

# UC Berkeley

## Research Reports

### Title

Travel Times on Changeable Message Signs: Pilot Project

### Permalink

<https://escholarship.org/uc/item/2wp2p0m6>

### Author

Chen, Chao

### Publication Date

2004-03-01

CALIFORNIA PATH PROGRAM  
INSTITUTE OF TRANSPORTATION STUDIES  
UNIVERSITY OF CALIFORNIA, BERKELEY

# **Travel Times on Changeable Message Signs: Pilot Project**

**Chao Chen**

**California PATH Research Report  
UCB-ITS-PRR-2004-5**

This work was performed as part of the California PATH Program of the University of California, in cooperation with the State of California Business, Transportation, and Housing Agency, Department of Transportation; and the United States Department of Transportation, Federal Highway Administration.

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 views or policies of the State of California. This report does not constitute a standard, specification, or regulation.

Final Report for Task Order 4306

March 2004

ISSN 1055-1425

# Travel Times On Changeable Message Signs: Pilot Project

Chao Chen  
PeMS Development Group  
University of California, Berkeley

December 19, 2003

# Contents

<b>1</b>	<b>Executive Summary</b>	<b>6</b>
<b>2</b>	<b>Introduction</b>	<b>7</b>
2.1	Background . . . . .	7
2.2	Technologies . . . . .	8
2.2.1	Travel time measurement . . . . .	8
2.2.2	Travel time prediction . . . . .	8
2.2.3	Message generation and delivery . . . . .	9
2.3	Tasks . . . . .	9
2.3.1	Site selection and benefit analysis . . . . .	9
2.3.2	Development of CMS content . . . . .	10
2.3.3	Interface between PeMS and CMS . . . . .	10
2.4	Results . . . . .	10
<b>3</b>	<b>Travel Time Prediction On A Freeway Route</b>	<b>11</b>
3.1	Problem description . . . . .	11
3.2	Real time measurement . . . . .	11
3.3	Linear regression prediction . . . . .	12
3.4	Discussion: time lag of prediction . . . . .	13
3.5	Calculating actual travel time . . . . .	13
3.5.1	Interpolating speed . . . . .	13
3.6	Fitting model parameter . . . . .	14
3.7	Application to the CMS problem . . . . .	15

<b>4</b>	<b>Potential Benefits</b>	<b>16</b>
4.1	Problem Formulation . . . . .	16
4.2	Route Description . . . . .	17
4.3	Route Selection . . . . .	17
4.4	Average Travel Time Reduction . . . . .	19
4.5	Buffer Time Reduction . . . . .	20
4.6	More Results . . . . .	22
<b>5</b>	<b>Performance Validation</b>	<b>24</b>
5.1	CMS Locations . . . . .	24
5.2	I-80 . . . . .	24
5.2.1	Data Quality . . . . .	26
5.3	I-5 . . . . .	27
<b>6</b>	<b>Implementation</b>	<b>31</b>
6.1	Destinations, routes, and freeways . . . . .	31
6.2	Segment travel times . . . . .	32
6.3	Segment prediction coefficients . . . . .	32
6.4	Travel Time Prediction Module . . . . .	33
6.5	Message Format . . . . .	33
6.6	System Architecture . . . . .	34
6.7	Testing under operating conditions . . . . .	35
<b>7</b>	<b>Conclusion and recommendations</b>	<b>36</b>
7.1	Follow-up project . . . . .	37

# List of Figures

3.1	Linearity of travel times . . . . .	14
4.1	San Diego: two alternate routes . . . . .	18
4.2	Travel times on routes 1 and 2. . . . .	19
4.3	Decision based on predicted travel time. . . . .	20
4.4	Average travel times using three strategies. . . . .	21
4.5	Travel time using prediction vs. using historical median only. . . . .	22
4.6	Ninety percent buffer times. . . . .	23
5.1	Map of I-80 corridor . . . . .	25
5.2	Data quality, I-80E . . . . .	26
5.3	CMS locations on I-5 in orange county, District 12. . . . .	27
6.1	Configuration tables: CMS, routes, and segments. . . . .	32
6.2	CMS application system and networking. . . . .	33
6.3	CMS type 1: single destination, multiple routes. . . . .	34
6.4	CMS type 2: single freeway, multiple destinations. . . . .	34

# List of Tables

4.1	Description of routes, San Diego test case . . . . .	17
4.2	Benefit summary for five origin-destination pairs. . . . .	23
5.1	Tach measurements on I-80. . . . .	29
5.2	Summary statistics from probe measurements. . . . .	30
5.3	Travel time measurements and estimates on I-5. . . . .	30
6.1	CMS and destinations on Interstate 80 in Bay Area. . . . .	35

## **Acknowledgment**

This project was made possible by Caltrans. We thank the cooperation of Caltrans District employees in Districts 4, 7, 11, and 12 for their input on the system design and evaluation. We especially thank Kane Wong and Alan Chow of District 4 and Paul King of District 12 for measuring travel times for the ground truth comparison. We also thank Caltrans Headquarters for devoting resources for automating CMS message delivery and guiding the development the CMS message content format.



# Chapter 1

## Executive Summary

We describe a system to display real time travel times on Changeable Message Signs (CMS) in California. CMS's show dynamic information and allow the Traffic Management Center (TMC) to communicate to drivers information about traffic diversion, incidents, and delays. This type of service is deployed in other parts of the country and world and has been shown to be useful. For this project, we implemented a system that uses existing algorithms for travel time estimation and prediction.

The system that we designed uses real time information to compute expected travel times at the current time on selected routes originating from the CMS location. The data come from the Freeway Performance Measurement System (PeMS) database. PeMS collects 30-second data from most California urban freeways. Travel time predictions are computed using an existing algorithm that is implemented in Perl. The prediction engine is shown to be functional, reliable, and scalable.

We evaluated the potential benefit of real time prediction on historical data from several California routes. The analysis shows that routing choice based on real time information reduces average travel time and improves its predictability. However, the accuracy of the system relies on good data quality. We found the current data quality needs to be improved in some locations.

We have proven the technical feasibility of displaying travel times on CMS's using available data in California. The calculation of travel time estimates are based on real time data or a combination of real time and historical information using algorithms. But the project was canceled before we could deploy it and study consumer reactions and actual benefits.

## Chapter 2

# Introduction

Changeable message signs (CMS's) along freeways can display dynamic information to help drivers choose the best route and plan for delays. Currently, operators in the Traffic Management Center (TMC) type messages about incidents, road conditions, and events onto the signs. More recently, CMS's are also used to display automatically generated travel time information. Such services exist in Europe and in US cities including Houston, San Antonio, and Atlanta. Real time traffic data and computer algorithms are used to generate and display the CMS content.

This report describes the implementation of an automated CMS service in California. The goal of this pilot project is to display predicted travel times from the message sign to several destinations along the route. A computer program processes real time data from loop detectors and calculates the predicted travel times. The predictions are sent to the TMC, where another program displays them onto the message signs.

### 2.1 Background

Many state Departments of Transportation (DOTs) provide real time traffic information. In Washington state, a network of loop detectors measures link speeds. These data are used to calculate instantaneous travel times, which are posted on the Central Puget Sound Travel Times web site (1). Changeable message signs deliver traffic information directly to drivers on many freeways. Often, traffic operators type messages onto the signs from the TMC, alerting drivers of lane closures or accidents ahead. Users of the TransGuide system in San Antonio “consider the [changeable message signs] to be more useful and reliable than radio or other sources of traffic information because of the signs proximity to traffic congestion and their higher level of accuracy” (2).

Increasingly, changeable message signs are displaying automatically generated travel time information. The Georgia Navigator is a network of detectors and message signs built for the 1996 Atlanta Olympic games (3). It computes link travel times using video traffic detectors and publishes route travel times on CMS's as well as on a web page. The Houston Transtar system (4) uses Automatic Vehicle Identification (AVI) technology to measure travel times and posts them on CMS's. AVIs measure travel times directly, so it has an advantage over loop-based travel time estimates.

But AVI tags and readers are installed in only a few places, whereas loop detectors are much more common traffic sensors. Loop detector data are the only data widely available in California. Since our travel time estimates are not directly measured they need to be verified with ground truth. The WSDOT observed good agreement between loop detector-based and probe vehicle travel times (1). The validity of loop-based travel time estimates are also supported by Coifman (5) and van Lint (6). Our own probe measurements show that accuracy of loop data-based estimates is good given good data quality.

The providers of the aforementioned CMS traveler information services do not have definitive social studies on their impact, or quantifying benefits to consumers or highway management. But their continued operation indicates widespread acceptance by the public. As part of this study, we planned to survey drivers' opinion on the effectiveness of CMS travel times, as well as measure any impact the messages have on traffic patterns. Since the system was not deployed, this part of the project was not done. We recommend that it be a part of a follow-up project.

## 2.2 Technologies

Delivering travel time information on message signs requires measurement of current traffic conditions, prediction of future travel times, and delivery of the message content onto the electronic signs.

### 2.2.1 Travel time measurement

Inductive loop detectors provide most of the traffic data available in California. There are also some microwave and magnetic sensors. All of these sensors provide spot measurements of some combination of vehicle count, occupancy, and speed. But they do not directly measure travel time. Therefore, we need to estimate travel time from these measurements using an algorithm. We use the PeMS database which contains real time and historical data from most California urban freeways. Note that with any method, we can estimate travel times only for completed trips, because travel times of uncompleted trips depend on future speeds.

### 2.2.2 Travel time prediction

Most CMS traveler information systems, including the ones mentioned above, show past travel times. For example, the Houston Transtar system shows travel times of vehicles that have just traversed a route. But the travel time of someone just beginning on the same route may be different, especially if travel times are changing quickly. What we want is to estimate the *future* travel time, which requires prediction. Van Zwet (7) showed that prediction that takes into account both historical trends and real time information performs much better than that using current measurements alone. We implemented van Zwet's algorithm and demonstrate its benefit.

### 2.2.3 Message generation and delivery

We built an application that generates travel time predictions on designated routes periodically. This application runs continuously in the background and wakes up every five minutes to compute the prediction for each route given current conditions. The results are formatted appropriately and stored in a file at PeMS. In the Caltrans TMC, another process periodically queries PeMS for the current travel time predictions on all routes. The TMC process retrieves the newest predictions and displays them onto the appropriate CMS's.

## 2.3 Tasks

The project proposal described these tasks:

1. Site selection
2. Develop CMS content
3. Interface CMS software with PeMS content
4. Deploy and monitor pilot program
5. Evaluate effectiveness of CMS
6. Final report

Tasks 1 and 2 have been completed. Task 3 is completed on PeMS's end according to Caltrans specifications, but the TMC process is not yet finished. Tasks 4 and 5 are not completed because Caltrans did not give the permission to deploy. Task 6 is this report.

### 2.3.1 Site selection and benefit analysis

Travel time information is most valuable on routes that have highly variable travel times, and origin-destination pairs that have several alternate routes. Of course, the sites we select must have functioning CMS installed and good measurements from detectors. We chose several CMS locations and corresponding destinations to find the potential benefits of travel time prediction to drivers on these routes. Using real data and our algorithms, we found that drivers potentially save time and reduce uncertainty at all locations.

Two locations were selected for deployment, one in District 4 and one in District 12. The selection was based on the Districts' preferences. Extensive evaluations on both prediction accuracy and estimation accuracy were carried out on in District 4; some verification was also performed in District 12.

### 2.3.2 Development of CMS content

The CMS contents, or travel time predictions, are generated using technologies outlined in sections 2.2.1 through 2.2.3. The process is implemented in Perl and completely automated. Currently the process generates travel times on the designated deployment routes; more CMS locations and routes can be added easily.

### 2.3.3 Interface between PeMS and CMS

The CMS's are controlled by software in the TMC. Using the HTTP protocol and the Internet, PeMS generates the CMS content and makes it available to the TMC on the Web as a text file. This file is refreshed every five minutes with the current predictions. The content is formatted according to Caltrans specifications and can contain messages for multiple CMS's. TMC side of the communications is not completed. But since it simply has to display the messages without applying any logic, this part should be easy.

## 2.4 Results

Although we did not deploy the system and were not able to evaluate its performance as perceived by the public, we implemented an operational system that can be deployed quickly when executive decisions are made. We were able to evaluate its performance using historical data, and quantify the travel time estimation accuracy, potential benefits, and reliability of the software package.

The validity of the travel time estimation and prediction algorithms have been established in research literature. We also found them to be accurate when applied to our data. We quantified the overall effectiveness of the system using several months of historical data from several test locations. It showed large potential savings in average travel time when competitive alternate routes are available, and large reductions in travel time uncertainty in all cases. The implemented system is reliable and stayed up for one week without errors. Because the real time portion of the prediction mechanism is simple, this system does not require intense computations and is easily scalable to more CMS locations.

The most significant technical hurdle that remain is the lack of reliable data. Currently, the system can be implemented only at locations where the data quality is good. On the other hand, the calibration process reveals the location of malfunctioning detectors and may be used as a tool in targeted detector maintenance.

## Chapter 3

# Travel Time Prediction On A Freeway Route

### 3.1 Problem description

We need to predict the travel time on a route between a given CMS location on the freeway to some destination in the freeway network, for a starting time  $t$ . For example, suppose the current time is 4:53 PM. We may display this message on the southbound CMS sign just upstream of the I-5/I-805 split in San Diego:

DOWNTOWN SD	
I-805	25-28 MIN
I-5	35-37 MIN

This message shows the predicted travel time to downtown San Diego on these two routes for drivers who are at the CMS location at the current time – 4:53 PM. Suppose the time of the most recent measurement is 4:50 PM, and call the time of the most recent measurement  $t_0$ . Our problem is to predict the travel time departing at  $t$ , given measurements made at  $t_0$ , where  $t \geq t_0$ .

### 3.2 Real time measurement

Travel time predictions can be derived from measurements of current speeds and a model of travel time behavior. Most urban freeways in California are equipped with loop detectors or other stationary sensors that measure speed directly or indirectly.<sup>1</sup> The detectors are about one third to one half mile apart on average. They have an update rate of every 30 seconds. PeMS computes five-minute

---

<sup>1</sup>Some detectors, such as double loop installations and radars, measure speed directly. There are also many single loop installations that measure volume and occupancy, which are then used to estimate speed. PeMS obtains accurate speed estimates from single loops using an algorithm described in a paper by Jia (8).

average speed from every location in real time.<sup>2</sup> Suppose we have a route that is  $y$  miles long and has  $n$  detector locations at miles  $x_1, \dots, x_n$  from the start of the route. From each detector  $i$  and at time  $t_j$ , we have speed measurements  $\hat{v}(x_i, t_j)$ , which represents the average speed of all vehicles who crossed point  $x_i$  during the time interval between  $t_j$  and  $t_j - 30$  seconds. If we divide the route into  $n$  links where the  $i$ th link is the part of the route that is closest in distance to the  $i$ th detector, a naive estimate of travel time departing at time  $t_0$  is the sum of the current link travel times:

$$T^*(t_0) = \sum_{i=1}^n \frac{l_i}{\hat{v}(x_i, t_0)}. \quad (3.1)$$

However, this is not the actual travel time for departure at  $t_0$  because the actual travel time depends on speeds after the departure, i.e. for times  $t \geq t_0$ . Equation (3.1) implicitly assumes that speeds in the future on all the links will be the same as the current speeds. This is why we need to do prediction.

### 3.3 Linear regression prediction

There is much research on how to predict travel times on freeways. The best method seems to be to use a combination of real time and historical knowledge about the route. We choose an algorithm developed by van Zwet (7) which uses linear regression. In this method, actual travel times for departure at time  $t$  is modeled as a linear function of  $T^*(t_0)$ :

$$T(t) = \bar{T}(t) + (T^*(t_0) - \bar{T}^*(t_0))\lambda(t_0, t) + \epsilon(t_0, t), \quad (3.2)$$

where

$$\begin{aligned} T(t) &\stackrel{def}{=} \text{travel time departing at } t \\ \bar{T}(t) &\stackrel{def}{=} \text{historical average travel time for departing at time of day } t \\ \bar{T}^*(t_0) &\stackrel{def}{=} \text{historical average of } T^* \text{ at time of day } t_0 \\ \lambda(t_0, t) &\stackrel{def}{=} \text{parameter indexed by both } t_0 \text{ and } t \\ \epsilon(t_0, t) &\stackrel{def}{=} \text{Gaussian noise.} \end{aligned}$$

This model has been verified on real data and found to be a good fit. It is very simple to implement because it requires only one parameter,  $\lambda(t_0, t)$ . Actually,  $\lambda$  depends on both the time of day of the measurement  $t_0$  and departure time  $t$ . Therefore the number of parameters is really the number of distinct combinations of  $(t_0, t)$  we care to have. We choose to have a five-minute resolution in both departure and measurement times, and we can realistically compute lags of up to 60-90 minutes. The lag is the difference between the departure and measurement times,  $t - t_0$ . If the lag is very large, then the measurements made at  $t_0$  is not very meaningful at  $t$ . Since there are 288 five-minute periods in a day, and 18 five-minutes in 90 minutes, the number of parameters we need to compute is  $288 \times 18 = 5184$ . The details of how these parameters are computed is found in Section 3.6 and in van Zwet (7).

<sup>2</sup>The five minute interval is a convenient aggregation level which represents ten 30-second raw samples. This update rate should be adequate for travel time prediction applications.

### 3.4 Discussion: time lag of prediction

The above travel time model allows for a delay between departure time  $t$  and measurement time  $t_0$ . In some applications this is necessary, for example, when we want to predict the travel time of departing an hour from now. While the CMS application only needs to predict for current departure time, having the ability to predict future travel times means we can take care of the case when measurement falls behind real time. For example, there can be delays of several minutes in the data path, from the moment measurements are made to the time when five-minute aggregates are computed. The ability to handle departure times that are different from measurement times smoothly compensates for any delays between measurement time and when the data become available for calculations.

### 3.5 Calculating actual travel time

In order to calculate the parameters  $\lambda(t_0, t)$  for prediction, we need historical measurements of real travel times  $T(t)$ . But actual travel time measurements are generally not available. On the other hand, we have measured speeds  $v(x_i, t_j)$  from detectors  $i$  at every 30 seconds  $t_i$ , so we should use these measurements to compute travel time. Since  $v(x_i, t_j)$  are spot measurements at discrete locations  $x_i$  and aggregated over times  $t_j$ , and travel time is a function of all locations  $0 < x < y$  and times between the departure time and the arrival time, what we can compute from measured speeds are only estimates of the true travel times. Research has shown that with intelligent interpolation between the measurement locations and times, we can obtain accurate estimates of the true travel time (5, 6).

We use an iterative method to calculate travel time from speed measurements using a method first developed by Oda (9). Starting at time  $s_0$  and location  $z_0$ , we interpolate the speed at the current location and time and hold this speed for  $\Delta$  seconds to find the next location,  $z_1$ . Then we repeat with process with  $s_1 = s_0 + \Delta$  and  $z_1$ , until the destination,  $z^*$ , is reached.

$$z_{k+1} = z_k + \tilde{v}(z_k, s_k)\Delta \quad (3.3)$$

We use an iteration interval of  $\Delta = 10$  seconds. The sequences of  $s_k$  and  $z_k$  represent the trajectory; the estimated travel time is

$$T(s_0) = \Delta * \min \{k : z_k > z^*\} \text{ seconds.} \quad (3.4)$$

#### 3.5.1 Interpolating speed

The above iteration method requires knowledge of speed at any given location and time. Since measurements are obtained from discrete locations and times, speed for an off-grid point has to be interpolated. Let  $v(x_i, t_j)$  be the measured speeds. We actually interpolate the *inverse* of the measured speeds as follows:

$$\tilde{v}(s, z) = \left[ \text{linear}\left(\frac{1}{v(x_i, t_j)}\right) \right]^{-1} \quad (3.5)$$

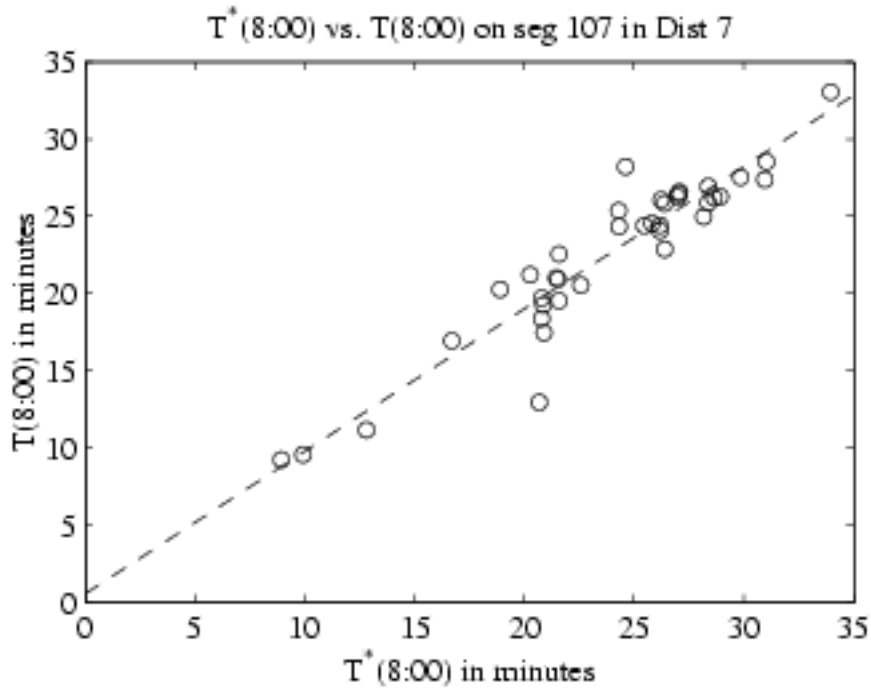


where  $\text{linear}(\bullet)$  linearly interpolates  $(s, z)$  from the four closest grid points  $(x_i, t_j)$ .

### 3.6 Fitting model parameter

Once we have historical values of  $T^*(t)$  and  $T(t)$ , we can easily compute parameters  $\lambda(t_0, t)$  for each combination of  $(t_0, t)$ . Given  $t_0$  and  $t$ , we can compute historical values  $T^*(d, t)$  and  $T(d, t)$  using equations (3.1) and (3.4). Here, we use the index  $d$  to indicate the day and  $t$  to indicate the time of day.

This prediction model assumes travel times of different days to be independent and identically distributed for a given time of day. In other words,  $T(d, t)$  is iid given  $t$ . This is suggested by the scatter plot in Figure 3.1, which shows a remarkably linear relationship between  $T(t)$  and  $T^*(t_0)$ .



**Figure 3.1**  $T(t)$  vs.  $T^*(t_0)$  for a segment in Los Angeles, where  $t_0 = t = 8$  AM. The relationship is linear.

The parameter  $\lambda(t_0, t)$  can be found by finding the slope of the straight line in Figure 3.1. We use a robust least-squares method. Suppose we have the following data:

- $T_d \equiv T(d, t)$ : travel time estimates for days  $d = 1, 2, \dots, n$  at time of day  $t$ ;
- $T_d^* \equiv T^*(d, t_0)$ :  $T^*$  for days  $d$  and measurement time  $t_0$ .

Compute the sample means  $\bar{T}$  and  $\bar{T}^*$  and standard deviations  $\sigma$  and  $\sigma^*$ . Remove the outlier travel times: keep days  $d$  in  $D$ :

$$D \stackrel{def}{=} \{d : T_d < \bar{T} + 3\sigma; T_d^* < \bar{T}^* + 3\sigma^*\}. \quad (3.6)$$

Recompute the means  $\bar{T}$  and  $\bar{T}^*$  using the reduced set  $D$ . Then, minimize least squared error to find  $\lambda$ :

$$\hat{\lambda} = \operatorname{argmin}_{\lambda} \sum_{d \in D} [T_d - \bar{T} - \lambda(T_d^* - \bar{T}^*)]^2. \quad (3.7)$$

This procedure is repeated for each combination of  $t_0$  and  $t$ . See Section 6.3 for the implementation in Matlab.

### 3.7 Application to the CMS problem

After prediction parameter  $\lambda(t_0, t)$  and travel time statistics  $\bar{T}(t)$  and  $\bar{T}^*(t)$  are computed, real time prediction is simple. At the current time  $t$ , suppose the most recent measurements were made at  $t_0 < t$ . The instantaneous travel time  $T^*(t_0)$  can be computed from the measured speeds  $v(x_i, t_0)$ . Apply the equation in (3.2) to estimate the travel time:

$$\hat{T}(t) = \bar{T}(t) + (T^*(t_0) - \bar{T}^*(t_0)) \hat{\lambda}(t_0, t). \quad (3.8)$$

# Chapter 4

## Potential Benefits

### 4.1 Problem Formulation

Changeable message signs are ideal for delivering real time information to drivers. If there are two alternate routes to the same destination, displaying their predicted travel times enables the driver to choose the quicker one; if travel time is abnormally high, the driver may decide to delay his trip and do something else instead. We quantify the benefit of real time travel time on several freeway routes. The benefits include average travel time savings and the reduction in uncertainty.

Suppose there are  $n$  alternate routes between a given origin-destination pair. Let  $X_i(d, t)$  be the travel time on the  $i$ th route on day  $d$  and time of day  $t$ . Model  $X_i(d, t)$  as a random variable whose distribution depends on  $i$  and  $t$  but is independent and identically distributed (iid) in  $d$ . At any time  $(d, t)$ , we can make a prediction of  $X_i(d, t)$ , call this  $\hat{X}_i(d, t)$ . Notice that we don't know the actual value of  $X_i(d, t)$  until some time after  $(d, t)$ , the departure time. The CMS displays the prediction  $\hat{X}_i(d, t)$  for each route  $i$ .

We compare two strategies. The first chooses the route with historically the lowest average travel time; the second chooses the route with the lowest predicted travel time. We assume that the actual travel time on either route is not affected by the driver's decision, discounting the possibility that route guidance itself alters the congestion patterns. Let  $T_0, T_1$  be the actual travel time realized following the two strategies. We hypothesize that the second strategy is better because it uses more information. The travel time realized is  $T_1$ , the actual travel time of the route with the shortest predicted travel time, where

$$T_1(d, t) = X_k(d, t), \text{ } k \text{ such that } \hat{X}_k(d, t) \leq \hat{X}_i(d, t) \text{ for all } i \leq n \quad (4.1)$$

In the first strategy, drivers only know the historical average travel time at each time of day  $t$ , which is their "prediction" of the current travel time since no other information is available. Thus,

$$\hat{X}'_i(t) = E[X_i(1, t)] \text{ for all } i \leq n, \text{ for all } d \quad (4.2)$$

is the predicted travel time on each route. Then

$$T_0(d, t) = X_k(d, t), \text{ } k \text{ such that } \hat{X}'_k(t) \leq \hat{X}'_i(t) \text{ for all } i \leq n \quad (4.3)$$

Route	Length	Detector Stations	Daily Volume
I-5 SB	14.61	12	90,174
I-805 to I-163	14.05	13	86,129/67,769

**Table 4.1 Description of routes, San Diego test case. Volumes are daily averages in August, 2003.**

From historical data, we can find  $\hat{X}_i(d, t)$  and  $\hat{X}'_i(t)$  and estimate  $T_0, T_1$  and their distributions.

For a thorough comparison, we introduce a third, hypothetical travel time  $T_2$ , which is the travel time on the actual fastest route,

$$T_2(d, t) = \min \{X_i(d, t) : i \leq n\}. \quad (4.4)$$

$T_2$  is the lower bound of travel time of any routing strategy.

## 4.2 Route Description

The performance of these route selection strategies is evaluated using data from San Diego freeways. There are two alternate routes between the I-5/I-805 interchange and the I-5/I-163 interchange. One can stay on I-5 SB for the entire trip or take I-805 SB and then I-163 SB. We number the routes 1 and 2 respectively. They have similar traffic characteristics. See Table 4.1 and Figure 4.1. We compute travel times  $X_i(d, t)$ , where  $i = 1, 2$ , for departure times between 5:00 AM and 10:00 PM on 8/1/2002 through 8/31/2002. There are 22 weekdays included in the study. Travel times are computed for 1320 departure times over the study period at every 17 minutes. For each departure time, we calculate a trajectory in space and time that satisfies the measured speeds by “walking” through the speed surface in time and space. For more details on the calculation of historical travel times see the SYSTEM IMPLEMENTATION section, Chen (10), or Oda (9).

The scatter plot of route 1 and route 2 travel times is shown in Figure 4.2. Each point represents travel times on the two routes with the same departure time; there are 1320 points. Travel times on the two routes are comparable, suggesting that they are realistic alternatives. However, for any given departure time, either route may have the shorter travel time.

## 4.3 Route Selection

Let  $R_0(t) \in \{1, 2\}$  be the route taken based on strategy 0, i.e.

$$R_0(d, t) = \underset{r}{\operatorname{argmin}} E[X_r(1, t)].$$

$R_0(d, t)$  is fixed for each time of day  $t$  and does not depend on  $d$  because it uses only historical information. Let  $R_1(t)$  to be the choice based on real time prediction, i.e.

$$R_1(d, t) = \underset{r}{\operatorname{argmin}} \hat{X}_r(d, t).$$



**Figure 4.1 San Diego: two alternate routes between I-5/I-805 split and I-5/SR-163 interchange.**

This is a random variable that depends on  $t$  but is assumed to be iid in  $d$ . Similarly, the route choice under full knowledge is

$$R_2(d, t) = \underset{r}{\operatorname{argmin}} X_r(d, t).$$

$R_0$  does not depend on real time data and therefore depends only on  $t$ , the time of day. But  $R_1$  depends on the current information, therefore it depends on both  $d$  and  $t$ . Figure 4.3 shows for each time of day, the probability that route 1 is chosen based on strategy 1. Recall that strategy 1 chooses the route with the minimum predicted travel time based on current information. Route choice is clearly affected by the real time information. On average, during the morning, route 1 is chosen on most days; during the afternoon peak at 18:00, about 40% of the time route 1 is chosen, and the other 60% of the time route 2 is chosen.

We now compare the performance of real time route selection versus route choice based only on historical knowledge. We calculate the reduction in average travel time and buffer time, a measure of uncertainty. The use of real time information also successfully avoids routes with

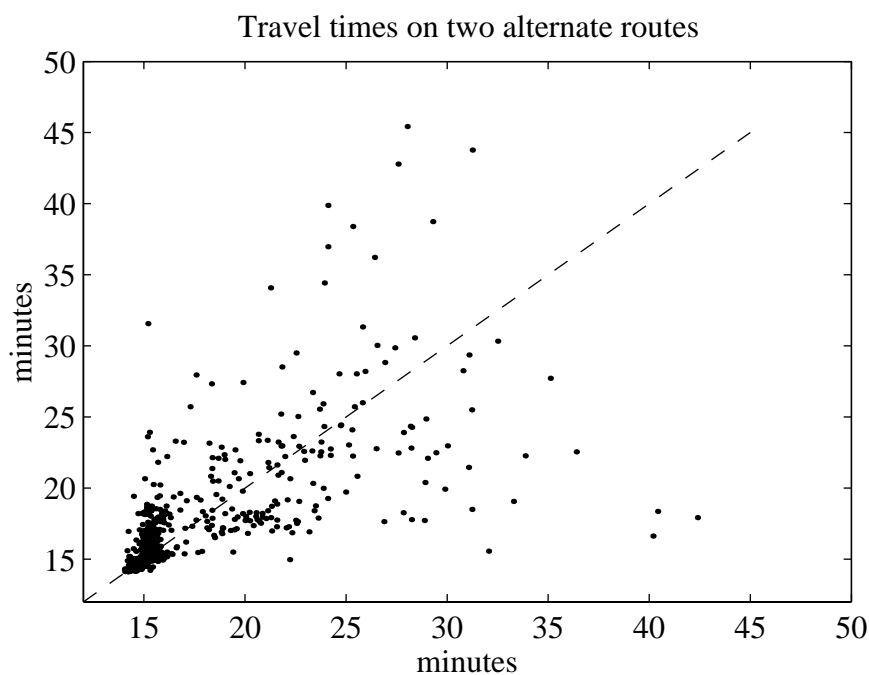


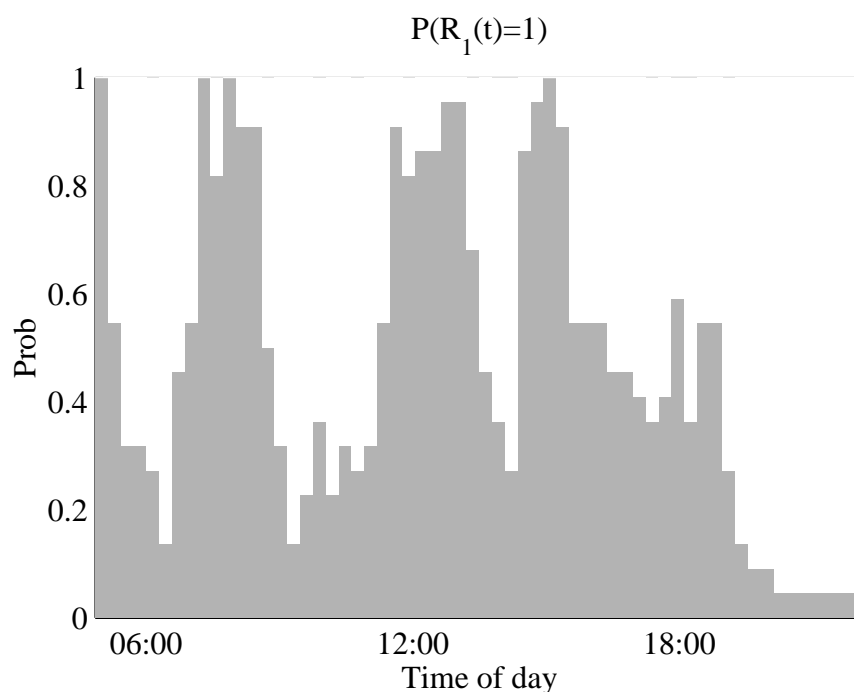
Figure 4.2 Travel times on routes 1 and 2.

abnormally high delays.

#### 4.4 Average Travel Time Reduction

Real time travel time predictions  $\hat{X}_i$  on routes  $i = 1, 2$  are computed for every departure time. The details of travel time prediction is given in the SYSTEM IMPLEMENTATION section. We also compute  $T_0$ ,  $T_1$ , and  $T_2$  – the achieved travel times of the three route decision strategies. Figure 4.4 shows the average travel times for each departure time of day during the test period. The quicker route according to CMS takes about 2 minutes less than the historically quicker route during the afternoon peak period, or about 9% of the average. The achieved CMS travel times are very close to the minimum possible travel time shown by the dotted line. This shows that CMS almost always picks the route with minimum travel time.

The improvement in average travel time may seem moderate because CMS reduces travel time only when the historically quicker route is the slower one on a given day. But individual trip savings can be great. Figure 4.5 shows the scatter plot of  $T_1(d, t)$  vs.  $T_0(d, t)$  for all 1320 departure times  $(d, t)$ . There are 60 departures per day over 22 days, representing one point every 17 minutes on each day and 1320 total points. For  $(d, t)$  when  $R_0(d, t) = R_1(d, t)$ ,  $T_0(d, t) = T_1(d, t)$  as well and the points fall on the diagonal. This is the case when the decisions based on historical and current information are the same. Most of the trips fall on the diagonal. But there are also many trips with  $T_1 < T_0$ . On these trips, using real time information leads to a shorter travel time. There are also a few trips that have  $T_1 > T_0$ : on these trips, the predicted travel times do not identify



**Figure 4.3 Decision based on predicted travel time.**

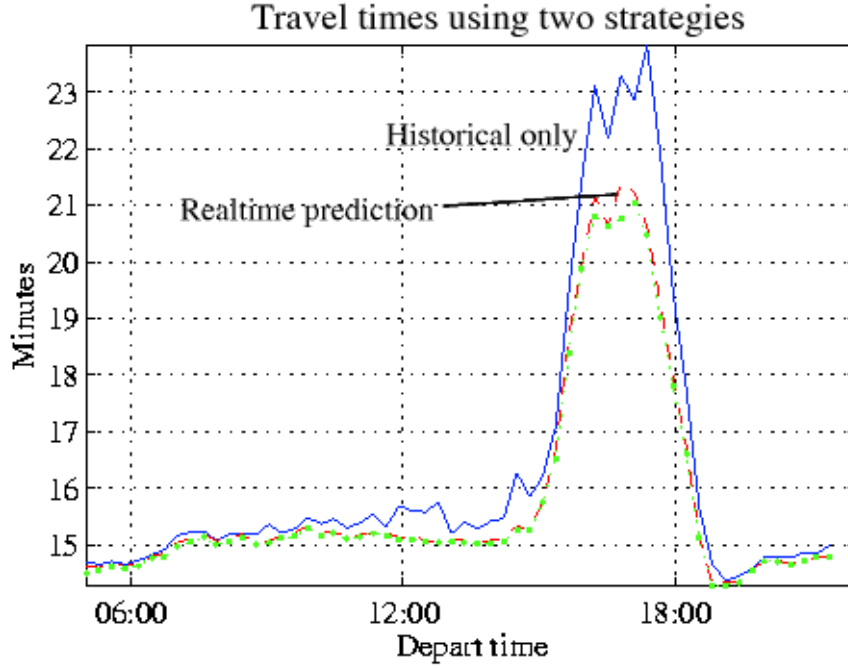
the shortest route, *and* historical prediction makes the correct decision. But even on these trips, the difference in the achieved travel times is small compared to the difference in the other cases. These results show that almost all trips would have been shorter using the CMS predictions.

## 4.5 Buffer Time Reduction

Even when no alternate route is available, accurate prediction of travel time reduces uncertainty and allows people to better plan activities that depend on their arrival time. Studies have found that drivers place a cost on the variability of travel time (11) and value the ability to make informed decisions even if no alternate routes are available (2). Therefore, the reduction of uncertainty through accurate prediction of travel time provides a useful service in itself.

The cost of uncertainty in travel time is quantified by buffer time. This is the total time that must be budgeted in order to arrive on-time with a certain probability, and can be much larger than average travel time. The Texas Transportation Institute's Mobility Study uses 95% level as the buffer time (12); we use 90% here because our data set is small.

When only historical information is available, the 90% buffer time is the 90<sup>th</sup> percentile of historical travel time, which depends on time of day. When real time information is used, the buffer time depends also on current measurements. The 90% buffer time  $y^{90}(s, t)$  given predicted travel time  $t$  and departure time  $s$  is estimated from historical data. Given  $n$  historical trips, let  $s_i$  be the



**Figure 4.4** Mean travel times using the three strategies. Solid line =  $T_0$ , dashed =  $T_1$ , dotted =  $T_2$ .  $T_1$  and  $T_2$  are almost on top of each other.

departure time of the  $i$ th trip,  $T_i$  be its actual travel time, and  $t_i$  be the predicted travel time. The  $i$ th prediction error is  $\epsilon_i = T_i - t_i$ . We estimate  $y^{90}(s, t)$  using the weighted 90th percentile of  $\epsilon_i$ , where the weights are given by

$$w_i(s, t) = k_1(s - s_i)k_2(t - t_i), \quad (4.5)$$

and  $k_1$  and  $k_2$  are Gaussian kernels with mean zero and variances  $\sigma_1 = 2$  minutes and  $\sigma_2 = 5$  minutes. These values are chosen to make the function appear smooth in  $s$  and  $t$ . The weighted (100 $p$ )th percentile of a sorted set  $\{x_i : i = 1, 2, \dots, n, x_i \leq x_{i+1} \text{ for all } i\}$  with weights  $w_i$  is

$$y = \begin{cases} \frac{1}{2}(x_i + x_{i+1}) & \text{if } \sum_{j=1}^i w_j = pW \\ x_{i+1} & \text{if } \sum_{j=1}^i w_j < pW < \sum_{j=1}^{i+1} w_j \end{cases} \quad (4.6)$$

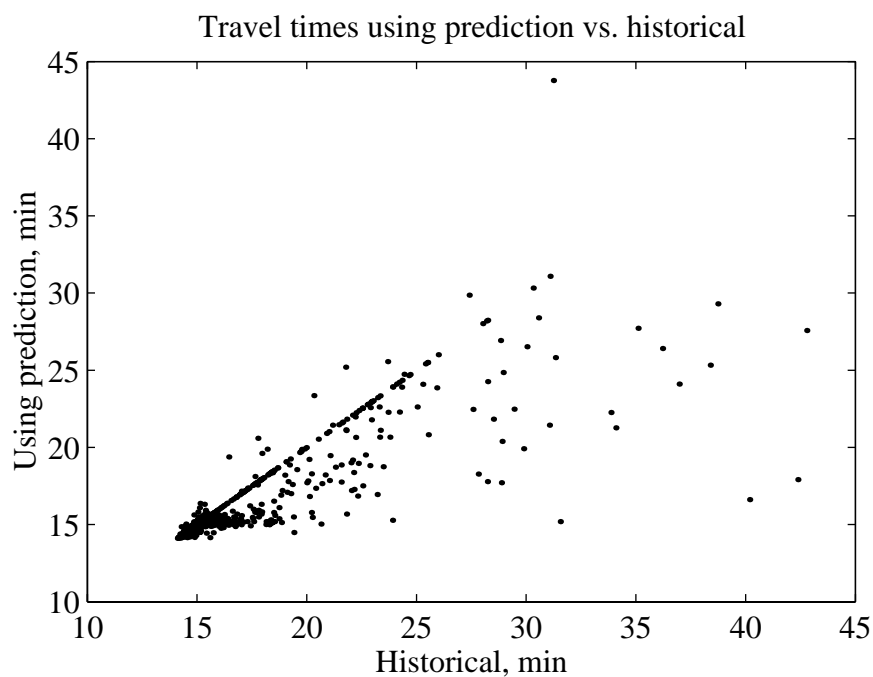
where  $W = \sum_{i=1}^n w_i$  is the sum of the weights. The interpretation of  $y^{90}(s, t)$  is that given the predicted travel time at  $s$  is  $t$ , the prediction error is less than  $y^{90}(s, t)$  with 90% probability.

The buffer time given a time of day  $s$  and travel time prediction  $t$  is the sum of the 90th percentile error and the predicted travel time:

$$T^{90}(s, t) = y^{90}(s, t) + t. \quad (4.7)$$

Figure 4.6 shows the average buffer times at different times of day. The first plot shows the buffer times when only route 1 is available. During the afternoon peak, the 90% buffer time when using





**Figure 4.5** Travel time using prediction vs. using historical median only.

real time information is five minutes less than that using only historical information, a saving of 17%. The second plot shows the savings when only route 2 is available, with similar results. This means that even when no alternate routes exist, prediction using real time information is worthwhile by significantly reducing the buffer time. The third plot in this series shows the effect of combining route selection with real time prediction. The saving in this case is slightly larger, about seven minutes at the peak.

## 4.6 More Results

Similar studies are carried out on four other origin-destination pairs. In each case, the benefits of using real time information are shown in terms of mean travel time and 90% buffer time in Table 4.2. The peak average travel time is computed by first calculating the average travel time by time of day over the test period of one month, and then taking the peak value over all times of day. The reduction in average travel time is similarly computed and represents the peak hour. For most O-D pairs, the average travel time reduction information is small, because often there are few competitive alternative routes available. But in most cases, using real time information results in a large reduction in buffer time. This reduction is between 7% and 31% for the five locations studied.

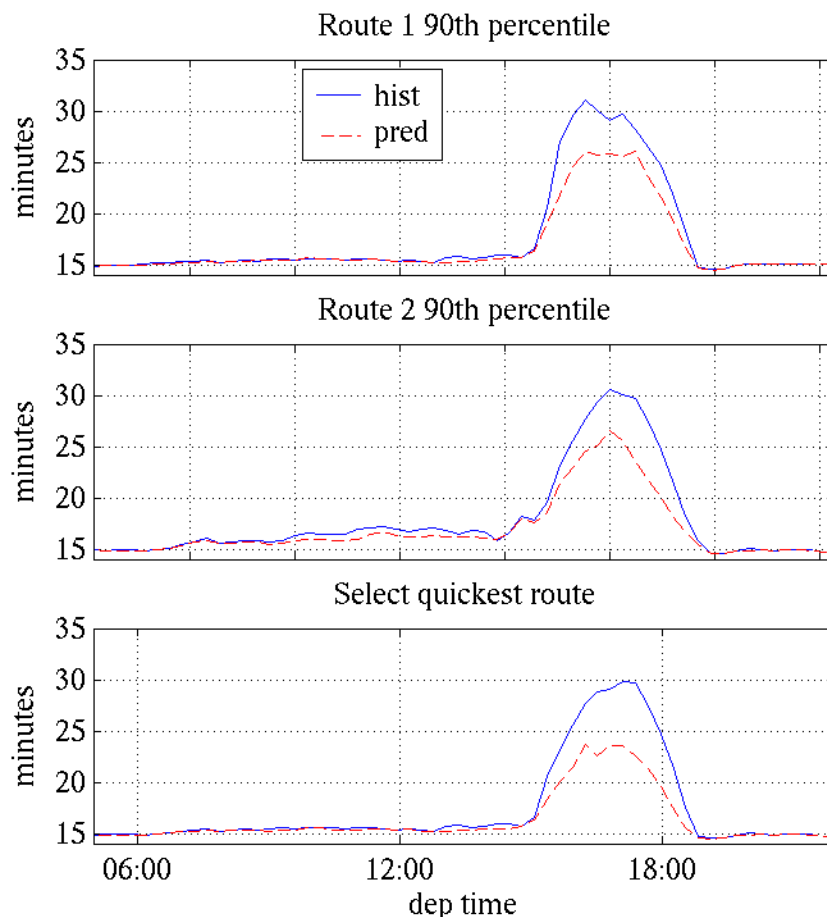


Figure 4.6 Ninety percent buffer times on both routes and using route selection.

Origin and destination	Number of routes	Peak avg. travel time	Travel time reduction	Peak 90% buffer time	Buffer time reduction
I-10 WB at White Ave. to downtown LA	3	41.7	2.9%	75.2	18%
I-5 SB at Terra Bella St. to downtown LA	2	29.8	17.1%	47.7	31%
I-15 SB at I-5 to downtown San Diego	2	22.9	8.7%	29.9	20.7%
I-5 NB at El Toro to Buena Park (I-5 & SR-91)	5	32.9	1.7%	41.8	11%
I-5 NB at El Toro to Seal Beach (I-405 & SR-22)	2	34.2	4.7%	43.5	7%

Table 4.2 Benefit summary for five origin-destination pairs.

## Chapter 5

# Performance Validation

### 5.1 CMS Locations

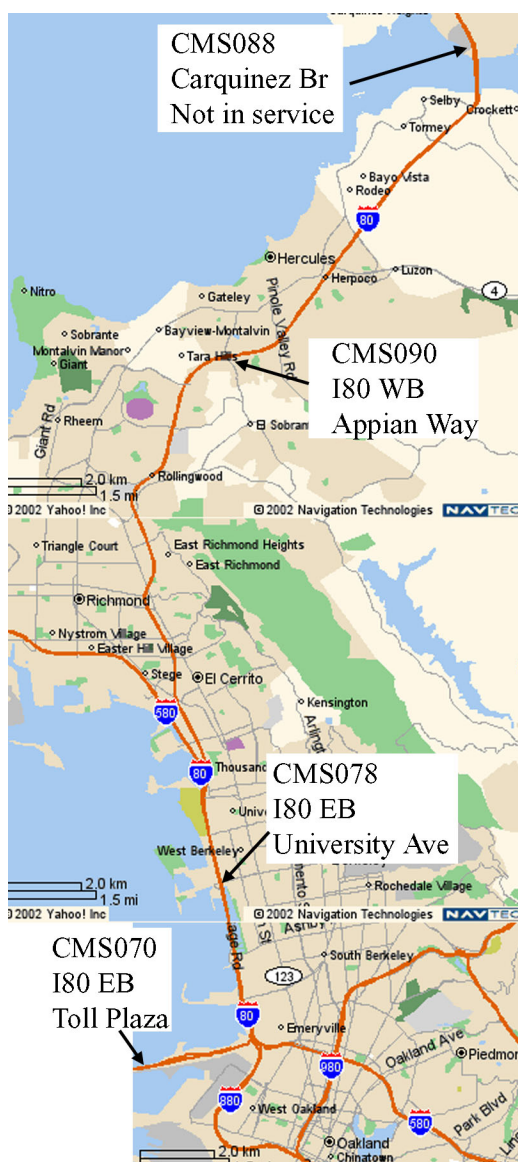
Because direct measurements of travel times are scarce, and loop detector data are abundant, we use point speed measurements to estimate historical travel times. Variants of this method are described by Coifman (5) and van Lint (6), who showed good agreement between loop-based estimates and measured and simulated segment travel times. We also measured travel times with probe vehicles and compared the measurements to estimates based on loop speeds.

Probe vehicle measurements were carried out on two routes. One is on Interstate 80 between the Carquinez Bridge and the Bay Bridge in the San Francisco Bay Area, a section of about 20 miles; the other is on I-5 between El Toro and Buena Park in Orange County, also about 20 miles long. Travel times estimates on I-5 agree with probe measurements, but travel times on I-80 are often affected by missing and bad data. Below is a detailed analysis of each route.

### 5.2 I-80

There are 135 Vehicle Detector Stations (VDS) in the two directions on I-80, where a VDS contains one detector per lane at that location. There is one VDS for each direction at the same location. There are a total of 690 detectors, of which only 284 provided good data. These detectors are double loops which measured speed directly. Probe travel times were measured with tach vehicles, which are equipped with wheel counters to record the number of revolutions of the wheel every second. After calibrating for the wheel size, this gives us the total travel time as well as a detailed trajectory of the vehicle. We made about 74 probe trips between Monday, April 21 and Friday, April 25, 2003.

Using loop detector data from the same period, we computed the travel time estimates for each five minute period. Table 5.1 shows the probe measurements and estimated travel times for the same departure times. The errors are large on 4/22 and 4/23 AM and 4/22 PM, but they are good on other dates such as 4/24 AM. An examination of the data quality revealed many missing samples,

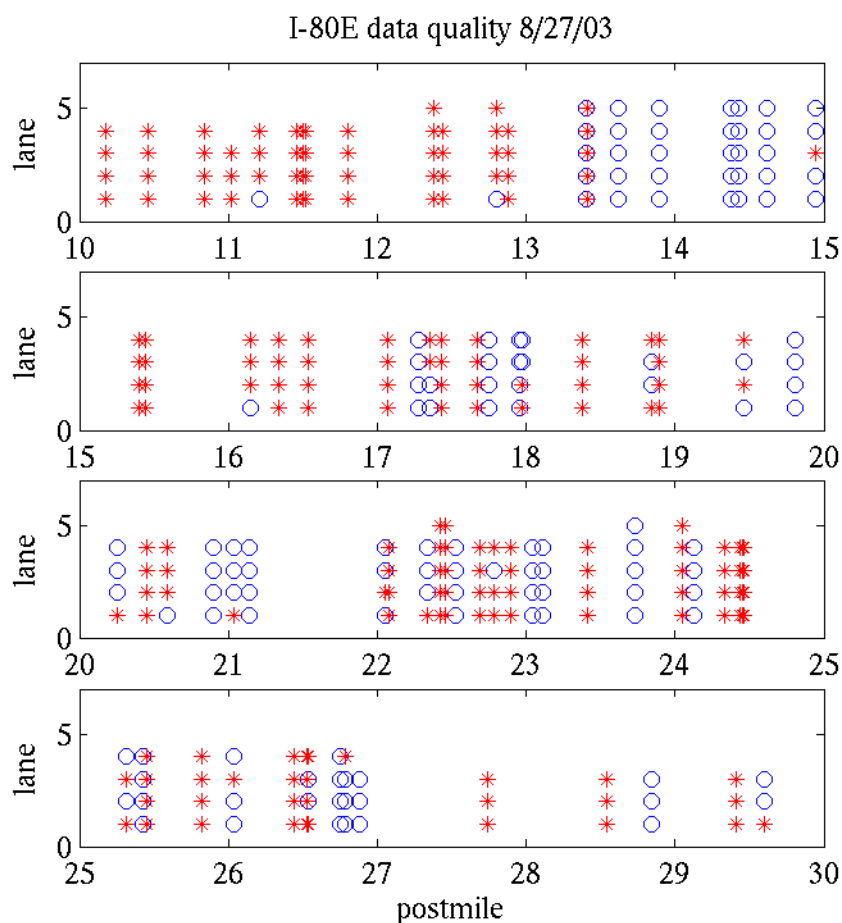


**Figure 5.1 I-80 corridor between Carquinez Bridge and Bay Bridge. CMS messages will be displayed on CMS090 and CMS078 only.**

especially during Tuesday and Wednesday. The cause of the missing data has since been found and the problem corrected, but the missing data from the test period cannot be recovered. The summary statistics of the travel time errors are shown in Table 5.2. After the trips with many missing samples are removed, the root-mean-squared error is 10%.

### 5.2.1 Data Quality

A closer look at the data quality reveals that although there is good detector coverage on the route, many of the detector stations are not functioning properly. Using an algorithm developed in PeMS (13), we diagnosed the data quality from each location and lane in both directions. Most of the bad loops are easily detected because they either give no data or give all zero values. A view of data quality is shown in Figure 5.2, which illustrates graphically the location of the bad loops, based on data from 8/27/2003. We see that although there is at least a detector station every mile,

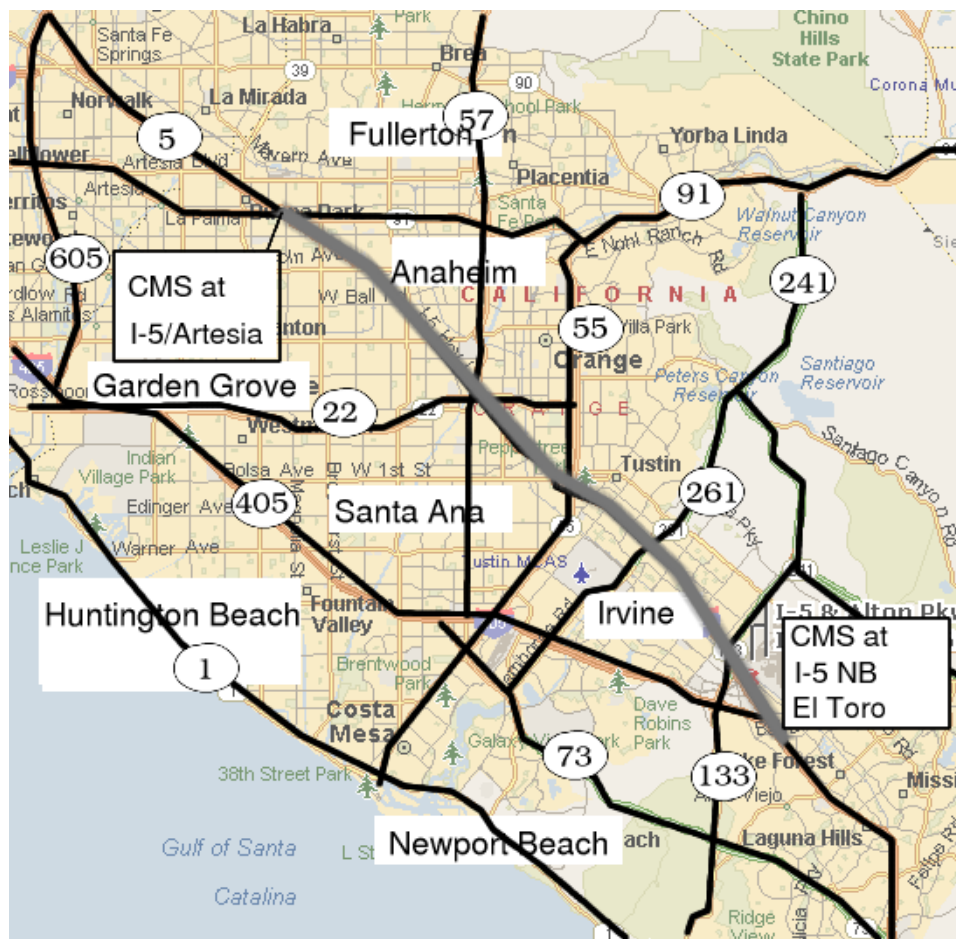


**Figure 5.2 Data quality on I-80 eastbound on 8/27. Circles show good detectors, asterisks show bad ones.**

only about half of all stations are working. This means there are long stretches where there is no detection, such as between miles 15 and 17, where there are five consecutive bad stations. The loss of data from these locations seriously degrades the accuracy of travel time estimates.

### 5.3 I-5

The route on I-5 in Orange County is shown in figure 5.3. There is a CMS in the northbound direction in El Toro at postmile 18.9, and a CMS in the southbound at Artesia, postmile 43.99. There are 120 VDS in both directions. The data quality here is better than on I-80. Travel time



**Figure 5.3 CMS locations on I-5 in orange county, District 12.**

measurements were made by District 12 personnel, first using manual data collection and later using tach vehicles. We analyzed the results of only the manually recorded travel times because we have not yet completed the analysis of tach travel times. In the manual data collection, the driver of an ordinary vehicle phoned in the times at which he crossed certain designated checkpoints, and the times were recorded by his partner in the office. Fourteen such probe runs were made. Table 5.3 shows the comparison between probe travel times and loop travel times. Most of the probe measurements matches detector estimates very well. The exceptions are three trips on 5/14 SB. There was a severe accident in the afternoon of this day which was noted by the driver. This may have caused the large error for the point at 2 PM. The other two points in the morning cannot be similarly explained. They may have been the result of gaps in the data coverage. We scheduled

tach measurements for this route, which will provide more detailed data to allow us to diagnose the exact cause of these discrepancies.

The RMS error is about 14%. See Table 5.2. It is not clear why the probe travel times are always higher than loop-based estimates in these comparisons.

Eastbound					Westbound				
Departure Time	Probe (min)	Estimate (min)	Abs Error	Bad data	Departure Time	Probe (min)	Estimate (min)	Abs Error	Bad data
<b>4/21</b> 6:57 AM	15.9	16.5	4%		6:57 AM	26.6	20.3	24%	
7:26 AM	15.8	16.0	1%		7:26 AM	31.7	26.7	16%	
8:09 AM	17.6	16.5	6%		7:55 AM	31.2	23.5	25%	
8:38 AM	16.1	17.7	10%		8:24 AM	28.9	22.2	23%	
2:38 PM	17.4	18.2	4%		9:07 AM	20.0	18.8	6%	
3:07 PM	18.7	18.7	0%		2:52 PM	20.6	20.0	3%	
3:36 PM	31.9	31.8	0%		3:36 PM	17.7	18.7	5%	
4:04 PM	29.1	34.8	20%		4:04 PM	17.3	18.3	6%	
<b>4/22</b> 6:57 AM	16.8	18.5	10%	*	6:57 AM	33.5	18.0	46%	*
7:40 AM	15.4	21.5	40%	*	7:26 AM	43.7	15.3	65%	*
8:09 AM	16.1	22.7	41%	*	7:55 AM	42.0	15.5	63%	*
8:38 AM	15.3	27.0	76%	*	8:38 AM	34.0	31.0	9%	*
2:24 PM	16.1	16.8	5%	*	9:07 AM	26.1	16.0	39%	*
2:52 PM	17.6	16.8	4%	*	2:52 PM	17.9	17.0	5%	*
3:36 PM	19.6	17.2	12%	*	3:36 PM	18.9	16.5	13%	*
4:04 PM	19.6	17.2	12%	*	4:04 PM	27.3	17.3	37%	*
					4:33 PM	25.4	18.8	26%	*
<b>4/23</b> 6:57 AM	16.8	17.5	4%	*	6:57 AM	30.6	17.8	42%	*
7:26 AM	15.5	15.3	1%	*	7:26 AM	38.6	17.7	54%	*
8:09 AM	16.2	27.2	68%	*	7:55 AM	38.8	18.7	52%	*
8:38 AM	15.7	16.0	2%	*	8:38 AM	36.8	32.5	12%	*
2:38 PM	16.4	17.3	6%		8:52 AM	30.8	28.0	9%	*
3:36 PM	20.0	22.8	14%		3:07 PM	18.2	19.5	7%	
					4:04 PM	20.4	21.5	5%	
<b>4/24</b> 6:57 AM	17.6	18.0	2%		7:26 AM	44.5	40.5	9%	
7:12 AM	17.9	18.2	1%		7:40 AM	49.4	44.0	11%	
8:09 AM	17.3	18.0	4%		8:24 AM	51.8	45.3	12%	
8:52 AM	17.2	17.2	0%		8:38 AM	51.2	43.5	15%	
2:38 PM	17.6	18.3	4%		9:07 AM	41.1	35.3	14%	
3:07 PM	18.0	19.3	7%		2:52 PM	16.7	17.8	7%	
3:36 PM	20.5	21.0	2%		3:36 PM	16.7	17.7	6%	
4:04 PM	21.1	21.8	3%		4:04 PM	16.3	17.0	4%	
					4:33 PM	16.9	16.3	3%	
<b>4/25</b> 6:57 AM	16.2	16.2	0%		7:12 AM	20.5	21.3	4%	
7:40 AM	16.1	16.3	1%		7:26 AM	21.3	19.5	8%	
8:09 AM	16.8	16.5	2%		7:55 AM	19.8	18.3	7%	
3:07 PM	21.1	19.0	10%		8:24 AM	18.8	18.8	0%	
4:04 PM	22.3	18.0	19%						

Table 5.1 Tach measurements on I-80.



Freeway	Mean err.	Mean absolute err.	RMS err.
I-80	-5%	16%	24%
I-80 (removed)	-2%	7%	10%
I-5	-11.5%	11.5%	14.1%

**Table 5.2 Summary statistics from probe measurements.**

	Date	Dir	Probe	Estimate	Error	Incident
<b>5/13</b>	8:09 AM	S	36.5	30.5	16%	
	2:24 PM	S	22.0	21.0	5%	
	2:38 PM	S	23.0	21.0	9%	
	7:40 AM	N	28.0	24.2	14%	
	1:55 PM	N	22.0	21.8	1%	
	3:07 PM	N	28.0	23.8	15%	
<b>5/14</b>	6:28 AM	S	31.0	27.7	11%	
	7:12 AM	S	40.0	30.0	25%	
	2:09 PM	S	38.0	26.7	30%	*
	6:43 AM	N	21.5	20.5	5%	
	7:12 AM	N	24.5	20.7	16%	
	1:55 PM	N	22.5	21.7	4%	
<b>5/15</b>	1:55 PM	S	22.6	20.8	8%	
	1:55 PM	N	22.5	21.5	4%	

**Table 5.3 Travel time measurements and estimates on I-5.**

## Chapter 6

# Implementation

### 6.1 Destinations, routes, and freeways

Conceptually, we have some CMS locations on freeways. On each CMS, we would like to display the travel times to several destinations; for each destination, we may display the travel times on several alternate routes. A *route* is made up of one or more freeway segments, where a segment is described by its freeway ID, direction, and postmile limits.

Information about CMS's, destinations, and routes are stored tables in the PeMS database. Each CMS is associated with a list of destination-route pairs in the CMS\_ROUTES table. For example, the CMS at Appian way on westbound I-80 in Contra Costa county may show travel times to several destinations along I-80, such as Hilltop Drive, University of California, and the Bay Bridge. They each have a corresponding route ID. The CMS\_ROUTE\_DEF table stores the association between routes and segments. Each route ID is associated with a list of segment IDs. In the ROUTING\_SEGMENTS table, each segment is described by its freeway ID, direction, and postmile limits.

The segment is the basic unit in travel time calculations. Travel times are predicted for each segment first, and segment travel times are combined to give the estimate for an entire route. This modular approach means we can first compute travel time prediction coefficients for each segment off line, and be able to predict travel times on any route that can be made up of segments. When we calculate route travel time, we offset the travel time on each successive segment instead of summing the segment travel times for the same departure. For example, suppose route *A* is made up of segments 1 and 2. To calculate the prediction for departing at  $t_1$ , we first calculate the prediction for segment 1 at  $t_1$ , call this  $T_1(t_1)$ . Then, we calculate the prediction for segment 2, but at departure time  $t_1 + T_1(t_1)$ , call this  $T_2(t_1 + T_1(t_1))$ . The route travel time is  $T_1(t_1) + T_2(t_1 + T_1(t_1))$ . In the following sections we describe how to predict segment travel times.

## 6.2 Segment travel times

A Matlab script called *calc\_seg\_tt* computes segment travel times from historical data. It implements the algorithm in Section 3.5 for each segment, which computes  $T_i(t)$ , the travel time for segment  $i$  and departure  $t$ . The algorithm uses speed measurements stored in the VDS\_5MIN\_SUMMARY table in the PeMS database. When running *calc\_seg\_tt*, the user specifies the time period, and the script calculates travel time for every segment in the database at every five minute starting times, and stores the results in the SEG\_TRAVEL\_TIME table. This script also computes  $T_i^*(t)$  for each five minute period. Both  $T_i(t)$  and  $T_i^*(t)$  are needed for fitting the prediction model.

## 6.3 Segment prediction coefficients

We compute segment prediction coefficients using another Matlab script called *tt\_coeffs*, which implements the algorithm described in Section 3.6. Taking  $T_i(t)$  and  $T_i^*(t)$  from the SEG\_TRAVEL\_TIME table, mentioned above, *tt\_coeffs* computes the prediction coefficient,  $\lambda_i(t_0, t)$ , for each segment  $i$ , measurement time  $t_0$ , and departure time  $t$ , at five minute intervals. The coefficients  $\lambda_i(t_0, t)$  are stored in the SEG\_COEFFS table, and statistics  $\bar{T}(t)$  and  $\bar{T}^*(t)$  are stored in SEG\_STATISTICS. The database tables and their relationships are shown in Figure 6.1.

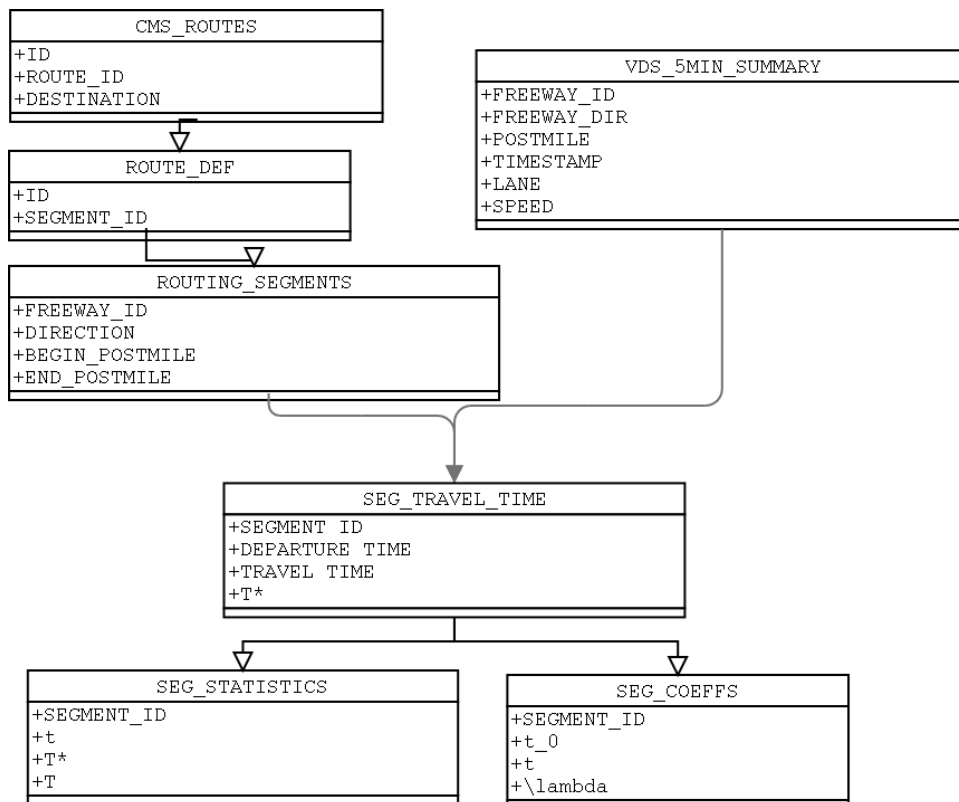


Figure 6.1 Configuration tables: CMS, routes, and segments.

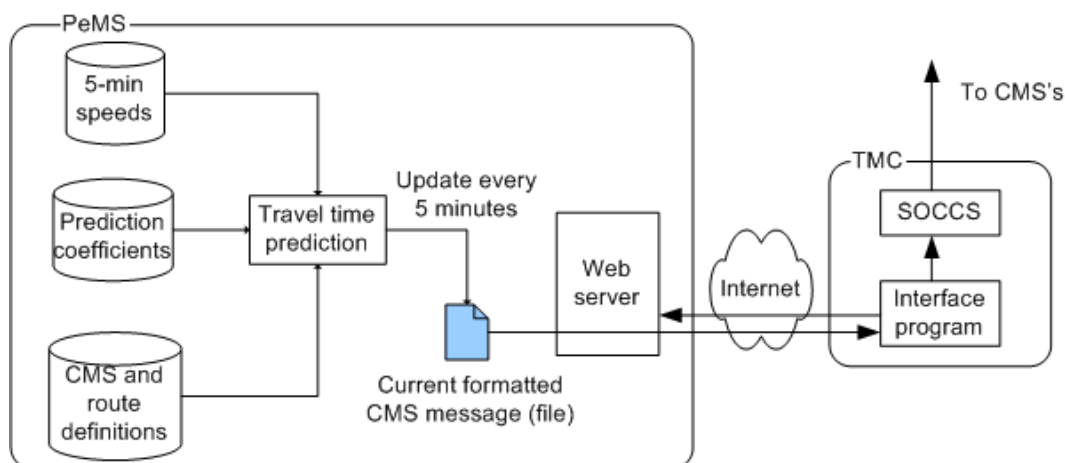


Figure 6.2 CMS application system and networking.

To account for seasonal changes in travel time patterns, prediction coefficients  $\lambda$  and travel time statistics  $\bar{T}, \bar{T}^*$  should be refreshed periodically, such as on a weekly basis. The data used should contain enough days for the regression, but should be restricted to the most recent weeks or months in order to track the seasonal changes in travel time.

## 6.4 Travel Time Prediction Module

Real time prediction is implemented as a Perl program called *cms*, which runs in the background and wakes up every five minutes. Every time it wakes up, it fetches the list of CMS's and their associated routes and destinations. For each route, it predicts the travel time for departing at the current time by predicting the travel times on each segment in an iterative process. Let  $i = 1, 2, \dots, n$  be the segments that make up route  $R$ . The route travel time is computed by

$$\hat{T}_R = \hat{T}_1(t_0, t) + \hat{T}_2(t_0, t + \hat{T}_1(t_0, t)) + \dots \quad (6.1)$$

In words, we first predict the travel time on the first segment, then take the arrival time as the departure time for the second segment, and so on. The measurement time for each segment prediction remains  $t_0$ .

The prediction on each segment,  $\hat{T}_i(t_0, t)$ , is done as in (3.8). For each segment  $i$ , measurement time  $t_0$ , and departure time  $t$ , the program retrieves  $\lambda(t_0, t), \bar{T}(t), \bar{T}^*(t_0)$  from the SEG\_COEFFS and SEG\_STATISTICS tables, computes  $T^*(t_0)$  from real time data, and apply (3.8) to calculate  $\hat{T}_i(t_0, t)$ .

## 6.5 Message Format

CMS's normally can display three lines of sixteen characters each. The messages can have a normal and a bold font; they can be flashing. Longer messages can be displayed as two pages that are

rotated periodically.

We devoted some thought to the message format that most effectively conveys information to the drivers. The messages need to have a balance of information content, simplicity, and utility. There can be two types of CMS messages. One type shows the travel times from the CMS location to a single destination via alternate routes. For example, the San Diego CMS at the I-5/I-805 junction presented in Section 4.2 would show travel times to the I-5/SR-163 junction via I-5 and via I-805. The other type of message shows travel times to different destinations along the same freeway. This is the case in the Bay Area study (Section 5.1) on Interstate-80. This corridor runs between the Carquinez Bridge and the Bay Bridge. The message signs in either directions show travel times from the CMS to three destinations each. These two types of messages are shown in figures 6.3 and 6.4.

DOWNTOWN SD	
I-5	25-27MIN
I-805	42-49MIN

**Figure 6.3 CMS type 1: single destination, multiple routes.**

UNIVERSITY	6MIN
POWELL	12MIN
TOLL PL	19MIN

**Figure 6.4 CMS type 2: single freeway, multiple destinations.**

## 6.6 System Architecture

There are two parts of the CMS system – calculation and display. The calculations are performed on a computer at UC Berkeley, where the PeMS database also resides. The system that displays messages on CMS’s is in the Caltrans District TMC. The way these systems communicate is very simple. The Berkeley component computes travel time predictions for each CMS in its database and stores the formatted messages in a text file that is accessible by a TMC computer program. The access is via HTTP. This file contains a string for each CMS ID; each string contains up to three lines to be displayed on the signs themselves. For example, they can be travel times for up to three alternate routes to the same destination, or up to three different destinations reachable from the current CMS location.

Figure 6.2 shows the major components of the system. On the Berkeley side, the three drum-shaped objects represent database tables. The box labeled “Travel time prediction” represents the Perl script that performs the predictions every five minutes. It produces a text file that can be accessed through the Web. On the TMC end, the software responsible for displaying messages on CMS’s is called Satellite Operation Center Command System (SOCCS). SOCCS communicates with CMS’s via telephone lines. A simple interface program fetches the CMS messages from PeMS periodically and tells SOCCS to display them on the appropriate signs. Notice that this report only describes what happens on the PeMS side; the TMC side is handled by Caltrans.

## 6.7 Testing under operating conditions

The Perl travel time prediction program was tested for several weeks. It last ran for one week between 10/21/2003 and 10/28/2003 without failing. The program currently calculates travel time predictions for two CMS's each containing three destinations. This is shown in Table 6.1. The

CMS location	Line number	Destination (distance)
I-80 EB at University Ave.	1	Hilltop Dr. (6.2 mi)
	2	SR-4 (10.3)
	3	Carquinez Br. (14.4)
I-80 WB at Appian Way	1	University Ave. (7.8)
	2	Powell St. (11.9)
	3	Toll Plaza (13.7)

**Table 6.1 CMS and destinations on Interstate 80 in Bay Area.**

program wakes up every five minutes to compute the travel times on each of the six routes. This takes about 30 seconds each time on a 2.8GHz Pentium 4 PC running Linux, with 1GB of memory. This figure extrapolates to about 20 CMS's at five-minute update intervals. Note, however, that this system can be optimized further to accommodate more calculations. We do not expect computation to be a limiting factor because the calculations to predict travel times are simple operations.

## Chapter 7

# Conclusion and recommendations

We designed and tested a system to display real time travel times on Changeable Message Signs (CMS). CMS's are effective for displaying real time traveler information because their content has great relevance to their audiences. The most important things for a driver to know are travel times and alternate routes. We demonstrated that such information can be displayed on existing CMS's.

We also demonstrated that route travel times can be accurately estimated using data from existing traffic detector stations. These are the components of a travel time prediction system: data collection, data processing, travel time estimation, and travel time prediction. We leverage the developments of the Freeway Performance Measurement System (PeMS), which already collects and cleans Caltrans detector data. We implemented a system that uses existing algorithms for travel time estimation and prediction.

The potential benefits of travel time prediction on CMS's are reduced average travel times and more predictable travel times. Using historical data from several freeway routes with CMS's, we evaluated the improvement in travel time if route choice was based on real time travel time predictions. We found potential reductions of 1% - 17% in average travel times. While our predictions have errors, they are much more accurate than predictions based on historical data alone. This reduction in uncertainty produces large reductions in the buffer time, a more realistic measure of travel cost. This reduction is between 10% and 30%.

The travel time prediction system is implemented in Matlab and Perl and is integrated into PeMS. CMS locations and routes are defined in database tables. An automated process computes travel times and formats CMS messages for each CMS in the database every five minutes. The formatted messages are made available in a text file, which is accessed by the Caltrans TMC to be displayed on the appropriate signs. The process on PeMS's side is fully functional and has been shown to be reliable.

## 7.1 Follow-up project

This system was not deployed as planned. We recommend a follow-up project that builds on the work done here and deploys the system. Several tasks remain; chief among them is data quality. We found that many detector stations are not working, causing travel time estimates to be inaccurate during some periods. We can identify the location of these detectors, but it is up to the appropriate authorities to fix them.

The process to display the formatted messages onto CMS's is in the real of Caltrans TMC. This process has not been completed and needs to be built and tested. This shouldn't be too hard.

The calculation of prediction coefficients can be made more automatic. Currently, some manual steps are needed to update them weekly or monthly.

Travel time prediction on CMS's allows drivers to make informed decisions. It is a valuable service, and is available in many areas in the country and world. We implemented a functional CMS system for California. This system uses existing detection capabilities and demonstrates quantifiable benefits to drivers. However, data quality need to be improved for it to realize its full potential.



# Bibliography

- [1] Washington State DOT website. <http://www.wsdot.wa.gov/pugetsoundtraffic/faq/>, accessed 7/08/2003.
- [2] M. Carter. Metropolitan model deployment initiative san antonio evaluation report. Technical report, Federal Highway Administration, May 2000.
- [3] Geogia Navigator. <http://www.georgia-navigator.com/about.shtml>, accessed 7/08/2003.
- [4] Houston TranStar traffic information. <http://traffic.tamu.edu/aviinfo/avi-tech.html>, accessed 7/08/2003.
- [5] Benjamin Coifman. Estimating travel times and vehicle trajectories on freeways using dual loop detectors. *Transportation research. Part A, Policy and practice.*, 36A(4):351–364, May 2002.
- [6] J.W.C. van Lint and N.J. van der Zijpp. An improved travel-time algorithm using dual loop detectors. In *Proceedings of 83rd Transportation Research Board*, Washington, DC., January 11-15 2004.
- [7] Erik van Zwet. A simple and effective method for predicting travel times on freeways. In *2001 IEEE Intelligent Transportation Systems Proceedings*, pages 227–232, Oakland, CA, 2001.
- [8] Z. Jia, B. Coifman, C. Chen, and P. Varaiya. The PeMS algorithm for accurate, real-time estimates of  $g$ -factors and speeds from single loop detectors. In *2001 IEEE Intelligent Transportation Systems Proceedings*, pages 527–532, Oakland, CA, 2001.
- [9] T. Oda. An algorithm for prediction of travel time using vehicle sensor data. In *Third International Conference on Road Traffic Control*, pages 40–44, London, England, 1990. Institution of Electrical Engineers.
- [10] Chao Chen. Travel time reliability as a measure of service quality. In *Transportation Research Record*, Washington, DC, 2003. TRB, National Research Council. In press.
- [11] K. Small, R. Noland, X. Chu, and D. Lewis. Valuation of travel-time savings and predictability in congested conditions for highway user-cost estimation. Technical Report 431, National Cooperative Highway Research Program, TRB, National Research Council, Washington DC, 1999.

- [12] David Shrank and Tim Lomax. The 2002 urban mobility report. Technical report, Texas Transportation Institute, Texas A&M University, 2002.
- [13] Chao Chen. Detecting errors and imputing missing data for single loop surveillance systems. In *Transportation Research Record*, Washington DC, 2003. TRB, National Research Council. In press.