



Real-time measurement of link vehicle count and travel time in a road network

Journal:	<i>Transactions on Intelligent Transportation Systems</i>
Manuscript ID:	draft
Manuscript Type:	Regular Papers
Date Submitted by the Author:	n/a
Complete List of Authors:	Kwong, Karric; Sensys Networks, Inc. Kavaler, Robert; Sensys Networks, Inc. Rajagopal, Ram; University of California, Electrical Engineering and Computer Sciences Varaiya, Pravin; University of California, Electrical Engineering and Computer Sciences
Keywords:	travel time, Traveler Information Systems, Road vehicle identification, Sensor Fusion, Sensory aids
Abstract:	A system is described for measuring the vehicle count and travel time in the links of a road network. The measurements require matching vehicle signatures recorded by a wireless magnetic sensor network. The matching algorithm is based on a statistical model of the signatures. The model itself is estimated from the data. The approach is first discussed for a single lane road, and extended to multiple lane roads. The algorithm yields a correct matching rate of 75% for a false matching rate of 5%, and reliably estimates the number of vehicles on each link and its travel time distribution. The system is tested on a 0.9 mile-long segment of San Pablo Avenue in Albany, CA.



Real-time measurement of link vehicle count and travel time in a road network

Karric Kwong and Robert Kavalier

Sensys Networks, Inc.

2560 Ninth Street, Berkeley, CA 94710

{karric,kavalier}@sensysnetworks.com

Ram Rajagopal and Pravin Varaiya

Department of Electrical Engineering and Computer Sciences

University of California, Berkeley, CA 94720

{ramr,varaiya}@eecs.berkeley.edu

Abstract

A system is described for measuring the vehicle count and travel time in the links of a road network. The measurements require matching vehicle signatures recorded by a wireless magnetic sensor network. The matching algorithm is based on a statistical model of the signatures. The model itself is estimated from the data. The approach is first discussed for a single lane road, and extended to multiple lane roads. The algorithm yields a correct matching rate of 75% for a false matching rate of 5%, and reliably estimates the number of vehicles on each link and its travel time distribution. The system is tested on a 0.9 mile-long segment of San Pablo Avenue in Albany, CA.

Index Terms

magnetic sensors, signature matching, link vehicle count, vehicle re-identification, travel time, spatial occupancy

Research supported in part by California Department of Transportation and by AROMURI-UCSC-W911NF-05-1-0246-VA-09/05

I. INTRODUCTION

A road network is an interconnection of links such as freeway sections, on- and off-ramps, and urban road segments. At any time a link has a certain *spatial occupancy* or *vehicle count*, the number of vehicles in the link. At the prevailing speed the number of vehicles that will move from an upstream link depends on its vehicle count, and the number of vehicles that a downstream link can accept is limited by the downstream link vehicle count. Thus the *state of the road network* at any time consists of the vehicle count and speed (or travel time) in every link.

At some link interconnection junctions, vehicle movement is controlled by programmable field elements such as intersection signals, ramp-metering signals, and message signs announcing emergency conditions, speed limits, tolls, and travel time estimates.

The evolution of traffic in the road network is governed by its state and the signal settings and messages selected by the algorithms being executed in the field elements. These selections are based on an estimate of the current network state. The better the quality of this estimate, the more effectively can algorithms improve road network performance. Because current detectors (loops, video, radar) measure vehicle volume, speed, and occupancy at fixed locations, they cannot directly measure the state of a road network.

We describe a system that measures the link vehicle count and travel time of a road network in real time. The system deploys magnetic wireless sensors at the ends of links. As vehicles move over the sensors, their magnetic signatures are recorded and a matching procedure is used to track their movement. A very efficient algorithm calculates optimum matchings. The system is tested in an urban arterial road segment. This appears to be the first reliable and cost effective means for measuring vehicle counts and travel times in arterial roads and freeways.

Related prior work is summarized in Section II. The test site and the measurement system are described in Section III. The matching problem and the statistical model used to evaluate matching algorithms occupy Section IV. Optimal matching algorithms are presented in Section V. A method to calibrate the statistical signature model is given in Section VI. A real time extension of the matching algorithm is described in Section VII. Empirical results and an evaluation of the algorithm's performance are presented in Section VIII. Conclusions and directions for future work can be found in Section IX.

II. LITERATURE REVIEW

Schemes for estimating travel times based on matching inductive loop signatures at two detector locations were demonstrated in [10], [8], [6]. All require an independent speed measurement that is

used to ‘normalize’ the signature waveform, assuming constant vehicle speed. If a vehicle is accelerating or decelerating, this assumption is invalid and, as [6] reports, the rate of correct matching then drops drastically. Lengths of vehicles in platoons at the two locations are compared in [1], also requiring an independent speed measurement. None of these schemes would work satisfactorily in a link with traffic signals causing stop-and-go movement. Platoon vehicle lengths used in [1] and platoon vehicle colors used in [11] would not work well for the additional reason that intersections would break platoons up. Vehicles can be re-identified by matching unique tags or license plates; but besides raising privacy concerns, these schemes require mounting an overhead camera or tag reader in each lane making it too expensive to deploy over an arterial network.

The method of [9] estimates the average travel time across a signalized link based on a kinematic wave model, using 30-second flow and occupancy measurements from an upstream loop detector and the exact times of the red and green phases. The similar method of [5] uses the exact time each vehicle crosses the detector. The two approaches require precise signal phase times, which must be synchronized with the detector times. For a link with multiple intersections, each intersection must be instrumented, which is expensive.

The system presented here requires no information about signal timings and does not require every intersection to be instrumented. Furthermore, it gives individual vehicle travel times and, unlike all these methods, also measures link vehicle counts. The use of the system to deduce signal phases and measure arterial performance is discussed in detail in [4].

III. MEASURING LINK VEHICLE COUNT AND TRAVEL TIME

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

As a vehicle crosses a sensor, it numbers the vehicle consecutively, registers the time, and records its magnetic signature (described more fully in Section IV). Thus the upstream sensor generates a triple (i, s_i, X_i) for each vehicle: $(5, s_5, X_5), (6, s_6, X_6), \dots$; the downstream sensor generates triples (j, t_j, Y_j) : $(18, t_{18}, Y_{18}), (19, t_{19}, Y_{19}), \dots$. The measurement triples are sent via radio to an Access Point (AP) on the side of the road.



Fig. 1. Vehicle re-identification by signature matching.

The AP matches the signatures in real time. As suggested by the figure, upstream vehicle 5 is the same as downstream vehicle 18, that is, their signatures X_5 and Y_{18} match; similarly, X_6 and Y_{19} match. On the other hand, upstream vehicle $i = 7$ has turned away from the lane before reaching the downstream sensor; similarly $\tau \rightarrow Y_{20}$ indicates that downstream vehicle $j = 20$ turned into the lane but did not cross the upstream sensor.

We now describe how the AP estimates link vehicle count and travel time. At any time t , the AP finds the number K of the most recent upstream vehicle that was registered before t ($K = 140$ in the figure) and the number I of the most recent upstream vehicle that was matched with a downstream signature J ($I = 6$, $J = 19$). Then the number of vehicles in the link at time t is estimated as $K - I$ ($140 - 6 = 134$), and the link travel time of the most recent departing vehicle is $t_J - s_I$ ($t_{19} - s_6$). This ‘vehicle re-identification via signature matching’ scheme gives in real time the link vehicle count and travel time.

We can bound the vehicle count measurement error assuming that every vehicle is detected, which is justified [3], but some vehicle matches may be missed. The index K of the most recent upstream vehicle is then correct. But the most recent upstream vehicle $I_{max} \geq I$ that leaves before t may be missed by the matching algorithm. Hence, if there are no turning movements, the true vehicle count $N(t)$ is

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

Equality will not hold if $I < I_{max}$, which happens when upstream vehicle I_{max} is not matched. If the matching probability is p , on average $I_{max} - I = p^{-1} - 1$, so if $p > 0.5$ (which it is in the test results), the estimate $K - I$ differs from $N(t)$ by at most 1 on average.

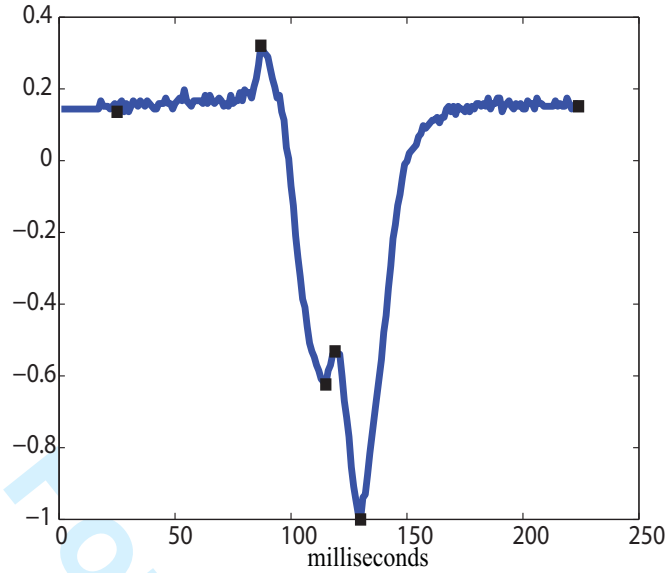


Fig. 2. Raw z-axis magnetic signal recorded by a vehicle and peak values.

If there are turning movements, the bound 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 $i = 7$ in the figure) that turned before reaching the downstream sensor, and n_{in} is the number of vehicle with index larger than J (like $j = 20$) that came into the link without crossing the upstream vehicle.

IV. MATCHING PROBLEM

Suppose over an observation interval the upstream and downstream sensors generate the arrays of triples $\{(i, s_i, X_i), 1 \leq i \leq N\}$ and $\{(j, t_j, Y_j), 1 \leq j \leq M\}$. The matching is done in two steps. In the *signal processing* step each pair (X_i, Y_j) of upstream and downstream signatures is compared to produce a distance $d(i, j) = \delta(X_i, Y_j) \geq 0$ between them. This step thereby reduces the signature data to the $N \times M$ matrix $D = \{d(i, j) \mid 1 \leq i \leq N, 1 \leq j \leq M\}$.

We will shortly describe the signal processing step, followed by a statistical model of the distance function $d(i, j)$. The model is used to evaluate any *matching* or function μ ,

$$\mu: \{1, \dots, N\} \rightarrow \{1, \dots, M, \tau\}, \quad (\text{IV.1})$$

with this interpretation: $\mu(i) = j$ or τ accordingly as upstream signature X_i is matched to downstream

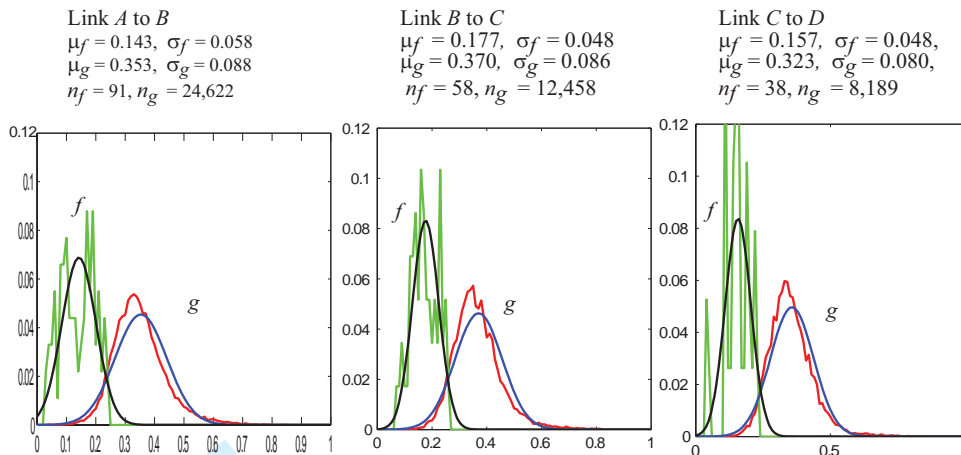


Fig. 3. The empirical pdfs f and g and their Gaussian approximations for links $A \rightarrow B$, $B \rightarrow C$ and $C \rightarrow D$.

signature Y_j or is not matched to any downstream signature, as in Figure 1. Let \mathcal{M} be the set of all matchings.

The second step formulates the matching problem, which is to find the matching $\mu \in \mathcal{M}$ that is ‘closest’ to the true matching, denoted $\bar{\mu}$.

A. Signal processing

A sensor comprises an array of seven nodes, each with a three-axis magnetometer that measures the x, y, z directions of the earth’s magnetic field sampled at 128Hz as a vehicle goes over the node. Figure 2 shows the raw z -axis measured signal from one node. The other axes measurements are similar.

The microprocessor in the node automatically extracts the sequence of peak values (local maxima and minima) from each of these signals. In the figure, 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 AP. The three axes yield three such arrays. The three arrays form a slice of the vehicle’s two-dimensional magnetic ‘footprint’. A slice 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. A vehicle’s signature is unique, making it possible to distinguish vehicles with the same model and make. (See [3] for a full description of a node and an AP.)

The signal processing algorithm takes two signatures, say $X = (X^1, \dots, X^7)$ and $Y = (Y^1, \dots, Y^7)$ (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 the minimum of the distances between all pairs of slices.

B. Statistical model of 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 (X_v, Y_v) between the signatures at the upstream and downstream sensors of the *same* randomly selected vehicle v ,

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

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

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

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

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

We assume that conditional on the true matching $\bar{\mu}$ 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

$$p(D | \bar{\mu}) = \prod_i p(D_i | \bar{\mu}(i)) \quad (\text{IV.2})$$

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

in which

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

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

Figure 3 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 V describes how the distributions in Figure 3 are estimated.

C. Matching problem

An *unconstrained matching* in the general form (IV.1) permits duplicate matches and overtaking, i.e., two upstream vehicles i_1, i_2 with $i_2 > i_1$ may be matched with j_1, j_2 in the reverse order $j_2 < j_1$. A *constrained matching* does not permit this. Thus, it is a pair of matched sequences like $(Up, Down)$:

$$\begin{array}{cccccccc} Up & = & X_1 & \tau & X_2 & X_3 & X_4 & X_5 & X_6 \\ & & \downarrow & \downarrow & \downarrow & \downarrow & \downarrow & \downarrow & \downarrow \\ Down & = & \tau & Y_1 & Y_2 & \tau & Y_3 & Y_4 & Y_5 \end{array} \quad (IV.5)$$

The interpretation of (IV.5) is clear. Formally, a constrained matching is a matching μ without duplicates (except for τ) and without overtaking, i.e.,

$$i_2 \geq i_1 \Rightarrow \mu(i_1) \not\geq \mu(i_2).$$

We want to find μ_c^* , the *maximum a posteriori* (MAP) matching for the constrained case. μ_c^* maximizes the posterior probability

$$p(\bar{\mu} | D) = \frac{p(D | \bar{\mu})p_c(\bar{\mu})}{p(D)}, \quad (IV.6)$$

in which $p_c(\bar{\mu})$ denotes the prior probability that $\bar{\mu}$ is the true constrained matching. In (IV.6), $p(D | \bar{\mu})$ is given by the signature distance model (IV.2)-(IV.4), so we only need to specify the prior $p_c(\bar{\mu})$, which we take to be the unconstrained prior $p(\bar{\mu})$ given below in (IV.8), conditioned by the requirement that $\bar{\mu}$ is a constrained matching. That is,

$$p_c(\bar{\mu}) = \begin{cases} p(\bar{\mu}) / \sum_{\bar{\mu} \in \mathcal{M}_c} p(\bar{\mu}) & \bar{\mu} \in \mathcal{M}_c \\ 0 & \bar{\mu} \notin \mathcal{M}_c \end{cases} \quad (IV.7)$$

in which \mathcal{M}_c denotes the set of constrained matchings.

The unconstrained prior $p(\bar{\mu})$ is the uniform distribution on $\bar{\mu}$ with turning probability β :

$$\begin{aligned} p(\bar{\mu}) &= \prod_i p(\bar{\mu}(i)); \quad p(\bar{\mu}(i) = j) = \alpha, \quad j = 1, \dots, M; \\ p(\bar{\mu}(i) = \tau) &= \beta, \end{aligned} \quad (IV.8)$$

with $M\alpha + \beta = 1$. Using (IV.2)-(IV.4) and (IV.8) gives

$$p(D | \bar{\mu})p(\bar{\mu}) = \prod_i p(D_i | \bar{\mu}(i))p(\bar{\mu}(i)), \quad (IV.9)$$

$$p(D_i | \bar{\mu}(i))p(\bar{\mu}(i)) = \begin{cases} L(d(i, j))\gamma(D_i)\alpha, & \bar{\mu}(i) = j \\ \gamma(D_i)\beta, & \bar{\mu}(i) = \tau \end{cases} \quad (IV.10)$$

Thus μ_c^* is given by

$$\mu_c^* = \arg \max_{\bar{\mu} \in \mathcal{M}_c} \frac{p(D | \bar{\mu})p(\bar{\mu})}{p(D)} = \arg \max_{\bar{\mu} \in \mathcal{M}_c} p(D | \bar{\mu})p(\bar{\mu}). \quad (\text{IV.11})$$

The last equality follows from $p(D) = \sum_{\bar{\mu} \in \mathcal{M}_c} p(D|\bar{\mu})p(\bar{\mu})$ not depending on $\bar{\mu}$. To find μ_c^* we must maximize the likelihood (IV.9) over the set \mathcal{M}_c . The algorithm for μ_c^* is developed in section V.

The MAP *unconstrained matching* μ_u^* , given by

$$\mu_u^* = \arg \max_{\bar{\mu} \in \mathcal{M}} p(D | \bar{\mu})p(\bar{\mu}), \quad (\text{IV.12})$$

is fully analyzed in [4].

The unconstrained MAP matching $\mu_u^*(i)$ is determined independently for each upstream vehicle i . But the constrained MAP matchings $\mu_c^*(i)$ are all jointly determined.

D. Multiple lane matching

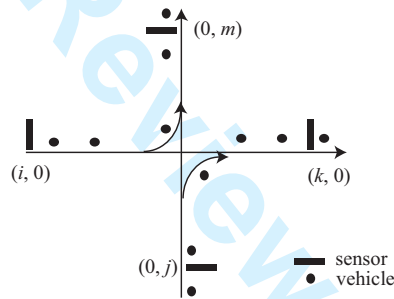


Fig. 4. Multiple lane matching.

We consider matching vehicles that may switch from one lane to another lane downstream, as in the junction in Figure 4. Vehicles i and j , denoted by $(i, 0)$ and $(0, j)$ in the figure, may continue straight ahead or turn at the junction. The no-overtaking condition now means that if vehicles i and i' with $i < i'$ are in the same lane upstream, and happen to be matched to vehicles in the same lane downstream, the match respects $\mu(i) < \mu(i')$.

Suppose I vehicles from the first input sequence and J vehicles from the second input sequence are to be matched to K vehicles in the first output sequence and M vehicles in the second output sequence. An input or output vehicle maybe unmatched, denoted as matching with τ .

We use the notation $(i, 0)$, $(0, j)$, $(k, 0)$, $(0, m)$ to index the four vehicle types, so we can distinguish between (say) the third vehicle in the first input sequence $(3, 0)$ and the third vehicle in the second input

sequence $(0,3)$. However, we reserve i, j, k, m to denote a generic vehicle in these four sequences, sometimes writing i or m instead of $(i,0)$ or $(0,m)$.

We are given the data array

$$D = \{d(i,k), d(i,m), d(j,k), d(j,m) | i \leq I, \\ j \leq J, k \leq K, m \leq M\}$$

of distances between the signatures of each input vehicle and each output vehicle, together with the times t_i, t_j, t_m, t_k when the signatures were recorded. A matching μ is now any function

$$\mu : \{(i,0), 1 \leq i \leq I\} \cup \{(0,j), 1 \leq j \leq J\} \rightarrow \\ \{(k,0), 1 \leq k \leq K\} \cup \{(0,m), 1 \leq m \leq M\} \cup \{\tau\},$$

with the natural interpretation: for example, $\mu(i,0) = (0,m)$ means vehicle i in the first input sequence is matched with vehicle m in the second output sequence; $m(0,j) = \tau$ means that vehicle j in the second input sequence is unmatched. Let \mathcal{M} be the set of all matchings. A *constrained matching* μ is a matching without duplicates (except for τ) and without overtaking, i.e.,

$$(i,j) \geq (i',j') \Rightarrow \mu(i',j') \not\geq \mu(i,j).$$

Here $(i,j) \geq (i',j')$ means $i \geq i', j' \geq j$. Let $\mathcal{M}_c \subset \mathcal{M}$ be the set of all constrained matchings.

We now generalize the statistical model. Conditional on the true matching $\bar{\mu}$, the data array D has the following distribution:

$$p(D|\bar{\mu}) = \prod_i p(D_{(i,0)}|\bar{\mu}(i,0)) \prod_j p(D_{(0,j)}|\bar{\mu}(0,j)) \\ p(D_{(i,0)}|\bar{\mu}(i,0)) = \begin{cases} \frac{f(d(i,\bar{\mu}(i,0)))}{g(d(i,\bar{\mu}(i,0)))} \prod_k g(d(i,k)) \prod_m g(d(i,m)) \\ \prod_k g(d(i,k)) \prod_m g(d(i,m)) \end{cases} \\ = \begin{cases} L(d(i,\bar{\mu}(i,0)))\gamma(i,0), & \bar{\mu}(i,0) \neq \tau \\ \gamma(i,0), & \bar{\mu}(i,0) = \tau \end{cases} \\ p(D_{(0,j)}|\bar{\mu}(0,j)) = \begin{cases} L(d(j,\bar{\mu}(0,j)))\gamma(0,j), & \bar{\mu}(0,j) \neq \tau \\ \gamma(0,j), & \bar{\mu}(0,j) = \tau \end{cases}$$

Here

$$L(d(i,\bar{\mu}(i,0))) = \frac{f(d(i,\bar{\mu}(i,0)))}{g(d(i,\bar{\mu}(i,0)))}, \quad \bar{\mu}(i,0) \neq \tau \\ L(d(j,\bar{\mu}(0,j))) = \frac{f(d(j,\bar{\mu}(0,j)))}{g(d(j,\bar{\mu}(0,j)))}, \quad \bar{\mu}(0,j) \neq \tau$$

and

$$\begin{aligned}\gamma(i, 0) &= \prod_k g(d(i, k)) \prod_m g(d(i, m)) \\ \gamma(0, j) &= \prod_k g(d(j, k)) \prod_m g(d(j, m))\end{aligned}$$

Similarly to the one-dimensional case, we assume that under the *prior distribution* on \mathcal{M} the $\mu(i, j)$ are all independent and uniformly distributed, subject to the condition that $p(\mu(i, j) = \tau) = \beta$. That is,

$$p(\mu(i, j) = (k, m) \neq \tau) = \alpha, \quad p(\mu(i, j) = \tau) = \beta,$$

with

$$(K + M)\alpha + \beta = 1.$$

For constrained matchings the prior distribution is

$$p_c(\mu) = \begin{cases} \frac{p(\mu)}{\sum_{\mathcal{M}_c} p(\mu)}, & \mu \in \mathcal{M}_c \\ 0, & \mu \notin \mathcal{M}_c \end{cases},$$

and the optimal constrained matching μ_c^* is

$$\mu_c^* = \arg \max_{\bar{\mu} \in \mathcal{M}_c} \frac{p(D | \bar{\mu}) p(\bar{\mu})}{p(D)} = \arg \max_{\bar{\mu} \in \mathcal{M}_c} p(D | \bar{\mu}) p(\bar{\mu}). \quad (\text{IV.13})$$

E. Applications

The multiple lane formulation applies with obvious changes to the junction of Figure 5 with several input and output streams of vehicles, provided the non-overtaking or first-in first-out (fifo) condition holds. (Each violation of the fifo condition will lead to one τ matching. If there are not too many fifo violations, the technique is sound.)

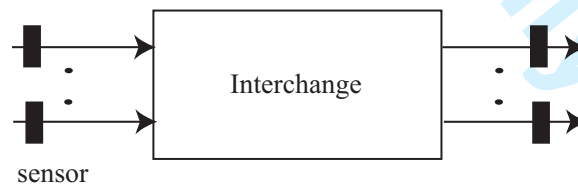


Fig. 5. A generalized setup.

Intersections: Figure 5 could represent an intersection of two streets one going West to East, the other going South to North as in Figure 4. The matching results yield an estimate of the number of vehicles making a turn; the start and end times of the matched vehicle pairs give the travel time to cross the intersection and to make a turn.

Roundabout: This is just an extension of the intersection with R input and output sequences if R roads terminate on the roundabout.

Weaving: Estimating the number and nature of weaving movements in (say) a four-lane weaving section on a freeway corresponds to the arrangement of Figure 5 with four input and output sequences and sensors placed at the beginning and end of the weaving sections.

Ramp queue and delay: A ramp with two lanes that merge at the signal corresponds to Figure 5 with two input and one output sequence. The vehicle matchings give the number of vehicles in each ramp lane and the delay each vehicle encounters on the ramp.

Work zone: A four-lane freeway that terminates into a two-lane freeway, because a work zone takes two lanes out of service, corresponds to Figure 5 with four input sequences and two output sequences. The sensors are placed upstream and downstream of the work zone. The matchings will estimate the number of vehicles queued up upstream of the work zone and the delay experienced by those vehicles. The information could be displayed on a message sign.

V. MATCHING ALGORITHM

We now develop an efficient algorithm for the optimal constrained single lane matching problem. The multiple lane extension follows as a direct generalization.

A. Single lane matching

Instead of maximizing the likelihood (IV.9) it is more convenient to minimize the negative ‘log-likelihood’,

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

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

$$\lambda(i, j) = -\ln L(d(i, j)) - \ln \gamma(D_i) - \ln \alpha, \tag{V.2}$$

$$\lambda(i, \tau) = -\ln \gamma(D_i) - \ln \beta. \tag{V.3}$$

Thus to find μ_u^* we must minimize the linear form (V.1) over the set $\bar{\mu} \in \mathcal{M}$, which leads to

$$\mu_u^*(i) = \begin{cases} j, & \lambda(i, j) \leq \lambda(i, k), \text{ all } k, \tau \\ \tau, & \lambda(i, \tau) \leq \lambda(i, k), \text{ all } k \end{cases}$$

To find μ_c^* we must minimize the linear form (V.1) over the ‘‘combinatorial’’ constraint $\bar{\mu} \in \mathcal{M}_c$. The difficulty is to find a convenient representation of \mathcal{M}_c .

We now describe a graph $\mathcal{G}(N, M)$ whose paths are in one-one correspondence with the set \mathcal{M}_c of all constrained matchings. Its $(N + 1) \times (M + 1)$ nodes are arranged in the form of a grid like in Figure 6, which is the graph for example (IV.5) with $N = 6$, $M = 5$. $\mathcal{G}(N, M)$ is called the *edit graph* in the context of sequence comparison algorithms in molecular biology [7].

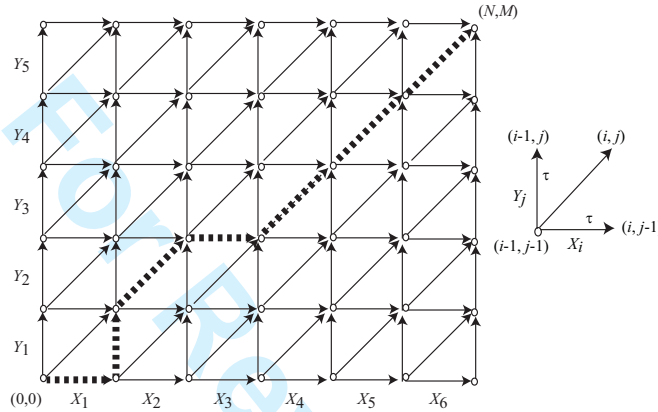


Fig. 6. The edit graph for example (IV.5). A diagonal edge corresponds to a signature match; a horizontal or vertical edge corresponds to a turn (match with τ).

$\mathcal{G}(N, M)$ is constructed as follows. Its nodes are labeled (i, j) , $0 \leq i \leq N$, $0 \leq j \leq M$. 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 $(i - 1, j - 1) \rightarrow (i, j)$ indicates the match $X_i \rightarrow Y_j$; the ‘horizontal’ edge $(i - 1, j - 1) \rightarrow (i, j - 1)$ indicates the match $X_i \rightarrow \tau$; the ‘vertical’ edge $(i - 1, j - 1) \rightarrow (i - 1, j)$ indicates the match $Y_j \rightarrow \tau$.

An obvious but very important fact is that each path in $\mathcal{G}(N, M)$ from node $(0, 0)$ to (N, M) corresponds to a constrained matching and vice versa. The constrained matching (IV.5) corresponds to the path in Figure 6 indicated by the thick dashed lines.

Having identified constrained matchings with paths in the edit graph, we identify (V.1) with the sum of the weights of the edges along the path, assigning edge weights according to

$$\begin{aligned}
 w((i - 1, j - 1) \rightarrow (i, j)) &= \lambda(i, j) \\
 w((i - 1, j - 1) \rightarrow (i, j - 1)) &= \lambda(i, \tau) \cdot \\
 w((i - 1, j - 1) \rightarrow (i - 1, j)) &= 0
 \end{aligned}
 \tag{V.4}$$

The value (V.1) for a 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 μ_c^* is obtained by finding the minimum weight path, which is accomplished by the following algorithm.

Let \mathcal{N} be the nodes of $\mathcal{G}(N, M)$. For each $(i, j) \in \mathcal{N}$ let $Pr(i, j)$ be the *predecessor* nodes of (i, j) , i.e., the nodes from which there is an edge to (i, j) . Evidently,

$$Pr(i, j) = \{(i-1, j), (i, j-1), (i-1, j-1)\}.$$

Let \prec be a total ordering of \mathcal{N} which respects the Pr relation, i.e., for $n = (i, j)$

$$n' = (i', j') \in Pr(n) \Rightarrow n' \prec n.$$

There are many such total orders, including lexicographic order.

Algorithm

1. Set $W(0, 0) = 0$.
2. Suppose $W(n')$ has been evaluated for all $n' \prec n$. Calculate

$$W(n) = \min\{W(n') + w(n' \rightarrow n) \mid n' \in Pr(n)\}, \quad (\text{V.5})$$

in which $w(n' \rightarrow n)$ is the weight of the edge $n' \rightarrow n$ given by (V.4), and let $pr(n)$ be a minimizing predecessor node in (V.5).

3. Return to step 2 with the node following n in the total order \prec .

Then $W(i, j)$ is the minimum weight of paths connecting $(0, 0)$ to (i, j) . The minimum weight path can be constructed by backtracking through $pr(n)$.

The algorithm requires $N \times M$ iterations. In each iteration (V.5) requires evaluation of the three edge weights $w(n' \rightarrow n)$ given by (V.4).

B. Multiple lane matching

To find μ_c^* requires minimizing the negative of the likelihood (IV.13) over $\mu \in \mathcal{M}_c$, which can be written as

$$\begin{aligned}
-\ln p(D | \bar{\mu})p(\bar{\mu}) &= \sum_i [\sum_k \lambda((i, 0), (k, 0)) \mathbf{1}(\bar{\mu}(i, 0) = (k, 0)) \\
&+ \sum_m \lambda((i, 0), (0, m)) \mathbf{1}(\bar{\mu}(i, 0) = (0, m)) \\
&+ \lambda((i, 0), \tau) \mathbf{1}(\bar{\mu}(i, 0) = \tau)] \\
&+ \sum_j [\sum_k \lambda((0, j), (k, 0)) \mathbf{1}(\bar{\mu}(0, j) = (k, 0)) \\
&+ \sum_m \lambda((0, j), (0, m)) \mathbf{1}(\bar{\mu}(0, j) = (0, m)) \\
&+ \lambda((0, j), \tau) \mathbf{1}(\bar{\mu}(0, j) = \tau)]. \tag{V.6}
\end{aligned}$$

In (V.6) $\mathbf{1}(\cdot)$ is the indicator function and

$$\begin{aligned}
\lambda((i, 0), (k, 0)) &= -\ln L(d(i, k)) - \ln \gamma(i, 0) - \ln \alpha \\
\lambda((i, 0), (0, m)) &= -\ln L(d(i, m)) - \ln \gamma(i, 0) - \ln \alpha \\
\lambda((i, 0), \tau) &= -\ln \gamma(i, 0) - \ln \beta \\
\lambda((0, j), (k, 0)) &= -\ln L(d(j, k)) - \ln \gamma(0, j) - \ln \alpha \\
\lambda((0, j), (0, m)) &= -\ln L(d(j, m)) - \ln \gamma(0, j) - \ln \alpha \\
\lambda((0, j), \tau) &= -\ln \gamma(0, j) - \ln \beta \tag{V.7}
\end{aligned}$$

Like in the single lane case, (V.6) is a linear form, the combinatorial constraint is $\bar{\mu} \in \mathcal{M}_c$, and the weights are given in (V.7). The constraint can be represented using the graph $\mathcal{G}(I, J, K, M)$, with $(1 + I) \times (1 + J) \times (1 + K) \times (1 + M)$ nodes corresponding to a four-dimensional grid with nodes indexed (i, j, k, m) .

From each node (i, j, k, m) there are four edges (labeled τ) connecting to ‘adjacent’ nodes $(i + 1, j, k, m)$, $(i, j + 1, k, m)$, $(i, j, k + 1, m)$, $(i, j, k, m + 1)$ and four ‘diagonal’ edges connecting to node $(i + 1, j, k + 1, m)$, $(i + 1, j, k, m + 1)$, $(i, j + 1, k + 1, m)$, $(i, j + 1, k, m + 1)$. The first four edges are interpreted to mean that vehicles $i + 1, j + 1, k + 1, m + 1$ respectively are unmatched. The last four edges are interpreted to mean that $i + 1$ is matched to $k + 1$, $i + 1$ is matched to $m + 1$, and so on.

$\mathcal{G}(I, J, K, M)$ may be called the *edit graph* in analogy with the single input-single output sequence case. It is an important result that constrained matchings are in 1-1 correspondence to paths in \mathcal{G} from

(0,0,0,0) to (I,J,K,M) . We can then identify (V.1) with the sum of the weights of the edges along the path, assigning edge weights according to

$$\begin{aligned}
w((i-1, j-1, k-1, m-1) \rightarrow (i, j-1, k, m-1)) &= \lambda((i, 0), (k, 0)) \\
w((i-1, j-1, k-1, m-1) \rightarrow (i, j-1, k-1, m)) &= \lambda((i, 0), (0, m)) \\
w((i-1, j-1, k-1, m-1) \rightarrow (i-1, j, k, m-1)) &= \lambda((0, j), (k, 0)) \\
w((i-1, j-1, k-1, m-1) \rightarrow (i-1, j, k-1, m)) &= \lambda((0, j), (0, m)) \\
w((i-1, j-1, k-1, m-1) \rightarrow (i, j-1, k-1, m-1)) &= \lambda((i, 0), \tau) \\
w((i-1, j-1, k-1, m-1) \rightarrow (i-1, j, k-1, m-1)) &= \lambda((0, j), \tau) \\
w((i-1, j-1, k-1, m-1) \rightarrow (i-1, j-1, k, m-1)) &= 0 \\
w((i-1, j-1, k-1, m-1) \rightarrow (i-1, j-1, k-1, m)) &= 0
\end{aligned} \tag{V.8}$$

In (V.8) $w(n' \rightarrow n)$ is the weight assigned to the edge $n' \rightarrow n$. It is easy to check the next result.

Theorem 5.1: For any path from $(0,0,0,0)$ to (I,J,K,M) the weight of the path, calculated as the sum of the edge weights given by (V.8), is equal to the sum (V.1). Hence μ_c^* is given by the minimum weight path.

Let \mathcal{N} be the nodes of $\mathcal{G}(I,J,K,M)$. For each $(i, j, k, m) \in \mathcal{N}$ let $Pr(i, j, k, m)$ be the eight predecessor nodes of (i, j, k, m) from which there is an edge to (i, j, k, m) :

$$\begin{aligned}
Pr(i, j, k, m) = \{ &(i-1, j, k-1, m), (i-1, j, k, m-1), \\
&(i, j-1, k-1, m), (i, j-1, k, m-1), (i-1, j, k, m), \\
&(i, j-1, k, m), (i, j, k-1, m), (i, j, k, m-1) \}.
\end{aligned}$$

Let \prec be a total ordering of \mathcal{N} which respects the Pr relation, i.e., for $n = (i, j, k, l)$

$$n' = (i', j', k', l') \in Pr(n) \Rightarrow n' \prec n.$$

The previous minimum weight path algorithm applies without change except that W is a four-dimensional array, initialized with $W(0,0,0,0) = 0$. $W(I,J,K,M)$ is the minimum weight of paths connecting $(0,0,0,0)$ to (I,J,K,M) . The minimum weight path can be constructed by backtracking through $pr(n)$. The algorithm requires $I \times J \times K \times M$ iterations.

VI. ESTIMATING THE MODEL

The statistical model (IV.2)-(IV.4) parameterizes the probability density of the observation matrix D as $p(D | \bar{\mu}, \mu_f, \sigma_f, \mu_g, \sigma_g)$, with parameter $\bar{\mu}$ for the true matching, and $(\mu_f, \sigma_f, \mu_g, \sigma_g)$ 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

$$(\hat{\bar{\mu}}, \hat{\mu}_f, \hat{\sigma}_f, \hat{\mu}_g, \hat{\sigma}_g) = \arg \max_{\bar{\mu}, \mu_f, \sigma_f, \mu_g, \sigma_g} p(D | \bar{\mu}, \mu_f, \sigma_f, \mu_g, \sigma_g).$$

While this 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 resort to coordinate-wise optimization:

- 1) Begin with an initial estimate $\bar{\mu}^0$ (for which we use a matching algorithm based solely on the distance).
- 2) At step i we have the estimate $\bar{\mu}^i$. Find

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

This is easy because $\bar{\mu}^i$ partitions 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

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

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

Since the likelihood $p(D | \bar{\mu}^i, \mu_f^i, \sigma_f^i, \mu_g^i, \sigma_g^i)$ increases with i , the iteration must converge to a local maximum of the likelihood. (The iteration is reminiscent of the EM algorithm [2].) For the test results the iteration converged in three to four steps.

VII. REAL-TIME MATCHING

The edit graph of Section V grows with the observation time interval, and with each new upstream or downstream vehicle, one must 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. For real-time implementation, we must 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 $d(i, k)$ as

$$d(i, k) = \begin{cases} \delta(X_i, Y_k) & \text{if } 0 \leq t_k - s_i \leq T_M \\ \infty & \text{else} \end{cases} \quad (\text{VII.1})$$

For each new downstream vehicle j (with signature Y_j) that arrives at time t_j compute the distance $d(m, j)$ as

$$d(m, j) = \begin{cases} \delta(X_m, Y_j) & \text{if } 0 \leq t_j - s_m \leq T_M \\ \infty & \text{else} \end{cases} \quad (\text{VII.2})$$

If T_M is an upper bound on the travel time, (VII.1)-(VII.2) must hold. The number of distances $\delta(X_i, Y_j)$ to be calculated is thus bounded, and 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, signal cycle times, etc. A better idea is this. Let μ be the minimum weight path matching until the current time. Pick an integer M . Let i_1, \dots, i_M be the most recently matched M downstream vehicles, and set

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

Thus T_M is (say) two times the maximum travel time experienced by these M vehicles. This choice will automatically adapt to changes in travel time.

VIII. EXPERIMENTAL RESULTS

We evaluate system performance using both synthetic and field test data. Synthetic ‘data’ are obtained by randomly generating the observation distance matrix D from the distributions in Figure 3(c), conditional on a randomly generated true matching $\bar{\mu}$. We can thereby vary the number of observations, turns, and overtakings or fifo violations, and compare the results of the optimal constrained and unconstrained matchings, μ_c^*, μ_u^* with $\bar{\mu}$.

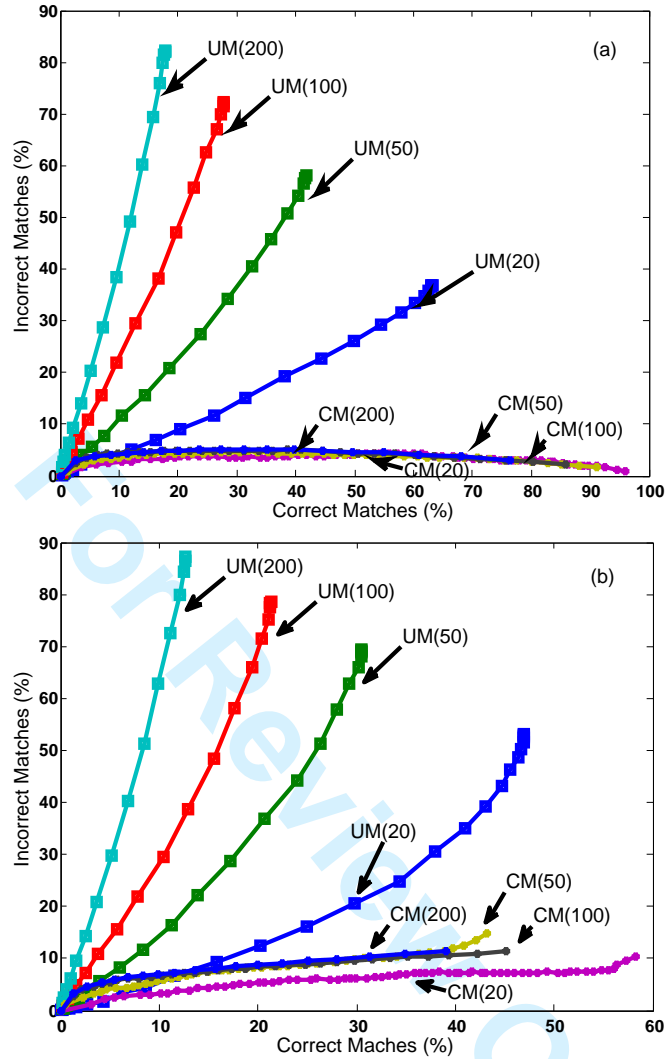


Fig. 7. Error curve for unconstrained (UM) and constrained (CM) matching for various choices of number of vehicles (M) and (a) no overtaking or turns and (b) 10% of vehicles overtake and 25% turn.

A. Synthetic Data

Performance is captured in the number C_M of correct matches when the estimated match is not a turn, and the number E_M of erroneous or incorrect matches when the estimated match is not a turn. $C_M + E_M$ is the number of ‘diagonal’ edges and $M - C_M - E_M$ is the number of ‘horizontal’ edges in the minimum weight path, which we denote by μ_c^* . If T is the number of true turns, i.e., the number of horizontal edges in $\bar{\mu}$, the number of erroneously estimated turns is $E_T = |(M - C_M - E_M) - T|$. (Incorrectly assigning matches as turns reduces the number of samples available for travel time estimates, but does

not necessarily bias the travel time estimate.)

The minimum weight path μ_c^* is a function of the ground truth $\bar{\mu}$, the distance matrix D , and the turn probability β in (V.3) which determines the weight of the horizontal edges. Hence C_M, E_M are functions of $\bar{\mu}, D, \beta$. The smaller β is, the larger is the weight of the horizontal edges, and the smaller will be their number $M - C_M - E_M$ in μ_c^* (and in μ_u^*). Indeed $M - C_M - E_M \rightarrow 0$ as $\beta \rightarrow 0$. Conversely, as the turn probability $\beta \rightarrow 1$, only horizontal edges will be included, so $C_M, E_M \rightarrow 0$.

We simulate two distinct scenarios: in Figure 7(a) there are no turns and no overtakings or fifo violations; in Figure 7(b) there are 25% turns and 10% overtaking vehicles. The figures give the performance of both optimum constrained (CM) and optimum unconstrained matchings (UM), using normalized proportions $\bar{E}_M = E_M/M$ and $\bar{C}_M = C_M/M$ to compare across different numbers of vehicles M . Also, $\bar{E}_T = |(1 - \bar{C}_M - \bar{E}_M) - T/M|$.

Figure 7(a) shows ROC curves for vehicle matching (correct match proportion vs. incorrect match proportion). The curves are parameterized by β : $\bar{C}_M + \bar{E}_M \rightarrow 0$ as $\beta \rightarrow 1$, and $\bar{C}_M + \bar{E}_M \rightarrow 1$ as $\beta \rightarrow 0$. When the fifo constraint is satisfied (a), the error in constrained matching is negligible (nearly 100% correct matches with 0.5% incorrect matches). Unconstrained matching is much worse, with incorrect match percentage increasing with M for a fixed correct match percentage, exactly as the analysis in [4] predicts. (The matchings in [10], [8], [6] are all unconstrained.)

Figure 7(b) shows that even with turns and fifo violations, constrained matching outperforms unconstrained matching, even though the latter permits overtaking. About 50% of vehicles are correctly matched, with an incorrect matching rate under 10%, independently of the number of vehicles. Including turns (25%), the maximum correct matching percentage for this scenario is 75%.

B. Field Data

Data at the test site have been continuously collected since mid-May, 2008; results for May 23 are presented here. Both lanes of the road have sensors. We split site A as 0 and 1; B as 2 and 3; C as 4 and 5; and D as 6 and 7. Link $A \rightarrow B$ comprises the fast or left lane $0 \rightarrow 2$, and the slow or right lane $1 \rightarrow 3$; the fast (even-numbered) and slow (odd-numbered) lanes of the three other links are labeled similarly.

Single lane matching refers to matching only fast lane signatures. Figure 8 displays the travel time distributions for single lane matching at links $A \rightarrow B$, $B \rightarrow C$ and $C \rightarrow D$ for a 30 minute time interval. The legend 'A to B 99/211' means that 99 vehicles out of a possible 211 were matched on $A \rightarrow B$, for a matching rate of 47%; the other rates are 59% and 51%. Also shown is the travel time distribution for the entire 0.9 mile segment $A \rightarrow D$, with a matching rate of 41% between vehicle signatures at A and

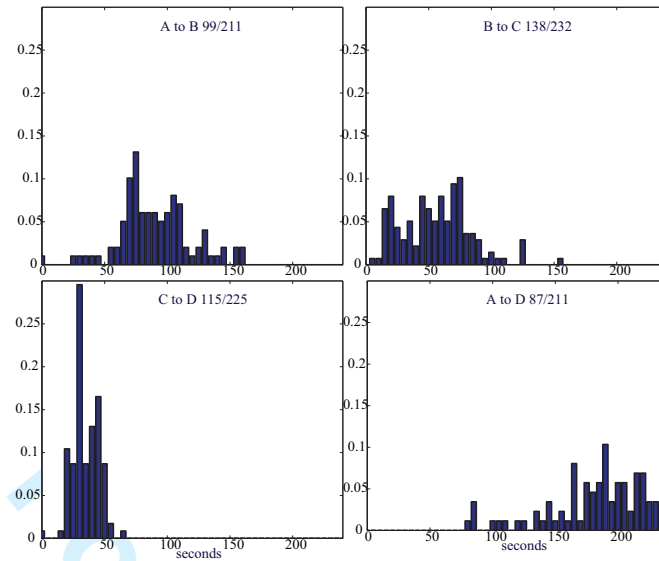


Fig. 8. Travel time distributions for May 23, 2008, 1-1:30PM.

D. (Note that there are six signals between *A* and *D*, Figure 1.) These matching rates, together with the results with simulated data, indicate turning rates near 25%.

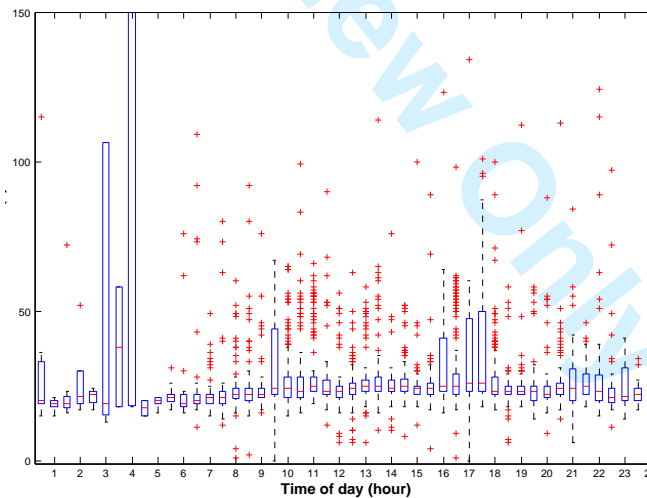


Fig. 9. Box plot of 30 min blocks of travel time samples (in sec) for May 23, 2008, 24 hours.

The measurement system allows construction of the box plots of Figure 9. Each box corresponds to a 30 minute time interval that starts at the time where the box is placed. In the early AM hours, travel times experience immense variability due to vehicles that briefly stop. Congestion forms during 9:30-10AM

and 4-6PM.

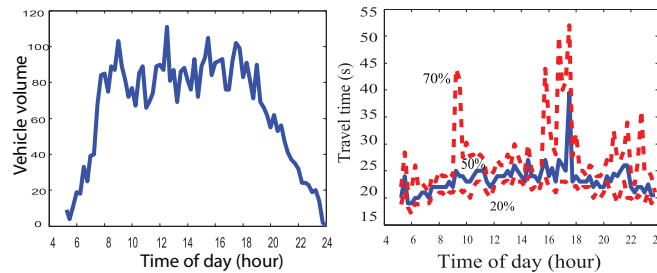


Fig. 10. Vehicle volumes and travel time statistics for 15 minute blocks for May 23, 2008

Figure 10 plots vehicle volume and travel time statistics (median, 20th and 70th percentiles) every 15 minutes, on the fast lane of link $C \rightarrow D$. The lane carries 440 vehicles/hour during the peak. The large variability in the 70th percentile between 9-10AM and 4-6PM is due not to a large vehicle volume but to the large number of turns.

Consider now multiple lane matching for the link $B \rightarrow C$. Figure 11 shows the result of applying the single lane matching algorithm *separately* for lanes $2 \rightarrow 4$ (50% matching rate), and $3 \rightarrow 5$ (38% matching rate). This matching misses vehicles that change lanes.

Figure 12 displays the results of *multiple* lane matching for the same data. The number of vehicles matched for lane $2 \rightarrow 4$ is 46 and for lane $3 \rightarrow 5$ is 35; 21 vehicles are matched for lane $2 \rightarrow 5$ and 17 vehicles are matched for lane $3 \rightarrow 4$.

Comparison of Figures 11 and 12 shows that the distribution of travel times for $2 \rightarrow 4$ and $3 \rightarrow 5$ remains very close for both single and multiple lane matching, confirming that the estimation of travel time distributions is not affected by accounting for each lane separately. However, in the multiple lane matching a total of 119 out of 200 vehicles are matched as compared with 88 out of 200 from single lane matching.

Figure 13 shows the travel time distributions for the four (2×2) matches between C and D : 36 matches for $4 \rightarrow 6$, 15 matches for $4 \rightarrow 7$, 30 matches for $5 \rightarrow 7$ and 7 matches for $5 \rightarrow 6$, for a total of 88 matches, compared with 75 matches (not shown) using two separate single lane matchings.

Figures 12 and 13 indicate that vehicles that change from the 'fast' to the 'slow' lane ($2 \rightarrow 5, 4 \rightarrow 7$) experience a longer travel time than those that remain in the fast lane ($2 \rightarrow 4, 4 \rightarrow 6$). On the other hand, vehicles that change from the 'slow' to the 'fast' lane ($3 \rightarrow 4, 5 \rightarrow 6$) travel faster than those that remain in the slow lane ($3 \rightarrow 5, 5 \rightarrow 7$).

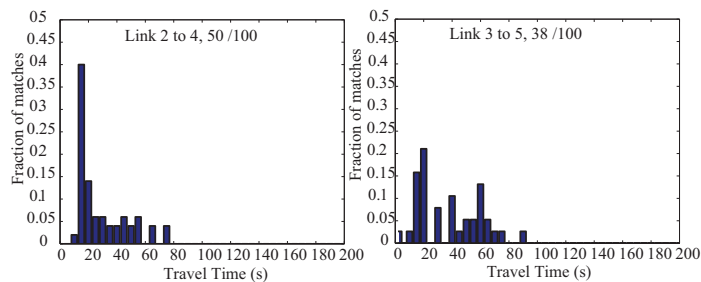


Fig. 11. Single lane matching for lanes 2 to 4 and 3 to 5 for May 23, 7:30-8:00AM

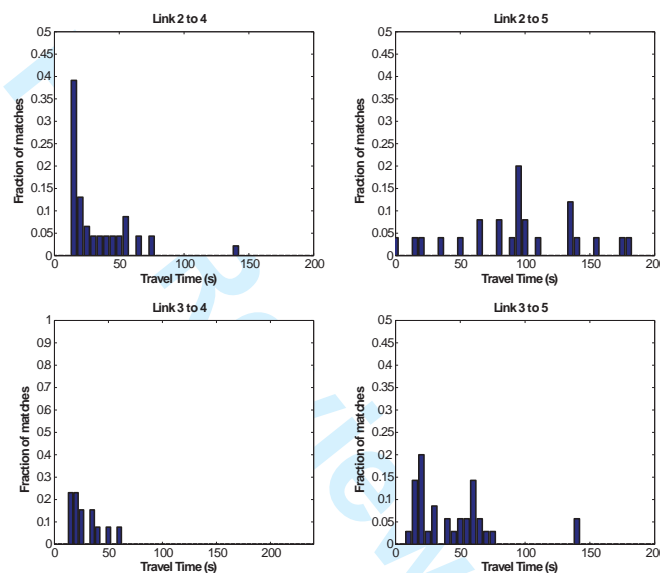


Fig. 12. 2 x 2 matchings for link B to C for May 23, 7:30-8:00AM

Figure 14 shows the use of the test system for traveler information. The figure plots the median travel time every half-hour over the 0.9 mile segment from midnight of 10/20/2008 to midnight of 10/24/2008. (The median travel time is set to 0 if 10 or fewer vehicles traversed the link during the half-hour.) On 10/22/2008 an accident caused the I-880 freeway to be shut down from 6:30AM until 7:25PM. Although the accident was several miles away from San Pablo Avenue, one can observe its impact in the tripling of the travel time median during the afternoon of 10/22 as southbound drivers diverted to San Pablo to avoid the freeway.

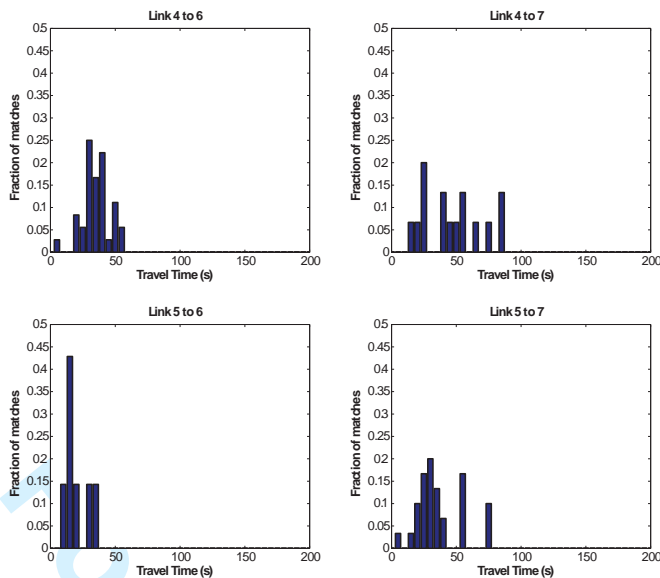


Fig. 13. 2×2 matchings for link $C \rightarrow D$ for May 23, 07:30-8:00 AM

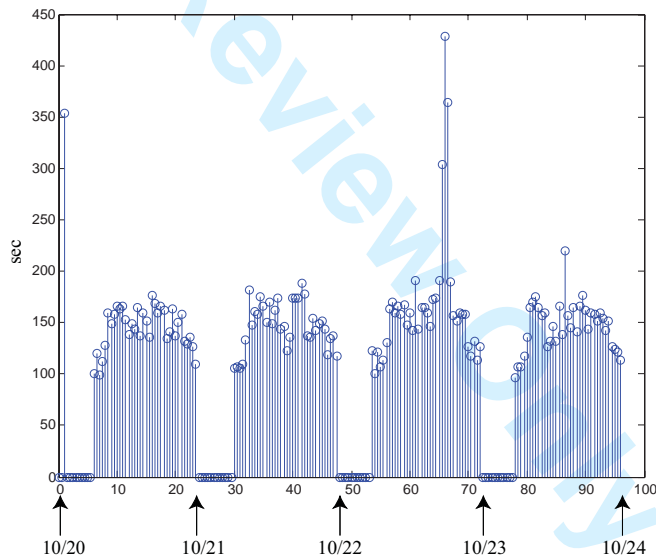


Fig. 14. Median travel time every 30 min from 10/20/2008 to 10/24/2008. Travel time is in sec, and time is in hours beginning midnight of 10/20/2008.

IX. CONCLUSION

We described a system for measuring the state of a road network in real time. This appears to be the first cost-effective, scalable system that provides real-time measurements of vehicle count and individual vehicle travel times in the links of a road network. The system requires deployment of wireless magnetic

sensors at locations that demarcate each link. The deployment is flexible in the way links are defined. A link may span several intersections and not all lanes in a link may be sensed. Deployment can be incremental. The system is based on anonymous matching of vehicle magnetic signatures recorded by sensors at the ends of a link.

The optimum matching algorithm relies crucially on the first in-first out (fifo) or non-overtaking constraint. Violation of this constraint reduces the number of matches, but does not bias the travel time or vehicle count estimates. The algorithm is very efficient and permits real time computation. The matching algorithm works for single-lane as well as multi-lane links. The multi-lane matching algorithm has several applications, including intersections with turns, multi-lane ramps, work zones, and weaving sections.

There are two promising directions of future work. First, suitable modifications of the matching algorithm appear to have use in several situations of mobile sensing. Second, the real-time measurement of the state can improve the performance of several traffic control algorithms. For example, real-time measurement of ramp queues can improve ramp metering control, and measurement of vehicle count in certain freeway sections can improve variable speed limit control.

REFERENCES

- [1] B. Coifman. *Vehicle reidentification and travel time measurement using loop detector speed traps*. PhD thesis, University of California, Berkeley, Berkeley, CA 94720, 1999.
- [2] A. Dempster, N. Laird, and D. Rubin. Likelihood from incomplete data via the EM algorithm. *J. Royal Statistical Society, B*, 39(1):1–38, 1977.
- [3] A. Haoui, R. Kavalier, and P. Varaiya. Wireless magnetic sensors for traffic surveillance. *Transportation Research Part C*, 16(3):294–306, 2008.
- [4] K. Kwong, R. Kavalier, R. Rajagopal, and P. Varaiya. Arterial travel time estimation based on vehicle re-identification using wireless sensors, 2008. Submitted to Transportation Research, Part C.
- [5] H. X. Liu and W. Ma. A virtual vehicle probe model for time-dependent arterial travel time estimation. Submitted for Publication, 2008.
- [6] 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.
- [7] E.W. Myers. An $O(ND)$ difference algorithm and its variations. *Algorithmica*, 1:251–266, 1986.
- [8] C. Oh and S.G. Ritchie. Real-time inductive-signature-based level of service for signalized intersections. *Transportation Research Record*, (1802):97–104, 2002.
- [9] 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.

- 1
2
3
4 [10] C. Sun, S.G. Ritchie, K. Tsai, and R. Jayakrishnan. Use of vehicle signature analysis and lexicographic optimization for
5 vehicle reidentification on freeways. *Transportation Research, C*, 7:167–185, 1999.
6
7 [11] C.C. Sun, G.S. Arr, R.P. Ramachandram, and S.G. Ritchie. Vehicle reidentification using multidetector fusion. *IEEE*
8 *Transactions on Intelligent Transportation Systems*, 5(3):155–164, September 2004.
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60