Technical Report Documentation Page

1. Report No. FHWA/TX-08/0-5141-1	2. Government Accession No.	3. Recipient's Catalog No.
4. Title and Subtitle Short Term Travel Time Prediction on Freeways in Conjunction with Detector Coverage Analysis		5. Report Date October 2007 6. Performing Organization Code
7. Author(s) S. Travis Waller, Ph.D, Yi-Chang Chiu, Ph.D, Natalia Ruiz-Juri, Avinash Unnikrishnan, Brenda Bustillos		8. Performing Organization Report No. 0-5141-1
9. Performing Organization Name and Address Center for Transportation Research The University of Texas at Austin 3208 Red River, Suite 200 Austin, TX 78705-2650		10. Work Unit No. (TRAIS) 11. Contract or Grant No. 0-5141
12. Sponsoring Agency Name and Address Texas Department of Transportation Research and Technology Implementation Office P.O. Box 5080 Austin, TX 78763-5080		13. Type of Report and Period Covered Technical Report August 2004–August 2007 14. Sponsoring Agency Code
15. Supplementary Notes Project performed in cooperation with the Texas Department of Transportation and the Federal Highway Administration.		

16. Abstract

As the technological complexities of and public demands upon our Intelligent Transportation Systems (ITS) infrastructure increase, new opportunities and requirements arise regarding how best to manage existing ITS assets and select future deployments. This research project aims to support such decision making by developing~ methods that clearly relate sensor coverage (and other ITS data sources) to Dynamic Message Sign (DMS) performance via algorithms that predict freeway traffic time.

The report documents the research conducted as a part of this project. A detailed literature review of the state-of-the-art and the state of the practice in travel time prediction has been conducted and some of the limitations of the existing models have been identified. Two innovative travel time prediction models have been proposed and their performance has been tested with both simulation and real data. Both of these models use as input traffic counts obtainable from single loop detectors, which are the most widely deployed traffic sensor. The first model is based on an integrated statistical simulation framework and the second model is based on comparing cumulative counts. Methodologies have been developed to arrive at optimal detector location and spacing for better travel time prediction. Some of the issues involved in real-time deployment have been studied and summarized.

17. Key Words Short-term travel time prediction, Detector coverage and spacing, travel time forecasting	18. Distribution Statement No restrictions. This document is available to the public through the National Technical Information Service, Springfield, Virginia 22161; www.ntis.gov.		
19. Security Classif. (of report) Unclassified	20. Security Classif. (of this page) Unclassified	21. No. of pages 140	22. Price

Form DOT F 1700.7 (8-72) Reproduction of completed page authorized

Short Term Travel Time Prediction on Freeways in Conjunction with Detector Coverage Analysis

S. Travis Waller, Ph.D
Yi-Chang Chiu, Ph.D
Natalia Ruiz-Juri
Avinash Unnikrishnan
Brenda Bustillos

CTR Technical Report:	0-5141-1
Report Date:	October 2007
Project:	0-5141
Project Title:	Short Term Travel Time Prediction on Freeways in Conjunction with Detector Coverage
Sponsoring Agency:	Texas Department of Transportation
Performing Agency:	Center for Transportation Research at The University of Texas at Austin

Project performed in cooperation with the Texas Department of Transportation and the Federal Highway Administration.

Center for Transportation Research
The University of Texas at Austin
3208 Red River
Austin, TX 78705

www.utexas.edu/research/ctr

Disclaimers

Author's Disclaimer: The contents of this report reflect the views of the authors, who are responsible for the facts and the accuracy of the data presented herein. The contents do not necessarily reflect the official view or policies of the Federal Highway Administration or the Texas Department of Transportation (TxDOT). This report does not constitute a standard, specification, or regulation.

Patent Disclaimer: There was no invention or discovery conceived or first actually reduced to practice in the course of or under this contract, including any art, method, process, machine manufacture, design or composition of matter, or any new useful improvement thereof, or any variety of plant, which is or may be patentable under the patent laws of the United States of America or any foreign country.

Engineering Disclaimer

NOT INTENDED FOR CONSTRUCTION, BIDDING, OR PERMIT PURPOSES.

Project Engineer: Victor De La Garza
Professional Engineer License State and Number: Texas No. 93717
P. E. Designation: Research Supervisor

Acknowledgments

The successful progress of this project could not have happened without the help and input of many personnel of TXDOT. The authors acknowledge Mr. Victor De La Garza, the project PD and Carlos Chavez, the project PC for facilitating the research and collaboration with TXDOT districts. Special thanks to the PAs, Mr. Charles Koonce and Edgar Fino, who were actively involved in the research activities and provided the research team with valuable guidance and inputs.

Products

This report includes product P1, “Travel Time Prediction Guidelines,” which is attached as an appendix.

Table of Contents

List of Figures

List of Tables

Chapter 1. Introduction

As the technological complexities of communication systems and public demands upon our ITS infrastructure increase, new opportunities and requirements arise regarding how best to manage existing Intelligent Transportation Systems (ITS) assets and select future deployments. This research project aims to support such decision making by developing methods that clearly relate sensor coverage (and other ITS data sources) to Dynamic Message Sign (DMS) performance via algorithms that predict freeway traffic time.

A fully functioning dynamic message sign (DMS) system provides full public value only when predicted travel time information is integrated into it. Currently, the San Antonio District is routinely displaying such information on their DMS systems, which has resulted in positive public reaction. The state-of-the-practice allows engineers to utilize existing high coverage detectors to design algorithms for calculating predicted travel times for DMS display. The basic idea of the algorithm is to divide distance between the DMS and prediction destination into segments; then determine travel time between two detectors by dividing the segment distance by detector measured speed and sum up travel times along the segments using conservative figures generated by different detector data. These algorithms are activated during off-peak hours and are often turned off in times of congestion and/or incidents because of unreliability of the prediction quality.

Given the success of the San Antonio district, one may think that the natural extension of this work is to apply the above methodology to other districts. One potential difficulty of such direct transfer of system capability is that different districts have differing levels of congestion, traffic flow compositions, and prediction accuracy requirements. Further, the requirements of detector coverage and algorithm structure are location specific. The optimal configuration in one district may not work well in others.

The travel time calculated, based on current freeway conditions as done in San Antonio, is essentially an instantaneous travel time measured at the particular time of calculation, which could significantly deviate from the travel time that drivers actually experience, particularly during onset or dissipation of congestion. The obvious reason for this potential deviation is fast changing traffic dynamics. To take a practical application; if a freeway segment takes 15 minutes to travel, freeway inflow/outflow could significantly vary during the period with congested and unstable traffic flows, not to mention the occurrence of incidents. The core concept of providing reliable freeway travel time prediction during unstable traffic conditions is to have a short-term traffic prediction capability, or alternatively, to have a certain level of artificial intelligence that discerns predicted travel time given a traffic condition.

Another issue with travel time prediction is the type of sensors used and the data necessary. Single and dual loop detectors are the most widely used traffic detectors measuring primarily counts, speed, and occupancy data. Even though the accuracy of prediction systems will improve if AVI data is available, travel time prediction models based on AVI data may not be widely used because AVI technologies have not been widely deployed. Therefore, there is a need to develop prediction models that use either counts, speed, or occupancy as the inputs.

Furthermore, existing practice employs a half-mile detector spacing scheme. The relationship between the accuracy of travel time prediction and spacing of detectors is unknown. It would seem intuitive to think that the quality of travel time prediction degrades with sparser detector spacing. Moreover, how large a spacing is acceptable under what accuracy

requirements? What degree of accuracy can one expect given a level of detector coverage? These unanswered questions are of great importance to any district considering developing the capability for specifying prediction accuracy requirements and budgeting capital investment for detectors.

In the current literature, there are a limited number of simulated or empirical studies addressing the issues discussed above, particularly the relationship between detector spacing and prediction accuracy using different detection data. Investigating such issues could be of great benefit to all Texas Department of Transportation (TxDOT) districts with DMS systems or where web-based systems are being considered for the dissemination of point-to-point travel time information. In this regard, this project considers the following research objectives:

1. Investigate the performance and features of existing algorithms deployed in the U.S. and Europe
2. Conduct a detailed review of existing travel time prediction models available in the literature and identify the advantages and disadvantages of each type of models
3. Based on the review conducted, identify a couple of potential prediction models for evaluation
4. Develop, calibrate, and demonstrate potential prediction models that complement and improve existing capabilities
5. Understand the performance of existing and proposed travel time prediction algorithms with respect to differing detector spacing specifications and various types of detection data
6. Determine detection data coverage requirements given a prediction accuracy requirement
7. Identify potential issues that may arise in real-time deployment and traffic management center (TMC) integration of the developed prediction models and provide recommendations to overcome the same

The next section illustrates the difference between different types of travel times available and emphasizes the difficulty involved in accurate travel time prediction.

1.1 Types of Freeway Travel Time

Travel time on freeway sections can be generally classified into:

- Instantaneous Travel Time (ITT),
- Reconstructed Travel Time (RTT), and
- Forecasted Travel Time (FTT).

Instantaneous Travel Time stands for the travel time of a vehicle traversing a freeway segment at time t if all traffic conditions remain constant until the vehicle exits the freeway segment. ITT generally underestimates travel time at the onset of congestion and overestimates at the dissipation of congestion. In other words, ITT is reliable during the off-peak hours in

which traffic conditions remain stable. San Antonio TransGuide's existing algorithm produces Instantaneous Travel Time, and it has been shown to produce reliable travel time estimation only during off-peak hours (Quiroga, 2000.)

Reconstructed Travel Time (RTT) means the travel time realized at time t when a vehicle leaves a freeway segment. An Automatic Vehicle Identification (AVI) travel time measure is a typical type of RTT because a vehicle's actual travel time is not measured until the vehicle passes a toll tag beacon or a Road Side Terminal (RST). While this type of travel time measurement can give precise actual travel times, it has been shown not to be a good measure for online travel time prediction because it introduces non-trivial time-lag in actual travel time detection (Chen and Chien, 2001.) The AVI data is however, very suitable for calibrating online algorithms in an offline manner because the precisely measured travel times allow the validation of model prediction accurately. The AVI data collected by Houston and TRANSCOM produce RTT.

Forecasting Travel Time (FTT) is defined as the travel time that is actually experienced by drivers who will traverse the freeway segment. This is the most useful information from a driver's perspective, yet the most difficult to produce precisely, particularly during peak-hours in which traffic conditions are less stable. The FTT is certainly the focus of this research because it is intended to be disseminated to the traveling public via Dynamic Message Signs (DMS) or the Internet. The traveling public demands FTT instead of ITT or RTT.

Estimating FTT requires predicting the short-term traffic conditions in the intended freeway segment for the next 10-20 minutes at each prediction instance. Historically, real-time combined with historical loop detector data are used by regression, time-series, artificial neural networks (ANN), etc. models. The following section gives a short review of past studies on freeway travel time studies.

1.2 Existing Freeway Travel Time Prediction Systems

Important features of existing travel time prediction systems such as algorithms used, data requirements, and prediction accuracy under different traffic flow conditions are summarized in this section, with special focus on the TransGuide System in San Antonio and the DACCORD Project in Netherlands. Outside of the travel time prediction models, some of the existing traffic detection capabilities in the U.S. and Europe are also identified.

1.2.1 San Antonio TransGuide, U.S.

The TransGuide traffic management center in San Antonio monitors traffic operations on a network of freeways and major arterial streets covering most of the metropolitan area (Figure 1.1). One of the main components of the monitoring system is a series of sensors (mainly loop detector pairs and sonic detectors) that provide the capability to measure point speeds. These detectors are roughly 0.5 miles apart. Based on these point speeds, the system estimates travel times to specific landmarks and displays the estimated travel times on dynamic message signs (DMSs) that are roughly 2 to 3 miles apart. The sign messages are usually made available on the TransGuide website (Figure 1.2).

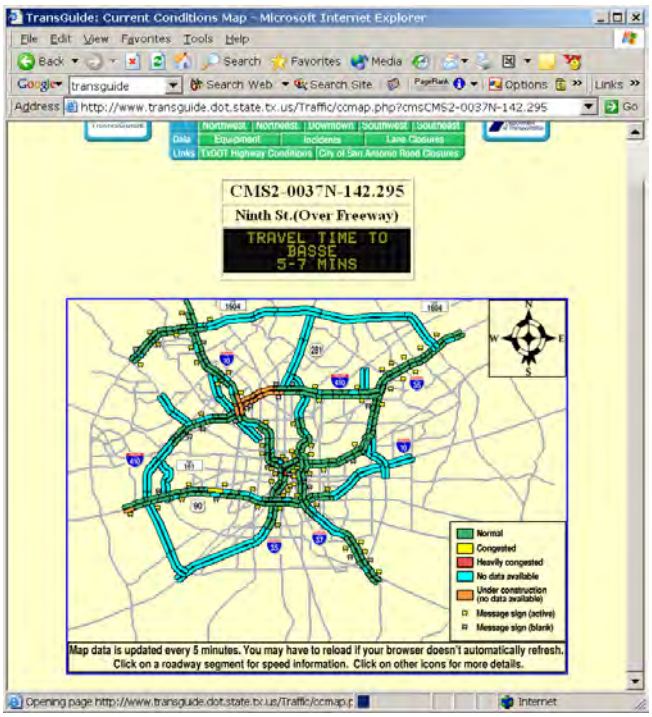

Figure 1.1: TransGuide Traveler Information system website

For travel time estimation, TransGuide uses very simple rules to predict the travel time between a specific DMS to an intended interchange or landmark. The traffic detection system in the San Antonio network consists of both inductive loop detector pairs and radar detectors that are roughly 0.5 miles apart. The key feature of the calculation relies on the rules that determine the detector stations that are located between the DMS location and the major interchange. For each pair of adjacent detector stations, the system determines the lowest of the two detector station speeds and assigns that speed to the road segment that connects the adjacent detector stations. For example, if the speeds associated with two adjacent detector stations are 52 and 58 mph, the system assigns 52 mph to the road segment that connects the two adjacent detector stations. The main effect of using this approach for estimating road segment speeds is that the system tends to underestimate speeds which, in turn, should result in slightly overestimated travel times under light traffic conditions.

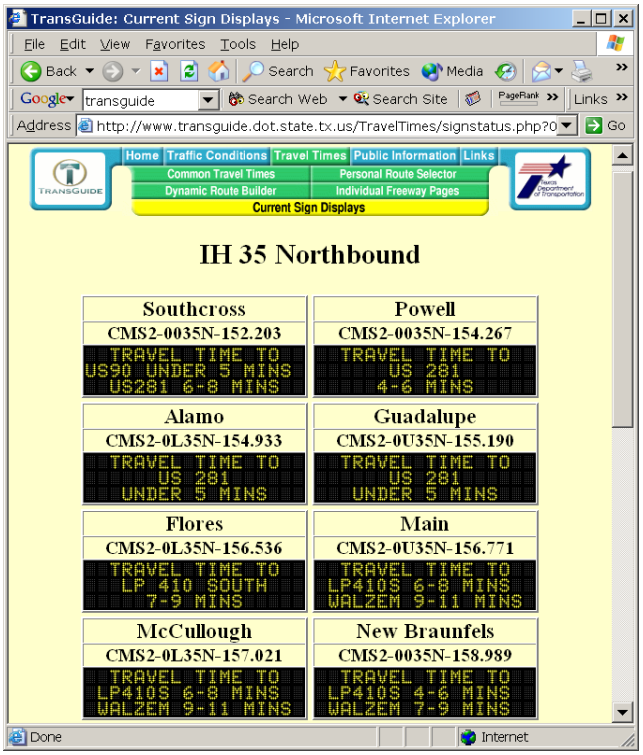

Figure 1.2: TransGuide DMS messages as shown on TransGuide website

The system converts each partial road segment speed into an equivalent travel time and adds all of the partial segment travel times to produce an estimate of the total travel time between the DMS location and the major interchange. Doing so creates systematic overestimation during off-peak and underestimation during peak hours, because traffic congestion builds up in a non-linear manner (Morin and Fevre, 1997). The study on the TransGuide DMS travel time estimation accuracy concludes that the currently employed schemes generate reasonable travel time prediction only in off-peak hours (still systematically overestimate actual travel time), in which travel time prediction capabilities do not seem very beneficial to drivers because drivers can easily estimate the travel time based on their own cruising speeds. As shown in Figure 1.3, the travel time accuracy significantly degrades during AM peak hours, in which accurate travel time prediction is needed far more than in off-peak hours.

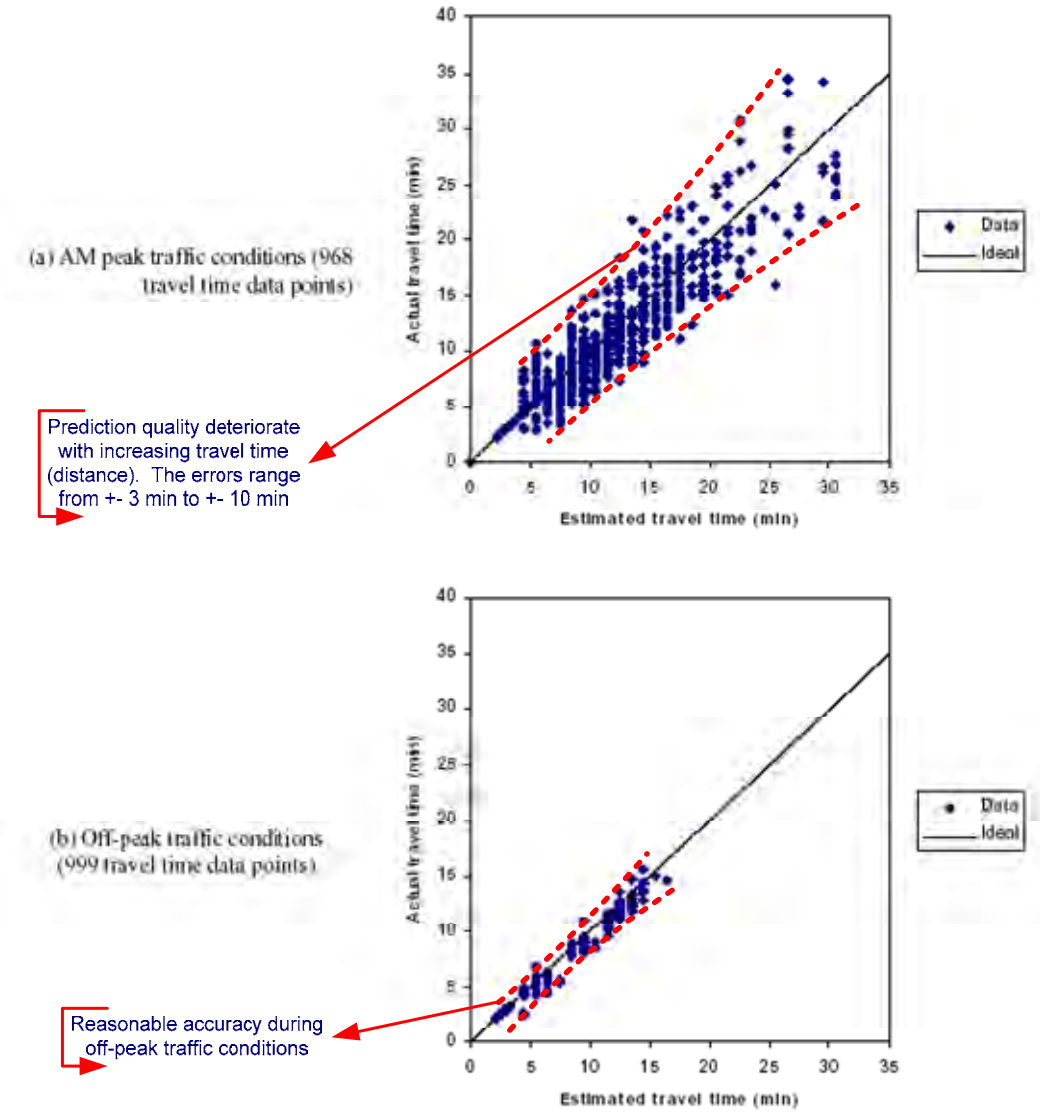

Figure 1.3: TransGuide travel time prediction accuracy during AM peak and off-peak hours (Quiroga, 2000)

Figure 1.4 clearly indicates that if a driver traverses a DMS site on IH-10 that is far (more than 10 miles away) from the intended interchange (i.e., IH-35) during the onset of congestion (7:10 or 7:30 AM onward), he/she will see the DMS travel time underestimated by 5-7 minutes (more than 30 percent) compared to the actual experienced travel time. If the traversed sign is less than 6 miles from the intended interchange, then the predicted travel time is overestimated by 1-3 minutes.

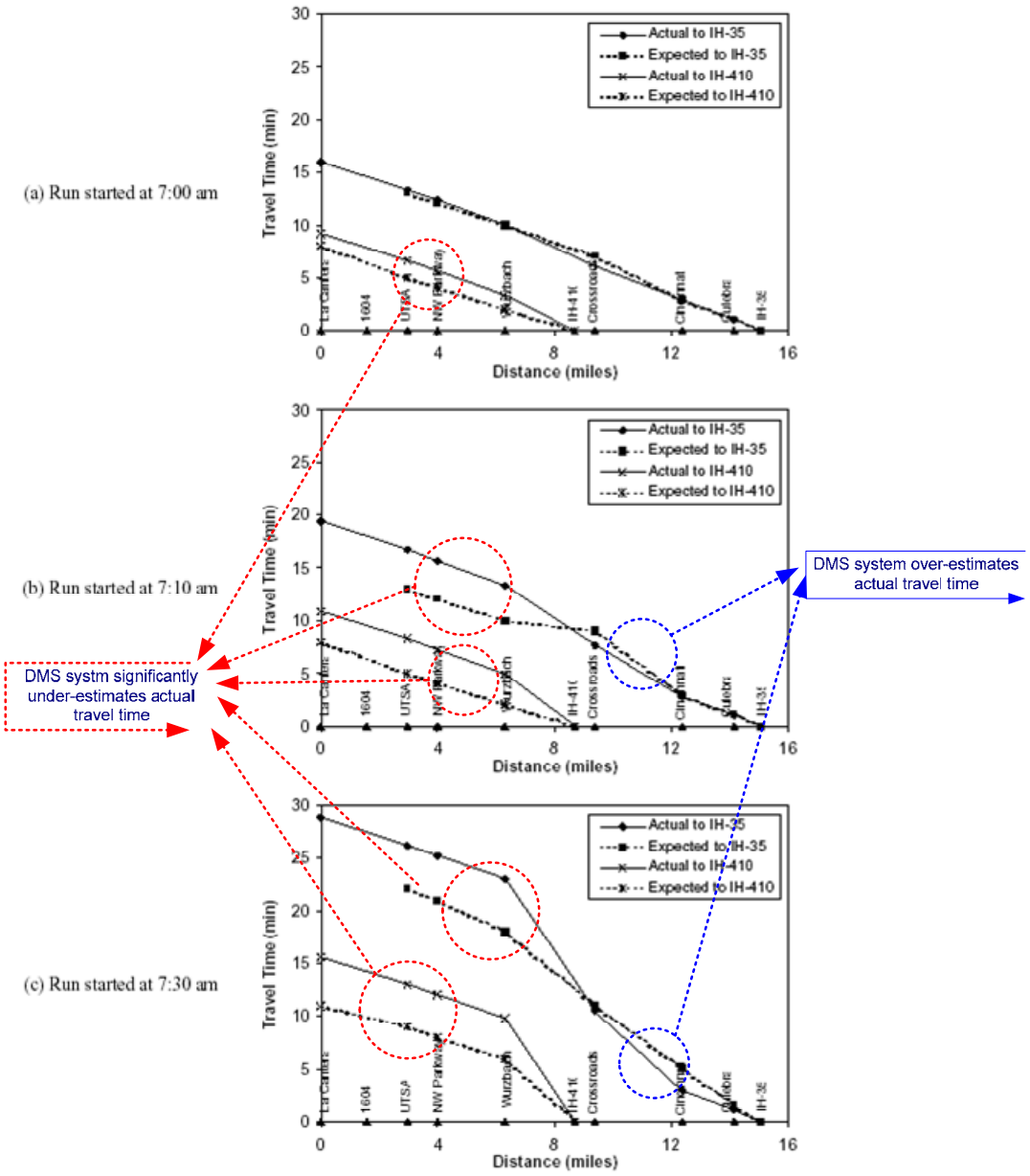

Figure 1.4: Travel time from specific DMS location to IH-410 and IH-35 on IH-10 (travel time runs made during the AM peak on May 5, 2000) (Quiroga, 2000)

1.2.2 Amsterdam (DACCORD, Netherlands)

The Development and Application of Coordinated Control of Corridors (DACCORD) project (1996–) aims at developing and testing coordinated control measures for motorway networks. It is part of the European Union Telematics Application Programme, sector Transport. CWI contributes with theoretical research into integrated control and routing control of motorway networks. The main objective of DACCORD is to create a practical Dynamic Traffic Management System (DTMS) for integrated and coordinated control of inter-urban motorway corridors. It addresses the needs of road authorities and traffic operators responsible for (parts of)

the European motorway network, as well as the drivers' needs. The project in particular aims to increase effective road capacity and reduce travel times and number of accidents.

An essential element is the integrated operation of control at a network-wide (i.e., not local) level, using the following control measures:

- Dynamic information on queue lengths, travel times, and routing directions through variable message signs
- Ramp metering systems for motorway access control
- Motorway-to-motorway control systems (ramp metering at motorway merges)
- Variable speed limits and lane closure signals
- Pre-trip and en-route information by broadcasting and radio data systems.

DACCORD mainly builds on the CHRISTIANE, EUROCOR, DYNA, and GERDIEN projects, which were parts of the EU program DRIVE. The research and development (R&D) work focuses on the optimization of the simultaneous use of multiple control instruments in order to maximize their combined effect and to avoid unwanted side effects. Three major test sites are used for practical validation in real life situations (the particular aspects are explored in brackets):

- The Amsterdam network, part of the EuroDelta initiative (traffic state monitoring and estimation, short time prediction models for traffic flows, speeds, queue and travel time, display of estimated dynamic travel times using VMS [shown in Figure 1.5], coordination and integration of ramp metering control, and VMS control)
- The Paris network, including the ringroad and the connecting motorways (motorway-to-motorway control including real-time ramp metering techniques, estimated travel time display on the Corridor Peripherique of Paris and the SIER motorway, and real data screening)
- Padua-Venice motorway, part of the Pitagora initiative (on-line traffic model system, including the OD estimation and prediction)

In general, DACCORD is a forward-looking advanced freeway management initiative and has demonstrated satisfactory traffic management capabilities. However, like other systems, it also faces technical challenges in providing accurate freeway travel time prediction.

Figure 1.5: Freeway travel time prediction via DMS in the DACCORD system in Amsterdam

1.2.3 Other Traffic Detection Capabilities in the U.S.

Outside of TransGuide there are three other major traffic data detection and archiving programs in Texas: TransVista in El Paso, TransStar in Houston and DalTrans in Dallas. The TransVista system uses a combination of point loop detectors and fifty-five cameras to monitor sections of the roadway. Dynamic Message Signs and Lane control signs are used to disseminate information about freeway conditions to the travelers. The freeway sections visible through the cameras can be accessed on the web.

The TransStar system in Houston uses Automatic Vehicle Identification (AVI) to monitor freeway traffic and disseminate information through popular media outlets such as radio, television outlets, and dynamic message signs. Primary information transmitted are travel time estimates and speed data. Vehicles with transponders are used as probes. AVI antennas are readers installed along freeways that detect when the probe vehicles pass the point of installation. The time difference between detection of a probe vehicle at successive AVI reader locations is used to arrive at travel time estimates and speed.

The DalTrans/TransVision system in Dallas uses a combination of loop detectors and closed circuit cameras to provide information about incidents, lane closures, and speeds in the Dallas and Fort Worth area. The information is provided to the public using dynamic message signs or it can be accessed on the web.

The Florida DOT collects real-time data on a 40-mile I-4 corridor near Orlando using dual loop detectors. A nonlinear time series model developed at the University of Central Florida was used to predict travel times. This forecasted travel time information was transmitted to the public through a website. The Wisconsin DOT provides an estimate of current travel times on the I-94, I 894, and I 43 corridors near Milwaukee using traffic sensors and freeway cameras.

According to the Federal Highway Administration (FHWA), there are around 300 traffic information sites in the United States. Some of the popular ones include NAVIGATOR in Atlanta, Statewide Traveler Information by WA DOT, TRI MARC maintained by Indiana DOT, and GCM travel along the Gary-Chicago Milwaukee corridor. However, many of these traffic information sites do not archive the travel time data and they are not readily accessible. One of the best data archiving systems in the U.S. is the PEMS Freeway Performance Measurement System and Berkeley Highway Laboratory in California.

In California a study is underway to start using remote sensor nodes to monitor traffic. Several magnetic sensors are placed in or above the road. These sensors detect vehicles and transmit the data back to an access point. The system is controlled by an energy saving protocol called PEDAMACS. This protocol controls the data transfer between the access point and the sensors via radio signals in order to minimize the time the sensors are functioning. When the sensors are not needed they go into sleep mode to save energy. The access point, which can be placed adjacent to the road, can be access by the TMCs to remotely obtain data and control the nodes.

Even though most point detectors can provide adequate information, they have several drawbacks as opposed to Wireless location technology (see Figure 1.6). Several papers and tests have been conducted in the U.S. in cooperation with cell phone providers to test wireless location technology. Wireless location technology uses the signal from cell phones to track the vehicles they are in. This way any number of vehicles can be tracked on any desired road so long as the vehicle contains a wireless device, such as a cell phone or GPS device. The advantage to using wireless location technology is its ability to obtain data from an existing infrastructure without the need to worry about power supply or environmental effects. A cell phone can be tracked even if it is not being used, as long as it is turned on. The location of the cell phone can be obtained using the signals sent to and from the tower; or if the phone has built-in GPS it can give the exact location of the phone.

Some companies have been commissioned to use wireless location technology by some states to provide data. One company, AirSage, conducted several tests in the past few years in Virginia, Utah, and California; however, they could not provide adequate information. Other companies such as Cellint, Delcan, Globis, and IntelliOne are still testing their wireless location technology; however the data is also not accurate enough. The difficulty with wireless location technology is accurately mapping the received data depending on how the location of the vehicle is obtained. It is crucial to accurately position the vehicle in order to determine the speed and the distance travel by a vehicle over a certain link.

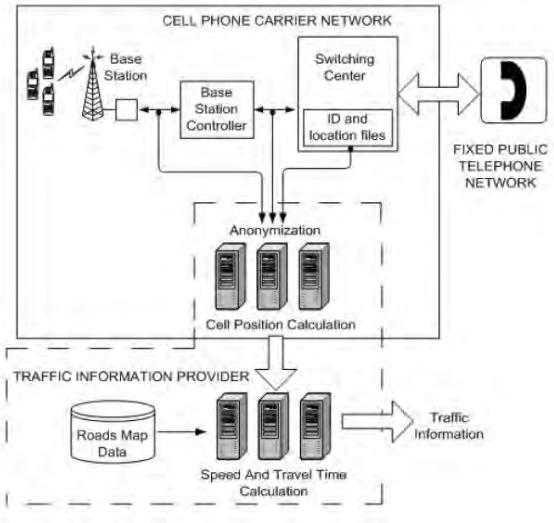

Figure 1.6: Schematic of wireless location technology

This section summarized the travel time prediction and data detection capabilities in the U.S. The next section provides a brief overview of travel time forecasting system in Europe.

1.2.4 Travel Time Prediction in Europe

Catalan Traffic Service, the department of transportation for the city of Barcelona and surrounding areas, collects real-time traffic data such as speed, flow, occupancy, and vehicle classification every minute using inductive loops and 127 closed circuit television cameras. Travel time forecasting is conducted based on a combination of historic information and real-time travel data. Travel time information and alternate route information to specific destinations is disseminated through sixty-seven Dynamic Message Signs. Level of service of the roadway sections is calculated and information is provided on the website using color codes (Figure 1.7).

Figure 1.7: Catalan Traffic Service DMS
(Source: http://international.fhwa.dot.gov/travelinfo/barcelona.htm)

In Berlin, the VMZ Berlin project headed by Siemens AG and Daimler Chrysler has led an effort to provide travel time and incident information on a website. The travel time information is calculated based on historical travel times and updated based on real-time reports obtained from the police. Efforts are underway to develop a travel time prediction model using detectors.

In Scotland, traffic detectors coupled with closed circuit television cameras are used to estimate travel time and congestion on freeways. Travel time estimation and prediction efforts along the lines of the DACCORD project described in Section 2.1.4 are currently underway in Amsterdam, Paris, Venice, and Athens.

In Scotland, research is being conducted to develop traffic detection devices that are placed into road studs. The road studs are built to withstand a lot of wear and tear as they are meant to delineate the lanes on a road. Traffic is monitored by two light-activated optical sensors placed in the road stud. The sensors will be equipped with temperature and light sensors to detect

inclement weather and warn against incidents and hazards. In the future, there will also be planes to add video detection to the studs in order to track vehicles using license plates.

1.2.5 Summary of the State-of-the-Practice

The fundamental deficiency of the existing travel time prediction system is that it is not predictive in nature. It does not anticipate the short-term evolution of traffic dynamics downstream. As concluded by Quiroga (2000), such a simple but non-robust travel time prediction scheme cannot provide satisfactory prediction during recurrent peak hours, during which travel time prediction is more important than the off-peak hours.

Similarly, the current practices for travel time predictions will not perform well under non-recurrent traffic conditions such as work zones, accidents, and special events. When motorists are stuck in a long queue without knowing the exact cause of the delay, reasonable travel time prediction on DMS becomes critical to help ease road rage and to assist motorists in evaluating whether to divert to alternative routes.

Based on the research team's personal communication with traffic engineers in the San Antonio district, the DMS travel time prediction messages are usually turned off during peak-hours, work zones, and accidents. Apparently, there is ample room for improving the freeway travel time prediction under different traffic conditions with an improved framework for all the DMS systems operated in Texas.

1.3 Travel Time Prediction—Literature Survey

Past work on travel time prediction can be classified into statistical models and heuristic models. Statistical techniques primarily use regression techniques or time series analysis to estimate travel times based on historical or real-time information. Purely statistical techniques do not perform very well during abnormal traffic conditions, i.e., conditions that may not be present frequently in the data sample used for calibration. Hence, purely statistical techniques may not be accurate during peak hours. Heuristic models use techniques like Artificial Neural Networks and the Kalman filtering approach for short term prediction of traffic flow models. Other types of methods include using dynamic traffic assignment and simulation-based techniques in combination with the above methods to better predict travel times. Some of the recent work on short term travel time prediction models are summarized in Figure 1.8.

1.3.1 Regression Models

Kwon, Coiffman et al. (2000) used linear regression and advanced statistical methods such as tree methods to develop models for predicting travel time. Simple prediction models, like linear regression based on current flow and occupancy etc., were found to be beneficial for short term travel time forecasts. However, the model uses travel time estimated using probe vehicles as the response variable. It might not be possible to obtain current travel time measurements using probe vehicles on a regular basis. Also, the model performance is not tested under irregular traffic conditions, such as incidents that may commonly occur during peak periods. Chakraborty and Kikuchi (2004) used transit bus vehicles as probe vehicles and developed a simple linear equation using regression to predict automobile travel time based on the bus travel time. Zhang et al. (2003) developed a linear model with time varying coefficients for short term travel time prediction. However, the above model may not perform well in the presence of incidents and other special traffic conditions.

1.3.2 Neural Networks-Based Models

Van Lint et al. (2002) used recurrent neural networks to predict freeway travel time. The model was tested on synthetic data and found to perform very well. One of the drawbacks of the above methodology is that it requires that the traffic data contain a sufficient amount of great detail. In another work conducted by the same authors in 2003, the researchers improved the robustness of the model to corrupt data by using simple imputation schemes where the missing data are replaced by reasonable ad hoc approximations. Van Lint (2004) extended the above work to develop an approach to quantify the uncertainty around the travel time predictions. Confidence intervals are developed around the travel time predictions. Uncertainty in travel time predictions can be caused due to uncertainty in data or due to the uncertainty in the model. Huisken et al. (2002) developed a travel time prediction method based on Artificial Neural Networks and compared the same with the current travel time prediction model in use in a corridor in the Netherlands.

Different methodologies for travel time estimation were tested to arrive at more accurate estimates of current travel time. The estimates of current travel time are used for short term prediction of future travel times. Predicted travel times using artificial neural networks are found to be more accurate than the currently used naïve methods. Mark and Sadek (2004) conducted a comprehensive statistical analysis of the impact of various factors, such as temporal resolution of the data, speed, flow, etc. on the experienced travel time predictions obtained using artificial neural networks in the presence of incidents. The data for this study was synthetically generated by simulation using the Cell Transmission Model as the traffic flow model.

INPUT DATA	**MODELING TECHNIQUE**	
	Statistical Models	***Heuristic Models***
Traffic Detectors	▪ Chakraborty et al., 2004 ▪ De Ruiter, Schouten et al., 2002 ▪ Ishak et al., 2003 ▪ Kwon, Coifman et al., 2002 ▪ Rice,Van Zwet, 2001 ▪ Van Arem et al., 1997 ▪ Zhang et al., 2003	▪ Mark et al., 2004 ▪ Ohba, Koyama et al., 1998 ▪ Palacharla and Nelson, 1999 ▪ Van Lint et al., 2002 ▪ Van Lint et al., 2003 ▪ Xie, Cheu et al., 2004
Probe Vehicles	▪ Nanthawichit et al., 2003	▪ Dharia and Adeli, 2003
Automatic Vehicle Identification (AVI)	▪ Chen and Chien, 2001 ▪ Chien and Kuchipudi, 2002 ▪ Chien and Kuchipudi, 2003 ▪ Kuchipudi et al., 2003	

Figure 1.8: Overview of travel time prediction literature

1.3.3 Kalman Filtering

Chien et al. (2002) used a Kalman filtering algorithm for short term prediction of travel time. Predictions using link-based travel times were found to be more accurate than prediction using path-based travel times. The study used a combination of historical and real-time data. Filtering of biased data was found to improve the accuracy of the model.

Chien et al. (2003) integrated concepts of traffic simulation and statistics to develop a model for predicting travel times based on spot speed/volume data obtained from various sensors on the freeway corridor. The spot speed/volume data was used to calibrate a traffic simulation model and the travel times in the corridor were obtained through simulation. The simulated travel times were fed into a Kalman filtering framework to predict the future travel times. Nanthawichit et al. (2002) developed a method for short term travel time prediction by combining a Kalman Filtering approach with a macroscopic traffic flow model. The proposed model was found to perform better for hypothetical traffic flow data. However, this model requires data about traffic flow states from probe vehicles, which may not be feasible. Kuchipudi et al. (2003) developed a model in which both path-based data and link-based data are used to predict travel times using a Kalman filtering framework. Depending on the prediction error obtained using path-based and link-based data, the corresponding travel times are chosen. Fan et al. (2004) developed an online adaptive least squares method for short term prediction of traffic parameters. The methodology proposed was shown to be a special case of the Kalman filtering approach.

1.3.4 Time Series/Multivariate State-Based Models

Stathapoulos and Karlaftis (2002) developed a multi-variate time series model for short term prediction of traffic flow parameters. Multi-variate time series models were found to perform better than univariate models for short term traffic predictions. Kamarianakis and Prastacos (2003) compared the forecasting performance of two univariate and two multivariate models of traffic flow. The main variable under study was the relative velocity, which was defined as the traffic volume divided by the occupancy. Multivariate models are found to be more effective than univariate models in capturing disturbances in traffic flow. An analysis of the performance of the traffic prediction system implemented on the I-4 corridor in Orlando, Florida was conducted by Ishak and Al-Deek (2003). The system was evaluated under a wide range of traffic conditions and prediction errors. The model used in this study is a non-linear time series model and was developed by D'Angelo et al. (2000). The performance of the model was found to deteriorate rapidly with the onset of congestion, with errors up to 25-30 percent.

Lindveld and Thijs (2000) provide an overview of the performance of several travel time estimation and prediction methods at three test sites in Europe. The travel time prediction was conducted using a statistical model, a time series analysis, and a real-time dynamic traffic assignment-based model. The real-time dynamic traffic assignment-based model was found to be more accurate. However, the performance of this model under congested conditions has not been studied.

1.3.5 Summary of the State-of-the-Art

Based on the survey conducted above, some of the benefits and deficiencies of each of these types of approaches are being identified to aid in development of a better model. Some of the salient findings of the study are summarized below.

(i) Most of the prediction methodologies developed in research or implemented in practice (like in The Hague-Rotterdam motorway in Netherlands and I-4 in the Orlando) have been calibrated and tested on one particular data set only. There is a need to calibrate and test the prediction capability of the developed model over multiple data sets to get a better idea of the model performance.

(ii) Prediction capability of most of the approaches is dependent on the data set used. For example, the performance of artificial neural network-based models is found to be dependent on whether future traffic states are present in the training samples. Short term forecasts during peak periods made by statistical techniques using purely historical data sets are found to inaccurate. Short term forecasts made using real-time data or a combination of real-time data and historical information are found to be more accurate in capturing real-time dynamics.

(iii) In some cases, the data set might contain some erroneous data due to detection errors or atypical traffic conditions. Smoothing of such data might result in better predictions during steady state conditions. However, care should be taken while smoothing to avoid losing valuable information about the state of the traffic.

(iv) Most of the information about the states of the system is obtained through spot detectors. The information provided by these detectors is primarily speeds, flows, or occupancy. There is a need to estimate travel time information from speeds or flows. A lot of techniques are available to estimate the current travel time from speed volume data; however, many of them are based on simple linear relationships. There are relatively few studies that use analytical traffic flow relationships or simulation based models to determine the current travel times. Usage of traffic flow theoretic relationships or simulation-based models may be more useful in extrapolating local conditions (like spot speed data) to that of a link (like travel time on a link). Because the accuracy of the predicted travel times is dependent on the accuracy of the estimated current travel times, it is very important to explore such methods to arrive at better estimates of current travel time. Another way to explore better methods of estimating current travel times would be to forecast future traffic states in terms of speed and flow, and use travel time flow relationships to extract future travel time information from future estimates.

Chapter 2. Overview of Selected Models

Based on the literature review conducted, four different models were selected for preliminary testing. The first three models are simple speed-based models. The speed-based models were selected as they are relatively easy to implement and are widely used in the field. The fourth model selected for implementation is the time series-based models applied in the DACCORD project. This innovative model uses both speed and traffic counts to estimate travel times. This model was primarily used for estimating travel times. In this project the efficacy of this model for prediction purposes is tested.

2.1 Speed-Based Estimation Models

Speed-based estimation models are one of the most common methods of estimating travel times in the system. They provide a relatively easy way of arriving at current travel times which are one of the primary inputs for most travel time forecasting models. Speed-based models estimate travel times by dividing the length of a section by the average travel time in the section. The data necessary for implementing these models can be collected using a variety of methods, such as probe vehicles with transponders, Automatic Vehicle Identification (AVI) readers, and simple dual loop detectors. Loop detectors are, however, the most commonly used data source for speed-based estimation models. In this section an overview of three speed-based estimation models will be provided: (i) Instantaneous Model (ii) Dynamic Instantaneous Model, and (iii) Dynamic Time Slice Model.

2.1.1 Instantaneous Model

In the instantaneous model, the corridor D1-D4 (see Figure 2.1) is split into reasonably sized sections S1, S2, and S3. Travel time in the corridor at time interval i is calculated by calculating the travel time in the individual sections at time interval *i* and summing them up. The travel time in the individual sections is calculated using the formula explained below.

Let $v(i,1)$, $v(i,2)$, $v(i,3)$ and $v(i,4)$ represent the speed on the freeway at time interval i. Let $L(s1)$, $L(s2)$ and $L(s3)$ denote the length of the section *S1, S2, and S3*, respectively. Now the travel time in the section *S1* at time *i* $TT_{S1}(i)$ is calculated using the following formula:

$$TT_{S1}(i) = \frac{L(s1) * 2}{v(i,1) + v(i,2)}$$

In a similar manner, the travel time in the section *S2* at time *i* $TT_{S2}(i)$ is calculated using the following formula:

$$TT_{S2}(i) = \frac{L(s2) * 2}{v(i,2) + v(i,3)}$$

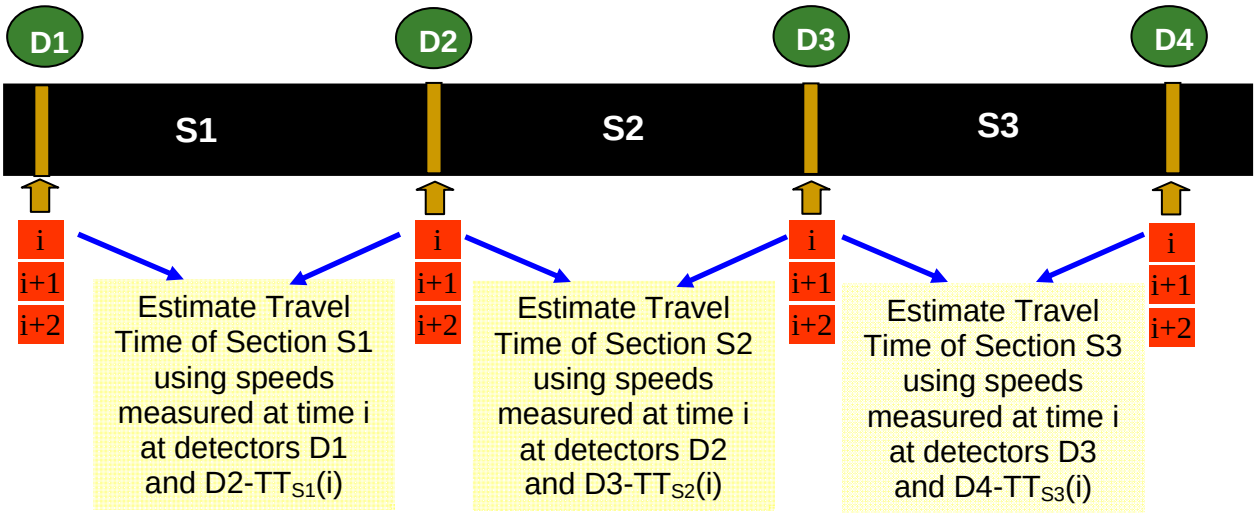

Figure 2.1: Instantaneous Model

The travel time in the corridor at time i $TT_{D1-D4}(i)$ is calculated as:

$$TT_{D1-D4}(i) = TT_{s1}(i) + TT_{s2}(i) + TT_{s3}(i)$$

2.1.2 Dynamic Instantaneous Model

One of the defects of the instantaneous model is that it does not account for spatial and temporal congestion along the freeway section. A person departing from *D1* at time *i* is assumed to experience the same travel time in section *S2* as a person departing from *D2* in time *i*. To address this defect, the dynamic instantaneous model (see Figure 2.2) attempts to calculate the travel time in the corridor at time *i* , $TT_{D1-D4}(i)$ by cascading the travel times. The travel time in the section *S1* at time *i* $TT_{S1}(i)$ is calculated using the following formula:

$$TT_{S1}(i) = \frac{L(s1) * 2}{v(i,1) + v(i,2)}$$

The travel time in the section *S2* at time $k = i + TT_{S1}(i)$ $TT_{S2}(i + TT_{S1}(i))$ is calculated using the following formula:

$$TT_{S2}(k) = \frac{L(s2) * 2}{v(k,2) + v(k,3)}$$

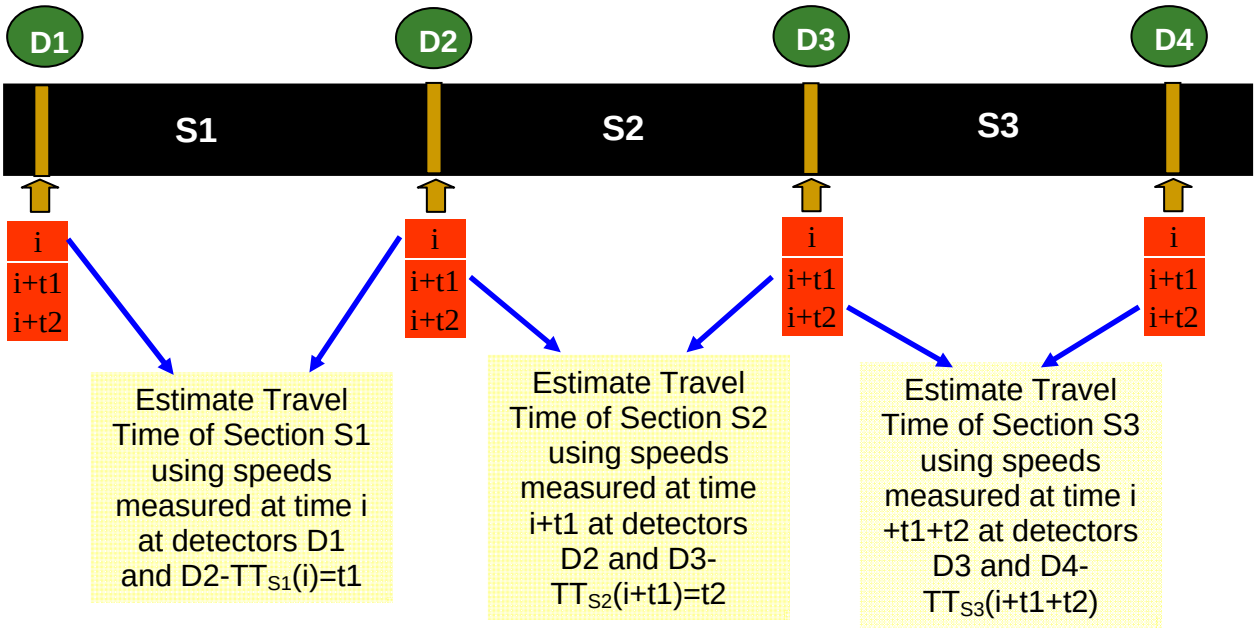

Figure 2.2: Instantaneous Model

The travel time in the section *S3* at time $l = k + TT_{S2}(k)$ $TT_{S3}(l)$ is calculated using the following formula:

$$TT_{S3}(l) = \frac{L(s3) * 2}{v(l,3) + v(l,4)}$$

The travel time in the corridor at time *i* $TT_{D1-D4}(i)$ is calculated as:

$$TT_{D1-D4}(i) = TT_{s1}(i) + TT_{s2}(k) + TT_{s3}(l)$$

2.1.3 Dynamic Time Slice Model

Despite accounting for dynamics across sections, dynamic instantaneous models do not account for the evolution of congestion within the section. To address this defect, a time slice model calculates travel times as given below.

The travel time in the section *S1* at time *i* $TT_{S1}(i)$ is calculated by solving the equation presented below using approximate solution techniques:

$$TT_{S1}(i) = \frac{L(s1) * 2}{v(i,1) + v(i + TT_{S1}(i),2)}$$

In a similar manner (Figure 2.3), the travel time in the section S2 at time $k = i + TT_{S1}(i)$ $TT_{S2}(k)$ and the travel time in the section S3 at time $l = k + TT_{S2}(k)$ $TT_{S3}(l)$ is calculated.

The travel time in the corridor at time *i* $TT_{D1-D4}(i)$ is calculated as:

$$TT_{D1-D4}(i) = TT_{s1}(i) + TT_{s2}(k) + TT_{s3}(l)$$

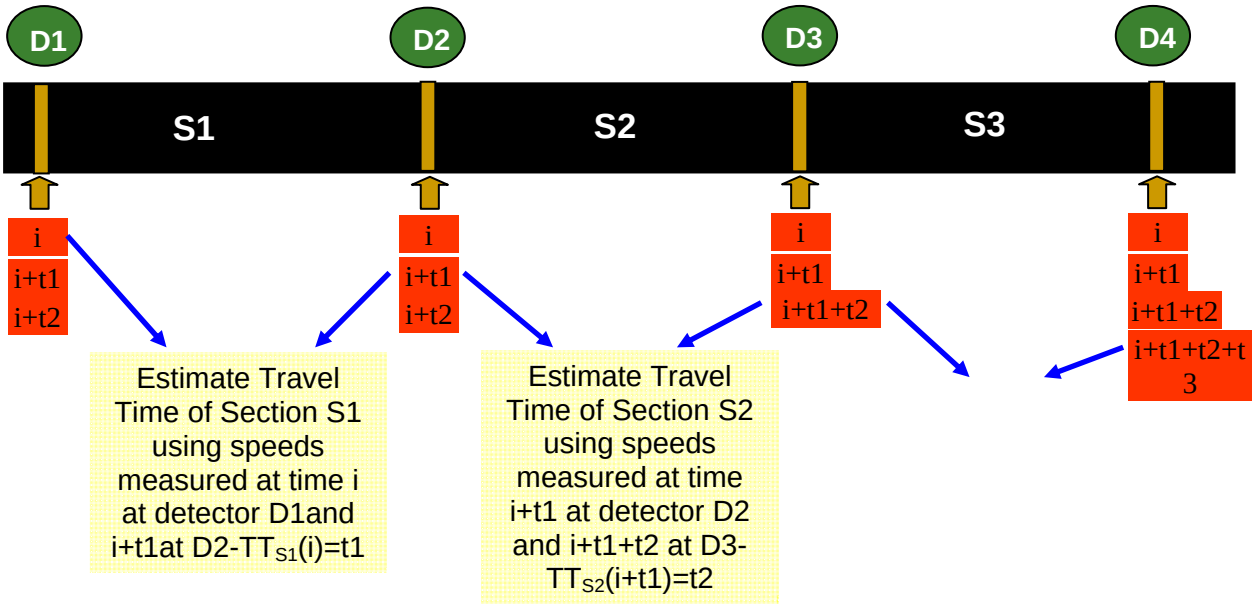

Figure 2.3: Dynamic Time Slice Model

Despite its obvious limitations in detecting the spatial and temporal evolution of congestion, speed-based estimation methods are often used, as they provide a relatively simplistic method of obtaining current travel times. Data requirements are relatively minimal and basic dual loop detectors, which are commonly deployed, can be used to obtain the required input data. The travel times obtained from speed-based methods are used as a basis to compare the performance of more complicated forecasting techniques and are sometimes used as inputs for many of the prediction techniques. These methods perform well under steady state conditions when there is relatively little congestion on the freeways. However, when the traffic conditions are unstable, such as during congestion build up and dissipation regimes, the speed-based methods are found to be relatively inaccurate as there is no mechanism to capture the congestion evolution.

2.1.4 Time Series-Based Models—Van Arem

A time series-based model was developed by Van Arem et. al. (1994) as a part of the GERDIEN project. As a part of the project, a network state monitoring and prediction (NSMP) system was developed, one of the primary functions of which is travel time estimation on the freeways. The NSMP has traffic detectors measuring speed, traffic volume, and occupancy on a temporal aggregation level of 1 minute. The model operates in two phases (see Figure 2.4). The first phase is to detect the presence of congestion. If the freeway section is found to be uncongested, the travel time on the freeway section is approximated to be equal to the free flow speed. If the freeway section is congested, the travel time on the freeway is equal to the sum of the free flow speed and the time delay experienced by a vehicle.

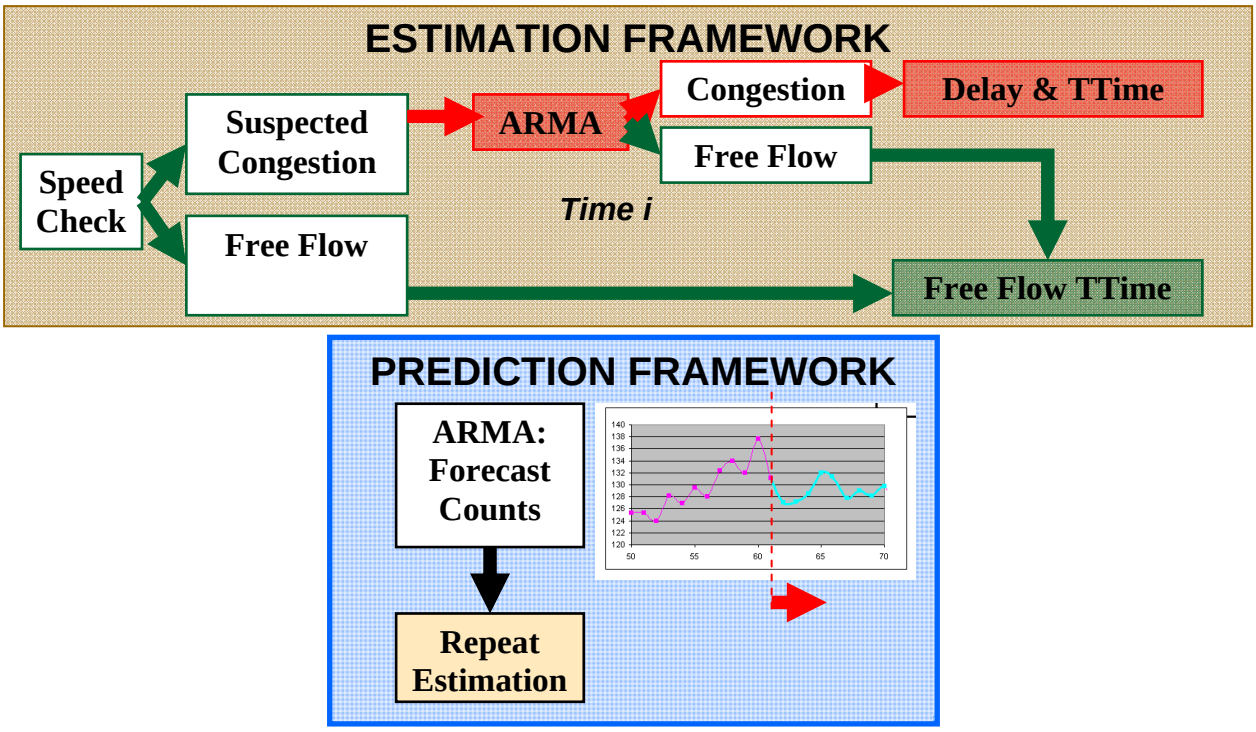

Figure 2.4: Travel time estimation framework of Van Arem model (Van Arem et al.1997)

Speed-based and time series-based criteria are used for congestion detection. In the speed-based criterion, if the minimum of the average speeds of vehicles entering or departing the freeway section is found to be lesser than a pre-specified threshold value, the freeway section is assumed to be congested. However, this does not account for cases when the congestion is in the middle of the section. In such cases, an ARMA time series-based model is used to detect congestion. In the time series model, the outflow in the section at time $Y(i)$ is written as a linear function of the outflow in the last n_o intervals and the inflow $X(i)$ in the previous n_i intervals starting at time $i-\Delta$ where Δ is the average time delay between inflow and outflow.

$$Y(i)=\sum_{j=1}^{n_o} a_j Y(i-j)+\sum_{k=1}^{n_i} b_k X(i-k-\Delta)+E(i)$$

The coefficients of the model a_j and b_k are calibrated from the data obtained from the previous M intervals. An impulse response function of the above ARMA model is calculated and used to detect the presence of congestion in the middle of the section. Thus if there is no congestion in the freeway section then the travel time in the section $TT(i)$ is approximated to be equal to the free flow speed $TT_F(i)$. If there is congestion in the system the travel time in the section is calculated as

$$TT(i)=TT_F(i)+W$$

Where W is a measure of time delay in the section. W is calculated as the ratio of the total number of excess vehicles in the system and the cumulative outflow.

In this study, the above model is extended in the following manner (see Figure 2.5). Speeds are used to calculate the travel time values in uncongested sections instead of free flow

speeds. Freeway segments with both in-ramps and out-ramps are considered and the time series model is used for prediction of travel times in the future instead of estimation.

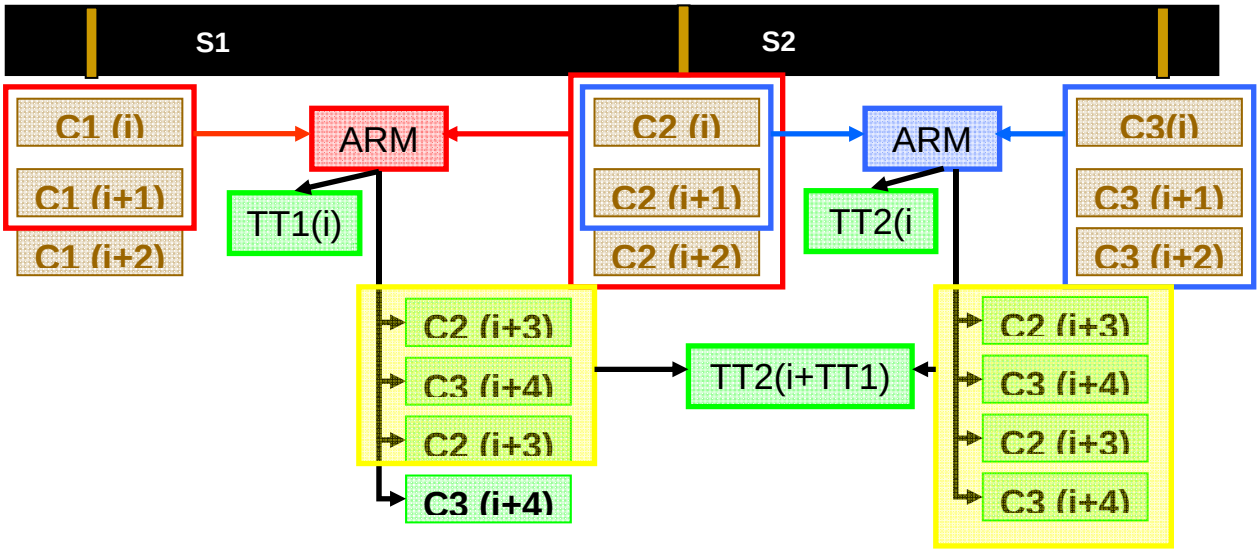

Figure 2.5: Applying Van Arem Model for Prediction

2.1.5 Numerical Analysis

In this section, the performance of two types of models—speed-based and time series-based estimation models—were compared using both simulated and real data. The simulated data was obtained from a 3.5-mile segment of El Paso, micro-simulated using the VISSIM model. The real-time data was obtained from the Berkeley highway laboratory. More details on the data sources will be provided in Chapter 5. The models were coded in C++ and speeds and counts were used as input. Half of the data set was used for calibrating the models, whereas the other half was used for testing the performance of the model.

The results for the estimation in the El Paso and the Berkeley Highway network are shown in Figure 2.6 and Figure 2.7:

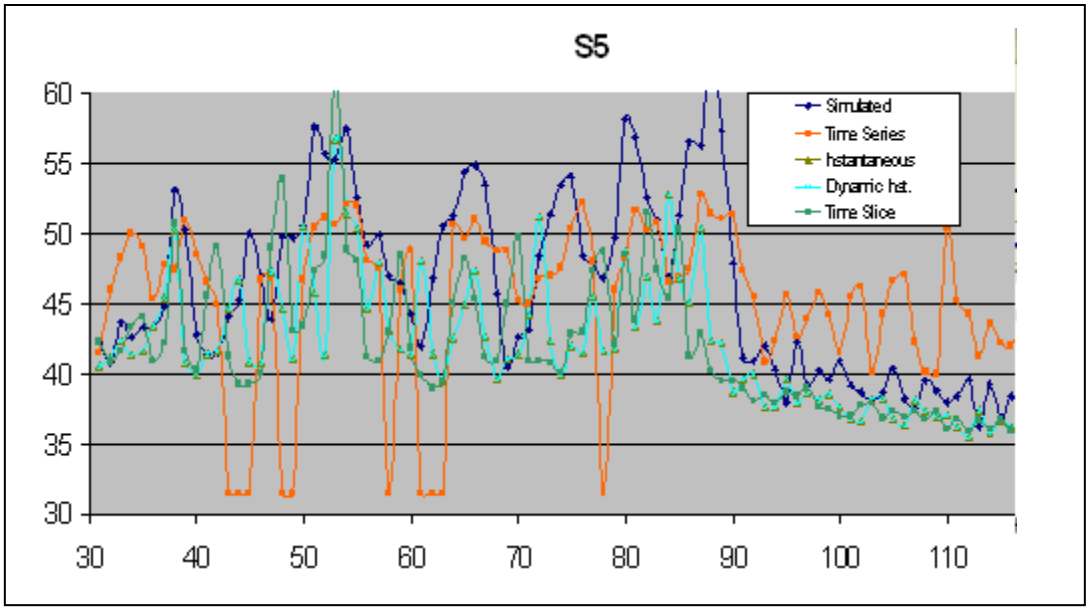

Figure 2.6: El Paso Speed-Based and Time Series Methods

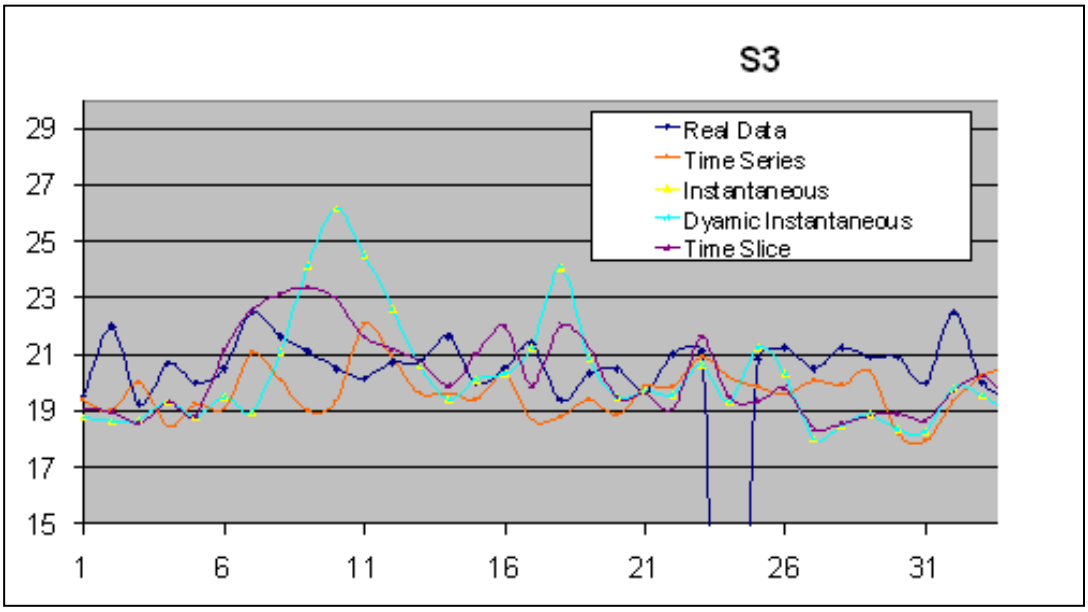

Figure 2.7: Berkeley Highway Network Speed-Based and Time Series Methods

The speed-based models were found to perform very well under uncongested conditions with an error rate of around 10 percent, while the time series models were found to over-estimate the travel times with an error rate of nearly 30 percent. Under congested conditions, the time series-based models over-perform the speed-based models, especially in the longer sections, with an error rate of around 10 percent.

For prediction, the performance of the model on a freeway section in El Paso is given in Figure 2.8.

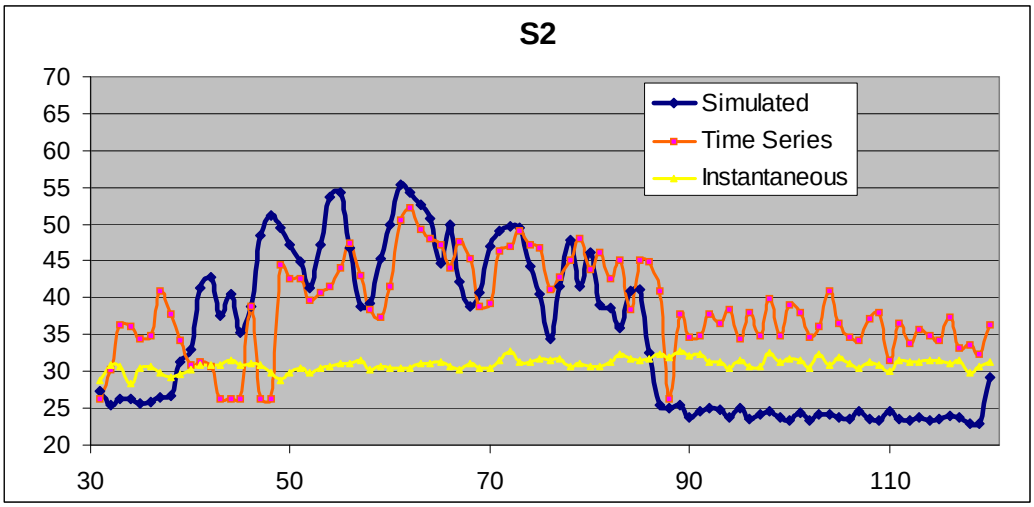

Figure 2.8: El Paso Network Travel Time Prediction

The analysis also shows that the model is highly sensitive to the temporal coverage of the data (see Figure 2.9). The model performance was found to deteriorate significantly when the temporal aggregation was 5 minutes compared to a 1-minute aggregation.

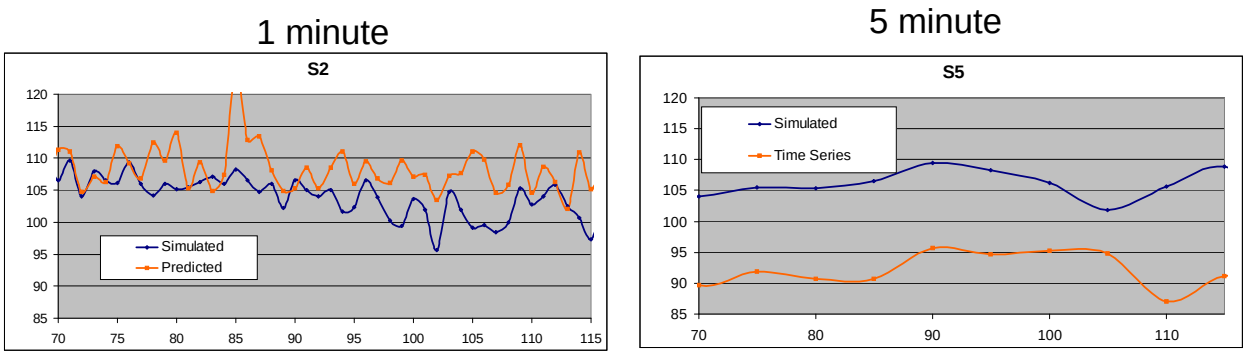

Figure 2.9: Temporal Aggregation

As with estimation, for prediction, the speed-based models were found to outperform the time series models during uncongested conditions, whereas the time series models outperformed the speed-based models for congested conditions.

2.1.6 Summary

The accuracy of the speed-based models is highly dependent on the quality and availability of speed data, which is generally more difficult to obtain than count data. Time series models require only count data and the parameters can be adjusted to reflect the current congestion levels. However, time series models require heavy detector coverage, especially in ramps. One of the disadvantages of the above models is that it does not account for traffic flow

relationships in the determination of the congestion state and in estimation of current travel times. To account for the evolving spatial and temporal dynamics, a simulation-based model complemented with a time series model for predicting inflows has been developed in this study. The simulation-based model also provides an excellent framework for analyzing detector locations and can handle absence of detectors on ramps.

Chapter 3. Integrated Traffic Simulation-Statistical Analysis Framework for the Online Prediction of Freeway Travel Time

Over the past few decades, congestion on the nation's highways has become a growing problem, especially in urban areas. Increasing the capacity of the transportation infrastructure by building new roads, or by adding lanes to the existing ones, is not always a feasible or desirable solution. Therefore, more emphasis is being placed on the implementation of Intelligent Transportation Systems (ITS) to enhance the efficient utilization of the available roadway capacity. An important component of ITS is the dissemination of information about the current and future state of the network, which is commonly done by displaying section travel-time estimates on Dynamic Message Signs (DMS). The provision of information to network users allows them to make better decisions, improving the system performance.

The information supplied by DMS can consist of instantaneous or future travel times. Instantaneous travel time estimates are based on present traffic conditions, and are therefore relatively easy to calculate. However, these predictions do not consider the impact of future downstream traffic conditions and can deviate significantly from experienced travel times during periods of moderate to high congestion. Therefore, methodologies capable of providing accurate and reliable travel time information should account for the evolution of traffic along the freeway.

The main objective of this chapter is the development of a simulation-based framework for the point-to-point freeway travel time prediction in the short term (3 to 7 minutes). The proposed methodology relies on traffic counts, provided by the simplest types of detectors, as its primary input, which makes it widely applicable.

While most of the research previously conducted in the area uses statistical or heuristic techniques to predict future travel time as a function of current and historical traffic flows (Chien et al., 2003), the present work derives travel time predictions from a calibrated traffic flow model fed by forecasted demands. The two-stage travel time prediction process introduced in this framework involves the use of a time series model to forecast the inflows into the traffic corridor and a Cell Transmission Model (CTM) to simulate the flow of these vehicles through the network. By doing this, the proposed approach takes advantage of the best characteristics of statistical techniques and traffic flow theory relationships. The utilization of a CTM (Daganzo, 1994, Daganzo, 1995*)* ensures that queue formation/dissipation, link spillovers, shockwave propagation, and other elements of traffic dynamics are accounted for. Additionally, the presented framework is computationally efficient, allowing for online updates of the freeway travel time estimates by means of a rolling-horizon approach (Wagner, 1977).

The overview of travel time prediction literature reveals the existences of deficiencies, some of which are addressed by the novel methodology developed in this work. First, by deriving travel times via simulation, the new approach is expected to be more effective than existing methods, which tend to approximate the complex, non-linear relationship between flows and travel times by a single statistical function. In addition, the proposed framework reserves the use of statistical methods for the forecast of the main sources of uncertainty (i.e., traffic demands), whereas it models explicitly the better-known traffic flow relationships. Finally, the model described in the next sections considers smaller prediction intervals (3 to 7 minutes) than the majority of previous work and relies on traffic counts as its principal input, which are available more often than probe-vehicle or automatic vehicle identification data.

3.1 Methodology

As described earlier, this work proposes a combined simulation-statistical analysis framework for the online, point-to-point freeway segment travel time prediction. Statistical techniques are used to predict vehicle inputs into the segment. The characteristics of the flow induced by these inputs, along with the corresponding travel times, are obtained via simulation. The proposed methodology requires the knowledge of online vehicle counts at the starting point of the segment under study. Besides, it makes use of the information provided by additional detectors located throughout the segment, when available.

For the purpose of this study, the traffic simulation is accomplished by means of a Cell Transmission Model (CTM), while vehicle inflows are predicted using Auto-Regressive Integrated Moving Average (ARIMA) time-series models. The following sections describe the implementation details of these approaches and the rolling horizon framework into which both model components have been integrated.

3.1.1 Traffic Simulation Component: a Cell Transmission Model

Cell transmission models, first introduced by Daganzo in 1994, simulate traffic behavior at a mesoscopic level. In these models, networks are represented as a string of connected cells in which vehicles are contained. CTMs use a discrete representation of time and compute cell-state variables (usually occupancy or density) at every time step via relatively simple numerical operations. The equations governing the displacement of vehicles from one cell to another are derived from the hydrodynamic theory of traffic flow (May 1990). Given that the modeling of individual freeway segments requires a relatively small number of cells, CTMs are highly efficient for online travel time prediction purposes. In addition, the discretization of the freeway segment into cells allows for an easy representation of traffic detectors, the location of entrance/exit ramps, and multiple origins and destinations for the point-to-point travel-time computation.

Despite some implicit simplifying assumptions, such as a trapezoidal flow-density relationship, the CTM framework adequately captures traffic dynamics, including queue formation and dissipation, link spillovers, and shockwave propagation. CTM is the traffic flow model embedded in VISTA (Waller et al., 1998, Ziliaskopoulos, 2000), a dynamic traffic simulator which has been successfully utilized by several transportation authorities to evaluate priority corridors, signal priority strategies, and impacts of network disruptions.

The next two sections provide details about the CTM implementation accomplished in this study and the travel time computation approach, respectively.

3.1.2 CTM Implementation

Figure 3.1 depicts the typical cell representation of a freeway segment in the context of this study. At each entrance ramp or traffic detector location, an auxiliary “gate cell”, holding an infinite number of vehicles at all times, is used to generate the appropriate inflows, which are regulated by the maximum capacity of the corresponding link. In order to accurately model the location of detectors and exit/entrance ramps, the flow of vehicles between cells is described according to a density-based CTM formulation (Munoz et al., 2004), which enables the use of variable cell lengths along the segment. For cells not connected by merge or diverge links, equation 1 relates the density (expressed in vehicles per mile) at time intervals k and $k+1$ as follows:

$$\rho_i(k+1) = \rho_i(k) + \frac{\Delta}{l_i}(q_{i-1,i}(k) - q_{i,i+1}(k))$$

where Δ is the simulation interval and l_i stands for the length of cell *i*. The flows leaving from and arriving to cell *i* during time interval *k* are given by $q_{i,i+1}(k)$ and $q_{i-1,i}(k)$ respectively. These are obtained as the minimum of two quantities: the highest flow which can be supplied by the upstream cell ($S_{i-1}(k)$), and the maximum flow that the downstream cell can receive ($R_i(k)$). Equations 2 and 3 describe the computation of these values for every cell:

$$S_{i-1}(k) = \min(\upsilon\rho_{i-1}(k), Q_{M,i-1})$$
$$R_i(k) = \min(Q_{M,i}, \omega*(\rho_J - \rho_i(k)))$$

where υ refers to the free flow speed, $Q_{M,i-1}$ is the capacity (in vehicles per hour) of cell *i-1*, and ρ_J is the jam density. The above equations constitute the core of a CTM. The reader may refer to Muñoz et al., 2004, for a detailed description of their derivation. Additionally, these relationships need to be modified appropriately in order to model flows among cells connected by merge or diverge links. The corresponding equations (for an occupancy-based CTM) are presented in Daganzo, 1994, and can be easily modified to fit in the density-based CTM formulation applied in this study.

The CTM formulation is based on the assumption that, at any simulation time step, vehicles can travel, at most, the length of a cell. Therefore, the selections of free-flow speeds, minimum cell length, and simulation time steps are closely related, and constitute a fundamental modeling component. Given that the CTM formulation implies an even distribution of vehicles within a cell (single density value per cell and time interval) along with constant link flows (including input flows) throughout a simulation step, small cell lengths and simulation steps are desirable. However, these values are limited by practical and computational considerations. Past experience in the usage of CTM for freeway traffic simulation (Chien et al., 2003, Daganzo, 1994) suggests the convenience of adopting step lengths between 3 and 6 seconds. For the model implementation presented in Section 4, a 4-second interval was selected. This choice provided the flexibility to define sufficiently small cell lengths for free flow speeds in the 55-65 mph range, allowing for an appropriated modeling of detectors and ramps location.

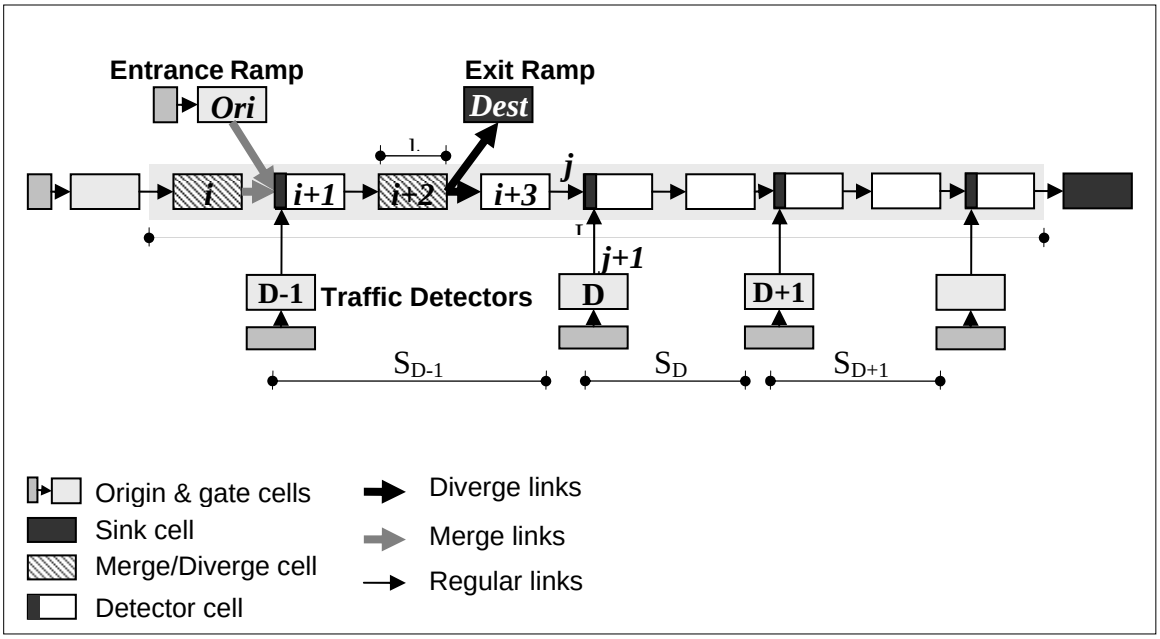

Figure 3.1: Cell Transmission Model description

3.1.3 Travel Time Computation

In the present CTM implementation, the point-to-point travel time computation is performed based on the comparison of cumulative counts at the desired entrance and exit points. When detectors, merge links, or diverge links are present within a segment, the travel time is computed by sub-segments containing only regular internal links. The corresponding segment travel time is obtained recursively, according to equations 4 and 5:

$$T_1(k) = \tau_1(k)$$
$$T_s(k) = \tau_s[k + T_{s-1}(k)] \quad \forall\, S > s > 2$$

where $T_1(k)$ is the cumulative travel time up to sub-segment *s* at time *k*, $\tau_1(k)$ is the sub-segment travel time, and *S* represents the total number of sub-segments.

If the freeway segment under analysis accounts for several traffic detectors, separate CTM model runs are conducted at every time interval. These simulations start at the location of each detector and use the corresponding input data. The travel time computation is accomplished on a "sector" basis, where sectors such as S_D (Figure 3.1) are defined as the set of cells located between detectors *D* and *D+1*. For all the origins and destinations contained within S_D, travel times are derived from the CTM simulation based on detector D.

The presence of intermediate detectors may lead to the definition of sub-segments shorter than 0.5 miles, on which the travel times can be extremely short. Under this setting, accurate cumulative-count-based travel time estimations would demand a very precise modeling of vehicles' position at each time interval, usually beyond the reach of mesoscopic traffic models. Therefore, a poorer framework performance can be expected when excessively short sub-segments are modeled.

3.2 Statistical Analysis Component: Traffic Counts Forecasts

Running a CTM simulation requires knowledge of the inflows at every network entrance point throughout the length of the simulation. The latter should be large enough to permit the exit of all the vehicles entering the section during the period of interest. If travel times estimates are desired for *N* time intervals starting at time *k*=0, the simulation should be run (at the very least) until all the vehicles that entered the freeway at times *k=1,2…N* reach the exit point. This demands a corresponding provision of input flows at all segment entrance points, which needs to be forecasted based on the traffic counts observed up to time *k=0*.

In this study, an Auto-Regressive Integrated Moving-Average (ARIMA) model has been chosen to predict the necessary traffic inflows. Although this is a relatively simple time series model (readers might refer to any time-series analysis text, such as Brockwell and Davis, 2002, for a detailed description of ARIMA models), its performance was found to be comparable to that of more complex methodologies. In addition, ARIMA models are relatively easy to implement, do not require extensive input data, and are computationally efficient, enabling their incorporation within a rolling horizon framework.

3.3 Online Framework Implementation

Online travel time predictions are obtained by integrating the simulation and statistical model components within a rolling-horizon approach, in which vehicle inflow data collected from traffic detectors and initial/border conditions are updated in real time. The vehicle counts provided by traffic detectors are aggregated appropriately and used to calibrate an ARIMA model, which provides counts forecasts. The online information is utilized in conjunction with these forecasts to run an offline calibrated CTM at every desired point in time "*t*", as depicted by Figure 3.2.

Even though the proposed rolling horizon framework implements a CTM online, most of the core model parameters require offline calibration. These include jam density $\rho_J(k)$, capacity $Q_M(k)$, diverge split ratio for diverge cells $B_t(k)$, priority factor for merge cells p, free flow speed v and congestion parameter ω. The details of the impact and usage of the above parameters can be found in Daganzo, 1994 and 1995. By allowing for time-varying densities and capacities, the approach presented in this paper provides the flexibility to model lane closures and incidents. Similarly, the utilization of a time-dependent B_t parameter permits modeling origin-destination patterns that vary throughout the day. Although the values of B_t can be exogenously determined by means of historical data, it is also possible to refine or correct them based on online counts.

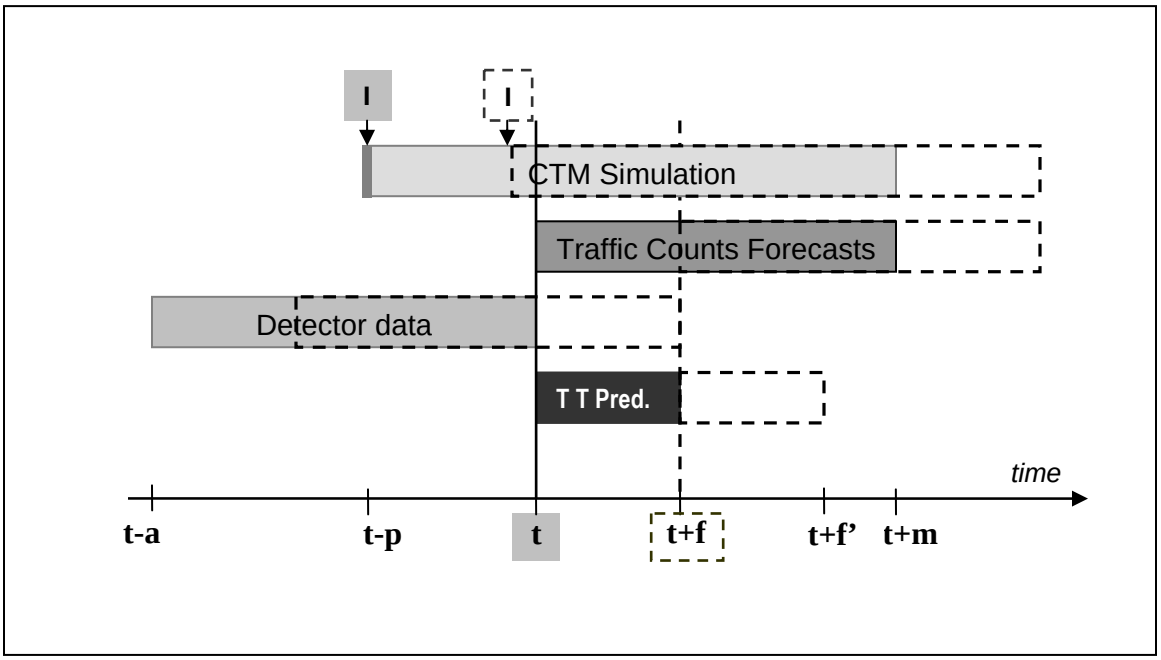

Figure 3.2: Rolling-horizon implementation

Figure 3.2 describes the implementation of the rolling horizon approach. At time *t*, the prediction horizon *f* is the maximum time for which travel time predictions will be provided. The forecasting horizon *m* represents the time up to which the CTM will be run, allowing for the exit of all the vehicles that entered the section during the [*t,t+f*] interval. The ARIMA model uses detector data cumulated during the last *a* intervals to provide the traffic counts forecasts needed to feed the CTM between times *t* and *t+m*. The CTM simulation starts at time *t-p*, and it provides the travel time predictions for the interval [*t,t+f*]. At time *t+f*, the prediction horizon is moved to *f*, and *a*, *m*, and *p* are shifted accordingly. Notice that the initial conditions for this new CTM run are taken from the appropriate density values predicted by the previous CTM run (time *I*). However, when detector counts are available at intermediate points, the comparison of real detector counts and CTM-predicted flows on links such as *j* and *j+1* can be used to heuristically improve the starting densities at the beginning of each CTM run. Section 3.3.1 presents a possible approach for the heuristic refinement of initial densities.

3.3.2 Heuristic Approach for the Update of Initial Conditions.

One important factor affecting the accuracy of a CTM is the precision of the initial conditions, which are given by the density at every cell at the beginning of the simulation. In the presented rolling-horizon approach, the initial conditions for each model run can be extracted from the density values provided by the previous model execution, as described in section 3.3. Because of unavoidable modeling imperfections, the cell densities predicted by the model may not correspond exactly to the actual distribution of vehicles throughout the freeway segment. Although these deviances are not anticipated to have a large impact in the travel time prediction of a single model run, their effect is likely to become noticeable as the rolling horizon moves forward.

For the purpose of this study, a simple heuristic approach has been adopted as a means to reduce the impacts of errors in the initial conditions. The methodology makes use of the

information provided by additional detectors placed along the freeway segment, which are modeled as indicated by Figure 1. The heuristic density correction is based on the comparison of real cumulative flows on links such as *j+1* and the cumulative link flows obtained from the model for links such as j. At time t, these can be computed according to equations given below:

$$MF_t^D = \sum_{k=t-i}^{k=t} q_j(k)$$

$$RF_t^D = \sum_{k=t-i}^{k=t} q_{j+1}(k)$$

where MF_t^D is the cumulative flow predicted by the model, and RF_t^D is the real cumulative flow, as provided by traffic detectors.

The heuristic approach proposed in this work is applied when $MF_t^D - RF_t^D$ exceeds a pre-defined threshold (heuristically set to 20 percent after extensive numerical analyses), and the appropriate refinements are decided based on the sign of this difference and the congestion level. If $MF_t^D < RF_t^D$ and the density at sector S_{D-1} is close to ρ_J, it can be concluded that the model is exaggerating the existing congestion. Given that the inputs to S_D are obtained from detector counts (and therefore assumed to be correct), the excessive congestion is attributed to overly large initial densities at S_{D-1} cells, which are reduced based on $MF_t^D - RF_t^D$. Conversely, if the densities at S_D are low while the model underestimates the cumulative outputs, the previous line of reasoning leads to the need for an appropriate density increase. Similar analyses are performed when $MF_t^D > RF_t^D$, resulting in heuristic augments and reductions of S_D densities when these are low or moderate, respectively.

3.4 Summary

In this chapter an integrated statistical simulation based model was presented for travel time predictions. The model predicts the inflow into the freeway network using an ARIMA model. The forecasted inputs are then simulated using a Cell Transmission Model to obtain the travel times. The new model accounts for the evolving dynamics of traffic flow in predictions and is computationally efficient. Therefore, it can be run in a normal personal computer as long as data as available.

Chapter 4. N Curve Model

4.1 Model Description

In 1997, Daganzo proposed travel time prediction on a freeway under work zone conditions [1]. Daganzo described this approach by modifying input-output diagrams to measure the time and distance spent by vehicles in a queue in a simpler manner than using a time-space diagram. This process requires the construction of a curve depicting the cumulative number of vehicles reaching the back of queue as a function of time, refer to Figure 4.1.

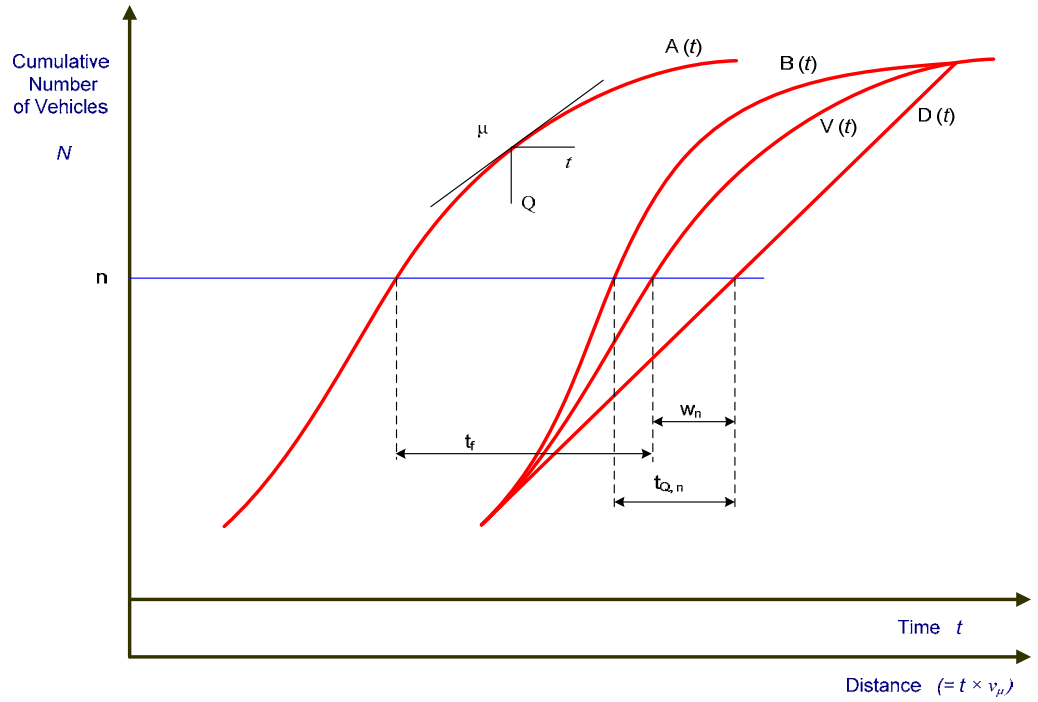

Figure 4.1: N-Curve (Input-Output) Based Method [1]

Figure 4.1 demonstrates the arrival time of each vehicle at an upstream observation point where it is measured and plotted on the figure as the curve $A(t)$. By translating the arrival time of each vehicle horizontally to the right by the free-flow travel time to the bottleneck, t_f, the desired (or "virtual") arrival time of each vehicle at the bottleneck can be plotted as the curve $V(t)$. Finally, the departure curve, defining the time that each vehicle departed the bottleneck, can then be constructed in the usual way to serve the virtual arrivals at a maximum rate m. For a given vehicle number n, the horizontal separation between $V(t)$ and $D(t)$ represents the delay for that vehicle, and is denoted w_n; and the horizontal separation between $A(t)$ and $D(t)$ represents the total delayed travel time for that vehicle, t_{q_n}.

Using the relationships derived from a space time diagram, the input-output diagram of the figure above can be modified to include the curve $B(t)$, the number of vehicles to reach the back of the queue by time t, or equivalently the times that each vehicle reaches the back of the queue (Figure 4.1). Time can be determined for each vehicle that joins the back of the queue by "extending" its delay, w_n. The locus of these points for all vehicles represents the "back of queue" curve, $B(t)$, which can now be constructed on the input-output diagram (Figure 4.1). Obviously, $B(t)$ will differ from $V(t)$ only for those vehicles for which $V(t)$ differs from $D(t)$; i.e., whenever a queue is present.

Daganzo's model development and case study are based on the hind-sight information on a closed system of a freeway. The advantages of utilizing the *N*-Curve model can be summarized as follows: 1) It requires less data for training calibrating by only requiring traffic flow counts for the upstream, downstream, and ramp points; 2) less data also means more robustness and generality when implemented; and 3) it can handle more traffic situations than other models while generating more accurate results. Nevertheless, this approach assumes a closed highway segment without considering on- or off-ramps between the upstream and downstream detectors. In the case of an off-ramp existing between the upstream and downstream detector, the highway segment loses its conversation of flow and the cumulative curve at the downstream detector cannot be directly used for travel time prediction without further modifications. In the event that both on- and off-ramps exist in the segment of interest, the situation could become more complicated.

4.2 Methodology

A generalized model development of an N-Curve based method which accounts for general freeway configuration, including on- and off-ramps, is proposed to estimate and predict travel time on a freeway segment.

Definition of Variables:

π_i: arrival time at detector i

N_i: Cumulative curve for detector i, $\forall\, i \in I : \{m, ..., j\}$

$N_i(\pi_i)$: N-curve marker when arriving at detector i at time π_i

$T_{i,i+1}(\pi_i)$: travel time between detector i and i+1 when arriving at detector i at time π_i.

$\Phi_{m,j}(\pi_m)$: arrival time at detector j when the entire journey starts at detector m at time π_m

By definition, $\Phi_{\mathrm{m,j}}(\pi_m) = N^{-1}\left[N_{j-1}(\pi_{j-1})\right]$, we can also show (proof omitted here) that

$$\Phi_{\mathrm{m,j}}(\pi_m) = N^{-1}\left[N_{j-1}(\pi_{j-1})\right] = \sum_{i=m}^{j} T_{i,i+1}(\pi_i) + \pi_m$$

This means that to find the arrival time at detector j, we can find the time-dependent travel time for each detector pair within m and j and sum up these travel times and the arrival time at detector m.

4.2.1 Travel Time Estimation Without Ramps

To initiate travel time prediction utilizing the base algorithm, the origin and destination points (or detectors) of the traveled path must be defined; in this case the origin detector is defined as m while the j is the destination detector. Detector i is defined to be in the set of I, where $I:\{m,...,j\}$. Travel time, $T_{i,i+1}(\pi_i)$, for two sensors for the complete simulation time, $\pi=\{0,...,T\}$.

The computations then iterates as follows:

Step 0: m = i

Step 1: $T_{i,\,i+1}(\pi_i)=N_{i+1}^{-1}[N_i(\pi_i)]-\pi_i$

Step 2: $\pi_{i+1}=\pi_i+T_{i,\,i+1}(\pi_i)$

Step3: $\Phi_{m,j}(\pi_m)=\pi_{i+1}$, stop if $i+1=j$

otherwise $i=i+1$, go to Step 1

Refer to Figure 4.2 for a graphical representation of the base algorithm.

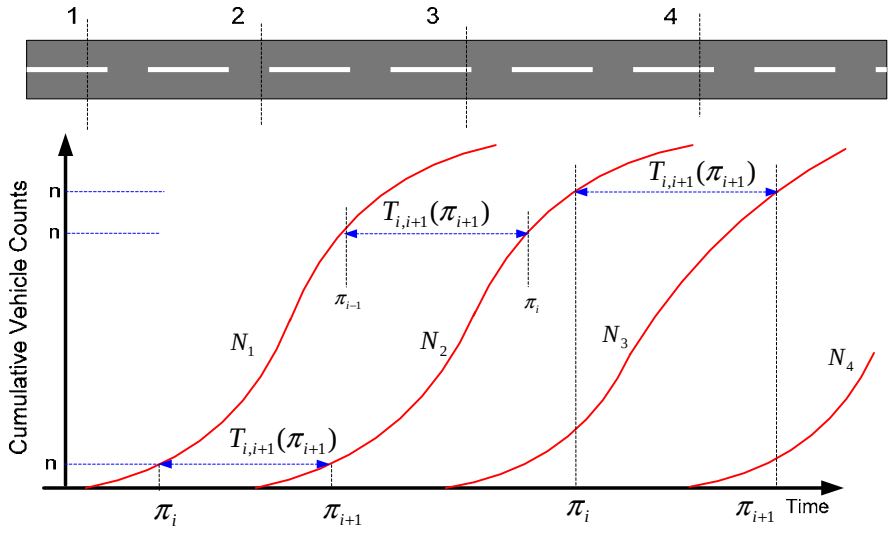

Figure 4.2: Mainlane N-Curve Method

4.2.2 Travel Time Estimation with Ramps

To account for the presence of on- and off-ramps in the prediction corridor of interest, the base algorithm requires of a couple of modifications. Clearly, the technique eliminates the *closed system* found in the without-ramp case, meaning traffic flow conservation is no longer true between sensors placed upstream and downstream of the ramp junction. Thus, the detector set *I* is now expanded to $I''=I+I'$, which includes the original detectors, $I:\{m,...,j\}$, and the virtual detectors , $I'=\{K'\}$. *K'* is the downstream virtual detector corresponding to *K*, where *K* is located upstream of either an on or off-ramp and is an element of *I*. This results in the creation of

'virtual' detectors *i'* for any main-lane detector upstream *i* of an on- or off-ramp. The *N*-curve for the virtual detector *i* then becomes:

$N_{i'} = N_i + N_{i''}$ if detector *i* is the upstream detector for an on-ramp

$N_{i'} = N_i - N_{i''}$ if detector *i* is the upstream detector for an off-ramp

Note that this virtual detector is assumed to be at the locations immediately downstream adjacent to the ramp. Following the previously defined iterative travel time process, travel times are calculated only for sensor pairs with flow conservation, including both actual and virtual detectors. As to be discussed in the later statements, the travel time between the detector *i* and *i"* needs to be estimated. At this moment, we assume that this travel time equals to ε.

Refer to Figures 4.3 and 4.4 for a graphical representation of the on- and off-ramp algorithm.

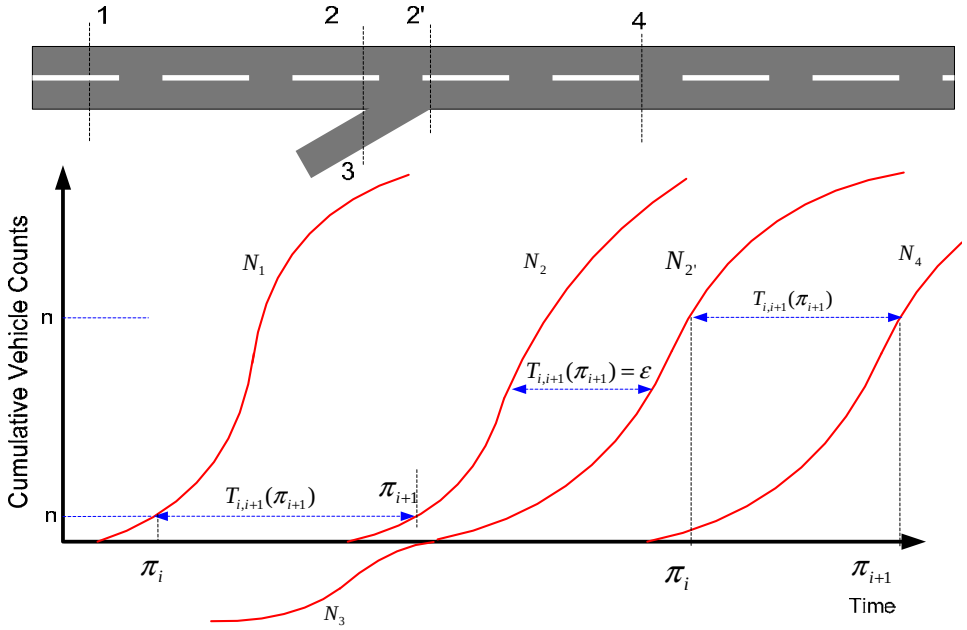

Figure 4.3: On-Ramp N-Curve Method

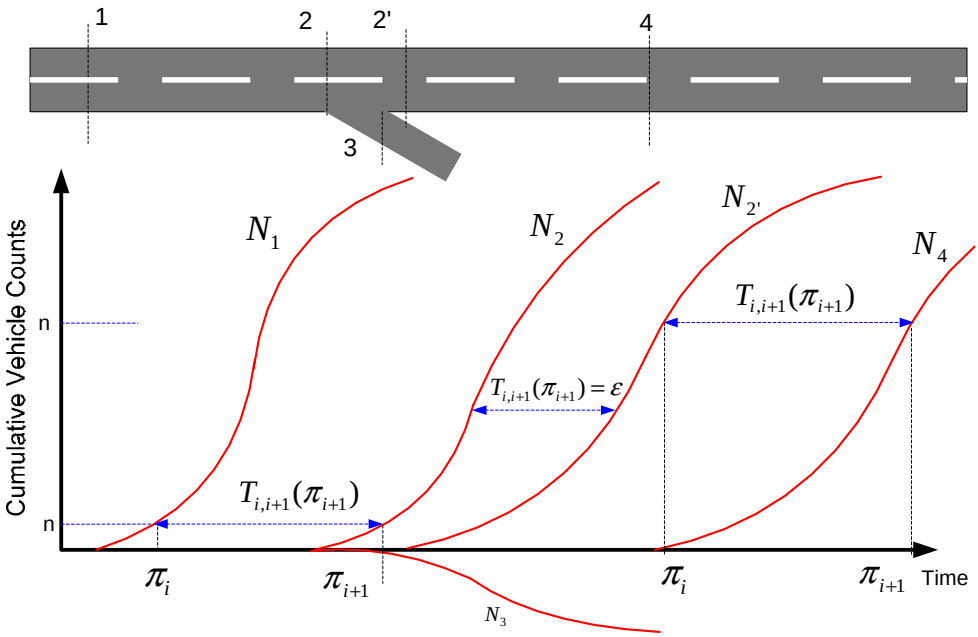

Figure 4.4: Off-Ramp N-Curve Method

Chapter 5. Data Requirements

This chapter describes the different types of data required to calibrate, validate, and ultimately implement the models developed for this project, along with the data sources utilized to obtain the results presented in the following chapters. Additionally, the final section provides recommendations for the pre-processing of input data, namely traffic counts, in real time. The later are necessary given that the measurements provided by detectors may contain errors which, if neglected, could have detrimental effects on the performance of some travel time prediction models.

5.1 Data Types

Two main types of data are required, alone or in combination, in order to utilize the models presented in this report: traffic data and travel time measurements. ***Traffic data*** consists of vehicle counts, as provided by detectors deployed on the field. The data may be available at different aggregation levels, and ideally the most disaggregated version compatible with a real-time deployment should be utilized. Traffic data is necessary for all the stages of model development and implementation. ***Travel time measurements*** were used in this research to validate the model performance, by comparing them to the travel time predictions generated using the models. They are also necessary to fine-tune model parameters, and when the model is utilized to determine optimal deployment strategies.

In addition to the main data types, the models need to be tailored to reflect the ***geometric characteristics*** (Figure 5.1) of the segment under analysis. These include the number of lanes throughout the segment, number and position of entry/exit ramps and the corresponding split ratios, and the number and location of traffic detectors. Experienced-based knowledge regarding prevailing driving speeds, recurrent congestion, and general drivers' behavior may be useful during the model calibration and validation process.

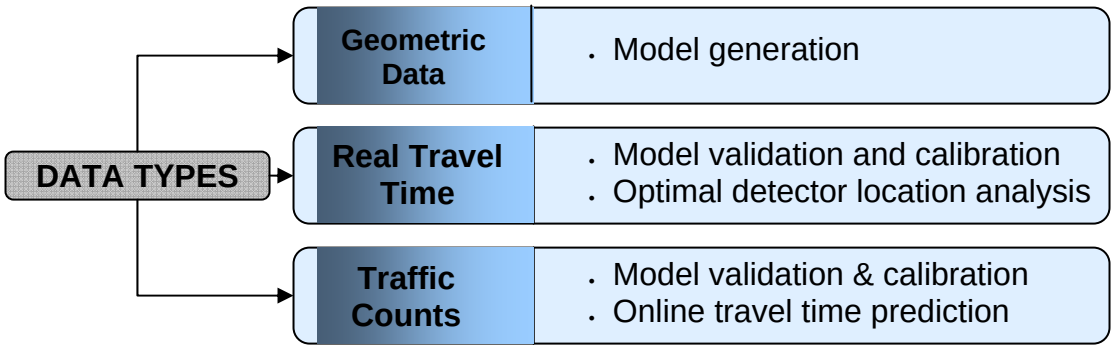

Figure 5.1: Travel-time prediction data types

5.2 Types of Data Sources

When the models presented in this report are used to generate online travel time predictions, the only valid source of data is given by traffic detectors, which provide the counts

used as the sole input at run-time. However, the validation and calibration stages demand additional data, including historical traffic counts at the segment under analysis and the corresponding real travel time measurements. Ideally both data sets should be collected on the site where the model is going to be implemented (***field data***), as shown in Figure 5.2. Historical traffic counts can be obtained from the same detectors that are going to be used for the predictions. At least one or two peak periods, including congestion formation and dissipation, should be available. The most straightforward source of real travel time measurements is the utilization of vehicles equipped with GPS systems. Alternatively, they may be derived analyzing AVI data or applying vehicle re-identification algorithms (Coifman, 1998, Sun et al., 1999). The travel time collection handbook (FHWA 1998) describes and compares these and other techniques. Real travel time measurements are required throughout the period used for calibration/validation, and a relatively dense data set is necessary to obtain accurate results.

Field data may not be available in the amount and/or level of aggregation demanded by the validation/calibration processes. In such scenario, traffic counts and real-time measurements may be ***simulated*** using commercial microsimulators after modeling the segment under study utilizing the corresponding geometric data. In order to generate traffic counts and travel times, it is also necessary to feed approximate vehicle counts at the main entry points to the segment under study, which can be collected using existing traffic detectors or estimated based on other sources of historical data. Simulated data has a number of advantages: it's error-free, can be used to test the model under different scenarios, and is available at any desired aggregation level. Furthermore, when optimal sensor location is analyzed, only the use of simulated data allows considering arbitrary deployment patterns. However, field data reflects the actual behavior of local drivers under real conditions and should be preferred over simulated data when available.

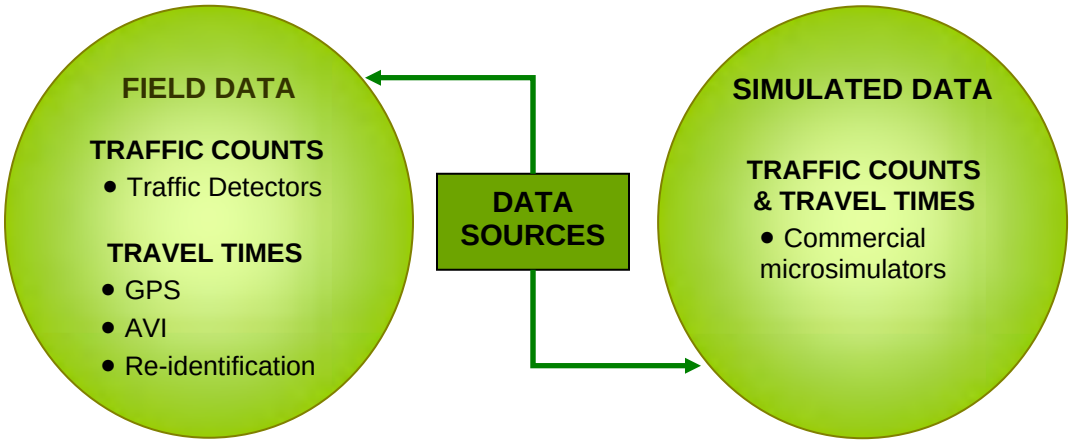

Figure 5.2: Travel-time prediction data sources

5.3 Utilized Data Sources

This section describes the different sources of data used for testing, validating, and calibrating the developed models. These include microsimulator-based data, as well as field data. Even though the latter was preferred when available, some model applications demanded simulated data. Furthermore, the complete set of field data became available during the final

project stages, and therefore most of the model validation and testing was performed using simulated data.

5.3.1 Simulated Data: VISSIM

VISSIM (PTV America Inc.) is a traffic microsimulator that models the movement of individual vehicles through complex equations. As are many advanced traffic simulation tools, it is expected to produce very realistic results by allowing for an accurate representation of geometric conditions, vehicles' characteristics, and drivers' behavior. The microsimulator can be adjusted to generate output files that replicate the data that would be provided by different types of real traffic detectors. Similarly, it can provide travel times between any pair of points in the roadway, consistent with the detector measurements. All the data originated from the simulator is available at any desired aggregation level, which permits the testing of the model performance under different data availability scenarios.

A 3-mile segment of the El Paso Border Highway, including five entry ramps and five exit ramps, was modeled using VISSIM (Figure 5.3). Simulations were run utilizing different levels of randomly generated trip demands, and varying positions of traffic detectors. Two hours of traffic were simulated in most cases, in order to allow the model to stabilize and produce reliable results. The output files, consisting of traffic counts and the corresponding travel times computed for every 4-second interval, were processed utilizing a C++ code to match the requirements of the different travel time prediction models. The program split the traffic data file into separate files per detector and aggregated the travel time measurements at different desired levels.

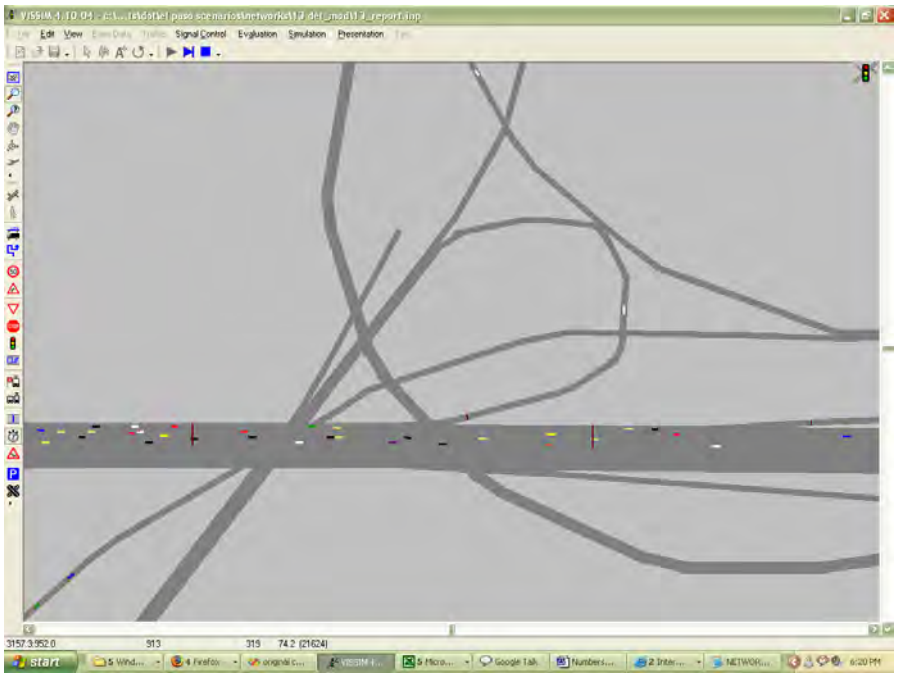

Figure 5.3: Snapshot of VISSIM microsimulation

5.3.2 Field Data: BHL

The Berkeley Highway Laboratory is a heavily equipped 2.7-mile section of Interstate 80, connecting Berkeley and Emeryville. Figure 5.4 depicts the geometrical characteristics of the highway segment. Historical data, as well as real-time traffic detector data, is available at a variety of aggregation rates at http://bhl.its.berkeley.edu:9006/bhl/traffic/historicaltraffic.html. Traffic counts are accompanied with the corresponding real travel times, obtained via a re-identification algorithm based on the recognition of vehicle patterns. The availability of both data types makes this dataset valuable, despite the relatively small length of the monitored highway segment and the lack of traffic volumes data for entry/exit ramps.

The BHL data was used to test and validate preliminary models for online travel time prediction. Traffic counts, corresponding to the morning peak hour during a week day, were downloaded from the abovementioned website at the minimum possible aggregation level and processed to match the requirements of the different models. The corresponding travel times were obtained by contacting the researchers at the BHL, as suggested in the webpage, and also required some processing in order to be analyzed and compared to models' results.

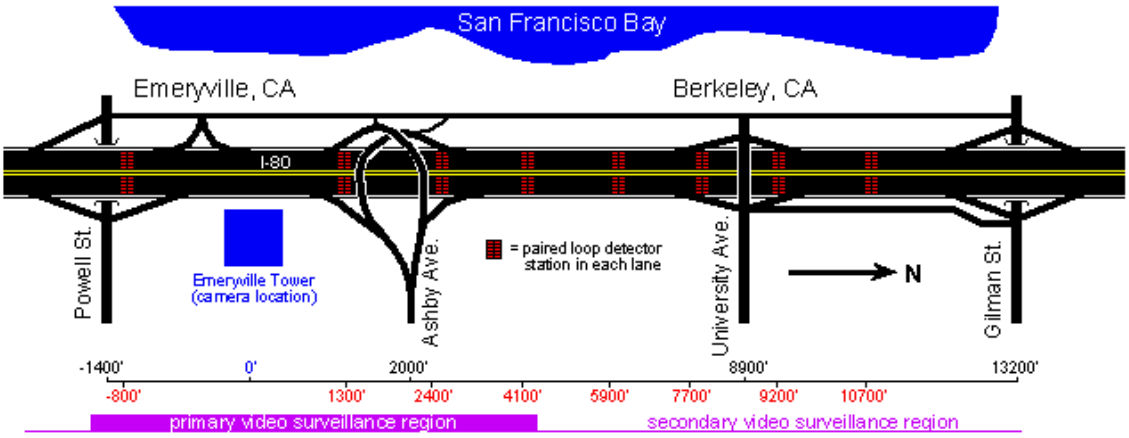

Figure 5.4: Berkeley Highway Laboratory Layout
Source http://bhl.its.berkeley.edu:9006/bhl/about.html

5.3.3 Field Data: TransVista traffic detectors and GPS data

TransVista is the El Paso Traffic Management System. It consists of a number of traffic detectors, changeable signs, and data processing/storing infrastructure, utilized to manage local traffic. Historic traffic counts for model validation and calibration were obtained from 77 of this system's detectors, spanning 37 miles of highway, 11 of which correspond to the El Paso Border Highway.

Detector data was processed and organized into a PostgreSQL database, spanning several days in June. The database, along with the corresponding usage instruction and complementary visualization software, were delivered as product P5. The software includes a tool to plot different data components, including traffic volumes and speed (Figure 5.5). Additionally, a Graphical User Interface (GUI) was provided (Figure 5.6), which can be used to retrieve the data collected by each detector in real time, provided that the database is updated accordingly.

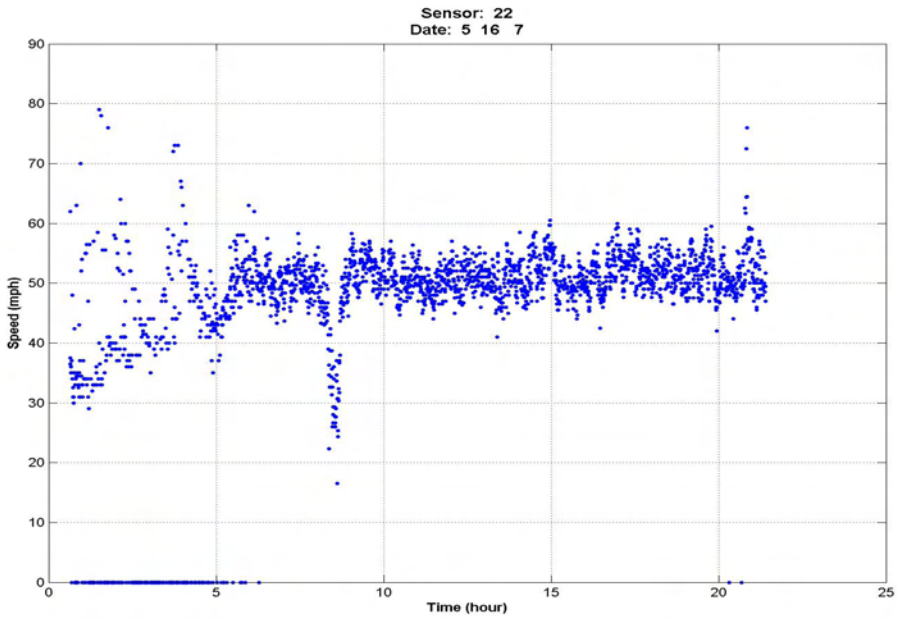

Figure 5.5: Sample of data plots (speed) from the traffic data base

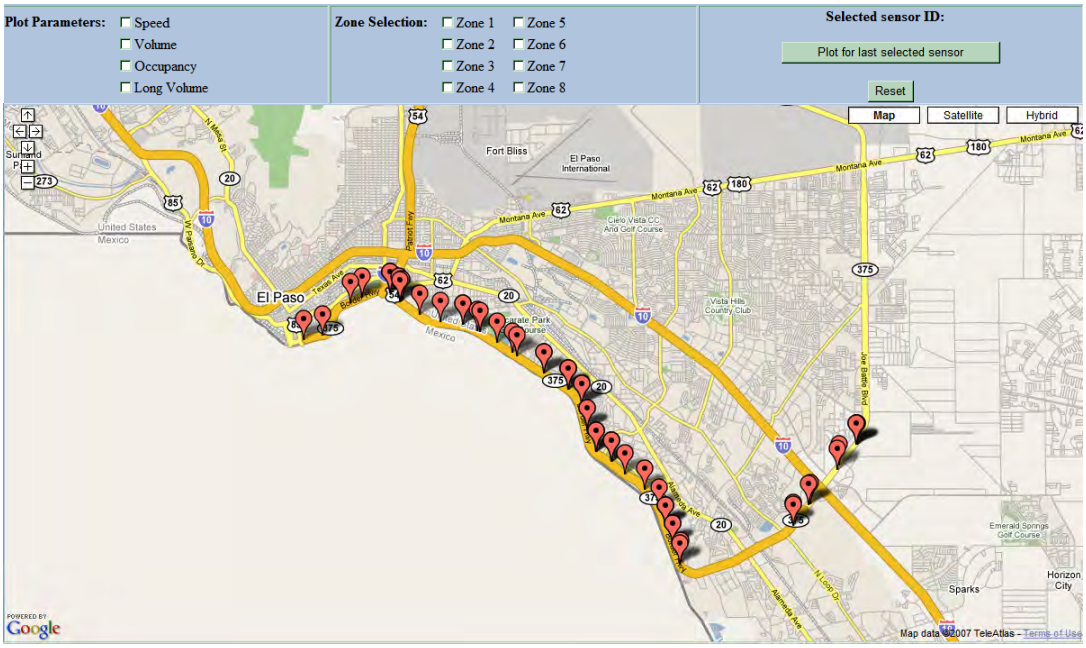

Figure 5.6: Graphical User Interface

Real travel time data was collected utilizing a vehicle equipped with a Geo Positioning System (GPS). The car traveled through a 10-mile section of the El Paso Border Highway on which TransVista traffic detectors were deployed (see Figure 5.7). Between June 4 and June 29, 2007, real travel time data was collected during the morning peak hour (6:00–8:00 AM), by completing two or three trips in each direction.

The GPS device recorded the coordinates of the vehicle every 5–10 seconds, and a C++ code was developed to process the resulting data. The program read the raw GPS file and generated travel times between roadway sections by subtracting the corresponding time stamps. By comparing the coordinates of each of the points retrieved by the GPS to the coordinates of the desired reference points on the highway (corresponding to the position of TransVista traffic detectors), the code was able to select the appropriate data pairs for the travel time computation. Each two-way trip was recorded into the same file, and the program automatically identified the point at which the vehicle switched directions. A table reflecting the processed data was provided with the deliverables as part of product P5.

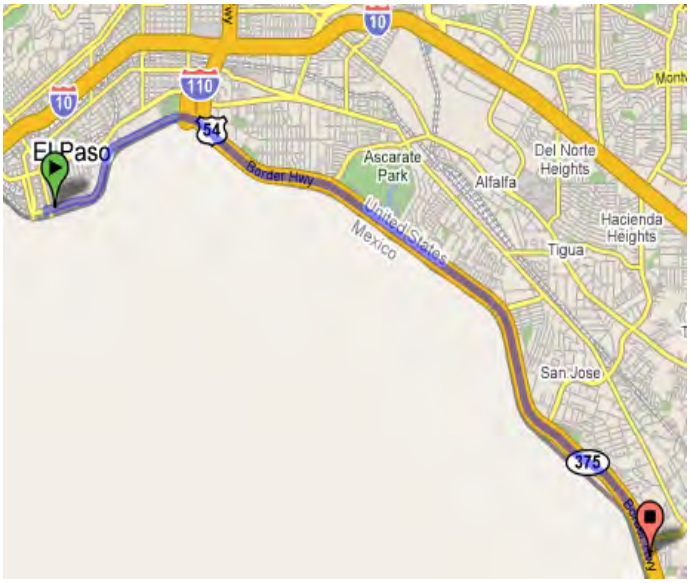

Figure 5.7: Route followed by the GPS equipped vehicle

5.4 Data Pre Processing

One of the salient advantages of the models presented in this report is that they only require traffic count data at deployment time. Such data is typically readily available and fairly reliable. Nevertheless, as most automatic data collection devices, traffic detectors may produce flawed data. This can clearly affect a model's performance, as it was observed during the experimental model deployment conducted at El Paso TMC. In order to mitigate the negative impacts of data errors, ***procedures are needed to determine whether detectors are working and whether their measurements are accurate***. Such processes are highly dependent upon the characteristics of the detectors used at each TMC, the prevailing communication protocols, and the structure of the implemented travel time prediction models. There has been a considerable number of works dealing with the issue of assessing the quality of traffic detector data. The following paragraphs will briefly describe the basic principles underlying existing research, and suggest possible references for an eventual deployment of the proposed models into a TMC. The data filtering process involves two basic operations, depicted in Figure 5.8.

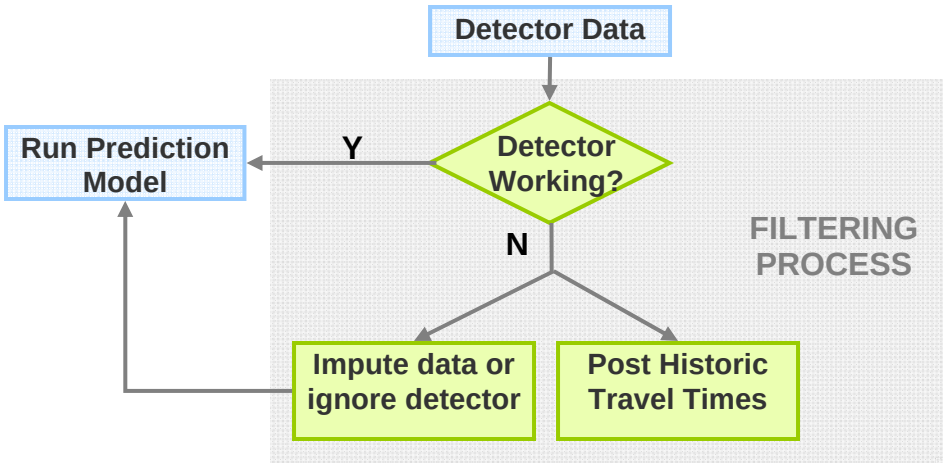

Figure 5.8: Simplified flow chart of the data filtering process

Several approaches can be taken to identify whether a detector is working properly. The simplest ones, which are often used given the time constraints imposed by the real-time nature of the travel time prediction procedure, include detecting implausible values of speeds, counts, or their combinations (Dahlgren, 2002). Coifman (2004) presents eight event-based validation tests that use microscopic data to assess the working status of single and dual loop detectors. The techniques range from comparing on/off time at each loop of a dual detector, to analyzing the succession of flow regimes observed at a particular detector. Other approaches rely on simple comparisons of the measurements retrieved by nearby detectors, or on a time-wise analysis of the observations of a single detector (Chen and May, 1987). Some alternative approaches to validate the performance of remote traffic microwave sensors (RTMS) are introduced in Coifman, 2006. Historical information regarding recurrent congestion patterns and typical traffic conditions can be also utilized to verify the likelihood of a specific measurement at a particular time and location, determining the need for further analysis.

Once malfunctioning detectors are identified, proper action needs to be taken in order to generate acceptable travel time predictions. The specific procedure will depend on the characteristics of the travel time prediction model under usage, data availability, and time constraints. One possible option involves imputing the missing measurements, either based on the values observed at nearby detectors, or by utilizing previous observations at the malfunctioning detector, provided that they are error free (for example, Al-Deek and Chandra, 2004). Additionally, average historical values may be utilized. Some of these techniques are explained in the technical report describing the data processing for California's PeMS (Urban Crossroads Inc. 2006), and in Nguyen and Scherer (2003). When imputation is not possible, a different type of action needs to be taken. Some models, such as the CTM-based one, can be run ignoring the malfunctioning detectors, with little effect on the prediction accuracy. For these models, each TMC should analyze the advantages in terms of model accuracy, data requirements, and computational efficiency, and of imputing data versus simply disregarding the detectors out of order at run time. Some travel time prediction models may not be able to run if error-free traffic measurements are not available for every detector. Under such scenario, traffic managers may choose to either temporarily display historic travel times for the affected sections, or not to display any value at all.

5.5 Summary

An overview of the data necessary to calibrate, validate, and implement the models proposed in this study has been provided. In this study both real and simulated travel time data has been used to validate and calibrate the models. The real travel time data is useful for testing the efficacy of the model against real world dynamics, whereas the simulated data is useful for testing the efficiency of the model under a number of scenarios. The simulated data was generated using a VISSIM microsimulator. Real travel time data were obtained from the Berkeley Highway network and from a freeway section in El Paso. The traffic counts in El Paso were obtained from the TransVISTA and a GPS system was used to record the actual travel times. The data obtained from the field may contain numerous errors due to detector malfunctions. Therefore, the data needs to be filtered and pre-processed before it is used for travel time prediction. Various recommendations on imputing missing or erroneous data have been provided.

Chapter 6. Models Calibration and Validation

This section describes the procedures utilized to calibrate and test the performance of the travel time prediction models developed for the present project. The first set of numerical studies presented herein is based on synthesized data corresponding to a 3-mile freeway segment located in El Paso, Texas, obtained as described in the previous chapter. The second type of numerical tests was conducted using field data. In both cases, the datasets were split, and some points (i.e., data corresponding to one day or one simulation run) were used to perform the model calibration, whereas the remaining ones were used for validation. It is important to note that during the experimental runs data was fed to the prediction models in an online interface replicating a real-time environment. By virtue of this, at each time interval the prediction software had access only to traffic data corresponding to previous intervals, and the observed performance is comparable to the expected results in an on-line setting.

The ***calibration*** process is focused on determining the appropriate values of the model parameters by selecting them in such a way that the model predictions match real travel time computations as closely as possible. Once such values are found, the model is applied to a different set of data, and the resulting differences between travel time predictions and real measurements are an indicator of the model performance (***model validation***). It is clear that the magnitude of the error is likely to vary among datasets. However, if the model is properly calibrated, and a meaningful error measurement is selected, such variations are not expected to be significant. In this work, travel time prediction errors were measured in terms of the Root Mean Square Error (RMSE), a fairly standard indicator, presented below. The remaining sections in this chapter present and discuss the numerical results obtained for different models and datasets.

6.1 Error Measurement

Given a set of real travel times, it is possible to compute a prediction error for each point in time for which travel time predictions are provided. Such errors are likely to vary across time and space, and a representative performance metric is necessary in order to assess model behavior and compare outcomes under different conditions. In this work, the travel time estimations produced by different model variations were evaluated based on their root mean squared error (RMSE):

$$RMSE(\%) = \sqrt{\frac{\sum_{i=1}^{N}\left[\frac{\left(t_i - \hat{t}_i\right)}{t_i}\right]^2}{N}}$$

where $\hat{t}_i$ and t_i are the real and predicted travel time at time i, respectively, and N is the number of points in time for which travel time forecasts are provided. This measure was computed separately for each origin destination pair for which predictions were generated. Additionally, model outcomes were compared in terms of the frequency distribution of the

absolute errors $\left(t_i - \hat{t}_i\right)$, and utilizing the value-based RMSE, computed similarly to the previous equation, but without dividing the absolute error by the actual travel time.

6.2 CTM-based Model: Validation and Calibration Using Simulated Data

In order to assess the performance of the model under different traffic conditions, detector location patterns, and data provision schemes, simulated data was utilized, obtained as described in Chapter 5.

In order to run the CTM-based travel time prediction model, a **cell-transmission representation** of the same highway segment modeled in VISSIM was created, consisting of 51 cells and 50 links. The user manual delivered along product P2 clearly describes how to construct such a model for an arbitrary highway segment. It also indicates how to prepare the required input files, which were built based on the parameter values presented in Tables 6.2 to 6.4 for this example. For tests conducted in this research, the CTM initial densities at every cell were set to zero, which is a sensible starting point if the simulation process can be deployed early in the morning. Otherwise, densities could be initialized to values based on historical data at the start time of the process. The sink cell was assigned an unbounded maximum density, implicitly assuming free flow conditions downstream of the analyzed segment. In any model implementation, this condition can be achieved by means of an appropriate selection of the length and location of the segment under study. **Travel times** were computed between the beginning of the first cell, and the beginning of each cell containing a detector. The highway segment between the starting point and detector ***d*** is denoted as ***Section d***.

The traffic counts predictions for every freeway entrance point were obtained from an ARIMA (3, 1, 2) model. The forecasts, provided at a 1-minute aggregation level, were uniformly distributed across the corresponding 4-second simulation intervals. The forecasting horizon was set equal to the sum of the prediction horizon and the section travel time under extremely high congestion. This guaranteed an appropriate simulation length, according to what was discussed in Chapter 3.

For the purpose of this application, the core CTM parameters were calibrated using a simple trial-and error procedure. A set of starting values (chosen based on Muñoz et al., 2004) was adjusted to minimize the deviations between CTM-based and real travel times (as provided by VISSIM). The use of field detector data, usually more limited than the one provided by a simulator, may demand more complex calibration procedures. Munoz et al., 2004 (17) described a CTM calibration methodology that produced satisfactory results.

For the initial model calibration, a triangle-shaped traffic ***demand*** was introduced at the beginning of highway segment modeled in VISSIM, and at each of the corresponding entry ramps. The demand curves spanned 2 hours, and represented conditions ranging from light traffic to congestion, including congestion build-up and dissipation stages. Figure 6.1 displays such demands for the main lane, and Table 6.1 shows the corresponding values for the entry ramps. The corresponding travel times ranged from 3 to 7 minutes.

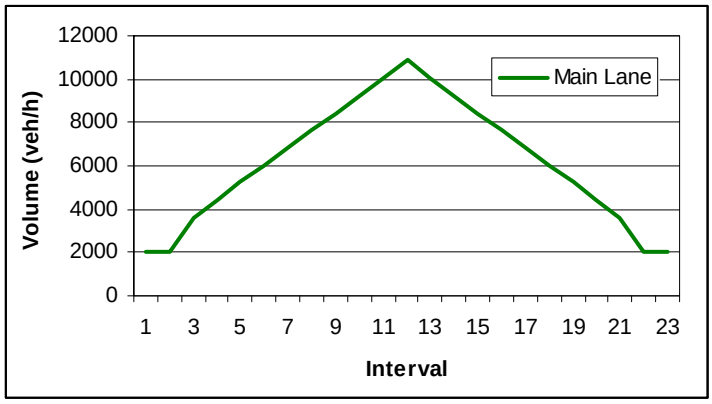

Figure 6.1: Demand distribution for the main lane input (Calibration)

Table 6.1: Demand distribution at entry ramps (Calibration)

Ramp	Min	Max
1	267.2	566
2	273.4	647
3	268.3	580
4	250	360
5	270	610
6	269.7	536

The simulation output files, consisting of detector measurements and actual travel times, were used to run the prediction models and compute the errors, respectively. Output files were also used to estimate ***split ratios*** at exit ramps. Even though the exact values can be extracted from the simulator, an estimation process was implemented in order to better replicate real implementation conditions (settings are displayed in Table 6.2). Four different split ratios were specified at each exit ramp, corresponding to different times of the day (Table 6.3). The estimation involved computing the cumulative counts every 30 minutes at two detector stations: the last detector before the exit ramp, and the first detector after the same (alternatively, we could have used a detector on the ramp, if available). The ratio was estimated as the percent of total flow using the ramp, and further adjusted during calibration. In a field setting, detector data may not be available to perform such estimation, and coarser approximations, based on experience and engineering judgment may be necessary. The later is also necessary to define the number and duration of time periods for which different split ratios are specified, given that those depend on the origin-destination patterns of each specific highway segment.

Table 6.4 shows the ***priority coefficients*** utilized at the merge cells, which were initially assumed to be 0.9, and later adjusted during calibration. Table 6.5 displays other important model assumptions.

Table 6.2: CTM-based model settings

Maximum travel time	1000 sec
Prediction interval	3 min
Sample size for arima	9 min
Aggregation for ARIMA	1 min
Initialization time for CTM	9 min
Simulation interval	4 sec
Free Flow Speed	50mph

Table 6.3: Split ratios at exit ramps

	Period 1	**Period 2**	**Period 3**	**Period 4**
Exit 1	0.92	0.93	94	0.96
Exit 2	0.92	0.92	0.92	0.9
Exit 3	0.92	0.9	0.9	0.9
Exit 4	0.94	0.94	0.92	0.92
Exit 5	0.92	0.92	0.92	0.92
Exit 6	0.92	0.92	0.92	0.92

Table 6.4: Priority coefficients at entry ramps (priority of the mainlane flow)

	Period 1	**Period 2**	**Period 3**	**Period 4**
Ramp 1	0.9	0.9	0.9	0.89
Ramp 2	0.9	0.9	0.9	0.9
Ramp 3	0.88	0.88	0.89	0.89
Ramp 4	0.88	0.88	0.88	0.88
Ramp 5	0.89	0.89	0.89	0.89
Ramp 6	0.9	0.9	0.9	0.89

Table 6.5: Cell properties and calibrated parameters

Cell #	Length (ft)	Nmax veh/ft/ln	Qmax veh/sec/ln	*w* ft/sec	Type	Lanes
0	400	100000	100000	20	0	4
1	400	100000	100000	20	0	4
2	485.8	0.162	1.9	17	0	4
3	485.8	0.162	1.9	17	0	4
4	485.8	0.162	1.9	17	0	4
5	485.8	0.162	1.9	17	2	4
6	498.8	0.162	1.9	17	0	4
7	525.16	0.162	1.9	17	0	4
8	525.16	0.162	1.9	17	1	4
9	525.16	0.162	1.9	17	0	4
10	652.9	0.13	1.9	17	0	4
11	652.9	0.13	1.9	17	1	4
12	515.15	0.13	1.9	17	0	4
13	515.15	0.13	1.9	17	2	4
14	515.15	0.13	1.9	17	0	4
15	515.15	0.13	1.9	17	2	4
16	515.15	0.13	1.9	17	0	4
17	515.15	0.13	1.9	17	0	4
18	698.9	0.13	1.9	17	1	4
19	538.3	0.13	1.9	17	1	4
20	538.3	0.13	1.9	17	2	4
21	687.15	0.16	1.9	17	0	4
22	687.15	0.16	1.9	17	0	4
23	664.75	0.16	1.9	17	1	4
24	664.75	0.17	2.5	17	0	5
25	502.433	0.17	2.5	17	2	5
26	502.433	0.17	2.5	17	0	5
27	502.433	0.17	2.5	17	0	5
28	525.85	0.17	2.5	17	0	5
29	525.85	0.17	2.5	17	1	5
30	494.42	0.17	2.5	17	2	5
31	494.42	0.17	2.5	17	0	5
32	494.42	0.17	2.5	17	0	5
33	494.42	0.17	2.5	17	0	5
34	494.42	0.17	2.5	17	0	5
35-53	400	100000	100000	17.2	0	5

Numerical experiments were used to determine the more appropriate **prediction step**. It was found that, even though the model performance remains relatively unchanged for values

ranging from 1 to 5 minutes, the later value led to more stable results (Table 6.6). In view of this, and considering that 5 minutes is a reasonable prediction interval for a real-time implementation (travel time forecasts are not likely to be updated at intervals smaller than 5 minutes), the validation experiments were conducted utilizing that aggregation level.

Table 6.6: Model performance for various prediction steps (RMSE), utilizing the calibration data

Prediction Step	Global	1	2	3	4	5	6	7	8	9	10	11	12
1	14	25%	13%	15%	14%	14%	11%	11%	12%	12%	10%	11%	12%
3	15	31%	17%	15%	12%	13%	13%	12%	11%	11%	10%	11%	11%
5	**13**	**30%**	**15%**	**12%**	**13%**	**12%**	**12%**	**10%**	**9%**	**9%**	**7%**	**8%**	**9%**
10	20	24%	6%	19%	20%	7%	36%	28%	24%	14%	11%	9%	7%

Calibration was achieved in two steps. In a first stage, the model was run in calibration mode, and free flow speed, capacities, split ratios, and priority coefficients were adjusted in order to minimize the difference between the cumulative counts predicted by the model at each detector, and the actual values provided by the simulator. Further refinements into these parameters, along with modifications to **Q**max, **N**max, and ***w***, were introduced on a second stage, based on travel time predictions errors. Table 6.5 shows cell characteristics along with the calibrated values for the aforementioned parameters. ***It is important to notice that these values were achieved assuming that seven detectors (1, 6, 13) were deployed and utilized for travel time prediction.*** After some numerical experiments, this reduced set of sensors was found to lead to more accurate results than the use of the existing 13 detectors. This is a consequence of the utilization of cumulative counts as the basis for the travel time computation in this particular model. When detectors are located close to each other, relatively small errors in the distribution of densities throughout the segment can lead to important differences between predicted and real travel times. The methodologies presented in Chapter 7 may be utilized to evaluate different detector deployment pattern strategies. ***If different sets of detectors than those utilized for calibration were used, modifications would be necessary in order to achieve desired levels of accuracy.*** TMCs may calibrate their models for a variety of detector availability scenarios, in order to be able to rapidly accommodate changes in the model due to detector malfunctioning.

Another important finding during the calibration process was that utilizing only cumulative traffic counts for model calibration may not produce desirable levels of accuracy. Parameters leading to relatively small errors in terms of volumes (RMSE below 5 percent) were found to result in considerable travel time prediction errors. This is a consequence of two combined factors: the methodology used to compute travel times, and the fairly large value of cumulative counts (in the order on hundreds and even thousands). In effect, an error of 1 percent in volumes may actually involve a difference of 100 or more vehicles between predicted and actual traffic counts at a sensor, which may translate into significantly different travel time estimations. It is therefore recommended to always collect real travel time data in order to refine the calibration of fundamental model parameters.

Tables 6.7 and 6.8 present the model performance after calibration. The error measurements were computed utilizing the formulas presented in the previous section, and discarding the values corresponding to the first 10 time intervals, which represent approximately

the time taken by a vehicle to traverse the segment under uncongested conditions. The error values range from 10 percent to 23 percent (15 percent in average), and average absolute value of such errors is approximately 32 seconds. Moreover, 41 percent of the total travel time prediction errors are below 1 minute.

Table 6.7: Model calibration and validation results (RMSE by section). Results obtained using a 5-minute prediction step for profiles 1-5, and a 3 minute prediction step for calibration

	Section RMSE (%)											
Demand	1	2	3	4	5	6	7	8	9	10	11	12
Calibration	**23%**	**21%**	**13%**	**10%**	**12%**	**19%**	**17%**	**16%**	**14%**	**12%**	**13%**	**13%**
Profile 1	31%	17%	15%	12%	13%	13%	12%	11%	11%	10%	11%	11%
Profile 2	23%	16%	15%	13%	12%	15%	15%	14%	13%	12%	11%	10%
Profile 3	37%	22%	15%	12%	12%	10%	9%	9%	8%	7%	7%	7%
Profile 4	46%	23%	22%	18%	14%	19%	17%	15%	14%	13%	12%	13%

Table 6.8: Summary of model calibration and validation results

Demand	Global RMSE (Val)	Global RMSE %	Error < 60s
Calibration	32	16	41%
Profile 1	31	15	34%
Profile 2	23	14	37%
Profile 3	24	15	33%
Profile 4	35	21	38%

It is interesting to observe that the model fit is fairly consistent across detectors, and relatively independent of the length of the segment. Furthermore, the model accurately replicates travel times throughout the simulated period, capturing congestion build-up and dissipation. In order to validate the model, similar tests were performing utilizing traffic counts generated from different demand profiles, representing various traffic conditions. The results for such experiments are summarized in Table 6.8. The first demand profile corresponds to the same conditions assumed during calibration, but was generated using a different random seed number, therefore simulating a different day. Figures 6.2 and 6.3 contrast forecasted and real travel times under this scenario, at a 1-minute aggregation level, from the first detector to the third and eleventh one, respectively. As expected, the model fit is very similar to the one observed during calibration.

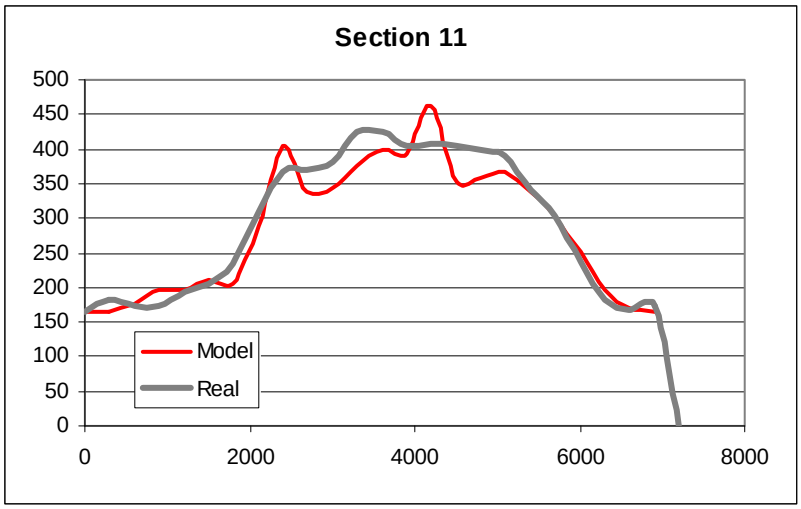

Figure 6.2: Example of model fit for a 2.5-mile section (demand profile 1, 5-minute prediction step)

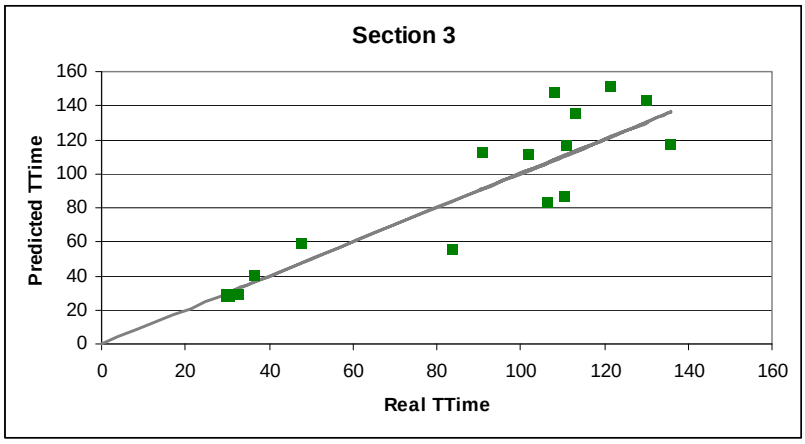

Figure 6.3: Example of model fit for a 0.5-mile section (demand profile 1, 5-minute prediction step)

The ***second demand*** profile has the same triangular shape assumed earlier, but the values are 30 percent lower than on the initial case. The ***third profile*** is given by Figure 6.4 and Table 6.9. It is trapezoidal, representing a longer period of time with high traffic demands. Again, the model accuracy remains fairly constant, indicating that the parameters are appropriate, and the model is robust under varying traffic conditions.

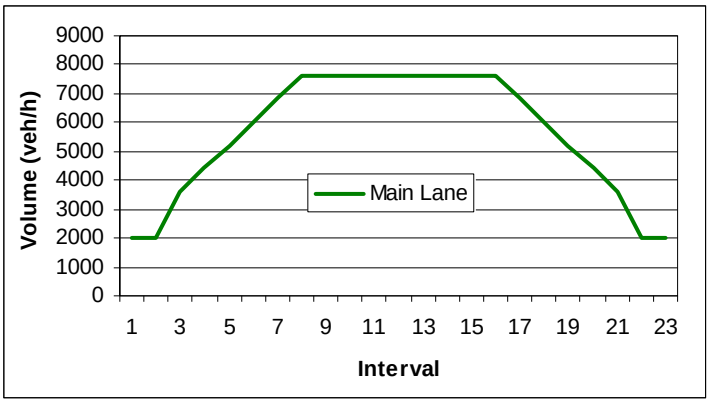

Figure 6.4: Demand profile 3: Trapezoidal demand

Table 6.9: Demand distribution at entry ramps (Profile 3)

Ramp	Min	Max
1	267.2	566
2	273.4	647
3	268.3	580
4	250	360
5	270	610
6	269.7	536

A second experiment explored the effects of changes in **OD demand patterns,** by utilizing demand profile 1 for all ramps except 2 and 5 (Figure 6.5), for which the corresponding volumes were increased by 30 percent. In this case, the model accuracy diminished by a 5 percent (on average). Nevertheless, the results are still around 20 percent for most sections, and the model captures the travel time variations fairly accurately.

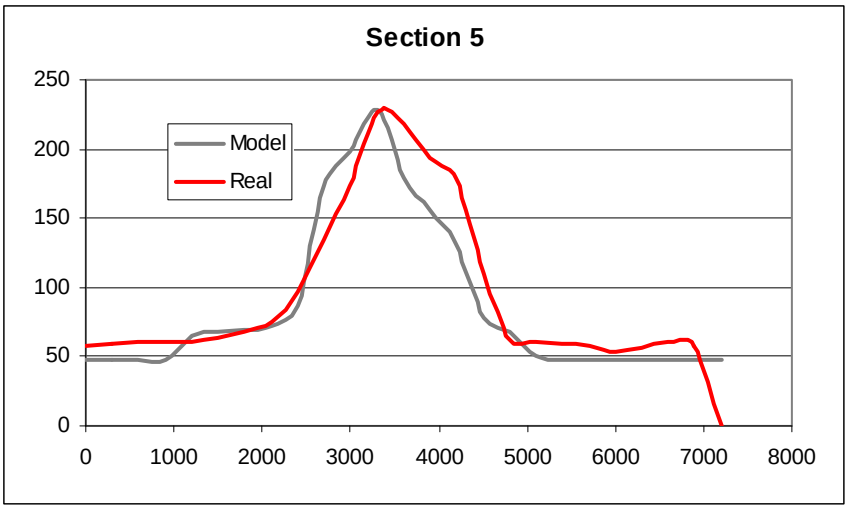

Figure 6.5: Example of model fit for a 0.75-mile section (demand Profile 3, 5 minute prediction step)

6.3 CTM-based Model: Validation and Calibration Using Field Data

Field data, described in Chapter 5, was used to test the model performance and to verify the feasibility of performing real-time travel time predictions. In order to do this, a cell representation of the segment for which real travels time measurements were available was constructed (Figure 6.6, and Table 6.11). The representation includes the actual location of 22 detectors, 4 entry ramps, and 2 exit ramps, for the west bound direction. However, preliminary data analyses revealed that some detectors were clearly out of order on the days selected for calibration and validation (Table 6.10), and only those working throughout the analysis period were included in the final model runs.

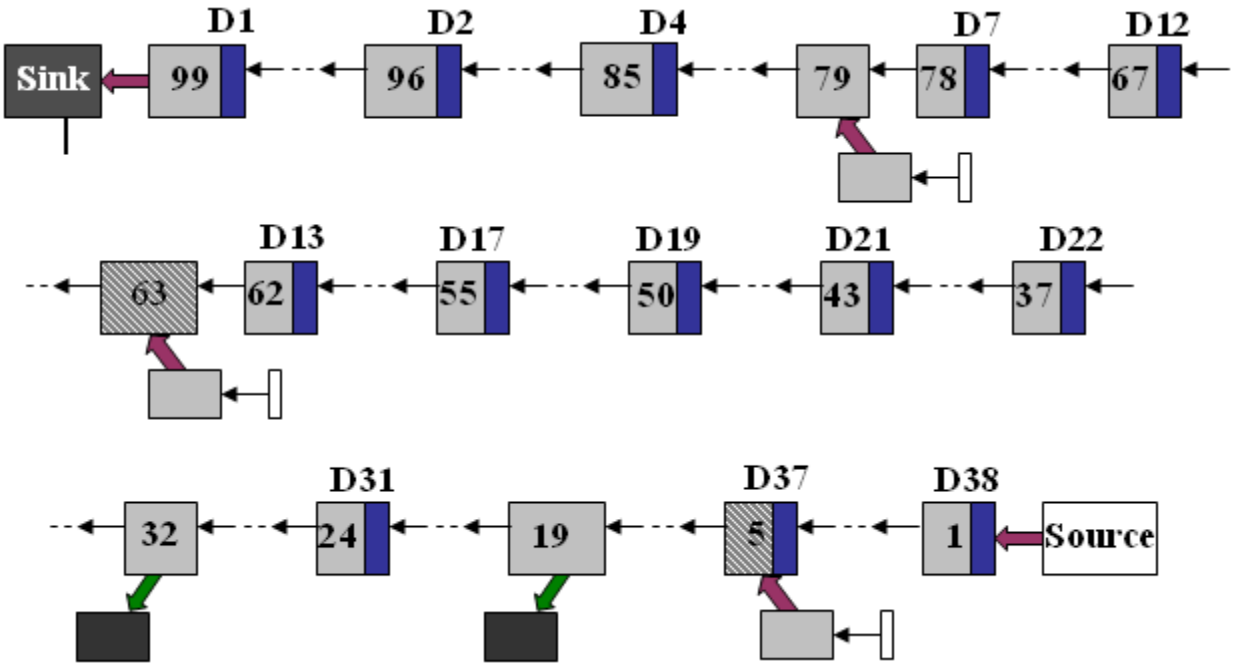

Figure 6.6: Cell transmission representation of the analyzed segment

Table 6.10: Map of detector data availability for day on which GPS data was collected

Day	Missing Detector Data from 6 to 8AM
June 4	3 11 26 28 36
June 5	3 11 26 28 36
June 6	3 11 26 28 36
June 7	All
June 15	3 11 26 28 31 33 35 36 38
June 18	3 11 26 28 31 35 36 38
June 19	3 11 26 28 31 35 36 38
June 22	3 4 11 26 28 31 35 36 38
June 25	3 4 11 26 28 31 35 36 38
June 26	3 4 11 26 28 31 35 36 38
June 28	All
June 29	All

The travel time prediction model is still able to generate forecasts under when **detectors are missing**, provided that there is data for the first detector in the system, and that malfunctioning detectors are identified and disregarded for travel time prediction purposes. The condition of each detector was manually verified for each of the numerical experiments conducted for this project. However, automatic data filtering procedures should be utilized

during an online implementation. It is important to notice that when the missing data correspond to entry ramps, the model performance may be seriously affected. This is particularly true when there are no working detectors relatively close to the entry point (and before any other entry/exit section), able to capture the changes in total main lane volume. Under such circumstances, it becomes critical to impute missing data.

Table 6.11: Cell properties and calibrated parameters for calibration and validation using field data

Cell #	Length (ft)	Nmax veh/ft/ln	Qmax veh/sec/ln	*w* ft/sec	Type	Lanes
1	500	500	1000	1000	17	0
2	610.104	500	1000	1000	17	0
3	610.104	610.104	0.1	1	17	0
4	528	610.104	0.1	1	17	0
5	528	528	0.1	1	17	0
6	600.072	528	0.1	1	17	0
7	600.072	600.072	0.1	1	17	0
8	528	600.072	0.1	1	17	0
9	528	528	0.1	1	17	0
10	528	528	0.1	1	17	0
11	686.928	528	0.1	1	17	0
12	686.928	686.928	0.1	1	17	0
13	528	686.928	0.1	1	17	0
14	528	528	0.1	1	17	0
15	528	528	0.1	1	17	0
16	555.456	528	0.1	1	17	0
17	555.456	555.456	0.1	1	17	0
18	555.456	555.456	0.1	1	17	0
19	555.456	555.456	0.1	1	17	0
20	529.5312	555.456	0.1	1	17	2
21	529.5312	529.5312	0.1	1	17	0
22	529.5312	529.5312	0.1	1	17	0
23	529.5312	529.5312	0.1	1	17	0
24	715.7568	529.5312	0.1	1	17	0
25	715.7568	715.7568	0.1	1	17	0
26	528	715.7568	0.1	1	17	0
27	528	528	0.1	1	17	0
28	709.2624	528	0.1	1	17	0

Cell #	Length (ft)	Nmax veh/ft/ln	Qmax veh/sec/ln	*w* ft/sec	Type	Lanes
29	709.2624	709.2624	0.1	1	17	0
30	528	709.2624	0.1	1	17	0
31	528	528	0.1	1	17	0
32	528	528	0.1	1	17	0
33	579.4272	528	0.1	1	17	2
34	579.4272	579.4272	0.1	1	17	0
35	579.4272	579.4272	0.1	1	17	0
36	579.4272	579.4272	0.1	1	17	0
37	689.7792	579.4272	0.1	1	17	0
38	689.7792	689.7792	0.1	1	17	0
39	528	689.7792	0.1	1	17	0
40	528	528	0.1	1	17	0
41	528	528	0.1	1	17	0
42	528	528	0.1	1	17	0
43	543.4915	528	0.1	1	17	0
44	543.4915	543.4915	0.1	1	17	0
45	543.4915	543.4915	0.1	1	17	0
46	543.4915	543.4915	0.1	1	17	0
47	543.4915	543.4915	0.1	1	17	0
48	543.4915	543.4915	0.1	1	17	0
49	543.4915	543.4915	0.1	1	17	0
50	556.7654	543.4915	0.1	1	17	0
51	556.7654	556.7654	0.1	1	17	0
52	556.7654	556.7654	0.1	1	17	0
53	556.7654	556.7654	0.1	1	17	0
54	556.7654	556.7654	0.1	1	17	0
55	697.356	556.7654	0.1	1	17	0
56	697.356	697.356	0.1	1	17	0
57	528	697.356	0.1	1	17	0
58	528	528	0.1	1	17	0
59	542.1504	528	0.1	1	17	0
60	542.1504	542.1504	0.1	1	17	0
61	542.1504	542.1504	0.1	1	17	0
62	542.1504	542.1504	0.1	1	17	0
63	784.344	542.1504	0.1	1	17	0

Cell #	Length (ft)	Nmax veh/ft/ln	Qmax veh/sec/ln	*w* ft/sec	Type	Lanes
64	771.8304	784.344	0.1	1	17	1
65	528	771.8304	0.1	1	17	0
66	529.2514	528	0.1	1	17	0
67	786.984	529.2514	0.1	1	17	0
68	786.984	786.984	0.1	1	17	0
69	528	786.984	0.1	1	17	0
70	528	528	0.1	1	17	0
71	598.8576	528	0.1	1	17	0
72	598.8576	598.8576	0.2	2	17	0
73	598.8576	598.8576	0.2	2	17	0
74	598.8576	598.8576	0.2	2	17	0
75	598.8576	598.8576	0.2	2	17	0
76	562.1405	598.8576	0.2	2	17	0
77	562.1405	562.1405	0.1	1	17	0
78	562.1405	562.1405	0.1	1	17	0
79	568.3445	562.1405	0.1	1	17	0
80	568.3445	568.3445	0.15	1.5	17	1
81	568.3445	568.3445	0.15	1.5	17	0
82	568.3445	568.3445	0.15	1.5	17	0
83	568.3445	568.3445	0.15	1.5	17	0
84	568.3445	568.3445	0.15	1.5	17	0
85	748.968	568.3445	0.15	1.5	17	0
86	748.968	748.968	0.15	1.5	17	0
87	681.0672	748.968	0.15	1.5	17	0
88	681.0672	681.0672	0.1	1	17	0
89	528	681.0672	0.1	1	17	0
90	528	528	0.1	1	17	0
91	528	528	0.1	1	17	0
92	528	528	0.1	1	17	0
93	528	528	0.1	1	17	0
94	528	528	0.1	1	17	0
95	528	528	0.1	1	17	0
96	735.9264	528	0.1	1	17	0
97	528	735.9264	0.1	1	17	0
98	528	528	0.1	1	17	0

Cell #	Length (ft)	Nmax veh/ft/ln	Qmax veh/sec/ln	*w* ft/sec	Type	Lanes
99	528	528	0.1	1	17	0
100	528	528	0.1	1	17	0
101-106	500	528	0.1	1	17	0

For ***calibration*** purposes, it was taken into account that the highway segment for which the model parameters were found via simulation in the previous section overlaps with the section used for field data collection. Consequently, the value of some fundamental parameters, such as free flow speed, ***w***, **Q**max, and **N**max were assumed to be similar for both cases. Small refinements were performed based on cumulative counts at selected detectors, given the reduced availability of real travel time measurements. Figures 6.7 and 6.8 exemplify the final model fit after calibration, based on cumulative counts comparisons. Errors (%RMSE) ranged from 4 percent to 8 percent for most detectors.

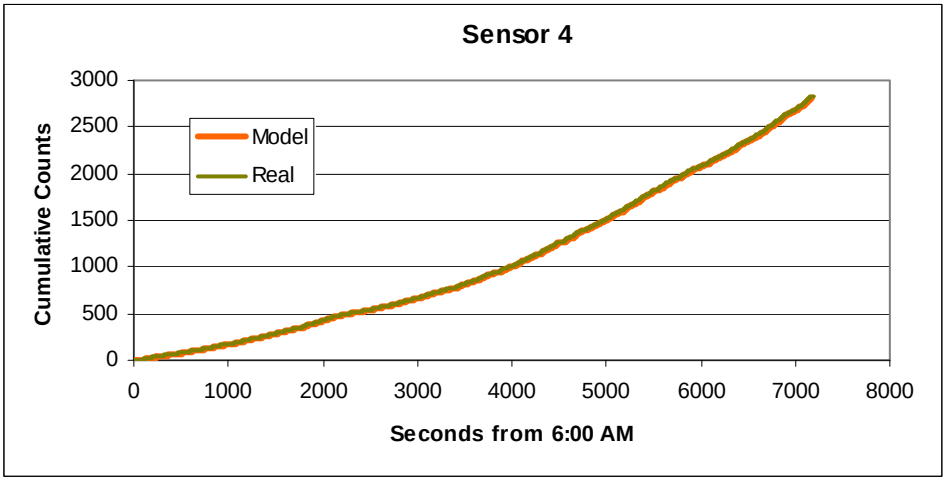

Figure 6.7: Predicted and observed cumulative counts at sensor 4 (2.5 miles from the origin)

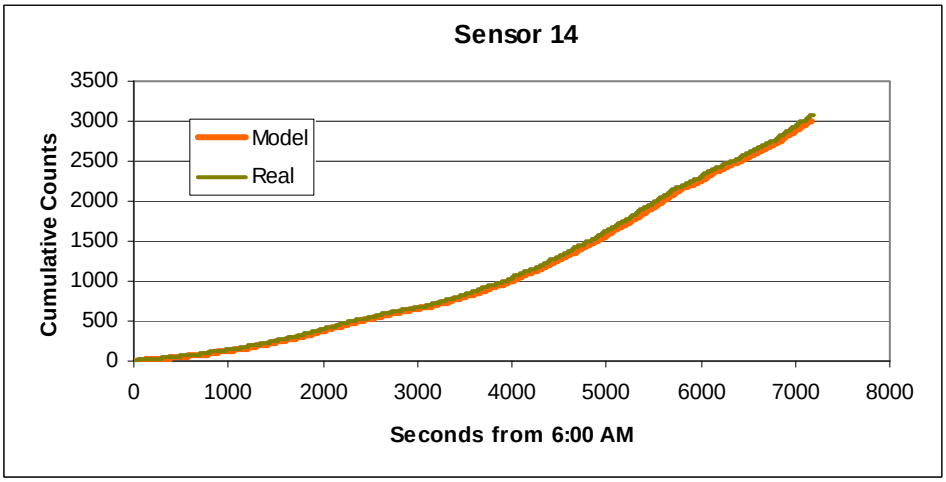

Figure 6.8: Predicted and observed cumulative counts at sensor 14 (9.3 miles from the origin)

This suggests an ***alternative procedure to perform a preliminary model calibration*** when real travel time measurements are not available: one may develop a microsimulation model of the entire section under study, and run it utilizing the measured traffic volumes as demands, outputting travel time measurements between desired points as needed.

Traffic demands were obtained from the database described in section 5.3 and processed using a C++ code to generate the appropriate input files. This included disaggregating the provided measurements into 4-second intervals, adding the counts reported at different detection zones in the same transversal section, and mapping field detector numbers to model detector and ramp numbers.

Split ratios at exit ramps were estimated based on the counts corresponding to 10 week days, which were not used for calibration or validation purposes (Table 6.12). For such days, the percent of traffic taking each exit was computed as the ratio between the cumulative counts at the exit ramp and the total flows at the upstream main lane detectors. Cumulative counts were computed starting at 6:00 AM on each day, and the split ratios were calculated every 30 minute interval. Due to data errors, it was not possible to estimate split ratios for ramps 1 and 3, which were assumed to be the same as for ramp 2. **Priority coefficients** at entry ramps were set to 0.9, based on the simulation experience. Table 6.13 shows the assumptions regarding the remaining model parameters.

Table 6.12: Split ratio computation for exit ramp 2

	Percentage of traffic remaining on main lane			
	6:00-6:30	**6:30-7:00**	**7:00-7:30**	**7:30-8:00**
28-Jun	0.95	0.93	0.93	0.92
29-Jun	0.95	0.93	0.91	0.91
1-Jul	0.96	0.92	0.94	0.90
3-Jul	0.98	0.92	0.91	0.91
5-Jul	0.93	0.93	0.95	0.91
6-Jul	0.95	0.91	0.93	0.90
9-Jul	0.93	0.91	0.92	0.92
10-Jul	0.96	0.91	0.93	0.91
11-Jul	0.96	0.93	0.93	0.91
12-Jul	0.94	0.93	0.92	0.91
Average	**0.95**	**0.92**	**0.93**	**0.91**

The calibration results, based on traffic data corresponding to the morning peak hour (6:00 to 8:00 AM on June 4 are summarized in Table 6.14). Based on preliminary data analyses, detectors 2, 7, and 12 were selected to contrast the variations in traffic volumes to the corresponding changes in predicted travel times, as a complement to the validation based on RMSE. The above-mentioned detectors capture representative portions of the traffic throughout the segment. The travel time predictions depicted in Figures 6.9 to 6.13 are consistent with the observed changes in demands (Figure 6.7), presenting fairly stable travel times for a relatively low demand, and a slight upward trend at the end of the period, consisting with larger traffic

volumes. This suggests that the model is able to capture changes in traffic volumes throughout the period.

Table 6.13: CTM-based model settings

Maximum travel time	1000 sec
Prediction interval	3 min
Sample size for arima	15 min
Aggregation for ARIMA	1 min
Initialization time for CTM	6 min
Simulation interval	4 sec
Free Flow Speed	**50mph**

Figure 6.9: Transversal plot of traffic counts from 7:55 AM to 8:00 AM

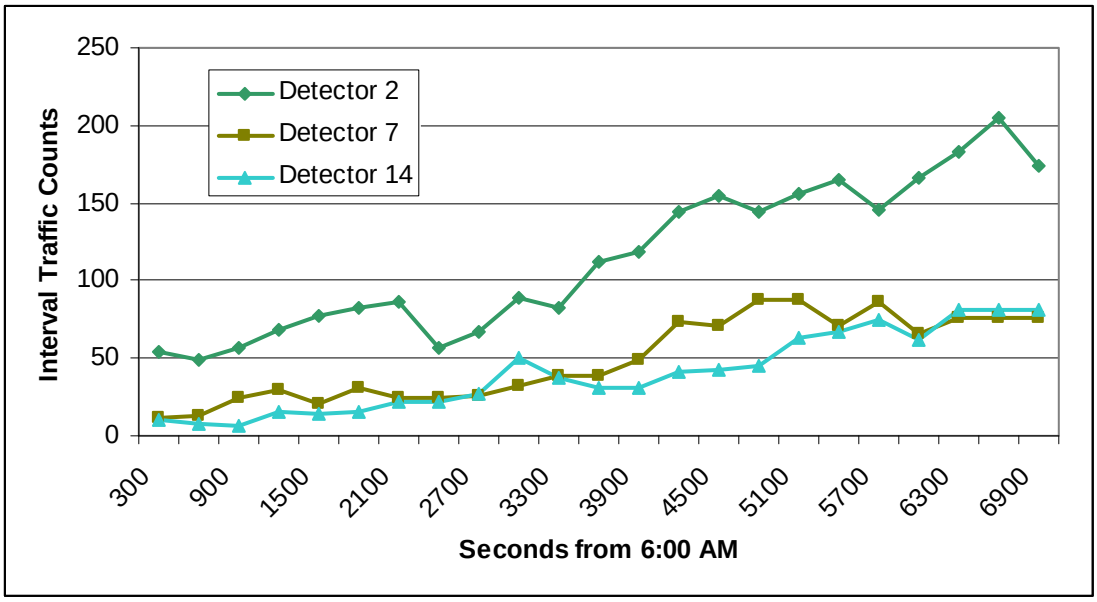

Figure 6.10: Interval traffic counts on June 4 at selected detectors

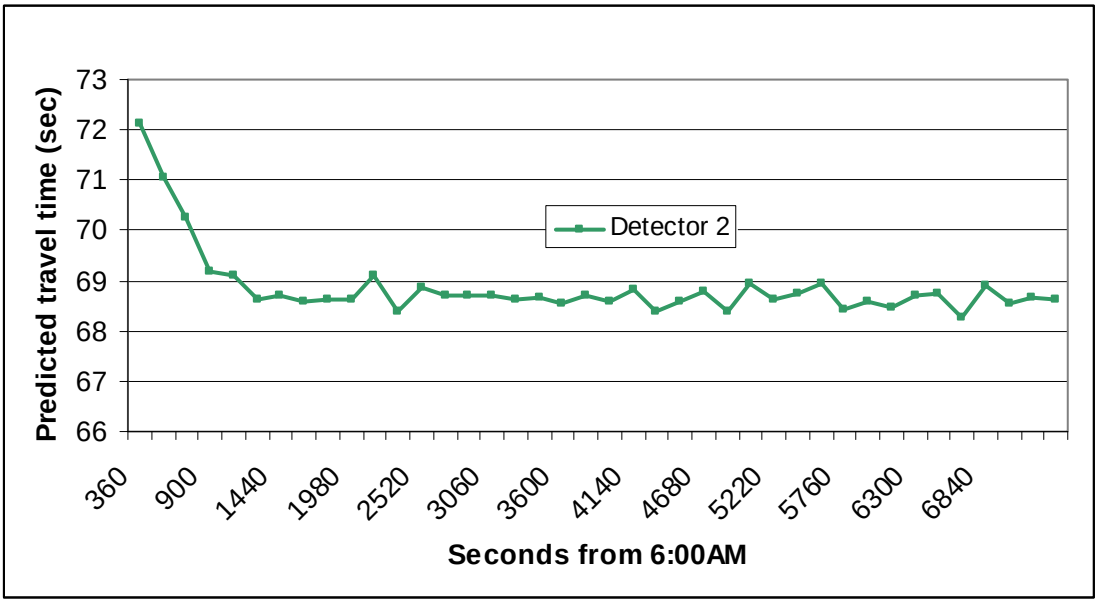

Figure 6.11: Travel time predictions on Section 2 (June 4).

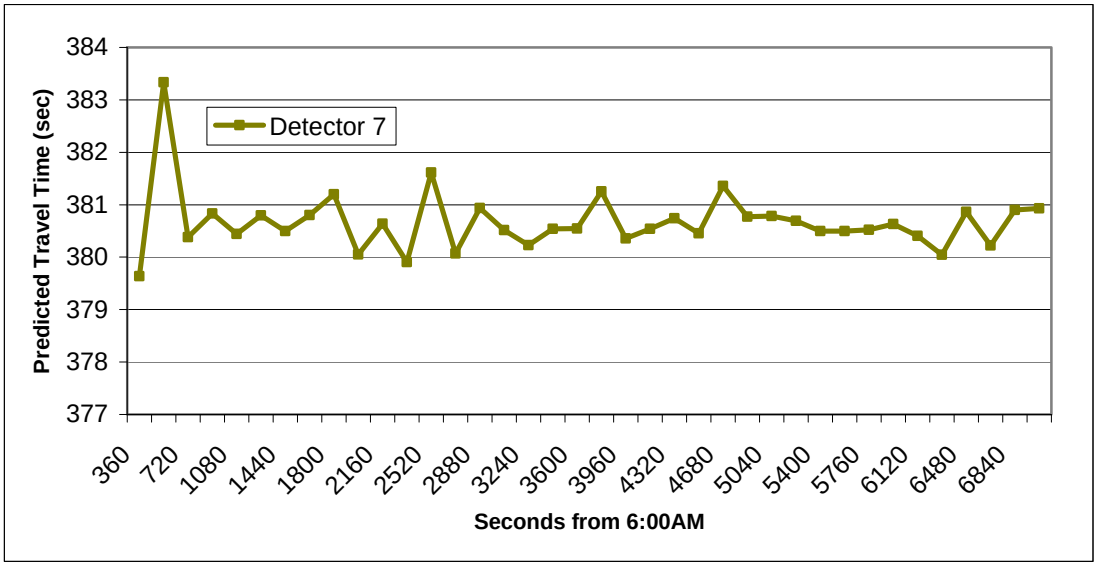

Figure 6.12: Travel time predictions on Section 7 (June 4).

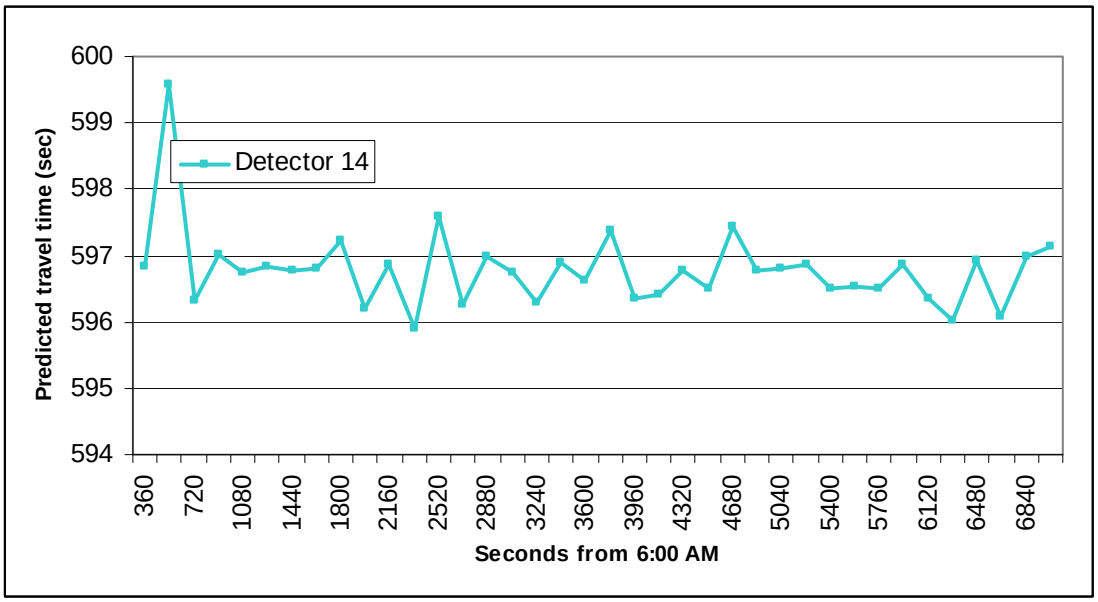

Figure 6.13: Travel time predictions on Section 14 (June 4).

Utilizing the newly adjusted parameters, the model was run in a similar way as described in section 5, and the real travel time measurements obtained via GPS were compared to model the corresponding model forecasts. The results, displayed in Table 6.14, show a very **satisfactory model performance, with average prediction errors of 1 minute for a ~10 mile highway section.** Additionally, notice that the errors remain fairly constant even when some detectors are out of order.

Table 6.14: Validation using field data: Prediction Errors

		Prediction Error per Section (sec)														RMSE	RMSE
	Trip	1	2	3	4	5	6	7	8	9	10	11	12	13	14	%	sec
4-Jun	1	3	5	39	43	55	68	71	66	51	64	64	44	75	114	22	65
	2	6	12	38	50	63	59	63	56	53	70	73	51	82	130		
	3	6	21	41	52	66	87	59	65	55	73	73	61	93	139		
5-Jun	1	7	10	33	37	50	53	44	53	29	48	55	25	59	98	22	58
	2	2	22	38	45	69	71	67	61	32	51	55	31	64	118		
	3	3	22	46	46	69	77	54	66	48	67	65	56	81	134		
6-Jun	1	5	4	38	45	68	82	64	61	41	58	61	43	75	115	15.5	48
	2	1	11	33	50	63	54	48	51	34	53	54	38	69	125		
	3																
15-Jun	1					91	96	88	77	62	82	72	58	90	131	18	63
Missing	2					99	110	109	89	83	103	107	96	120	157		
1,3,4,5	3					85	96	96	97	84	101	102	83	114	164		
18-Jun	1			43		68	61	67	53	50	68	68	49	83	124	17	61
Missing	2			49		73	83	74	81	74	86	87	69	104	164		
1,3,5	3			50		76	84	88	74	71	91	94	75	110	152		
19-Jun	1			32		61	71	64	56	53	61	60	48	83	129	14.5	54
Missing	2			31		63	61	59	44	45	68	70	50	81	124		
1,3,5	3			35		57	68	54	58	52	68	69	55	88	132		
22-Jun	1			35		56	67	53	58	51	68	68		174	135	22	83
Missing	2			33		54	60	54	62	39	60	59		167	131		
1,3,5,14	3			30		61	75	58	51	50	67	70		171	132		
25-Jun	1			31		59	59	57	61	46	66	66		171	140	23	87
Missing	2			29		65	62	51	59	49	68	68		176	132		
1,3,5,14	3			32		67	68	64	70	48	68	80		179	147		
26-Jun	1			32		67	21	29	43	30	27	49		40	8	11	27
Missing	2																
1,3,5,14	3																

6.4 N-Curve Model Calibration and Validation Based on Simulated Data

A 12-mile segment of both the east and westbound direction of El Paso's Border Highway were modeled utilizing DYNASMART-P (refer to Figure 6.14). The simulation network includes five on-ramps and six off-ramps in the eastbound direction whereas in the westbound direction it includes six on-ramps and four off-ramps. In total the network consists of 117 nodes and 115 links. The simulation duration was a total of 120 minutes, while vehicle demand loading took place during the first 60 minutes to allow for departure from origin and arrival at destination of all vehicles. Simulations were run utilizing different levels of uniform generated trip demands, varying from 8,000 vehicles (free-flow conditions) to 18,000 vehicles (congested conditions). The output files consist of traffic counts and the corresponding travel times computed for time intervals of 1 minute. Python codes were written to process data according to individual detectors and determine travel time measurements according to the simulation and N-Curve method. Results are shown in Tables 6.15–6.18 and Figures 6.15–6.18.

Figure 6.14: Aerial View and DYNASMART-P Network of El Paso's Border Highway

Travel time values obtained through the use of simulation are presented in the following table. As observed, low-to-medium vehicle demand loadings (approximately 8,000 to 13,000 vehicles) experience travel times similar to those experienced under free flow conditions. However, the travel times with heavy demand loading (greater than 13,000 vehicles) reflect higher travel times, due to congestion and queue build up.

Table 6.15: Simulated (Actual) Travel Time: Eastbound and Westbound Freeway Segment

Eastbound						
	Departure Time					
Generated Vehicles	***10 min.***	***20 min.***	***30 min***	***40 min***	***50 min***	***60 min***
7830	13.20	13.24	13.24	13.22	13.13	13.00
10392	14.05	13.80	14.00	14.02	13.84	13.00
13132	19.43	25.19	30.28	34.11	36.67	40.97
15685	24.83	31.36	37.13	42.41	48.58	51.58
18244	31.04	39.79	47.62	58.65	66.69	72.12
Westbound						
	Departure Time					
Generated Vehicles	***10 min.***	***20 min.***	***30 min***	***40 min***	***50 min***	***60 min***
7830	13.23	13.26	13.35	13.23	13.26	13.22
10392	13.66	13.60	13.53	13.72	13.59	13.17
13132	14.57	14.80	14.58	14.84	14.63	13.21
15685	16.05	17.20	20.11	24.37	24.09	23.58
18244	19.25	27.74	34.93	47.19	51.39	48.19

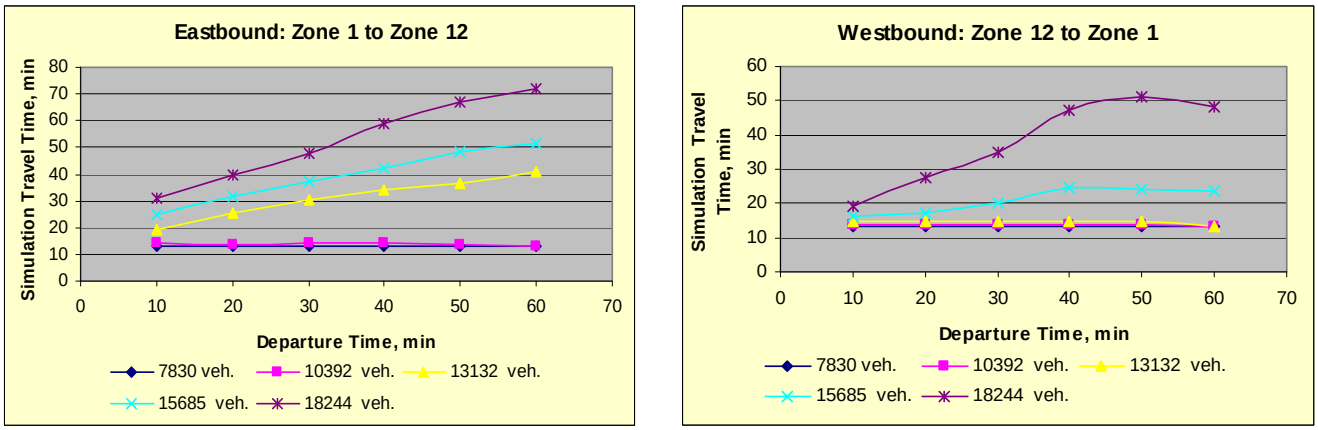

Figure 6.15: Simulated Data: Eastbound and Westbound Freeway Segment

The estimated travel time values obtained through the use of the *N*-Curve method are presented in the following table. Similarly, to results obtained from the simulation, the low-to-medium demand loading experience travel times similar to free flow conditions. In addition, travel times with heavy demand loading also experience higher travel times, due to same reasons stated above.

Table 6.16: Estimated Travel Time: Eastbound and Westbound Freeway Segment

Eastbound						
	Departure Time					
Generated Vehicles	***10 min.***	***20 min.***	***30 min***	***40 min***	***50 min***	***60 min***
7830	13.905	15.741	13.810	13.829	13.829	13.796
10392	13.462	14.723	14.425	14.597	14.597	14.807
13132	15.336	23.223	30.318	33.932	33.932	37.580
15685	18.664	32.667	38.735	43.286	43.286	49.298
18244	21.970	26.189	46.322	51.735	51.735	66.028
Westbound						
	Departure Time					
Generated Vehicles	***10 min.***	***20 min.***	***30 min***	***40 min***	***50 min***	***60 min***
7830	13.496	13.412	13.450	13.498	13.510	13.660
10392	13.565	13.682	13.932	13.632	13.717	13.797
13132	13.979	14.971	14.924	14.824	15.252	14.766
15685	14.522	15.351	16.494	20.855	29.309	32.077
18244	15.367	18.112	31.657	44.231	51.021	62.090

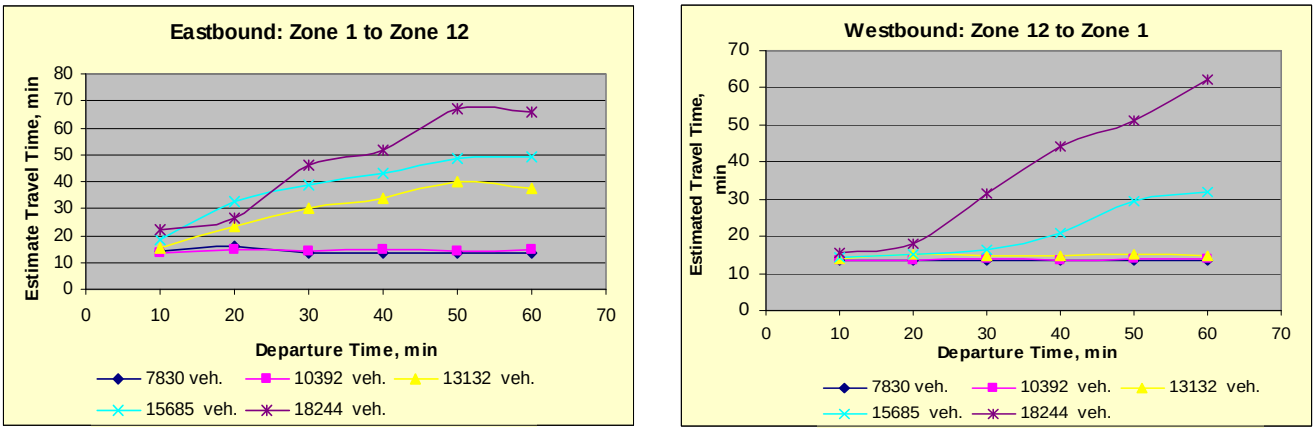

Figure 6.16: Estimated Data: Eastbound and Westbound Freeway Segment

Obtaining results for both the simulated and estimated travel times allows for the approximation of error that may occur due to the discrepancy between the two values. The equation used to determine this error is the following:

$$\text{Percent Error } (\%) = \left[\frac{b-a}{a}\right] \times 100$$

In the equation, b is defines as the estimated travel time value while a is the simulated travel time value, not equal to zero. The results, presented in the Table 6.17 and Figure 6.17, demonstrate that for the most part percentage errors range within ± 10 percent, which is rather satisfactory considering that the test cases also includes heavily congested scenarios. There are exceptions where percentage error values are greater than 10 percent, which are mostly associated to two factors. The first factor is high demand loading and the effects of congestion. The second factor is 'warming up' and 'cooling down' of both the simulation and *N*-Curve. This results in underestimation during earlier departure times and overestimation during later departure times.

Percentage errors were further placed in terms of volume weighted percent error for each of the six time stamps of interest. The equation used each of the time stamps is the following:

$$\text{Weighted Percentage Error}\,(\%) = \frac{\sum e \times v}{\sum v} \times 100$$

In the equation, e is equal to the error (previously obtained) and v is the volume associated with the error, more specifically, in terms of the number of vehicles generated at the specific time intervals. In both the east and westbound direction a general increase is observed from the beginning to end of simulation.

Table 6.17: Percent Error of Simulated and Estimated Data: Eastbound and Westbound Freeway Segment

Eastbound						
	Departure Time					
Generated Vehicles	***10 min.***	***20 min.***	***30 min***	***40 min***	***50 min***	***60 min***
7830	5.338	18.893	4.271	4.606	4.071	6.120
10392	-4.183	6.689	3.033	4.097	4.460	13.898
13132	-21.070	-7.801	0.125	-0.514	9.513	-8.271
15685	-24.816	4.159	4.331	2.058	0.577	-4.418
18244	-29.211	-34.174	-2.722	-11.790	0.738	-8.445
Westbound						
	Departure Time					
Generated Vehicles	***10 min.***	***20 min.***	***30 min***	***40 min***	***50 min***	***60 min***
7830	2.049	1.122	0.749	2.037	1.882	3.355
10392	-0.694	0.601	2.947	-0.657	0.952	4.749
13132	-4.033	1.158	2.350	-0.136	4.285	11.800
15685	-9.521	-10.751	-17.999	-14.410	21.681	36.022
18244	-20.183	-34.701	-9.380	-6.278	-0.708	28.842

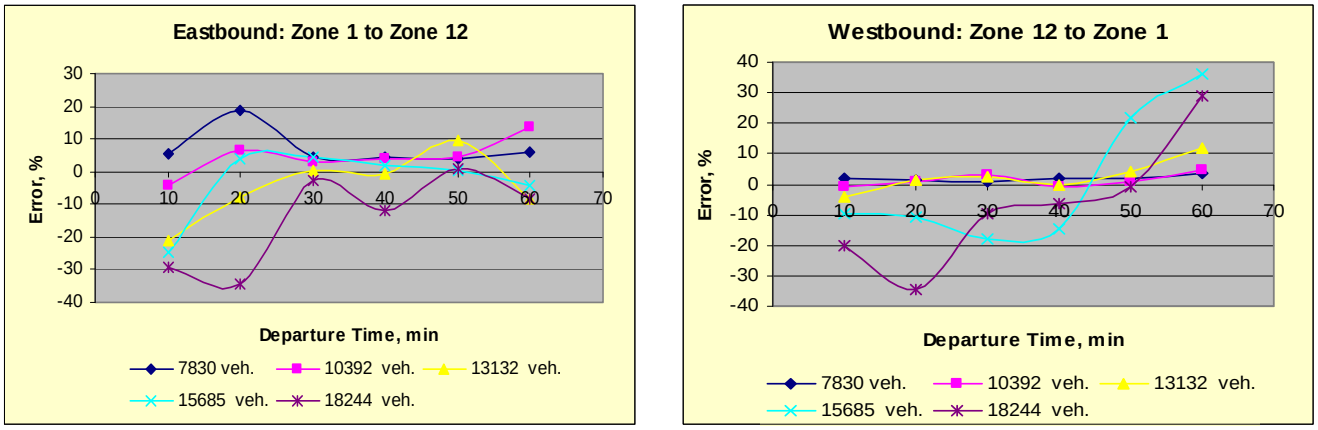

Figure 6.17: Percent Error: Eastbound and Westbound Freeway Segment

Table 6.18: Weighted Percent Error of Simulated and Estimated Data: Eastbound and Westbound Freeway Segment

	Departure Time					
Direction	***10 min.***	***20 min.***	***30 min***	***40 min***	***50 min***	***60 min***
Eastbound	-18.390	-6.789	1.300	-1.699	3.456	-2.139
Westbound	-8.604	-11.817	-5.914	-5.104	6.251	20.247

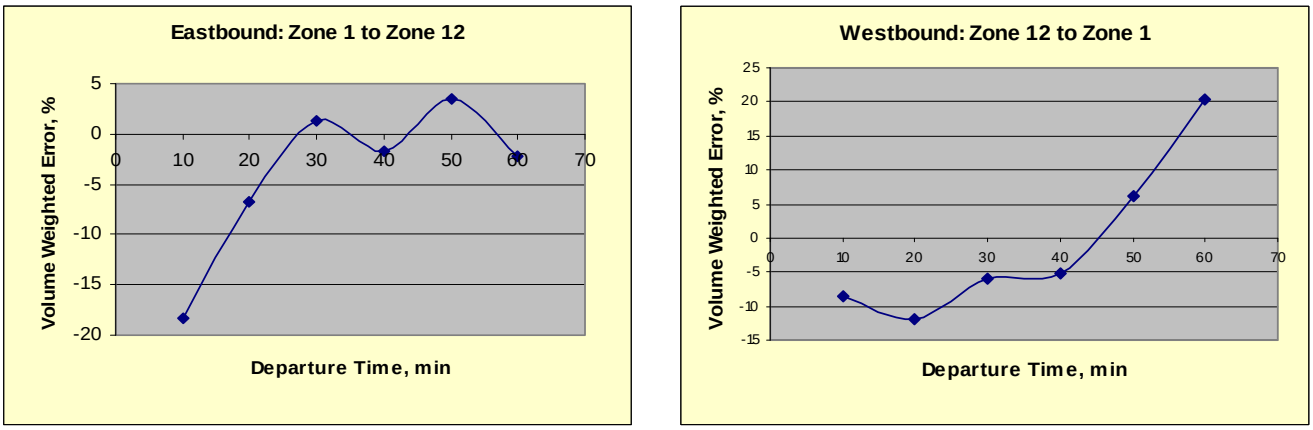

Figure 6.18: Volume Weighted Percent Error: Eastbound and Westbound Freeway Segment

The estimated results produced by the *N*-Curve method proved to have low percentage error discrepancy when compared to the simulation (actual) output results for low to medium demand loadings. In the case of heavy demand loading, vehicles loaded on the later portions of the hour experienced higher travel times. In addition, the percentage errors when comparing the estimated and simulated heavy demand loading cases yield higher values.

6.5 N-Curve Model Calibration and Validation Based on Field Data

The calibration and validation of the propose *N*-Curve approach was attempted during the course of research. As discussed previously, the errors present in the real-time data received from TxDOT created a significant amount of "noise," which made difficult to characterize the actual performance of the algorithm. That was the primary reason that the testing was primarily done in the simulation environment. Because in the simulation environment the traffic counts and experienced travel time can be accurately captured and the performance of the algorithm can then be corrected assessed.

6.6 Summary

The calibration and validation experiments conducted in this section exemplify the adequacy and robustness of the models developed for this project. Tests performed using simulated data suggests that high accuracies in travel time forecasts are possible for traffic conditions varying from free flow to congested states. Moreover, the model performance is not severely affected during the transition period, which is typically the hardest state to capture appropriately. Travel time prediction errors ranging between 10 and 23 percent were observed for a relatively short 3-mile section when the CTM-based model was implemented for a bell-shaped demand. Furthermore, 40 percent of the values were below 60 seconds. For different demand profiles, the resulting errors were found to be approximately in the same range of values, demonstrating the robustness of the model. Slightly better performance was observed under lighter traffic conditions, and the demand profile involving changes in the OD demand configuration was found to lead to the largest deterioration in performance. However, most of the prediction errors remained around 15 percent. The simulation experiments also suggest that

predictions for intervals of 3 to 5 minutes are more likely to be stable, which is appropriate for practical implementations.

The estimated results produced by the N-Curve method proved to have low percentage error discrepancy when compared to the simulation (actual) output results for low to medium demand loadings. In the case of heavy demand loading, vehicles loaded on the later portions of the hour experienced higher travel times. In addition, the percentage errors when comparing the estimated and simulated heavy demand loading cases yield higher values.

The numerical experiments conducted on field data are also encouraging, provided that necessary data filtering steps are accomplished. The average RMSE of 61 sec (18 percent) observed across all days for which GPS real traffic data was available also confirms the fact that the model is robust for varying lengths of highway segments, given that the test area spanned 10 miles. Furthermore, the model results were still accurate on those days on which detector malfunctioning reduced the availability of data at run time.

In summary, both models are capable of providing very accurate travel time predictions, robust with respect to traffic demands and detector location. Whereas the CTM-based model demands some additional work in terms of model preparation and calibration, it has the potential to adapt to varying conditions on the field. Moreover, once a cell-transmission representation is developed for a highway segment, it can be used as the basis to analyze other traffic management strategies. The N-Curve based model also responds the changes in traffic conditions, and is a powerful tool provided that appropriate data filtering and verification procedures are developed.

Chapter 7. Offline Detector Coverage Analysis and Integrated Analytical Simulation Platform

For this project, a specialized offline detector coverage analysis tool was developed. Existing literature regarding traffic sensor deployment was found to focus mainly on identifying optimal location of sensors for purposes other than travel time, such as improving Origin-Destination matrix estimation (for example, Ehlert et al., 2006, Sherali et al., 2006). Moreover, much of the surveyed work intends to select which links of a network should be equipped (Thomas and Upchurch 2002, Sisiopiku et al 1994), rather than identify detector positions within a link that may present grater advantages. The latter has only been studied in an urban context, and generally for traffic management purposes, such as actuated traffic signs (Liu et al., 2004). By performing a ***fine-resolution (cell level) analysis of the impact of sensor location on travel time prediction accuracy, which is explicitly modeled, the proposed methodology improves upon existing techniques.*** Moreover, when combined with the previously introduced travel time prediction methods, the methodology constitutes a powerful integrated simulation and analytical framework.

7.1 Offline Detector Coverage Analysis

The offline detector coverage analysis tool can be used to analyze travel time prediction performance for different detector deployment patterns. Such patterns are defined in terms of the cells on which sensors are placed, providing a finer level of resolution than traditional models. For example, if three detectors are available (see Figure 7.1), the following are possible location patterns within a link:

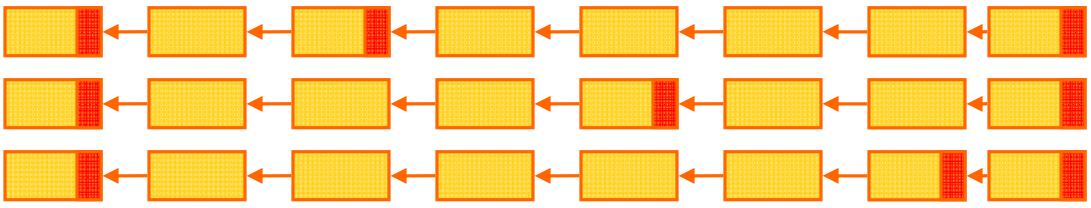

Figure 7.1: Possible detector deployment patterns on a highway segment (cell representation)

The proposed model evaluates the performance of the CTM-based travel time prediction model. The same structure can be used to assess the behavior of other models. In general, one may expect similar results for all prediction methodologies that utilize cumulative traffic counts as the basis for their travel time forecasts. Given the complexity of the relationship between traffic volumes, actual travel times, and predicted travel times, it is not feasible to write a single mathematical expression capturing the model performance. The proposed optimization approach enumerates all desirable detector deployment patterns, compares simulated versus real travel times, computes error measurements, and selects the pattern leading to smaller prediction errors. The developed methodology requires the use of an external simulation tool for the following purposes:

- Provide “real” travel times under a range of traffic conditions
- Provide traffic counts for every possible hypothetical location of the detectors

The optimal detector deployment model ***can be run in two ways***: allowing the software to generate and evaluate all possible detector deployment patterns, or providing the set of patterns to be analyzed as an input.

While enumerating all possible patterns is a more comprehensive option, it may increase the computational work excessively, because the number of possible patterns grows very fast as the number of cells increases. If C is the number of cells, and D is the total number of available detectors, then the number of possible combinations is given by:

$$NC = \frac{C!}{D!(C-D)!}$$

A way to reduce the number of patterns to be considered is to include a threshold, limiting the minimum separation between detectors. It was observed in the simulation experiments that errors in travel time computations increased significantly for detector spacing lesser than 0.5 miles apart. Moreover, deploying detectors closer than a half-mile apart may not be economically feasible. When the software provided in this package is used to generate all possible patterns (with or without a spacing threshold), it informs the user about the total number of strategies to be analyzed, providing a chance to cancel the model execution if such number were considered excessive. Furthermore, when more than one million feasible patterns are generated (in this context, *feasible* implies that they respect the threshold requirement specified by the user), the model execution is automatically cancelled, and the user is prompted to either feed a smaller number of combinations manually, or to increase the provided threshold. Other possibilities for path generation include the use of heuristic procedures, such as genetic algorithms, which may be coded separately and linked to the source. The following diagram (see Figure 7.2) depicts the optimal detector location process:

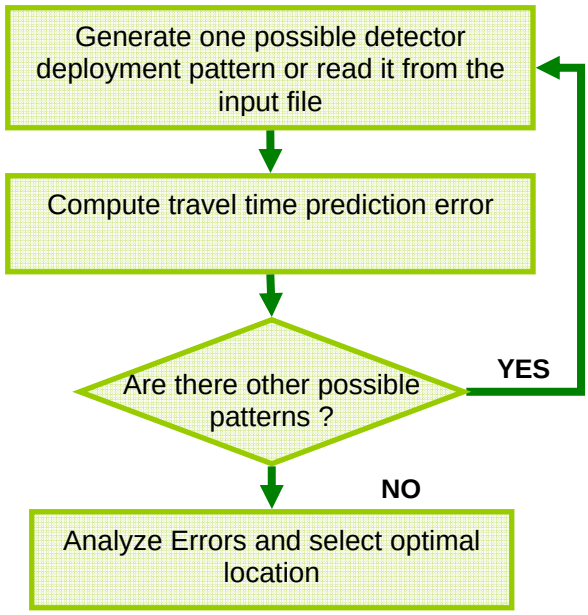

Figure 7.2: Flowchart for optimal detector location

The output file generated by this software provides, for each analyzed pattern, a number of error measurements for every OD pair considered for traffic prediction, as well as a global error measurement. The error definitions are included in the corresponding templates. Notice that, in virtue of the existence of multiple origins and destinations, it is not always straightforward to identify whether one pattern performs better than others. A pattern exhibiting a low global error may involve high errors for specific OD pairs. One possible criterion to select an optimal pattern is to search for the one which minimizes the maximum error across OD pairs. The user may develop and code other criteria into the source code.

7.2 Numerical Experiments

Different types of numerical experiments were conducted in order to gain insights into the optimal sensor deployment problem. All such experiments were performed utilizing the network described in Chapter 6 (validation using simulation), and the second demand profile. New simulation runs were necessary to generate the traffic counts corresponding to each hypothetical sensor location.

The first of such experiments simply considered subsets consisting of different number of detectors spaced as evenly as possible. The results suggest that for the travel time methodology proposed in this model, detector separations ranging from 0.75 mi to 1.5 miles lead to more accurate results than smaller ones. This was further confirmed by the second set of experiments, in which the best results obtained using 10 detectors were found to be less accurate that those based on 3 detectors.

The second set of experiments generated all the possible detector placements for sets of 3 and 10 detectors, and minimum spacing of 1900 ft and 1650 ft, respectively. Tables 7.1 and 7.2 show all the feasible combinations (which respect the minimum separation threshold), and the

corresponding global RMSE (value and percent). The last column in Table 7.1 displays the maximum RMSE across sections corresponding to each deployment strategy.

Table 7.1: Feasible deployment patterns for 3 detectors and a 1900ft separation threshold

	Cell			RMSE %	RMSE Sec	MAX RMSE
	Det 1	Det 2	Det 3			
1	1	5	34	**13%**	**25.1**	**25%**
2	1	6	34	**14%**	**28.6**	**26%**
3	1	7	34	**13%**	**27.8**	**27%**
4	1	8	34	**13%**	**26.5**	**26%**
5	1	9	34	**15%**	**30.4**	**29%**
6	1	10	34	**15%**	**29.7**	**23%**
7	1	11	34	**13%**	**27.9**	**22%**
8	1	12	34	**13%**	**27.4**	**22%**
9	1	13	34	**16%**	**36.2**	**22%**
10	1	14	34	**12%**	**25.5**	**22%**
11	1	15	34	**12%**	**23.8**	**22%**
12	1	16	34	**12%**	**24.6**	**22%**
13	1	17	34	**12%**	**24.3**	**22%**
14	1	18	34	**12%**	**26.8**	**22%**
15	1	19	34	**12%**	**26.5**	**22%**
16	1	20	34	**12%**	**25.7**	**22%**
17	1	21	34	**13%**	**26.2**	**23%**
18	1	22	34	**13%**	**26.1**	**23%**
19	1	23	34	**13%**	**28.1**	**23%**
20	1	24	34	**13%**	**28.6**	**23%**
21	1	25	34	**13%**	**27.4**	**23%**
22	1	26	34	**12%**	**26.7**	**23%**
23	1	27	34	**12%**	**26.1**	**23%**
24	1	28	34	**13%**	**26.1**	**23%**
25	1	29	34	**13%**	**26.3**	**23%**

Table 7.2: Feasible deployment patterns for 10 detectors and a 1650ft separation threshold

	Cell										RMSE %	RMSE Sec
	Det 1	Det 2	Det 3	Det 4	Det 5	Det 6	Det 7	Det 8	Det 9	Det 10		
1	1	5	9	13	16	19	22	25	28	34	**23%**	**55.8**
2	1	5	9	13	16	19	22	25	29	34	**23%**	**55.6**
3	1	5	9	13	16	19	22	25	30	34	**23%**	**55.4**
4	1	5	9	13	16	19	22	26	29	34	**24%**	**56.2**
5	1	5	9	13	16	19	22	26	30	34	**23%**	**56.0**
6	1	5	9	13	16	19	23	26	29	34	**23%**	**52.8**
7	1	5	9	13	16	19	23	26	30	34	**23%**	**52.5**
8	1	5	9	13	16	20	23	26	29	34	**22%**	**51.9**
9	1	5	9	13	16	20	23	26	30	34	**22%**	**51.6**
10	1	5	9	13	17	20	23	26	29	34	**22%**	**48.6**
11	1	5	9	13	17	20	23	26	30	34	**21%**	**48.3**
12	1	5	9	14	17	20	23	26	29	34	**17%**	**36.5**
13	1	5	9	14	17	20	23	26	30	34	**17%**	**36.6**
14	1	5	10	14	17	20	23	26	29	34	**16%**	**35.5**
15	1	5	10	14	17	20	23	26	30	34	**16%**	**35.6**
16	1	6	10	14	17	20	23	26	29	34	**18%**	**41.6**
17	1	6	10	14	17	20	23	26	30	34	**18%**	**41.7**
18	1	7	10	14	17	20	23	26	29	34	**19%**	**41.6**
19	1	7	10	14	17	20	23	26	30	34	**19%**	**41.7**

An interesting observation is that the optimal ***results corresponding to 3 detector strategies are consistently lower than those obtained based on 10 detectors***. This is a consequence of the methodology used by this model to compute travel times and it has **very important implications**. Firstly it suggests that a careful analysis of sensor deployment that focuses on their main purpose may reduce the required investment. The second implication of these outcomes is that, even though sensor deployment decisions may respond to needs other than travel time prediction, relatively sparse coverage may lead to fairly accurate results if the appropriate sensors are selected to generate the forecasts. Furthermore, thanks to this property the model can accommodate missing or malfunctioning with relatively little changes in its performance.

The same experiment was run utilizing ***different thresholds (***1500 ft and 1700 ft) for a 10-detector scenario. It was observed that even though most of the feasible patterns for the 1500 ft scenario led to higher global RMSE than those feasible for a 1700 ft one, a few of them actually exhibited errors in the order of 13-15 percent. This suggests that the minimum threshold

should be selected carefully, because it impacts the ability of the model to place detectors in critical positions, which may compensate for the reduced spacing between sensors.

Another important property of this deployment analysis tool can be observed in Table 7.3, which shows that the ***deployment strategy leading to the smallest RMSE is not necessarily the one resulting in the least error for all sensors***. This is the reason why the output file is designed to provide decision makers with a broader set of information about each analyzed detector deployment pattern. Such files include the prediction error in percentage and value format for each OD pair, as well as the absolute value of the average error, and the corresponding frequency distributions. It can be used to select the pattern that better fits the interests of a particular agency, and to decide between alternatives exhibiting the same global error measurements.

Table 7.3: RMSE by section for 3 detector deployment patterns (1900 ft distance threshold)

	% RMSE by Section											
Combo	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**
1	9.4	17.1	25.5	15.7	8.0	8.8	10.0	9.7	9.1	10.6	9.8	9.2
2	9.4	18.5	26.4	17.0	9.1	10.4	11.3	10.8	10.3	11.9	11.0	10.4
3	9.4	10.8	26.9	17.8	10.0	10.7	11.0	10.6	10.3	11.7	10.8	10.3
4	9.4	10.8	25.8	17.3	10.2	10.6	10.9	10.6	10.3	10.8	10.1	9.8
5	9.4	10.8	29.2	20.0	11.3	11.9	11.7	11.2	10.9	12.6	11.8	11.3
6	9.4	10.8	22.2	22.7	21.8	12.3	12.0	11.2	10.5	13.5	12.2	11.5
7	9.4	10.8	22.2	20.8	10.9	11.3	11.1	10.5	9.9	12.3	11.2	10.7
8	9.4	10.8	22.2	13.9	9.5	11.5	11.9	11.4	10.7	12.2	11.3	10.7
9	9.4	10.8	21.8	13.5	13.3	19.2	19.3	17.3	16.7	15.9	15.8	15.0
10	9.4	10.8	22.2	13.9	9.5	10.6	11.0	10.5	9.8	10.9	10.0	9.5
11	9.4	10.8	22.2	13.9	9.2	9.2	10.0	9.7	9.2	10.2	9.4	8.9
12	9.4	10.8	22.2	13.9	9.2	10.5	11.0	10.6	10.0	10.7	9.9	9.3
13	9.4	10.8	22.2	13.9	9.2	10.4	11.0	10.6	9.9	10.4	9.6	9.2
14	9.4	10.8	21.8	13.5	8.9	9.7	11.5	11.3	11.0	10.7	10.3	10.0
15	9.4	10.8	22.2	13.9	9.2	9.9	12.0	11.4	10.7	10.9	10.3	9.9
16	9.4	10.8	22.2	13.9	9.2	9.9	11.6	11.2	10.8	10.3	9.9	9.5
17	9.4	10.8	23.0	14.7	9.7	10.3	12.1	11.5	10.9	11.4	10.8	10.2
18	9.4	10.8	23.0	14.7	9.7	10.3	12.1	11.5	11.0	11.1	10.7	10.0
19	9.4	10.8	23.2	14.9	9.9	10.4	12.3	11.6	11.6	10.4	10.9	10.5
20	9.4	10.8	23.2	14.9	9.9	10.4	12.3	11.6	11.5	10.9	11.4	11.0
21	9.4	10.8	23.0	14.7	9.8	10.3	12.1	11.5	10.9	10.4	10.8	10.4
22	9.4	10.8	23.1	14.8	9.8	10.3	12.2	11.5	10.9	10.0	10.3	9.9
23	9.4	10.8	23.1	14.8	9.8	10.3	12.2	11.5	10.9	10.2	10.1	9.7
24	9.4	10.8	23.0	14.7	9.8	10.3	12.1	11.5	10.8	11.9	11.3	10.5
25	9.4	10.8	23.2	14.9	9.9	10.4	12.3	11.6	11.0	12.2	11.3	10.5

The final experiments explored all possible ***locations of a detector within a link***. Similar results were observed when the selected link was immediately after an exit or an entry ramp. The highway sections between detector 10 and 11, and between 11 and 12, were used as example of the former cases. The first segment is preceded by an input ramp, and it was found that the most accurate results were obtained when a detector was placed immediately after the ramp. Similarly, for the link connecting detectors 11 and 12, which leads to an exit ramp, it was observed that

more accurate results were obtained if the detector was placed on the diverge cell. There results, although clearly not conclusive, are reasonable. General observations from other numerical experiments suggest that whenever entry or exit ramps are not properly measured, detectors should be placed as close to the corresponding merge/diverge section as possible (and downstream of the same). In general, detectors are always desirable after diverge sections, particularly when the split ratios vary widely along the day.

Another important consideration for optimal sensor placement is its robustness with respect to changes in demand variations. Even though the optimization process proposed in this work does not take explicitly into account ***different demand patterns***, it was observed that the resulting deployment strategies were fairly stable for different demand profiles. For example, the optimal location found using profile demand 1 resulted in almost the same average error computed for demand profile 4 under the base deployment strategy (22 percent RMSE, with 31 percent of the travel time estimations presenting less than a 1 minute error). However, if major changes in OD configurations are expected, several experiments may be conducted, corresponding to the various demand scenarios, and the robustness of each solution evaluated appropriately.

7.3 Integrated Analytical Simulation Framework

As a part of this project, a generic analytical and simulation-based framework (Figure 7.3) has been developed, which can be used for conducting future travel time prediction studies. This framework primarily consists of the insights learned during the various phases of the project streamlined into one cohesive and concise flowchart.

The first step in any travel time prediction study is to determine the data to be used as input for determining the travel times. The models developed as a part of this study use the count data as it is the easiest to obtain from the field. Single loop detectors are commonly used to arrive at the count data in any freeway section. However, depending on the availability of the technology, other data collection devices like automatic vehicle identification devices can also be used. Ideally another metric of the traffic flow such as speed must be collected so that the validity and accuracy of the count data can be verified at regular time intervals. Another factor to be decided is the temporal aggregation level of the count data. As the focus of this project is on short term forecasting accounting for traffic dynamics, efforts should be taken to make the aggregation level as fine as possible. Ideal case would be count data at a minute level.

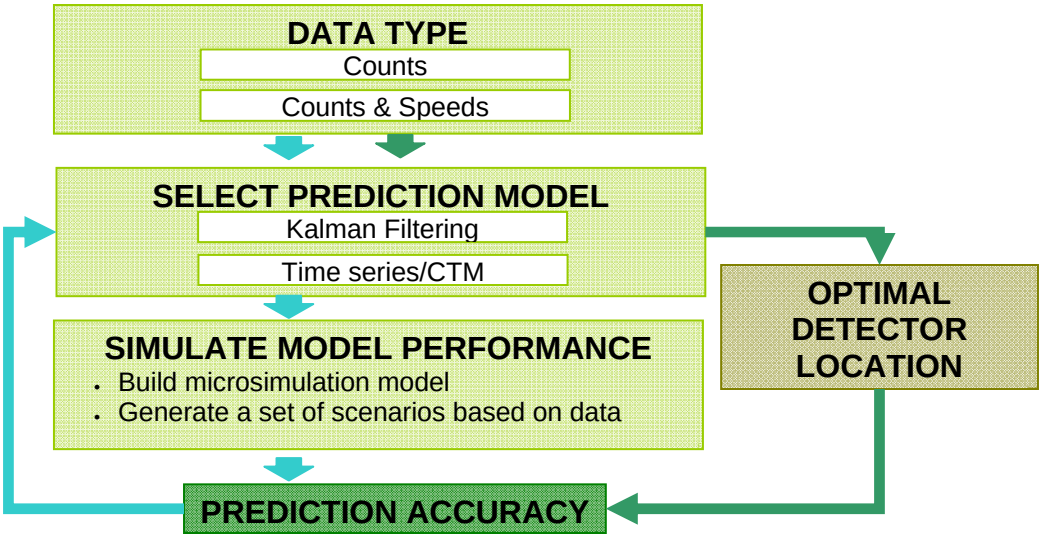

Figure 7.3: An integrated analytical simulation framework

The choice of the data type has a great bearing on the type of the prediction model to be used. If count data is available at a reasonable aggregation level, then either of the models developed in this project can be used. However, the parameters of both the models needs to be calibrated before it can be applied. For calibration two types of data will be used: (i) archived real-time data sets coupled with the actual travel times and (ii) simulation data.

The advantage of using archived real data is that it captures real freeway dynamics subject to detector error. However, depending on the amount of effort put into data coverage archived data has limited availability and spatial coverage. The archived data is useful for model calibration and validation. A microscopic simulation model of the freeway model should be created. The microscopic model will be useful for generating simulation based count data and tracking the vehicle trajectories to obtain the actual travel times. The advantage of the simulation data is that it allows for detailed testing of model performance under multiple scenarios. The performance of the prediction model can be compared for numerous peak and off-peak demand scenarios. The set of scenarios to be tested should account for daily, monthly and seasonal variations in demand which can be obtained by studying past trends.

For determining optimal detector locations, it is very difficult and in most cases infeasible to do field testing for different configurations. Therefore the microscopic simulation model must be used to generate counts at detector for various detector configurations and calculate the actual travel times from vehicle trajectories. The prediction model must be run for each set of counts for each configuration and then compared against actual travel times. Repeat the procedure for numerous scenarios and then obtain the configuration which gives the least error across multiple scenarios.

Thus, in order to obtain a resilient travel time prediction model that performs well under numerous scenarios, it is important to conduct a detailed study involving both simulation data and real-time data as illustrated in this framework.

7.4 Summary

This section presented a methodology to evaluate different detector deployment strategies, and select the one leading to the most accurate travel time predictions. The proposed technique is novel in several aspects: it explicitly considers the impact of the position of a detector in the travel time forecast accuracy, and it analyzes detector location at a cell resolution level, instead of the typical link-level analysis.

The computer model developed to implement the new methodology can be run in two different modes. The first one analyzes a pre-defined set of possible patterns, whereas the second one automatically generates all possible deployment strategies. The later involves enumerating all possible cell combinations of "d" elements, where "d" is the number of available detectors. This figure can be very large, and it is possible to reduce it by incorporating a distance threshold, specifying the minimum spacing between contiguous detectors. The methodology is based on the performance of the CTM-based travel time prediction software, and demands the utilization of a micro-simulation tool to synthesize the corresponding input data. The results are expected to be valid for most cumulative-count based techniques, and the general insights obtained from its application can be extended to more general methods.

The numerical experiments reported in this section suggest that the location of detectors does have an impact on the model performance. As was somewhat expected, the outcomes highlight the importance of locating sensors close to merge/diverge sections. Due to the characteristics of the travel time prediction procedure, it was also observed that utilizing detectors placed closer than half mile apart may affect the model stability. Additional insights from the numerical testing suggest that the optimal detector placement resulting from the proposed optimization process is fairly robust with respect to changes in demand patterns. These results are encouraging, and indicate that the model can be used to gain insights regarding the number an approximate location of detectors which leads to better travel time predictions. However, the fact that model parameters can be calibrated and adjusted for each possible detector deployment pattern leaves room for better observed performances than those considered by the optimization software. The latter implies that the results obtained using the provided detector deployment analysis tool should be considered as guidance, which must be complemented with experience regarding additional practical considerations and engineering judgment.

The proposed detector coverage analysis tool, in combination with the previously introduced travel time prediction methods, constitutes a powerful integrated simulation and analytical framework. The insights learned during the various phases of the project can be streamlined into one cohesive and concise flowchart, which can be used to analyze future ITS deployment strategies, improve the utilization of existing sensors, and asses the need for system expansions or modifications.

Chapter 8. Conclusions

Dynamic message sign (DMS) systems provide full public value only when predicted travel time information is integrated into it. The research conducted towards this project lead to innovative travel time prediction tools, accurate for a range of traffic conditions and detector coverage scenarios. It also contributed to a deeper understanding of the relationship between sensor coverage and the quality of freeway traffic time forecasts.

During the first stages of the project, studies were conducted to understand the state of the art and the practice in travel time prediction. A fundamental flaw found in most applied methods is that they are not predictive in nature and therefore do not anticipate the short-term evolution of traffic dynamics downstream. As concluded by Quiroga (2000), such a simple but non-robust travel time prediction scheme cannot provide satisfactory prediction during recurrent peak hours, in which travel time prediction is more important than the off-peak hours. Similarly, current practices for travel time predictions will not perform well under non-recurrent traffic conditions such as work zones, accidents, and special events. When motorist are stuck in a long queue without knowing the exact cause, reasonable travel time prediction on DMS becomes critical to help ease road rage, and to assist motorists to evaluate whether they will divert to alternative routes. Based on the research team's personal communication with traffic engineers in San Antonio district, the DMS travel time prediction messages are usually turned off during peak-hours, work zones, and accidents. There is ample room for improving the freeway travel time prediction under different traffic conditions with an improved framework for all the DMS systems operated in Texas.

The analysis of existing travel time prediction literature revealed that the prediction capability of most of existing approaches depends on the data set used for validation and calibration. There are relatively very few studies that use analytical traffic flow relationships or simulation based models to determine the current travel times. Usage of traffic flow theoretic relationships or simulation-based models may be more useful in extrapolating local conditions (like spot speed data) to that of a link (like travel time on a link). Preliminary tests of existing methodologies confirmed the inadequacy of simple analytical models to reflect varying traffic conditions. This motivated the fundamentally novel methodologies developed for this project, which are able to capture driver's behavior and dynamic traffic evolution.

The integrated simulation statistical analysis based model predicts the inflow into the freeway network using a time Series (ARIMA) model. The forecasted inputs are then simulated using a Cell Transmission Model to obtain the travel times. The new model accounts for the evolving dynamics of traffic flow in predictions and is computationally efficient. Therefore, it can be run in a normal personal computer as long as data is available. The calibration and validation experiments conducted to test the model confirmed their adequacy and robustness. Tests performed using of simulated data suggests that high accuracies in travel time forecasts are possible for traffic conditions varying from free flow to congested. Travel time prediction errors ranging between 10 and 23 percent were observed for a relatively short 3-mile section when the CTM-based model was implemented for a bell-shaped demand. Furthermore, 40 percent of the values were below 60 seconds. For different demand profiles, the resulting errors were found to be approximately in the same range of values, demonstrating the robustness of the model.

The numerical experiments conducted on field data are also encouraging, provided that necessary data filtering steps are accomplished. The average RMSE of 61 sec (18 percent)

observed across all days for which GPS real traffic data was available also confirms the fact that the model is robust for varying lengths of highway segments, given that the test area spanned 10 miles. Furthermore, the model results were still accurate on those days on which detector malfunctioning reduced the availability of data at run time.

The estimated results produced by the N-Curve method proved to have low percentage error discrepancy when compared to the simulation (actual) output results for low to medium demand loadings. In the case of heavy demand loading, vehicles loaded on the later portions of the hour experienced higher travel times. In addition, the percentage errors when comparing the estimated and simulated heavy demand loading cases yield higher values.

The models proposed in this study use as main inputs traffic counts, which are provided by point detectors, and widely available. However, additional data is necessary to calibrate and validate the models. An overview of such data was provided, which also described the main datasets used throughout the project. These involve two types of data: traffic counts and real travel time measurements. Whereas the first type is relatively easy to obtain, corresponding travel times are typically harder to gather, which is one of the main motivations behind the utilization of simulated data. In this study both real and simulated travel time data has been used to validate and calibrate the models. The real travel time data is useful for testing the efficacy of the model against real world dynamics whereas the simulated data is useful for testing the efficiency of the model under a number of scenarios. The simulated data was generated using a VISSIM microsimulator. Real travel time data were obtained from the Berkeley highway network and in El Paso. The traffic counts in El Paso were obtained from the TransVISTA and a GPS system was used to record the actual travel times. The data obtained from the field may contain errors due to detector malfunctions. Therefore, filtering and pre-processing techniques were also recommended.

In addition to online travel time prediction methodologies, this project provided a methodology to evaluate different detector deployment strategies, and select the one leading to the most accurate travel time predictions. The proposed technique is novel in several aspects: it explicitly considers the impact of the position of a detector in the travel time forecast accuracy, and it analyzes detector location at a cell resolution level, instead of the typical link-level analysis.

The methodology is based on the performance of the CTM-based travel time prediction software, and demands the utilization of a micro-simulation tool to synthesize the corresponding input data. The results are expected to be valid for most cumulative-count-based techniques, and the general insights obtained from its application can be extended to more general methods. The numerical experiments conducted to assess the proposed methodology suggest that the location of detectors does have an impact on the model performance. As was somewhat expected, the outcomes highlight the importance of locating sensors close to merge/diverge sections. Additional insights from the numerical testing suggest that the optimal detector placement resulting from the proposed optimization process is fairly robust with respect to changes in demand patterns. These results are encouraging, and indicate that the model can be used to gain insights regarding the number an approximate location of detectors, which leads to better travel time predictions. However, the fact that model parameters can be calibrated and adjusted for each possible detector deployment pattern leaves room for better observed performances than those considered by the optimization software. The later implies that the results obtained using the provided detector deployment analysis tool should be considered as guidance, which must be

complemented with experience regarding additional practical considerations and engineering judgment.

The proposed detector coverage analysis tool, in combination with the previously introduced travel time prediction methods, constitutes a powerful integrated simulation and analytical framework. The insights learned during the various phases of the project can be streamlined into one cohesive and concise flowchart, which can be used to analyze future ITS deployment strategies, improve the utilization of existing sensors, and assess the need for system expansions or modifications.

The final stage of the project considered the deployment of travel time prediction methodologies in a real TMC. During this stage, data quality, pre-processing, and communication protocols were identified as the vital issues to be addressed in a real-time model implementation.

In summary, this project developed analytical models that can be used to improve the quality of travel time prediction for ITS implementations, by capturing dynamic traffic variations and congestion evolution. It also provided software tools to implement such models, and a thorough analysis of data requirement, as well as data pre- and post-processing. The latter led to a database software capable of displaying detector measurements and travel times in real-time. The online travel time prediction models, coupled with the offline detector coverage analysis tool, can be used in an integrated framework to assess and enhance the performance of existing ITS infrastructures and to plan future deployment strategies.

References

1. Al-Deek, H M, and C.V.S.R. Chandra, C.V.S.R. 2004. New Algorithms For Filtering And Imputation Of Real-Time And Archived Dual-Loop Detector Data In I-4 Data Warehouse. Transportation Research Record 1867, 2004.

2. Brockwell, P. J., and R. A. Davis. *Introduction to time Series and Forecasting.* Springer, New York, NY., 2002.

3. Sun, C., S.G. Ritchie, K. Tsai, and R. Jayakrishnan. 1999. Use of Vehicle signature analysis and lexicographic optimization for vehicle re-identification on freeways. Transportation Research Part C. vol 7. 167-185

4. Chakroborty, P. and S. Kikuchi. Estimating travel times on urban corridors using bus travel time data. *Presented at the 83rd Annual Meeting of Transportation Research Board*, Washington D.C., 2004.

5. Chen, L., and A.D. May. 1987. Traffic Detector Errors and Diagnoses. Transportation Research record 1132, pp. 82-93.

6. Chen, M. and S. I. J. Chien. Dynamic Freeway Travel Time Prediction Using Probe Vehicle Data: Link-based vs. Path-based. *Presented at 80th Annual Meeting of Transportation Research Board*, Washington D.C., 2001.

7. Chien, S. I.-J. and C.M. Kuchipudi. Dynamic Travel Time Prediction with Real-Time and Historic Data. *Journal of Transportation Engineering*, Vol. 129, No. 6, 2003, pp. 609-616.

8. Chien, S., and C. M. Kuchipudi. Dynamic travel time prediction with real-time and historical Data. *Presented at 81st Annual Meeting of Transportation Research Board*, Washington D.C., 2002.

9. Coifman, B and S. Dhoorjaty. 2004. Event Data-Based Traffic Detector Validation Tests. J. Transp. Engrg., Volume 130, Issue 3, pp. 313-321 (May/June 2004)

10. Coifman, B.,. 2006. Vehicle level evaluation of loop detector and the remote traffic microwave sensor. J. Transp. Engrg., Volume 132, Issue 3, pp. 213-226.

11. Coifman, B.,1998, Vehicle Reidentification and Travel Time Measurement in Real-Time on Freeways Using the Existing Loop Detector Infrastructure, Transportation Research Board, 1998.

12. Daganzo, C. The cell transmission model. Part II: Network traffic**.** *Transportation Research B*, Vol. 29, No. 2, 1995, 79-93.

13. Daganzo, C. The cell transmission model: A dynamic representation of highway traffic consistent with the hydrodynamic theory. *Transportation Research B*, Vol. 28, No. 4, 1994, 269-287.

14. Daganzo, C. Using Input-Output Diagrams to Determine Spatial and Temporal Extents of a Queue Upstream of a Bottleneck. Transportation Research Record 1572, 1997

15. Dahlgren, J., S. Turner, and Reinaldo C. Garcia. (2002). "Collecting, Processing,

16. De Ruiter, J. C. C., Schouten, W. J. J. P. and J. H Frijdal. Automation of travel times calculation in the Netherlands. *Proceedings of eleventh International Conference on Road Transport Information and Control*, London, UK, 2002, pp. 196-200.

17. Dharia, A. and H. Adeli. Neural network model for rapid forecasting of freeway link travel time. *Engineering Applications of Artificial Intelligence*, Vol. 16, No. 7-8, 2003, pp. 607-613.

18. Ehlert, A., M.G.H. Bell, and S. Grosso .2006. The optimization of traffic count locations in road networks. Transportation Research Part B. Vol. 40(6) .pp. 460-479 , 2006.

19. FHWA. 1998. Travel Time Collection Handbook. Report FHWA-PL-98-035.

20. Ishaak,S., and H. Al-Deek. Statistical Evaluation of I-4 Prediction System. *Presented at 82nd Annual Meeting of Transportation Research Board*, Washington D.C., 2003.

21. Kuchipudi, C. M. and S. Chien. Development of a Hybrid model for Dynamic Travel Time Prediction. *Presented at 82nd Annual Meeting of Transportation Research Board*, Washington D.C., 2003.

22. Kwon, J., Coifman, B. and P. Bickel. Day-to-Day Travel Time Trends and Travel Time Prediction from Loop Detector Data. *Presented at 82nd Annual Meeting of Transportation Research Board*, Washington D.C,2002.

23. Mark, C.D, Sadek, A.W., and D. Rizzo. Predicting experienced travel times with Neural Networks: a PARAMICS simulation study. *Proceedings of the 7th International IEEE conference on Intelligent Transportation Systems, 2003.*

24. May, A.D. *Traffic Flow Fundamentals*. Prentice-Hall, Englewood Cliffs, NJ., 1990.

25. Muñoz, L., Sun, X., Sun, D., Gomes, G., and R. Horowitz, R.:. "Methodologicalcalibration of the cell transmission model." *Proceedings of the 2004 AmericanControl Conference*, Boston, MA, 2004, pp. 798-803.

26. Nanthawichit, C., Nagatsuji, T., and H. Suzuki. Application of Probe Vehicle Data for Real-time Traffic state estimation and Short Term Travel time prediction on a freeway. *Presented at the 82nd Annual Meeting of Transportation Research Board*, Washington D.C., 2003.

27. Nguyem, L.N., and W.T. Scherer. 2003. Imputation Techniques to Account for Missing Data in support of Intelligent Transportation Systems Applications. Research Report No. UVACTS-13-0-78. http://cts.virginia.edu/docs/UVACTS-13-0-78.pdf .

28. Ohba, Y., Koyama, T. and S. Shimada. Online-Learning Type of Traveling Time Prediction Model in Expressway. *Proceedings of the 7th International IEEE conference on Intelligent Transportation Systems,* 1998, pp. 351-355.

29. Palacharla, P. V. and P. C. Nelson. Application of fuzzy logic and neural networks for dynamic travel time estimation. *International Transactions in Operational Research*, Vol. 6, No. 6, 1999**,** pp. 145-160.

30. Rice, J. and E. Van Zwet. A simple and effective method for predicting travel times on freeways. *Proceedings of Proceedings of IEEE Conference on Intelligent Transportation Systems*, Oakland, U.S., 2001, pp. 227- 232.

31. Urban Crossroads, Inc. 2006. PeMS Data Extraction Methodology and Execution. Technical Memorandum for the southern California Association of Governments. http://scag.ca.gov/modeling/pdf/Pems_Technical_Memorandum_Final.pdf (Accessed October 2007).

32. Sherali, H.D., J. Desai, and H. Rakha. 2006. A discrete optimization approach for locating Automatic Vehicle Identification readers for the provision of roadway travel times. Transportation Research Part B. Vol 40 (10) pp 857-871, 2006.

33. Sisiopiku, V.P., N.M. Rouphail, and A. Santiago.1994. Analysis of Correlation between Arterial Travel Time and Detector Data from Simulation and Field Studies. Transportation Research Record 1457, TRB, National Research Council, Washington, D.C., 1994.

34. Thomas, G.B., and J.E. Upchurch. 2002. Simulation of Detector locations on an arterial street management system. Proceedings of the mid-continent transportation symposium.2002-206, 2001.

35. Van Arem, B., Van Der Vlist, M.G.M, Muste M.R., and Smulders S.A. Travel time estimation in the GERDIEN project. *International Journal of Forecasting*, Vol. 13, 1997, pp. 73-85.

36. Van Lint, J.W.C. Confidence intervals for real-time freeway travel time prediction. *Proceedings of IEEE Conference on Intelligent Transportation Systems*, Shanghai, China, 2003, pp. 1453-1458.

37. Van Lint, J. W. C. , Hoogendoorn S. P. , and H.J.Van Zuylen. Freeway travel time prediction with state-space neural networks: Modeling state-space dynamics with recurrent neural networks. *Presented at 82nd Annual Meeting of Transportation Research Board*, Washington D.C., 2002.

38. Wagner, H.M. *Principles of Operations Research*. Prentice Hall, Englewood Cliffs, NJ, 1977.

39. Wunnava, S., Yen, K., Babij, T., Zavaleta, R., Romero, R., Archilla, C., **"**Travel Time Estimates Using Cell Phones on Highways and Roads." Final Report Prepared for the Florida Department of Transportation, Florida International University, January 29, 2007.

40. Waller, S.T. and A.K. Ziliaskopoulos. A Visual Interactive System for Transportation Algorithms. *Presented at the 78th Annual Meeting of the Transportation Research Board,* Washington D.C., 1998.

41. Xie, C., Cheu, R. L. and D.H. Lee. Improving Arterial Link Travel Time Estimation by Data Fusion. *Proceedings of the 83rd Transportation Research Board Annual Meeting*, Washington, D.C., 2004.

42. Zhang, X. and J.A. Rice. Short-term Travel Time Prediction. *Transportation Research Part C: Emerging Technologies*, Vol. 11, 2003, pp. 2-4.

43. Ziliaskopoulos, A.K. and S.T. Waller. An Internet Based Geographic Information System that Integrates Data, Models, and Users for Transportation Applications. *Transportation Research,* Vol. 8C, 2000, pp. 427-444.

Appendix A: Travel Time Prediction Guidelines

Dynamic message sign (DMS) systems provide full public value only when predicted travel time information is integrated into it. This guidebook presents the basic concepts necessary to develop and implement successful travel time prediction methodologies, including an introduction to statistical forecasting and traffic simulation, which are fundamental components of most advanced techniques. The main challenges faced by efficient travel time prediction methods are described next, along with the desirable mathematical model properties and corresponding data requirements. Additionally, two novel travel time prediction techniques developed in the context of TxDOT project 0-5141 are introduced, and their usage explained. The present guideline also describes a procedure to analyze detector coverage in relation to travel time prediction accuracy, and presents a software tool developed towards that end.

Travel Time Types

Travel time can be generally classified into:

- Instantaneous Travel Time (ITT),
- Reconstructed Travel Time (RTT), and
- Forecasted Travel Time (FTT).

Instantaneous Travel Time stands for the travel time of a vehicle traversing a freeway segment at time t if all traffic conditions remain constant until the vehicle exits the freeway segment. ITT generally underestimates travel time at the onset of congestion and overestimates at the dissipation of congestion. In other words, ITT is reliable during the off-peak hours in which traffic conditions remain stable. San Antonio TransGuide's existing algorithm produces Instantaneous Travel Time, and it has been shown to produce reliable travel time estimation only during off-peak hours (Quiroga, 2000.)

Reconstructed Travel Time (RTT) means the travel time realized at time t when a vehicle leaves a freeway segment. An Automatic Vehicle Identification (AVI) travel time measure is a typical type of RTT because a vehicle's actual travel time is not measured until the vehicle passes a toll tag beacon or a Road Side Terminal (RST). While this type of travel time measurement can give precise actual travel times, it has been shown not to be a good measure for online travel time prediction because it introduces non-trivial time-lag in actual travel time detection (Chen and Chien, 2001.) The AVI data is, however, very suitable for calibrating online algorithms in an offline manner because the precisely measured travel times allow the validation of model prediction accurately. The AVI data collected by Houston and TRANSCOM produce RTT.

Forecasting Travel Time (FTT) is defined as the travel time that is actually experienced by drivers who will traverse the freeway segment. This is the most useful information from a driver's perspective, yet the most difficult to produce precisely, particularly during peak-hours in which traffic conditions are less stable. The FTT is certainly the focus of this research because it is intended to be disseminated to the traveling public via Dynamic Message Signs (DMS) or the Internet. The traveling public demands FTT instead of ITT or RTT.

Travel Time Prediction Models: Existing Work and Improvement Directions

Important features of existing travel time prediction systems such as algorithms used, data requirements, and prediction accuracy under different traffic flow conditions were explored during the completion of TxDOT project 0-5141, and summarized in the corresponding report. The fundamental **deficiency found in existing travel time prediction practice** was that it is not predictive in nature. Many agencies utilize simple speed-based models, based on instantaneous data measured at traffic detectors. As a consequence, they do not account for the short-term evolution of traffic dynamics downstream, which leads to simple but non-robust travel time prediction schemes unable to provide satisfactory prediction during recurrent peak hours, non-recurrent congestion, accidents, or special events (Quiroga, 2000). This is problematic, given that the latter circumstances are the ones that would benefit the most from accurate travel time predictions. When motorist are stuck in a long queue without knowing the exact cause, reasonable travel time prediction on DMS becomes critical to help ease road rage, and to assist drivers in the evaluation of alternative routes.

From a **state-of-the-art perspective**, past work on travel time prediction can be primarily classified into statistical models and heuristic models. Statistical methods primarily use regression techniques or time series analysis to estimate travel times based on historical or real-time information. Purely statistical techniques do not perform very well during abnormal traffic conditions, which are typically not frequent in the data sample used for calibration. Heuristic models implement approaches such as Artificial Neural Networks and Kalman filtering for short term prediction of traffic flows. Other methodologies found in the literature include using dynamic traffic assignment and simulation-based techniques in combination with the abovementioned methods in order to better predict travel times. The prediction capability of most of the described approaches was found to be dependent on the data set used. Furthermore, very few of the existing studies utilize analytical traffic flow relationships or simulation based models exist, which is unfortunate. In effect, such models are likely to be better suited for the extrapolation of local conditions (like spot speed data) to that of a link (like travel time on a link).

The models developed for TxDOT project 0-5141 include many characteristics that mitigate some of the undesirable properties of many existing methodologies. Both models take into account the inherent predictive nature of any accurate travel time estimation. Furthermore, they are able to capture traffic dynamics, including congestion formation and dissipation, and shockwave propagation. More importantly, the CTM-based model explicitly represents traffic flow relationships, and is therefore likely to adapt appropriately to changing conditions. Another important advantage of using models that respond to traffic flow relationships is that such models are relatively data-independent. Even though appropriate calibration and validation is required when the model is first implemented, such procedures do not involve major changes in the model structure, which is expected to perform adequately in any location.

Introduction to Statistical Forecasting

Most travel time prediction methodologies involve some sort of statistical forecasting, a procedure in virtue of which the future values of a variable are predicted using analytical formulations. This section presents two of the most popular forecasting techniques, which are fundamental components of the travel time prediction methodologies developed for TxDOT project 0-5141 and described in the following sections. The time series technique is used in the

combined simulation/statistical framework in order to predict traffic counts at entry ramps. Kalman filtering is the statistical method used by the N Curve model to forecast cumulative count curves at the freeway entry points.

Time Series Analysis

Time Series models have been applied in numerous domains to forecast the future parameters characterizing the system based on the trends observed from the past data collected. *Time series* refers to a sequence of data points collected or measured at uniform intervals. Examples of time series include average daily temperature in Austin, monthly profit of IBM, population in the U.S., and total number of accidents in the U.S. for every year (see Figure A1).

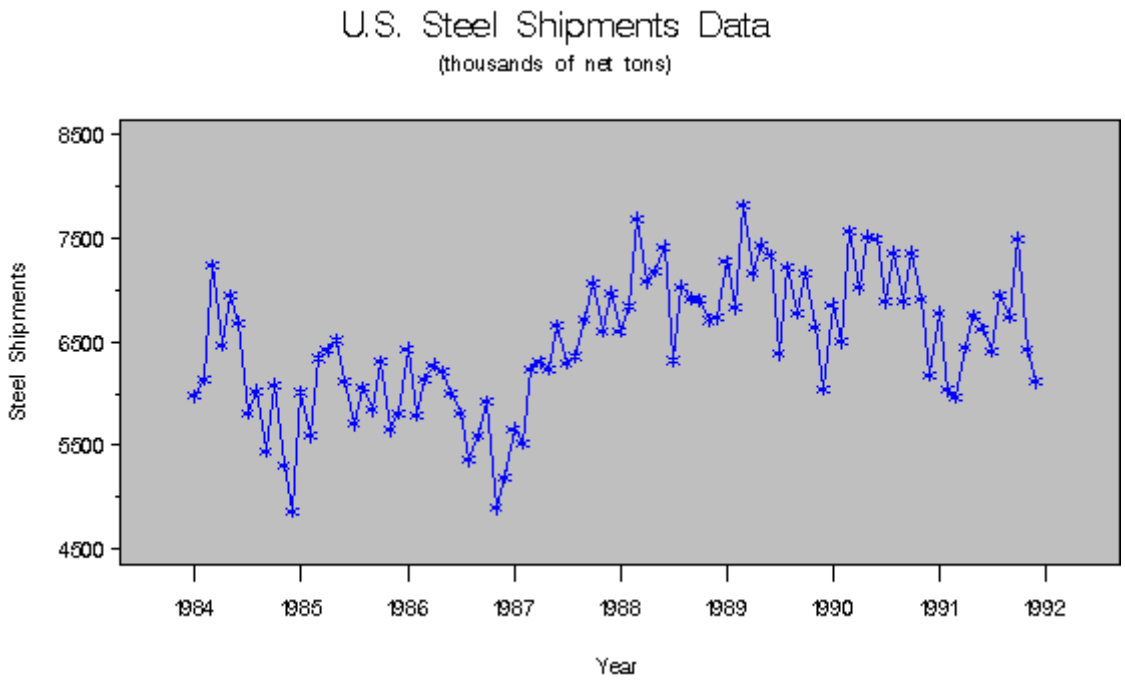

Figure A1: Time series example

Time series models study the trend and variation in past data points and make predictions on future data points before they are measured and occur.

Autoregressive Models

Autoregressive models of order p or AR (p) models predict future data points, X_t^{pred}, based on p most recent data points.

$$X_t^{pred} = c + a_1 X_{t-1} + a_2 X_{t-2} + ... + a_p X_{t-p}$$

The coefficients of the model $c, a_1, ..., a_p$ are estimated from past data points using the method of least squares estimation. The parameters are chosen such that the sum of square errors

between the predicted value and the actual values is minimized. Mathematically the AR (p) model is represented as

$$X_t = c + \sum_{i=1}^{p} a_i X_{t-i} + \varepsilon_t$$

Where ε_t is the error term. ε_t are normally distributed independent and identical random variables.

Moving Average models

Moving Average models of order q or MA(q) models refer to those models where the current data point is a linear function of the error terms of the past q data points.

$$X_t = \sum_{i=1}^{q} \theta_i \varepsilon_{t-i} + \varepsilon_t$$

The coefficients of the model $\theta_i \forall i = 1,...q$ are obtained by method of least squares. Given n data points, and for a particular assumed value for the q coefficients, the error between the actual data value and the predicted data point can be calculated. Method of least squares involves choosing the value of the coefficients $\theta_i \forall i = 1,...q$, which minimizes the sum of square errors between the actual value and the predicted value for the n data points.

Autoregressive Moving Average Models (ARMA)

Autoregressive Moving Average Models (ARMA) combines the auto regression and the moving average concept explained in the previous sections.

$$X_t = c + \sum_{i=1}^{p} a_i X_{t-i} + \sum_{i=1}^{q} \theta_i \varepsilon_{t-i} + \varepsilon_t$$

The above ARMA model is denoted as ARMA (p, q) as there are p auto regressive coefficients and q moving average coefficients which need to be estimated. ARMA models are extremely efficient in modeling stationary time series. Stationary time series involve data points which vary around a constant mean value. Most stationary time series can be decomposed into a deterministic part and a disturbance component. Auto regressive part of the ARMA model captures the deterministic variation of the stationary time series while the moving average part of the ARMA model captures the disturbance component.

Stationary assumption may not be valid in most time series. Many time series data points exhibit a clear upwards or downwards trend. An example is the volume of vehicles entering a ramp during the build up to peak period where an upward trend is expected. In such cases the trend is removed by successive differentiation. Successive differentiation is a process where a new time series is created by taking the difference of successive data points.

$$Y_1 = X_2 - X_1,$$
$$Y_2 = X_3 - X_2,$$
$$Y_3 = X_4 - X_3, \ldots\ldots\ldots Y_{t-1} = X_t - X_{t-1}$$

The order of the differentiation process is the number of times the process is repeated for the entire data set to generate a stationary time series. For example, if the time series $Y_1, Y_2, ..., Y_{t-1}$ is not stationary and still exhibits an upward trend, then the process will have to be repeated again to generate another time series $Z_1, Z_2, ..., Z_{t-2}$. If $Z_1, Z_2, ..., Z_{t-2}$ is found to be stationary then the order of differentiation is 2.

ARIMA models are used to predict time series that exhibit such upward or downward trends. ARIMA models are generally represented as ARIMA (p, q, r) where p represents the number of auto regressive terms, q represents the number of times the time series is differentiated to obtain a stationary series, and r denotes the number of moving average terms. For example, ARIMA (3,1,2) implies that the original time series data points are first differenced once to obtain a stationary time series $X_t, ...,1$. The new data points are then predicted using three auto-regressive terms and two moving average terms.

$$X_t = c + \sum_{i=1}^{3} a_i X_{t-i} + \sum_{i=1}^{2} \theta_i \varepsilon_{t-i} + \varepsilon_t$$

Kalman Filtering

The Kalman filter is essentially a set of mathematical equations that implement a predictor-corrector type estimator that is optimal in the sense that it minimizes the estimated *error* covariance—when some presumed conditions are met. Since the time of its introduction, the Kalman filter has been the subject of extensive research and application, particularly in the area of autonomous or assisted navigation. This is likely due in large part to advances in digital computing that made the use of the filter practical, but also to the relative simplicity and robust nature of the filter itself. Rarely do the conditions necessary for optimality actually exist, and yet the filter apparently works well for many applications in spite of this situation.

Defining the Problem

Discrete time linear systems are often represented in a state variable format given by the equation:

$$x_j = ax_{j-1} + bu_j$$

where the state, x_j, is a scalar, a and b are constants and the input u_j is a scalar; j represents the time variable. Note that many texts don't include the input term (it may be set to zero), and most texts use the variable k to represent time. I have chosen to use j to represent the time variable because we use the variable k for the Kalman filter gain later. The equation above can be represented pictorially as shown in Figure A2 where the block with T in it represents a time delay.

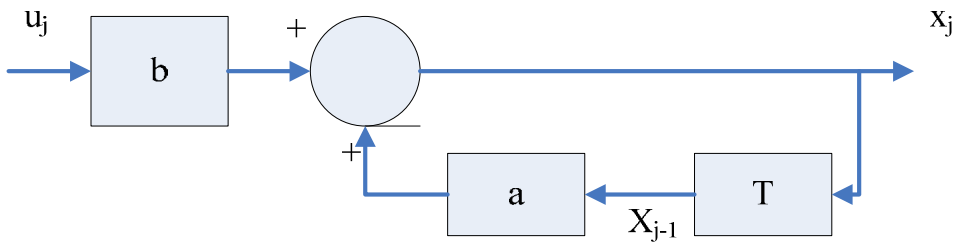

Figure A2: Kalman Filtering Step 1

Now imagine some noise is added to the process such that:

$$x_j = ax_{j-1} + bu_j + w_j$$

The noise, w, is white noise source with zero mean and covariance Q and is uncorrelated with the input. The process can now be represented as shown in Figure A3:

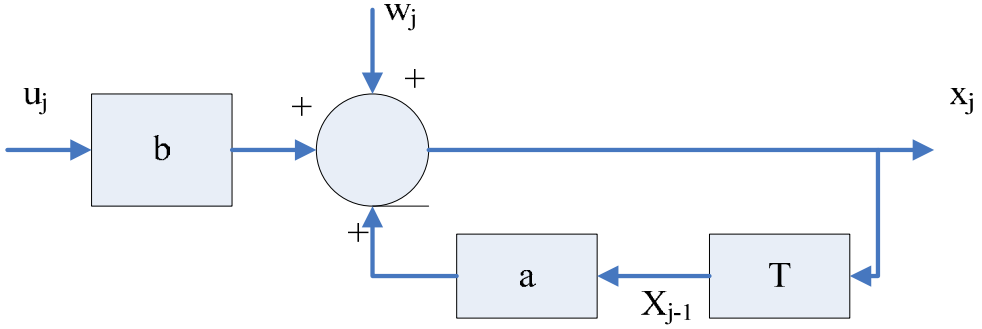

Figure A3: Kalman Filtering Step 2

With Kalman filters we can go one step further. Let us assume that the signal x is measured, and the measured value is z.

$$z_j = hx_j + v_j$$

The measured value z depends on the current value of x, as determined by the gain h. Additionally, the measurement has its own noise, v, associated with it. The noise, v, is white noise source with zero mean and covariance R that is uncorrelated with the input or with the noise w. The two noise sources are independent of each other and independent of the input. See Figure A4.

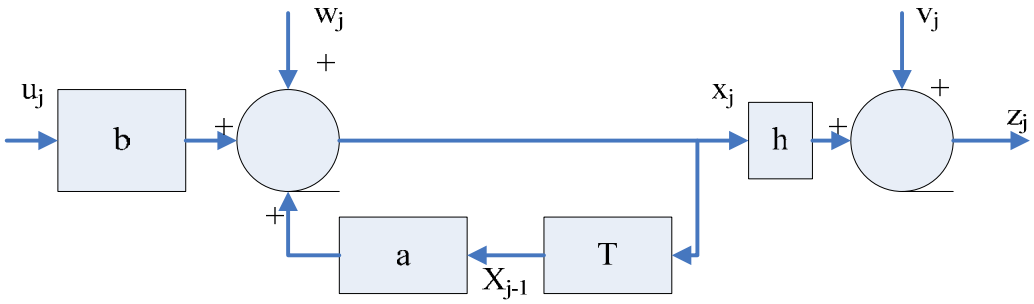

Figure A4: Kalman Filtering Step 3

It seems reasonable to achieve an estimate of the state (and the output) by simply reproducing the system architecture. This simple (and ultimately useless) way to get an estimate of x_j (which we will call $\hat{x}_j$), is diagrammed in Figure A5.

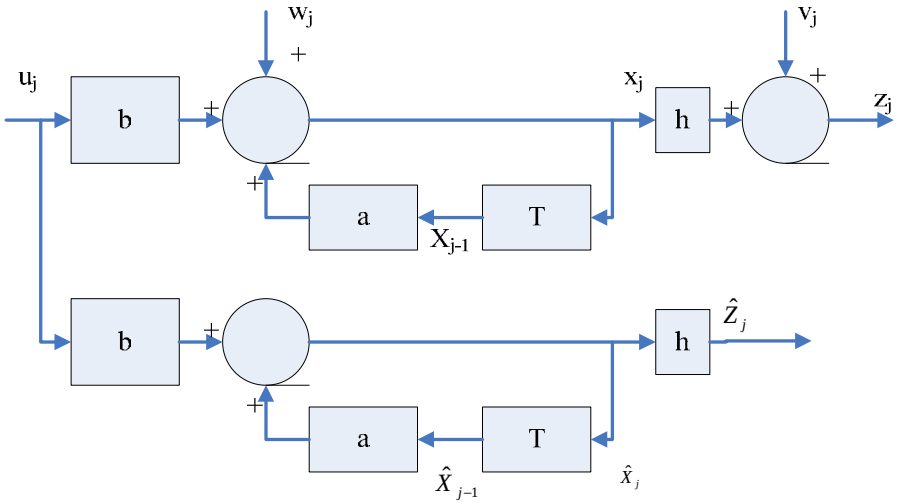

Figure A5: Kalman Filtering Step 4

This approach has two glaring weakness. The first is that there is no correction. If we don't know the quantities a, b or h exactly (or the initial value x_0), the estimate will not track the exact value of x. Secondly, we don't compensate for the addition of the noise sources (w and v). An improved setup that takes care of both of these problems is shown in Figure A6.

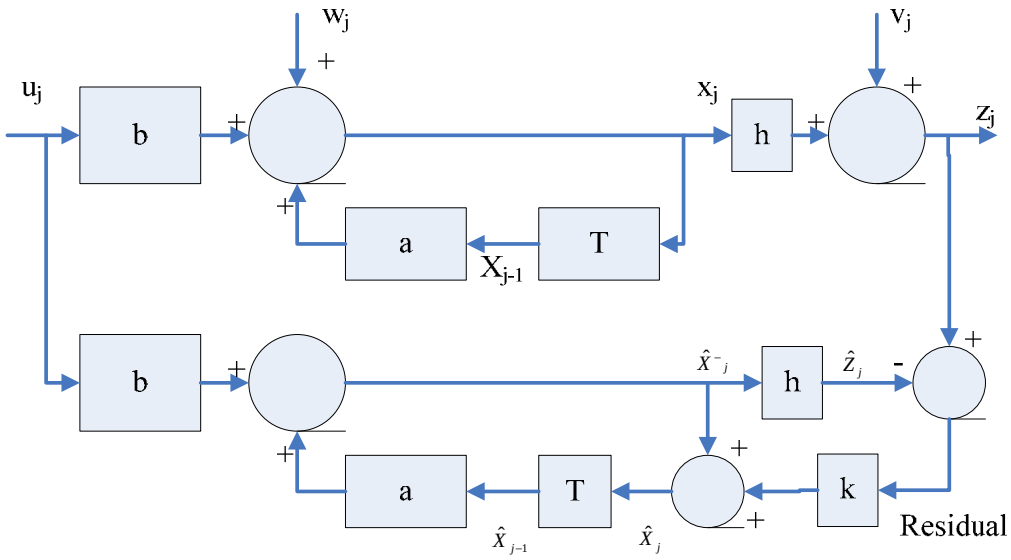

Figure A6: Kalman Filtering Step 5

This figure is much like the previous one. The first difference noted is that the original estimate of x_j is now called $\hat{x}^-_j$; we will refer to this as the a priori estimate.

$$\hat{x}^-_j = a\hat{x}_{j-1} + bu_j$$

We use this a priori estimate to predict an estimate for the output, $\hat{z}_j$; the difference between this estimated output and the actual output is called the residual, or innovation.

$$\text{Residual} = z_j - \hat{z}_j = z_j - h\hat{x}^-_j$$

If the residual is small, it generally means we have a good estimate; if it is large the estimate is not so good. We can use this information to refine our estimate of x_j; we call this new estimate the a posteriori estimate, $\hat{x}_j$. If the residual is small, so is the correction to the estimate. As the residual grows, so does the correction. The pertinent equation is (from the block diagram):

$$\hat{x}_j = \hat{x}^-_j - k \times (\text{Residual}) = \hat{x}^-_j - k \times (z_j - h\hat{x}^-_j)$$

The only task now is to find the quantity k that is used to refine our estimate, and it is this process that is at the heart of Kalman filtering. We are trying to find an optimal estimator, and thus far we are only optimizing the value for the gain, k. We have assumed that a copy of the original system (i.e., the gains a, b and h arranged as shown) should be used to form the estimator. This begs the question: "Is the estimator as developed above optimal?" In other words, should we simply copy the original system in order to estimate the state, or is there perhaps a better way? The answer is that the estimator, as shown above, is the optimal linear estimator that can be developed.

Finding K (The Kalman Filter Gain)

To begin, let us define the errors of our estimate. There will be two errors, an a priori error, e^{-}_{j}, and an a posteriori error, e_j. Each one is defined as the difference between the actual value of x_j and the estimate (either a priori or a posteriori).

$$e_j^- = x_j - \hat{x}_j^-$$

$$e_j = x_j - \hat{x}_j$$

Associated with each of these errors is a mean squared error, or variance:

$$p_j^- = \{(e_j^-)^2\}$$

$$p_j = \{(e_j)^2\}$$

where the operator E**{}** represents the expected, or average, value. These definitions will be used in the calculation of the quantity k.

A Kalman filter minimizes the a posteriori variance, p_j, by suitably choosing the value of k. We start by substituting equation above.

$$p_j = \{(x_j - \hat{x}_j)^2\}$$

$$p_j = \{(x_j - \hat{x}_j^- + k \times (z_j - h\hat{x}_j^-))^2\}$$

To find the value of k that minimizes the variance, we differentiate this expression with respect to k and set the derivative to zero. Be patient here; the expression gets much messier before it becomes simple.

$$\frac{\partial p_j}{\partial k} = 0 = \frac{\partial \mathrm{E}\{(x_j - \hat{x}_j^- + k \times (z_j - h\hat{x}_j^-))^2\}}{\partial k}$$

$$= 2\mathrm{E}\{(x_j - \hat{x}_j^- + k \times (z_j - h\hat{x}_j^-))(z_j - h\hat{x}_j^-)\}$$

$$= 2\mathrm{E}\{x_j z_j - \hat{x}_j^- z_j + k z_j^{\,2} - kh\hat{x}_j^- z_j - h\hat{x}_j^- x_j + h(\hat{x}_j^-)^2 - kh\hat{x}_j^- z_j + kh^2(\hat{x}_j^-)^2\}$$

We take this last expression and use it to solve for k.

$$k = \frac{\mathrm{E}\{x_j z_j - \hat{x}_j^- z_j - h\hat{x}_j^- x_j + h(\hat{x}_j^-)^2\}}{\mathrm{E}\{z_j^{\,2} - 2h\hat{x}_j^- z_j + h^2(\hat{x}_j^-)^2\}}$$

This expression is still quite complicated. To simplify it, we will consider the numerator and the denominator separately. We start with the numerator, and substitute for z_j.

$$\begin{aligned}\text{numerator} &= \mathrm{E}\{x_j z_j - \hat{x}_j^- z_j - h\hat{x}_j^- x_j + h(\hat{x}_j^-)^2\} \\ &= \mathrm{E}\{x_j(hx_j + v_j) - \hat{x}_j^-(hx_j + v_j) - h\hat{x}_j^- x_j + h(\hat{x}_j^-)^2\} \\ &= \mathrm{E}\{h(x_j)^2 + x_j v_j - hx_j\hat{x}_j^- - v_j\hat{x}_j^- - h\hat{x}_j^- x_j + h(\hat{x}_j^-)^2\} \\ &= \mathrm{E}\{h(x_j)^2 - 2hx_j\hat{x}_j^- + h(\hat{x}_j^-)^2 + (x_j - \hat{x}_j^-)v_j\}\end{aligned}$$

The measurement noise, v, is uncorrelated to either the input or the a priori estimate of x, so:

$$\mathrm{E}\{(x_j - \hat{x}_j^-)v_j\} = \mathrm{E}\{e_j^- v_j\} = 0$$

This simplifies the expression for the numerator.

$$\begin{aligned}\text{numerator} &= \mathrm{E}\{h(x_j)^2 - 2hx_j\hat{x}_j^- + h(\hat{x}_j^-)^2\} \\ &= h\mathrm{E}\{(x_j - \hat{x}_j^-)^2\} = h\mathrm{E}\{(e_j^-)^2\} \\ &= hp_j^-\end{aligned}$$

Now, in the same way, consider the denominator.

$$\begin{aligned}\text{denominator} &= \mathrm{E}\{z_j{}^2 - 2h\hat{x}_j^- z_j h^2(\hat{x}_j^-)^2\} \\ &= \mathrm{E}\{(hx_j + v_j)^2 - 2h\hat{x}_j^-(hx_j + v_j) + h^2(\hat{x}_j^-)^2\} \\ &= \mathrm{E}\{h^2 x_j{}^2 + 2hx_j v_j + v_j{}^2 - 2h^2\hat{x}_j^- x_j - 2h\hat{x}_j^- v_j + h^2(\hat{x}_j^-)^2\} \\ &= \mathrm{E}\{h^2 x_j{}^2 - 2h^2 x_j\hat{x}_j^- + h^2(\hat{x}_j^-)^2 + v_j{}^2 + 2h(x_j - \hat{x}_j^-)v_j\}\end{aligned}$$

Again, we can use the orthogonality condition from equation above to set the last term to zero, so:

$$\begin{aligned}\text{denominator} &= \mathrm{E}\{h^2 x_j{}^2 - 2h^2 x_j\hat{x}_j^- + h^2(\hat{x}_j^-)^2 + v_j{}^2\} \\ &= h^2\mathrm{E}\{x_j{}^2 - 2x_j\hat{x}_j^- + (\hat{x}_j^-)^2\} + \mathrm{E}\{v_j{}^2\} \\ &= h^2 p_j^- + R\end{aligned}$$

where we used the simplification from equation above for the first term in the expression, and using the definition of the measurement noise for the second term.

Using the expression for numerator and denominator, we finally get a simple expression for k:

$$k = \frac{hp_j^-}{h^2 p_j^- + R}$$

However, there is still a problem because this expression needs a value for the a priori covariance, which in turn requires knowledge of the system variable x_j. Therefore our next task will be to come up with an estimate for the a priori covariance.

Finding the Priori Covariance

Finding the a priori covariance is straightforward, starting with its definition.

$$\begin{aligned}p_j^- &= \mathrm{E}\{(x_j - \hat{x}_j^-)^2\} \\ &= \mathrm{E}\{(ax_{j-1} + bu_j + w_j - (a\hat{x}_{j-1} + bu_j))^2\} \\ &= \mathrm{E}\{(a(x_{j-1} - \hat{x}_{j-1}) + w_j)^2\} \\ &= \mathrm{E}\{a^2(x_{j-1} - \hat{x}_{j-1})^2 + 2aw_j(x_{j-1} - \hat{x}_{j-1}) + w_j{}^2)\}\end{aligned}$$

The middle term drops out as before because the process noise is uncorrelated with previous values of the either the state or it's a priori estimate.

$$\mathrm{E}\{2aw_j(x_{j-1}-\hat{x}_{j-1})\} = 2a\mathrm{E}\{w_j e_{j-1}\} = 0$$

So

$$\begin{aligned} p_j^- &= \mathrm{E}\{a^2(x_{j-1}-\hat{x}_{j-1})^2 + w_j{}^2)\} \\ &= a^2 p_{j-1}^- + Q \end{aligned}$$

We are still not finished, however, because we need an expression for p_j, the a posteriori estimate.

Finding the Posteriori Covariance

As with the a priori covariance, we find the a posteriori covariance by starting with its definition.

$$\begin{aligned} p_j &= \mathrm{E}\{(x_j-\hat{x}_j)^2\} \\ &= \mathrm{E}\{(x_j-(\hat{x}_j^- + k\times(z_j - h\hat{x}_j^-)))^2\} \\ &= \mathrm{E}\{(x_j-(\hat{x}_j^- + kz_j - kh\hat{x}_j^-))^2\} \\ &= \mathrm{E}\{(x_j-(\hat{x}_j^- - kh\hat{x}_j^- + k(hx_j + v_j)))^2\} \\ &= \mathrm{E}\{((x_j-\hat{x}_j^-)(1-kh) - kv_j)^2\} \\ &= \mathrm{E}\{(x_j-\hat{x}_j^-)^2(1-kh)^2 - 2kv_j(x_j-\hat{x}_j^-)(1-kh) + k^2v_j^2\} \end{aligned}$$

The middle term drops out as before because the measurement noise is uncorrelated with the current values of the either the state or it's a priori estimate.

$$\begin{aligned} \mathrm{E}\{2kv_j(x_j-\hat{x}_j^-)(1-kh)\} &= 2k(1-kh)\mathrm{E}\{v_j(x_j-\hat{x}_j^-)\} \\ &= 2k(1-kh)\mathrm{E}\{v_j e_j^-\} = 0 \end{aligned}$$

So

$$\begin{aligned} p_j &= \mathrm{E}\{(x_j-\hat{x}_j^-)^2(1-kh)^2 + k^2v_j^2\} \\ &= (1-kh)^2\,\mathrm{E}\{(x_j-\hat{x}_j^-)^2\} + k^2\mathrm{E}\{v_j^2\} \\ &= (1-kh)^2\,p_j^- + k^2R \end{aligned}$$

We can simplify this by using our previous definition for k

$$R = \frac{p_j^-(h - h^2k)}{k}$$

$$\begin{aligned} p_j &= (1-kh)^2 p_j^- + k^2\,\frac{p_j^-(h-h^2k)}{k} \\ &= p_j^-(1-2hk+k^2h^2+kh-k^2h^2) \\ &= p_j^-(1-hk) \end{aligned}$$

Kalman Filter Procedure

Any Kalman filter operation begins with a system description consisting of gains *a*, *b* and *h*. The state is *x*, the input to the system is *u*, and the output is *z*. The time index is given by *j*.

$$x_j = ax_{j-1} + bu_j + w_j$$

$$z_j = hx_j + v_j$$

The process has two steps, a predictor step (which calculates the next estimate of the state based only on past measurements of the output), and a corrector step (which uses the current value of the estimate to refine the result given by the predictor step).

Predictor Step:

We form the a priori state estimate based on the previous estimate of the state and the current value of the input.

$$\hat{x}_j^- = a\hat{x}_{j-1} + bu_j$$

We can now calculate the a priori covariance.

$$p_j^- = a^2 p_{j-1} + Q$$

Note that these two equations use previous values of the a posteriori state estimate and covariance. Therefore the first iteration of a Kalman filter requires estimates (which are often just guesses) of these two variables. The exact estimate is often not important as the values converge towards the correct value over time; a bad initial estimate just takes greater number of iterations to converge.

Corrector Step:

To correct the a priori estimate, we need the Kalman filter gain, k.

$$k_j = \frac{hp_j^-}{h^2 p_j^- + R}$$

This gain is used to refine (correct) the a priori estimate to give us the a posteriori estimates.

$$\hat{x}_j = \hat{x}_j^- + k_j(z_j - h\hat{x}_j^-)$$

We can now calculate the a posteriori covariance.

$$p_j = p_j^-(1 - hk_j)$$

Introduction to Traffic Simulation Models

Depending on the scope and the resolution of the analysis traffic flow models can be classified into microscopic, macroscopic, and mesoscopic models. Microscopic simulators achieve the finest resolution by modeling the movement and behavior of individual vehicles. Various factors like lane changing behavior, gap acceptance, and individual driver characteristics such as compliance can be modeled. Even though microscopic models achieve a high degree of realism, microscopic models are computationally intensive and cannot be used to model large networks. Moreover, such models involve significant amount of calibration of numerous individual driver characteristics like gap acceptance, which are difficult to obtain. Examples of microscopic simulators include CORSIM, VISSIM, PARAMICS, etc.

Macroscopic simulators, on the other hand, model the behavior of a larger platoon of vehicles over time. For example, macroscopic simulator can be used to model the variation of more aggregate performance measures such as number of vehicles in a link over time. Macroscopic models thus do not attempt to capture the impact of various individual driver factors such as lane changing, gap acceptance, or link performance. However, modeling at a higher resolution results in significant computational savings and macroscopic simulators can therefore be used to model large networks. TRANSYT-7F is an example of a macrosimulator.

Mesoscopic simulators are defined as those models that are neither macroscopic nor microscopic. Mesoscopic simulators adopt an intermediate position, attempting to capture some of the detail of a microsimulator while performing some abstraction to allow larger regions to be modeled. These often use efficient traffic propagation procedures such as the cell transmission model (Daganzo, 1994). RouteSim is an example of a mesoscopic simulator.

In this study the objective is to develop a simulation model that, given the input flows, simulates the conditions on the freeway in a computationally efficient manner without losing too much on the realism side. With this in mind a mesoscopic model the cell transmission model developed by Daganzo was chosen. The cell transmission model is easy to code, and captures various freeway dynamics like queue formation/dissipation and shockwave propagation. To give an idea about its computational efficiency, CTM can simulate a 5-mile freeway section for a time period of 15 minutes in less than a second. The next section provides an overview of the cell transmission model used in this work.

Cell Transmission Model

As explained in the previous section, the Cell Transmission Model simulates traffic at a mesoscopic level. Cell Transmission Model converts the freeway network into cells connected by links. Vehicles are contained in cells and are transferred from one cell to another every simulation interval using simple mathematical relationship. In order to use the cell transmission model the freeway section must be converted into a cell network representation. This is done in two stages.

Choosing the Simulation Interval

In the first step, choose an appropriate simulation interval. Choice of the simulation interval has a significant impact on the computational efficiency and the accuracy of the simulation model. High values of the simulation interval, despite causing considerable computational savings, result in considerable loss of accuracy especially during highly congested phases. If the simulation interval is too low, then the computational time increases significantly.

Previous studies have shown that the simulation interval in the range of 4-10 seconds is optimal. For future reference the length of the simulation interval will be denoted as δ.If δ = 4 seconds and the time period of analysis is 10 minutes, then the total number of simulation intervals is 10*60/4 = 150.

Converting Freeway to Cell Representation

The second stage is to convert the freeway network into the representation of the cell transmission model. Before converting, three main parameters of the network must be determined using engineering judgment: (i) Free Flow Speed υ, (ii) Capacity Q, and (iii) Jam Density ρ. Consider a straight freeway section with no ramps of length L. Starting from the upstream node, the freeway section is then divided into cells of uniform length $\upsilon\delta$. All of these cells are connected using links. Thus the total number of cells generated will be equal to $N = \lfloor L/\upsilon\delta \rfloor$ where $\lfloor L/\upsilon\delta \rfloor$ denotes the largest integer less than or equal to $L/\upsilon\delta$. The residual length $L - N\upsilon\delta$ is added to the last cell. Note that all the cells in the network should have a minimum length of $\upsilon\delta$. The cell length can be longer than $\upsilon\delta$. ***The model does not work if any one cell is shorter than*** $\upsilon\delta$**.**

For example consider a freeway section of length $L = 1500$ ft. Let it have a free flow speed of $\upsilon = 80$ mph or 117.33 fps. Consider a simulation interval $\delta = 4$ seconds. Thus the length of each cell is 117.33*4 = 469.33 feet. The freeway is then divided into 2 cells of length 469.33 feet and one cell of length 561.33 feet as shown in Figure A7.

Figure A7: Cell Representation of the freeway network

The cell representation of a straight line freeway segment was presented above. Modeling a freeway segment will involve modeling merging on ramps, diverging off ramps, and intersections. The process of converting sections with ramps will be discussed next.

Merge Segment

A merge segment, depicted in Figure A8, consists of two links 1 and 2 entering a single cell C. Cell C is called a merge cell. Links 1 and 2 are called merge links.

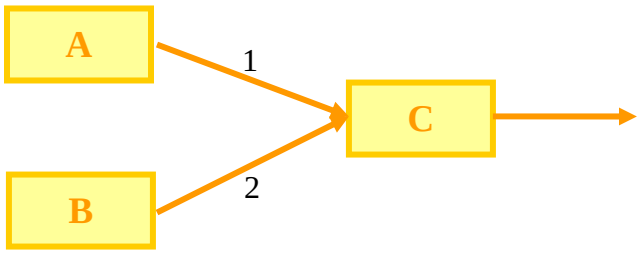

Figure A8: Merge Cell Representation of the freeway network

Diverge Segment

A diverge segment, depicted in Figure A9, consists of two links 1 and 2 diverging from a single cell C. Cell C is called a diverge cell. Links 1 and 2 are called diverge links.

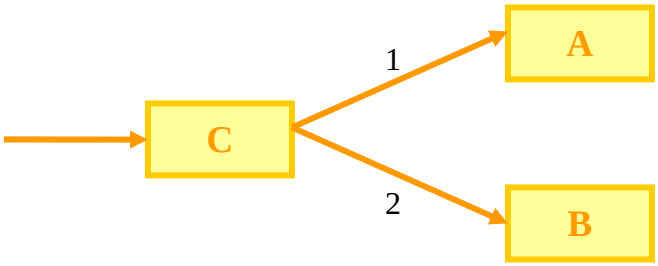

Figure A9: Merge Cell Representation of the freeway network

In the cell representation while modeling intersection, a single link cannot function as a merge link and a diverge link, because of the way flows are determined. Consider the following (Figure A10):

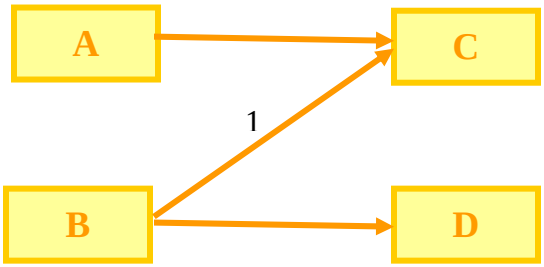

Figure A10: Invalid Representation of the freeway network

In this network, link 1 serves as both as a diverge link and a merge link, which is not allowed. Hence Figure A10 is an incorrect cell representation. The correct cell representation is given in Figure A11 where an extra cell E is introduced to split link 1 into a diverge link and a merge link.

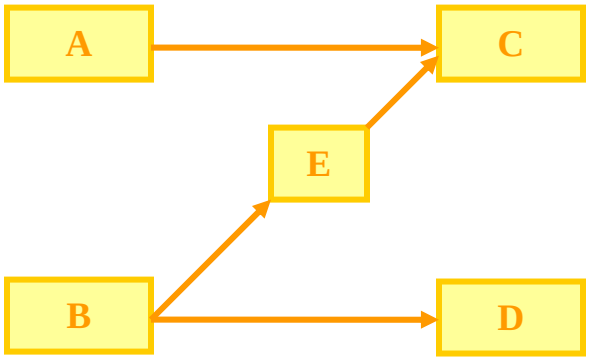

Figure A11: Valid Cell Representation of the freeway network

Thus an overview how to convert a normal freeway segment into a cell network representation. The next section describes the mathematical relationships used to transfer vehicles from one cell to another every simulation interval.

Traffic Flow Relationship

Consider a straight line freeway segment consisting of cells $i-1, i, i+1$ (Figure A12).

Figure A12: Cell Representation of the freeway network

Let $\rho_i(k+1)$ denote the density of cell i (expressed in vehicles per mile) in simulation interval $k+1$. In the CTM model $\rho_i(k+1)$ is determined from $\rho_i(k)$ using the following relationship:

$$\rho_i(k+1) = \rho_i(k) + \frac{\delta}{l_i}(q_{i-1,i}(k) - q_{i,i+1}(k)) \qquad [1]$$

Where δ is the simulation interval and l_i stands for the length of cell *i*. The flows leaving from and arriving to cell *i* during time interval *k* are given by $q_{i,i+1}(k)$ and $q_{i-1,i}(k)$ respectively. These are obtained as the minimum of two quantities: the highest flow that can be supplied by the upstream cell ($S_{i-1}(k)$), and the maximum flow that the downstream cell can receive ($R_i(k)$). Equations 2 and 3 describe the computation of these values for every cell:

$$S_{i-1}(k) = \min(\upsilon\rho_{i-1}(k), Q_{M,i-1}) \qquad [2]$$

$$R_i(k) = \min(Q_{M,i}, \omega^*(\rho_J - \rho_i(k))) \qquad [3]$$

where υ refers to the free flow speed, $Q_{M,i-1}$ is the capacity (in vehicles per hour) of cell *i-1*, and ρ_J is the jam density. Equations [1] to [3] constitute the core of the Cell Transmission Model for a straight line freeway segment. The next section describes how the cell transmission model can be applied for a merge and diverge segment.

Consider the following merge network (Figure A13):

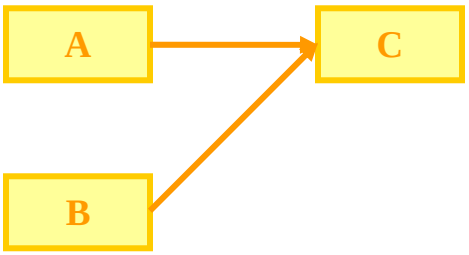

Figure A13: Cell Representation of the freeway network

If the amount of flow that can be received by cell C $R_C(k)$ is greater than the combined flow that can sent from link A , $S_A(k)$ and B, $S_{Bi}(k)$, then all the flow flows into cell C. In mathematical notation, the above statement can be represented as

$$\begin{aligned} q_{A,C}(k) &= S_A(k) \\ q_{B,C}(k) &= S_B(k) \end{aligned} \qquad \text{If } \{ R_C(k) > S_A(k) + S_{Bi}(k) \} \qquad [4]$$

If the amount of flow that can be received by cell C $R_C(k)$ is lesser than the combined flow that can sent from link A , $S_A(k)$ and B, $S_B(k)$, then the following mathematical relationship is used to determine $q_{A,C}(k)$ and $q_{B,C}(k)$.

$$q_{A,C}(k) = \text{Median } \{ S_A(k), R_C(k) - S_B(k), \alpha * R_C(k) \} \qquad [5]$$

$$q_{B,C}(k) = \text{Median } \{ S_B(k), R_C(k) - S_A(k), (1-\alpha) * R_C(k) \} \qquad [6]$$

Where α is an exogenous parameter taking values between 0 and 1, which indicates the priority mainline flows have over ramp flows. Once the flows are known, the densities can be calculated from equation 1.

Consider the following diverge cell network (Figure A14).

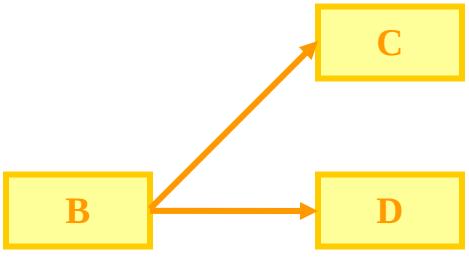

Figure A14: Cell Representation of the freeway network

For calculating the flows for diverge cell network, the splits at cell B, β have to be pre-specified. Calculate

$$\gamma = \min\{S_B(k), \frac{R_D(k)}{\beta}\} \quad [7]$$

$$\eta = \min\{\gamma, \frac{R_C(k)}{1-\beta}\} \quad [8]$$

$$q_{B,D}(k) = \beta * \eta \quad [9]$$

$$q_{B,C}(k) = (1-\beta) * \eta \quad [10]$$

Once the flows are known, the densities can be calculated from equation 1.

Modeling input and output cells

Output cells are modeled using a gate cell of infinite jam density and infinite capacity. Input cells are modeled using two cells as shown in Figure A15. Cell 1 is the gate cell. Cell 2 serves to control the input to the simulator in accordance with the actual input flow.

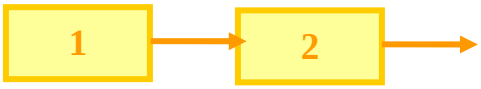

Figure A15: Input output Cell Representation

If the observed input flow at time k is $r(k)$ then set $S_1(k) = r(k)$. Cell 2 is assumed to have infinite capacity and infinite jam density. Therefore, $R_2(k)$ will always be equal to infinity. This implies that $q_{12}(k) = \min\{R_2(k), S_1(k)\}$ will be equal to $r(k)$, which is the input flow into the system.

Determining the travel time of a freeway segment

This section describes the procedure for estimating the travel time on a segment using the cell transmission model. The concept is illustrated using a single cell example and then extended to longer sections. Consider the cell A shown in Figure A16:

Figure A16: Freeway Cell

The objective is to determine the travel time experienced by a vehicle entering cell A in simulation interval k. Let $q_{in}(k)$ represent the flow into the cell A in interval k. Let $\rho_A(k)$ denote the density of vehicles in cell A in interval k. The travel times are determined by comparing cumulative inflows and outflows. For example, to determine the travel time

experienced by $q_{in}(k)$, keep track of the cumulative outflows starting from interval k. Determine the interval K when

$$\rho_A(k)l_A + q_{in}(k)\delta < \sum_{t=k}^{K} q_{out}(t)\delta$$

The travel time experienced by the vehicles entering cell A at time k is $(K-k)\delta$.

The above concept can be extended to a sequence of cells to determine the travel time of a segment, as shown in Figure A17.

Figure A17: Freeway Cell Network

The travel time of the segment consisting of cells A, B, and C at interval k can be estimated by determining the interval K when

$$\rho_A(k)l_A + \rho_B(k)l_B + \rho_C(k)l_C + q_{in}(k)\delta < \sum_{t=k}^{K} q_{out}(t)\delta$$

The travel time experienced by the vehicles entering cell A on the segment consisting of cells A, B, and C at time k is $(K-k)\delta$. The travel time for longer segments can be calculated by using the above procedure for smaller individual segments and cascading. This process is explained with an example in Figure A18.

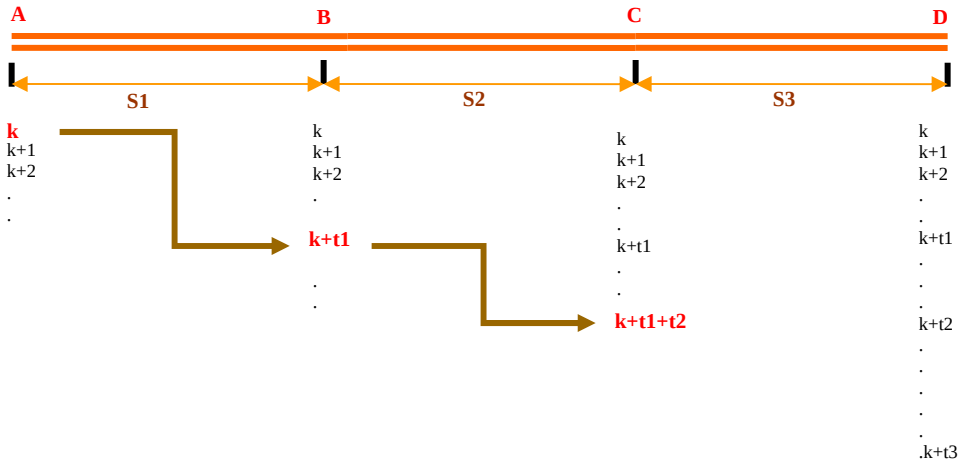

Figure A18: Cascading Travel times on freeways

Consider a freeway section consisting of three segments: *S1, S2, and S3*. The objective is to estimate the travel time experienced by a user departing A in simulation interval k. Find the travel time in section *S1, S2 and S3* for all simulation intervals by comparing the cumulative inputs and inputs as outlined above. Denote the section travel times as

$t_1(t), t_2(t), t_3(t) \forall t = k, k+1, \ldots..$ Now the instantaneous travel time in the freeway segment at simulation interval k, $\tau(k)$ is calculated as:

$$\tau(k) = t_1(k) + t_2(k) + t_3(k)$$

Now the issue with instantaneous travel times is that a person who departs A at simulation interval k will not experience $\tau(k)$. This is because when the vehicle reaches B at time $k + t_1(k)$, the traffic conditions on section *S2* would have changed. So the travel time experienced by the vehicle on section *S2* would be $t_2(k + t_1(k))$, which is different from $t_2(k)$. Therefore, as the experienced travel time does not correspond to the calculated travel times, the information provided may not be useful. The actual experienced travel time must be estimated as

$$\tau(k) = t_1(k) + t_2(k + t_1(k)) + t_3(k + t_1(k) + t_2(k + t_1(k)))$$

Consider a vehicle departing from A at 8:00 A.M. Let the travel time on the three sections be 5 minutes at 8:00. The instantaneous travel time will be reported as 15 minutes. However, when the person reaches B, the conditions on freeway section *S2* can change so that the travel time on *S2* is no longer 5 minutes, but 8 minutes. Thus the person will reach C at 8:13 instead of the 8:10 reported. If the travel time on *S3* is 10 minutes at 8:13 due to congestion build up, the experienced travel time will be 23 minutes, which is different from the instantaneous travel time of 15 minutes.

Therefore, to correctly estimate the user's experienced travel time, the traffic conditions on section *S2* at 8:05 and the conditions on *S3* at 8:13 must be predicted at 8:00 A.M itself. Therefore, accurate travel time estimation entails prediction of future conditions.

Overview of the simulation process

Step 1: Obtain the free flow speed υ, Capacity Q and Jam density ρ_J of the freeway segment

Step 2: Determine the simulation interval δ

Step 3: Convert the freeway segment into the cell representation.

Step 4: For every simulation interval k $(1,...,T)$

For every cell i Calculate $S_i(k), R_i(k)$

$$S_i(k) = \min(\upsilon \rho_i(k), Q_{M,i})$$

$$R_i(k) = \min(Q_{M,i}, \omega^*(\rho_J - \rho_i(k)))$$

For every link $(i, i+1)$ Calculate $q_{i,i+1}(k)$ using

$q_{i,i+1}(k) = \min\{S_i(k), R_i(k)\}$ if $(i, i+1)$ is a straight link

Formulas [4],[5] and [6] for a merge link

Formulas [7],[8],[9] and [10] for a diverge link

For every cell i Calculate

$$\rho_i(k+1) = \rho_i(k) + \frac{\delta}{l_i}(q_{in,i}(k) - q_{out,i}(k))$$

Data Requirements

This chapter describes the different types of data required to calibrate, validate, and ultimately implement travel time prediction models in general, and the methodologies developed for this work in particular. Additionally, the final section provides recommendations for the pre-processing of input data, namely traffic counts, in real-time. The latter are necessary given that the measurements provided by detectors may contain errors which, if neglected, could have detrimental effects on the performance of some travel time prediction models.

Data types

Two main types of data are required, alone or in combination, in order to calibrate and validate travel time prediction models: traffic data and travel time measurements.

Traffic data, in the context of this work***,*** consists of vehicle counts, as provided by detectors deployed on the field. The data may be available at different aggregation levels, and ideally the most disaggregated version compatible with a real-time deployment should be utilized. Traffic data is necessary for all the stages of model development and implementation.

Even though vehicle counts are the most readily available (and reliable) form of traffic data, some travel time predictions models rely on different traffic data, such as average speed or occupancy. Moreover, given the increasing popularity of AVI systems, some models have been developed that utilize the sample of existing travel times measured by the automatic system to predict the conditions throughout the freeway.

Travel time measurements were used in this research to validate the model performance, by comparing them to the travel time predictions generated using the models. They are also necessary to fine-tune model parameters, and when the model is utilized, to determine optimal deployment strategies.

In addition to the main data types, the models need to be tailored to reflect the ***geometric characteristics*** of the segment under analysis. These include the number of lanes throughout the segment, number and position of entry/exit ramps and the corresponding split ratios, and the number and location of traffic detectors. Experience-based knowledge regarding prevailing driving speeds, recurrent congestion, and general driver behavior may be useful during the model calibration and validation process.

Types of Data Sources

When the models presented in this report are used to generate online travel time predictions, the only valid source of data is given by traffic detectors, which provide the counts used as the sole input at run-time. However, the validation and calibration stages demand additional data, including historical traffic counts at the segment under analysis and the corresponding real travel time measurements. Ideally both data sets should be collected on the site where the model is going to be implemented (***field data***). Historical traffic counts can be obtained from the same detectors that are going to be used for the predictions. At least one or two peak periods, including congestion formation and dissipation, should be available. The most straightforward source of real travel time measurements is the utilization of vehicles equipped with GPS systems. Alternatively, they may be derived analyzing AVI data, or applying vehicle re-identification algorithms (Coifman, 1998, Sun et al., 1999). The travel time collection handbook (FHWA, 1998) describes and compares these and other techniques. Real travel time

measurements are required throughout the period used for calibration/validation, and a relatively dense data set is necessary to obtain accurate results.

Field data may not be available in the amount and/or level of aggregation demanded by the validation/calibration processes. In such a scenario, traffic counts and real-time measurements may be simulated using commercial microsimulators, after modeling the segment under study utilizing the corresponding geometric data. In order to generate traffic counts and travel times, it is also necessary to feed approximate vehicle counts at the main entry points to the segment under study, which can be collected using existing traffic detectors, or estimated based on other sources of historical data. Simulated data has a number of advantages: it's error-free, can be used to test the model under different scenarios, and is available at any desired aggregation level. Furthermore, when optimal sensor location is analyzed, only the use of simulated data allows considering arbitrary deployment patterns. However, field data reflects the actual behavior of local drivers under real conditions, and should be preferred over simulated data when available. See Figure A19.

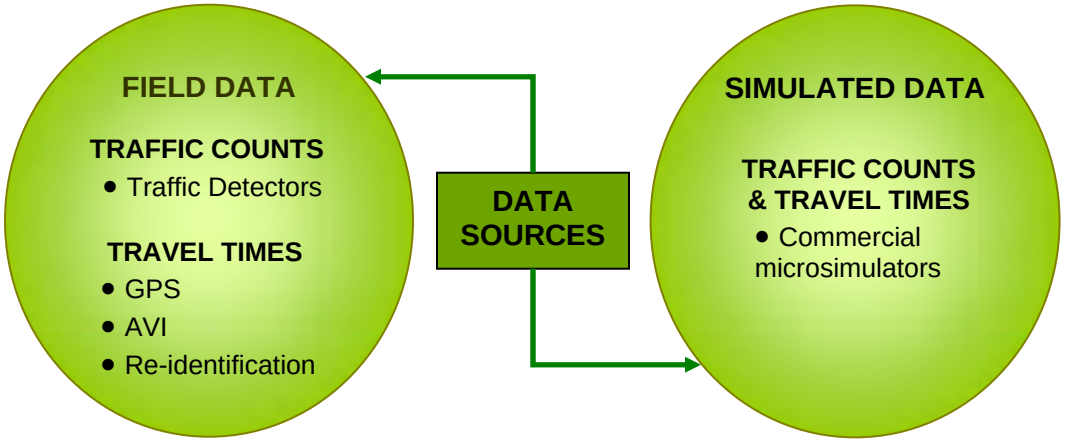

Figure A19: Travel-time prediction data sources

Data Filtering

One of the salient advantages of the models presented in this report is that they require only traffic count data at deployment time. Such data is typically readily available and fairly reliable. Nevertheless, as most automatic data collection devices, traffic detectors may produce flawed data. This can clearly affect models performance, as it was observed during the experimental model deployment conducted at El Paso TMC (see Chapter 5). In order to mitigate the negative impacts of data errors, procedures are needed to determine whether detectors are working, and if their measurements are accurate. Such processes are highly dependent upon the characteristics of the detectors used at each TMC, the prevailing communication protocols, and the structure of the implemented travel time prediction models. There has been a considerable number of works dealing with the issue of assessing the quality of traffic detector data. The following paragraphs will briefly describe the basic principles underlying existing research, and suggest possible references for an eventual deployment of the proposed models into a TMC.

The data filtering process involves two basic operations, depicted in Figure A20.

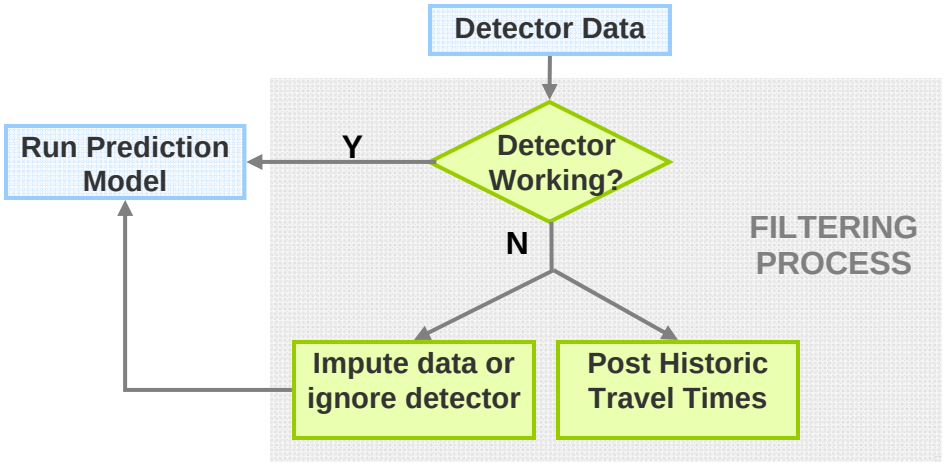

Figure A20: Simplified flow chart of the data filtering process

Several approaches can be taken to identify whether a detector is working properly. The simplest ones, which are often used given the time constraints imposed by the real-time nature of the travel time prediction procedure, include detecting implausible values of speeds, counts, or their combinations (Dahlgren, 2002). Coifman (2004) presents eight event-based validation tests which use microscopic data to assess the working status of single and dual loop detectors. The techniques range from comparing on/off time at each loop of a dual detector, to analyzing the succession of flow regimes observed at a particular detector. Other approaches rely on simple comparisons of the measurements retrieved by nearby detectors, or on a time-wise analysis of the observations of a single detector (Chen and May, 1987). Some alternative approaches to validate the performance of remote traffic microwave sensors (RTMS) are introduced in Coifman, 2006. Historical information regarding recurrent congestion patterns and typical traffic conditions can be also utilized to verify the likelihood of a specific measurement at a particular time and location, determining the need for further analysis.

Once malfunctioning detectors are identified, proper action needs to be taken in order to generate acceptable travel time predictions. The specific procedure will depend on the characteristics of the travel time prediction model under usage, data availability, and time constraints. One possible option involves imputing the missing measurements, either based on the values observed at nearby detectors, or by utilizing previous observations at the malfunctioning detector, provided that they are error free (for example, Al-Deek and Chandra, 2004). Additionally, average historical values may be utilized. Some of these techniques are explained in the technical report describing the data processing for California's PeMS (Urban Crossroads Inc., 2006), and in Nguyen and Scherer (2003). When imputation is not possible, a different type of action needs to be taken. Some models, such as the CTM-based one, can be run ignoring the malfunctioning detectors, with little effect on the prediction accuracy. For these models, each TMC should analyze the advantages, in terms of model accuracy, data requirements, and computational efficiency, of imputing data versus simply disregarding the detectors out of order at run time. Some travel time prediction models may not be able to run if error-free traffic measurements are not available for every detector. Under such scenario, traffic

managers may choose to either temporarily display historic travel times for the affected sections, or not to display any value at all.

Data Products

During the development of TxDOT project 0-5141, a database containing traffic data, as measured by TransVista's detectors, was generated. TransVista is the El Paso Traffic Management System. It consists of a number of traffic detectors, changeable signs, and data processing/storing infrastructure, utilized to mange local traffic. Historic traffic counts for model validation and calibration were obtained from 77 of this system's detectors spanning 37 miles of highway, 11 of which correspond to El Paso Border Highway. ***Detector data*** was processed and organized into a PostgreSQL database, spanning several days during the month of June. Such database, along with the corresponding usage instruction and complementary visualization software were delivered as product P5. The software includes a tool to plot different data components, including traffic volumes and speed. Additionally, a Graphical User Interface (GUI) was provided, which can be used to retrieve the data collected by each detector in real time, provided that the database is updated accordingly.

The traffic database is complemented by a real travel time database, constructed utilizing the information collected by a vehicle equipped with GPS. The car traveled through a 10 mile section of the El Paso Border Highway on which TransVista traffic detectors were deployed. Between June 4 and June 29, real travel time data was collected during the morning peak hour (6:00–8:00 AM), by completing two or three trips on each direction.

The GPS device recorded the coordinates of the vehicle every 5–10 seconds, and a C++ code was used to process the resulting data. The program read the raw GPS file, and generated travel times between roadway sections by subtracting the corresponding time stamps. By comparing the coordinates of each of the points retrieved by the GPS to the coordinates of the desired reference points on the highway (corresponding to the position of TransVista traffic detectors), the code was able to select the appropriate data pairs for the travel time computation. Each two-way trip was recorded into the same file, and the program automatically identified the point at which the vehicle switched directions. A table reflecting the processed data was provided with the deliverables as part of product P5.

Combined Statistical/Simulation Framework (CTM-based model)

This model implements a simulation-based framework for the point-to-point freeway travel time prediction in the short term (3 to 7 minutes). The proposed methodology relies on traffic counts, provided by the simplest types of detectors, as its primary input, which makes it widely applicable.

While most of the research previously conducted in the area use statistical or heuristic techniques to predict future travel time as a function of current and historical traffic flows (Chien et al., 2003), the present work derives travel time predictions from a calibrated traffic flow model fed by forecasted demands. The two-stage travel time prediction process introduced in this framework involves the use of a time series model to forecast the inflows into the traffic corridor, and of a Cell Transmission Model (CTM) to simulate the flow of these vehicles through the network. By doing this, the proposed approach takes advantage of the best characteristics of statistical techniques and traffic flow theory relationships. The utilization of a CTM (Daganzo, 1994, Daganzo, 1995*)* ensures that queue formation/dissipation, link spillovers, shockwave

propagation, and other elements of traffic dynamics are accounted for. Additionally, the presented framework is computationally efficient, allowing for online updates of the freeway travel time estimates by means of a rolling-horizon approach (Wagner, 1977).

The model is implemented via ***computer software,*** utilizing a program coded in C++, which combines an open source statistical analysis tool (R) and simulation algorithm written at UT Austin. If users preferred to implement a different software and/or statistical forecasting technique, such as Kalman filtering, they may do so, following the suggestions presented in the user guide provided along with the software. The same also describes the data input process, which encompasses the preparation of network and link data files, and the introduction of additional information via a graphical user interface (GUI).

At each specific location, this model needs to be ***calibrated,*** which may be achieved utilizing the same software mentioned earlier. The calibration step is used to adjust the value of some of the input parameters in order to match the prevailing conditions on the highway segment under analysis. Although some authors propose rigorous calibration procedures (Muñoz, 2004), the trial-and-error process presented in Figure A21 can be used to find approximate parameter values.

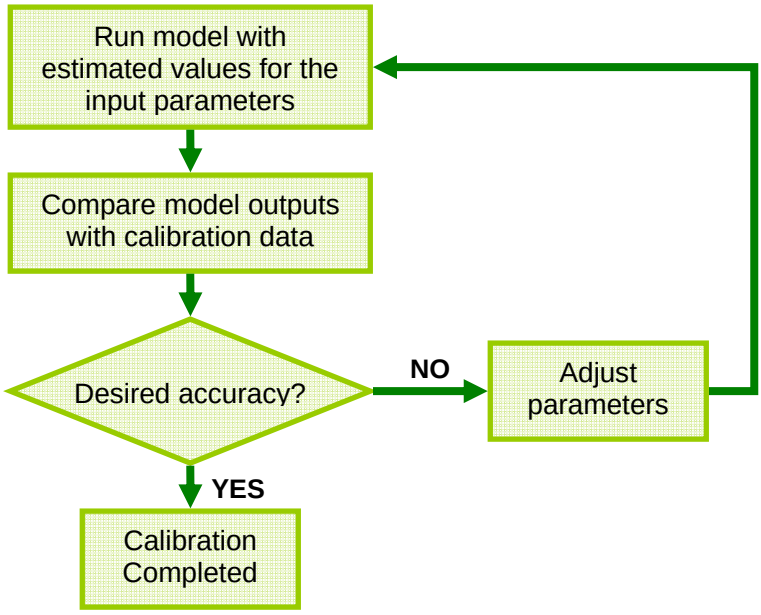

Figure A21: Flowchart for calibration

Table A1 describes the parameters contained in the inputs files, which can be adjusted in order to improve the model fit, along with reference values, based on Muñoz, 2004.

Table A1: Parameter Values

Parameter	Description	Reference Values
Nmax	Jam density. Maximum number of vehicles that can fit per unit distance.	150–180 vehic./mile/lane 0.02936-0.03409 v/ft/lane
Qmax	Maximum flow that can move from one cell to the other in a time interval	1500-2000 veh/hour/ln 0.417-0.555 veh/hour/ln
w	Speed of the backward moving shockwave. Reflects how fast congestion travels upstream	20 fps ~13.6 mph
f	Free flow speed	75-85 mph 102-124 fps

Two types of ***calibration data***, described below, can be used to adjust model parameters.

Travel time data: if ACTUAL travel times are known for the desired OD pairs, for a sufficient number of minutes during the day (60-120), one can compare such values with the travel times predicted by the model. For example, if a probe vehicle departing location A at 9:00 AM experienced a travel time of 10 minutes to destination B, it should be compared with the average travel time predicted by the model for the interval starting at 9:00 AM. Actual travel times may be obtained by measuring the travel time experienced by probe vehicles equipped with GPS units, or using some type of re-identification algorithm based on video data or AVI.

Cumulative counts data: The cumulative traffic counts at each sensor can be compared to the cumulative counts simulated by the model, which are saved in an output file generated when the model is run in Calibration mode. It is important to notice that when traffic volumes are relatively high, and the simulation time exceeds one hour (which is advisable), cumulative counts may be very large. As a consequence, even small percent differences between real and predicted traffic volumes may involve fairly large differences in terms of the actual number of vehicles counted at each simulation step, which may in turn affect the travel time prediction quality. If only volumes are used for calibration, strict thresholds in terms of model fit should be enforced. Additionally, it is recommended to plot the travel time profile for the simulated period, and assess its feasibility based on experience and engineering judgment.

In order to ***run the model*** simulating a real-time environment, the sample input files provided in the corresponding CD can be used. The model reads the input files as if data were provided at a given frequency by sensors, and prediction was necessary to run the model and compute travel times. Similar files can be created using new detector data for any given freeway segment. However, in order to deploy this model in a real-time environment, it is necessary to adjust imperfect sensor data to fit the minimum model requirements (section data requirements) and determine when the database should be read and the predictions computed.

In summary, this model, implemented via computer software, provides the tools to deploy a travel time prediction model capable of capturing traffic dynamics, which only demands traffic counts as inputs at run time. The preparation of input data and model calibration demands

some additional data and work. The later is justifiable given the desirable characteristics of the prediction model. Additionally, the simulation model can be used for purposes other than travel time prediction, if properly adjusted.

N Curve Model

Model Description

In 1997 Daganzo proposed travel time prediction on a freeway under work zone conditions [1]. Daganzo described this approach by modifying input-output diagrams to measure the time and distance spent by vehicles in a queue in a simpler manner than using a time-space diagram. This process requires the construction of a curve depicting the cumulative number of vehicles reaching the back of queue as a function of time, depicted in Figure A22:

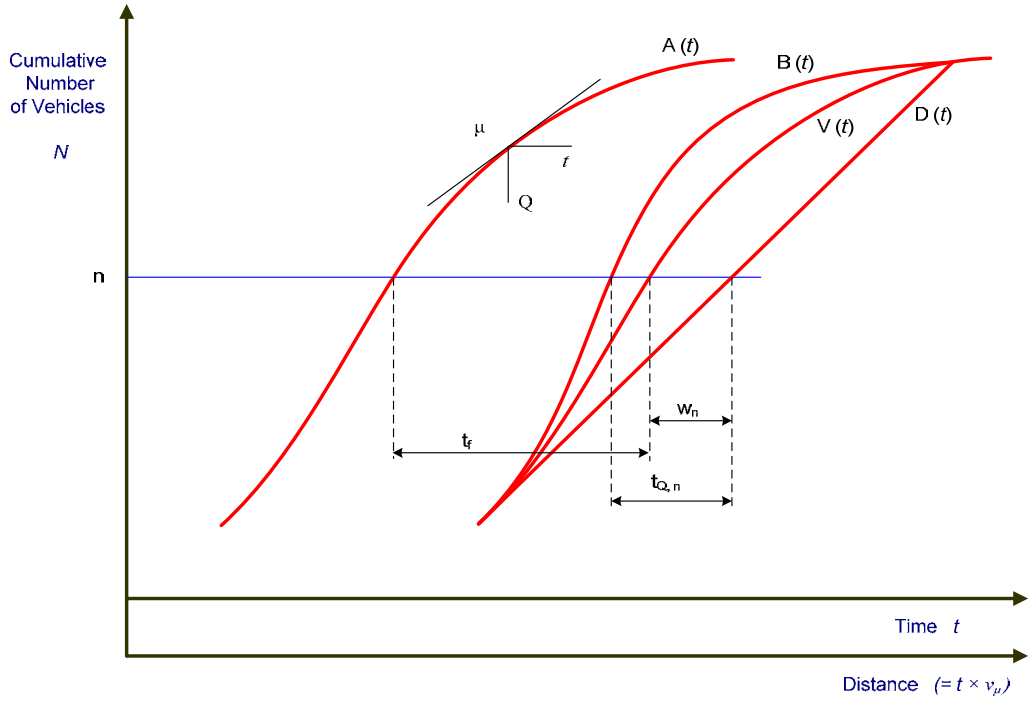

Figure A22: N-Curve (Input-Output) Based Method [1]

The previous figure demonstrates the arrival time of each vehicle at an upstream observation point is measured, and plotted on the figure as the curve $A(t)$. By translating the arrival time of each vehicle horizontally to the right by the free-flow travel time to the bottleneck, t_f, the desired (or "virtual") arrival time of each vehicle at the bottleneck can be plotted as the curve $V(t)$. Finally, the departure curve, defining the time that each vehicle departed the bottleneck, can then be constructed in the usual way to serve the virtual arrivals at a maximum rate m. For a given vehicle number n, the horizontal separation between $V(t)$ and

$D(t)$ represents the delay for that vehicle, and is denoted w_n; and the horizontal separation between $A(t)$ and $D(t)$ represents the total delayed travel time for that vehicle, t_{q_n}.

Using the relationships derived from a space time diagram, the input-output diagram of figure above can be modified to include the curve $B(t)$, the number of vehicles to reach the back of the queue by time t, or equivalently, the times that each vehicle reaches the back of the queue. Time can be determined for each vehicle joins the back of the queue by "extending" the delay of each vehicle, w_n. The locus of these points for all vehicles represents the "back of queue" curve, $B(t)$, which can now be constructed on the input-output diagram. Obviously, $B(t)$ will differ from $V(t)$ only for those vehicles for which $V(t)$ differs from $D(t)$; i.e., whenever a queue is present.

Daganzo's model development and case study are based on the hind-sight information on a closed system of a freeway. The advantages of utilizing the *N*-Curve model can be summarized as follows: 1) it requires less data for training calibrating, needing only traffic flow counts for the upstream, downstream, and ramp points, 2) less data also means more robustness and generality when implemented, and 3) it can handle more traffic situations than other models while generating more accurate results. Nevertheless, this approach assumes a closed highway segment without considering on- or off-ramps between the upstream and downstream detectors. In the case of an off-ramp existing between the upstream and downstream detector, the highway segment loses its conversation of flow and the cumulative curve at the downstream detector can not be directly used for travel time prediction without further modifications. In the event that both on and off ramps exist in the segment of interest, the situation could become more complicated.

Methodology

A generalized model development of an N-Curve based method that accounts for general freeway configuration, including on and off-ramps, is proposed to estimate and predict travel time on a freeway segment.

Definition of Variables:

π_i: arrival time at detector i

N_i: Cumulative curve for detector i, $\forall\, i \in I : \{m, ..., j\}$

$N_i(\pi_i)$: N-curve marker when arriving at detector i at time π_i

$T_{i,i+1}(\pi_i)$: travel time between detector i and i+1 when arriving at detector i at time π_i.

$\Phi_{m,j}(\pi_m)$: arrival time at detector j when the entire journey starts at detector m at time π_m

By definition, $\Phi_{\mathrm{m,j}}(\pi_m) = N^{-1}\left[N_{j-1}(\pi_{j-1})\right]$, we can also show (proof omitted here) that

$$\Phi_{\mathrm{m,j}}(\pi_m) = N^{-1}\left[N_{j-1}(\pi_{j-1})\right] = \sum_{i=m}^{j} T_{i,i+1}(\pi_i) + \pi_m$$

This means that to find the arrival time at detector j, we can find the time-dependent travel time for each detector pair within m and j, and sum up these travel time and the arrival time at detector m.

Travel Time Estimation without Ramps

To initiate travel time prediction utilizing the base algorithm, the origin and destination points (or detectors) of the traveled path must be defined; in this case the origin detector is defined as m while the j is the destination detector. Detector i is defined to be in the set of I, where $I:\{m,...,j\}$. Travel time, $T_{i,i+1}(\pi_i)$, for two sensors for the complete simulation time, $\pi=\{0,...,T\}$.

The computations then iterates as follows:

Step 0: m = i

Step 1: $T_{i,\,i+1}(\pi_i)=N_{i+1}^{-1}[N_i(\pi_i)]-\pi_i$

Step 2: $\pi_{i+1}=\pi_i+T_{i,\,i+1}(\pi_i)$

Step3: $\Phi_{m,j}(\pi_m)=\pi_{i+1}$, stop if $i+1=j$

otherwise $i=i+1$, go to Step 1

Refer to Figure A23 for a graphical representation of the base algorithm.

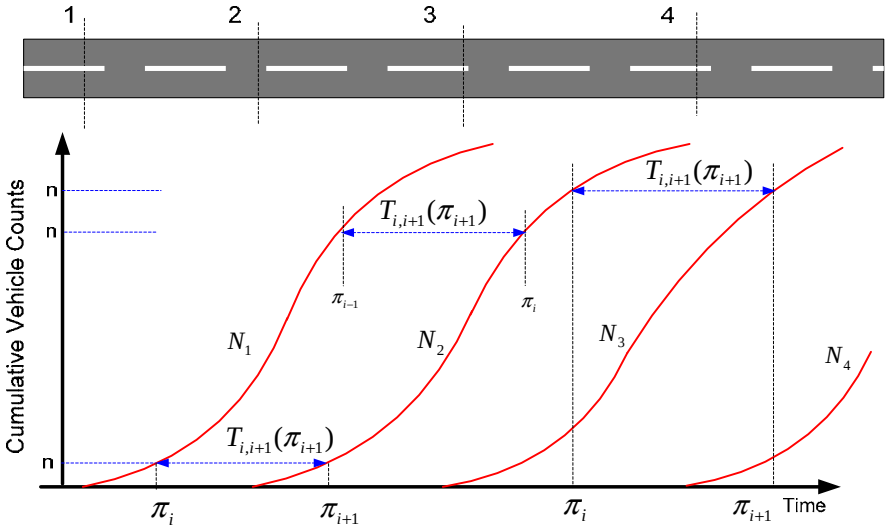

Figure A23: Mainlane N-Curve Method

Travel Time Estimation with Ramps

To account for the presence of on- and off-ramps in the prediction corridor of interest, the base algorithm requires of a couple of modifications. Clearly, the technique eliminates the 'closed system' found in the without-ramp case, meaning traffic flow conservation is no longer true between sensors placed upstream and downstream of the ramp junction. Thus, the detector

set I is now expanded to $I'' = I + I'$, which includes the original detectors, $I : \{m, ..., j\}$, and the virtual detectors , $I' = \{K'\}$. *K'* is the downstream virtual detector corresponding to *K*, where *K* is located upstream of either an on or off-ramp and is an element of *I*. This results in the creation of 'virtual' detectors *i'* for any main-lane detector upstream *i* of an on- or off-ramp. The *N*-curve for the virtual detector *i* then becomes:

$N_{i'} = N_i + N_{i''}$ if detector *i* is the upstream detector for an on-ramp

$N_{i'} = N_i - N_{i''}$ if detector *i* is the upstream detector for an off-ramp

Note that this virtual detector is assumed to be at the locations immediately downstream adjacent to the ramp. Following the previously defined iterative travel time process, travel times are calculated only for sensor pairs with flow conservation, including both actual and virtual detectors. As to be discussed in the later statements, the travel time between the detector *i* and *i"* needs to be estimated. At this moment, we assume that this travel time equals to ε.

Refer to the following two figures (Figures A24 and A25) for a graphical representation of the on- and off-ramp algorithm.

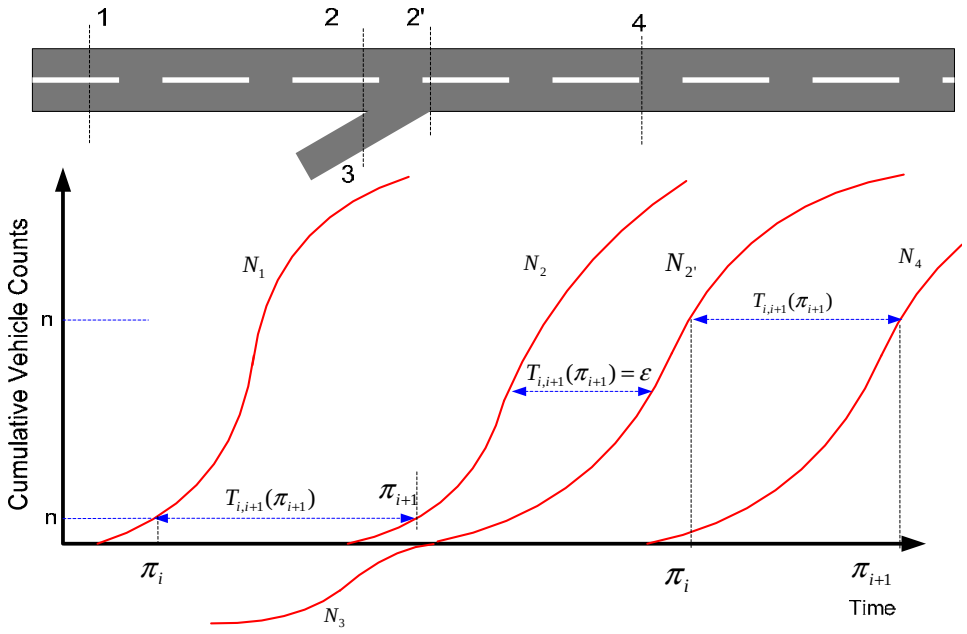

Figure A24: On-Ramp N-Curve Method

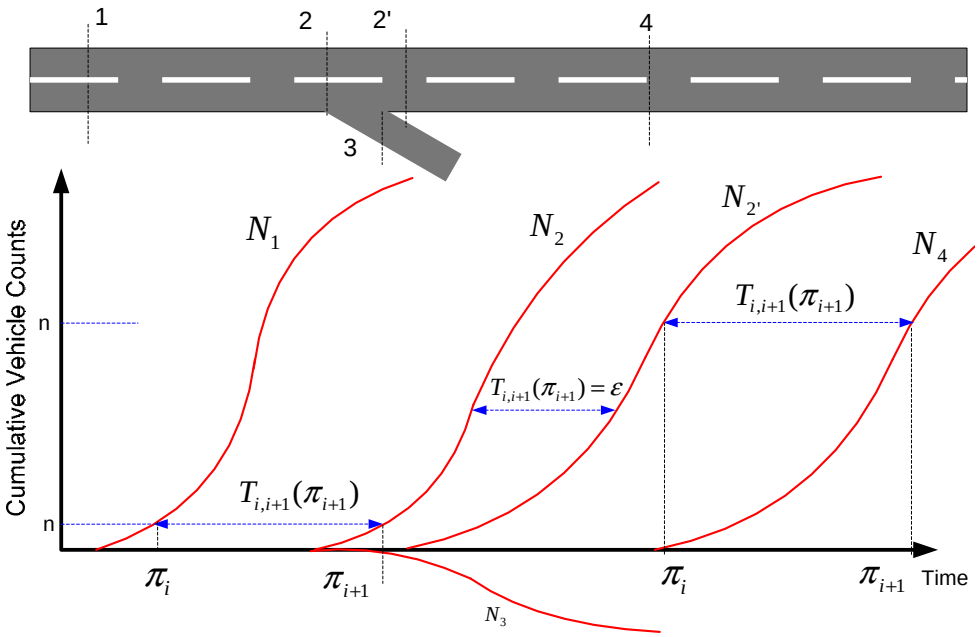

Figure A25: Off-Ramp N-Curve Method

Detector Coverage Analysis

The data availability, limited somehow by the detector coverage, plays an important role on the performance of travel time prediction models. However, most of the existing literature regarding traffic sensor deployment was found to focus mainly on identifying optimal location of sensors for purposes other than travel time, such as improving Origin-Destination matrix estimation (for example, Ehlert et al., 2006, Sherali et al., 2006). Moreover, much of the surveyed work intends to select which links of a network should be equipped (Thomas and Upchurch 2002, Sisiopiku et al 1994), rather than on identifying detector positions within a link that may present grater advantages. The latter has only been studied in an urban context, and generally for traffic management purposes, such as actuated traffic signs (Liu et al., 2004). By performing a ***fine-resolution (cell level) analysis of the impact of sensor location on travel time prediction accuracy, which is explicitly modeled, the proposed methodology improves upon existing techniques.***

For TxDOT project 0-5141, a software tool was developed, based on ***the CTM-based travel time prediction model***. The same structure adopted in such software can be used to assess the behavior of other models. Furthermore, one may expect some of the results and insights obtained from this model to be valid for other prediction methodologies that utilize cumulative traffic counts as the basis for their travel time forecasts.

Given the complexity of the relationship between traffic volumes, actual travel times, and predicted travel times, it is not feasible to write a single mathematical expression capturing the model performance. ***The proposed optimization approach enumerates all desirable detector deployment patterns, compares simulated versus real travel times, computes error measurements, and selects the pattern leading to smaller prediction errors.*** The developed methodology requires the use of an external simulation tool for the following purposes:

- Provide “real” travel times under a range of traffic conditions
- Provide traffic counts for every possible hypothetical location of the detectors

The optimal detector deployment model ***can be run in two ways***: allowing the software to generate and evaluate all possible detector deployment patterns, or providing the set of patterns to be analyzed as an input.

While enumerating all possible patterns is a more comprehensive option, it may increase the computational work excessively, because the number of possible patterns grows very fast as the number of cells increases. A way to reduce the number of patterns to be considered is to include a threshold, limiting the minimum separation between detectors. It was observed in the simulation experiments that travel time computations based on detectors placed very close together could be less accurate than those generated using detectors at least 0.5 miles apart. Moreover, deploying detectors closer than a half-mile apart may not be economically feasible. Other possibilities for path generation include the use of heuristic procedures, such as genetic algorithms, which may be coded separately and linked to the source. The following diagram (Figure A26) depicts the optimal detector location process:

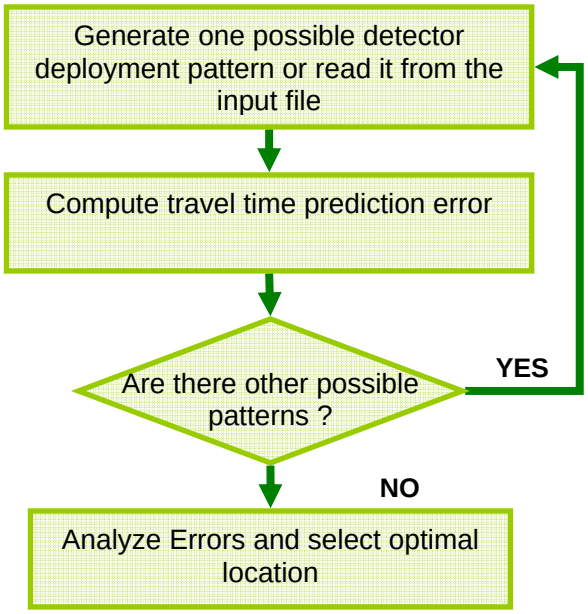

Figure A26: Flowchart for optimal detector location

The output file generated by this software provides, for each analyzed pattern, a number of error measurements for every OD pair considered for traffic prediction, as well as a global error measurement. The error definitions are included in the corresponding templates. Notice that, in virtue of the existence of multiple origins and destinations, it is not always straightforward to identify whether one pattern performs better than others. A pattern exhibiting a low global error may involve high errors for specific OD pairs. One possible criterion to select an optimal pattern is to search for the one which minimizes the maximum error across OD pairs. The user may develop and code other criteria into the source code.

The combination of the detector deployment tool and the previously introduced travel time prediction methods gives rise to a powerful integrated simulation and analytical framework,

which can be used to answer questions such as these: how large of a spacing is acceptable under what accuracy requirements, and what degree of accuracy can one expect given a level of detector coverage? These are of great importance to any district considering developing the capability for specifying prediction accuracy requirements, and budgeting capital investment for detectors.