A General Purpose Methodology for Link Travel Time Estimation using Multiple Point Detection of Traffic

UCI-ITS-TS-WP-01-5

Cristián E. Cortés
Riju Lavanya
Jun-Seok Oh
R. Jayakrishnan

Department of Civil and Environmental Engineering and Institute of Transportation Studies
University of California, Irvine
cortes@uci.edu, rlavanya@uci.edu
jun@translab.its.uci.edu, rjayakri@uci.edu

November 2001

Institute of Transportation Studies
University of California, Irvine
Irvine, CA 92697-3600, U.S.A.
http://www.its.uci.edu
A General Purpose Methodology for Link Travel Time Estimation Using Multiple Point Detection of Traffic

Cristián E. Cortés, Riju Lavanya, Jun-Sook Oh and R. Jayakrishnan

Department of Civil Engineering and Institute of Transportation Studies
University of California, Irvine

ABSTRACT

This paper develops a methodology to find appropriate travel times for highway links using data from point detectors that could be at various points within the link, or could even be outside the link. The travel times are found using a definition that the appropriate value is the one experienced by a virtual vehicle reaching the mid-point of the link at the mid-point of the time step. A simple iterative scheme is proposed to find the travel time profiles. The accuracy of the scheme will depend on whether aggregated detector data or whether individual vehicle spot speeds are used. Comparison of estimated travel times with actual experienced travel times of all vehicles in a microscopic simulation shows the technique to give very good results, comparable to having a high number of probe vehicles reporting travel times.
1. INTRODUCTION

With the advent of route guidance systems, the proper calculation and prediction of link travel times over freeway networks has become an important issue. The existing infrastructure at most urban contexts is for point-detection of traffic, mostly using pavement loop detectors. Area-detection to get section-measures such as travel times would require more expensive alternatives. Even alternatives such as video detection would perform better in detecting vehicles and estimating their speeds at a particular location (as in point-detection) rather than over longer stretches. This means that other techniques are required to estimate travel times over stretches of highways based on the data detected at point locations. This paper develops a simple general-purpose methodology to combine the data detected at multiple locations such as successive loop-detector or video-detection stations to estimate the travel times over a network link, which may include a series of such stations.

For accurate travel time estimation, many advanced techniques have been applied (1): electronic distance-measuring instruments, computerized and video license plate matching, cellular phone tracking, automatic vehicle identification, automatic vehicle location, video imaging, etc. However, existing surveillance systems rely on inductive loop detectors, mostly single loop detectors. Realizing the limitations and problems in speed estimation using single loop data, several researchers have tried to develop better algorithms for accurate speed estimation (2, 3, 4). Also, there have been some efforts to estimate travel time using single loop detectors (5-9). The basis of these works was stochastic models of traffic flow and estimated travel times by investigating traffic flows.

In this paper, we propose a relatively straightforward method to compute link travel times that is primarily applicable for freeway situations, based on traffic information that is already available to us from point detection stations. The method can be applied to various levels of accuracy—the most accurate being the case when individual vehicles’ spot speeds are available. If only aggregated speeds (for 30 seconds or a minute) are available, as is the usual situation, the technique can still be used. Averaging parameter value can be used for combining multiple detectors’ speed data. Alternatively, more accurate results can be obtained by calibrating averaging parameters from known ground truth data for a given network context as well. Based on preliminary simulation results, the technique shows promising results when compared to benchmark values obtained from simulated GPS probe data. The scheme could potentially be made better with stochastic and statistical techniques, but we have not attempted this in this paper.

2. BACKGROUND AND METHODOLOGICAL APPROACH

In this section we introduce a simple method to compute freeway link travel times based on single-loop detector data, which is already available. The development of our methodology is theoretically strict and neat, however, in terms of application, some assumptions have to be made in order to implement it in the real world considering the limited data provided by detector loops just as they are set nowadays. We shall mention these simplifications in section 4, in the context of the numerical application. In section 2.1 we describe the basic methodology, based on a two-detector case. In section 2.2 we extend that to a general n-detector case. Finally, in section 2.3 we describe the iterative processes required in order to apply this general methodology.

2.1 Basic Methodology: Two-detector Case

The most common variables used for surface streets and freeway control analysis are: flow (volume), occupancy (used as a surrogate for density), speed, headway, and queue length. Generally, loop detectors or video detection stations sense these point traffic variables. We first describe the well-known basic relationships.

The basic definition of speed is the distance traveled by a vehicle per unit time. Either one or two detectors can measure speed, using the following expression:

\[ V = \frac{3.6 \times 10^4 d}{5,280 (t_2 - t_1)} \]  

(1)

where for one detector case

\[ V = \text{speed, in mph} \]
\( \bar{u} = \text{mean vehicle length plus effective loop length, in ft} \)
\( t_1 = \text{time when detector turns on, in milliseconds (ms)} \)
\( t_r = \text{time when detector turns off, in ms} \)

When only one detector is used, a mean vehicle length must be assumed. Because the actual vehicle length varies considerably, using two detectors yields more accurate speed measurements. However, speed traps at representative locations can be used periodically to update the average effective vehicle length by the passage-time method a single-detector locations.

We are interested in the estimation of the space-mean speed over a section of roadway, which is the total distance traveled by all vehicles divided by the total time of travel. This can be measured over a small length on the roadway over a finite "long" time period, or over a finite "long" stretch of roadway for a certain small time period. Both would be the case of a "point" loop detector and the latter would be the case of a pulse camera with short time-lapse. A "true" mean speed can also be defined over time and space, by using the total travel time and total travel distance of all vehicles such as a video record of a highway stretch over a certain time period.

Consider for example a time interval \( t = t_1 - t_r \), corresponding to the time-lapse between sequential photographs at time \( t_1 \) and \( t_r \) (considering a section of road), with each vehicle \( i \) moving \( \Delta x_i \) distance. The space-mean speed \( \bar{u}_i \) can be calculated for \( K \) vehicles in the stretch as

\[
\bar{u}_i = \frac{1}{K} (u_1 + u_2 + \ldots + u_K) = \frac{1}{K} \left( \frac{\Delta x_1}{t_r} + \frac{\Delta x_2}{t_r} + \ldots + \frac{\Delta x_K}{t_r} \right)
\]

(2)

This is the space mean speed, at estimate of which can be made with a harmonic mean of "spot" speeds if only point detection such as at loop detectors or speed traps are available. In this case, we can also compute the space mean speed estimate at detector \( i \) at time \( t_1 \) using the expression:

\[
\bar{u}_i(t_1) = \frac{N_{i(i,-\Delta t)}}{\sum_{i=1}^{N} \bar{u}_i}
\]

(3)

where \( N_{i(i,-\Delta t)} \) is the number of vehicles crossing detector \( i \) in the time interval \([t_1-\Delta t, t_1]\).

We can actually construct the space-mean speed and travel time "profiles" for each detector for any time \( t \). This is done by considering the cars crossing for detector in a period \( \Delta t \) before the time point \( t \). We, of course, assume here that the flow is homogeneous and aggregated over all the lanes. In the practical situation, where the detectors report only the aggregated space-mean speeds, the profile can be made only for every time steps of aggregation where the \( \Delta t \) is the aggregation interval (say 30 sec or 1 minute). If individual speed drops are available, then our methodology in the following sections can be based on harmonic means calculated over the previous \( \Delta t \) at any point in time.

It is not always reasonable to assume that a point measure of speed at a given location on the road can represent the mean speed for the whole section. With traffic data coming in from more than one detector on a link, we can make a better speed approximation, which will result in more precise travel-time information. The basic methodology can be stated as follows: collect the data from loop detectors, particularly instantaneous vehicle speed (spot speed measures). We can define two time variables, \( \Delta t \), which represents the time period over which the calculated average travel times are assumed to exist, and \( \Delta \bar{t} \), which will be the assumed as our speed aggregation time interval. In other words, link travel times will be updated every \( \Delta \bar{t} \) seconds. The space mean speed at any
time \( t \) is calculated using the above-mentioned harmonic mean, considering all the vehicles reaching the detector location within \( \delta t \) seconds before time \( t \). Again, in the practical case, this \( \Delta t \) may be a multiple of the speed aggregation interval of the detector hardware, but our methodology can use any \( \Delta t \) value.

Suppose that we have the following detector locations along a section of a one-way freeway (FIGURE 1):

\[ <INSERT \ FIGURE \ 1 \ HERE> \]

In this figure, consider the travel time for a vehicle leaving detector 1 at time \( t \) and crossing the distance \( X_{12} \). As the speeds may change during this travel, the estimate made at detector 1 may not be applicable further downstream, and the vehicle may experience a different travel time than estimated. A reasonable estimate of the space-mean speed over the entire road section of distance \( X_{12} \) for any arbitrary vehicle that crosses detector 1 at time \( t \) can be computed as a linear combination of the harmonic mean estimates of all individual speed counts at detectors 1 and 2, over time intervals \([t - \delta t, t]\) and \([t + \Delta t - \delta t, t + \Delta t]\) respectively, where \( \Delta t \) is the time required by that vehicle to cross the stretch 1-2.

Let us assume that the space mean speed over the stretch \( X_{12} \) is given by a linear combination of the space mean speeds at 1 and 2. That is, the speed over the length of link is dependent on the conditions in front and behind the vehicle. For a virtual vehicle leaving 1 at \( t_1 \), we therefore claim that the “experienced” space mean speed is given by:

\[
\bar{v}(t_1) = \alpha \bar{u}_1(t_1) + (1 - \alpha) \bar{u}_2(t_1 + t_2)
\]

where \( \bar{v} \) is an estimate of the space mean speed for stretch 1 to 2 at time \( t_1 \), defined to be the speed experienced by a virtual vehicle that left 1 at \( t_1 \), and \( \alpha \) is an averaging parameter. The parameter \( \alpha \) could be 0.5, which will yield a simple average. It may however be more reasonable to use a value such as 0.6 or 0.55, because the travel speed is probably closer to the average calculated at the detector behind (which included the speeds of some vehicles immediately ahead of this virtual vehicle). Alternatively this factor could be context-specific and could be calibrated, which we describe later.

\( \bar{u}_1, \bar{u}_2 \) are computed using expression (3) as follows:

\[
\bar{u}_1(t_1) = \frac{\sum_{j=1}^{N_1(t_1)} 1}{\sum_{j=1}^{N_1(t_1)}} \frac{1}{u_j}
\]

\[
\bar{u}_2(t_1 + t_2) = \frac{\sum_{j=1}^{N_2(t_1, t_2)} 1}{\sum_{j=1}^{N_2(t_1, t_2)}} \frac{1}{u_j}
\]

This requires us to know the travel time \( t_2 \) so that \( \bar{u}_2 \) can then be calculated at the appropriate point in time. This can be done recursively, because we know the following expression:
\[ t_{12} = \frac{X_{12}}{S_2(t_1)} = \frac{X_{12}}{\alpha R(t_1) + (1 - \alpha) R(t_1 + t_{12})} \] (7)

Finding the travel time \( t_{12} \) that is consistent with the two sides of equation (7) requires the implementation of an iterative algorithm (see section 2.3 for details). The iterations can be expected to converge to unique \( t_{12} \) values unless the traffic variations are too drastic, close to violating the well-known first-in-first-out requirement for average traffic, and in our experience convergent solutions are easily obtained. In the next section, we extend our methodology to an \( n \)-detector case (defining a freeway link).

2.2 Freeway Link Travel Time Estimation: \( n \)-detector Case

We first describe the definitions of the term "link" we have used. Essentially any stretch of freeway, where a separate travel time measure is to be found, is a link. As such, a link could be between two milepost points, between two entries or exits, or between two locations where the freeway geometry or other details change. In other words, a link can be any stretch that may be coded as a link on any computer model of the freeway stretch. We can estimate the travel time on a link using the following method provided it is between a minimum of two detectors whose traffic data is applicable towards estimating the link’s travel time. One detector station each beyond either end point of the link is a requirement for the method to be used. There is no limit on the number of detector stations within any link as well.

Any link could be part of separate paths, when we consider travel times being used for purposes such as route guidance, however. Consider the situation in FIGURE 2, where two diverging paths are traversing a link. In this case the travel times across the link could indeed be different for each path, especially in the practical situations such as when lanes diverge to separate freeways. We assume that separate travel times for the link corresponding to each path will be calculated based on the appropriate detectors beyond the stretch.

<INSERT FIGURE 2 HERE>

Now consider a general case of a freeway link having four detector stations associated with it, two within the link and two beyond the end points, as depicted in the following figure:

<INSERT FIGURE 3 HERE>

We are interested in estimating the travel time from the origin O to the destination D for a time interval \( [t_0, t_0 + \Delta t] \), based on loop-detector data. Since this is a dynamic calculation, time profiles of estimated space-mean speeds from all four detectors are used in the following methodology.

First we must define a consistent way to compute a link travel time estimate representative of each time interval. We suggest that the most representative link travel time during time interval \( [t_0, t_0 + \Delta t] \) is that experienced by a virtual vehicle that crosses the mid-point of the link at the middle point of the interval; that is at \( t_0 + \Delta t/2 \). Let us first divide the link into two pieces of equal length \( y \), such that \( t_{00} = 2y = X_{00} + X_{32} + X_{13} \).

Note that, the known point (reached at \( t_0 + \Delta t/2 \) ) is now the mid-point of the link, so we start our calculations from there and move backward and forward accordingly, till we reach the link extreme points. Based on the methodology developed earlier and by doing some analytical work, we can rewrite the equations above to get the following set of equations:
\[ t_{i2} = \frac{X_{i2}}{\bar{S}_{i2}[\overline{t}_i(t_0 + \Delta t/2 - t_{f2} - X_{i2})] + \overline{u}_i(t_0 + \Delta t/2 - t_{f2})} \]  

(8)

\[ t_{f2} = \frac{X_{f2}}{\bar{S}_{f2}[\overline{t}_i(t_0 + \Delta t/2 - t_{f2})] + \overline{u}_i(t_0 + \Delta t/2)} \]  

(9)

\[ t_{b2} = \frac{X_{b2}}{\bar{S}_{b2}[\overline{t}_i(t_0 + \Delta t/2) + \overline{u}_i(t_0 + \Delta t/2 + t_{b2})]} \]  

(10)

\[ t_{iD} = \frac{X_{iD}}{\bar{S}_{iD}[\overline{t}_i(t_0 + \Delta t/2 + t_{b2} + \overline{u}_i(t_0 + \Delta t/2)]} \]  

(11)

In this example the midpoint falls between detectors 2 and 3, splitting stretch 2-3 into two equal pieces \( X_{f2} = y - X_{i2} \) and \( X_{b2} = X_{i2} + X_{iD} - y \). In addition, we have two new travel time estimates \( t_{f2}, t_{b2} \), for the stretch behind and ahead of the midpoint of the link to the closest detector locations, respectively.

Expressions (8) to (11) is a set of 4 equations with 4 unknowns, which can be solved iteratively according to the methodology explained later. The denominators \( \bar{S}_{iD}, \bar{S}_{f2}, \bar{S}_{b2} \) can be solved using expression (4). Now that we have a framework for calculating travel time for the basic case, we generalize it for a link with \( m \) detectors.

The method can be extended to the general case of \( m \) detectors. We have \( n \) detectors behind the midpoint of the link and \( m-n \) detectors ahead of it. Using the same methodology as above, we can write the following equations:

\[ t_{i2} = \frac{X_{i2}}{\bar{S}_{i2}[\overline{t}_i(t_0 + \Delta t/2 - (t_{f2} - \sum_{j=2}^{n} t_{f,\text{prev}} + t_{f,\text{next}}))] + \overline{u}_i(t_0 + \Delta t/2 - (\sum_{j=2}^{n} t_{f,\text{prev}} + t_{f,\text{next}}))} \]  

\[ \vdots \]

\[ t_{f,\text{next}} = \frac{X_{f,\text{next}}}{\bar{S}_{f,\text{next}}[\overline{t}_i(t_0 + \Delta t/2 - t_{f,\text{next}})] + \overline{u}_i(t_0 + \Delta t/2]} \]
\[
\begin{align*}
\text{for the pair of detectors } n \text{ and } n+1 \text{ between which the link mid-point falls.} \\
1_{n+1} &= \frac{X_{n+1}}{\bar{s}_{n+1} \left[ \bar{u}_n \left( t_0 + \Delta t / 2 + \sum_{k=1}^{n+1} t_{n,k} + th_{n+1} \right) + \bar{u}_n \left( t_0 + \Delta t / 2 + \sum_{k=1}^{n+1} t_{n,k} + th_{n+1} \right) \right]} \\
&\vdots \\
1_{n+1,n} &= \frac{X_{n+1,n}}{\bar{s}_{n+1} \left[ \bar{u}_n \left( t_0 + \Delta t / 2 + \sum_{k=1}^{n+1} t_{n,k} + th_{n+1} \right) + \bar{u}_n \left( t_0 + \Delta t / 2 + \sum_{k=1}^{n+1} t_{n,k} + th_{n+1} \right) \right]} \\
\end{align*}
\]

Note that if the link mid-point has only one detector (the one beyond the end point) ahead, then the 11 equation replaces the last equation. Similarly if the mid point has only one detector behind it (the one upstream of its start point), then the \( n \)th equation replaces the first equation in the set. There could be even a case when the link has no detectors within it, in which case the 11 and \( n \) th equations would be the only ones used.

These equations can be solved using the iterative algorithms that are described in the next section. Now, there are different possibilities, depending on the type of data available from the detectors. The best results would of course be if individual spot speeds were available from the detectors. Loop detector stations measure the off and on time points for the vehicle inductance pulses, but are normally set up only to and report aggregate occupancy, volume and speeds (after local computation) back to the traffic management center. While double-loop detectors can directly measure vehicle speeds using so called “speed trap”, single-loop detectors do not directly provide spot speed information. However, there have been some efforts to estimate speed from single-loop measurements (4).

Thus there are two ways in which the above expressions can be used.

Case 1: (individual spot speeds available). If such data were available, we can use the above expressions with travel time profiles for any time, as we work back from the mid-point of the link at the mid-point of the \( \Delta t \) time interval. For this case, we calculate the harmonic mean of the previous \( \Delta t \) period at any time point when we look at a detector’s mean-speed value in the above expressions. Note that it may be very difficult to add such capabilities to existing detector hardware in the future, now that communication of the data is perhaps not as prohibitively expensive as in the past. Also, any point video detection station would be able to report such data.

Case 2: (individual speeds not available). In this case, we lose potential accuracy, as we only have the aggregated mean-speed values for every aggregation interval available. The method works without any problems,
and in fact the results reported in the next section show that it provides very good results as well, though not as good as in Case 1.

3. SOLUTION ALGORITHMS

In this section we describe the specific iterative algorithms to solve and compute link travel times according to the methodology presented in the previous section. The algorithms are simple for implementation, and in our experience they have always converged easily to the proper solutions. Next, we show an iterative algorithm to solve equation (7). Next, we extend the iterative process to compute link travel times based on the methodology presented in section 2.1, for the example case summarized by equations (8)-(11), and extendible to the most general case of equation (12).

3.1 Basic iterative algorithm: two-detectors case

The simple expression for computing the travel time required by an arbitrary vehicle to cross a link of length \(X_{ij}\) is given by expression (7), assumed that the vehicle leaves detector 1 at time \(t_i\). The denominator of expression (7) is a linear combination, which depends on the left-hand side variable, so an iterative process is needed to solve it. Thus, assumed known \(\alpha, t_i, X_{ij}\), we use the following algorithm:

1) Find a feasible initial value for \(t_{ij}^{(0)} = \bar{u}_i(t_i)\), i.e., we assume (for the first iteration) that the traffic conditions over the stretch depend only upon upstream traffic variables (captured from detector \(t_i\) equivalent to set \(\theta_i = 1\)). \(\bar{u}_i(t_i)\) is calculated using expression (3) in section 2.1. In addition, set \(k = 1\) and precision \(\zeta = \infty\).

2) While precision\(_{k+1}\) > \(\zeta\) do

(a) Compute \(t_{ij}^{(k)} = \frac{X_{ij}}{\sum_{n=1}^{N} \bar{u}_n(t_i, \bar{u}_i(t_i), t_{ij}^{(k-1)})}\)

(b) Compute \(\bar{u}_i(t_i + t_{ij}^{(k)})\) using expression (3) in section 2.1

(c) Compute the new value of \(t_{ij}^{(k+1)} = \alpha \bar{u}_i(t_i) + \left(1 - \alpha\right) \bar{u}_i(t_i + t_{ij}^{(k)})\)

(d) Compute precision\(_{k+1}\) = \(\frac{t_{ij}^{(k)} - t_{ij}^{(k+1)}}{t_{ij}^{(k)}}\)

(e) Set \(k = k + 1\)

3) Stop.

The value assumed for \(\zeta\) will determine the accuracy of the calculation. Let’s call it Algorithm 1.

3.2 Extended iterative algorithm: link travel time estimation

We will extend our algorithm in order to solve the case presented in the previous section, summarized in expressions (8)-(11). Briefly, here we shall develop Algorithm 2, which depends on Algorithm 1 presented above. Algorithm 2 is easily extendible towards the most general case of expression (12).

Now, assumed known \(\alpha, t_i, \Delta t, X_{ij}, X_{j}, X_{i}, X_{j}^{CD}, X_{i}^{CD}\), we propose the following algorithm to find \(t_{ij}^{*} = t_{ij} + t_{ij} + t_{ij}^{*}\).
1) Solving for stretch $X_{23}$:
   (a) Compute $\bar{t}_{23}$ using Algorithm 1, starting from the initial feasible value for $\bar{x}^1 = \bar{u}(t_0 + \Delta t/2)$, i.e., we assume (for the first iteration) that the traffic conditions over the stretch $X_{12}$ depend only upon upstream traffic variables (obtained from detector 2).
   (b) Compute $\bar{t}_{23}$ using Algorithm 1, starting from the initial feasible value for $\bar{x}^0 = \bar{u}(t_0 + \Delta t/2)$, i.e., we assume (for the first iteration) that the traffic conditions over the stretch $X_{23}$ depend only upon downstream traffic variables (obtained from detector 3).

2) Solving for stretches upstream $X_{12}$:
   (a) Once $\bar{t}_{23}$ has been calculated (from 1b), compute $\bar{t}_{12}$ using Algorithm 1, starting from the initial feasible value $\bar{x}^2 = \bar{u}(t_0 + \Delta t/2 - \bar{t}_{23})$, downstream influence initially.

3) Solving for stretches downstream $X_{23}$:
   (a) Once $\bar{t}_{23}$ has been calculated (from 1a), compute $\bar{t}_{23}$ using Algorithm 1, starting from the initial feasible value $\bar{x}^0 = \bar{u}(t_0 + \Delta t/2 + \bar{t}_{23})$, upstream influence initially.

This generalization of Algorithm 2 to case (12) is straightforward.

3.3 Averaging Parameter

The parameter $\alpha$ in expression (3) has some effect on the results from using the above methodology. The ideal value for the parameter for any stretch at any time actually would depend on the nature of the traffic variation. Thus it is somewhat similar to the smoothing factors used in macroscopic simulations where the value actually depends on the direction and magnitude of the characteristic wave or shock wave speed. In our case, the best value could also depend on the time periods of aggregation and the distances between the detectors. In practice, unless we find an elaborate scheme for online update of this parameter, we will need to use a fixed value for any link. Our experience shows that even a simple value of 0.5 works well and perhaps a value of 0.55 giving a little more weight to the detector behind may result in better numbers. We conducted some preliminary studies based on macroscopic simulation on the real network in the case study described in the next section, and attempted to derive a simple regression for the best value, using the complete true travel time averages available from the simulation. The simple regression model is,

$$\bar{x}_2 = \alpha \bar{x}_1 + (1 - \alpha) \bar{x}_3 + \nu_2$$  \hspace{1cm} (13)

With a normal distribution is assumed for the error term $\nu$ we get a traditional linear regression model.

The case study results show the use of the resulting parameter value (calibrated for the whole simulation network, not just the link we focus on), as well as a simple equal weight averaging.

4. CASE STUDY

In this section, we show a numerical example of the proposed algorithm. As it is virtually impossible to find the ground-truth actual travel time of all vehicles on a realistic urban freeway stretch, the case study has to be based on a microscopic simulation. We used a microscopic traffic simulation model, Paramics (PARAllel MICroscopic...
Simulation) for data collection. **Paramics** is a suite of high-performance software tools for microscopic traffic simulation. The movement and behavior of individual vehicles are modeled in detail for the duration of their entire trip, providing reasonably accurate and dynamic information about traffic flow, speed, and congestion. The primary detail in Paramics that affects the value reported from the simulated detectors is the car following model and the resulting traffic dynamics. The model used in Paramics is as in (10) and limited calibrations and validations of the Paramics model have shown that the traffic simulations are acceptable. A perfect case study for our algorithm would not be based on microscopic simulation, but would rather be with complete travel time profiles of all vehicles on a stretch of highway. Such data is currently unavailable, and thus we have to rely on the microscopic simulation.

The study site for the analysis is a stretch of Interstate freeway 405, Irvine, California. FIGURE 4 shows the stretch of freeway. The freeway section is southbound direction from Culver to Jeffrey, having a total of six mainline loop detectors in it. Paramics simulation was based on traffic input volumes matching the measurements from actual freeway detectors using the capabilities in the ATMS Research Testbed at University of California, Irvine. One set of data was simulated for a P.M. peak two-hour period (4:00 – 6:00 p.m.).

We compare the application of our methodology with a travel time estimation from benchmark-simulated data, capturing the exact time-space location of all vehicles crossing the link during a specific time interval (probe data), tagged at different spots along the freeway link. We also compare that with the same individual vehicle data, but assuming known only the time-space conditions of a percentage of the total vehicles crossing the link in a certain time interval.

We perform a Monte Carlo simulation in order to compute statistics for the travel time distributions obtained from simulating those percentages of the probe data. Finally, we also find the worst case estimates, based on an average of the conditions at the link extreme points over the required time interval.

<INSERT FIGURE 4 HERE>

### 4.1 Design of Simulation Experiment

From the Paramics microscopic simulation we obtained two different data sets: Individual vehicle data and field information from the loop detectors as shown in FIGURE 4.

The former set corresponds to the specific time instant at which vehicles cross certain defined points, including detector locations along with some additional intermediate points. This data set is assumed to be our “Probe” simulated data, and is used for benchmarking purposes. The latter set is simply simulated information, of the same level of detail that we can now get from current loop-detectors (counts and occupancy aggregated every 30 seconds). This latter data set will be the main input for applying our methodology.

We also consider different Probe data sample sizes. The benchmark is, of course, the 100% data set. In addition, we consider samples of 0.5%, 1% and 5%, which in terms of number of vehicles are quite significant. In order to compute the sample mean travel time for each sample size experiment, we performed 50 Monte Carlo replications at any case. We also report a “Rough” case, which is basically the result of averaging the estimated speeds associated only with the loop detectors located at the link extreme points.

The averaging parameter \( \alpha \) was calibrated using freeway data from the same section, including not only detector pairs over the selected stretch, but over the whole freeway network. We tried to incorporate detector pairs of different geometry and features, in order to perform a linear regression (as in equation 13), representative of a wide urban area, and therefore, extendable to different cases under similar conditions. Calibration results are shown in the next section along with the final travel time estimations and performance indicators.
As a performance measure, a mean absolute percentage error (MAPE) is calculated by comparing the case of 100% Probe data. MAPE is computed as follows:

$$ MAPE = \frac{\sum_{i=1}^{N} \left( \frac{ATT_{obs,i} - ATT_{est,i}}{ATT_{obs,i}} \right) \times 100}{N} $$

(14)

where

- $ATT_{obs,i}$: Observed average travel time at time step $i$ (based on 100% Probe data)
- $ATT_{est,i}$: Estimated average travel time at time step $i$ (it depends on the specific methodology)
- $N$: Total number of time steps. This number is dependent on the step size interval $\Delta t$. (For our experiments we used 1 and 3 minutes)

In the next section we report a summary of our numerical simulations, emphasizing important issues such as performance of each method compared to the benchmark.

4.2 Numerical Results

In this section we report the results of our experiments. We consider two scenarios in order to test the efficacy of our methodology under different traffic conditions. The first scenario ($S_1$) represents medium traffic conditions, where the network is not significantly congested. In the second scenario ($S_2$), we use the same demand level as in $S_1$, but here, we inject an incident in the stretch of highway under consideration. The duration of this incident is 20 minutes and causes large variations in travel time because of high congestion. In this case the incident is very severe, resulting in new travel times which are more than twice the travel times found in $S_1$ for certain time periods.

In FIGURE 5, we show how all the simulated methods perform for $S_1$, based on the MAPE calculation described above in expression (14). These calculations are compared always with the 100% Probe data usage. We consider two link travel-time reporting intervals (1 and 3 minutes for $\Delta t$) and use a speed aggregation interval size of 30 seconds at the detectors.

4.2 Numerical Results

In this section we report the results of our experiments. We consider two scenarios in order to test the efficacy of our methodology under different traffic conditions. The first scenario ($S_1$) represents medium traffic conditions, where the network is not significantly congested. In the second scenario ($S_2$), we use the same demand level as in $S_1$, but here, we inject an incident in the stretch of highway under consideration. The duration of this incident is 20 minutes and causes large variations in travel time because of high congestion. In this case the incident is very severe, resulting in new travel times which are more than twice the travel times found in $S_1$ for certain time periods.

In FIGURE 5, we show how all the simulated methods perform for $S_1$, based on the MAPE calculation described above in expression (14). These calculations are compared always with the 100% Probe data usage. We consider two link travel-time reporting intervals (1 and 3 minutes for $\Delta t$) and use a speed aggregation interval size of 30 seconds at the detectors.

FIGURE 5 and TABLE 1 illustrate the measures of performance obtained from each method, which are:

1. "Rough" method
2. Probed vehicles with varying rates
3. Proposed algorithm based on spot speeds, available in continuous time (Case 1), and
4. Proposed algorithm based on discrete, 30 second aggregated occupancy data from loop detectors (Case2)

In FIGURE 5 we show an equal-weight averaging scheme across detectors (i.e., $\alpha = 0.5$) and a “calibrated” case which is based on the value (0.56) obtained from independent simulation of the whole Irvine area network, and using a linear-regression on selected links therein, as explained in the previous section. The calibrated parameter value tells us that the average stretch speed is somewhat more influenced by the speed conditions behind, which seem to be reasonable for forward-moving traffic.

As evident from the histogram, the proposed algorithm with spot speeds data categorically outperformed all the other methods. One interesting conclusion is that, even without the spot speeds, the proposed method still performed well and compared nicely with probe rate of 5% which represents a rate of around 650 an hour for the stretch that we studied. We report the actual number of required probe vehicles itself, to convey to the reader how difficult it is to get probe vehicle data from 5% of the vehicles. Even 1% probe data requires about 150 vehicles per hour in our case study example, which is quite a challenge to attempt to get in the real world.

Notice that, at least in this case study, the calibration of the averaging parameter $\alpha$ did not seem to yield significantly better performance of the proposed system. This does not mean that better numbers cannot be obtained if separate calibrations are ever possible for individual links (or if more elaborate schemes of regression models for $\alpha$ for different types of link/detector geometry and for different traffic conditions are developed). It is an encouraging result that the simple averaging itself yields good numbers.

FIGURE 6 shows the variation of travel time of the proposed method, when comparing it with probe data at 100% (the true travel time average) and at 1%.

FIGURE 6 and FIGURE 7 report the corresponding link travel time estimates applying three candidate methods, based on an aggregation step size of one minute and 3 minutes for $S_3$ (not congested case). Note that our algorithm with spot speed data using the calibrated averaging factor (Cal. Method) seems to follow the true travel time curve without much deviation.

Note that the scenario mentioned above did not have a significant variance in travel time. In order to fully test our methodology, we need to perform similar analysis but this time with more unstable traffic conditions. These conditions are effectively captured by scenario $S_5$. The following figures and tables show the performance of different methods under this new situation, following the same steps as we did earlier.

We can see that the proposed methodology can “pick up” sharp changes in travel time without any significant lag, even under extremely unstable travel times. In addition, the accuracy of the proposed method under this scenario is again comparable to probe data sampled at 1% and 5%, and is much better than the results obtained by the “rough” methodology, which is in practice today.

In conclusion, we can mention that our algorithm performs quite well, and better than the traditional methods used by transportation agencies (which are similar to the “Rough” method we report) to compute freeway link travel times based on point detection measurements even under extreme traffic conditions. In addition, it reaches the same accuracy as that obtained from the 5% Probe data, which is very desirable.
5. CONCLUSIONS

The approach in this paper for calculating the travel time on network links using multiple detectors is a simple one and can be easily used with data reported by practical detectors. The scheme also provides for more accurate calculations, if individual spot speeds are available. The results indicate that the scheme works better than even having a relatively high fraction (say 5%) of the vehicles being equipped to report travel times as probe vehicles.

One underlying assumption in our methodology in terms of potential future field applications is that the rate of failure of the detectors involved in the calculations is not too high.

Much further theoretical developments are possible with the model, such as using statistical techniques to smooth the detector data, using traffic flow theory based concepts for finding the averaging parameter, location-based calibration of the averaging parameters etc. The results with the very simple scheme of averaging itself yields good numbers, however, and thus the method is already useful from a purely practical viewpoint.

REFERENCES


List of Tables

TABLE 1 MAPE statistics (%) for $S_1$
TABLE 2 MAPE statistics (%) for $S_2$

List of Figures

FIGURE 1 Detector locations along a Freeway Section.
FIGURE 2 Diverging paths traversing a link.
FIGURE 3 Travel time calculation.
FIGURE 4 Section of freeway used in the experiment.
FIGURE 5 Performance measures for each method under $S_1$.
FIGURE 6 Comparison of algorithms (1 min interval) $S_1$.
FIGURE 7 Comparison of algorithms (3 min interval) $S_1$.
FIGURE 8 Performance measures for each method under $S_2$.
FIGURE 9 Comparison of algorithms (1 min interval) $S_2$.
FIGURE 10 Comparison of algorithms (3 min interval) $S_2$. 
<table>
<thead>
<tr>
<th>Reporting time $S_1$</th>
<th>MAPE (1 minute)</th>
<th>MAPE (3 minutes)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rough Method</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.5% probe rate (65 vph)</td>
<td>14.22392</td>
<td>14.30983</td>
</tr>
<tr>
<td>1% probe rate (130 vph)</td>
<td>6.676243</td>
<td>7.346492</td>
</tr>
<tr>
<td>5% probe rate (659 vph)</td>
<td>6.06252</td>
<td>6.785007</td>
</tr>
<tr>
<td>Case 1 (calibrated)</td>
<td>3.13241</td>
<td>4.008362</td>
</tr>
<tr>
<td>Case 1 ($\mu = 0.5$, uncalibrated)</td>
<td>5.00725</td>
<td>3.955988</td>
</tr>
<tr>
<td>Case 2 (calibrated)</td>
<td>5.365421</td>
<td>4.185405</td>
</tr>
<tr>
<td>Case 2 ($\mu = 0.5$, uncalibrated)</td>
<td>2.97968</td>
<td>1.858380</td>
</tr>
<tr>
<td></td>
<td>2.908875</td>
<td>1.817951</td>
</tr>
</tbody>
</table>
TABLE 2 MAPE statistics (%) for $S_2$

<table>
<thead>
<tr>
<th></th>
<th>Reporting time $S_2$</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>MAPE (1 minute)</td>
<td>MAPE (3 minutes)</td>
</tr>
<tr>
<td>Rough Method</td>
<td>22.45288</td>
<td>22.10923</td>
</tr>
<tr>
<td>0.5% probe rate (65 vph)</td>
<td>7.284522</td>
<td>5.573667</td>
</tr>
<tr>
<td>1% probe rate (130 vph)</td>
<td>5.61832</td>
<td>4.222725</td>
</tr>
<tr>
<td>5% probe rate (650 vph)</td>
<td>2.17802</td>
<td>1.782642</td>
</tr>
<tr>
<td>Case 1 (calibrated)</td>
<td>9.58547</td>
<td>8.613899</td>
</tr>
<tr>
<td>Case 1 (n = 0.5, uncalibrated)</td>
<td>8.98767</td>
<td>7.754546</td>
</tr>
<tr>
<td>Case 2 (calibrated)</td>
<td>6.16328</td>
<td>5.10683</td>
</tr>
<tr>
<td>Case 2 (n = 0.5, uncalibrated)</td>
<td>5.84107</td>
<td>4.479967</td>
</tr>
</tbody>
</table>
FIGURE 1 Detector locations along a Freeway Section.
FIGURE 2 Diverging paths traversing a link.
FIGURE 3 Travel time calculation.
FIGURE 4 Section of freeway used in the experiment.
FIGURE 5 PERFORMANCE MEASURES FOR EACH METHOD UNDER S₆.
FIGURE 6 Comparison of algorithms (1 min interval) $S_1$. 
FIGURE 7 Comparison of algorithms (3 min interval) $S_I$. 
FIGURE 8 PERFORMANCE MEASURES FOR EACH METHOD UNDER $S_0$. 
FIGURE 9 Comparison of algorithms (1 min interval) S2.
FIGURE 10 Comparison of algorithms (3 min interval) $S_9$. 