these two data points as they share the same cumulative arrival count number since they are obtained from the same GPS probe:

$$n\left(x_D, t_1 - \frac{d\left(x_D, x_1\right)}{w}\right) + d\left(x_D, x_1\right) \times k_{j\tau} + \varepsilon_1 = n\left(x_D, t_2 - \frac{d\left(x_D, x_2\right)}{w}\right) + d\left(x_D, x_2\right) \times k_{j\tau} + \varepsilon_2 \tag{4.22}$$

By rearranging and combining the error terms, we will have the following equation:

$$n\left(x_D, t_2 - \frac{d\left(x_D, x_2\right)}{w}\right) = n\left(x_D, t_1 - \frac{d\left(x_D, x_1\right)}{w}\right) + d\left(x_2, x_1\right) \times k_{j\tau} + \varepsilon' \tag{4.23}$$

We now add the following objective function for GPS sample q= 1 to Q:

$$obj_{GPS} = \sum_{q=1}^{Q} \left| n\left(x_D, t_2 - \frac{d\left(x_D, x_2\right)}{w}\right) - n\left(x_D, t_1 - \frac{d\left(x_D, x_1\right)}{w}\right) - d\left(x_2, x_1\right) \times k_{j\tau} \right| \tag{4.24}$$

4.6.2.5 Constraints

- Nonnegative and Nondecreasing Constraints for the Cumulative Counts on the Boundaries

The most essential constraints the cumulative counts to be observed are nonnegative, that is:

$$n\left(x_D, t\right) \geq 0 \tag{4.25}$$

$$n\left(x_A, t\right) \geq 0 \tag{4.26}$$

and nondecreasing, that is, for any $t_2 > t_1$, we have:

$$n\left(x_D, t_2\right) \geq n\left(x_D, t_1\right) \tag{4.27}$$

$$n\left(x_A, t_2\right) \geq n\left(x_A, t_1\right) \tag{4.28}$$

- Max Flow Constraints

Another important set of constraints that must be observed are the max change of the cumulative vehicle counts during a specific time period at the boundaries. The max flow changes per estimation time interval or per second cannot exceed some maximum possible instantaneous flow rate or link capacity per hour. In this experiment, these values are extracted from the NGSIM data. The maximum flow changes per second is set to 5 vehicles on the 5 lanes of interest. Accordingly, the max flow constraints are formulated as follows:

$$n\left(x_D, t+10\right) - n\left(x_D, t\right) \leq 5 \tag{4.29}$$

$$n\left(x_A, t+10\right) - n\left(x_A, t\right) \leq 5 \tag{4.30}$$

- Upstream and Downstream Flow Constraints

A set of inequality constraints are used to represent the three-detector model on this homogeneous segment with corresponding cumulative vehicle counts at upstream location *A* and downstream location *D*:

$$n\left(x_D, t\right) \leq n\left(x_A, t - \frac{d\left(x_D, x_A\right)}{w}\right) + d\left(x_D, x_A\right) \times k_{j\tau} \tag{4.31}$$

$$n\left(x_D, t\right) \leq n\left(x_A, t - \frac{d\left(x_D, x_A\right)}{v_f}\right) \tag{4.32}$$

If we also consider the middle point sensor at location *M*, we can divide this segment into two subsegments: *A* to *M* and *M* to *D*. In this way, Eqs. (4.31) and (4.32) are then expanded to the following inequality constraints:

$$n\left(x_M,t\right)\le n\left(x_D,t-\frac{d\left(x_D,x_M\right)}{w}\right)+d\left(x_D,x_M\right)\times k_{j\tau} \tag{4.33}$$

$$n\left(x_D,t\right)\le n\left(x_M,t-\frac{d\left(x_D,x_M\right)}{v_f}\right) \tag{4.34}$$

$$n\left(x_A,t\right)\le n\left(x_M,t-\frac{d\left(x_M,x_A\right)}{w}\right)+d\left(x_M,x_A\right)\times k_{j\tau} \tag{4.35}$$

$$n\left(x_M,t\right)\le n\left(x_A,t-\frac{d\left(x_M,x_A\right)}{v_f}\right) \tag{4.36}$$

4.6.3 Estimation Results

The constructed model uses a 0.1-second temporal resolution over a 10-minute estimation time horizon, as the first 5 minutes are used for the warm up period. A laptop with an Intel Core i5-2430M processor and 8GB memory is used to solve these estimation models. Table 4.2 lists the number of variables, the number of constraints and the running time for each model. Overall, the model size is relatively large but the solution time is still reasonable for this high-fidelity estimation problem.

Table 4.2: Optimization Model Summary

Case no	# of Variables	# of Constraints	Computation Time (sec)
1	13073	43876	25.8
2	19612	68814	39.0
3	13375	44027	25.3
4	13225	43952	28.6

As shown in Figure 4.7, different time intervals have varying density along the same backward wave. Thus, we model the jam density as time-dependent variables for each 2-minute time interval (5 time intervals during the estimation time horizon) at the downstream boundary, and estimation results will not only report final cumulative flow counts but also the time-dependent jam density. For example, the estimated time-dependent jam density profile under case 1 is shown in Figure 4.10. In this study, we use a constant backward wave speed of 12 mph, but it should be noticed that variations in backward wave speed w_b have also received attention in a number of recent studies (e.g., Kim and Zhang, 2008).

The link density is used as the key indicators of estimation performance. The mean absolute percentage error (MAPE), which represents the average of the normalized absolute error, is used to compare the benchmark and estimated value at each time step (0.1 seconds). Table 4.3 compares the estimation quality under different degrees of data

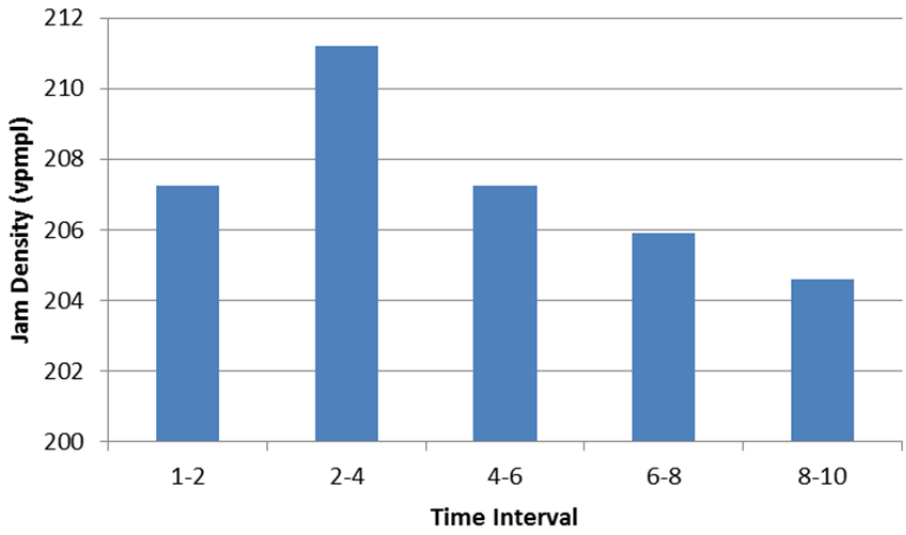


Figure 4.10: Estimated Time-dependent Jam Density

Table 4.3: MAPE of Estimated Density per Second under 30-second Flow Counting Interval

Case No	Density	
	MAPE	Relative error reduction w.r.t base case
1: base case	14.80	--
2: with middle sensor	13.63	7.9%
3: with AVI data (5% market penetration rate)	14.56	1.6%
4: with GPS (20% market penetration rate)	10.55	28.72%

availability, and the results show that additional sensor data generally lead to reduced estimation errors. Figure 4.11 shows the detailed profiles of estimated density vs. ground truth density in each case.

We are also interested in examining the impact of different AVI or GPS market penetration rates of the estimation results. Each set of the following experiments is repeated 5 times to reduce possible random sampling errors. Because the relative value of additional data is highly dependent on the base line estimation quality (provided by boundary sensors), we construct various base cases below under different flow counting intervals at the boundary sensors, ranging from 30 seconds to 2.5 minutes.

Figure 4.12 shows that, without GPS data, a finer flow counting resolution (i.e., 30 seconds) provides less estimation errors compared to the 1-minute counting interval. Adding more GPS data with sampling rates higher than 5% into the estimation model can significantly improve the estimation quality, under both 30-second and 1-minute flow counting intervals. The final estimation quality with GPS data is similar under different flow counting intervals, which might indicate that the overall estimation quality of internal traffic states in the latter two cases is mainly determined by the GPS sample rate, with less dependence on the boundary sensor counting interval.

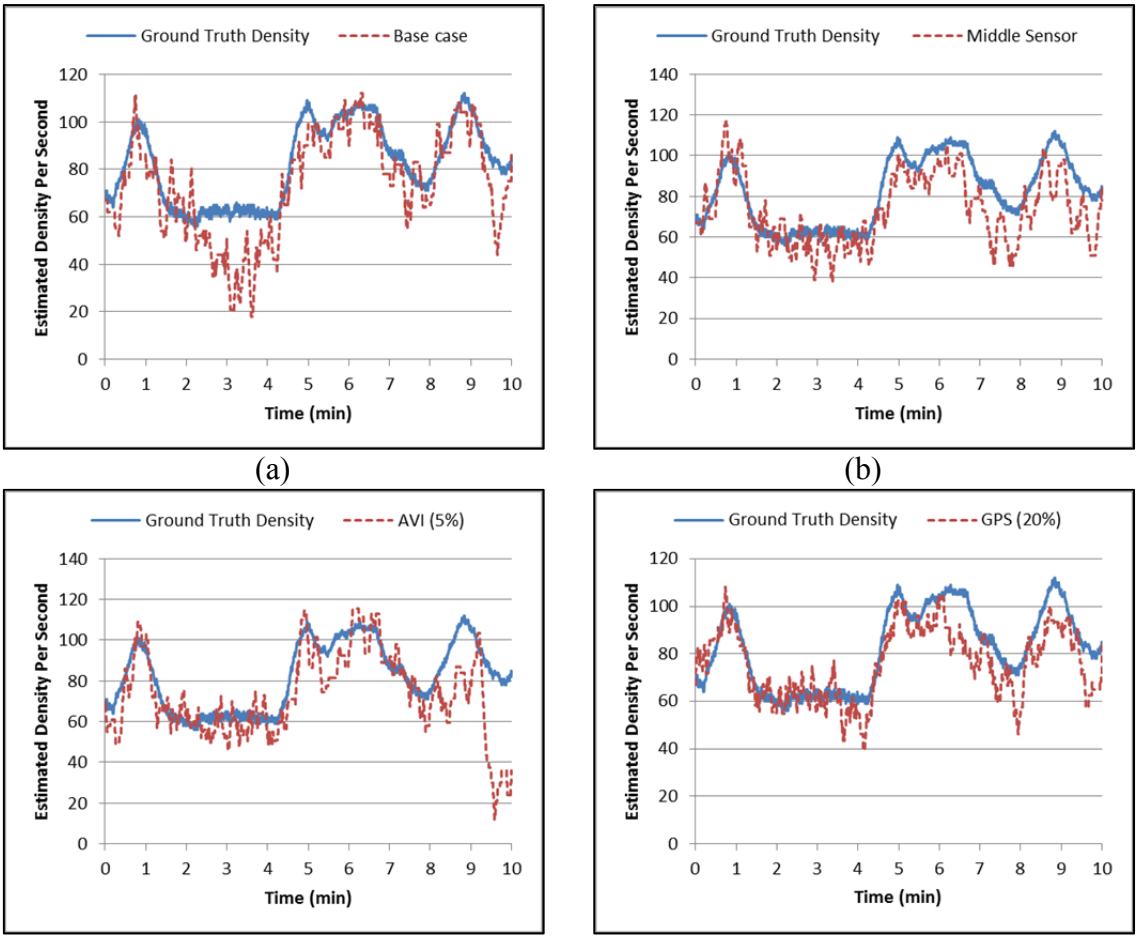


Figure 4.11: Estimated Density vs. Ground Truth Density.

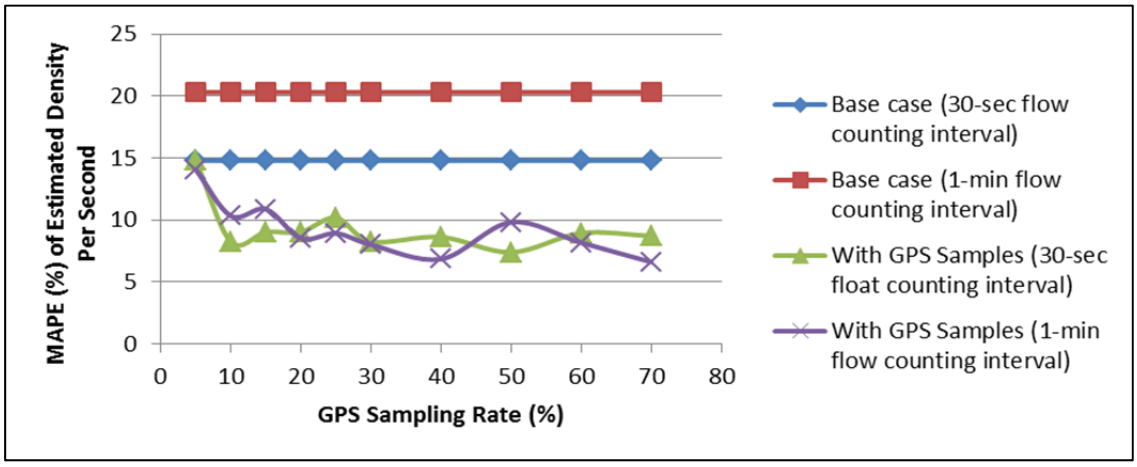


Figure 4.12: Impacts of GPS Samples on Estimation Quality under Different Flow Counting Intervals

As shown in Figure 4.13, adding a limited number of AVI samples shows meaningful estimation quality improvement only when the base line estimation quality is relatively poor, specifically under the relatively longer flow counting intervals of 2 and 2.5 minutes. Given a finer 30-second or 1-minute flow counting resolution at boundary sensors, the existing traffic state estimates are already sufficiently accurate and adding AVI samples does not show significant or even positive quality improvement. By examining the detailed cause of the potential negative impact of AVI sensors, we find that Eq. (4-21) imposes strict FIFO assumptions through aggregated travel times B_t for the same aggregation time interval t, which might not be consistent with the observed flow counts at a finer resolution.

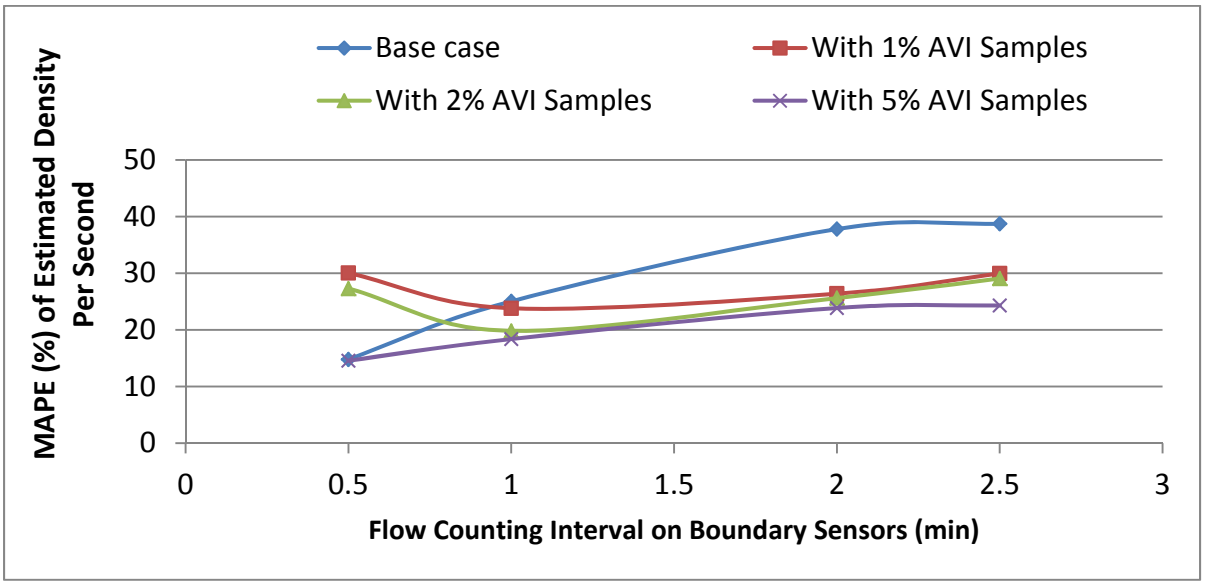


Figure 4.13: Impacts of AVI Samples on Estimation Quality under Different Flow Counting Intervals

4.7 Model Complexity Reduction for Real-time Online Applications

It is necessary to reduce the model complexity if we want to extend the proposed method for online traffic state estimation applications. As indicated in the numerical experiments, the computation performance of the proposed traffic state estimation model is largely dependent on the number of the system variables in a model. Thus, one potential solution to improve the computation efficiency is to reduce the number of the system variables. In the numerical experiment, the estimation time interval is set to 0.1 second. Thus, for an estimation horizon of 10 minutes, the total number of system variables for the base model considering only upstream and downstream boundary is 12,000. If we use more aggregated intervals, e.g., 6 seconds, the number of system variables would reduce to 200. However, the level of aggregation must be carefully selected to maintain the estimation quality.

In addition, we should also take the time-varying traffic pattern into consideration if the proposed traffic state estimation model would be applied to real-time applications. To provide accurate real-time estimates, it is important to adjust the length of the estimation horizon so that no noticeable variation can be observed during the selected estimation horizon. For example, during the nonpeak hours, the traffic may operate at a free-flow or near free-flow state with small traffic state changes. In this case, it is appropriate to adapt a relatively long estimation horizon, e.g., 15 minutes. However, significant traffic state variation is present during the peak hours or whenever due to the accidents. In this case, we should adapt an estimation horizon, e.g., 30 seconds, to capture the change of traffic state in a short time interval. If combined with other traffic monitoring systems, it is also

possible to implement an auto-adaptive system that can automatically adjust the estimation horizon, based on the traffic conditions under surveillance.

4.8 Summary

While there is a growing body of work on the estimation of traffic states from different sources of surveillance techniques, much of the prior work has focused on single representations, including loop detectors, GPS data, AVI tags and other forms of vehicle tracking. This study investigated cumulative flow count-based system modeling methods that estimate macroscopic traffic states with heterogeneous data sources on a freeway segment. We further adapt the proposed model to a real-world data set and examine the relative value of information under different GPS and AVI market penetration rates.

It is widely recognized that traffic state estimation is a complex nonlinear and stochastic estimation problem. By capturing the essential forward and backward wave propagation characteristics under possible random measurement errors, the proposed parsimonious model structure aims to offer a unified representation with the least complex explanation for traffic observations under free-flow, congested and dynamic transient conditions. This study presented an information-theoretic approach to quantify the value of heterogeneous traffic measurements for specific fixed sensor location plans and market penetration rates of Bluetooth or GPS flow vehicle data.

CHAPTER 5

INCORPORATE A MICROSCOPIC EMISSION MODEL INTO A TRAFFIC SIMULATION MODEL

In order to conduct project-level or network-level traffic environmental impact studies, microscopic emissions models are often adopted in many transportation evaluation projects. These models enable finer spatial and temporal analyses of emissions, but they require very detailed vehicle operating dynamics and traffic fleet parameters.

Simulation-based large-scale traffic simulation models are widely used for better understanding and operating traffic systems on a large-scale network. The mesoscopic dynamic traffic assignment and simulation models, such as DTALite, are capable of capturing the traveler's route choice and modeling different scales of networks, for example, from a single freeway corridor to a large-scale regional network. It is also suitable for evaluating traffic control measures (TCM), such as road pricing, high occupancy vehicle (HOV) lanes and a variety of traveler information strategies. These TCMs, such as adaptive traffic signal control, ramp metering, incident management and dynamic capacity allocation, are likely to significantly impact fuel consumption and emissions. This motivates the practical needs of integrating a microscopic emission model into large-scale traffic simulation models for evaluating both the operating and

environmental Measure of Effectiveness (MOEs) of the TCMs.

One of the key challenges for a traffic simulation model in integrating a microscopic emission model is how to provide the appropriate microscopic high-fidelity trajectory data, namely, second-by-second vehicle speed and acceleration, and then feed the data to a microscopic emission model. In this dissertation, Newell's simplified car-following model is applied to reconstruct the vehicle trajectories from the mesoscopic traffic simulation model. Then, the instantaneous velocity and acceleration rate are derived from the reconstructed vehicle trajectories and fed into the emissions estimation model to generate the environmental MOEs.

The chapter is organized as follows. Section 5.1 describes a simplified MOVES model, MOVES lite. Section 5.2 first introduces Newell's car-following model and how this model can be applied to construct the vehicle trajectories with an output of second-by-second vehicle operating parameters. Section 5.3 describes the system architecture of integrating the traffic simulation model with the microscopic emission model. Experiment results are reported in Section 5.4.

5.1 A Simplified MOVES Emission Model

5.1.1 MOVES

MOVES (Motor Vehicle Emission Simulator) is the air pollution emissions estimation software designed by US Environmental Protection Agency (US EPA). The state-of-the art emission model implemented in MOVES is able to estimate emissions from a wide range of on-road vehicles, for example, cars, trucks, motorcycles and buses. The release of MOVES 2010 in 2010 officially replaced the previous widely adopted

emission model, MOBILE 6.2.

Compared with the MOBILE model, MOVES can estimate a total emission inventory as well as emission rates while MOBILE only provided emission rates and extensive external postprocessing is required to produce an emission inventory. In addition, MOVES can also generate emissions estimates at broader geographic scales and temporal scales, including, hourly, daily, weekly, monthly or yearly levels. MOBILE, on the other hand, only produces emissions on a regional level and has very limited temporal capabilities. MOBILE was based on aggregate driving cycles and only accounted for differences in average speed. MOVES adapts a modal-based estimation approach, which can account for not only average speed, but also different patterns of acceleration, cruising and deceleration.

In this modal-based estimation approach, according to the factors affecting emissions, vehicle activities are grouped into two sets of bins: source bins and operating mode bins. Source bins are categorized by vehicle characteristics such as vehicle types, fuel and engine technologies, ages and model years, and engine size and average weight fraction.

Operating model bins are categorized by second-by-second vehicle activity characteristics represented by Vehicle Specific Power (VSP). VSP is a second-by-second function of vehicle speed, road grade and acceleration that accounts for kinetic energy, rolling resistance, aerodynamic drag and gravity in combination (Jimènez, 1999). The equation to calculate VSP is expressed as follows:

$$VSP = (A/M)\times v + (B/M)\times v^2 + (C/M)\times v^3 + \left(a + \sin(\phi)\right)\times v \qquad (5.1)$$

where A (metric ton), B (metric ton/(m/s)) and C (metric ton/(m/s)^2) = rolling term,

rotating term and drag term, respectively; M = vehicle mass (metric ton); v = vehicle speed (m/s); a = vehicle acceleration (m/s^2); ϕ = road grade. The individual A, B, C terms and vehicle mass are listed in Table 5.1.

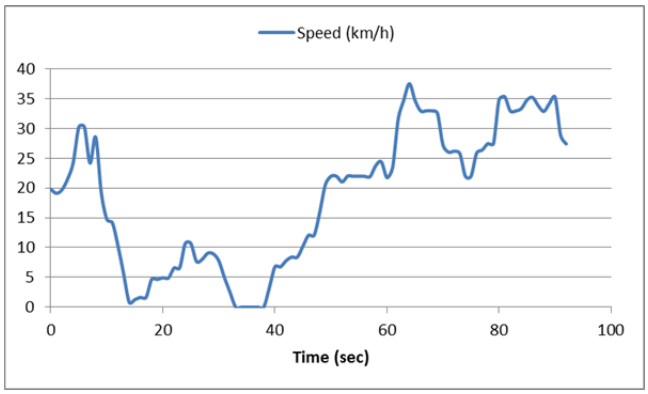
Figure 5.1-a and c, b and d show the speed and acceleration of two types of vehicles: passenger car and passenger truck. These second-by-second speed and acceleration data are obtained from the NGSIM dataset. The lead vehicle is the passenger car while the following vehicle is a passenger truck, and these two vehicles have a very similar speed and acceleration profile. The total distance traveled by these two vehicles is 1600 feet and the travel times and average speeds of these two vehicles are 92 and 91 seconds, equivalent to 18.57 km/h and 18.78 km/h, respectively. The vehicle trajectories are shown in Figure 5.2.

Using Eq. (5.1) and the second-by-second speed and acceleration data, we can calculate the second-by-second VSPs for these two vehicles. The calculated second-by-second VSPs of the passenger car and the passenger truck are shown in Figure 5.3.

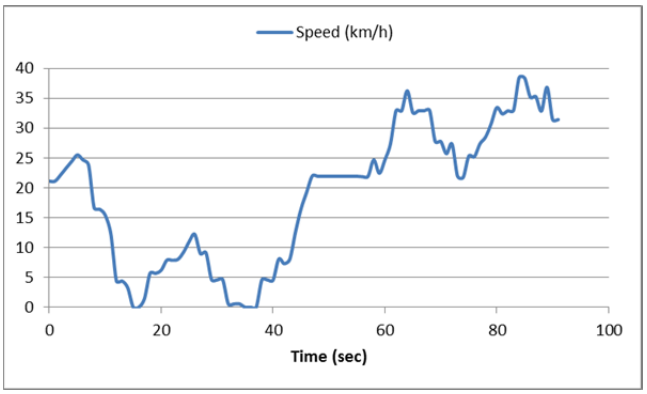
The calculated VSPs are then categorized by speed and VSP ranges. The detailed classification is defined in Table 5.2. Based on Table 5.2, the VSP distribution of the example passenger car and passenger truck is displayed in Figure 5.4.

Table 5.1: Terms and Mass by Vehicle Type

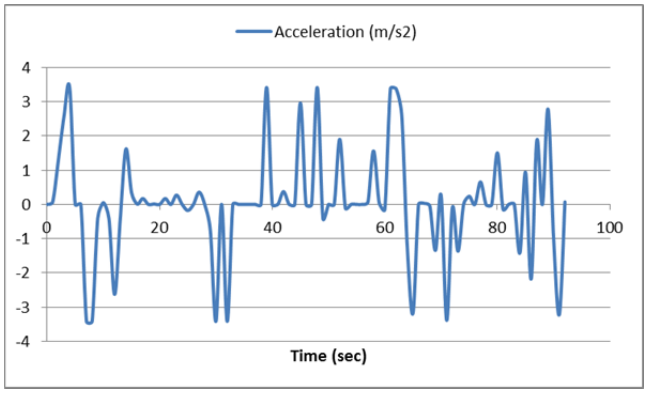
Vehicle Type	Rolling Term	Rotating Term	Drag Term	Vehicle Mass (metric ton)
Passenger Car	0.156	0.002	0.000493	1.4788
Passenger Truck	0.221	0.002	0.000698	1.8668
Light Commercial Truck	0.235	0.003	0.000748	2.0597
Single Unit Short-haul Truck	0.561	0	0.001603	7.6415
Combination Long-haul Truck	2.081	0	0.004188	31.4038



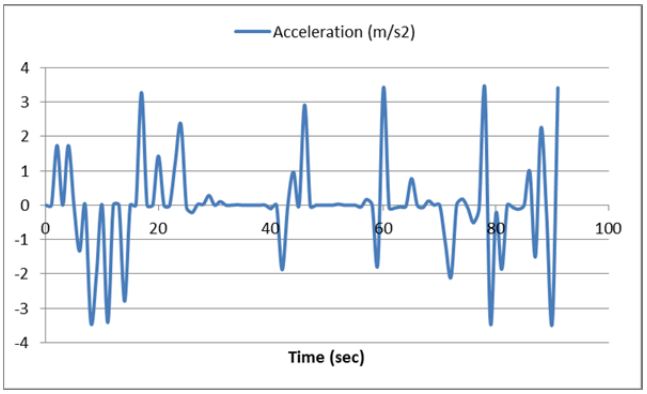
a



b



c



d

Figure 5.1: Example Vehicle Speed and Acceleration. (a) Speed of Passenger Car, (b) Speed of Passenger Truck, (c) Acceleration of Passenger Car, (d) Acceleration of Passenger Truck.

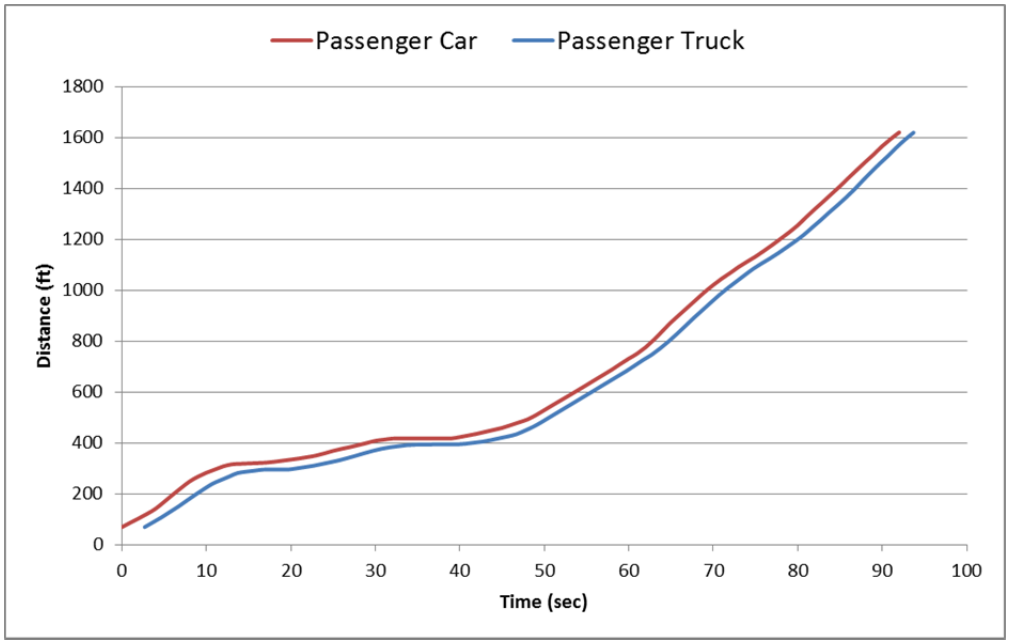


Figure 5.2: Vehicle Trajectories from NGSIM Data

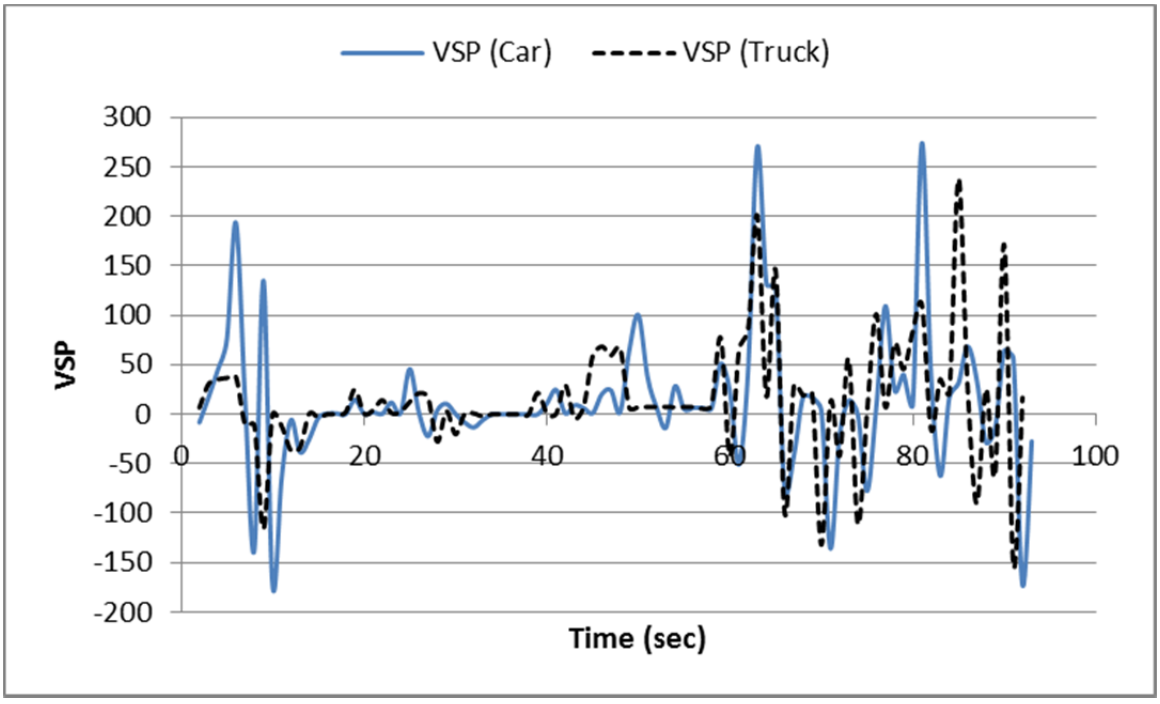


Figure 5.3: Vehicle Specific Power Time Series of Passenger Car and Truck

Table 5.2: Definition of MOVES Operating Mode Bins by Speed and VSP Ranges

0 mph < v_i ≤ **25 mph**		25 mph < v_i ≤ **50 mph**		v_i > **50 mph**	
OpMode ID	Description	OpMode ID	Description	OpMode ID	Description
11	VSP< 0	21	VSP< 0		
12	0≤VSP< 3	22	0≤VSP< 3		
13	3≤VSP< 6	23	3≤VSP< 6	33	VSP< 6
14	6≤VSP< 9	24	6≤VSP< 9	35	6≤VSP<12
15	9≤VSP<12	25	9≤VSP<12		
16	12≤VSP	27	12≤VSP<18	37	12≤VSP<18
Other:		28	18≤VSP<24	38	18≤VSP<24
0	Braking	29	24≤VSP<30	39	24≤VSP<30
1	Idling	30	30≤VSP	40	30≤VSP

v_i: instantaneous speed of the i^{th} second

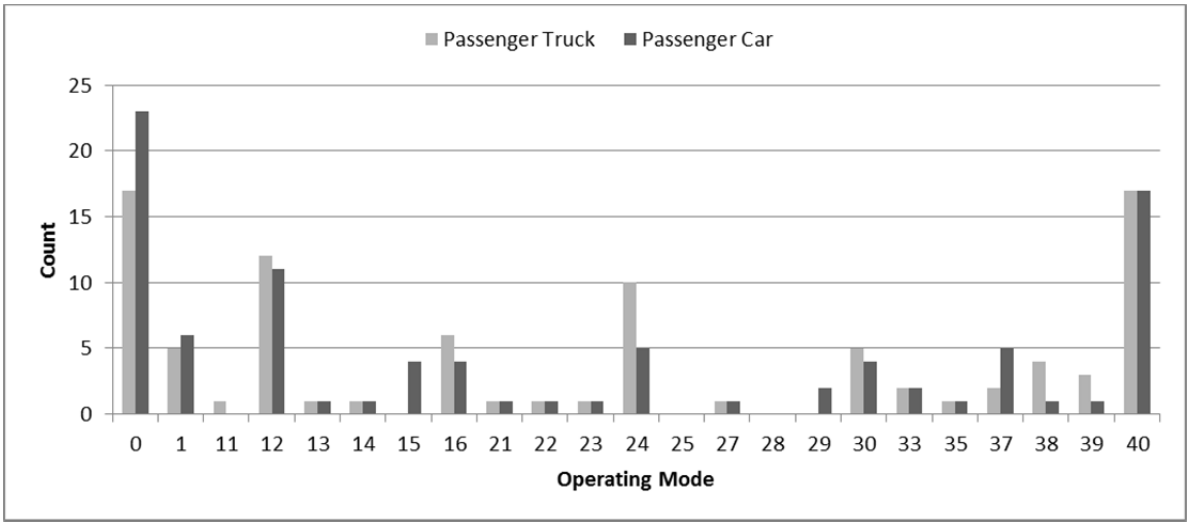


Figure 5.4: VSP Distribution for Passenger Truck and Passenger Car

5.1.2 MOVES Lite

Although MOVES is able to provide a very fine-grained modeling of emission factors, it requires extremely detailed emission rate inputs for every possible combination of source bin and operating mode bin. These data usually are too large to store in the computer main memory, and rational databases are commonly utilized to keep these data. However, querying databases to obtain the corresponding emission rate by a specific source bin and operating mode bin is a relatively slow operation (millisecond) compared with direct memory access (nanosecond). These software system implementation factors make MOVES a computationally and data intensive model and it has been very difficult to link MOVES with a traffic simulation model. This motivates the development of a simplified and light-weight version of MOVES, MOVES Lite (Frey et al., 2011).

Compared with MOVES, MOVES Lite has made the following major simplifications to reduce the amount of data required and improve the calculation efficiency.

- Considers a limited number of vehicle types

Unlike MOVES, which models more than 10 vehicles types, MOVES Lite only considers a limited number of representative vehicle types (95% of on road fleet)

- Passenger cars
- Passenger truck
- Light commercial truck
- Single unit short-haul truck
- Combination long-haul truck

Using a limited number of vehicle types can significantly reduce the number of the source bins, and therefore dramatically decrease the complexity of the emission rate

searching process.

- Consider average emission rates for each vehicle types

Instead of enumerating the emission rates corresponding to the possible vehicle activities for each specific vehicle types, MOVES Lite uses average emission rates stratified by vehicle type and vehicle operating mode. These average emission rate data are calibrated by Portable Emissions Monitoring Systems (PEMS). Table 5.3 and Table 5.4 list the calibrated emission rate and energy consumption under different operating mode for the 0-year old passenger car and truck, respectively (Frey and Liu, 2013).

5.1.3 Calculate Emissions and Energy Consumptions for Example Vehicles

Using the calculated VSPs from NGSIM data shown in Section 5.1.1 and the calibrated average vehicle emission rates in Tables 5.3 and 5.4, this section illustrates the detailed calculating procedure for emissions and energy consumptions of example vehicles.

We first compare in Table 5.5 the vehicle speed, acceleration, VSP and operating mode at one specific second (e.g., the 3rd second) between the passenger car and passenger truck. The big difference of the obtained VSPs at this timestamp is attributed to two factors: 1) The different instantaneous speed and acceleration rate; and 2) The different characteristics of the vehicle types, i.e., the A, B and C terms and the vehicle masses. Based on the VSP and the speed, we need to look up Table 5.2 to obtain the corresponding operating mode. Then, the associated emission and energy consumption rates are further obtained from Table 5.3 and Table 5.4, respectively. The emission rates

Table 5.3: Average Emission Rate for 0-year Passenger Car (Source: MOVES Lite, Frey and Rouphail, 2012)

Operating Mode	Energy (KJ/h)	CO2 (g/h)	NOX (g/h)	CO (g/h)	HC (g/h)
0	49206	3536	0.05	2.37	0.04
1	45521	3271	0.01	4.06	0.00
11	71581	5144	0.15	6.52	0.02
12	98841	7103	0.16	2.82	0.02
13	137367	9872	0.36	9.77	0.05
14	173571	12474	0.66	14.21	0.07
15	206979	14875	1.19	20.88	0.10
16	249989	17966	2.53	35.99	0.17
21	97383	6999	0.25	5.82	0.04
22	110849	7966	0.36	9.33	0.05
23	135007	9703	0.51	13.18	0.07
24	173205	12448	0.93	25.90	0.12
25	231143	16611	1.52	18.53	0.12
27	304713	21899	2.14	34.77	0.22
28	410729	29518	8.21	200.66	3.25
29	562702	40440	11.14	216.01	4.05
30	706632	50783	12.84	969.88	7.12
33	138741	9971	0.42	10.92	0.07
35	222473	15988	1.41	23.85	0.09
37	289809	20828	2.28	28.95	0.12
38	377895	27158	7.50	68.20	1.16
39	503348	36174	13.34	167.61	2.12
40	641649	46113	14.34	407.60	2.73

Table 5.4: Average Emission Rate for 0-year Passenger Truck (Source: MOVES Lite, Frey and Rouphail, 2012)

Operating Mode	Energy (KJ/h)	CO2 (g/h)	NOX (g/h)	CO (g/h)	HC (g/h)
0	69191	4973	0.97	8.51	0.29
1	53350	3834	0.40	4.53	0.14
11	86302	6202	1.18	14.02	0.44
12	107972	7760	1.11	6.11	0.33
13	155314	11162	2.46	17.31	0.69
14	196764	14141	3.76	24.18	0.84
15	237241	17050	4.53	34.70	1.11
16	296319	21295	7.52	50.34	1.61
21	115436	8296	2.28	15.72	0.36
22	123936	8907	3.84	19.27	0.55
23	153030	10998	3.85	22.82	0.64
24	198225	14246	5.43	35.80	0.84
25	254545	18293	5.76	29.74	0.72
27	350453	25186	6.05	43.90	0.73
28	467979	33632	12.65	204.30	3.74
29	641507	46103	16.40	230.27	4.98
30	755659	54307	20.11	970.12	8.02
33	166778	11986	5.35	27.83	0.21
35	255239	18343	6.98	38.96	0.23
37	337463	24252	8.83	44.26	0.40
38	438398	31506	14.14	82.68	1.28
39	574357	41277	20.53	178.12	2.22
40	759990	54618	23.71	408.70	3.07

Table 5.5: VSP and Operating Model at the 3rd Second

Vehicle Type	Speed (mph)	Speed (m/s)	Acceleration (m/s^2)	VSP	Operating Mode
Passenger Car	44.08	19.71	0.60	**17.05**	**27**
Passenger Truck	49.58	22.16	1.04	**30.55**	**30**

associated with the calculated operating modes are shown in Table 5.6.

Based on the above calculation procedure, the total energy consumption and emissions for these two vehicles are computed and compared in Figure 5.5. It can be observed that under the same driving condition, the passenger truck has both higher emissions and energy consumption compared to the passenger car.

5.2 Mesoscopic Traffic Simulation Model

The native output of a mesoscopic traffic simulation model is vehicles' arrival and departure times on the links, and it does not include second-by-second vehicle operating parameters. Given the arrival and departure time of the individual vehicle on a link, we can adapt the Newell's simplified car-following model (Newell, 2002) to reconstruct the vehicle trajectory in the link. Then, the detailed second-by-second vehicle speed and acceleration are derived from the reconstructed vehicle trajectory.

Table 5.6: Emission and Energy Consumption Rate at the 3rd Second

Vehicle Type	Age	Operating Mode	Energy (KJ/h)	CO2 (g/h)	NOX (g/h)	CO (g/h)	HC (g/h)
Passenger Car	0	**27**	304713	21898	2.14	34.77	0.22
Passenger Truck	0	**30**	755658	54306	20.11	970.11	8.02

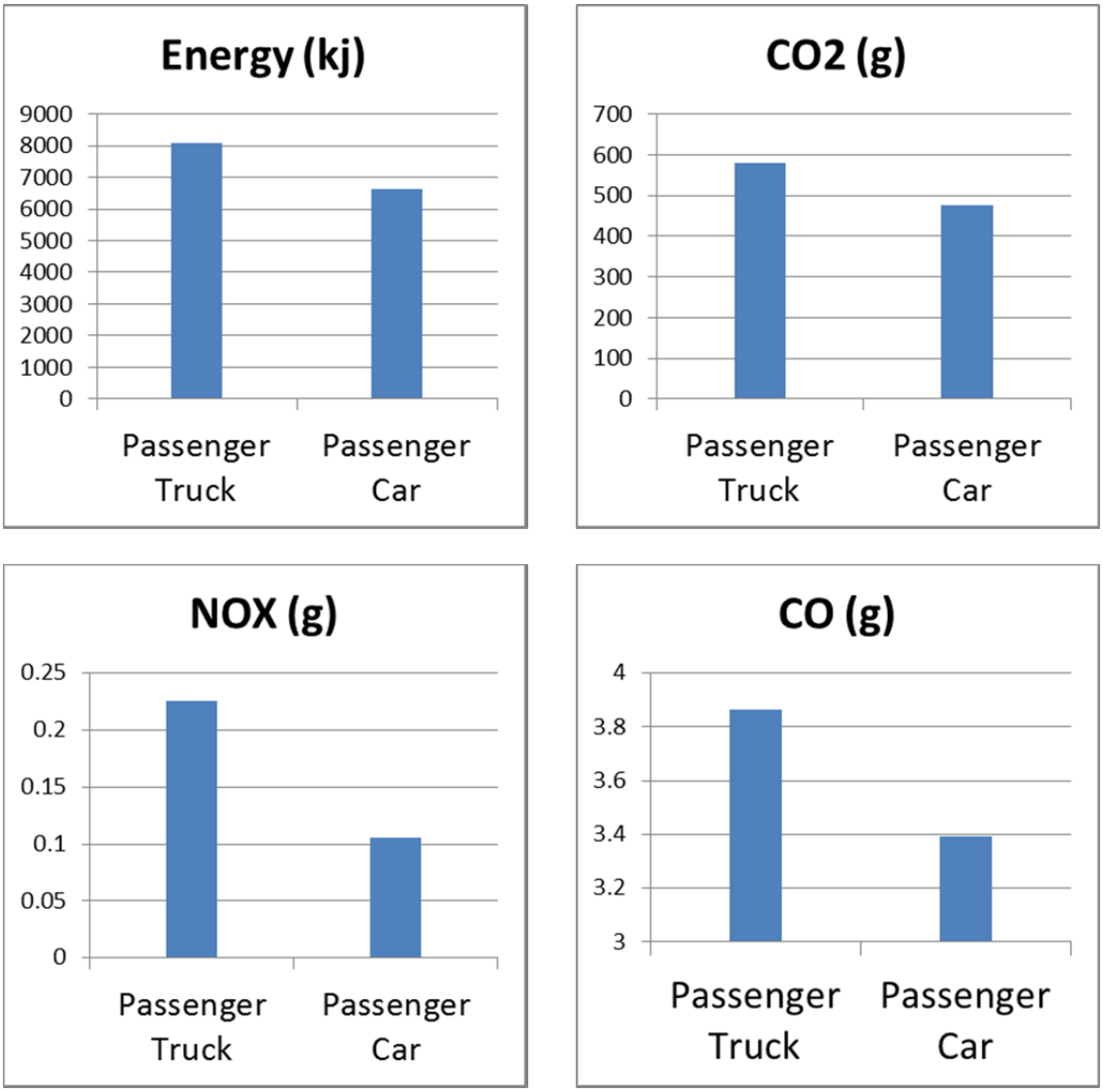


Figure 5.5: Total Energy and Emissions

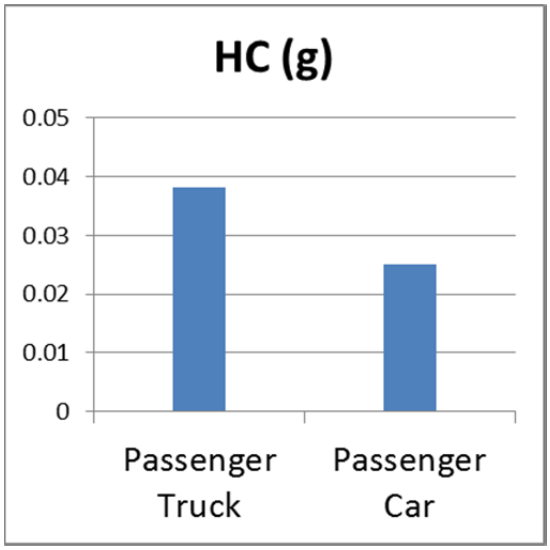


Figure 5.5: Continued

5.2.1 Review of Newell's Car-following Model

The main idea of Newell's car-following model is that a (following) vehicle *n* maintains a minimum space and time gap between it and the preceding vehicle *n-1*. When the lead vehicle *n-1* changes its speed, the following vehicle *n* changes its speed along the backward wave, w. The car-following trajectory is illustrated in Figure 5.6.

Mathematically, the Newell's car-following model states that the vehicle trajectory relationship between the following vehicle *n* and the lead vehicle *n-1* is defined as:

$$x_n\left(t+\tau_n\right)=x_{n-1}\left(t\right)-d_n \tag{5.2}$$

where $x_{n-1}\left(t\right)$ is the position of the lead vehicle at time *t*, $x_n\left(t+\tau_n\right)$ is the position of the following vehicle at time $t+\tau_n$; τ_n and d_n are the appropriate time and space gaps, respectively.

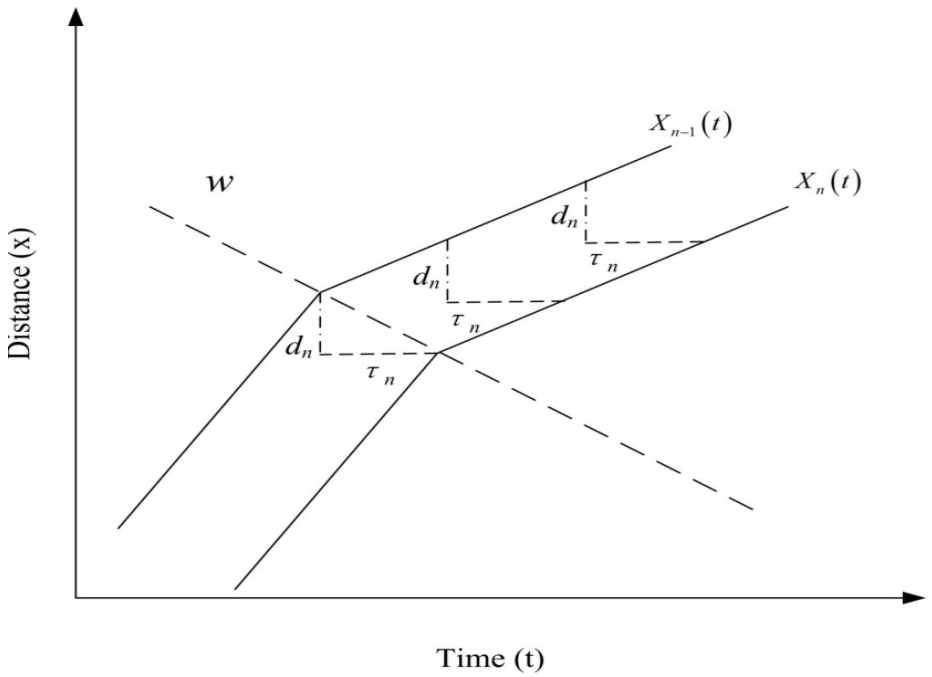


Figure 5.6: Newell's Car-following Model

In reality, τ_n and d_n are expected to vary considerably from vehicle to vehicle. For simplicity, in our proposed vehicle trajectory reconstruction method, the minimum time and space gap is assumed the same for all vehicles so that the subscript *n* is dropped in the following analysis. The minimum space gap *d* is determined by the maximum number of vehicles on the link or the link jam density, k_j:

$$d = \frac{1}{k_j} \tag{5.3}$$

while τ_n is constrained by both the jam density k_j and the backward wave speed, w:

$$\tau = \frac{1}{w \times k_j} \tag{5.4}$$

5.2.2 Vehicle Trajectory Construction Using Newell's Simplified Car-following Model

Given the vehicles' arrival and departure times on the link from a macroscopic dynamic traffic assignment and simulation model such as DTALite, the procedure for calculating the vehicle trajectories along this link is described below.

Input:

L: the length of the link;

v_f : the free-flow speed on the link;

d : be the minimum space gap;

τ : be the minimum time gap;

$Arr(n)$: the arrival time of vehicle *n* at the upstream node of the link;

$Dep(n)$: the departure time of vehicle *v* at the downstream node of the link;

ΔT : the time step increment.

Variables:

$X(n,t)$: the position of vehicle *n* at time *t*;

For each vehicle $n = 1,\ldots,N$

Initialize the starting position of a link $X(n, Arrival(n))$=0;

For each time interval *t* = $Arr(n)$ to $Dep(n)$

Calculate free-flow driving position:

$$X^F(n,t+\Delta T) = \min\left\{X(n,t) + v_f \times \Delta T, L\right\};$$

If *n* is the first vehicle, where $n == 0$

$$X(n,t+\Delta T) = X^F(n,t+\Delta T)$$

Else

Calculate position determined by backward wave propagation from the leader *n-1*:

$$X^B(n,t+\Delta T) = X(n-1,t+\Delta T-\tau) - d;$$

Calculate the final feasible position:

$$X(n,t+\Delta T) = \min\left\{X^F(n,t+\Delta T), X^B(n,t+\Delta T)\right\};$$

End If

$t = t + \Delta T$;

End For

End For

5.2.3 Illustrative Example

In this section, we use a simple hypothetical corridor to illustrate the underlying traffic flow model used by DTALite.

Consider the above simple corridor Figure 5.7, which consists of 3 single-lane links. The links are numbered from left to right as 1, 2 and 3. The capacity of links 1 and 2 is 1800 vphpl (vehicles per hour per lane) and the capacity of link 3 drops from 1200 vpmpl to 900 vpmpl. The jam density and backward wave on these links are all 180 vpmpl (vehicles per mile per lane) and 12 mph. The hourly demand on this corridor is 1800 vph (vehicles per hour). Detailed link information is listed in Table 5.7.

The vehicle loading horizon in this experiment is set to 1 hour and the total number of vehicles to be simulated is 1800. Figure 5.8 shows the simulated link outflow volumes from DTALite. Because the inflow rate of link 3 (900 vphpl) is lower than the outflow rate (1800 vphpl) of link 2, this leads to a bottleneck at the boundary of link 2 and link 3. The bottleneck constrains the inflow rate of link 2 to 900 vphpl. Additionally, the bottleneck is then propagated from link 2 backward to link 1 and causes the outflow rate at link 1 to be reduced to 900 vphpl.

Figure 5.7: Simple Corridor with Capacity Drop

Table 5.7: Link Information of the Simple Corridor

Link	# of Lanes	Length (miles)	Capacity (vphpl)	Free Speed (mph)	Jam Density (vpmpl)	Backward Wave (mph)
1	1	1	1800	60	180	12
2	1	1	1800	60	180	12
3	1	1	900	60	180	12

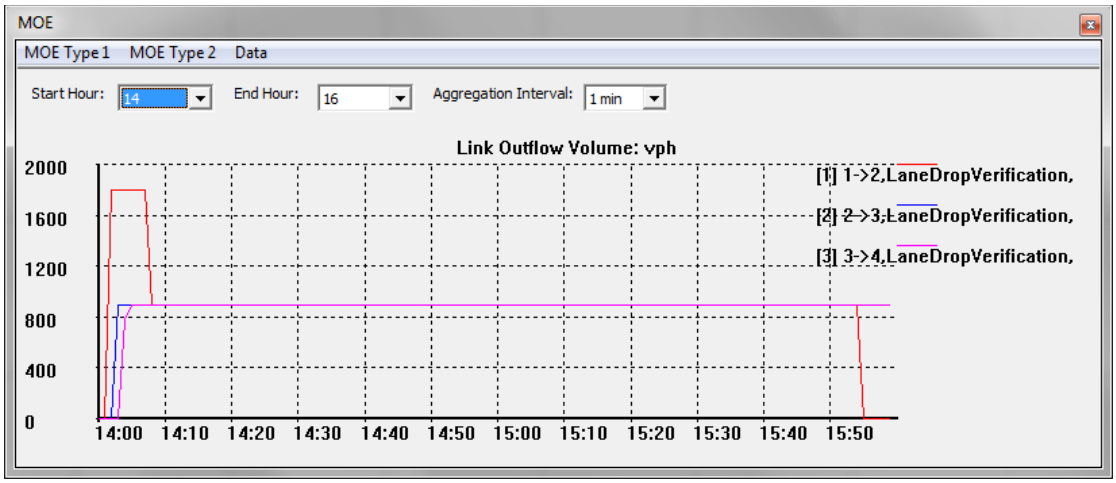


Figure 5.8: Link Outflow Volume

The simulated link density is shown in Figure 5.9. The traffic in link 3 is operating at the free-flow state and the density in this link is calculated as 15 vpmpl based on the flow-density relationship. Given the specified free-flow speed (60 mph), backward wave speed (12 mph) and jam density (180 vpmpl), the triangular fundamental density-flow diagram of link 2 is constructed and shown in Figure 5.10. This density-flow diagram is divided by the critical density line *AB* into two sides: 1) the free-flow side (left side of *AB*) and 2) the congestion side (right side of *AB*). Based on this traffic flow relationship, when the capacity is 900 vphpl, the density could be either on the left side of *AB* at point

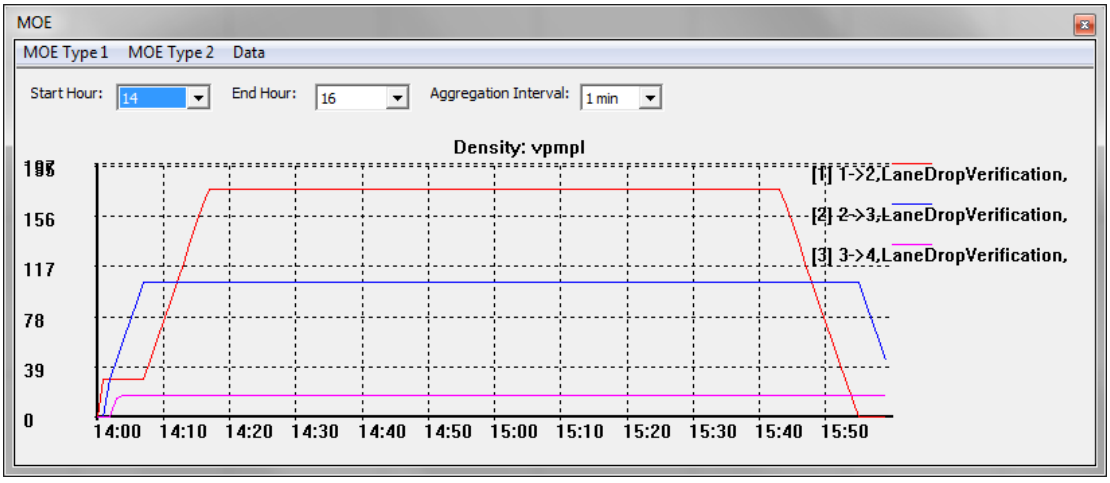


Figure 5.9: Link Density

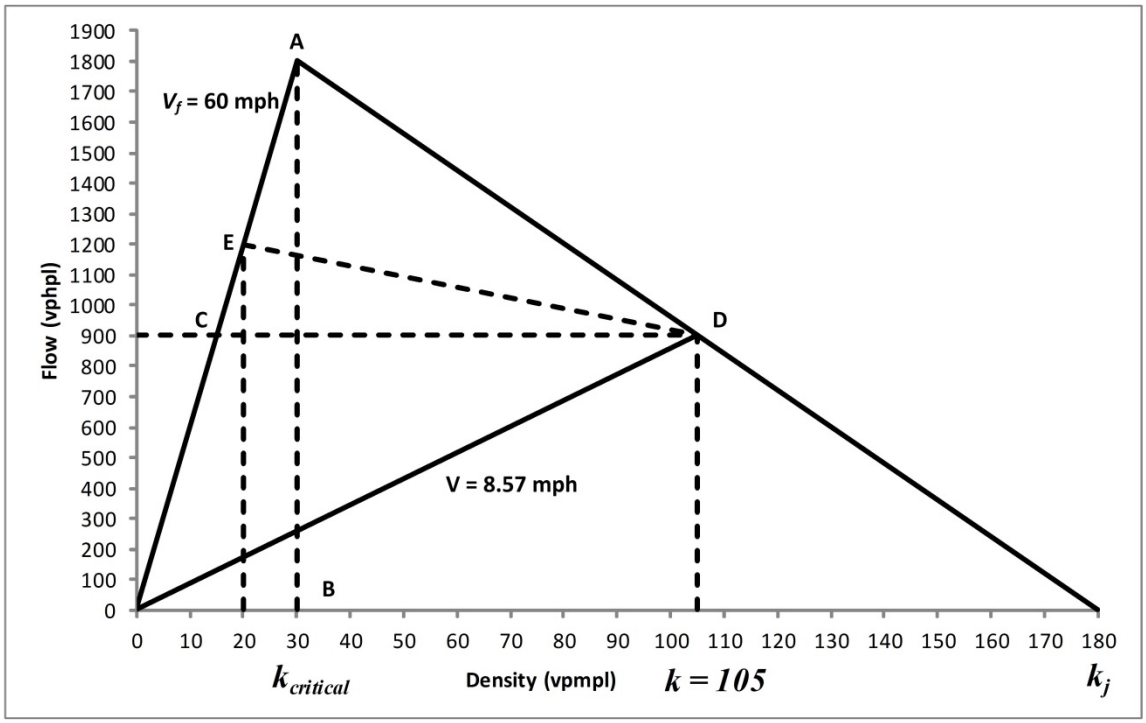


Figure 5.10: Fundamental Diagram of Link 2

C with a value of 15 vpmpl or on the congestion side at point *D* with a value of 105 vpmpl. It is known that the traffic on link 2 is under the congestion state. Thus, the corresponding density can only be on the congestion side at point *D*. This is consistent with the simulated density of link 2 shown in Figure 5.9. According to the flow-density relationship, the speed under the flow rate of 900 vphpl is 8.57 mph. Meanwhile, a shock wave is formed at the boundary of link 2 and link 3 due to the change of the states. The flow rate before the boundary is 1800 vphpl (State A in Figure 5.10) and the flow rate after the boundary is 900 vphpl (State D in Figure 5.10), the shockwave speed, ω, is calculated with the shockwave equation:

$$\omega = \frac{\Delta q}{\Delta k} = \frac{1800 - 900}{30 - 105} = -12 \text{ mph} \tag{5.5}$$

where Δq and Δk are the changes of flow and density, respectively. The shock wave in this special demand-supply configuration is the same as the backward wave speed. In Figure 5.8, it takes 1mile/12 mph*60 = 5 min for the shock wave to traverse through link 2->3. That is, the outflow rate of link 1->2 is reduced to the bottleneck capacity 5 minutes after link 2->3 reduces its full capacity. Similarly, we can observe the same density change on link 1->2 at 8 minutes. In Figure 5.9, the link density on link 2->3 is close to the theoretical value of 105 vehicles per mile per lane calculated from Figure 5.10. Link 1->2 is a demand loading link so its actual density is relative high in this bottleneck case. Figure 5.11 further shows the constructed vehicle trajectories using Newell's car-following model on link 2. At the beginning, the vehicle travels at the free-flow speed, v_f at 60 mph (green arrow). The shock wave travels from the downstream to the upstream (yellow arrow). It takes approximately 0.08 hours (4.8 min) for the

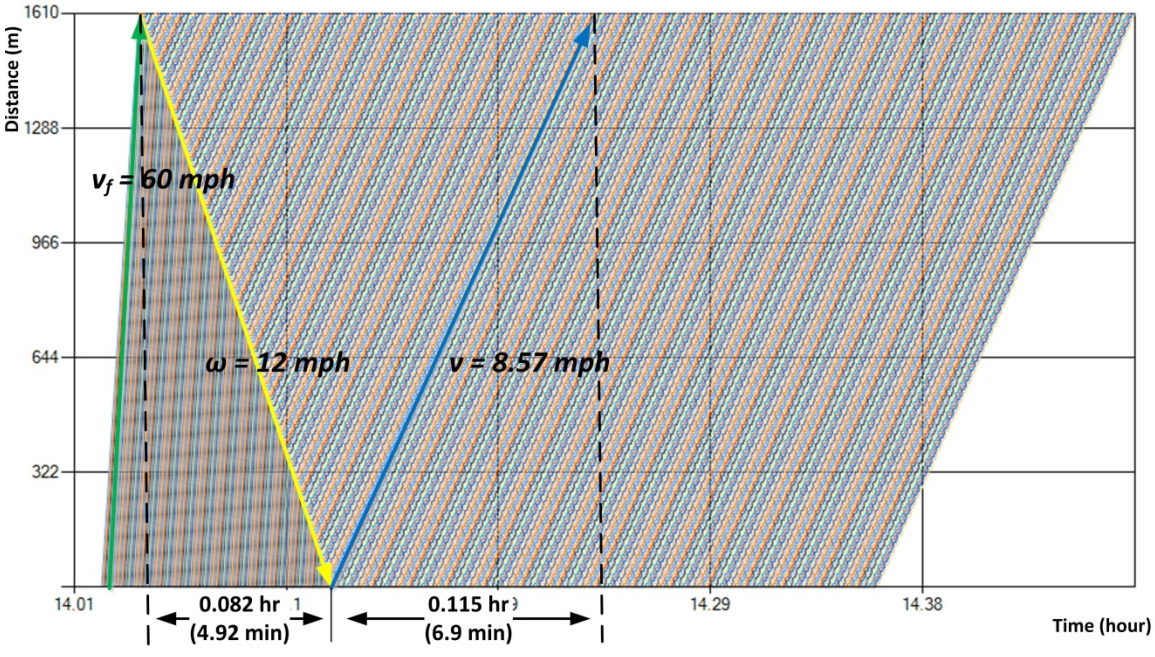


Figure 5.11: Constructed Vehicle Trajectory by Newell's Car-following Model on Link 2

backward wave to travel from the downstream node to the upstream node of the link. It is known that the length of link is 1 mile. The shock wave speed is then computed as $1/0.082 = 12.2$ mph. When the free-flow moving vehicle encounters the backward wave, the speed is reduced from the free-flow speed. The reduced speed calculated from the vehicle trajectory is $1/0.115 = 8.69$ mph (blue arrow), which indicates that the speed from the constructed vehicle trajectories is able to correctly approximate the one derived from the fundamental diagram.

As shown in Figure 5.9, the traffic on link 1 shows a different pattern compared to link 2. Because the inflow volume (1800 vphpl) is higher than the outflow volume (900 vphpl), vehicles are queued in link 1 until the density finally reaches the jam density (180 vpmpl). It should be remarked that the shock wave speed under this link configuration case for link 2 is a special case where the shock wave speed is equal to the backward wave speed.

The following discussion will consider a more general case where the flow rate in link 2 is 1200 vphpl (State E in Figure 5.10). In this case, the change of flow is 1200 – 900 = 300 vphpl and the change of density is 20 – 105 = -85 vpmpl. According to Eq. (5.6), the shock wave speed $\omega = \frac{300}{-85} = -3.53$ mph and the travel time from the downstream node of link 2 to the upstream node of link 3 would be 17 minutes. In addition, the travel speed after the shock wave is still 8.57 mph since the flow rate after the shock wave is still 900 vph. Figure 5.12 shows the constructed vehicle trajectories and shock wave under this hypothetical demand-supply setting.

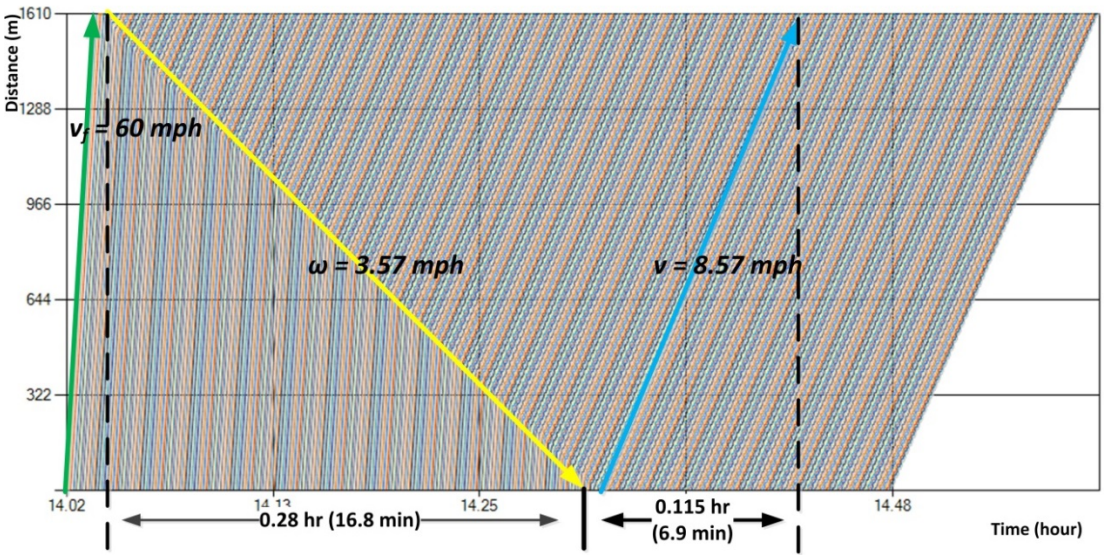


Figure 5.12: Vehicle Trajectories and Shock Wave

5.2.4 Limitation of the Newell's Car-following Model

We have shown that using Newell's car-following model to construct vehicle trajectories is able to theoretically represent the underlying traffic dynamics, especially backward wave propagation, but several limitations still exist and should be carefully addressed in this and future research.

- Unrealistic acceleration at the transition points

Let us consider the simplest example trajectory of the first vehicle on a freeway link constructed with Newell's car-following model, as shown Figure 5.13. In this trajectory, there are 2 transition points: A and B. As it is the first vehicle in the link, its travel speed at t_1 is the maximum allowed speed, the link free-flow speed at 60mph. Since the speed of the vehicle at t_0 was 0, the acceleration rate would be 60mph/s. This is very unrealistic given the fact that the maximum practical vehicle acceleration is less than 10 mph/s.

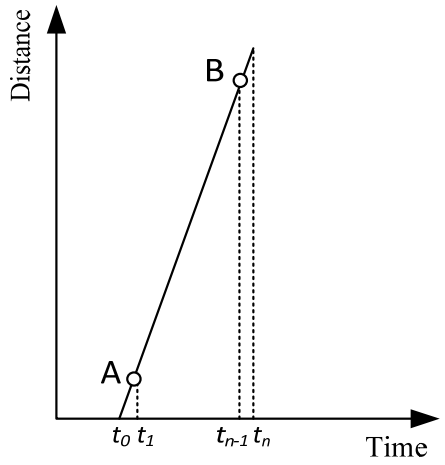


Figure 5.13: Limitation of Newell's Car-following Model

Similarly, this vehicle will come to a complete stop at t_n from the free-flow speed at t_{n-1} with an unrealistic deceleration rate of -60 mph/s.

To address the issue of unrealistic acceleration, we may adjust the trajectory of the vehicle to follow a series of reasonable acceleration profiles to complete the speed up process, as shown in Figure 5.14. However, the acceleration stage must continue even if the speed has reached the desired free-flow speed to catch up the travel distance as well. Conceptually, we may divide this "catch-up" procedure into three stages: the accelerating stage $[t_0, t_a]$, the cruising stage $[t_a, t_b]$ and the decelerating stages $[t_b, t_c]$. The vehicle continues accelerating to a prespecified speed (e.g., 10 mph above free-flow speed) and this prespecified speed must be higher than the free-flow speed during the accelerating stage. During the cruising stage, the vehicle maintains at that prespecified speed to catch up the travel distance. When approaching the catch-up point, the vehicle starts to

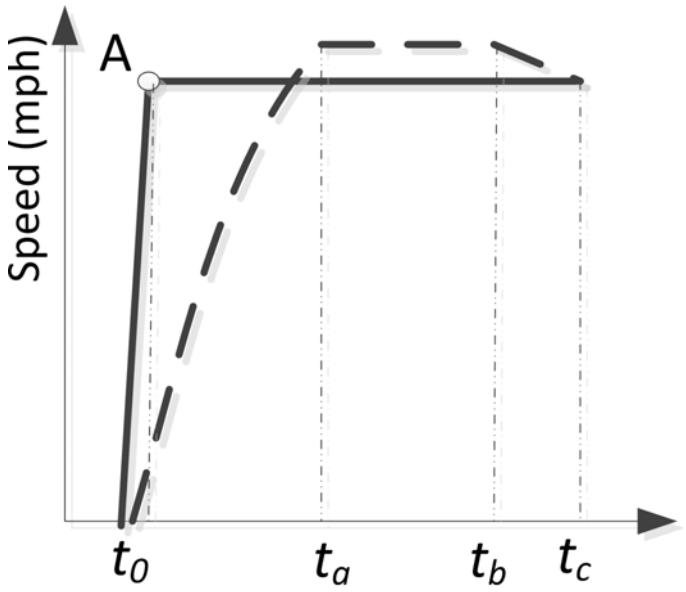


Figure 5.14: Correcting Unrealistic Acceleration Profile

decelerate until the free-flow speed. The catch-up point must be carefully set so that the acceleration and declaration rates during this procedure are realistic.

For the unrealistic instantaneous deceleration situation at point B, we may utilize a similar correcting procedure as to adjust the unrealistic acceleration, as shown in Figure 5.15. During the stages $\left[t_e, t_f\right]$ and $\left[t_f, t_g\right]$, the vehicle has to accelerate and maintain to a prespecified speed to get the buffer distance for the following deceleration stage so that the deceleration rates during stage $\left[t_g, t_n\right]$ would be realistic.

- Lack of speed variation

Between the transition points mentioned in Limitation 1, the theoretical car-following trajectory has no speed fluctuation and the vehicle travels at the constant speed, from space-time point A to B. This is rarely observed in reality, as demonstrated from the real-world speed profile obtained from a field test carried out by North Carolina State

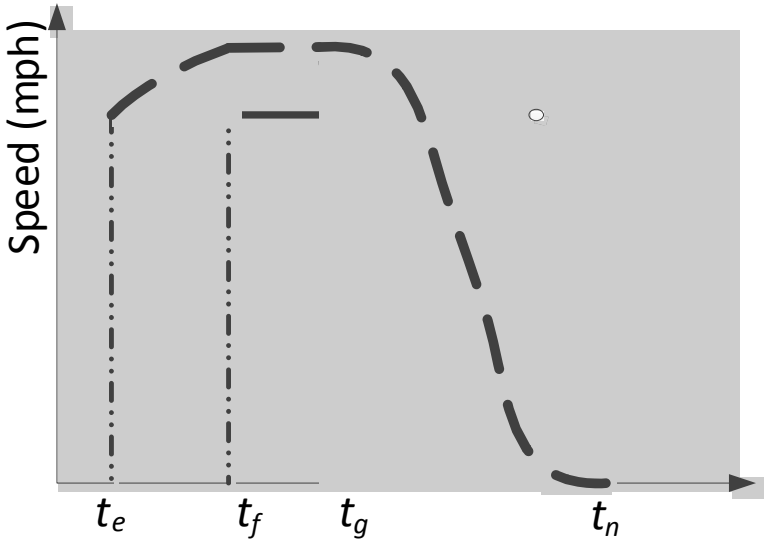


Figure 5.15: Correcting Unrealistic Deceleration Profile

University in Figure 5.16. In this figure, two segments of the stable driving stretches have been marked. As we can see, significant speed fluctuations are observed in these two segments.

One way to address this limitation is to map the constant speed segments with the representative microtrips collected from the real-world field testing driving cycles. Specifically, the representative microtrips collected from the real driving cycles are

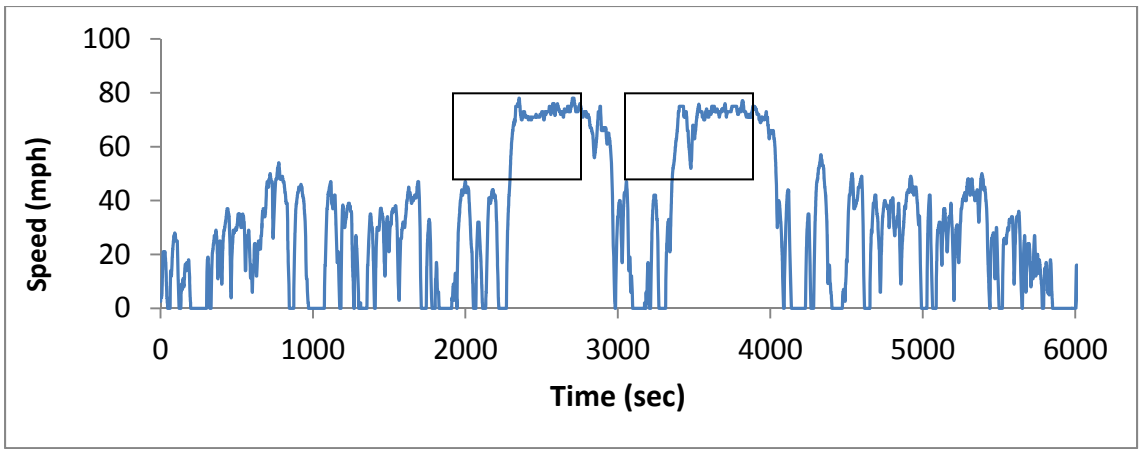


Figure 5.16: Second-by-second Speed from Empirical Emission Measurement on 2004 Honda Civic

stored in the database. After a vehicle trajectory is simulated by the car-following model, the unrealistic parts of the trajectory are identified. Given the start and end speeds, and the total distance traveled over this unrealistic duration, we first query the microtrip database to obtain the matched representative trajectories. Then, we combine the queried representative microtrips to reconstruct the vehicle trajectory. This is illustrated in Figure 5.17.

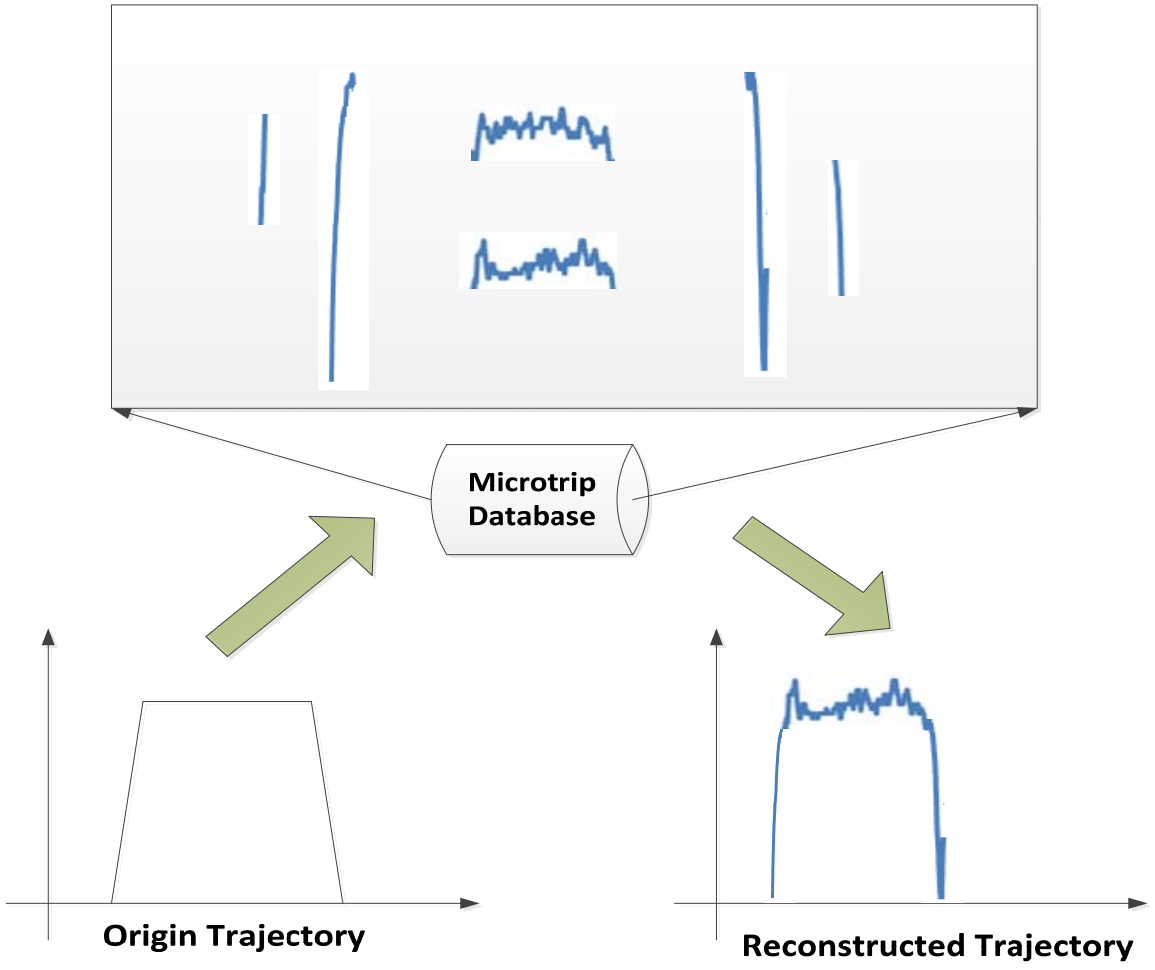


Figure 5.17: Microtrip-based Vehicle Trajectory Reconstruction

Although limitations are present in the constructed trajectories by Newell's car-following model, this method holds the potential to provide a practical way to bridge the data requirements between the macroscopic traffic model and the microscopic emission estimation model.

5.3 System Framework for Incorporating Mesoscopic Traffic Simulation Model with Microemission Model

Figure 5.18 illustrates how the traffic simulation model is incorporated with the microscopic emission estimation model.

The detailed modeling processes are described in the following.

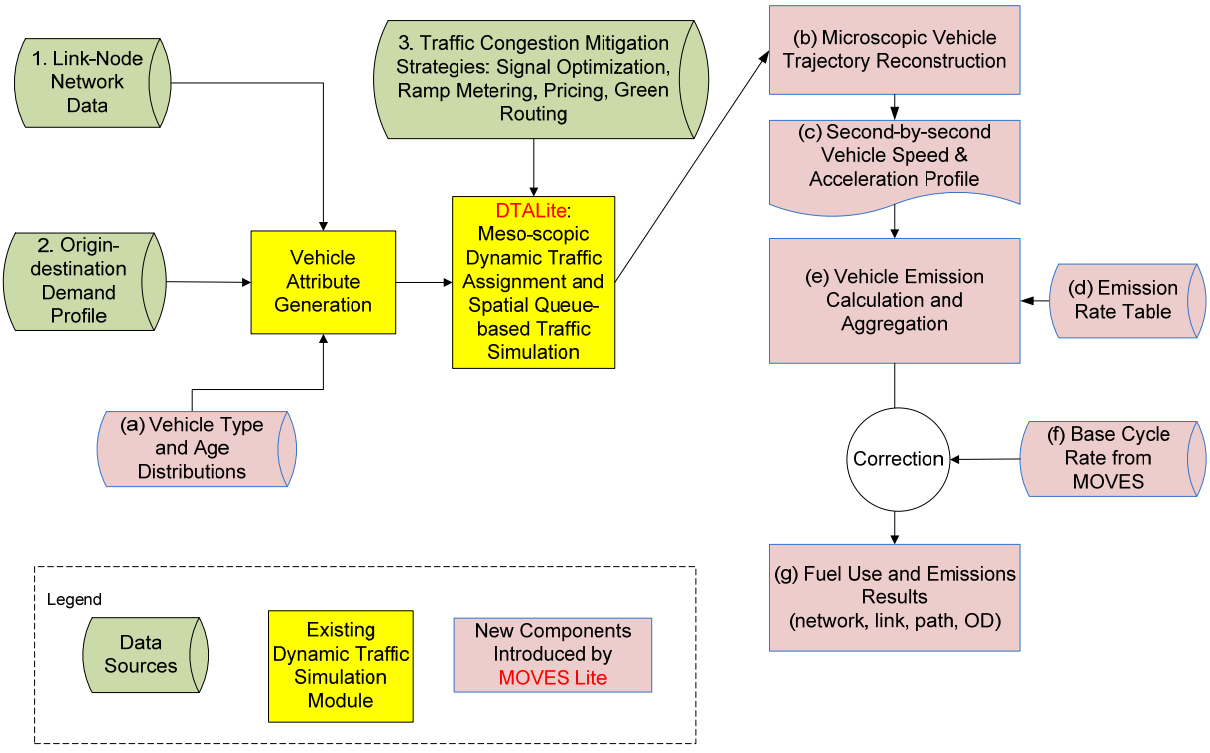


Figure 5.18: System Framework of DTALite and MOVES lite

5.3.1 Vehicle Generation

The inputs of this vehicle generation process are:

a) Traffic demand data (number of vehicles between each origin and destination pair, vehicle departure time and so on);

b) Vehicle data, including vehicle type and age and their distribution, as shown in Table 5.8.

Based on the number of vehicles between each origin and destination pair, vehicle departure time and vehicle type and age distribution information, the vehicle generator generates the vehicles for the traffic assignment and simulation module. Each vehicle is characterized by the origin and destination node, the departure time at the origin node, vehicle type and vehicle age.

5.3.2 Traffic Assignment and Simulation Modules (DTALite)

The inputs for the DTALite simulator are:

a) Network data, including links and nodes;

b) Vehicles generated from the vehicle generator;

c) Traffic control measures, i.e., signal optimization, ramp metering, pricing

Table 5.8: Vehicle Age Distribution by Type and Age (Source: MOVES Lite, Frey and Rouphail, 2012)

Vehicle Type	Name	Vehicle Distribution by Age			
		Age 0	Age 5	Age 10	Age 15
1	Passenger Car	0.06	0.48	0.28	0.19
2	Passenger Truck	0.03	0.43	0.26	0.27
3	Light Commercial Truck	0.03	0.44	0.26	0.27
4	Single Unit Short-haul Truck	0.04	0.52	0.23	0.21
5	Combination Long-haul Truck	0.04	0.52	0.23	0.21

strategies.

As the most important module, DTALite first calculates the time-dependent shortest path for each vehicle based on its origin and destination and the time-dependent link travel times at each iteration. Then, the traffic simulation is performed to move the vehicles from their origins to their destinations. During this process, the time-dependent link travel times are updated. The assignment and simulation are repeated until the equilibrium is achieved.

The outputs of DTALite for emission estimation purpose are:

a) Cumulative vehicle arrival and departure counts on each link;

b) The link arrival and departure times for each vehicle.

5.3.3 Vehicle Trajectory Construction Module

Based on the arrival and departure time of the vehicle at each link, the microscopic car-following-based vehicle trajectory construction module simulates the vehicle trajectory and generates the second-by-second vehicle speed and acceleration profile.

5.3.4 Microscopic Emission Module

MOVES Lite takes the following inputs to calculate the final results of emissions and fuel consumptions:

a) Second-by-second vehicle operating parameters (speed and acceleration);

b) Operating mode bin table;

c) Emission rate table;

d) Base cycle rates from MOVES.

MOVES Lite first calculates the second-by-second VSPs based on the corresponding vehicle operating parameters. Using a combination of calculated VSPs and the vehicle speed, the calculation process then looks up the operating model bin table to obtain the associated operating modes. This is followed by another table lookup at the vehicle emission rate table based on operating mode, vehicle type and age. The emission and fuel consumption are accumulated and corrected with the base cycle emission rates from MOVES. Additional reporting mechanisms are applied to generate the final emission reports.

5.4 Numerical Experiments

The numerical experiments are performed with a real-world network in this section. The goal is to demonstrate how this integrated framework can be applied to assess the travel time and emissions influenced by different traffic conditions.

5.4.1 Test Network Description

The test network used in this study is the Fort Worth network (TRB, 2010), as shown in Figure 5.19. This network has 180 nodes, 447 links and 148 zones. The major traffic is carried by a 5-mile north-south freeway corridor (blue lines in Figure 5.19), which has two 4 general lanes and 1 HOV (High Occupancy Vehicle) lane each way. $1 and $1.5 tolls are charged for the SOVs (Single Occupancy Vehicle) and trucks, respectively, for using HOV lanes while no fee is charged for HOVs.

The purpose of the tests is to systematically evaluate the impacts of demand level and work zone on the travel time and the emissions.

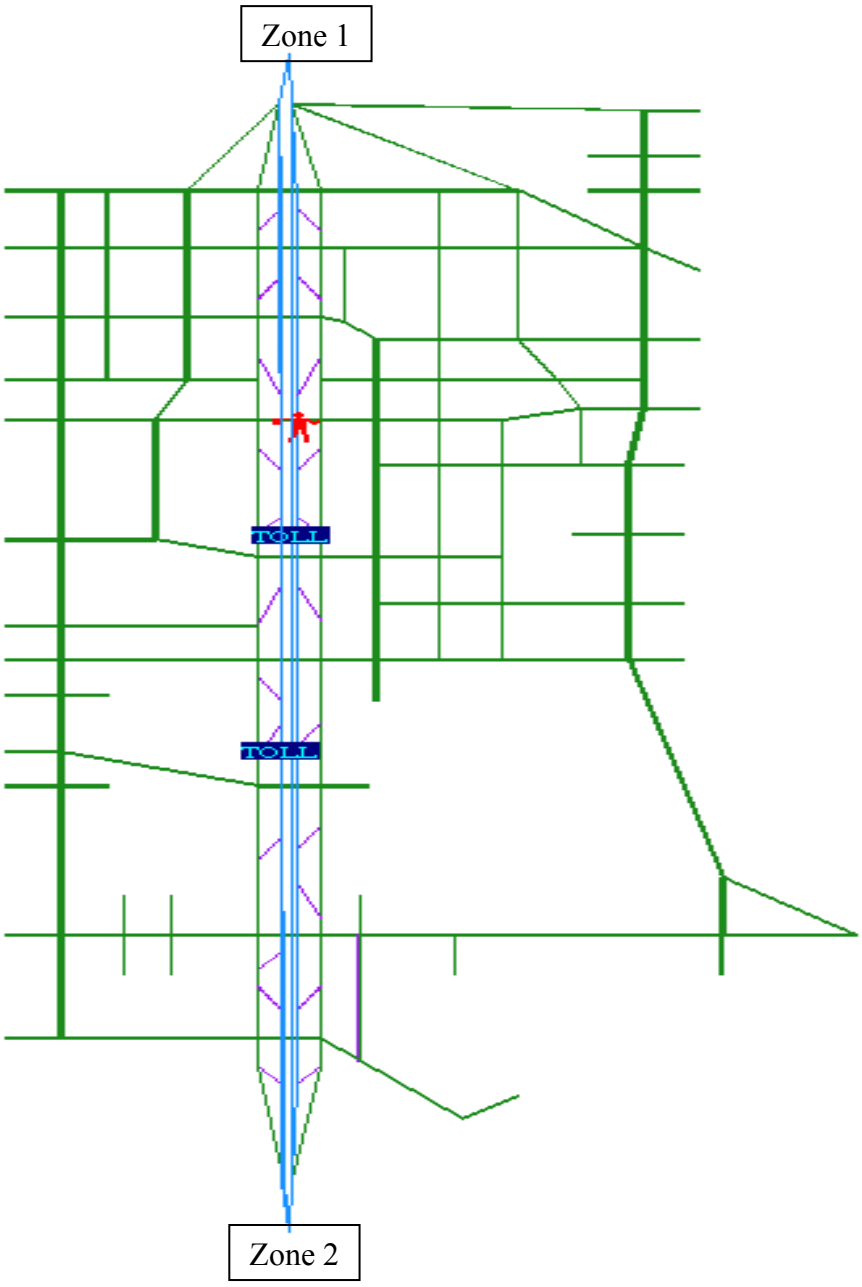


Figure 5.19: Fort Worth Network

5.4.2 Traffic and Emissions Impacted by Demands

In this test, 5 levels of demands are applied to the test network to investigate travel time and emissions impacts by travel demand. The simulation is performed between 7:00 AM and 9:00 AM for a total of 20 days. During these days, a day-to-day learning mechanism is applied so that the travelers adjust their routes based on the experienced travel time on the previous day.

The base demand level has a demand multiplier 1, while the demand multiplier 1.28 means the total number of vehicles in this demand level is 1.28 times of the base demand level. The numbers of vehicles at each demand level are listed in Table 5.9.

Three general types of vehicles, SOV, HOV and truck, are modeled in the test. The same vehicle distribution is applied to all demand levels, which are:

- 80% of SOVs, consisting of 80% of passenger cars, 20% of passenger trucks
- 10% of HOVs, consisting of 80% of passenger cars, 20% of passenger trucks
- 10 % of trucks, consisting of 30% of light commercial trucks, 30% of single-unit long-haul trucks and 40% of combination long-haul truck

The detailed proportion of individual vehicles is listed in Table 5.10.

Along with the traffic time, the following emission factors are examined in this test:

- Energy consumption

Table 5.9: Fort Worth Demand Levels

Demand Multiplier	Number of Vehicles
1	41,100
1.07	44,040
1.14	46,985
1.21	49,897
1.28	52,839

Table 5.10: Percentage of Vehicle Composition

Vehicle Type	**Percentage**
Passenger Car	72%
Passenger Truck	18%
Light Commercial Truck	3%
Single Unit Long-haul Truck	3%
Combination Long-haul Truck	4%

- CO2
- NOX (Nitrogen Oxides)
- CO
- HC (Hydro-Carbon)

As expected, the average speed is reduced from 47.6 mph to 40.5 mph as the demand level increases from 1 to 1.28, while the travel time index increases from 1.28 to 1.47 (Figure 5.20). Travel time index, defined as the ratio between the actual travel time to the free-flow travel time, has been widely used as a congestion indicator, while travel time index of 1 means no significant traffic congestion.

From Figure 5.21, it can be observed that due to the increase of the demand, the energy consumption and emission levels increase significantly, especially when the total demand approaches the network capacity.

So far we have demonstrated the changes of network-level travel time index, emission and fuel consumption rate in response to the change of demand level. It is still of particular interest to examine the changes associated with different vehicle types. The results from the demand levels of 1 and 1.28 are used for comparison.

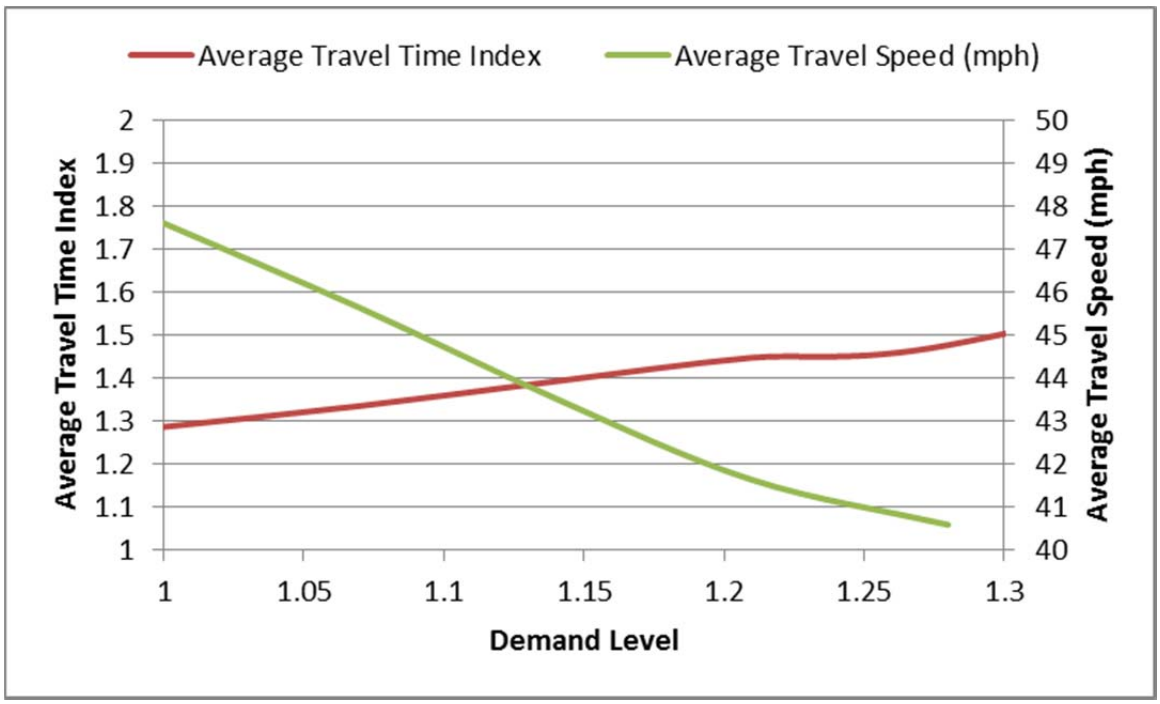


Figure 5.20: Network Performance under Different Demand Levels

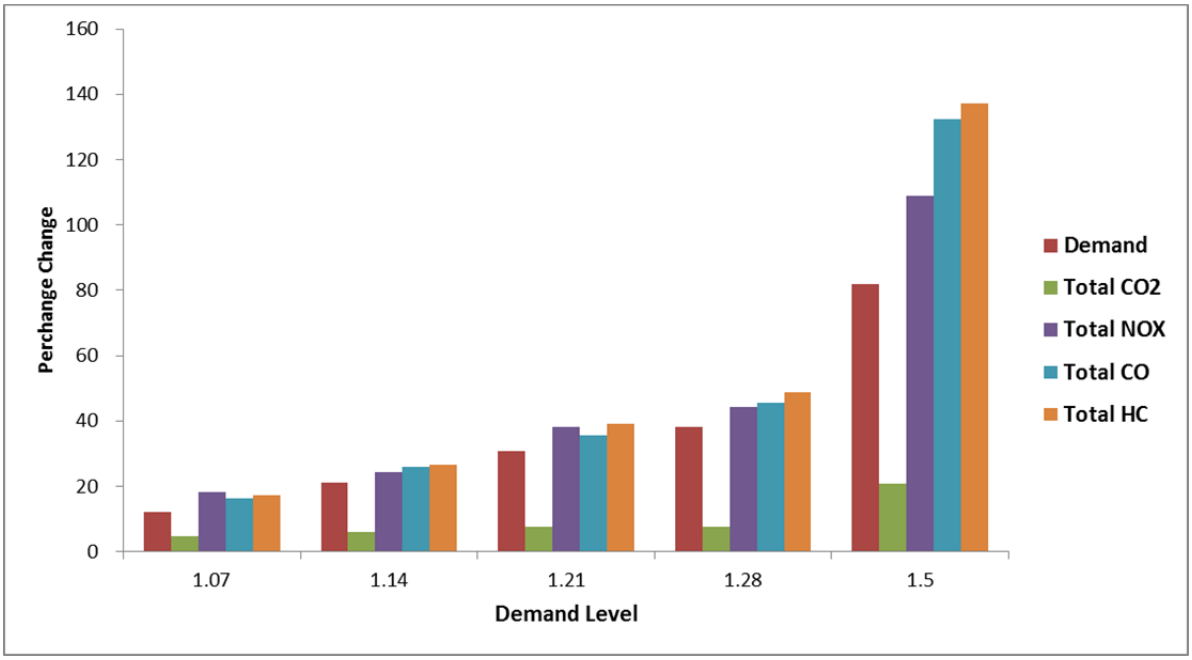


Figure 5.21: Percentage Change of Total Energy and Emissions

Due to the increase of travel demand, the average travel speed for all three vehicles is reduced, as shown in Table 5.11, but with different change rate. Among these vehicle types, HOV has less travel speed reduction rate.

It is interesting to notice in Table 5.12 that the average emission rates are reduced as the demand level increases.

The possible explanation for this may be:

1. All the HOV vehicles travel in the HOV lane.
2. The traffic in the HOV lane is stable due to the limited accessibility (one entrance and one exit). This implied no significant variation of travel speed and the VSP is only influenced by the travel speed, for a given vehicle type.
3. According to Eq. (5.1), given the vehicle type, the lower the speed, the lower the VSP.
4. The speeds of the consisting vehicles (passenger car and passenger trucks) are between 25 mph and 50 mph. According to Table 5.2, the range of operating modes is within 20 to 30.

Table 5.11: Average Travel Speeds at Different Demand Level

	Demand Level		% Change
	1	1.28	
SOV	47.46	40.16	-15.4%
HOV	49.01	44.85	-8.5%
Truck	47.30	40.21	-15.0%

Table 5.12: Percentage Change of Average Emission Factors

	Avg. CO2 (g/mi)	Avg. NOX (g/mi)	Avg. CO (g/mi)	Avg. HC (g/mi)
SOV	11.9%	6.0%	5.8%	7.4%
HOV	-3.4%	-10.8%	2.2%	-0.9%
Truck	3.9%	5.4%	6.9%	10.4%

5. Based on average emission rate tables, such as Table 5.3, the average emission rates within this operating mode range decrease when the value of operating mode decreases. As a result, when the HOV speed reduces, the emission rates decrease.

Unlike the HOV lane, the traffic states on other facilities are not that stable. In addition, when SOVs and trucks move from one type of facility to another, for example, from a ramp to the freeway corridor, there is significant speed variation between the facilities. Under these unstable situations, the VSP and operating mode highly depends on both speed and speed variation. This may explain why the average speed is reduced but the average emission rates increase for SOVs and trucks.

In addition, if we further distinguish the vehicle types, we have observed that different vehicles have different increase rates for each pollutant, as shown in Table 5.13. This variation of increase rates is jointly determined by vehicle type, traffic state and facility.

5.4.3 Work Zone

The goal of this test is to examine the impact of the work zone on the travel time and the emissions. As shown in Figure 5.19, two of the four lanes of a northbound freeway link with a length of 0.413 miles are closed on day 21 due to the maintain requirement.

Table 5.13: Percentage Change of Average Emission Factors by Vehicle Types

Vehicle Type	Avg. CO2 (g/mi)	Avg. NOX (g/mi)	Avg. CO (g/mi)	Avg. HC (g/mi)
Passenger Car	9.75%	2.03%	5.09%	4.98%
Passenger Truck	9.37%	6.04%	6.59%	10.40%
Light Commercial Truck	10.15%	4.24%	1.71%	1.51%
Single Unit Long-haul Truck	6.73%	2.69%	9.20%	8.58%
Combination Long-haul Truck	3.13%	10.33%	17.24%	16.59%

The day-to-day simulations are carried out until the average travel time index is stable and this is shown in Figure 5.22. Before the work zone starts, the average travel time index is stabilized at a level of 1.28. Due to an introduction of the work zone, the averagetravel time index surges up to 5.4 at day 21. With the day-to-day learning mechanism, the simulator is able to divert travelers to alternative routes until the average travel time index reaches another stabilized value at 1.36.

As suggested by Figure 5.19, the major traffic impacted by the work zone is the north bound traffic from zone 2 to zone 1. Therefore, the average travel time and emission rates between these two zones are compared at different days.

Table 5.14 further illustrates that as the result of the work zone, there are significant increases of emissions between zone 2 and zone 1. However, the emission rates decrease due to the day-to-day learning mechanism. The same trend is observed at the link where the work zone is present, as shown in Table 5.15.

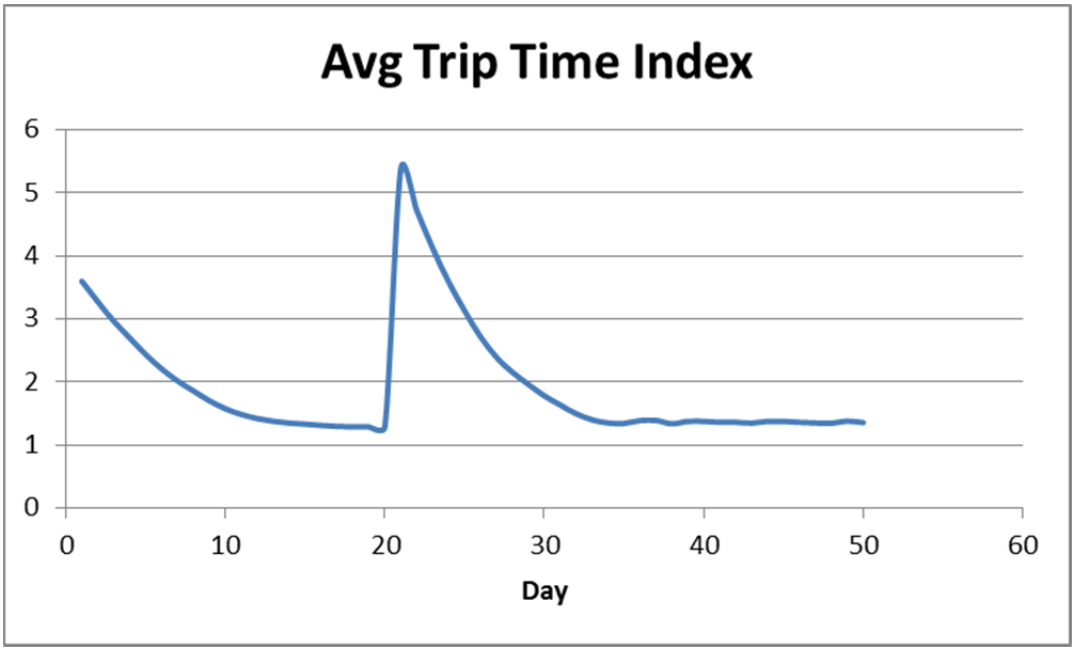


Figure 5.22: Average Trip Time Index with Work Zone Setup

Table 5.14: Percentage Change of Average Emission Factors with Work Zone - OD

Day	% Change of Avg. CO2	% Change of Avg. NOX	% Change of Avg. CO	% Change of Avg. HC
25	260%	230%	200%	260%
30	110%	110%	100%	110%
50	70%	70%	70%	70%

Table 5.15: Percentage Change of Average Emission Factors with Work Zone - Link

Day	% Change of Avg. CO2	% Change of Avg. NOX	% Change of Avg. CO	% Change of Avg. HC
25	670%	600%	570%	690%
30	630%	540%	530%	650%
50	400%	420%	430%	450%

CHAPTER 6

CLOUD COMPUTING-BASED ARCHITECTURE FOR ADVANCED TRAFFIC INFORMATION DISSEMINATION

This chapter focuses on the discussion of a demonstrative implementation of a cloud computing-based advanced traveler information system which utilizes the large volume of heterogeneous data sources to provide reliability-based routing information. The chapter is organized as follows. Section 6.1 presents the background information of this topic. Section 6.2 briefly introduces the cloud computing and its benefits. Section 6.3 describes the system architecture of the proposed system. Special attention is given to the MapReduce-based reliable routing engine in Section 6.4. The system test is summarized in Section 6.5.

6.1 Introduction

Over the last few decades, federal, state and local agencies have deployed many traffic sensor systems to monitor and manage freeway and arterial networks. With these installed sensors (e.g., widely-used inductive loop detectors) valuable traffic measurements are collected, processed and further disseminated to travelers and system managers to make informed decisions. Recently, emerging techniques, such as Automatic

Vehicle Identification (AVI) and Global Positioning Systems (GPS), have been widely incorporated into network monitoring, signal control, toll collection and other traffic applications. Other wireless communication technologies, such as Bluetooth and WiMax, have also been applied to monitor the point-to-point travel time on highways and arterials. Consequently, transportation planning and management agencies have found a critical need for efficiently storing, processing and extracting desirable corridor-level and network-level information through a systematic and seamless integration of these data sources.

Thanks to the advances of telecommunication and information technologies, the emerging practice of Cloud Computing provides a revolutionary solution platform to combine data archiving and information extraction processes. It represents a fundamentally new approach and opportunity for large-scale system monitoring, data archiving and information fusion in transportation planning and operations. In a cloud computing environment, the computational power and data storage are integrated into a unified platform. The utility computing model employed by cloud computing service providers dramatically reduces the cost and complexity of managing computer servers, datacenters and communication networks. The utility computing model is analogous to how traditional utility services, such as water and electricity, are operated – by consumption. This provides cloud computing users with the ability to choose customized computational power and storage space and pay for only what they consume. The Internet-based nature of cloud computing enables distributed storage, universally-accessible data and flexible client-server application implementations. In essence, cloud computing provides a heterogeneous, robust and open distributed computing platform.

Recognizing the benefits of cloud computing, some researchers and engineers have incorporated cloud computing into the field of transportation engineering. To provide governments and consulting firms with software-as-a-service (SaaS) benefits, a web-based, hosted scalable platform for transportation modeling called Mint (i.e., Modeling on Internet) (Wang, 2010) has been developed, enabling faster result retrieval and cost reduction. Li et al. (2011) have used intelligent traffic clouds in their prototype urban-traffic management system to overcome the enormous computing and power resource requirements imposed by the large-scale use of mobile agents.

In this paper, the applicability of the cloud computing technique is demonstrated, particularly by addressing the following three key challenges, to accommodate demanding traffic information requirements of the general public and traffic operations and planning organizations.

6.1.1 Different Data Sources and Data Representations

Due to their associated low unit equipment cost and relatively high performance, inductive loops have become the predominant fixed vehicle detection device in the United States. Currently, Radio Frequency Identification (RFID) -based Automatic Vehicle Identification (AVI) systems are widely deployed in road pricing and parking lot management, as well as real-time travel time information provision. In the past several years, in-car navigation using Global Positioning System (GPS) technology has matured into a rapidly growing industry and its penetration rate in the U.S. was expected to exceed 10% in 2010. Data from different sources (that is, point, point-to-point, and semi-continuous data) usually have different representations in databases and require different

processing techniques. Thus, an ability to integrate valuable traffic information, such as travel time reliability information, from different data sources in different representations is critically needed.

6.1.2 Large Data Volume

Besides the heterogeneity of the data sources, the volume of the data received every day is a considerably large amount for storing and processing. For instance, the Southern California Region (covering Los Angeles and Orange Counties) with 5.3 million households and 7.1 million vehicles generate about 41.2 million motorized and nonmotorized trips every day. If the location data (in terms of longitude and latitude) are recorded every second for each trip (assuming 30-minute trips), about 0.6 TB of data will be generated in a single day.

There are always privacy concerns about the trip trace information collected by mobile devices, AVI and other vehicle identification techniques. Therefore, before these data can be used to extract valuable traffic information, personal information, such as license plate numbers, mobile device identification numbers and other information which can pinpoint to a specific person, must be deleted or replaced to eliminate the exposure.

As a result, methods for cost-effective and computationally-efficient storage and post-processing for these large volumes of privacy-sensitive data must be carefully examined and planned, especially for regional real-time travel information provision applications.

6.1.3 Data Sharing and Information Provision

Datacenters for storing collected traffic data by agencies are geologically distributed. Therefore, an effective data sharing platform is necessary to provide safe and transparent data access. Moreover, in order to support informed decisions, it is vital to have a flexible traffic information provision system to meet the different requirements of users and customers. In addition, this information provision system must be easily modified and upgraded to accommodate new and evolving needs.

6.2 Cloud Computing

6.2.1 Definition of Cloud Computing

As a new form of distributed computing (ITU, 2009), cloud computing is defined as a software model, as in Figure 6.1, in which shared and interconnected computing

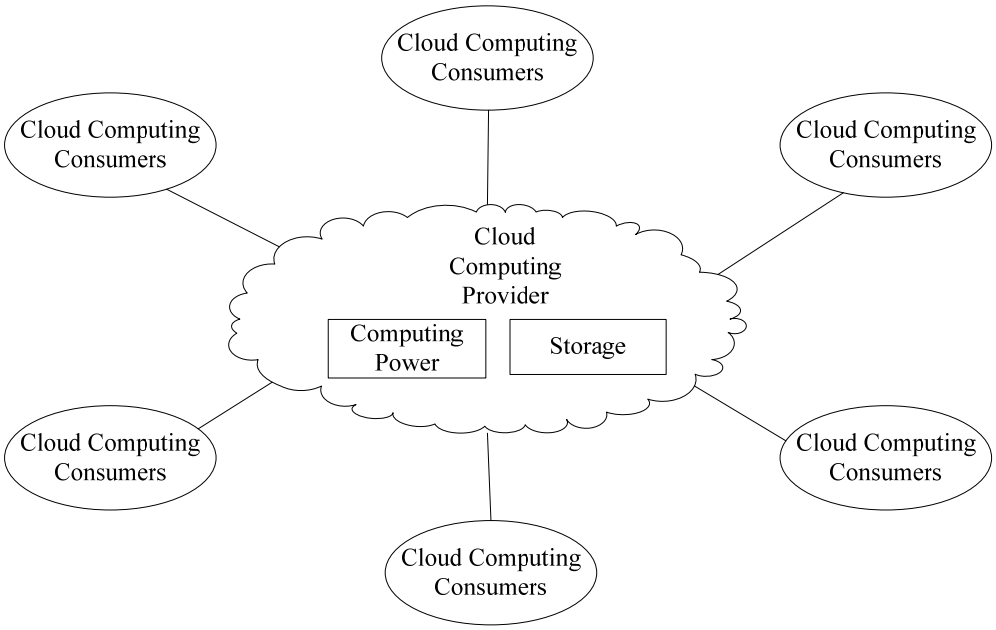


Figure 6.1: Cloud Computing Model

resources, e.g., networks, servers, storage, applications and services, can be rapidly and automatically acquired and released as an on-demand self-service (Chappell, 2009). In other words, computing power and storage space are shared and purchased by multiple customers as services. These computing resources can be automatically provisioned and released, with little to no human interaction with each service provider. Customers decide the initial storage space and computing power their applications require. Later, customers can dynamically adjust their needs whenever their applications require more or less space or power.

Cloud computing's major evolution from traditional distributed computing is that the underlying computing resources of cloud-based services and software are entirely abstracted from the consumers of the software and services. The consumers focus on the business developments and service definitions while the cloud service providers take responsibility for the performance, reliability and scalability of the computing environment.

6.2.2 Advantages of Cloud Computing

The benefits of selecting cloud computing as the underlying platform for an advanced information provision system are listed below.

6.2.2.1 Integrated Computing and Storage

As stated in its definition, the cloud computing model integrates computing power and storage. And these computing resources are abstracted from the cloud computing consumers, in this case, the advanced traveler information provider, which eliminates the

burdens of setting up hardware and software to store collected traffic information. With the help of the cloud computing providers, geologically distributed datacenters can be seamlessly accessed.

6.2.2.2 Scalable and Customized Computing Resources

Parallel to how people consume electricity power and other utilities, cloud computing resource customers do not need to understand the component devices or infrastructure required to provide the service. Computing power and storage are rented from the cloud computing service providers and can be provisioned and released as the demand for computing power and storage goes up and down. With this flexibility in computing environment, situations such as service breakdowns due to surging inquiries for traffic information from an advanced traveler information system during a severe weather event can be prevented. Furthermore, costs can be reduced when unnecessary resources are released.

6.2.2.3 Performance, Security and Maintenance

Cloud computing service providers have already addressed many vital performance and security issues faced by common web application designs. For example, load balancing is a decisive issue related to application performance, and its implementation requires specific expertise and experience, of which smaller organizations and agencies often lack. Fortunately, many cloud computing platforms have implemented built-in load balancers which help to preserve users' precious time and resources so that they can focus on business delivery. Prominent cloud computing service providers, such as

Microsoft, Amazon and Google, have strong technical and supportive teams and can provide quick responses to potential security issues, thus assuring system security and integrity.

6.2.2.4 Ease of Information Provision

To eliminate the need to install and run the application on the customer's own computers and simplify maintenance and support, cloud application services, or Software as a Service (SaaS), deliver software as a service over the Internet. The software and its associated data are hosted centrally in the cloud while the service is accessed by users using a web browser.

This software delivery model is well-suited for a traveler information provision service since the most preferable way for users to access the traffic information is by a web browser. This web-based service model also enables service compatibility for multiple platforms, including Windows, Linux and Macintosh computers, and any other mobile devices with web browsing capabilities. In addition, a traveler information provision service implemented as a cloud service also circumvents distributed software updating and patching as all core components reside on the server side.

In summary, due to the above-listed benefits, cloud computing is a very appropriate, reliable information provision platform for agencies. Its open characteristics also encourage data sharing with other agencies and private partners.

6.2.3 MapReduce as a Specific Cloud Computing Software Framework

The following discussions aim to introduce a specific cloud computing model, MapReduce, to design data-intensive software systems for managing and manipulating a large volume of data. MapReduce was introduced by Google Inc. in 2004 for processing huge datasets on a cluster of computers. It is derived from the *map* and *reduce* combiners from a functional language like Lisp.

In a typical MapReduce model, as shown in Figure 6.2, a computational process takes in a set of input key/value pairs and produces a set of output key/value pairs (Dean and Ghemawat, 2004). The computation consists of two user defined functions: *Map* and *Reduce*.

Map takes an input key/value pair and produces a set of intermediate key/value pairs. All intermediate values associated with the same intermediate key are grouped together by the function provided MapReduce library and passed to the user-defined *Reduce* function.

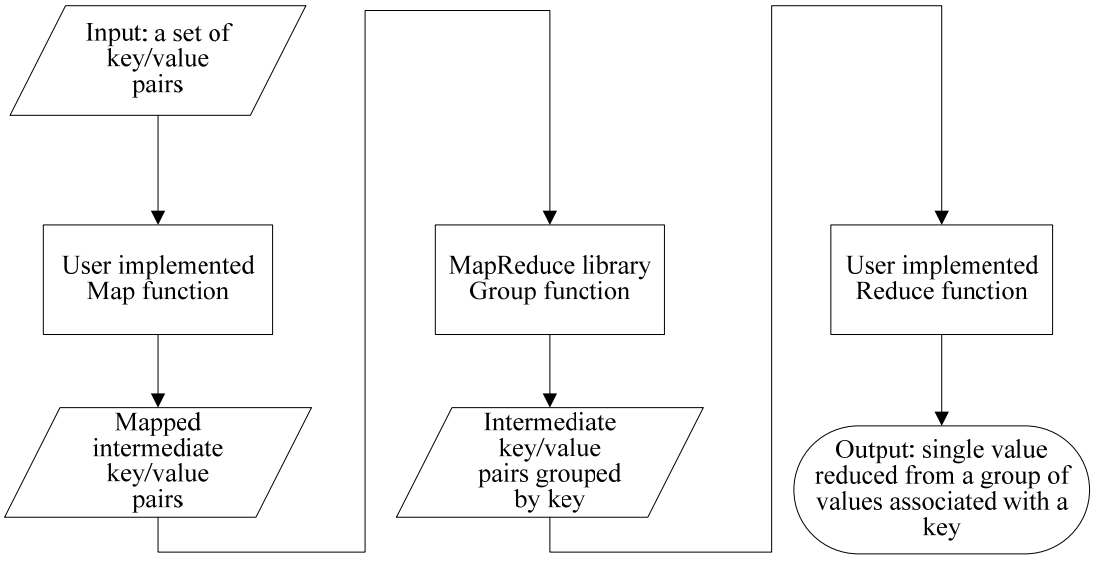


Figure 6.2: A Typical MapReduce Computation Process

The *Reduce* function accepts an intermediate key and a set of values associated with that key. These values are then merged together to generate an output of a smaller set of values.

The strength of MapReduce is that it allows for distributed processing of the map and reduction operations. If each mapping operation is independent of the others, all maps can be executed in parallel – though practically it is limited by the data source and/or the number of CPUs or processing units available. Similarly, a set of reducers can perform the reduction phase – all that is required is that all outputs of the map operation that share the same key are presented to the same reducer at the same time. While this process can often appear inefficient compared to algorithms that are more sequential, MapReduce can be applied to significantly larger datasets. Additionally, this process also provides the possibility of recovering from partial servers or storage failure during the operation. For instance, if one mapper or reducer fails, the work can be rescheduled, as long as the input data are still available.

The MapReduce model has been adopted by Google for processing large amounts of crawled documents. Their experience demonstrates that a great performance has been achieved in a large amount of data (Dean and Ghemawat, 2004).

Meanwhile, although MapReduce was originally introduced by Google, there are several open source MapReduce frameworks, such as Apache Hadoop MapReduce and Disco. MapReduce libraries have been implemented in a number of programming languages, including C++, Java, Python, C# and so on, which gives the developers a number of choices to integrate MapReduce into their existing systems.

Recently, Microsoft introduced a new scripting language, SCOPE (Structured

Computations Optimized for Parallel Execution) for large-scale data analysis. The target of SCOPE is to provide easy and efficient parallel processing of massive data sets. SCOPE is a high level declarative scripting language and has a strong resemblance to SQL. Thus, users with experience with rational database and SQL require little or no specific training to use SCOPE.

It has been noted that our research is to design and implement a travel time reliability-based traveler information provision service over the cloud computing platform. Compared with the MapReduce model, there are few real-world application implementations with SCOPE on the cloud as SCOPE is a still under development by Microsoft. Besides, the MapReduce programming model provides a well-defined map and reduce programming model, which is more suitable for implementation of the underlying travel time reliability engine in our system.

6.3 System Architecture and Data Flow

The system architecture of this travel time reliability-based traveler information provision system is shown in Figure 6.3 and Section 6.3.1 introduces the major modules and data flow among them; Section 6.3.2 briefly presents the client design considerations; Section 6.3.3 demonstrates how the MapReduce model is adapted to calculate the travel time reliability.

6.3.1 Server-side Components

As shown in Figure 6.3, the server is composed of four core components:

- Traffic measurement and historical pattern databases

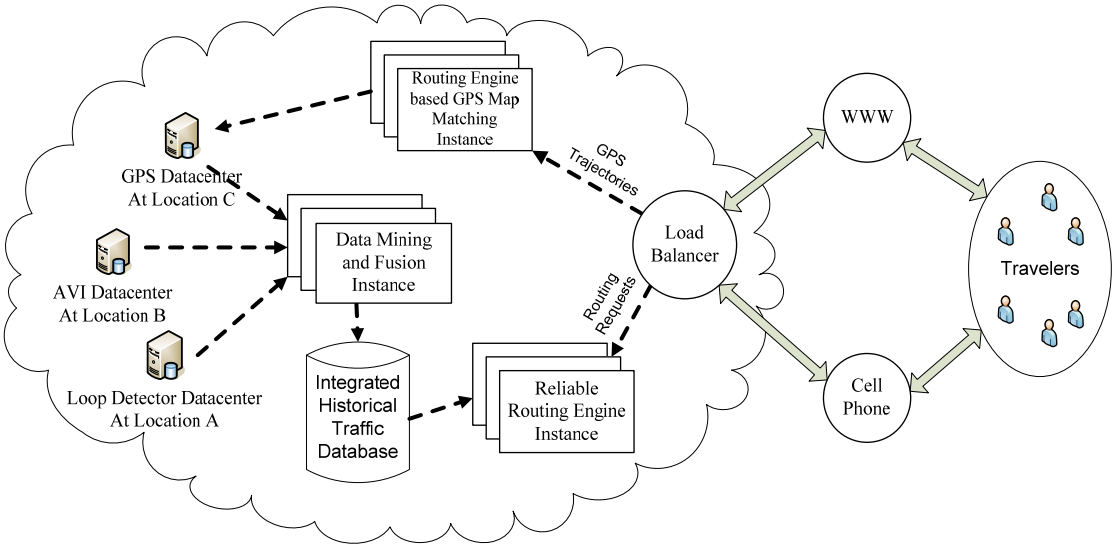


Figure 6.3: Cloud Computing-supported System Architecture

- GPS map matching engine
- Data mining and fusion engine
- Reliable routing engine

6.3.1.1 Databases

It is desirable to store a large volume of loop detector, AVI and GPS trace data in distributed rational databases. The physical locations of these datacenters are transparent to the users and traffic system managers. Possible access interfaces, accordingly, must be provided to allow seamless information sharing among commuters, public-sector agencies and private traveler information providers. A cloud computing storage system offers new opportunities for potential public-private partnerships in sharing and trading different data streams.

6.3.1.2 GPS Map Matching Engine

Map matching is an essential data processing service that converts raw GPS location data samples to useful traffic information in node-link traffic network representation form. User-provided GPS trajectories are processed in this module and fed into the subsequent data mining and fusion tasks.

6.3.1.3 Data Mining and Fusion Engine

The data mining and fusion module extracts and integrates valuable end-to-end trip travel time variability information from various sources. In this engine, data from loop detectors, GPS, AVI and other sources are integrated into a single historical database, which is then queried and manipulated by the reliable routing engine.

6.3.1.4 Reliable Routing Engine

As the building block of the entire traffic information provision system, the reliable routing engine calculates routes under different criteria, based on live traffic data from the traffic data fusion and prediction engine, and further generates final route guidance information to end users. Upon receiving user request for the reliable path between a user-specified origin and destination (OD) pair, this engine first generates a list of alternative paths between this OD pair. After that, path travel time statistics for each alternative path are calculated by a MapReduce mechanism. Detailed descriptions of MapReduce-based reliable routing are presented in Section 6.4.

6.3.2 Client-side Components

To make the service available to the greatest number of browsers, including mobile browsers on the smart phones, the computing requirements on the client-side are designed to be limited to the minimum. The main functions of client-side code are displaying maps/results and accepting user input while all the computations are carried out at the server-side. The client-side code is implemented with AJAX to provide interactive web pages.

As shown in Figure 6.4, the left side of the page is designed to display a route network with Microsoft Bing Maps. Microsoft Bing Maps' Interactive SDK for AJAX v6.3 is utilized to provide user interaction with the map and display the routes. In order to request path travel time, users must first specify their origin and destination of a path. Two flexible input methods are supplied: 1) users can select their origin and destination on the map; or 2) users can input addresses for their origin and destination in the textboxes. Once a user successfully provides the origin and destination information, a travel time reliability request is sent to server.

Five possible routes are calculated on the server and returned back to the client for display: quickest (in time) route, shortest (in distance) route, eco (most environmentally-friendly) route, safest route and detour for each origin and destination. Computed route information includes individual route travel times, route travel distances, safety indices and route coordinates.

The calculated results are processed in the client-side browser with embedded JavaScript, including displaying alternative routes on the map, presenting detailed

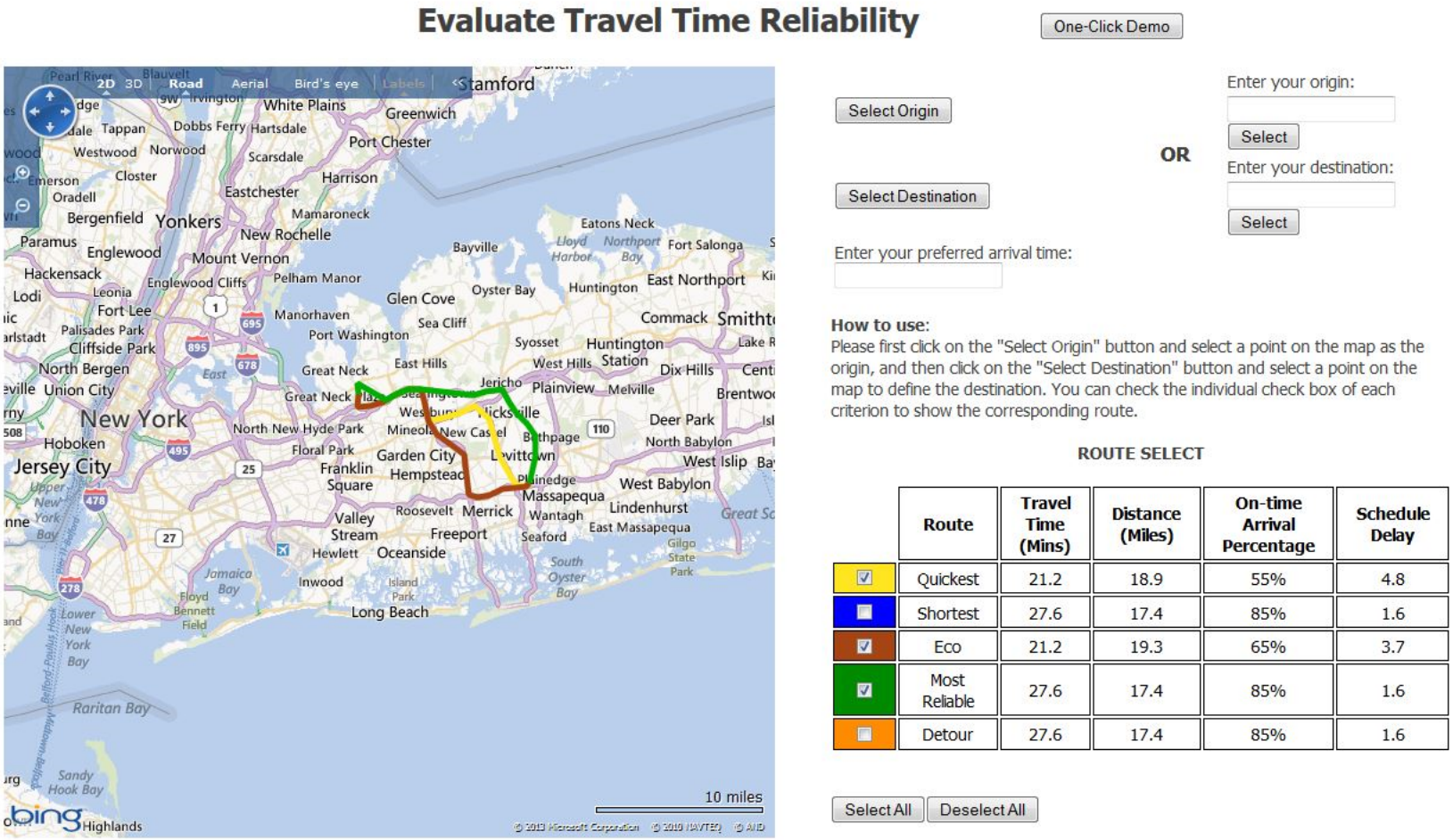


Figure 6.4: Travel Time Reliability Information Provision Interface

information about the alternative routes and displaying route travel time reliability. Bing Map's Interactive SDK provides rich APIs for customized route display.

As an example presented in Figure 6.4, there are 3 alternative routes between origin A and destination B, because 1) the quickest and shortest routes are the same routes, and 2) the eco and safest routes are the same as well. Average travel times, distances and safety ratings for each route are displayed on the right table. Daily travel time variability for each route is also displayed as an upper and lower bound on the rightmost bottom side of the display. This is based on the computation result generated for this specific routing request.

6.4 MapReduce-based Travel Time Reliability Engine

To fully utilize the parallel computing capability and improve system performance, a MapReduce-based reliable routing engine is implemented on the server-side. Our MapReduce-based travel time reliability engine is implemented under the Microsoft Daytona MapReduce framework, which is an iterative MapReduce runtime targeted for the Microsoft Azure cloud computing platform. Like in the other MapReduce framework, Map and Reduce functions are the key building blocks of a single Windows Azure service in the Daytona MapReduce framework.

In addition, a controller class is another critical module in the Daytona framework. This module is implemented to handle the details of computation job configuration, task (Map or Reduce) submission and management within the Daytona runtime. Typical computation job configuration settings include storage locations for input and output, and the maximum number of map and reduce tasks that can be executed in parallel. The

maximum number of the map and reduce tasks are directly related to the overall performance of the system. These settings are defined as configuration files and provided during the deployment. The Daytona Developer Guide provides detailed design guidance on the interfaces that must be implemented for each of these building blocks and other necessary functionalities.

Figure 6.5 shows the overall flow of the travel time reliability calculation operation upon a user request, implemented with MapReduce framework.

The MapReduce-based travel time reliability calculation process is summarized in the following sequence:

i. The input of the travel time reliability engine is a list of key/value pair [K1, V1], where the key K1 is a string to identify a user-specified origin and destination (OD) pair and the value V1 is the specified departure time.

ii. The first map function maps an OD pair to an intermediate key/value pair [K2, V2], where the key K2 is a string including the consisting link information of an alternative path from O to D and the value V2 is the departure time.

Notice that, in step 2, the first map function maps a given origin-destination pair into a set of alternative routes, each of which is assigned a unique intermediate key. In step 3the second map function maps the alternative route travel time information request to a specific data source. The data sources include historical travel time data, real-time sensor data and GPS trace data. Thus, the appropriate travel time estimation methods can be applied with respect to different data sources. For example, 90^{th} or 95^{th} percentile travel time can be computed from a historical database, while a time-series-based method is more suitable for estimating travel time variability from real-time loop detector data. The

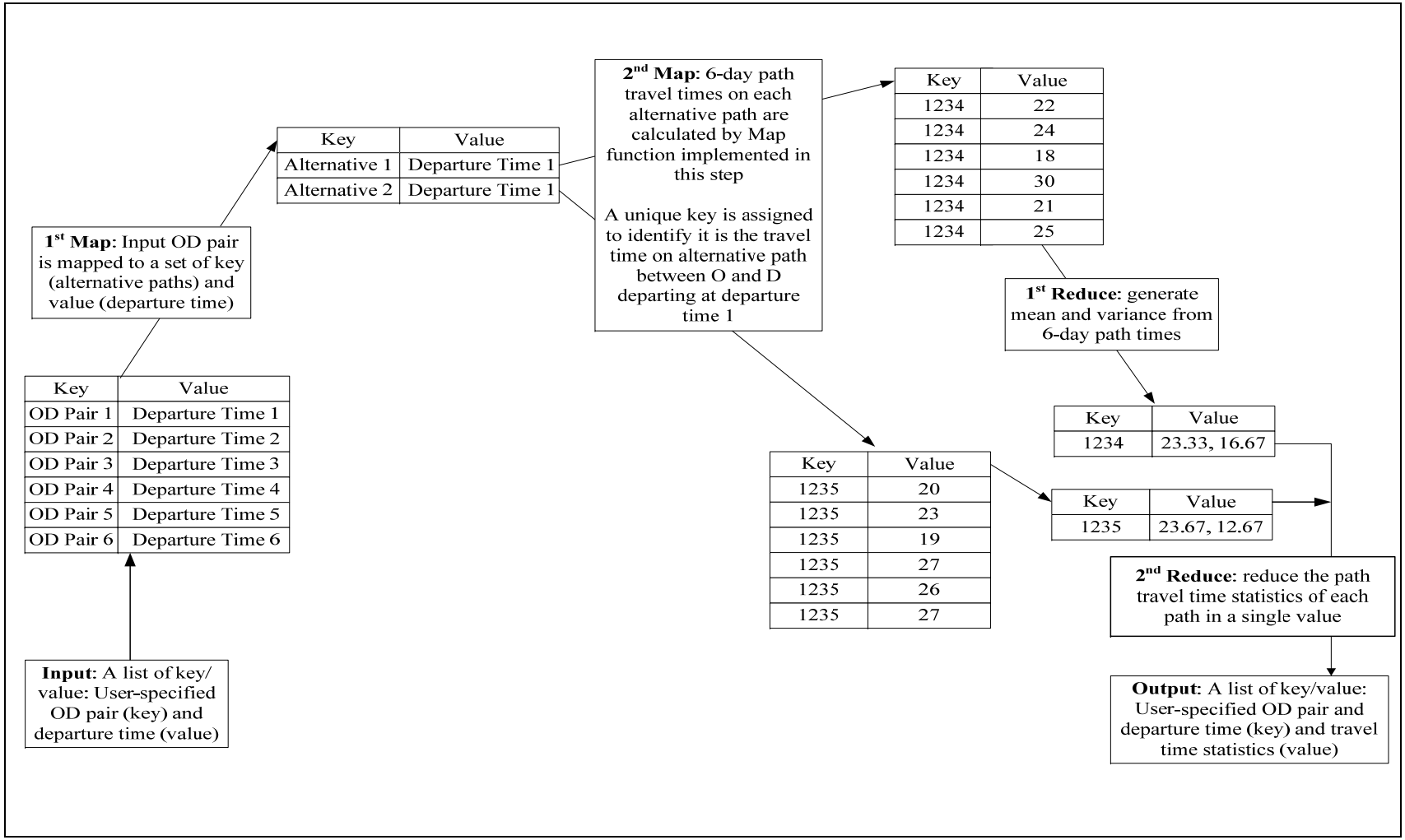


Figure 6.5: MapReduce-based Travel Time Reliability Calculation Process

estimated travel time reliability information for this alternative route is later grouped by the key assigned in step 2. Finally, the statistics for all alternative routes are further reduced to a single set of travel time statistics and returned to the user.

It has also been realized that computing the travel time statistics (the second map task) is a computing-intensive procedure and requires numerous database queries. Therefore, more computing resources are distributed to this task using the flexible system configuration mechanism provided by the Daytona framework.

6.5 System Test

In order to demonstrate the applicability of cloud computing on travel time reliability information provision service, a system performance test is conducted on a large-scale real-world transportation network at Bay Area, CA.

In Section 6.5.1, we briefly introduce the test environments, including the transportation network and server configuration adopted in the following system performance test. Section 6.5.2 presents the methods designed for the test. The preliminary test results are provided in Section 6.5.3. Further test and verification of our cloud computing-based system are provided in Section 6.5.4.

6.5.1 Test Environment

The test bed for the proposed cloud computing platform-based travel time reliability information provision system is the Bay Area, CA (Figure 6.6), which comprises 53,124 nodes and 93,900 links. The routing engine is coded in C# and deployed on the Microsoft Azure platform. The instance of the routing engine is equipped with a 1.6 GHz CPU and

Figure 6.6: Test Network: Bay Area, CA

1.75 gigabytes of memory, the smallest compute instance size of the four unique sizes provided by the Microsoft Azure platform.

6.5.2 Test Methods and Tools

Two key system performance indices of the traveler information provision service are (1) the average number of simultaneous user requests the server can handle per unit time and (2) the average response time per request. The incoming user request rate is assumed to follow a Poisson distribution to reflect the randomness of the user requests. User

requests are simulated by Oracle Load Testing for Web Application (OLTWA). Detailed performance test result figures are generated by OLTWA.

6.5.3 Test Results

The test result in Figure 6.7 shows that, after 3 or 4 minutes of system warming-up, the number of requests that the tested server can handle oscillates around 25 to 30 requests per second. Typically, if a user sends a request every 30 seconds, the computational instance capacity is 750 to 900 users per user request cycle.

The average response time for receiving and processing a user request is 1.23 seconds, as shown in Figure 6.8, which means the reliability information can be calculated by the proposed system in an acceptable time.

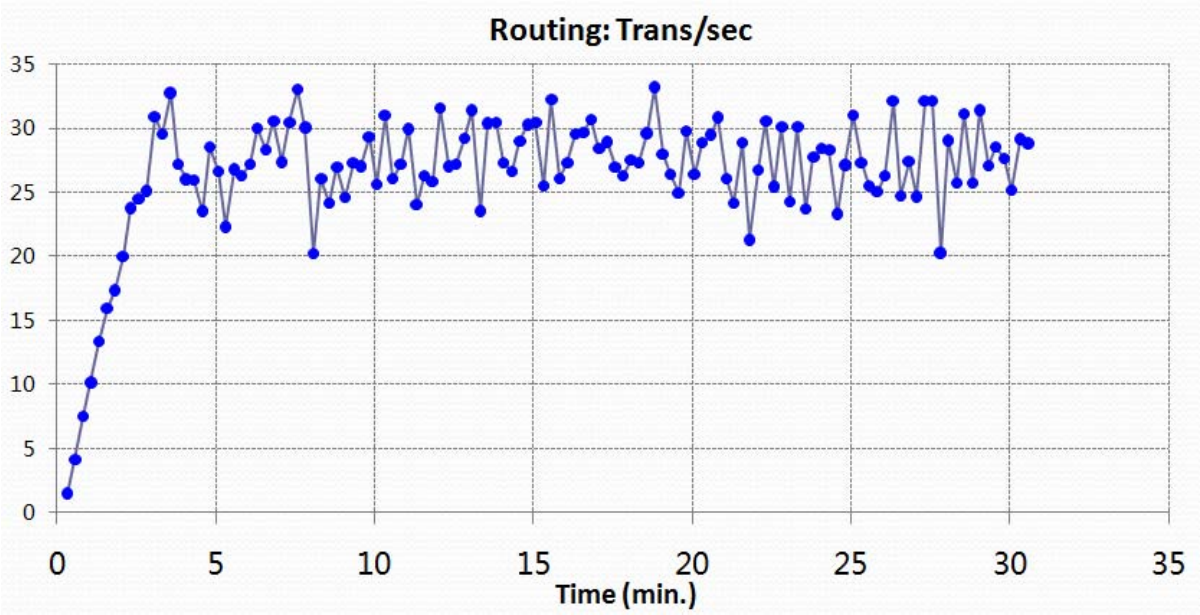


Figure 6.7: System Capacity Performance

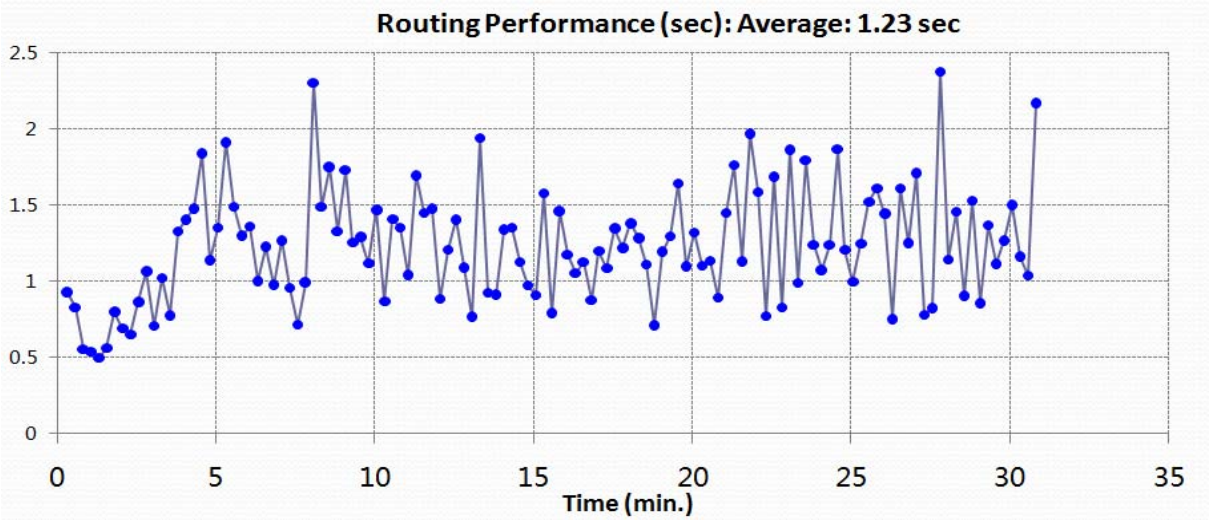


Figure 6.8: System Response Time Performance

6.5.4 Further Tests and Verification

The tests carried out in the previous section only show that our implementation can be applied to a large-scale network for a travel time reliability information provision service. However, more tests must be designed and performed to demonstrate that the cloud computing-based architecture can be a cost-effective choice over the traditional in-house dedicated server-based traveler information provision systems. For example, it is necessary to verify the cost-effectiveness of our cloud computing-based system to maintain the same system performance requirements as the dedicated server-based systems. Moreover, it is also necessary to compare the system performance with and without MapReduce implementation to investigate the advantages and disadvantages of adopting the MapReduce framework into the travel time reliability information calculation.

6.6 Concluding Remarks

Real-time traveler information provision service is a data-rich, computation-intensive application. If travel time reliability information needs to be provided along with the mean travel time, this will lead to increased storage and computational power requirements. The emerging cloud computing technique provides a new approach for large-scale system modeling, data storage and diverse data fusion and mining. Thus, the travel time reliability-oriented travel information provision system implemented on the cloud computing platform makes a good demonstration of the applicability of cloud computing in such traffic applications.

Particular attention is focused on how to incorporate revolutionary information techniques, such as MapReduce, to solve the challenges of large-scale real-time traveler information provision applications. The study described in this paper provides the following contributions to existing models and implementations:

1) A cloud computing-based system architecture which provides a unified data storage and computing platform to manage and manipulate large volumes of data.
2) A browser-based user interface providing a convenient user interaction.
3) A MapReduce-based reliable routing engine which incorporates the advanced technique in massive data processing.

CHAPTER 7

CONCLUSION AND FUTURE RESEARCH

7.1 Research Highlights

This dissertation presents several emerging issues in ATDM by addressing a series of important research questions, including (1) design a traffic time estimating methodology to quantify corridor level travel time and its distribution under stochastic supply and demand; (2) investigate cumulative flow count-based system modeling methods that estimate macroscopic traffic states with heterogeneous data sources on a freeway segment; (3) integrate the microscopic emission estimation model into the mesoscopic dynamic traffic assignment and simulation model to enhance the analyses of both the operating and environmental MOEs of the traffic control measures; and (4) implement a demonstrative cloud computing-based reliable routing engine.

7.1.1 Methodology to Quantify Corridor Level Travel Time and Distribution under Stochastic Supply and Demand

This dissertation has presented a travel time estimation methodology to quantify the corridor level travel time and distribution under both stochastic supply and demand. It does this on the basis of relatively simple information about the corridor's geometric configuration, its entering and exiting flows, its capacities and the lane-by-lane

distributions of traffic at each bottleneck. Monte Carlo simulation and mathematical approximation methods are used to calculate travel times lane-by-lane by tracking probe vehicles through the network. Ground-truth vehicle trajectory data from the NGSIM project have been used to validate the model's estimates. Several interesting observations are obtained from the research:

1) The model offers a theoretically sound method to estimate corridor-level end-to-end travel time and its distribution under different capacity and demand variations, and with possible on-ramp and off-ramp volume changes;
2) The variation of lane-specific traffic flow parameters (such as the number of vehicles on the lanes and lane discharge rates) are significant lane-by-lane travel time diversity.
3) A lane-level rather than link-level representation of the system is critical in developing accurate route-level travel time distribution estimates.

7.1.2 Cumulative Flow Count-based Traffic State System Modeling Method

It has been widely recognized that traffic state estimation is a complex nonlinear and stochastic estimation problem. By capturing the essential forward and backward wave propagation characteristics under possible random measurement errors, the proposed model offers a unified representation with the least complex explanation for traffic observations under free-flow, congested and dynamic transient conditions. This study also presented an information-theoretic approach to quantify the value of heterogeneous traffic measurements for specific fixed sensor location plans and market penetration rates

of Bluetooth or GPS flow vehicle data.

7.1.3 Integrate the Microscopic Emission Estimation Model into the Mesoscopic Dynamic Traffic Assignment and Simulation Model

To bridge the data gap between the microscopic emission model and mesoscopic dynamic traffic assignment and simulation model, Newell's simplified car-following is applied in this study to generate vehicle trajectories, from which second-by-second vehicle operating parameters can be derived. This greatly enhances the capabilities of the traffic simulation model on providing operational and environmental MOEs under both network and project levels for various traffic control measures.

7.2 Summary of Contributions

Major contributions in the study of corridor-level travel time variability include: (1) a unified modeling framework was developed to consider a corridor with multiple bottlenecks, and (2) a series of close-form formulas was derived to quantify the travel time distribution under both stochastic demand and capacity, with possible on-ramp and off-ramp flow changes.

For the problem of traffic state estimation, various sources of sensor data are incorporated with a cumulative flow counts-based traffic state estimation framework and the value of information is evaluated. A large-scale linear programming model is constructed and solved to overcome some of the inherent limitations of the existing methods, for example, difficulty to incorporate inequality constraints by Kalman

filtering-based methods.

By utilizing Newell's simplified car-following model, we seamlessly incorporated the microscopic and mesoscopic traffic models for more flexible evaluation of the policies and operating strategies.

7.3 Future Research

Our future research on corridor level travel time quantification problem will further consider the impacts of downstream queue spillback on the upstream travel time. Under queue spillback, the discharge rate of the upstream bottleneck is constrained by the downstream bottleneck discharge rate, and this significantly increases dynamics and complexity in estimating the capacity for a queuing model. Moreover, it is also desirable to examine the influence of lane change frequencies on the estimated number of vehicles waiting in the queue.

In the study of traffic state estimation problem, the future research will focus on the following three major aspects. First, the proposed single-segment-oriented methodology will be further extended for a corridor model with merges/diverges for possible medium-scale traffic state estimation applications. Second, the proposed model for the traffic state estimation problem can be further extended to a real-time recursive traffic state estimation and prediction framework involving multiple OD pairs with stochastic demand patterns or road capacities. Third, given the macroscopic state estimation results, one can quantify the uncertainty of other quantities in many emerging transportation applications, e.g., fuel consumption and emissions that are mainly dependent on cell-based or vehicle-based speed and acceleration measures; and link-based travel times that can be related to

the cumulative vehicle counts on the boundary.

Although the Newell's car-following model can be applied to construct vehicle trajectories and second-by-second speed and acceleration, there still exists unrealistic realization in the final acceleration and deceleration profiles. In addition, the simulated vehicle speed and acceleration is not sufficient to reflect the real-world speed/acceleration distribution. Therefore, the future research on this topic will continue on how to reconstruct close-to-reality vehicle trajectories to provide robust emission estimation based on the simulated vehicle trajectories from a mesoscopic simulator.

REFERENCES

Ahn, K., H. Rakha, A. Trani, and M. Van Aerde. Estimating Vehicle Fuel Consumption and Emissions Based on Instantaneous Speed and Acceleration Levels. *Journal of Transportation Engineering*, Vol. 128, No.2, 2002, pp. 182–190.

Ajax on Wikipedia. en.wikipedia.org/wiki/Ajax_(programming). Accessed Jul. 31, 2011.

Apache Hadoop MapReduce. hadoop.apache.org/mapreduce. Accessed Jul. 31, 2011.

Ashok, K., and M. Ben-Akiva. Alternative Approaches for Real-time Estimation and Prediction of Time-dependent Origin-destination Flows. *Transportation Science*, Vol. 34, No. 1, 2000, pp. 21-36.

Bing Maps. www.bing.com/maps. Accessed Jul. 31, 2011.

Microsoft Bing Maps Interactive SDK for AJAX v6.3. www.microsoft.com/maps/isdk/ajax. Accessed Jul. 31, 2011.

Cambridge Systematics, Inc. *NGSIM BHL Data Analysis. Technical report, September 2004. Summary Report*. Prepared for Federal Highway Administration.

Cambridge Systematics, Inc. *NGSIM I-80 Data Analysis (4:00 p.m. to 4:15 p.m.). Technical report, September 2005. Summary Report*. Prepared for Federal Highway Administration.

Cambridge Systematics, Inc. *NGSIM I-80 Data Analysis (5:00 p.m. to 5:15 p.m.). Technical report, September 2005. Summary Report*. Prepared for Federal Highway Administration.

Cambridge Systematics, Inc. *NGSIM I-80 Data Analysis (5:15 p.m. to 5:30 p.m.). Technical report, September 2005. Summary Report*. Prepared for Federal Highway Administration.

Chappell, D. *Introducing Windows Azure*. Chappell & Associates, San Francisco, CA, 2009.

Chen, C., A. Skabardonis, and P. Varaiya. Travel-time Reliability as a Measure of Service. In *Transportation Research Record: Journal of the Transportation Research*

Board, No. 1855, Transportation Research Board of the National Academies, Washington, D.C., 2003, pp. 74-79.

Clark, S. and D. Watling. Modeling Network Travel Time Reliability under Stochastic Demand. *Transportation Research Part B*, Vol. 39, No.2, 2005, pp. 119–140.
Cloud Computing on Wikipedia. en.wikipedia.org/wiki/Cloudcomputing. Accessed Jul. 31, 2011.

Coifman, B. Estimating Travel Times and Vehicle Trajectories on Freeways Using Dual Loop Detectors. *Transportation Research Part A: Policy and Practice*, Vol. 36, No. 4, 2002, pp. 351-364.

Cowan, R. J. Useful Headway Models. *Transportation Research*, Vol. 9, No. 6, 1975, pp. 371–375.

Cremer, M. and Papageorgiou, P. Parameter Identification for a Traffic Flow Model. *Automatica*, Vol. 17, 1981, pp. 837-843.

Daganzo, C. F. The Cell Transmission Model: a Dynamic Representation of Highway Traffic Consistent with the Hydrodynamic Theory. *Transportation Research Part B: Methodological*, Vol. 28, No. 4, 1994, pp. 269-287.

Daganzo, C. F. Properties of Link Travel Times under Dynamic Load. *Transportation Research*, Vol. 29, No. 2, 1995, pp. 95-98.
Daganzo, C. F. *Fundamental of Transportation and Traffic Operations*. Pergamon, Oxford, UK, 1997.

Daganzo, C. F. A Simple Traffic Analysis Procedure. *Networks and Spatial Economy*, Vol. 1, 2001, pp. 77-101.

Daganzo, C. F. A Variational Formulation of Kinematic Waves: Solution Methods. *Transportation Research Part B: Methodological*, Vol. 39, pp. 10, 2003, pp. 934-950.

Daganzo, C. F. A Variational Formulation of Kinematic Waves: Basic Theory and Complex Boundary Conditions. *Transportation Research Part B: Methodological*, Vol. 39, No. 2, 2005, pp. 187-196.

Daganzo, C. F. In Traffic Flow, Cellular Automata=Kinematic Waves. *Transportation Research Part B: Methodological*, Vol. 40, No. 5, 2006, pp. 396-403.

Daytona – Developer Guide. research.microsoft.com/en-us/downloads/cecba376-3d3f-4eaf-bf01-20983857c2b1/default.aspx., Accessed Nov. 2, 2011.

Dean, J. and S. Ghemawat. *MapReduce: Simplified Data Processing on Large Clusters*, Sixth Symposium on Operating System Design and Implementation, San Francisco, CA, December, 2004.

Deng, W., H. Lei and X. Zhou. Traffic State Estimation and Uncertainty Quantification based on Heterogeneous Data Sources: A Three Detector Approach. Manuscript Submitted for Publication, 2013.

Disco. discoproject.org. Accessed Jul. 31, 2011.

Next Generation SIMulation Fact Sheet. Technical Report. Publication FHWA-HRT-06-135. FHWA, U.S. Department of Transportation, 2006.

Recurring Traffic Bottlenecks: A Primer Focus on Low-Cost Operational Improvements. Technical Report. Publication FHWA-HOP-09-037. FHWA, U.S. Department of Transportation, 2009.

Frey, H. C., K. Zhang and N. Rouphail. Fuel Emissions Comparisons for Alternative Routes, Time Day, Road Grade, and Vehicles Based on In-use Measurements. *Environmental Science and Technology*, Vol. 42, 2008, pp. 2483-2489.

Frey, H. C., N. M. Rouphil, H. Hu, B. Liu, and X. Zhou. *2011 Progress Report: Framework for Context-Sensitive Spatially- and Temporally-Resolved Onroad Mobile Source Emission Inventories*. U.S. EPA- R834550, 2011.

Gazis, D. C., and M. W. Szeto. Application of Kalman Filtering to the Surveillance and Control of Traffic Systems. *Transportation Science*, Vol. 6, No. 4, 1972, pp. 419-439.

Giuliano, G. Incident Characteristics, Frequency, and Duration on a High Volume Urban Freeway. *Transportation Research A: Policy and Practice*, Vol. 23, 1989, pp. 387–396.

Godunov, S.K. A Difference Scheme for Numerical Solution of Discontinuous Solution of Hydrodynamic Equations, *Mathematics of the USSR. Sbornik*, 47(89), 1959, pp. 271-306.

Golob, T. F., W. W. Recker, and J. D. Leonard. An Analysis of the Severity and Incident Duration of Truck-Involved Freeway Accidents. *Accident Analysis and Prevention*, Vol. 19, No. 5, 1987, pp. 375–395.

Greenberg, I. The Log-normal Distribution of Headways. *Australian Road Research*, Vol. 2, No.7, 1966, pp. 14-18.

Guo, F., H. Rakha, and S. Park. (2010) Multistate Model for Travel Time Reliability. In *Transportation Research Record: Journal of the Transportation Research Board*, No. 2188, Transportation Research Board of the National Academies, Washington, D.C., 2010, pp. 46-54.

Hazelton, M. L. Inference for Origin–Destination Matrices: Estimation, Prediction and Reconstruction. *Transportation Research Part B: Methodological*, Vol. 35, No. 7, 2001, pp. 667–676.

Herrera, J. C., and A. M. Bayen. Incorporation of Lagrangian Measurements in Freeway Traffic State Estimation. *Transportation Research Part B: Methodological*, Vol. 44, No. 4, 2010, pp. 460-481.
Hurdle, V. F., and B. Son. Road Test of a Freeway Model. *Transportation Research Part A: Policy and Practice*, Vol. 34, No. 7, 2000, pp. 537-564.

Hurdle, V. F., and B. Son. (2001). Shock Wave and Cumulative Arrival and Departure Models. In *Transportation Research Record: Journal of the Transportation Research Board*, No. 1776, Transportation Research Board of the National Academies, Washington, D.C., 2001, pp. 159-166.

International Telecommunication Union. *Distributed Computing: Utilities, Grids & Clouds. ITU-T Technology Watch Report 9*. 2009 .

Kai., N., and M. Schrekenberg. A Cellular Automaton Model for Freeway Traffic. *Journal de Physique* I 2, 1992, pp. 2221-2229.

Kripalani, A., and W. Scherer. (2007). *Estimating Incident Related Congestion on Freeways Based on Incident Severity*. Research Report No. UVACTS-15-0-113. 2007.

Kwon, J., T. Barkley, R. Hranac, K. Petty, and N. Compin. (2011) Decomposition of Travel Time Reliability into Various Sources. In *Transportation Research Record: Journal of the Transportation Research Board*, No. 2229, Transportation Research Board of the National Academies, Washington, D.C., 2011, pp. 28-33.

Lam, W.H.K., H. Shao, and A. Sumalee. Modeling Impacts of Adverse Weather Conditions on a Road Network with Uncertainties in Demand and Supply. *Transportation Research Part B: Methodological*, Vol. 42, No. 10, 2008, pp. 890–910.

Lebacque, the Godunov Scheme and What it Means for First Order Traffic Flow Models. In J. B. Lesort, editor, 13th ISTTT Symposium, Elsevier, New York, 1996, pp. 647–678.

Lighthill, M. J., and G. B. Whitham. On Kinematic Waves II. *Physical and Engineering Sciences*, Vol. 229, No. 1178, 1955, pp. 317-345.

Li, Z., C. Chen and K. Wang. Cloud Computing for Agent-Based Urban Transportation Systems. *Intelligent Systems, IEEE*, Vol. 26, No.1, 2011, pp.73-79.

Lo, H. K. and Y. K. Tung. Network with Degradable Links: Capacity Analysis and Design. *Transportation Research Part B: Methodological*, Vol. 37, No. 4, 2003, pp. 345–363.

MapReduce on Wikipedia. en.wikipedia.org/wiki/MapReduce. Accessed Jul. 31, 2011.

Merchant, D.K., and G.L.Nemhauser, G.L. A Model and an Algorithm for the Dynamic Traffic Assignment Problem. *Transportation Science*, Vol. 12, No. 3, 1978, pp. 183-199.

Merchant, D.K., and G.L.Nemhauser, G.L. Optimality Conditions for a Dynamic Traffic Assignment Model. *Transportation Science*, Vol. 12, No. 3, 1978, pp. 200-207.
Mihaylova, L., B. Rene, and A. Hegyi. Freeway Traffic Estimation Within Particle Filtering Framework. *Systems Research*, 2004.

Muñoz, L., X. Sun, R. Horowitz, and L. Alvarez. Traffic Density Estimation with the Cell Transmission Model, In *2003 American Control Conference Proceedings*, Denver, CO, 2003, pp. 3750-3755.

Nanthawichit, C., T. Nakatsuji, and H. Suzuki. Application of Probe-Vehicle Data for Real-Time Traffic-State Estimation and Short-Term Travel-Time Prediction on a Freeway. In *Transportation Research Record: Journal of the Transportation Research Board*, No. 1855, Transportation Research Board of the National Academies, Washington, D.C., 2003, pp. 49-59.

Newell, G. F. A Simpified Theory of Kinematic Waves in Highway Traffic: (i) General Theory; (ii) Queuing at Freeway Bottlenecks; (iii) Multi-dimensional Flows. *Transportation Research Part B: Methodological*, Vol. 27, 1993, pp. 281-313.

Newell, G. A Simplified Car-following Theory: a Lower Order Model. *Transportation Research Part B: Methodological*, Vol. 36, No. 3, 2002, pp. 195-205.

NIST Definition of Cloud Computing (Draft). csrc.nist.gov/publications/drafts/800-145/Draft-SP-800-145_cloud-definition.pdf. Accessed Jul. 15, 2010.

Ng, M. W., and S. T. Waller. A Computationally Efficient Methodology to Characterize Travel Time Reliability Using the Fast Fourier Transform. *Transportation Research Part B: Methodological*, Vol. 44, No. 10, 2010, pp. 1202-1219.

Noland, R.B. and J.W. Polak. Travel Time Variability: A Review of Theoretical and Empirical Issues. *Transportation Reviews*, Vol. 122, No. 1, 2002, pp. 39-54.

Noland, R. B., K. A. Small, P. M. Koskenoja, and X. Chu. Simulating Travel Reliability. *Regional Science and Urban Economics*, Vol. 28, No. 5, 1998, pp. 535-564.

Oh, J. S., and Y. Chung. (2006). Calculation of Freeway Travel Time Variability. In *Transportation Research Record: Journal of the Transportation Research Board*, No. 1945, Transportation Research Board of the National Academies, Washington, D.C., 2006, pp. 12-23.

Oracle Load Testing for Web Applications Tutorial. Oracle Application Testing Suite Getting Started Guide, Chapter 5, Oracle Corp., Redwood Shores, CA, 2008.

Papageorgiou, M., J. M. Blosseville, and H. Haj-Salem. Modeling and Real-time Control of Traffic Flow on the Southern Part of Boulevard Peripherique in Paris—Part I: Modeling. *Transportation Research A*, Vol. 24, 1990, pp. 345–359.

Real-time Traveler Information Systems, NCHRP Synthesis 399. *National Cooperative Highway Research Program*. 2009.

Richardson, A. J. Travel Time Variability on an Urban Freeway. TUTI Report 25-2003. Presented at the 25th Conference of Australian Institutes of Transport Research (CAITR), University of Adelaide, Dec. 2003.

Richards, P. I. Shock Waves on the Highway. *Operations Research*, Vol. 4, No. 1, 1956, pp. 42-51.

Ruud, P. A. Approximation and Simulation of the Multinomial Probit Model: An Analysis of Covariance Matrix Estimation. Department of California, Berkeley, 1996.

Son, B. A Study OF G.F. Newell's "Simplified Theory of Kinematic Waves in Highway Traffic." Ph.D. Thesis, Graduate Department of Civil Engineering, University of Toronto. 1996.

Sun, X., L. Muñoz, and R. Horowitz. Highway Traffic State Estimation Using Improved Mixture Kalman Filters for Effective Ramp Metering Control. *Mechanical Engineering*, Vol. 1, No. 1, 2003.

Sun, X., L. Muñoz, and R. Horowitz. Mixture Kalman Filter Based Highway Congestion Mode and Vehicle Density Estimator and its Application. *System*, 2004, pp. 2098-2103.

Sundaram, S., N. H. Koutsopoulos, M. Ben-Akiva, C. Antoniou, and R. Balakrishna. Simulation-based Dynamic Traffic Assignment for Short-term Planning Applications. *Simulation Modelling Practice and Theory*, 19, 2011, pp. 450–462.

Travel Time Reliability. Federal Highway Administration. ops.fhwa.dot.gov/publications/tt_reliability/brochure/ttr_brochure.pdf. Accessed March 5, 2012.

Highway Capacity Manual. Transportation Research Board, National Research Council, Washington, D.C., 2000.

Understanding the Contributions of Operations, Technology, and Design to Meeting Highway Capacity Needs. SHRP 2 Project C05 Draft Final Report. Transportation Research Board, National Research Council, Washington, D.C., 2010.

Treiber, M., and D. Helbing. Reconstructing the Spatio-Temporal Traffic Dynamics from Stationary Detector Data. *Cooperative Transportation Dynamics*, Vol. 1, 2002, pp. 3.1-3.24.

Unnikrishnan, A., S. Ukkusuri, and S. T. Waller. Sampling Methods for Evaluating the Traffic Equilibrium Problem under Uncertain Demand. Transportation Research Board 84th Annual Meeting Compendium of Papers, Washington, D.C., 2005.

Waller, S.T. and A. K. Ziliaskopoulos. Stochastic Dynamic Network Design Problem. In *Transportation Research Record: Journal of the Transportation Research Board*, No. 1771, Transportation Research Board of the National Academies, Washington, D.C., 2001, pp. 106–113.

Wang, Y., and M. Papageorgiou. Real-time Freeway Traffic State Estimation Based on Extended Kalman Filter: a General Approach. *Transportation Research Part B: Methodological*, Vol. 39, No. 2, 2005, pp. 141-167.

Wang, M. A New Way of Transportation Modeling—"Cloud Computing" Based Internet Solution for Transportation Modeling. Proceedings of the 10th International Conference of Chinese Transportation Professionals, Beijing, China, 2010.

Work, D. B. A Traffic Model for Velocity Data Assimilation. Applied Mathematics Research Express, Vol. 2010, No. 1, 2010, pp. 1-35.

Xing T., and X. Zhou. Finding the Most Reliable Path with and without Link Travel Time Correlation: A Lagrangian Substitution Based Approach. *Transportation Research Part B: Methodological*, Vol. 45, No. 10, 2011, pp. 1660-1679.

Zhang, H., and T. Kim. A Car-following Theory for Multiphase Vehicular Traffic Flow. *Transportation Research Part B: Methodological*, Vol. 39, No. 5, 2005, pp. 385-399.

Zhou, X., N. M. Rouphail, and M. Li. Analytical Models for Quantifying Travel Time Variability Based on Stochastic Capacity and Demand Distributions. Presented at 90th Annual Meeting of the Transportation Research Board, Washington, D.C., 2011.

Zhou, X., and G. F. List. An Information-Theoretic Sensor Location Model for Traffic Origin- Destination Demand Estimation Applications. *Transportation Science*, Vol. 44, No. 2, 2010, pp. 254-273.