Arterial Travel Time Estimation Based On Vehicle Re-Identification Using Wireless Magnetic Sensors

Karric Kwong and Robert Kavaler
Sensys Networks, Inc.
2560 Ninth Street, Berkeley, CA 94710
Tel. (510) 548-4620, Fax: (510) 548-8264
{karric,kavaler}@sensysnetworks.com

Ram Rajagopal and Pravin Varaiya[1]
Department of Electrical Engineering and Computer Sciences
University of California, Berkeley, CA 94720-1700
Tel: (510)642-5270; Fax: (510)642-1785
{ramr, varaiya}@eecs.berkeley.edu

19 July, 2008

[1]Corresponding author

Abstract

A practical scheme is described for the real-time estimation of travel time across an arterial segment with multiple intersections. The scheme relies on matching vehicle signatures from wireless sensors. The sensors provide a noisy magnetic signature of a vehicle and the precise time when it crosses the sensors. A match (re-identification) of signatures at two locations gives the corresponding travel time of the vehicle. The travel times for all matched vehicles yield the travel time *distribution*. Matching results can be processed to provide other important arterial performance measures including capacity, volume/capacity ratio, queue lengths, and number of vehicles in the link. The matching algorithm is based on a statistical model of the signatures. The statistical model itself is estimated from the data, and does not require measurement of 'ground truth'. The procedure does *not* require measurements of signal settings; in fact, signal settings can be inferred from the matched vehicle results. The procedure is tested on a 0.9 mile-long segment of San Pablo Avenue in Albany, CA, under different traffic conditions. The segment is divided into three links: one link spans four intersections, and two links each span one intersection.

Keywords: real-time travel time estimation; vehicle reidentification; arterial performance measures; queue length; discharge rate; magnetic signature

1 Introduction and Previous Work

Estimating arterial travel time is difficult. Since the movement of vehicles is interrupted by signals, estimates based on average speeds from loop detectors or radar are inaccurate.

Approaches for estimating travel times on arterial links include speed vs. volume to capacity ratio relationships or procedures based on the Highway Capacity Manual. The latter calculates average travel time as the sum of the running time, based on arterial design characteristics, and the intersection delay, based on a deterministic point delay model. These approaches are not suited for real-time applications with variable traffic conditions.

Statistical models have been proposed for estimating travel times from surveillance data. For example, Zhang (1999) estimates link-speed as a function of volume to capacity ratio and volume and occupancy measured by loop detectors. Since the estimation itself requires collection of travel times, the model is site-specific and impractical to implement.

By contrast, Skabardonis and Geroliminis (2005) develop a generally applicable kinematic wave model to construct a link travel time estimate from 30-second flow and occupancy data from an upstream loop detector and the exact times of the red and green phases. Their procedure for the case when queues clear in each cycle can be explained as follows. The upstream detector gives the number n of vehicles that arrive in a 30-sec interval, say during $[s, s+30]$. These n vehicles are assumed to cross the detector at uniformly spaced times $s+30i/n, i = 1, \cdots, n$. From the known or estimated free flow travel time T_f, these vehicles will arrive at the intersection at times $T_f + s + 30i/n$. Knowing the signal phase at these arrival times and the previously calculated queue at the intersection, and using the kinematic wave model (with known or estimated congestion wave speed and jam density), it is straightforward to figure out the delay faced by each of the n vehicles, and the queue remaining at the end of the 30 seconds. The procedure is then repeated, to yield the average travel time across the link and the average delay.

Liu and Ma (2008) use a similar model. However, they measure individual vehicle detector actuations, so they know the exact times that the vehicles crossed the detector, instead of assuming that these are uniformly spaced times. The rest of their procedure is similar. The models of individual vehicle trajectories in both Skabardonis and Geroliminis (2005) and Liu and Ma (2008) are more elaborate than the uniform free flow speed assumed above, and take into account the vehicle's deceleration as it approaches a queue and its acceleration as it departs from the signal. However, this elaboration does not measurably affect the average travel time and delay estimates (Liu and Ma, 2008, Figure 10). Both Skabardonis and Geroliminis (2005) and Liu and Ma (2008) also treat the case when arrivals are not cleared in each cycle.

The two approaches outlined above have limitations. They require precise signal phase times, and these must be synchronized with the detector times. Moreover, for a link with multiple intersections, each intersection must be instrumented. Such instrumentation is expensive. Second, both approaches require knowledge of parameters such as free flow travel time, which may not be constant across the entire range of traffic conditions, and lead to bias in the estimates. Third, *average* travel time and delay are insufficient to calculate interesting arterial performance measures provided by the scheme proposed here, as seen in Section 3.

In principle, vehicle re-identification schemes overcome these limitations. These schemes work as follows. Sensors placed at the two ends of a link record when a vehicle crosses them and measure its signature. When a vehicle's signature is matched at the two sensors, its travel time is obtained.

Signal phase information is not needed. If sufficiently many vehicles are re-identified, the travel time distribution can be estimated. Vehicles can be re-identified by matching unique tags or license plates; but besides raising privacy concerns, these schemes are too expensive to deploy over an arterial network.

Re-identification schemes for *freeway* travel times have been demonstrated. Sun et al. (1999) match waveforms from inductive loops. The waveforms are first normalized using independently measured speeds. Features from the normalized waveform pairs are extracted and compared in a multi-criterion optimization framework to obtain the best match. Coifman (1999) compares lengths of vehicle platoons at the two detector locations. The length estimate too requires independent speed measurements.

M.Ndoye et al. (2008) and Oh and Ritchie (2002) report experiments using inductive loop signatures. Again, vehicle speed is used to normalize the raw signature and produce a speed-independent signature. The speed normalization procedure assumes that vehicle speed is constant. If a vehicle is accelerating or decelerating, this assumption is invalid: as Ndoye et al M.Ndoye et al. (2008) report, the rate of correct matching then drops drastically. Oh and Ritchie (2002) only report results for a non-peak period. Neither scheme would perform well in a link with significant acceleration and deceleration, caused by traffic signals. Platoon lengths used in Coifman (1999) would not work well for the additional reason that signalized intersections would break platoons up.

Sun et al. (2004) use video images (for vehicle color) in addition to the loop-based signature and speed in a data 'fusion' algorithm that achieves a high matching rate for vehicle platoons in a link that does not span an intersection. The selection of the parameters of the fusion algorithm requires an extensive and expensive collection of 'ground truth' measurements. The fusion algorithm weights loop signature, speed, vehicle color and platoon traversal time. In the best fusion scheme, color receives a weight of 95%. The scheme is impractical.

This paper presents a scheme to estimate the travel time *distribution* of a single arterial link, spanning several signalized intersections. The scheme is based on matching individual vehicle signatures obtained from wireless magnetic sensors placed at the two ends of the link. The signature consists of the sequence of peak values of the 'raw' magnetic signal. The peak values are independent of the vehicle speed, so speed measurements are not needed.

Unlike Skabardonis and Geroliminis (2005); Liu and Ma (2008) the scheme requires *no* signal phase measurements. Indeed, signal phases, queue lengths, delay distributions and other performance measures can all be evaluated from the matched vehicles. The scheme is tested on a 0.9 mile-long segment of San Pablo Avenue in Albany, CA, spanning six intersections. The segment is divided into three links: one link spans four intersections, and two links each span one intersection. The peak hour per lane flow over the segment is 500-600 vph.

The paper is organized as follows. The test site and measurement system are described in Section 2. Test results are presented in Section 3. The matching problem and the statistical signature model used to evaluate matching algorithms occupy Section 4. The optimal unconstrained matching algorithm and the optimal constrained matching algorithm are described in Sections 5 and 6, respectively. (The results of Section 3 are based on optimal constrained matching.) A real-time version of the optimal constrained algorithm is presented in Section 7. Section 8 explains how the statistical signature model is estimated without ground truth. Section 9 collects the conclusions. Some of the more technical material is presented in the Appendix.

2 Test site and measurement system

On the left in Figure 1 is a map of the 0.9 mile-long test segment on southbound San Pablo Avenue in Albany, CA, starting at A (Fairmount) and ending at D (Buchanan). The segment is divided into three links, $A \to B$, $B \to C$, $C \to D$. Link $A \to B$ spans four signalized intersections (the three circles plus the intersection at Washington), links $B \to C$ and $C \to D$ each span one signalized intersection. Sensors at A, B, C, and D are located immediately downstream (12m) of the corresponding intersection. Thus each link has one uplink and one downstream sensor, as shown in the middle of Figure 1. For example, link $C \to D$ has its upstream sensor at C (12m downstream of the intersection at Solano) and its downstream sensor at D (12m downstream of the intersection at Buchanan).

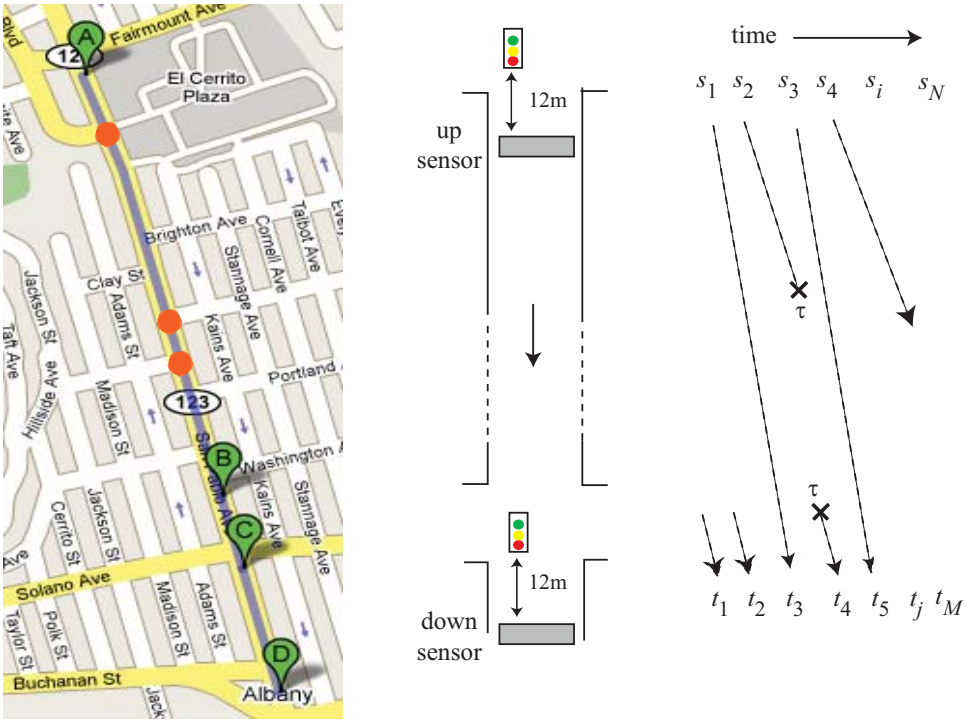

Figure 1: Test site, sensor locations, and vehicle matches.

San Pablo Avenue has two lanes, each with its own sensors. Vehicles in each lane are matched separately. The results below are for lane 1, next to the median. We now describe the data. Consider the link illustrated in the middle of the figure. During a measurement time interval, vehicles indexed $i = 1, \cdots, N$ cross the upstream sensor at times $s_1 < s_2 < \cdots$. Vehicles indexed $j = 1, \cdots, M$ cross the downstream sensor at times $t_1 < t_2 < \cdots$. The upstream sensor measures the 'signature' X_i of each vehicle i that crosses it and the corresponding time s_i. The downstream sensor measures the signature Y_j of each vehicle j that crosses it and the corresponding time t_j. Thus the measurement data consists of two arrays $\{(s_i, X_i), i = 1, \cdots, N\}$ and $\{(t_j, Y_j), j = 1, \cdots, M\}$. Vehicle speed is *not* measured. As suggested by the figure on the right, upstream vehicle 1 is the same as downstream vehicle 3, which will be denoted as $X_1 \to Y_3$; similarly $X_3 \to Y_5$. On the other hand upstream vehicle 2 has turned (either into the other lane or at the intersection) before reaching the downstream sensor, denoted $X_2 \to \tau$; similarly $\tau \to Y_4$ indicates that downstream vehicle 4 turned into the lane and did not cross the upstream sensor.

Each sensor consists of an array of seven 3"× 3"× 2" nodes, embedded in the pavement 1 foot apart, perpendicular to the direction of motion. A node has electronic circuits that incorporate into a system a magneto-resistive sensor measuring the earths magnetic field, a radio transceiver, an antenna, a microprocessor, memory and a battery. The magnetic field is distorted as a vehicle goes over the sensor. The node records the measured field sampled at a rate of 128Hz and extracts from the record a feature vector of the vehicle. The seven feature vectors constitute the vehicle's signature. The nodes transmit the time when the vehicle crossed it together with its signature via radio to a roadside 'access point'. The access point in turn transmits the data to a server using a cellular service. Data from the server are analyzed off-line. In a deployment, the data could be analyzed on-line at the access point. The nodes and access point are products of Sensys Networks, Inc. and described in Haoui et al. (2008).

As we have seen, data arriving at the server from a link during a time interval consist of the upstream array $\{(s_i, X_i), i = 1, \cdots, N\}$ and the downstream array $\{(t_j, Y_j), j = 1, \cdots, M\}$. The time interval should be so short that the travel time distribution does not change but long enough for meaningful statistical estimates. Results are reported for 30-minute intervals during the peak period (1-2PM) and 60-minute intervals in the off-peak period (11PM-12AM); during the 30-min peak period between 200 and 300 vehicles traversed each link of the test segment.

The matching of upstream and downstream arrays is done in two steps. In the first or *signal processing* step, each pair of (X_i, Y_j) of upstream and downstream signatures is compared to produce a measure of dissimilarity or distance $d(i, j) \geq 0$ between them, i.e., the signal processing algorithm implements a function δ, $d(i, j) = \delta(X_i, Y_j)$. The signal processing step thereby reduces the two signature arrays to the $N \times M$ matrix $D = \{d(i, j) \mid 1 \leq i \leq N, 1 \leq j \leq M\}$. The signal processing step is described in the Appendix. In the second step the *matching problem* is formulated and solved.

The solution to the matching problem is a set of matches of the form $i \to j$, $i \to \tau$, or $\tau \to j$, indicating respectively that upstream vehicle i is declared to be downstream vehicle j, i cannot be matched to any downstream vehicle, or j cannot be matched to any upstream vehicle. For a vehicle match $i \to j$, s_i and t_j are the times the vehicle was at the beginning and end of the link, so $(t_j - s_i)$ is its link travel time. The other matches may indicate turns. The next section discusses the solution of the matching problem for the test site.

3 Test results and analysis

We first consider the solution of the matching problem for the single link $B \to C$, and then analyze the implications of the solution. The solution is obtained using the optimal constrained matching algorithm of Section 6. Figure 2 shows one way to display the matches for link $B \to C$ during the peak hour period 1-1:23 PM on May 23, 2008. The x-axis is the time in 10^4s that a matched vehicle crosses the upstream sensor at B (top) or the downstream sensor at C (bottom). (Midnight is 0s, $4.68 \times 10^4/3600 = 13$hours or 1PM, and $4.86 \times 10^4/3600$ is 1:23PM.) The y-axis is the matched vehicle's travel time in seconds. For illustration, two matches are highlighted: one vehicle with start time s_1, end time t_1 and travel time $t_1 - s_1$, and another vehicle with corresponding times $s_{10}, t_{10}, t_{10} - s_{10}$. The travel time samples can be used to estimate *travel time distributions* as in Figure 5 below. Vehicles that are not matched are not shown in Figure 2 (but see Figure 4). Figure 2 also reveals information about delay, signal phase, queues, number of vehicles in the link, and turning movements.

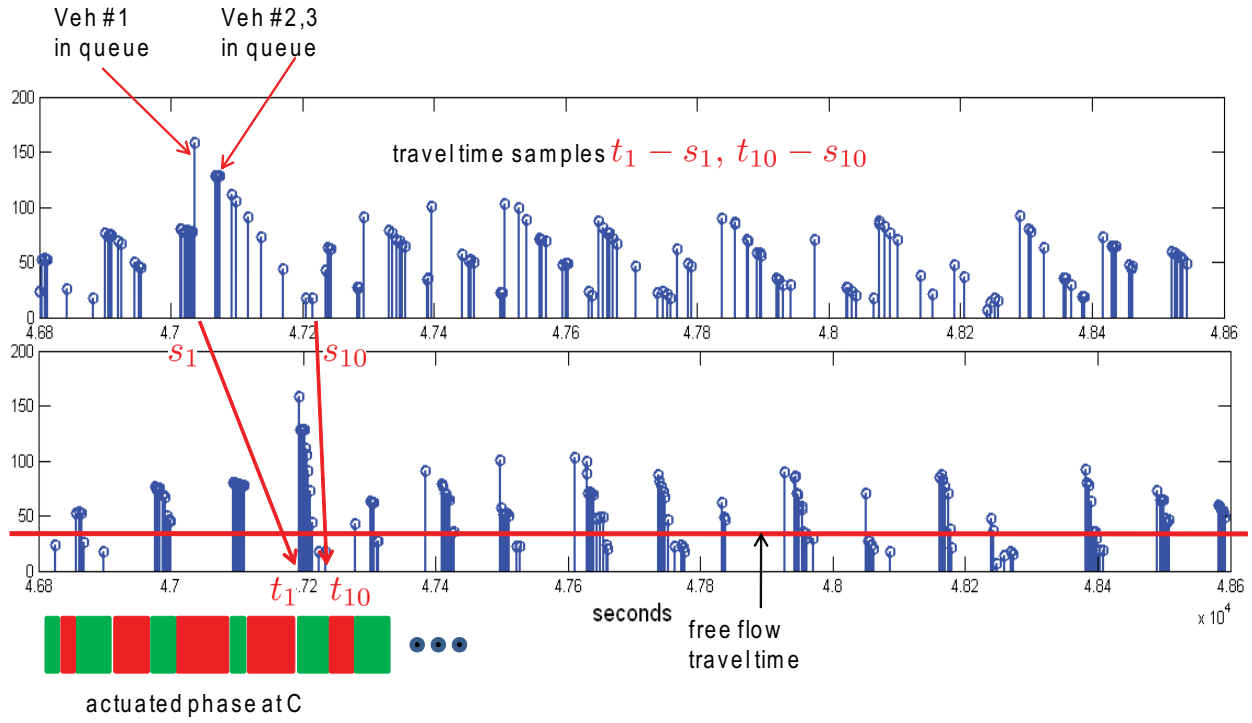

Figure 2: Sample results for link $B \rightarrow C$, May 23, 2008, 1-1:23PM: travel time vs. start time (top), travel time vs. end time (bottom).

Delay

The free flow travel time T_f is the shortest travel time, 20-30s in these plots, as indicated in the lower plot. The difference between a vehicle's travel time and the free flow travel time is the delay it encountered at the intersection from deceleration and acceleration, and waiting in queue or at the stop bar during the red phase. Thus it is straightforward to estimate the total delay, average delay, and delay statistics such as percent of vehicles experiencing delay larger than some amount.

Signal phase

The sensor at C is 12m downstream of the intersection at C, so the end time of a vehicle is virtually the same as the time that it crosses the intersection. Hence the signal phase at C must be green at every vehicle end time. The arrival of a cluster or platoon of vehicles indicates that these vehicles had formed a queue at the stop bar. So the end time of the first vehicle in a platoon must coincide with the start of the green phase (and end of the red phase) at C. From these facts we can precisely infer the start of the green phase at C. On the other hand, the first vehicle in the platoon must be the first to arrive in the red phase. Thus the start of the red phase at C must occur before the start time of the first vehicle in the platoon plus its free flow travel time. This reasoning leads to the construction of the signal phases as illustrated in the bottom of the figure. The signal at C is actuated, so the phase durations are not constant.

Discharge rate

Intersection capacity estimates are based on the maximum rate at which a queue discharges during a green phase. This maximum rate r is easily obtained from the times $\{t_j\}$ when vehicles cross the

downstream sensor by finding the minimum time needed to discharge (say) five vehicles:

$$r = \frac{5}{\min_j (t_{j+5} - t_j)}.$$

By taking different numbers (than 5) more detailed characteristics of queue discharge can also be obtained (Lin and Thomas (2005)).

Queues

A platoon arriving at the downstream sensor indicates that these vehicles were earlier queued at the stop bar. Vehicles in the platoon arrive in order of their position in the queue. The first vehicle in the queue will experience the longest delay (and travel time) and successive vehicles in the queue will experience shorter queueing delays. Thus from the top plot in the figure we can infer that the vehicle with start time s_1 is first in the queue and it waited in the queue for time $t_1 - s_1 - T_f$, the second vehicle in the queue waited for time $t_2 - s_2 - T_f$, and so on. The vehicle with start time s_{10} (or perhaps the vehicle with start time s_8) is the vehicle that joined the queue after it had cleared and hence it faced no delay. Since these eight or ten vehicles arrived in a platoon, we may infer that the queue size reached a maximum value of ten or eight vehicles during this cycle.

Cycle failure

In most cycles the travel time of the first vehicle in a platoon is larger than the travel time of the preceding vehicle, and the latter equals the free flow travel time, T_f. This implies that the queue at the stop bar is cleared in these cycles, indicating no cycle failure. But in Figure 2 the vehicle with start time s_1 left B with the preceding platoon, but did not leave C with that platoon. The vehicle with start time s_1 is delayed by more than one red phase, indicating cycle failure. Thus one can estimate cycle failure (Zheng et al. (2006)).

Vehicles in link

Figure 3 shows how to estimate $N(t)$, the number of vehicles in the link between the upstream and downstream sensors at time t. Let i, s_i denote a vehicle's index and the time it crosses the upstream sensor, and let j, t_j correspond to the downstream sensor. Let J be the index of the downstream vehicle with the largest time $t_J \leq t$ that is matched with some upstream vehicle with index I, say. Let K be the index of the upstream vehicle with the largest time $s_K \leq t$. In the figure, $J = 19$, $I = 6$, $K = 10$. The upstream vehicle with index I left the link before time t. So if I_{max} is the largest index of an upstream vehicle that left the link before t, $I_{max} \geq I$. On the other hand, K is the largest index of an upstream vehicle that is still in the link at time t. Hence if there are no turning movements,

$$N(t) = K - I_{max} \leq K - I.$$

Equality will not hold above only if $I_{max} > I$, which happens only when upstream vehicle I_{max} is not matched. (In the figure, $I_{max} = I = 6, K - I = 4$.) If the matching probability is p, on average $I_{max} - I = p^{-1} - 1$, so if $p > 0.5$, the estimate $K - I$ differs from $N(t)$ by atmost 1 on average.

If there are turning movements, the bound above changes to

$$N(t) \leq K - I - n_{out} + n_{in},$$

in which n_{out} is the number of upstream vehicles with index between I and K (like vehicle with index $i = 8$ in the figure) that went out of the link before crossing the downstream sensor and n_{in} is the number of vehicles with index larger than J (like $j = 20$) that came into the link without crossing the upstream sensor.

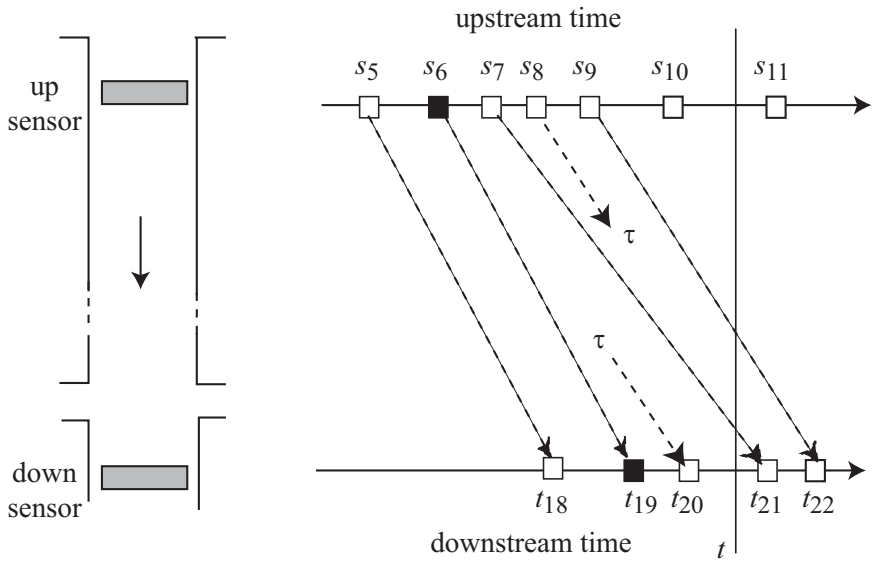

Figure 3: Calculation of number of vehicles N_t in link at time t. Dark squares are vehicles that have been matched before t.

Remark If sensors are placed at the beginning and end of an on- or off-ramp, the scheme above will give an accurate real-time estimate of the number $N(t)$ of vehicles on the ramp at any time t. On a ramp, there are no turning movements, so $n_{out} = n_{in} = 0$. Note, too, that in terms of the notation above, $t_J - s_I$ is the ramp delay experienced by the most recent departing vehicle. The scheme can also be used to estimate the number of vehicles (and hence the average headway) within a freeway segment to obtain the true spatial density. Such estimates may be useful in setting a 'variable speed control' policy.

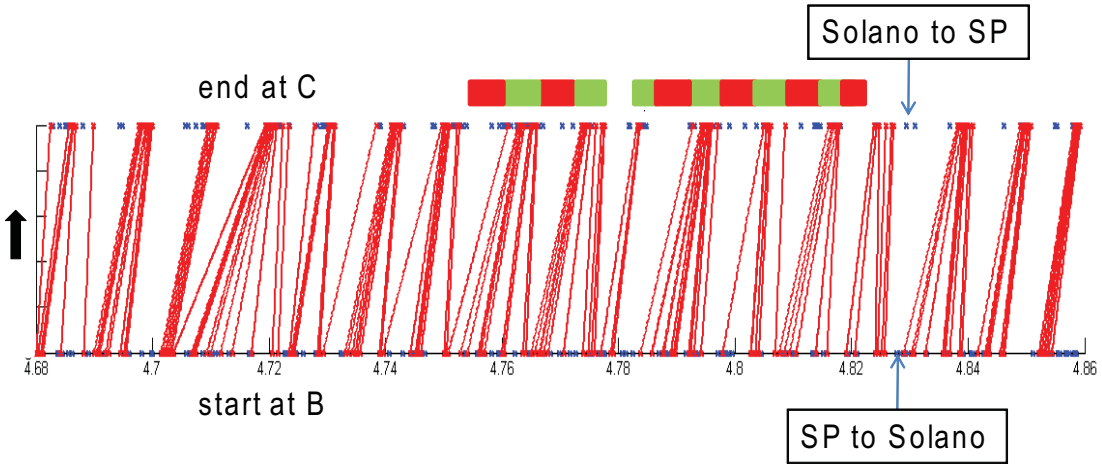

Figure 4: Matched vehicles on link $B \rightarrow C$ are connected; unmatched vehicles are shown as isolated crosses. Vehicles more from bottom (B) to top (C).

Turning

Figure 4 gives more information than Figure 2. Matched vehicles are connected; unmatched vehicles are shown as isolated crosses. The signal phases at C were inferred as explained above. It appears

that the unmatched vehicles that cross C during a red phase are those that turn into San Pablo from Solano (see map in Figure 1). Similarly, some unmatched vehicles at B appear to have turned from San Pablo into Solano without crossing the sensor at Solano.

Travel time distribution

From the matched vehicles one obtains the four travel time distributions of Figure 5.

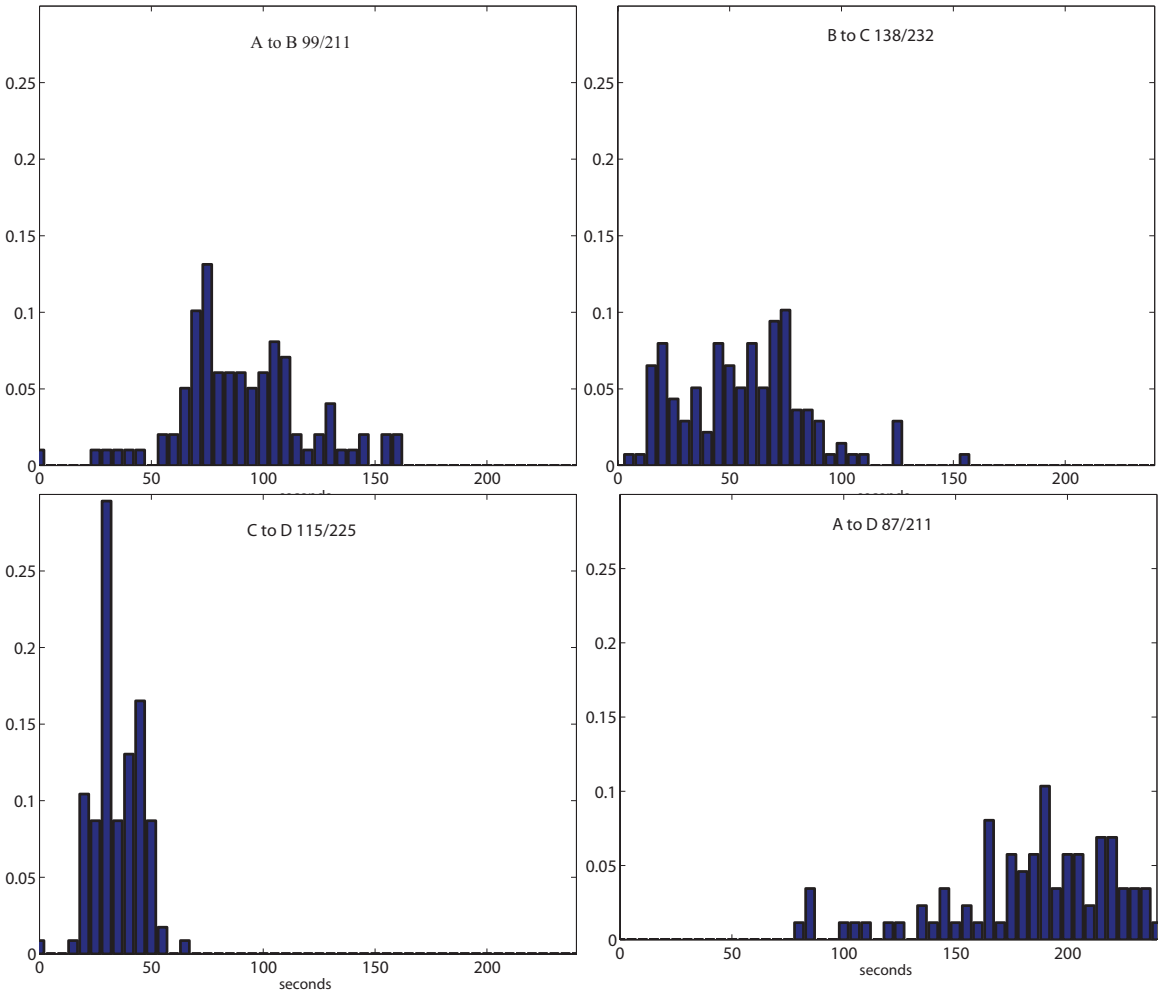

Figure 5: Travel time distributions for May 23, 2008, 1-1:30PM.

The x-axis is travel time in seconds and the y-axis is probability. The travel time distributions of Figure 5 also make clear that the mean travel time is an insufficient indication of travel time *reliability*.

Table 3 gives the means and variances of the four travel time distributions in Figure 5. The sum of the means and variances for links $A \rightarrow B$, $B \rightarrow C$ and $C \rightarrow D$ can be compared with the mean and variance for link $AtoD$. As expected, the mean travel time over $A \rightarrow D$ is the sum of its component link mean travel times. More interesting is that the variance for $A \rightarrow D$ is smaller than the sum of the component link variances, indicating a degree of effective signal coordination.

Matching Rate

Link	Mean	Variance
$A \rightarrow B$	92.4	860
$B \rightarrow C$	58.4	784
$C \rightarrow D$	36.8	109
Sum	187.6	1753
$A \rightarrow D$	187.7	1455

Table 1: Mean and variance of travel times.

The legend 99/211 in the plot for link $A \rightarrow B$ in Figure 5 means that 211 vehicles crossed the sensor at A of which 99 were matched at B. If a fraction τ of vehicles entering A turned before crossing B, the matching rate is $99/(1-\tau)211$. For an estimated $\tau = 0.3$ (there are four intersections between A and B) this gives a matching rate of $99/[0.7 \times 211]$ or 67%. The matching rates for the other links are $138/[0.8 \times 232] = 74\%$ for $B \rightarrow C$; $115/[0.8 \times 225] = 64\%$ for $C \rightarrow D$; and $87/[0.6 \times 211] = 69\%$ for $A \rightarrow D$ (which has six intersections).

Peak vs off-peak

Figure 6 displays the travel time distributions for links $A \rightarrow B$ and $C \rightarrow D$ for an off-peak period, 11-12PM. Comparison with Figure 2 shows that the mean travel time is lower: 77.12sec vs. 92.4sec for link $A \rightarrow B$; 29.6sec vs. 36.8sec for link $C \rightarrow D$. More interestingly, about 25% of peak vehicles face the same travel time as off-peak vehicles.

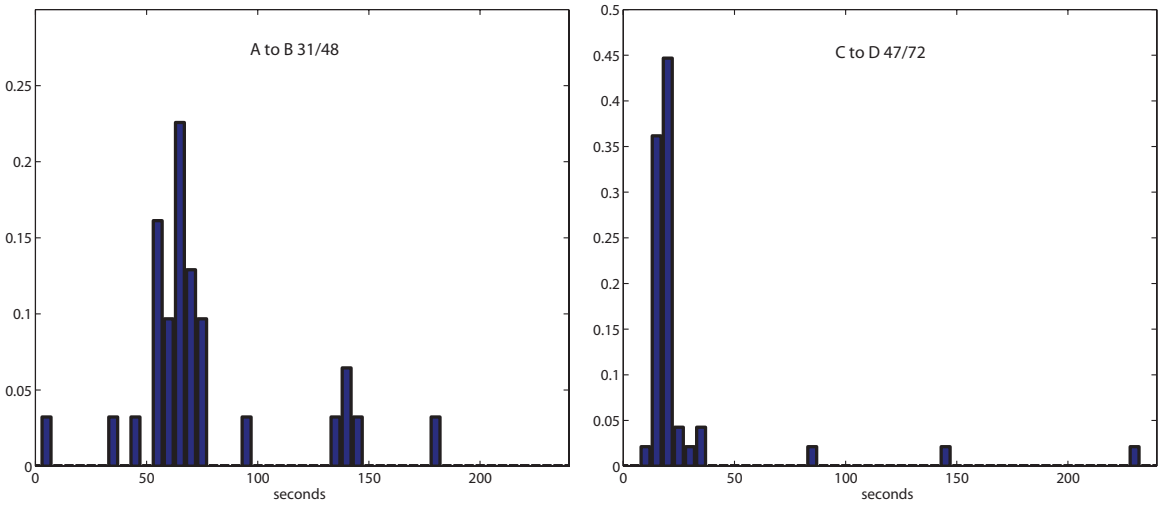

Figure 6: Travel time distributions for May 23, 2008, 11-12PM.

Optimum matching

The results described above are obtained from the optimum matching algorithm presented in Section 6. The algorithm takes the distance matrix as data and produces the optimum match, as illustrated in Figure 7. The plot on the left is a gray scale coding of the distance matrix. There is one pixel for the distance between the signatures of each pair of vehicles; a darker color indicates shorter distance. Clearly visible in the plot is a dark diagonal line of short distances. The graph on the right is the assignment from the optimal matching function. The data are for about 250 vehicles that traversed link $C \rightarrow D$ on May 23, 2008, 1-1:30PM.

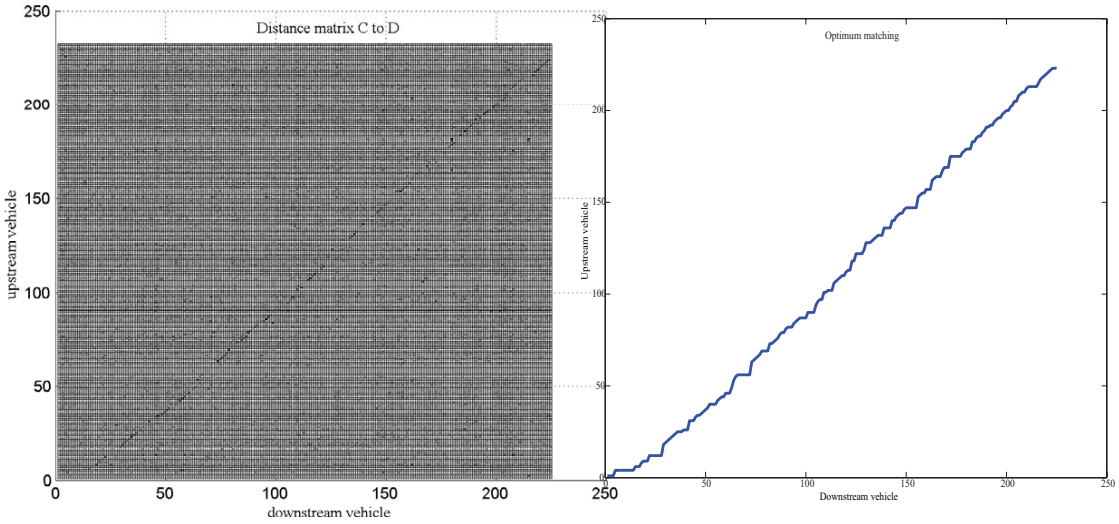

Figure 7: The distance matrix (left) and the optimum constrained match (right) for link $C \rightarrow D$, May 23, 2008, 1-1:30PM

4 Matching problem

The *signal-processing* step assigns to each pair of signatures (X_i, Y_j) the distance $d(i,j) = \delta(X_i, Y_j) \geq 0$ between them. The smaller is $\delta(X_i, Y_j)$ the more likely it is that X_i, Y_j are signatures of the same vehicle. At the end of the signal processing step we thus arrive at the data (D, S, T) consisting of the $N \times M$ distance matrix $D = \{d(i,j),\ 1 \leq i \leq N, 1 \leq j \leq M\}$; the array of N ordered upstream times $S = (s_1, \cdots, s_N)$; and the array of M ordered downstream times $T = (t_1, \cdots, t_M)$.

A *matching function* μ assigns to each distance matrix D the matching $\mu(D)$,

$$\mu(D) : \{1, \cdots, N\} \rightarrow \{1, \cdots, M, \tau\},$$

with this interpretation: $\mu(D)(i) = j$ means that the upstream vehicle i is declared to match (be the same as) downstream vehicle j; $\mu(D)(i) = \tau$ means i is declared not to match any downstream vehicle. If vehicle i is matched with j, its start, end and travel times are s_i, t_j, and $t_j - s_i$ are obtained from the arrays S and T; an unmatched vehicle does not yield a travel time, and may indicate a turning movement.

The *true matching* (ground truth) is denoted by $\bar{\mu}$. The problem is to design a matching $\mu(D)$, based on the observation matrix D, that is close to $\bar{\mu}$, without of course knowing $\bar{\mu}$.

An important, intuitive example is the *minimum distance* matching function, μ_{minD}, which declares an upstream vehicle i to be the same as the downstream vehicle j that is closest to it, provided also that the distance $d(i,j)$ is smaller than a threshold, say d^*; otherwise it assigns τ. Formally,

$$\mu_{minD}(D)(i) = \begin{cases} j & \text{if } d(i,j) \leq d(i,k)\ \forall k; d(i,j) \leq d^* \\ \tau & \text{if } d(i,k) > d^*\ \forall k \end{cases} . \tag{1}$$

Statistical model of signature distance

In order to evaluate the performance of μ_{minD} and to develop better matching functions, we propose a statistical model of the signature distance.

We assume that the distance matrix is characterized by two probability density functions (pdf), f and g: f is the pdf of the distance $\delta(X_v, Y_v)$ between the signatures at the upstream and downstream sensors of the *same* randomly selected vehicle v,

$$f(d) = p(\delta(X_v, Y_v) = d);$$

and g is the pdf of the distance $\delta(X_v, Y_w)$ between two *different* randomly selected vehicles $v \neq w$:

$$g(d) = p(\delta(X_v, Y_w) = d).$$

Then, conditional on the true matching $\bar{\mu}$, the coefficients of the random observation matrix D have the pdf

$$d(i,j) \approx \begin{cases} f & \text{if } \bar{\mu}(i) = j \\ g & \text{if } \bar{\mu}(i) \neq j \text{ or } \bar{\mu}(i) = \tau \end{cases}.$$

We assume that conditional on $\bar{\mu}$ the $d(i,j)$ are independent random variables. Let $D_i = \{d(i,j),\ 1 \leq j \leq M\}$ be the array of distances between X_i and all the Y_j. Then

$$p(D \mid \bar{\mu}) = \prod_i p(D_i \mid \bar{\mu}(i)), \tag{2}$$

$$\begin{aligned} p(D_i \mid \bar{\mu}(i)) &= \begin{cases} f(d(i,j)) \prod_{k \neq j} g(d(i,k)) & \text{if } \bar{\mu}(i) = j \\ \prod_k g(d(i,k)) & \text{if } \bar{\mu}(i) = \tau \end{cases} \\ &= \begin{cases} L(d(i,j))\gamma(D_i) & \text{if } \bar{\mu}(i) = j \\ \gamma(D_i) & \text{if } \bar{\mu}(i) = \tau \end{cases}, \end{aligned} \tag{3}$$

in which

$$L(d(i,j)) = \frac{f(d(i,j))}{g(d(i,j))},\ \ \gamma(D_i) = \prod_{k=1}^{M} g(d(i,k)). \tag{4}$$

Relations (2)-(4) constitute the signature distance statistical model.

Figure 8 displays the empirical pdfs and the Gaussian approximations of f and g for the three links. The annotation above the left plot for link $A \rightarrow B$ means that μ_f and σ_f are the mean and standard deviation for f; μ_g and σ_g are the mean and standard deviation for g; $n_f = 91$ and $n_g = 24,622$ are the number of samples used to estimate the statistics for f and g, respectively. That is, there were 91 matched vehicle pairs and 24,622 unmatched pairs. (There always are many more unmatched pairs.) Section 8 describes how the distributions in Figure 8 are estimated.

5 Optimal unconstrained matching

We evaluate the performance of μ_{minD} and the unconstrained MAP (maximum a posteriori) matching function, μ_{uMAP}, defined later. These matching functions are *unconstrained* since no restriction is placed on the assignment:

$$\mu(D) : \{1, \cdots, N\} \rightarrow \{1, \cdots, M, \tau\}.$$

In particular, the matching function permits a vehicle to overtake other vehicles in front of it, even though, as in the case of a single arterial link, this is unlikely. Matching functions considered in the

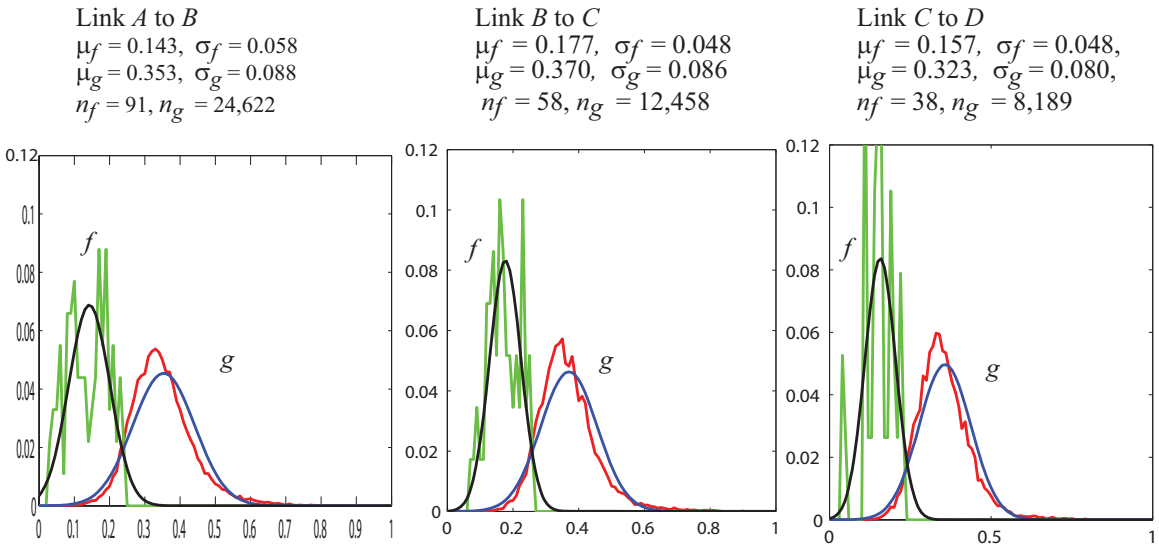

Figure 8: The empirical pdfs f and g and their Gaussian approximations for links $A \to B$, $B \to C$ and $C \to D$.

literature are unconstrained. In the next section we consider constrained matching functions, which do not permit overtaking. Such a constraint greatly improves performance.

Define the cumulative and complementary distribution functions

$$G(d) = \int_0^d g(x)dx; \quad \tilde{G}(d) = 1 - G(d).$$

Similarly define $F(d)$ and $\tilde{F}(d)$.

5.1 Minimum distance matching

Theorem 5.1 gives the performance of the minimum distance matching function, μ_{minD}. It is proved in the Appendix.

Theorem 5.1 *The probability of a correct match, $\mu_{minD}(i) = \bar{\mu}(i)$, is*

$$p(\mu_{minD}(D)(i) = j \mid \bar{\mu}(i) = j) = \int_0^{d^*} f(x)[\tilde{G}(x)]^{M-1}dx. \tag{5}$$

The probability of an incorrect match, $\mu_{minD}(i) \neq \bar{\mu}(i)$, is

$$p(\mu_{minD}(D)(i) \neq j \mid \bar{\mu}(i) = j) = (M-1)\int_0^{d^*} [\tilde{G}(x)]^{M-2} g(x)\tilde{F}(x)dx. \tag{6}$$

The probability that a vehicle is unmatched, $\mu_{minD}(i) = \tau$, is

$$p(\mu_{minD}(D)(i) = \tau \mid \bar{\mu}(i) = j) = \tilde{F}(d^*)[\tilde{G}(d^*)]^{M-1}. \tag{7}$$

The three probabilities (5)-(7) add up to 1.

Theorem 5.1 allows us to predict how the performance depends on the threshold d^* and the number M of potential vehicle matches. From (5)-(6), the probabilities of both correct and incorrect match increase as d^* increases. Thus, as in hypothesis testing generally, the proper choice of the threshold value must compromise between correct and incorrect re-identification.

Second, from (5)-(6), the probability of a correct match decreases and the probability of an incorrect match increases as M increases. This is intuitive: the larger is the number M of potential matches, the worse is the performance of the minimum distance matching function. So one way to improve the matching algorithm is to reduce M. A common way of reducing M is to place an upper bound T on the link travel time and limit the matching of an upstream vehicle i to those downstream vehicles j for which $t_j - s_i \leq T$, e.g., M.Ndoye et al. (2008).

Third, the probability of a vehicle being unmatched decreases as d^* or M increases. This, too, is intuitive: a larger d^* implies a less stringent condition on matching, while a larger M increases the chance of finding a potential match.

Formulas (5)-(7) help determine the range of values of d^* and M for which the re-identification scheme gives satisfactory performance. Figure 9 plots the probabilities of correct and incorrect matches using the Gaussian approximations for the distributions f, g in Figure 8 in (5), (6) for link $A \rightarrow B$. For a per lane flow of 500 vph, $M = 50$ corresponds to a time interval of 6 minutes, which is the travel time window over a three-mile long link at an average speed of 30 mph. For $d^* = 0.15$, the minimum distance matching function is predicted to give 45% correctly matched, 25% incorrectly matched, and 30% unmatched vehicles.

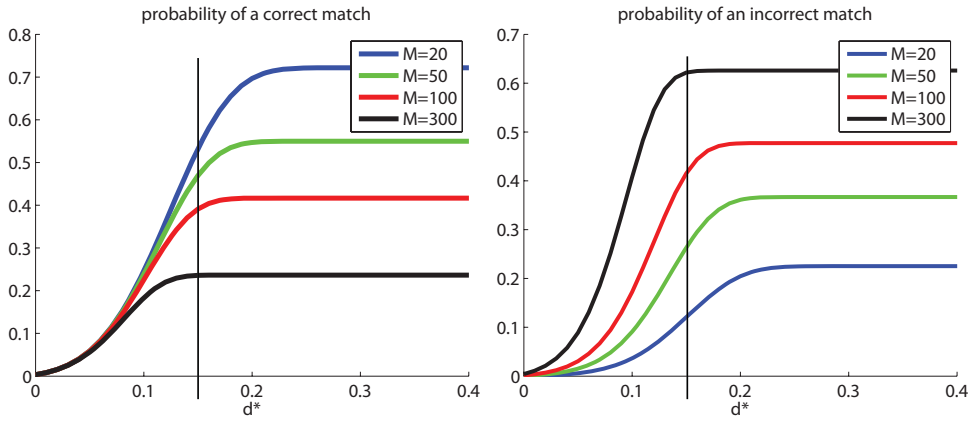

Figure 9: Probabilities of correct and incorrect matches of μ_{minD} for different values of d^*, M.

5.2 Unconstrained MAP matching μ_{uMAP}

We shall evaluate the performance of any matching function μ by its (normalized) *reward* $\rho(\mu)$:

$$\rho(\mu) = \frac{1}{N} E \sum_{i=1}^{N} \mathbf{1}(\mu(D)(i) = \bar{\mu}(i)), \tag{8}$$

in which $\mathbf{1}(\cdot)$ is the indicator function: $\mathbf{1}(\mu(D)(i) = \bar{\mu}(i)) = 1$ if $\mu(D)(i) = \bar{\mu}(i)$, and $= 0$ otherwise. Thus $\rho(\mu)$ is the correct matching rate, the fraction of correctly matched vehicles on average.

To evaluate the expectation operator E in (8) we need the joint probability distribution on $(D,\bar{\mu})$. Since $p(D \mid \bar{\mu})$ is already specified by (2)-(4) we only need the (prior) distribution of $\bar{\mu}$. We assume that the number of upstream vehicles $N = (1+\beta)M$ of which M vehicles cross the downstream sensor and βM vehicles turn before reaching the downstream sensor. Thus β is the turning probability. Subject to this assumption, we impose a uniform distribution on $\bar{\mu}$:

$$p(\bar{\mu}(i) = j) = \alpha,\ j = 1,\cdots,M;\ \ p(\bar{\mu}(i) = \tau) = \beta, \tag{9}$$

with $M\alpha + \beta = 1$.

Let μ^* denote the *optimal* or reward-maximizing matching function. Theorem 5.2 is proved in the Appendix.

Theorem 5.2 *μ^* is given by*

$$\mu^*(D)(i) = \begin{cases} j & \text{if } L(d(i,j)) \geq L(d(i,k))\ \forall k;\ L(d(i,j)) \geq \beta/\alpha \\ \tau & \text{if } L(d(i,k)) < \beta/\alpha\ \forall k \end{cases}. \tag{10}$$

To implement (10) we need β or α, since the $d(i,j)$ and M are known from the data. In the present context, β is the turning probability, which may be determined from field observations or experience. But another consideration may govern the choice of β. From (10) one sees that the larger is β, the more stringent is the requirement of a match, and lower is the probability of an incorrect match. So, depending on the application, one should choose a larger value for β, if the 'cost' of an incorrect match is high.

Observe that $\mu^*(D)$ maximizes the posterior probability

$$p(\bar{\mu} \mid D) = \frac{p(D \mid \bar{\mu})(p(\bar{\mu})}{p(D)}, \tag{11}$$

with prior probability $p(\bar{\mu})$ given by (9). So $\mu^* = \mu_{uMAP}$ is also the *(unconstrained) maximum a posteriori (MAP)* matching function.

The minimum distance matching μ_{minD} and unconstrained MAP matching μ_{uMAP} have a similar structure. One calculates a statistic for each pair (i,j) of downstream and upstream vehicles—$d(i,j)$ in the μ_{minD} case and the likelihood ratio $L(d(i,j))$ in the μ_{uMAP} case—and matches i to the best j in terms of this statistic, provided that it meets a threshold. However, μ_{minD} does not take into account the uncertainty in the distance measurements, whereas μ_{uMAP} does.

Intuitively, correct matching of a downstream vehicle i to one of the upstream vehicles $1,\cdots,M$ should be more difficult as M increases. The next result shows this is indeed the case. Corollary 5.1 can be compared with (5).

Define

$$f_L(l) = p(L(d(i,j)) = l \mid \bar{\mu}(i) = j) \text{ and } G_L(l) = p(L(d(i,k)) \leq l \mid \bar{\mu}(i) \neq k), \tag{12}$$

the pdf of $L(d(i,j))$ and the cumulative distribution function (cdf) of $L(d(i,k))$, conditional on $\bar{\mu}(i) = j \neq k$. That is, $f_L(l)$ is the pdf of $L(d)$ when d has pdf f, and $G_L(l)$ is the cdf of $L(d)$ when d has pdf g.

Corollary 5.1 *The maximum reward is given by the explicit formula:*

$$\rho(\mu_{uMAP}) = M\alpha \int_{\beta/\alpha}^{\infty} f_L(l)[G_L(l)]^{M-1} dl + \beta[G_L(l)]^M. \tag{13}$$

Moreover, $\rho(\mu_{uMAP})$ *decreases as* M *increases, keeping* $M\alpha$ *and* β *constant (*$M\alpha+\beta=1$*).*

Proof See Appendix. □

Remark on the Gaussian case In the Gaussian case,

$$f(d) = \frac{1}{\sqrt{2\pi\sigma_f^2}} \exp - \frac{(d-\mu_f)^2}{2\sigma_f^2}, \quad g(d) = \frac{1}{\sqrt{2\pi\sigma_g^2}} \exp - \frac{(d-\mu_g)^2}{2\sigma_g^2}.$$

Here μ_f and σ_f denote the mean and standard deviation of the Gaussian pdf f while μ_g and σ_g denote analogous quantities for g. To characterize μ^* it is more convenient to maximize the 'log-likelihood' $l(d)$ instead of the 'likelihood' $L(d)$:

$$l(d) = \ln L(d) = \ln \frac{\sigma_g}{\sigma_f} - \frac{(d-\mu_f)^2}{2\sigma_f^2} + \frac{(d-\mu_g)^2}{2\sigma_g^2}.$$

It is easy to check by differentiating this quadratic expression that $l(d)$ is *decreasing* in d for $0 \leq d \leq \mu_g$ for the estimated parameters of Figure 8. Thus in this range maximizing $l(d)$ is equivalent to minimizing d. Since $[0,\mu_g]$ is likely to include the range $L(d(i,j)) \geq \beta/\alpha$ (in (10)), this means that μ^* coincides with μ_{minD} (with an appropriate threshold).

6 Optimal constrained matching

Minimum distance μ_{minD}, and unconstrained MAP μ_{uMAP}, and the matchings in (Ritchie et al. (2002); M.Ndoye et al. (2008)) are examples of unconstrained matching. Unconstrained matching may violate two constraints. First, a matching may allow duplicates: two different upstream vehicles $i_1 \neq i_2$ may be matched to the same downstream vehicle j. Second, a matching may permit overtaking: an upstream vehicle i_2 which follows i_1, $i_2 > i_1$, may be matched to downstream vehicles j_1, j_2 in the reverse order, $j_1 > j_2$. A *constrained* matching should not allow duplicates and it should not permit overtaking.

Suppose the data comprise a sequence of upstream vehicle signatures X_i, $i = 1,\cdots,N$ and a sequence of downstream vehicle signatures Y_j, $j = 1,\cdots,M$. A *constrained matching* is a pair of matched sequences like $(Up, Down)$:

$$\begin{array}{rcccccccc} Up & = & X_1 & \tau & X_2 & X_3 & X_4 & X_5 & X_6 \\ & & \downarrow & \downarrow & \downarrow & \downarrow & \downarrow & \downarrow & \downarrow \\ Down & = & \tau & Y_1 & Y_2 & \tau & Y_3 & Y_4 & Y_5 \end{array} \tag{14}$$

The interpretation of (14) is that the upstream vehicles with signatures X_2, X_4, X_5 and X_6 are respectively matched with the downstream vehicles with signatures Y_2, Y_3, Y_4 and Y_5; the upstream vehicles with signatures X_1 and X_3 are matched with τ, meaning that these vehicles turned before reaching the downstream sensor (or are unmatched for some other reason) ; and the downstream vehicle with signature Y_1 is matched with τ, meaning this vehicle turned into the link without crossing

the upstream sensor but it did cross the downstream sensor. Note that in (14) one can distinguish the two types of turns.

In general, a constrained matching $(Up, Down)$ is a pair of equal length sequences, comprising the original signature sequences, $X_1, \cdots, X_N$ and $Y_1, \cdots, Y_M$, together with arbitrarily many insertions of τ within these sequences, with the restriction that an occurrence of τ in Up can only be matched with a Y_j in *Down* and a τ in *Down* can only be matched with a X_i in Up. Observe that in a constrained matching the matches of the form $Y_j \rightarrow \tau$ are determined once all matches of the form $X_i \rightarrow Y_j$ and $X_i \rightarrow \tau$ are specified.

We want to find μ_{cMAP}, the *maximum a posteriori* (MAP) matching function for the constrained case. μ_{cMAP} maximizes the posterior probability

$$p(\bar{\mu} \mid D) = \frac{p(D \mid \bar{\mu}) p_c(\bar{\mu})}{p(D)}, \tag{15}$$

in which $p_c(\bar{\mu})$ denotes the prior probability that $\bar{\mu}$ is the true constrained matching. In (15), $p(D \mid \bar{\mu})$ is given by the signature distance model (2)-(4), so we only need to figure out the prior $p_c(\bar{\mu})$, which is just the unconstrained prior $p(\bar{\mu})$ given in (9), conditioned by the requirement that $\bar{\mu}$ is a constrained matching. That is,

$$p_c(\bar{\mu}) = \begin{cases} p(\bar{\mu}) / \sum\limits_{\bar{\mu} \in M_c} p(\bar{\mu}) & \bar{\mu} \in M_c \\ 0 & \bar{\mu} \notin M_c \end{cases}, \tag{16}$$

in which M_c denotes the set of constrained matchings. Thus μ_{cMAP} is given by

$$\mu_{cMAP} = \arg\max_{\bar{\mu} \in M_c} \frac{p(D \mid \bar{\mu}) p(\bar{\mu})}{p(D)} = \arg\max_{\bar{\mu} \in M_c} p(D \mid \bar{\mu}) p(\bar{\mu}).$$

The last equality follows from the fact that $p(D) = \sum_{\bar{\mu} \in M_c} p(D \mid \bar{\mu}) p(\bar{\mu})$ does not depend on $\bar{\mu}$.

Recall that the unconstrained prior is the uniform distribution on $\bar{\mu}$ with turning probability β:

$$p(\bar{\mu}) = \prod_i p(\bar{\mu}(i)); \quad p(\bar{\mu}(i) = j) = \alpha, \ j = 1, \cdots, M; \ \ p(\bar{\mu}(i) = \tau) = \beta, \tag{17}$$

with $M\alpha + \beta = 1$. Using (2)-(4) and (17) gives

$$p(D \mid \bar{\mu}) p(\bar{\mu}) = \prod_i p(D_i \mid \bar{\mu}(i)) p(\bar{\mu}(i)), \tag{18}$$

$$p(D_i \mid \bar{\mu}(i)) p(\bar{\mu}(i)) = \begin{cases} L(d(i,j)) \gamma(D_i) \alpha & \bar{\mu}(i) = j \\ \gamma(D_i) \beta & \bar{\mu}(i) = \tau \end{cases}. \tag{19}$$

To find μ_{cMAP} we must maximize the likelihood (18) over the set M_c. It will be more convenient to minimize the negative 'log-likelihood',

$$\begin{aligned} -\ln p(D \mid \bar{\mu}) p(\bar{\mu}) &= -\sum_i \ln p(D_i \mid \bar{\mu}(i)) - \ln p(\bar{\mu}(i)) \\ &= \sum_i \sum_j \lambda(i,j) \mathbf{1}(\bar{\mu}(i) = j) + \sum_i \lambda(i,\tau) \mathbf{1}(\bar{\mu}(i) = \tau), \end{aligned} \tag{20}$$

in which $\mathbf{1}(\cdot)$ denotes the indicator function and

$$\lambda(i,j) = -\ln L(d(i,j)) - \ln \gamma(D_i) - \ln \alpha, \tag{21}$$

$$\lambda(i,\tau) = -\ln \gamma(D_i) - \ln \beta. \tag{22}$$

Thus to find μ_{cMAP} we must minimize the linear form (20) over the "combinatorial" constraint $\bar{\mu} \in M_c$. The difficulty is to find a convenient representation of M_c.

We now describe a graph $\mathcal{G}(N,M)$ whose paths are in one-one correspondence with the set M_c of all constrained matchings. $\mathcal{G}(N,M)$ comprises $(N+1)\times(M+1)$ nodes arranged in the form of a grid like the one shown in Figure 10, which is the graph for example (14) with $N=6$, $M=5$. $\mathcal{G}(N,M)$ is called the *edit graph* in the literature on sequence comparison algorithms in computer science and molecular biology Myers (1986).

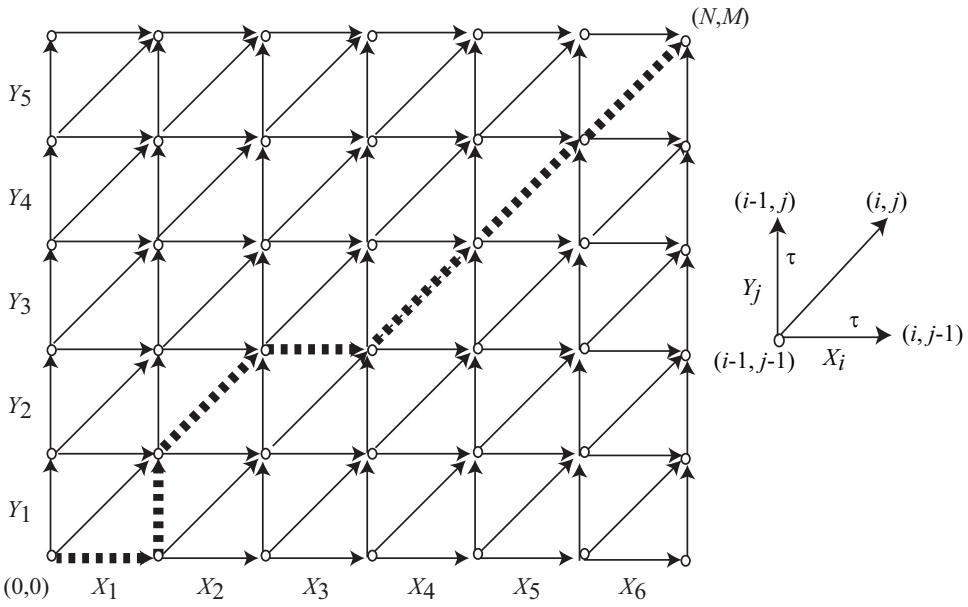

Figure 10: The edit graph for example (14). A diagonal edge corresponds to a signature match; a horizontal or vertical edge corresponds to a turn (match with τ).

$\mathcal{G}(N,M)$ is constructed as follows. Its nodes are labeled (i,j), $0 \le i \le N$, $0 \le j \le M$. A node $(i-1,j-1)$ has three directed edges connected to nodes $(i-1,j)$, $(i,j-1)$ and (i,j) (unless $i>N$ or $j>M$). The 'diagonal' edge $(i-1,j-1)\to(i,j)$ indicates the match $X_i \to Y_j$; the 'horizontal' edge $(i-1,j-1)\to(i,j-1)$ indicates the match $X_i \to \tau$ (vehicle i did not cross the downstream sensor); the 'vertical' edge $(i-1,j-1)\to(i-1,j)$ indicates the match $Y_j \to \tau$ (vehicle j did not cross the upstream sensor).

It is an obvious but very important fact that each path in $\mathcal{G}(N,M)$ from node $(0,0)$ to (N,M) corresponds to a constrained matching (no duplicates, no overtaking) and vice versa. The constrained matching (14) corresponds to the path in Figure 10 indicated by the thick dashed lines.

Having identified constrained matchings with paths in the edit graph, we identify (20) with the sum of the weights of the edges along the path, assigning edge weights according to

$$\begin{array}{lcl} w((i-1,j-1)\to(i,j)) & = & \lambda(i,j) \\ w((i-1,j-1)\to(i,j-1)) & = & \lambda(i,\tau) \\ w((i-1,j-1)\to(i-1,j)) & = & 0 \end{array} . \qquad (23)$$

Evidently, the value (20) for any constrained matching $\bar{\mu}$ is equal to the weight of the corresponding path (defined as the sum of the edge weights) in the edit graph. Thus finding μ_{cMAP} is equivalent to finding the minimum weight path.

The minimum weight path can be found via the algorithm described in Table 2, which recursively computes $W(i,j)$, the weight of the shortest path from node $(0,0)$ to node (i,j); in particular

$$W(N,M) = \min_{\mu} W(\mu).$$

The array $E(i,j)$ can be used to backtrack and recover the shortest path or μ_{minW}. The algorithm requires a single pass through all the MN nodes taken in topological order, so its complexity is $O(MN)$.

$W(0,0) \leftarrow 0$.
For $j \leftarrow 1$ to M do $W(0,j) \leftarrow W(0,j-1) + w((0,j-1) \rightarrow (0,j))$.
For $i \leftarrow 1$ to N do $W(i,0) \leftarrow W(i-1,0) + w((i-1,0) \rightarrow (i,0))$.
For $i \leftarrow 1$ to N do
  For $j \leftarrow 1$ to M do {
    $W(i,j) \leftarrow \min\{W(i-1,j) + w((i-1,j) \rightarrow (i,j)), W(i,j-1) + w((i,j-1) \rightarrow (i,j)), W(i-1,j-1) +$
      $+ w((i-1,j-1) \rightarrow (i,j))\}$;
    $E(i,j) \leftarrow \arg\min\{W(i-1,j) + w((i-1,j) \rightarrow (i,j)), W(i,j-1) + w((i,j-1),(i,j)), W(i-1,j-1) +$
      $+ w((i-1,j-1) \rightarrow (i,j))\}$;
  }

Table 2: Shortest path algorithm for μ_{cMAP}.

As evident in Figure 10, from the shortest path one can read off the vehicles that are re-identified $X_i \rightarrow Y_j$. Since we know the times s_i, t_j when these vehicles crossed the two sensors, we know their travel time $T_i = t_j - s_i$. One can also read off which vehicles were declared to have crossed the upstream sensor but not the downstream sensor (these are the horizontal edges in the path) and which vehicles were declared to have crossed the upstream sensor but not the downstream sensor (these are the vertical edges).

The results in Section 4 are based on this shortest path algorithm. Figure 7 illustrates the result for link $C \rightarrow D$. Approximately 250 vehicles crossed this link in 30 min. The horizontal edges in the shortest path are presumably vehicles that crossed the sensor at C but not at D; the vertical edges are vehicles that crossed D but not C.

Implementation of the algorithm requires the edge weights, which in turn require knowing the Gaussian distributions f, g of Figure 8 and the turning probability β. We consider these in turn.

Section 8 explains how the distributions are obtained through an iterative procedure. The iteration starts from an initial matching, which is obtained by a modified shortest path algorithm applied to edge weights obtained directly from the distance matrix $D = \{d(i,j)\}$. That algorithm called *saturation method* works as follows:

Step 1. The distance matrix is scanned and values $d(i,j) > .75$ are set to .75. (The signal processing step assigns distance values between 0 and 1.) Call the resulting matrix DM.
Step 2. The shortest path is calculated with this iteration rule:

$$\Delta(i,j) = \min\{\Delta(i-1,j-1) + 2 * DM(i,j), \Delta(i-1,j) + DM(i,j), \Delta(i,j-1) + DM(i,j)\}$$

in which $\Delta(i,j)$ is the cost of the shortest path from node $(0,0)$ to node (i,j).

As in the unconstrained case, the choice of β is determined by experience or observation. More importantly, the choice of β controls the probability that μ_{cMAP} makes incorrect matches: the larger

is β, the lower the probability of an incorrect match; but the larger is β, the lower is also the probability of a correct match. It is possible to obtain a formula for the expected number of correct matches with μ_{cMAP} like formula (13) for the performance of μ_{uMAP}. But the formula for μ_{cMAP} involves a multi-dimensional integral that cannot be evaluated in closed form (unlike (13)) and it is not presented here.

Of course the performance of μ_{cMAP} is much better than that of μ_{uMAP} because of the restriction to constrained matching.

7 Real-time matching

As formulated in Section 6 the edit graph grows with the observation time interval, and so with each new upstream or downstream vehicle, one needs to calculate the distance of its signature from all previous signatures. The effort to compute these distances for each new vehicle grows linearly with the observation interval, which is unsuitable for real-time implementation.

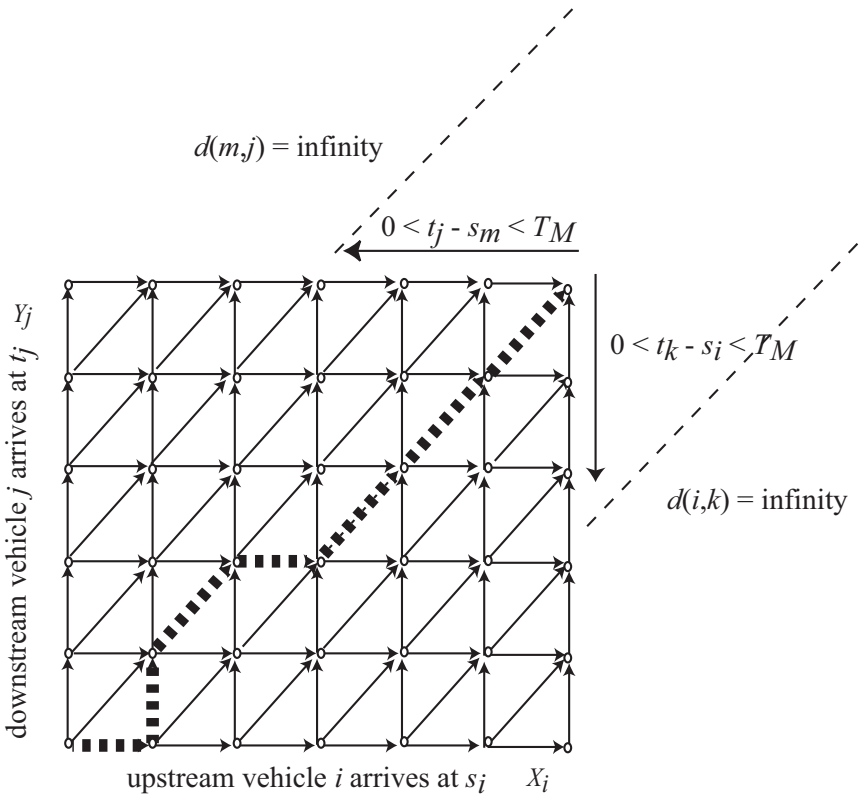

Figure 11: The distances to be calculated in real-time implementation.

For real-time implementation, we want to restrict the growth of the edit graph. One way of doing this is as follows. For each new upstream vehicle i (with signature X_i) that arrives at time s_i compute the distance $\delta(X_i, Y_k)$:

$$d(i,k) = \begin{cases} \delta(X_i, Y_k) & \text{if } 0 \le t_k - s_i \le T_M \\ \infty & \text{else} \end{cases} . \tag{24}$$

For each new downstream vehicle j (with signature Y_j) that arrives at time t_j compute the distance $\delta(X_m, Y_j)$:

$$d(m,j) = \begin{cases} \delta(X_m, Y_j) & \text{if } 0 \le t_j - s_m \le T_M \\ \infty & \text{else} \end{cases} . \tag{25}$$

If T_M is an upper bound on the travel time, (24)-(25) must hold. The distances $\delta(X_i, Y_j)$ to be calculated are thus bounded by the dashed lines in Figure 11. So the computational burden for each new upstream or downstream vehicle is essentially constant.

A simple choice for T_M would be to assume a minimum speed and take T_M to be the corresponding travel time, which requires knowledge of the link length, cycle times, etc. A better idea is this. Let μ be the shortest path matching until the current time. Pick an integer M. Let $i_1, \cdots, i_M$ be the most recent M matched downstream vehicles, and set

$$T_M = 2 \times \max\{t_{\mu(i_m)} - s_{i_m},\ 1 \le m \le M\}.$$

Thus T_M is twice the maximum travel time experienced by these M vehicles. This choice will automatically adapt to changes in travel time.

8 Estimating the statistical model

The statistical model (2)-(4) parameterizes the probability density of the observation matrix D as $p(D \mid \bar{\mu}, \mu_f, \sigma_f, \mu_g, \sigma_g)$, with parameter $\bar{\mu}$ for the true matching, and $(\mu_f, \cdots, \sigma_g)$ for the four parameters of the Gaussian distributions of f, g.

Ideally the optimum matching and the parameters of f, g should be jointly determined as the maximum likelihood estimate

$$(\hat{\bar{\mu}}, \hat{\mu_f}, \hat{\sigma_f}, \hat{\mu_g}, \hat{\sigma_g}) = \arg \max_{\bar{\mu}, \mu_f, \sigma_f, \mu_g, \sigma_g} p(D \mid \bar{\mu}, \mu_f, \sigma_f, \mu_g, \sigma_g). \tag{26}$$

While (26) is a well-defined optimization problem, it is computationally very expensive to solve because of the combinatorial nature of the variable $\bar{\mu}$. Instead we perform a coordinate-wise optimization:

1. Begin with an initial estimate $\bar{\mu}^0$ (which we do using the saturation method).
2. At step i we have the estimate $\bar{\mu}^i$. Find

   $$(\mu_f^i, \sigma_f^i, \mu_g^i, \sigma_g^i) = \arg \max_{\mu_f, \sigma_f, \mu_g, \sigma_g} p(D \mid \bar{\mu}^i, \mu_f, \sigma_f, \mu_g, \sigma_g)).$$

   This step is easy because the given match $\bar{\mu}^i$ divides elements of the observation matrix D into distances of pairs of matched and unmatched vehicles, so (μ_f^i, σ_f^i) are the empirical moments of the matched pairs and (μ_g^i, σ_g^i) are corresponding values for the unmatched pairs.
3. Use the optimal matching algorithm to find

   $$\bar{\mu}^{i+1} = \arg \max_{\bar{\mu}} p(D \mid \bar{\mu}, \mu_f^i, \sigma_f^i, \mu_g^i, \sigma_g^i),$$

   and return to 2. with $i+1$.

Since the likelihood $p(D \mid \bar{\mu}^i, \mu_f^i, \sigma_f^i, \mu_g^i, \sigma_g^i)$ increases with i, the iteration must converge to a local maximum of the likelihood. For the test results the iterations converged in three to four steps.

9 Conclusion

A procedure is described for the real-time estimation of the distribution of travel time across an arterial segment with several intersections. The scheme relies on re-identification of vehicle signatures from wireless magnetic sensors. The procedure is tested for a 0.9-mile segment with six intersections. The procedure requires no measurement of vehicle speed or signal phase information. Matched vehicle results yield signal phase, queues at stop bars, number of vehicles in the link, and other arterial performance measures, in addition to travel time and delay distributions.

The scheme can be used in an off- or on-ramp for the real-time estimation of the number of vehicles within the ramp and the ramp delay experienced by every vehicle. The scheme can also be used two determine the number of vehicles between two locations on a freeway in order to estimate true spatial density. These properties make deployment of the scheme immediately practicable.

The procedure is based on a statistical model of signature distance, whose parameters are themselves estimated from the data, so that *no* ground truth measurements are needed. The model can be used to predict the rates of correct, incorrect, and missed matches.

References

B. Coifman. *Vehicle reidentification and travel time measurement using loop detector speed traps*. PhD thesis, University of California, Berkeley, Berkeley, CA 94720, 1999.

A. Haoui, R. Kavaler, and P. Varaiya. Wireless magnetic sensors for traffic surveillance. *Transportation Research Part C*, 16:294–306, 2008.

F.B. Lin and D.R. Thomas. Headway compression during queue discharge at signalized intersections. *Transportation Research Record*, (1920):81–85, 2005.

H. X. Liu and W. Ma. A virtual vehicle probe model for time-dependent arterial travel time estimation. Submitted for Publication, 2008.

M.Ndoye, V. Totten, B. Carter, D.M. Bullock, and J.V. Krogmeier. Vehicle detector signature processing and vehicle reidentification for travel time estimation. In *Proceedings of 88th Transportation Research Board Annual Meeting*, Washington, D.C., January 2008.

E.W. Myers. An $O(ND)$ difference algorithm and its variations. *Algorithmica*, 1:251–266, 1986.

C. Oh and S.G. Ritchie. Real-time inductive-signature-based level of service for signalized intersections. *Transportation Research Record*, (1802):97–104, 2002.

S.G. Ritchie, S. Park, C. Oh, and C. Sun. Field investigation of advanced vehicle reidentification techniques and detector technologies phase 1. Technical Report PATH Research Report UCB-ITS-PRR-2002-15, Institute of Transportation Studies, University of California, Berkeley, California, 2002.

A. Skabardonis and N. Geroliminis. Real-time estimation of travel times along signalized arterials. In H.S. Mahmassani, editor, *Proceedings of the 16th International Symposium on Transportation and Traffic Theory*, pages 387–406. Elsevier, 2005.

C. Sun, S.G. Ritchie, K. Tsai, and R. Jayakrishnan. Use of vehicle signature analysis and lexicographic optimization for vehicle reidentication on freeways. *Transportation Research, C*, 7: 167–185, 1999.

C.C. Sun, G.S. Arr, R.P. Ramachandram, and S.G. Ritchie. Vehicle reidentification using multidetector fusion. *IEEE Transactions on Intelligent Transportation Systems*, 5(3):155–164, September 2004.

H.M. Zhang. Link-journey-speed model for arterial traffic. *Transportation Research Record*, (1676): 109–115, 1999.

J. Zheng, Y. Wang, and N.L. Nihan. Detecting cycle failures at signalized intersections using video image processing. *Computer-Aided Civil and Infrastructure Engineering*, 21:425–435, 2006.

Appendix

Signal processing algorithm

As mentioned in the text, a sensor comprises an array of seven nodes, each of which has a three-axis magnetometer that measures the x, y, z directions of the earth's magnetic field at a sampling rate of 128Hz as a vehicle goes over the node. Figure 12 shows the raw z-axis measured signal from one node. The x- and y-axis signals are similar.

The microprocessor in the node automatically extracts the sequence of peak values (local maxima and minima) from each of the x, y, z signals. (The times when the peaks occur are not recorded.) In the example, there are six peak values (including the initial and terminal values of the signal), denoted by squares. The node transmits the array of these peak values to the access points (AP). There will be three such arrays, corresponding to the three axes. Together, the three arrays constitute a *slice* of the two-dimensional magnetic 'footprint' of the vehicle.

A slice measured by a node is determined by the distribution of the ferrous material in the vehicle within 12" from the node. For each vehicle, the AP receives one slice from each of the seven nodes. The seven slices constitute the vehicle's signature at the sensor.

The signal processing algorithm takes two signatures, say $X = (X^1, \cdots, X^7)$ and $Y = (Y^1, \cdots, Y^7)$ (X^i, Y^j are the slices), and computes a distance (a measure of dissimilarity) between each pair of slices. The distance $\delta(X,Y)$ is defined as the minimum of the distances between all pairs of slices (X^i, Y^j).

Proof of Theorem 5.1
Applying the definition (1) and using (2), the probability of a correct match, $\mu_{minD}(i) = \bar{\mu}(i)$, is

$$\begin{aligned} p(\mu_{minD}(D)(i) = j \mid \bar{\mu}(i) = j) &= p\left(d(i,j) \le d(i,k) \forall k;\ d(i,j) \le d^* \mid \bar{\mu}(i) = j\right) \\ &= \int_0^{d^*} p(d(i,j) = x \mid \bar{\mu}(i) = j) \prod_{k \neq j} Prob(d(i,k) \ge x \mid \bar{\mu}(i) = j) dx \\ &= \int_0^{d^*} f(x) [\tilde{G}(x)]^{M-1} dx. \end{aligned}$$

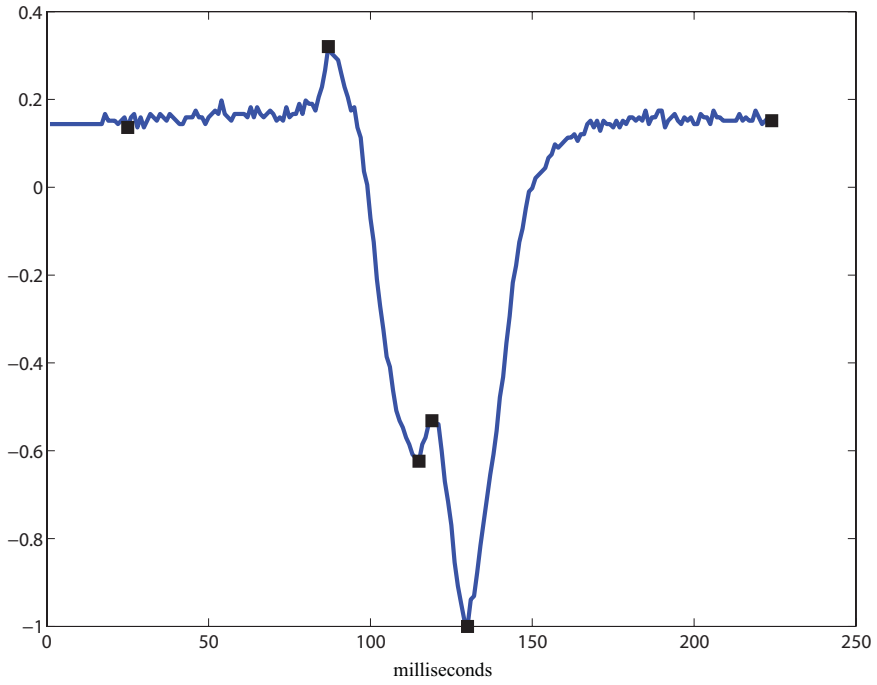

Figure 12: The raw z-axis magnetic signal generated by a vehicle; the squares denote the peaks extracted from the signal.

The probability of an incorrect match is

$$\begin{aligned} p(\mu_{minD}(D) \neq j \mid \bar{\mu}(i) = j) &= p(d(i,j) > \min_{k \neq j} d(i,k);\ \min_{k \neq j} d(i,k) \leq d^* \mid \bar{\mu}(i) = j) \\ &= \int_0^{d^*} Prob(d(i,j) > x \mid \bar{\mu}(i) = j) p(\min_{k \neq j} d(i,k) = x \mid \bar{\mu}(i) = j) dx \\ &= (M-1) \int_0^{d^*} [\tilde{G}(x)]^{M-2} g(x) \tilde{F}(x) dx. \end{aligned}$$

To arrive at the last equality, we use the facts that $Prob(d(i,j) > x \mid \bar{\mu}(i) = j) = \tilde{F}(x)$ and

$$Prob(\min_{k \neq j} d(i,k) \leq x \mid \bar{\mu}(i) = j) = 1 - Prob(d(i,k) \leq x, k \neq j \mid \bar{\mu}(i) = j) = 1 - [\tilde{G}(x)]^{M-1},$$

which, upon differentiating with respect to x, gives the density

$$p(\min_{k \neq j} d(i,k) = x \mid \bar{\mu}(i) = j) = (M-1)[\tilde{G}(x)]^{M-2} g(x).$$

Lastly, the probability that $\mu_{minD}(i) = \tau$ is

$$\begin{aligned} p(\mu_{minD}(D)(i) = \tau \mid \bar{\mu}(i) = j) &= p\left(d(i,j) \geq d^*;\ d(i,k) \geq d^*, k \neq j \mid \bar{\mu}(i) = j\right) \\ &= \tilde{F}(d^*)[\tilde{G}(d^*)]^{M-1}. \end{aligned}$$

This proves (5)-(7). □

Proof of Theorem 5.2

For any matching function μ, we can express $\rho(\mu)$ as

$$\begin{aligned} N\rho(\mu) &= \sum_{i=1}^{N}\Big[\sum_{j=1}^{M} p(\mu(D)(i)=j \mid \bar{\mu}(i)=j)p(\bar{\mu}(i)=j) \\ &\qquad + p(\mu(D)(i)=\tau \mid \bar{\mu}(i)=\tau)p(\bar{\mu}(i)=\tau)\Big] \qquad (27) \\ &= \sum_{i=1}^{N}\Big[\alpha\sum_{j=1}^{M}\int p(\mu(D)(i)=j \mid D)p(D \mid \bar{\mu}(i)=j)dD \\ &\qquad + \beta\int p(\mu(D)(i)=\tau \mid D)p(D \mid \bar{\mu}(i)=\tau)dD\Big]. \qquad (28) \end{aligned}$$

From (2)-(4) and (9)

$$p(D \mid \bar{\mu}(i)=j) = p(D_i \mid \bar{\mu}(i)=j)p(D_{-i}) = L(d(i,j))\gamma(D_i)p(D_{-i}),\ \ p(D \mid \bar{\mu}(i)=\tau) = \gamma(D_i)p(D_{-i}),$$

in which $p(D_{-i}) = \prod_{l\neq i} p(D_l)$. Substitution into (28) yields

$$\begin{aligned} N\rho(\mu) &= \sum_{i=1}^{N}\Big[\alpha\sum_{j=1}^{M}\int p(\mu(D)(i)=j \mid D)L(d(i,j))\gamma(D_i)p(D_{-i})dD \\ &\qquad +\beta\int p(\mu(D)(i)=\tau \mid D)\gamma(D_i)p(D_{-i})dD\Big], \qquad (29) \end{aligned}$$

We see from (29) that $\mu^*(D)(i)$ is given by that j which maximizes the integrand. This selection leads to (10). □

Proof of Corollary 5.1
From (27)

$$N\rho(\mu^*) = \sum_{i=1}^{N}\Big[\alpha\sum_{j=1}^{M} p(\mu^*(D)(i)=j \mid \bar{\mu}(i)=j) + \beta p(\mu(D)(i)=\tau \mid \bar{\mu}(i)=\tau)\Big]. \qquad (30)$$

From (10)

$$\begin{aligned} p^{\alpha}(M) &= p(\mu^*(D)(i)=j \mid \bar{\mu}(i)=j) \\ &= p(L(d(i,j)) \geq \max\{L(d(i,k)), k\neq j, \beta/\alpha\} \mid \bar{\mu}(i)=j) \qquad (31) \\ p^{\beta}(M) &= p(\mu(D)(i)=\tau \mid \bar{\mu}(i)=\tau) = p(\beta/\alpha \geq \max\{L(d(i,k))\} \mid \bar{\mu}(i)=\tau). \qquad (32) \end{aligned}$$

In (31)-(32), the random variables $d(i,j)$ and $d(i,k)$ are independent; $d(i,j)$ is distributed according to f and the $d(i,k)$ are all distributed according to g. The random variable on the right hand side of the inequalities in (31)-(32) increases with M, because it is the maximum of more random variables, whereas the random variable on the left hand side does not change with M. Thus both probabilities $p^{\alpha}(M)$ and $p^{\beta}(M)$ decrease with M. So, from (30) $\rho(\mu^*) = M\alpha p^{\alpha}(M) + \beta p^{\beta}(M)$ decreases with M.

We can use (31), (32) to calculate $\rho(\mu^*)$. From (31),

$$\begin{aligned} p^{\alpha}(M) &= p(L(d(i,j) \geq \max_{k\neq j} L(d(i,k)), L(d(i,j) \geq \beta/\alpha \mid \bar{\mu}(i)=j) = \int_{\beta/\alpha}^{\infty} f_L(l)[G_L(l)]^{M-1}dl, \\ p^{\beta}(M) &= p(\max_{k} L(d(i,k) \leq \beta/a \mid \bar{\mu}(i)=\tau) = [G_L(l)]^M. \end{aligned}$$

This gives (13). □