Transportation Research Part C 75 (2017) 30–44

Contents lists available at ScienceDirect

Transportation Research Part C

journal homepage: www.elsevier.com/locate/trc

Predicting travel time reliability using mobile phone GPS data

Dawn Woodard [a,*,1], Galina Nogin [a,2], Paul Koch [b], David Racz [c,3], Moises Goldszmidt [d,4], Eric Horvitz [b]

[a] Cornell University, Ithaca, NY, United States
[b] Microsoft Research, Redmond, WA, United States
[c] Microsoft Corp., Palo Alto, CA, United States
[d] Microsoft Research, Mountain View, CA, United States

ARTICLE INFO

Article history:
Received 23 March 2015
Received in revised form 6 September 2016
Accepted 24 October 2016
Available online 20 December 2016

Keywords:
Location data
Traffic
Travel time
Forecasting
Statistics

ABSTRACT

Estimates of road speeds have become commonplace and central to route planning, but few systems in production provide information about the reliability of the prediction. Probabilistic forecasts of travel time capture reliability and can be used for risk-averse routing, for reporting travel time reliability to a user, or as a component of fleet vehicle decision-support systems. Many of these uses (such as those for mapping services like Bing or Google Maps) require predictions for routes in the road network, at arbitrary times; the highest-volume source of data for this purpose is GPS data from mobile phones. We introduce a method (TRIP) to predict the probability distribution of travel time on an arbitrary route in a road network at an arbitrary time, using GPS data from mobile phones or other probe vehicles. TRIP captures weekly cycles in congestion levels, gives informed predictions for parts of the road network with little data, and is computationally efficient, even for very large road networks and datasets. We apply TRIP to predict travel time on the road network of the Seattle metropolitan region, based on large volumes of GPS data from Windows phones. TRIP provides improved interval predictions (forecast ranges for travel time) relative to Microsoft's engine for travel time prediction as used in Bing Maps. It also provides deterministic predictions that are as accurate as Bing Maps predictions, despite using fewer explanatory variables, and differing from the observed travel times by only 10.1% on average over 35,190 test trips. To our knowledge TRIP is the first method to provide accurate predictions of travel time reliability for complete, large-scale road networks.

1. Introduction

Several mapping services provide predictions of the expected travel time on an arbitrary route in a road network, in real time and using traffic, time of day, day of the week, and other information. They use these predictions to recommend a route or routes with minimum expected travel time. Microsoft's mapping service (Bing Maps) predicts travel time for large-scale

* Corresponding author.
E-mail address: moises.goldszmidt@gmail.com (M. Goldszmidt).
URL: http://people.orie.cornell.edu/woodard (D. Woodard).
[1] Research performed while Visiting Researcher at Microsoft Research.
[2] Research performed while intern at Microsoft Research.
[3] Current address: Google, Mountain View, CA, United States.
[4] Current address: Apple, Cupertino, CA, United States.

http://dx.doi.org/10.1016/j.trc.2016.10.011
0968-090X/

road networks around the world using a method called Clearflow (Microsoft Research, 2012), which employs probabilistic graphical models learned from data to predict flows on arbitrary road segments. The method, which has its roots in the earlier Smartphlow effort on forecasting highway flows and reliability (Horvitz et al., 2005), considers evidence about real-time traffic conditions, road classifications, topology of the road network, speed limits, time of day and day of week, and numerous other variables. With Clearflow, travel time predictions made on all segments across a geographic region are used in route-planning searches (Delling et al., in press).

Beyond expected flows, it is important to consider uncertainty in travel time caused for instance by unpredictable traffic light schedules, accidents, unexpected road conditions, and differences in driver behavior. Such travel time variability (conversely, its *reliability*) also strongly affects the desirability of routes in the road network (Jenelius, 2012; Texas Transportation Institute, 2015). For fleets of delivery vehicles, such as those transporting perishables, decisions including routing need to provide on-time deliveries with high probability. In the case of ambulance fleets, taking into account uncertainty in the travel time of an ambulance to potential emergency scenes leads to improved ambulance positioning decisions, and consequently increases the survival rate of cardiac arrest patients (Erkut et al., 2007). A prediction of the probability distribution of travel time can be more valuable than a deterministic prediction of travel time, by accounting not just for measured traffic congestion and other known conditions, but also for the presence of unmeasured conditions. Distribution predictions of travel time can be used for risk-averse routing, for reporting travel time reliability to a user (e.g. the travel time is predicted to be in the range 10–15 min), and as a component of fleet vehicle decision-support systems (Samaranayake et al., 2012; Westgate et al., 2016).

We introduce a statistical solution to predicting the distribution of travel time on an arbitrary route in the road network, at an arbitrary future time. We call the method TRIP (for travel time reliability inference and prediction). For typical road networks of interest, the number of possible routes is extremely large, and any particular route may have very few or no observed trips in the historical data. For these reasons it is infeasible to apply methods designed for prediction on a particular set of heavily traveled routes, such as Jenelius and Koutsopoulos (2013), Ramezani and Geroliminis (2012), Rahmani et al. (2015). TRIP uses information from all the trips in the historical data to train a model for travel time on routes, learning the characteristics of individual roads and the effect of time of the week, road classification, and speed limit. We model travel time variability both at the trip level and at the level of the individual road network links included in the route. This decomposition is appropriate because some sources of variability affect the entire trip (such as driver habits, vehicle characteristics, or unexpected network-wide traffic conditions), while other sources of variability are localized (such as a delay due to a train crossing or construction). We define a network link to be a directed section of road that is not divided by an intersection, and on which the measured features of the road (road classification, speed limit, number of lanes, etc.) are constant.

TRIP captures important features of the data, including weekly cycles in congestion levels, heavy right skew of travel time distributions, and probabilistic dependence of travel times for different links within the same trip (for example, if the travel speed is high on the first link of the route, the speed is also likely to be high on the other links of the route). We capture the multimodality of travel time distributions using a mixture model where the mixture components correspond to unobserved congestion states, and model the probabilistic dependence of these congestion states across the links of the route using a Markov model. Because we model travel time for individual links, the travel time prediction can be updated en route.

We introduce a computational method for training and prediction based on maximum a posteriori estimation via Expectation Conditional Maximization (Meng and Rubin, 1993). This yields an iterative training process with closed-form update equations that can be computed using parallelization across links and trips. As a result it is computationally efficient even on large road networks and for large datasets.

TRIP uses GPS data from vehicle trips on the road network; we obtain large volumes of such trips using anonymized mobile phone GPS data from Windows phones in the Seattle metropolitan region. We compare the accuracy of our predictions to a variety of alternative approaches, including Clearflow. The GPS location and speed measurements are illustrated in Fig. 1, which shows that they contain valuable information regarding the speed of traffic on individual roads. Unlike other sources of vehicle speed information (Hofleitner et al., 2012b), vehicular GPS data does not require instrumentation on the roadway, and can achieve near-comprehensive coverage of the road network. Additionally, there is increasing evidence that traffic conditions can be estimated accurately using only vehicular GPS data (Work et al., 2010). One challenge of mobile phone GPS data is that it is often sampled at low frequency (typically 1–90 s between measurements). A related data source, GPS data from fleet vehicles, is also often sampled at low frequency (Rahmani et al., 2015), and TRIP can also be applied to such data.

Some existing approaches to predicting the probability distribution of travel time on a road network model exclusively link-level variability, and assume independence of travel time across the links in the route (Westgate et al., 2013; Hunter et al., 2013). This leads to considerable underprediction of the amount of variability (Westgate et al. (2013) and Section 4). Dependence across links is incorporated by Hofleitner et al. (2012a,b), who use a mixture model for travel time on links where the mixture component represents a congestion state (as we do). They allow these congestion states to depend on the link and the time, and model dependence of their congestion states across the road network and across time using a dynamic Bayesian network. This approach is intuitive but computationally demanding (leveraging a high-dimensional particle filter in each iteration of the algorithm), so is unlikely to be efficient enough to use on complete road networks (they apply it to 800 links in the San Francisco arterial network). Additionally, the method still underpredicts the amount of variability in travel time. Motivated by evidence in the data (Section 4), we allow the congestion state to additionally depend on the whole trip. We model dependence of this congestion state across the links of the route, instead of across all links of the

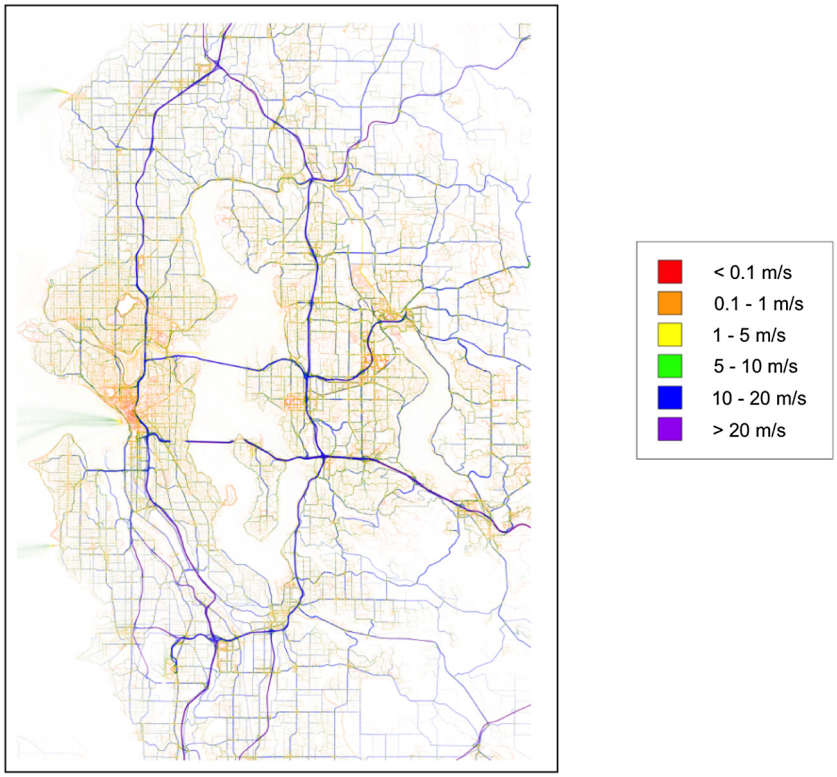

Fig. 1. Anonymized GPS locations from Windows phones in the Seattle metropolitan region, aggregated into 4.8 m × 3.2 m grid boxes and colored by average speed in the grid box. Image resolution has been reduced for privacy reasons. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

network. This improves the flexibility of the model, leading to accurate variability prediction. It also facilitates computation: because our specification corresponds to a one-dimensional Markov model, computation can be done exactly and efficiently in each iteration, by using the forward-backward algorithm (cf., Russell and Norvig (2009)).

Another existing method for prediction of travel time variability (Hunter et al., 2013) is designed for high-frequency (e.g. every second) GPS measurements. These authors estimate the number of stops of the vehicle on each link of the route, and introduce a model for the number of stops and a model for the distribution of travel time conditional on the number of stops. There are also some methods designed for emergency vehicles, which directly model the distribution of travel time on the entire trip, as a function of quantities like the route distance (Budge et al., 2010; Westgate et al., 2016). This is appropriate for emergency vehicles because the relevant data source (lights-and-sirens emergency vehicle trips) is too low-volume to effectively model the distribution of travel times for individual links; for non-emergency data it does not work as well as our approach, as we show in Section 4.

The scale of the road network for which we do prediction of travel time distributions (221,980 links) is an order of magnitude larger than networks studied in existing work for non-emergency vehicles. Hunter et al. (2013) consider a network of over half a million links, but then remove the links with too few observations; presumably this limits the predictions to routes that do not include any of the deleted links.

A major use case for TRIP is in risk-averse routing. Conceptually speaking, one can recommend a route by optimizing a route selection criterion based on variability (for example, minimizing a particular percentile of travel time). However, unlike expected travel time, most variability-based criteria are not additive across the links in the route. For this reason, variability predictions from TRIP cannot be directly used to do route planning via standard algorithms like Dijkstra, A*, or hub-labeling (Delling et al., in press). However, one could use a hybrid procedure that first obtains a set of candidate routes by minimizing an additive criterion such as expected travel time, then ranks those routes according to a variability-based criterion. Ranking according to the 90th percentile of travel time, for example, would provide a risk-averse route selection. Alternatively, the candidate routes could be directly shown to users, along with associated predictive ranges for travel time, allowing the user to select the route according to their risk preferences.

A second use case for TRIP is in fleet vehicle decision systems, such as ride-sharing platforms. These are similar to the ambulance positioning use case, in that they require travel time predictions conditional on origin and destination but unconditional on route. This use case could be addressed by obtaining an optimal route as above and then predicting the travel

time distribution for that route. Such an approach is taken for ambulances, as described in Westgate et al. (2013, 2016). Although some bias is introduced due to the driver not always following the optimal route, one can adjust for this bias.

In Section 2 we describe our statistical model and in Section 3 present methods for training and prediction with the model. Section 4 gives the Seattle case study, including providing support for our modeling choices and reporting our prediction results. We draw conclusions and discuss extensions in Section 5.

2. Modeling

TRIP uses GPS measurements recorded periodically during vehicle trips. Each GPS observation consists of location, speed, and heading measurements, along with a time stamp. For mobile phones, a GPS measurement may be recorded whenever phone applications access GPS information, many of which are not mapping or routing applications. For this reason, the frequency of GPS measurements vary, and the phone is not necessarily in a moving vehicle at the time when the measurements are taken. However, motorized vehicle trips can be isolated using a set of heuristics.

For the Seattle case study, we identify sequences of at least 3 GPS measurements from the same device that satisfy requirements such as: (a) starting and ending with speed measurements of at least 3 m/s, (b) having median speed of at least 5 m/s and maximum speed of at least 9 m/s, and (c) covering distance at least 1 km (as measured by the sum of the great circle distances between pairs of sequential measurements). The resulting GPS sequences appear to consist almost exclusively of motorized vehicular trips that follow paths in the road network. The requirements regarding the median and maximum speeds, for example, eliminate most trips from non-motorized travel such as biking or walking. We define each trip to start with the first GPS reading and end with the last GPS reading in the sequence. Consequently, the total trip duration is observed precisely (as the difference of the two time stamps). The median time between GPS readings in the resulting trips is 16 s.

The next step is to estimate the route taken in each trip $i \in \mathcal{I}$, by which we mean the sequence $R_i = (R_{i,1}, \ldots, R_{i,n_i})$ of links traversed (so that $R_{i,k}$ for $k \in \{1, \ldots, n_i\}$ is an element of the set $\mathcal{J}$ of network links), the distance $d_{i,k}$ traversed for each link $R_{i,k}$ (so that $d_{i,k}$ is equal to the length of link $R_{i,k}$ for all except possibly the first and last link of the trip), and the travel time $T_{i,k}$ on each link $R_{i,k}$. Obtaining this estimate is called *map-matching*, a problem for which there are numerous high-quality approaches (Newson and Krumm, 2009; Hunter et al., 2014).

Some of those approaches provide the uncertainty associated with the path and with the link travel times. Although this uncertainty can be taken into account during statistical modeling, we have found little benefit to this approach in prior work on predicting travel time distributions on routes (see Section 5.2 of Westgate et al. (2016)). This is due to the fact that the start and end locations and times for the trips are known with a high degree of accuracy, and the uncertainty is primarily with regards to the allocation of that total time to the links of the route. Ignoring this uncertainty can affect the estimates of the model parameters (Jenelius and Koutsopoulos, 2015), but in our analysis did not significantly affect the predictions for the travel time on the entire trip.

It is common practice to obtain estimated link travel times for the historical data, and then to estimate parameters of a travel time model using those link travel times (Hofleitner et al., 2012b; Westgate et al., 2016). In order to take into account the uncertainty, one either has to (a) handle a large number of *latent variables* (unobserved quantities) that make estimation of travel time parameters far more computationally intensive (Hunter et al., 2009; Westgate et al., 2013); or (b) assume that travel times on links are multivariate Gaussian or independent gamma distributed (Jenelius and Koutsopoulos, 2013; Hofleitner et al., 2012a; Hunter et al., 2013). The latter assumptions don't hold even approximately in our dataset; see Section 4.

For these reasons, we use deterministic rather than probabilistic estimates of $R_{i,k}$, $d_{i,k}$, and $T_{i,k}$ as obtained from a standard map-matching procedure. First, we obtain the route estimate using the method of Newson and Krumm (2009). Then, we allocate the total travel time to the links of the route as follows. Map the GPS observations to the closest location on the route; for sequential pairs of GPS points, allocate the time spent between those points proportionally to the length of the full or partial links along the route between the two points. Then the total time spent on each link is calculated as the sum of the full or partial times associated with that link. Trips that don't follow the road network closely are discarded; these can be from travel by train, for example. Although our focus is on personal vehicular travel, it is possible that some of the trips are from bus or other transit modes. We note that this is an unavoidable challenge when using mobile phone location data.

Having obtained the values $T_{i,k}$, we model $T_{i,k}$ as the ratio of several factors:

$$T_{i,k} = \frac{d_{i,k}}{E_i S_{i,k}} \qquad i \in \mathcal{I}, k \in \{1, \ldots, n_i\} \tag{1}$$

where E_i and $S_{i,k}$ are positive-valued latent variables associated with the trip and the trip-link pair, respectively. The latent variable E_i captures the fact that the trip i may have, say, 10% faster speeds than average on every link in the trip. This could occur for example due to traffic conditions that affect the entire trip, or to driver habits or vehicle characteristics. The latent variable $S_{i,k}$ represents the vehicle speed on the link before accounting for the trip effect E_i, and captures variability in speed due to local conditions such as local traffic situations, construction on the link, and short-term variations in driver behavior.

The model (1) decomposes the variability of travel time on route R_i into two types: link-level variability captured by $S_{i,k}$, and trip-level variability captured by E_i.

We model E_i with a log-normal distribution:

$$\log(E_i) \sim N(0, \tau^2) \tag{2}$$

for unknown variance parameter τ^2. The evidence from the Seattle mobile phone data that supports our modeling assumptions is discussed in Section 4. Other data sets may have different characteristics, and the assumption (2) can be replaced if needed with a t-distribution on $\log(E_i)$ without substantively affecting the computational method described in Section 3 (Liu, 1997). Additionally, the variance τ^2 can be allowed to depend on the time at which trip i begins, the type of route (highway, arterial, etc.) and other factors. Again, this has little impact on the computational method or complexity. For the Seattle data, we found that the distribution of estimated trip effects E_i varied too little across times of week and parts of the road network to motivate this extension.

We model $S_{i,k}$ in terms of an unobserved discrete congestion state $Q_{i,k} \in \{1, \ldots, \mathcal{Q}\}$ affecting the traversal of link $R_{i,k}$ in trip i. Notice that this congestion state is allowed to depend on the trip, so that $Q_{i,k}$ could be different for two trips traversing the same link $R_{i,k}$ at the same time. This is motivated by features of the data, as we show in Section 4. Assume that the week has been divided into time bins $b \in \mathcal{B}$ that reflect distinct traffic patterns, but do not have to be contiguous (such as "morning rush hour" or "weekend daytime"), and let $b_{i,k}$ be the time bin during which trip i begins traversing link $R_{i,k}$. Conditional on $Q_{i,k}$, we model $S_{i,k}$ with a log-normal distribution:

$$\log(S_{i,k}) \,|\, Q_{i,k} \sim N\left(\mu_{R_{i,k},b_{i,k},Q_{i,k}}, \sigma^2_{R_{i,k},b_{i,k},Q_{i,k}}\right) \tag{3}$$

where $\mu_{j,b,q}$ and $\sigma^2_{j,b,q}$ are unknown parameters associated with travel speed on link $j \in \mathcal{J}$ under conditions $q \in \{1, \ldots, \mathcal{Q}\}$, in time bin $b \in \mathcal{B}$. The normal distribution for $\log(S_{i,k})$ can be replaced with a t distribution, or a skew normal or skew t distribution, as needed without substantively changing the computational method described in Section 3 (Lee and McLachlan, 2013).

We use a Markov model for the congestion states $Q_{i,k}$ (motivated by features in the data; Section 4):

$$\begin{aligned} &\Pr(Q_{i,1} = q) = \gamma_{R_{i,1},b_{i,1}}(q) \\ &\Pr(Q_{i,k} = q | Q_{i,k-1} = \tilde{q}) = \Gamma_{R_{i,k},b_{i,k}}(\tilde{q}, q) \qquad k \in \{2, \ldots, n_i\};\ q, \tilde{q} \in \{1, \ldots, \mathcal{Q}\} \end{aligned} \tag{4}$$

where $\gamma_{j,b}$ is an unknown probability vector for the initial congestion state for trips starting on link j, and $\Gamma_{j,b}$ is the transition matrix for the congestion state on link j conditional on the congestion state in the previous link of the trip, during time bin b. This model captures weekly cycles in the tendency of the link to be congested; for example, there may be a high chance of congestion during rush hour. It also provides a second way to capture dependence of travel time across links (in addition to the trip effect E_i). Our specifications (1)–(4) imply a normal mixture model for $\log(T_{i,k})$; for instance, when $k > 1$ and conditioning on $Q_{i,k-1}$ we obtain

$$\log(T_{i,k}) \,|\, Q_{i,k-1} = \tilde{q} \ \sim \sum_{q=1}^{\mathcal{Q}} \Gamma_{R_{i,k},b_{i,k}}(\tilde{q}, q) N\left(\log d_{i,k} - \mu_{R_{i,k},b_{i,k},q},\ \sigma^2_{R_{i,k},b_{i,k},q} + \tau^2\right). \tag{5}$$

This mixture model captures important features of the data, including heavy right skew of the distributions of the travel times $T_{i,k}$, and multimodality of the distributions of $\log(T_{i,k})$; in particular, a mixture of log-normal distributions provides a good approximation to the distribution of vehicle speeds on individual links (see Section 4). In order to enforce the interpretation of the mixture components q as increasing levels of congestion, we place the restriction $\mu_{j,b,q-1} \leqslant \mu_{j,b,q}$ for each $j \in \mathcal{J}, b \in \mathcal{B}$, and $q \in \{2, \ldots, \mathcal{Q}\}$.

Typically, there are some network links $j \in \mathcal{J}$ with insufficient data (in terms of the number of link traversals $L_j \equiv |\{i \in \mathcal{I}, k \in \{1, \ldots, n_i\} : R_{i,k} = j\}|$) to accurately estimate the link-specific parameters $\mu_{j,b,q}$, $\sigma^2_{j,b,q}, \gamma_{j,b}$, and $\Gamma_{j,b}$. For such links, we use a single set of parameters within each *road category*, by which we mean the combination of road classification (e.g., "highway", "arterial", or "major road") and speed limit, and which is denoted by $c(j)$ for each link j. Defining a minimum number L of traversals, for links with $L_j < L$ we set

$$\begin{aligned} &\mu_{j,b,q} = \mu_{c(j),b,q},\ \sigma^2_{j,b,q} = \sigma^2_{c(j),b,q},\ \gamma_{j,b} = \gamma_{c(j),b},\ \Gamma_{j,b} = \Gamma_{c(j),b} \\ &\quad \text{for } q \in \{1, \ldots, \mathcal{Q}\}, b \in \mathcal{B}, j \in \mathcal{J} : L_j < L \end{aligned} \tag{6}$$

where $\mu_{c,b,q}, \sigma^2_{c,b,q}, \gamma_{c,b}$, and $\Gamma_{c,b}$ are parameters associated with the road category $c \in \mathcal{C}$.

Our travel time model (1)–(6) incorporates both trip-level variability like driver effects, and link-level variability due for example to construction. It also captures the effect of weekly cycles, speed limit, and road classification. Combined with an assumption regarding changes in vehicle speed across the link, it provides a realistic model for the location of the vehicle at all times during the trip. Since links are typically short, we assume constant speed of each vehicle across the link. This assumption can be relaxed using the approach described in Hofleitner et al. (2012a), although those authors find only a mod-

est improvement in predictive accuracy relative to a constant-speed assumption. Our model does not currently take into account observations about real-time traffic conditions, although this is a direction of ongoing work; see Section 5.

3. Training and prediction

Computation is done in two stages: model training (obtaining parameter estimates) and prediction (using those estimates in the model to obtain a forecast distribution). Training can be done offline and repeated periodically, incorporating new data and discarding outdated data. This can be used to accumulate information about rarely observed links and to account for changes in the weekly cycles of travel times, or trends like gradual increases in total traffic volume. Prediction is done at the time when a user makes a routing or travel time query, so must be very computationally efficient.

An alternative Bayesian approach is to take into account uncertainty in the parameter values when doing prediction, by integrating over the posterior distribution of the parameters (their probability distribution conditional on the data). This is typically done using Markov chain Monte Carlo computation (Gelman et al., 2013). However, this approach is more computationally intensive, with complexity that can scale poorly (Woodard and Rosenthal, 2013). Additionally, in other work on travel time distribution prediction, we have found that the parameter uncertainty is dwarfed by the travel time variability (Westgate et al., 2016), so that there is little change in the predictions using the more computationally intensive approach. For notational simplicity, in this description we drop the use of common parameters as in (6).

3.1. Training

We train the model using maximum a posteriori (MAP; cf. Cousineau and Helie (2013)) estimation, an approach in which one estimates the vector θ of unknown quantities in the model to be the value that maximizes the posterior density of θ. Latent variables can either be integrated out during this process (if computationally feasible), or included in the vector θ. For example, in image segmentation using Markov random field models, there is a long history of MAP estimation where θ is taken to include all of the latent variables. In this case the latent variables correspond to the segment associated with each image pixel, of which there can be hundreds of thousands, and approximate MAP estimation of the latent variables provides a computationally tractable solution (Besag, 1986; Grava et al., 2007). We take an intermediate approach, integrating over the congestion variables $Q_{i,k}$ (for which the uncertainty is high), and doing MAP estimation of the trip effects E_i (for which the uncertainty is lower due to having multiple observations $T_{i,k}$ for each trip). So we take $\theta \equiv (\tau, \{\mu_{j,b,q}, \sigma_{j,b,q}, \gamma_{j,b}, \Gamma_{j,b}\}_{j\in\mathcal{J},b\in\mathcal{B},q\in\{1,\ldots,\mathcal{Q}\}}, \{\log E_i\}_{i\in\mathcal{I}})$ to include the parameters and the trip effects. We are able to do MAP estimation of θ using an efficient iterative procedure with closed-form updates. MAP estimation does not depend on the transformations chosen in θ (e.g., the exponentiated MAP estimate of $\log E_i$ is equal to the MAP estimate of E_i, if both are unique). Our computational procedure is guaranteed to obtain only a *local* maximum of the posterior density, but by repeating the procedure multiple times using random initializations one can increase the chance of finding the global maximum.

MAP estimation requires specification of a prior density for the parameters; we use the prior $p(\tau, \{\mu_{j,b,q}, \sigma_{j,b,q}, \gamma_{j,b}, \Gamma_{j,b}\}) \propto 1$ that is uniform on the support of the parameter space. This prior is non-integrable, but leads to valid MAP estimation. Such uniform priors on unbounded parameter spaces are commonly used in situations where there is little or no prior information regarding the parameter values (Section 2.8 of Gelman et al. (2013)); see for instance Gelman (2006) for the use of uniform priors for standard deviation parameters like τ and $\sigma_{j,b,q}$.

Now consider the observed data to consist of the transformed values $\{\log \tilde{S}_{i,k}\}_{i\in\mathcal{I},k\in\{1,\ldots,n_i\}}$ where $\log \tilde{S}_{i,k} \equiv \log d_{i,k} - \log T_{i,k}$ is the log speed during link traversal i,k. Because the prior density is uniform, MAP estimation of θ corresponds to maximizing over θ the product of the likelihood function times the probability density of the trip effects:

$$p(\theta|\{\log \tilde{S}_{i,k}\}) = p(\{\log \tilde{S}_{i,k}\}|\theta)\, p(\theta)/p(\{\log \tilde{S}_{i,k}\}) \propto p(\{\log \tilde{S}_{i,k}\}|\theta)\, p(\{\log E_i\}|\tau). \tag{7}$$

Maximum likelihood estimation of θ would maximize $p(\{\log \tilde{S}_{i,k}\}|\theta)$; by including the second term $p(\{\log E_i\}|\tau)$ in our objective function (7), we reduce noise in the estimated $\log E_i$ values (a technique called regularization; James et al. (2013)). Notice that to compute the likelihood function $p(\{\log \tilde{S}_{i,k}\}|\theta) = \sum_{\{Q_{i,k}\}} p(\{Q_{i,k}, \log \tilde{S}_{i,k}\}|\theta)$ directly, one would have to sum over all the possible values of the latent variables $\{Q_{i,k}\}$, which is computationally infeasible.

Expectation Maximization (EM) is an iterative method for maximum likelihood or MAP estimation in the presence of many latent variables; accessible introductions are given in Hofleitner et al. (2012a) and, in more detail, Bilmes (1997). EM is most efficient when the parameter updates in each iteration can be done in closed form, which is not the case for our model. However, we can obtain closed-form updates using a variant called Expectation Conditional Maximization (ECM; Meng and Rubin (1993)). ECM allows for closed-form updates in situations where the parameter vector can be partitioned into subvectors, each of which would have a closed-form EM update if the remaining parameters were known.

We apply ECM by partitioning the parameter vector θ into the two subvectors $\theta_1 = (\tau, \{\mu_{j,b,q}, \sigma_{j,b,q}, \gamma_{j,b}, \Gamma_{j,b}\})$ and $\theta_2 = \{\log E_i\}$. In ECM, attention focuses on the *complete-data* log posterior density $\log p(\theta|\{Q_{i,k}, \log \tilde{S}_{i,k}\})$, which is equal to a term that does not depend on θ, plus

$$\log p\Big(\{Q_{i,k}, \log \tilde{S}_{i,k}\}|\theta\Big) + \log p(\{\log E_i\}|\tau) = \sum_{i\in\mathcal{I}} \log(\gamma_{R_{i,1},b_{i,1}}(Q_{i,1})) + \sum_{i\in\mathcal{I},k\in\{2,\dots,n_i\}} \log(\Gamma_{R_{i,k},b_{i,k}}(Q_{i,k-1}, Q_{i,k}))$$
$$+ \sum_{i\in\mathcal{I},k\in\{1,\dots,n_i\}} \left[-\frac{\log \sigma^2_{R_{i,k},b_{i,k},Q_{i,k}}}{2} - \frac{(\log \tilde{S}_{i,k} - \log E_i - \mu_{R_{i,k},b_{i,k},Q_{i,k}})^2}{2\sigma^2_{R_{i,k},b_{i,k},Q_{i,k}}} \right]$$
$$+ \sum_{i\in\mathcal{I}} \left[-\frac{\log \tau^2}{2} - \frac{(\log E_i)^2}{2\tau^2} \right]. \tag{8}$$

To update a parameter subvector θ_j in each iteration of ECM, one treats the remaining subvectors $\theta_{[-j]}$ as fixed, and maximizes over θ_j the expectation of (8) with respect to $p(\{Q_{i,k}\}\,|\,\theta^{(t)}, \{\log \tilde{S}_{i,k}\})$. The latter quantity is the probability distribution of the congestion variables conditional on the data and the current value $\theta^{(t)}$ of θ. Such an update leads to an increase in the value of the objective function (7) in each iteration t, and ultimately to a (local) maximizer of (7), as shown in Meng and Rubin (1993).

This yields the procedure given in Algorithm 1. The first step is to run the forward-backward algorithm for each trip. Briefly, the model (4) is a one-dimensional Markov model for $\{Q_{i,k} : k \in 1,\dots,n_i\}$ for each trip i. Also, the model (1)–(3) implies a conditional distribution for the observed data $\log(\tilde{S}_{i,k})$ given $Q_{i,k}$, namely $\log(\tilde{S}_{i,k})\,|\,Q_{i,k}, \log E_i^{(t)} \sim N(\log E_i^{(t)} + \mu^{(t)}_{R_{i,k},b_{i,k},Q_{i,k}}, \sigma^{2,(t)}_{R_{i,k},b_{i,k},Q_{i,k}})$. So $p(\{Q_{i,k}\}\,|\,\theta^{(t)}, \{\log \tilde{S}_{i,k}\})$ is a *hidden Markov model* for each i, for which computation can be done exactly and efficiently using the forward-backward algorithm (cf., Russell and Norvig (2009)).

Algorithm 1. MAP Estimation for TRIP.

1: Initialize $\theta^{(0)}$ and $t = 0$
2: **while** $t = 0$ or $\|\theta^{(t-1)} - \theta^{(t)}\| > \epsilon$ **do**
3: Use the forward-backward algorithm to calculate

$$\phi_{i,k}(q) \equiv \Pr(Q_{i,k} = q\,|\,\theta^{(t)}, \{\log \tilde{S}_{i,k}\})$$
$$\psi_{i,k}(\tilde{q}, q) \equiv \Pr(Q_{i,k-1} = \tilde{q}, Q_{i,k} = q\,|\,\theta^{(t)}, \{\log \tilde{S}_{i,k}\})$$
$$\text{for } i \in \mathcal{I}, k \in \{1,\dots,n_i\}, \text{ and } \tilde{q}, q \in \{1,\dots,\mathcal{Q}\}. \tag{9}$$

4: Update the link parameter and τ estimates as:

$$\mu^{(t+1)}_{j,b,q} = \frac{\sum_{\{i,k:R_{i,k}=j,b_{i,k}=b\}} \phi_{i,k}(q)(\log \tilde{S}_{i,k} - \log E_i^{(t)})}{\sum_{\{i,k:R_{i,k}=j,b_{i,k}=b\}} \phi_{i,k}(q)}$$
$$\sigma^{2,(t+1)}_{j,b,q} = \frac{\sum_{\{i,k:R_{i,k}=j,b_{i,k}=b\}} \phi_{i,k}(q)(\log \tilde{S}_{i,k} - \log E_i^{(t)} - \mu^{(t+1)}_{j,b,q})^2}{\sum_{\{i,k:R_{i,k}=j,b_{i,k}=b\}} \phi_{i,k}(q)}$$
$$\gamma^{(t+1)}_{j,b}(q) = \left(\sum_{i:R_{i,1}=j,b_{i,1}=b} \phi_{i,1}(q) \right) \Big/ \left(\sum_{i:R_{i,1}=j,b_{i,1}=b} 1 \right)$$
$$\Gamma^{(t+1)}_{j,b}(\tilde{q}, q) = \left(\sum_{i,k:R_{i,k}=j,\,b_{i,k}=b,\,k>1} \psi_{i,k}(\tilde{q}, q) \right) \Big/ \left(\sum_{i,k:R_{i,k}=j,\,b_{i,k}=b,\,k>1} \phi_{i,k-1}(\tilde{q}) \right)$$
$$\tau^{2,(t+1)} = \frac{1}{|\mathcal{I}|} \sum_{i\in\mathcal{I}} (\log E_i^{(t)})^2. \tag{10}$$

5: Calculate

$$a_{i,k} = \sum_{q=1}^{\mathcal{Q}} \frac{\phi_{i,k}(q)}{\sigma^{2,(t+1)}_{R_{i,k},b_{i,k},q}}$$
$$h_{i,k} = \sum_{q=1}^{\mathcal{Q}} \frac{\phi_{i,k}(q)\mu^{(t+1)}_{R_{i,k},b_{i,k},q}}{\sigma^{2,(t+1)}_{R_{i,k},b_{i,k},q}}$$

6: Update the trip effect estimates:

$$\log E_i^{(t+1)} = \frac{\sum_{k\in\{1,\dots,n_i\}} (a_{i,k} \log \tilde{S}_{i,k} - h_{i,k})}{1/\tau^{2,(t+1)} + \sum_{k\in\{1,\dots,n_i\}} a_{i,k}}. \tag{11}$$

7: $t = t + 1$
8: **endwhile**
9: Take the parameter estimates to be $\hat{\theta} = \theta^{(t)}$

We iterate until there is no change in the parameter estimates up to the third significant figure. The time complexity of each iteration of the ECM algorithm, implemented without parallelization, is $O(\mathcal{Q}^2|\mathcal{J}|(|\mathcal{B}| + \sum_{i\in\mathcal{I}} n_i/|\mathcal{J}|))$, i.e. the procedure is linear in the average number of traversals per link $(\sum_{i\in\mathcal{I}} n_i)/|\mathcal{J}|$, which is a measure of the spatial concentration of data. If this quantity is held fixed (for example if the same data-gathering mechanism is applied to a larger geographic region), the complexity grows linearly in the size of the road network, $|\mathcal{J}|$. The number of ECM iterations required for convergence is also likely to grow with the size of the network and the spatial concentration of data, but this change in efficiency is difficult to characterize in general. In practice, ECM often converges quickly; for the Seattle case study of Section 4, fewer than 50 iterations are required.

For the case study, training takes about 15 min on a single processor. The road map of North America has about 400 times as many directional links as the Seattle metropolitan region, so it would take roughly 4.2 days to run our implementation for the continent of North America, using the same number of iterations of ECM that we used in Seattle. This time can be reduced dramatically, since each of the parameter updates (10) and (11) can be computed using parallelization across trips and/or links. Since training can be done offline, and the model only needs to be retrained occasionally (e.g., monthly or weekly), the training procedure is expected to be sufficiently fast for use in commercial mapping services (which operate at a continental scale). Even if the spatial concentration of data is increased by several orders of magnitude (e.g., where smartphone-based GPS systems provide coverage for a significant portion of drivers) the training procedure is likely to be efficient enough for commercial use by exploiting parallelism. As explained in the introduction, the trained TRIP model would be applied as a second stage of route planning, after obtaining a set of candidate routes that minimize an additive criterion like expected travel time. The running time of this predictive step is discussed in Section 3.2.

The updates of $\mu_{j,b,q}$, $\sigma_{j,b,q}$, and $\gamma_{j,b}$ in (10) are the same as those in EM for normal mixture models (Bilmes, 1997), except that we restrict the calculation to relevant subsets of the data and adjust for the estimated trip effect $\log E_i^{(t)}$. The update of $\Gamma_{j,b}$ is analogous to that of $\gamma_{j,b}$, but for a transition matrix instead of a probability vector. The update of τ is the maximum likelihood estimate conditional on $\{\log E_i^{(t)}\}_{i\in\mathcal{I}}$. The update of $\log E_i$ in (11) is not a standard form. However, in the special case where the $\sigma^2_{j,b,q}$ are equal for all j, b, and q, the updated value $\log E_i^{(t+1)}$ is approximately the average across k of the difference between $\log \tilde{S}_{i,k}$ and its expectation under the model, which is a reasonable estimator for the trip effect.

3.2. Prediction

Prediction in model (1)–(4) is done by simulation, as shown in Algorithm 2 for a new trip i. For the specified route R_i, starting at a specified time, we simulate $\mathcal{M}$ vectors of travel times $(T_{i,1}^{(m)}, \ldots, T_{i,n_i}^{(m)})$ directly from the model, using the trained parameter values $\hat{\theta}$ from Algorithm 1. During prediction the time bins $b_{i,k}$ for $k > 1$ are not fixed, but rather depend on the simulated value of $\{T_{i,1}^{(m)}, \ldots, T_{i,k-1}^{(m)}\}$. Having obtained simulated values from the distribution of total travel time $\sum_{k=1}^{n_i} T_{i,k}$ in this way, Monte Carlo is used to approximate the expectation, quantiles, percentiles, and other summaries of that distribution.

Algorithm 2. Prediction for TRIP.

for $m \in \{1, \ldots, \mathcal{M}\}$ **do**
  Sample $\log E_i^{(m)} \sim N(0, \hat{\tau}^2)$
  Sample $Q_{i,1}^{(m)}$ from model (4), given the initial time bin $b_{i,1}$ and the estimated vector $\hat{\gamma}_{R_{i,1},b_{i,1}}$
  for $k \in \{1, \ldots, n_i\}$ **do**
    if $k > 1$ **then**
      Determine the time bin $b_{i,k}$ based on the trip start time and $\{T_{i,1}^{(m)}, \ldots, T_{i,k-1}^{(m)}\}$
      Sample $Q_{i,k}^{(m)}$ from model (4), given $Q_{i,k-1}^{(m)}, b_{i,k}$, and $\hat{\Gamma}_{R_{i,k},b_{i,k}}$
    end if
    Sample $S_{i,k}^{(m)}$ from model (3), given $Q_{i,k}^{(m)}, b_{i,k}, \hat{\mu}_{R_{i,k},b_{i,k},Q_{i,k}^{(m)}}$, and $\hat{\sigma}_{R_{i,k},b_{i,k},Q_{i,k}^{(m)}}$
    Set $T_{i,k}^{(m)} = d_{i,k}/(E_i^{(m)} S_{i,k}^{(m)})$
  end for
end for

The time complexity of prediction, without parallelization, is $O(\mathcal{M} \times n_i)$. For the analysis of Section 4 we use $\mathcal{M} = 1000$ Monte Carlo simulations per route; we also tried increasing to $\mathcal{M} = 10,000$, which yielded no measurable changes in the values of several summary statistics of the predictions, and consequently no measurable improvements in the predictive accuracy. One can also calculate the estimated Monte Carlo standard error for a particular route and set $\mathcal{M}_i$ to obtain a specified accuracy level for the prediction on that route i.

Prediction took an average of only 17 ms per route, on a single processor. Commercial mapping services require computation times of less than 50 ms for a route recommendation. By taking the two-stage approach to route recommendation discussed in the introduction (ranking candidate routes according to a probabilistic criterion), and by utilizing parallelization, the running time of our approach is fast enough for routing in commercial mapping services. It should even be efficient enough for more complex queries involving multiple routes, like ranking points of interest by driving time.

4. Seattle case study

We obtained anonymized mobile phone GPS data gathered from Windows phones, for the Seattle metropolitan region during a time window in 2014 (Fig. 1; the precise duration of the time period is kept confidential). No personal identifiers were available. We isolated vehicle trips and estimated the corresponding routes as described in Section 2, yielding 145,657 trips that have distance along the road network of at least 3 km. These trips have mean trip duration of 791 s, maximum trip duration of 6967 s, mean trip distance of 11.4 km, and maximum trip distance of 97.2 km. We divide this dataset into a training dataset consisting of the 110,467 trips from the first three-quarters of the time period, and a test dataset consisting of the 35,190 trips from the last one-quarter of the time period.

In Fig. 2, we show the volume of trips in the training data, by the hour of the week in which the trip began. The volume dips overnight, and peaks associated with morning and evening rush hour are present on weekdays. The volume of recorded trips is highest on Saturday and Sunday daytimes, most likely due to higher usage of GPS-utilizing phone applications on weekends. We define the time bins $b \in \mathcal{B}$ based in part on the changes in volume over the week as seen in Fig. 2, yielding five bins: "AM Rush Hour": weekdays 7–9 AM; "PM Rush Hour": weekdays 3–6 PM; "Nighttime": Sunday-Thursday nights 7 PM–6 AM, Friday night 8 PM–9 AM, and Saturday night 9 PM–9 AM; "Weekday Daytime": remaining times during weekdays; and "Weekend Daytime": remaining times during weekends.

There are 221,980 directional network links in the study region. Of the link traversals in our dataset, 32.8% are on highways, 34.8% are on arterials, 14.0% are on major roads, 12.1% are on surface streets, and 6.3% are on other road classifications. We take the minimum number of observations per parameterized link (as used in (6)) to be $L = 30$. There are 24,990 directional links in the Seattle area road network that satisfy this criterion. Although this is only 11.3% of the links in the network, they account for 85.5% of link traversals.

Using two congestion states ($\mathcal{Q} = 2$), in Fig. 3 we validate the estimated distribution of travel time during evening rush hour for some road links, based on our trained model. We focus on the links that have the highest number of observed travel times L_j, showing the two most commonly observed highway links (omitting links on the same section of highway), the most commonly observed "arterial" link, and the most commonly observed "major road" link. For each of these links, Fig. 3 gives the histogram of the travel times during evening rush hour from the training data, adjusted for the estimated trip effect (i.e., the histogram of $\log T_{i,k} + \log \hat{E}_i$). This adjustment is done so that we can overlay the estimated density from the model in a way that's comparable across trips (the curve in the plot, calculated as $\sum_{q=1}^{\mathcal{Q}} \gamma_{R_{i,k},b_{i,k}}(q) N(\log d_{i,k} - \mu_{R_{i,k},b_{i,k},q}, \sigma^2_{R_{i,k},b_{i,k},q})$).

Fig. 3 illustrates the multimodality of the distribution of travel time. Histograms restricting to particular 15-min time periods have a similar shape to those in Fig. 3, indicating that the multimodality is not caused by aggregating over time periods. The mixture of log-normals used by TRIP appears to fit the observations well. By contrast, assuming a single gamma, normal, or log-normal distribution for travel times in a particular time period, as done for example in Hofleitner et al. (2012a,b), Westgate et al. (2013) and Hunter et al. (2013), leads to poor model fit for this mobile phone data.

Next we motivate use of the Markov model (4) for the congestion states $Q_{i,k}$, and our choice to allow $Q_{i,k}$ to depend on the trip rather than just the link and time. First, we will give evidence that the autocorrelation of log travel times within a trip is high and decreases with distance. Second, we show that the correlation of log travel times for vehicles traversing the same link at roughly the same time is not consistently high. These observations suggest that it is more appropriate to model the congestion level as a property of the individual trip, rather than as a property of the link and time. They also suggest the use of a Markov model for $Q_{i,k}$, which can capture association that decays with distance.

Our example corresponds to the first 10 links of the second route shown in Fig. 4 (highway 520 West). In Fig. 5, we illustrate the correlation of log travel times within and across trips on this sequence of links. The plots in the left column show that the autocorrelation of log travel times within the same trip is high and decreasing with distance. The plots in the middle column show that the correlation of log travel times is not consistently high for pairs of distinct trips traversing the sequence of links in the same 15-min time period. Although the correlation appears high in one of the two plots in the middle column, the scatterplots in the right column show that the travel times do not have a strong association across trips. In summary, the congestion level experienced appears to depend on the trip, and not just the links traversed.

This effect occurs in part due to a high-occupancy vehicle (HOV) lane on this section of highway, so that vehicles traveling in the HOV lane experience less congestion than other vehicles. Additionally, this section of 520 West is just before the interchange with highway 405. The traffic can be congested on 405, which can cause lane-specific congestion on 520 West extending significant distances from congested exits. Such effects from HOV lanes and congested interchanges are common throughout the road network, so that vehicles traveling on the same link in the same time period (per GPS resolution) can experience very different congestion levels depending on the choice of lane. Moreover, the choice of lane can be a function of the intended route.

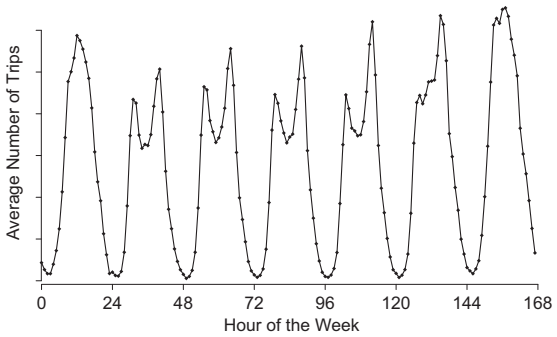

Fig. 2. Volume of trips in the Seattle data, by hour of the week. The scale of the y-axis is omitted for confidentiality.

Next we summarize the model fit and parameter estimates. Fig. 6 shows the distributions of the estimated travel speed parameters, across the links j in the road network. The distribution over links of the estimated mean log-speed parameters $\hat{\mu}_{j,b,q}$ for the uncongested state ($q = 2$) shows two distinct peaks. These peaks correspond to highways (the right peak, at 60–65 mph) and non-highways (the left peak, at 30–40 mph). The values of $\hat{\mu}_{j,b,q}$ for the congested state $q = 1$ are typically much lower than for the uncongested state, and vary considerably across links. This supports the idea that congestion patterns vary considerably across links. These conclusions are consistent across time bins b, although there are some noticeable differences in the $\hat{\mu}_{j,b,q}$ across those bins. In particular, during evening rush hour the high-speed peak is less pronounced, and the low-speed peak is shifted left, relative to nighttime. This corresponds to the fact that speeds tend to be lower during peak times.

The estimated variability $\hat{\sigma}_{j,b,q}$ tends to be lower in the uncongested state than in the congested state. This parameter varies more across links for the congested state than for the uncongested state, again reflecting differences between links in the speed characteristics associated with congestion.

The initial probability $\hat{\gamma}_{j,b}(2)$ of congestion varies widely among links, but is typically below 0.5. Not surprisingly, $\hat{\gamma}_{j,b}(2)$ tends to be higher during evening rush hour than during the night. Regardless of the time period, the probability $\hat{\Gamma}_{j,b}$ of transitioning between congestion states is typically low. This corresponds to the fact that congestion patterns tend to affect more than one link. Additionally, the probability of transitioning from the congested state to the uncongested state tends to be higher than the chance of transitioning from uncongested to congested. This is consistent with the above observation that the uncongested state is more common than the congested state.

The distribution of estimated trip effects $\log \hat{E}_i$ is also shown in Fig. 6. This distribution is roughly symmetric about zero and typical values are in the range -0.2 to 0.2. A value of 0.2, for example, means that the travel speeds in trip i were $\exp(0.2) = 1.22$ times higher than expected.

Having interpreted the model estimates, we now evaluate predictive accuracy. Fig. 4 shows three of the most common routes in the road network: two highway routes and one arterial route. Histograms of the observed travel times for these routes in the test data during evening rush hour are shown, along with the predicted probability density of evening rush hour travel time obtained from TRIP. The observed travel times are obtained from all trips that include the route of interest; these

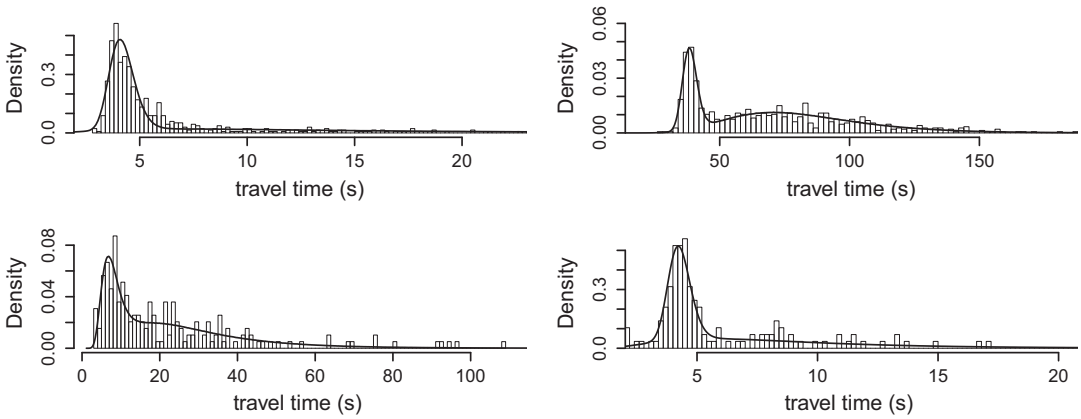

Fig. 3. Validating the distribution of travel times estimated by TRIP on four road links. For each plot, the histogram shows the observed travel times during evening rush hour from the training data, adjusted by removing the estimated trip effect (i.e., $\log T_{i,k} + \log \hat{E}_i$). The curve is the corresponding estimated density from TRIP.

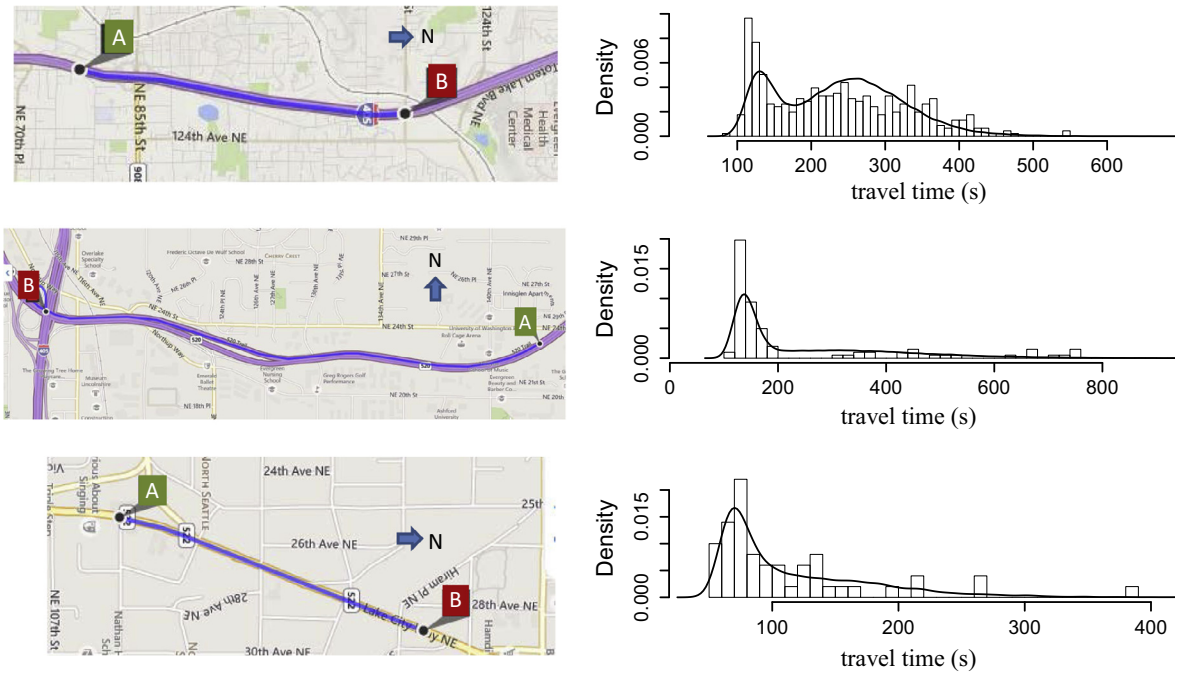

Fig. 4. Three routes in the road network and their predicted travel time distributions during evening rush hour, compared to observed travel times from the test data. Left: the routes; right: histograms of the observed travel times and curves showing the predictive densities. Top: a 3.4 km section of highway 405 North; middle: a 3.3 km route consisting of a section of 520 West followed by the exit ramp to 405 South; bottom: a 0.9 km section of northbound Lake City Way NE.

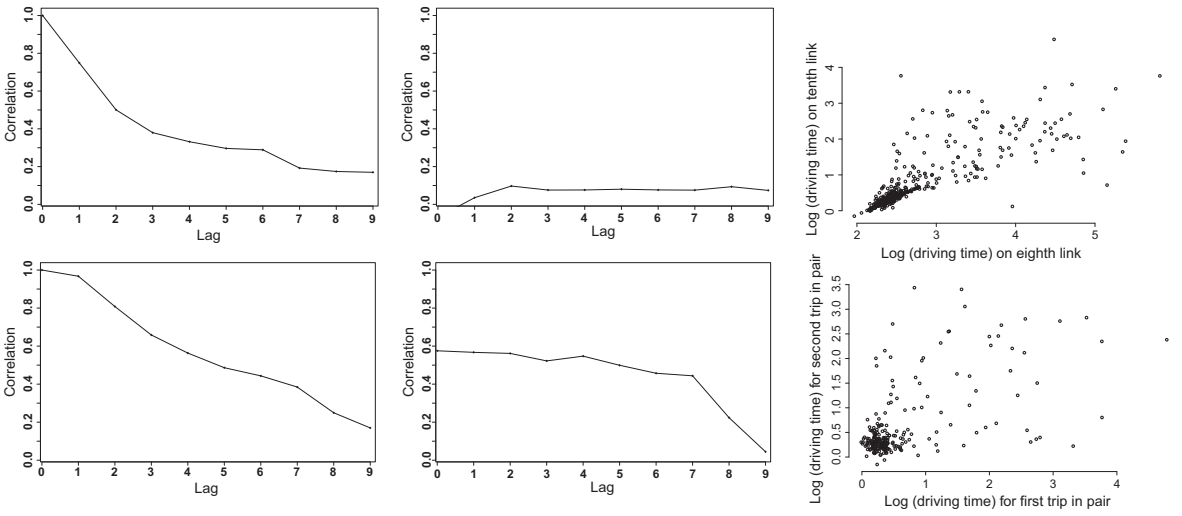

Fig. 5. Association of travel times within and between trips, on a 10-link route. Top left: Correlation of log travel time on the first link with that on each link, within the same trip. Bottom left: Correlation of the last link with each link, within the same trip. Top middle: Correlation of the first link with each link, for pairs of distinct trips in the same 15-min time period. Bottom middle: Correlation of the last link with each link, for pairs of trips in the same time period. Top right: Scatterplot for the eighth vs. tenth links, within the same trip. Bottom right: Scatterplot for pairs of trips on the tenth link in the same time period.

typically are trips that have longer routes, but for the purpose of this figure we focus on the travel time only for the portion corresponding to the route of interest. The predictive densities match the histograms generally well, capturing the heavy right skew and multimodal nature of the travel times, and accurately predicting the amount of variability of the distributions.

Next we report predictive accuracy on the entire test dataset (35,190 trips on routes throughout the network). We report the accuracy of deterministic predictions in Tables 1 and 2, and of interval predictions in Fig. 7. To obtain a deterministic

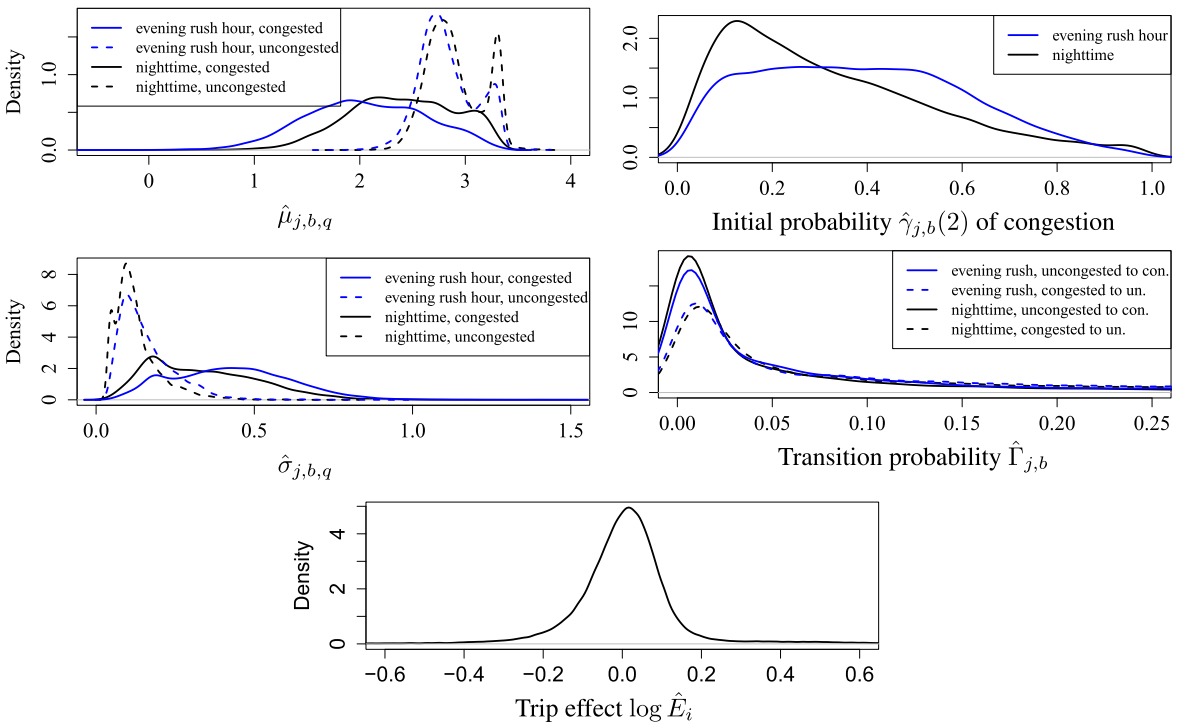

Fig. 6. Parameter and latent variable estimates from TRIP. Top left: the distribution over links j of the estimated mean parameters $\hat{\mu}_{j,b,q}$, for two time bins b and two congestion states q. Middle left: the distribution over links j of the standard deviations $\hat{\sigma}_{j,b,q}$, for various b and q. Top right: the distribution over j of the initial probability $\hat{\gamma}_{j,b}(2)$ of congestion, for various b. Middle right: the distribution over j of the transition probabilities $\hat{\Gamma}_{j,b}$ between congestion states, for various b. Bottom: the distribution of the trip effect $\log \hat{E}_i$, over trips i in the training data.

Table 1
Accuracy of deterministic predictions for the Seattle test data. Since the deterministic prediction captures the center of the predicted distribution, these results are similar across the TRIP variants (which differ according to how they model variability).

	TRIP	TRIP, no trip effect	TRIP, no Markov model	TRIP, no dependence	Clearflow	Linear regression
Geometric mean of |predicted - actual|/actual	10.1%	9.6%	10.0%	9.8%	10.4%	12.8%
Mean absolute error (s)	121.9	119.7	121.3	120.6	124.5	145.6
Bias on log scale	.030	.014	.028	.024	.033	−.005

Table 2
Accuracy of deterministic predictions, broken down according to whether the links in the route are heavily traveled.

	Heavily Traveled			Sparsely Traveled		
	TRIP	Clearflow	Linear regression	TRIP	Clearflow	Linear regression
Geometric mean of |predicted - actual|/actual	9.8%	10.4%	12.7 %	13.7%	11.3%	13.7%
Geometric mean of |predicted - actual|/actual after bias correction	9.4%	9.6%	12.9%	11.0%	10.2%	12.4%
Mean absolute error (s)	122.5	126.9	148.4	114.8	100.6	117.7
Mean absolute error after bias correction (s)	120.6	124.3	149.0	101.3	96.4	110.3
Bias on log scale	.019	.029	−.013	.108	.066	.077

prediction, we use the geometric mean of the travel time distribution. This is more appropriate than the arithmetic mean, due to the heavy right skew of travel time distributions. To obtain an interval prediction with theoretical coverage $100(1-\alpha)$ where $\alpha \in (0,1)$, we take the lower and upper bounds of the interval to be the $100(\alpha/2)$ and $100(1-\alpha/2)$ percentiles of the predicted travel time distribution. An interval with theoretical coverage of $100(1-\alpha)$ means that the interval is intended to include $100(1-\alpha)$ percent of future trips. For example, the 95% interval is given by the 2.5 and 97.5 percentiles of the predicted travel time distribution.

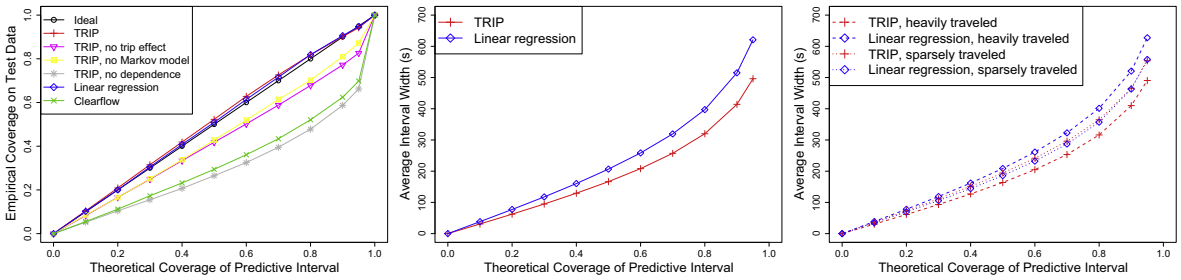

Fig. 7. Accuracy of interval predictions for the Seattle test data. Left: Coverage; Middle: Average interval width, for the methods that have approximately correct coverage; Right: Average interval width, broken down according to whether the links in the route are heavily traveled.

In Tables 1 and 2 we report the geometric mean across trips of the percentage error of deterministic predictions (the absolute difference between the predicted and observed travel time, divided by the observed travel time). We also report the mean absolute error. Third, we report the bias of the deterministic prediction on the log scale: precisely, the mean across trips of the log of the prediction minus the log of the observed travel time. The bias does not measure how far the predictions are from the observations, so a small amount of nonzero bias is not a problem. However, adjusting the deterministic predictions to remove the bias can sometimes improve the accuracy as measured using the percentage error or the mean absolute error. Specifically, in Table 2 we also report those accuracy measures after multiplying the deterministic prediction by the exponent of -1 times the log-scale bias. Such a bias adjustment is also done by Westgate et al. (2016).

To measure the accuracy of interval predictions in Fig. 7, we first report their empirical coverage, meaning the percentage of test trips for which the observed travel time is inside of the predictive interval. If the variability is predicted correctly, the empirical coverage is close to the theoretical coverage. Additionally, we report the average width of the interval. As an example, if two methods both have 95% interval predictions with 95% empirical coverage, the method with the narrower intervals is preferred. This is because both methods are accurately characterizing variability, but one is taking better advantage of the information in the data to give a narrow predicted range (Gneiting et al., 2007).

Tables 1 and 2 and Fig. 7 compare TRIP to several alternatives. First, we compare to three simplified versions of TRIP that drop one or both of the types of dependence across links. One type of dependence is induced by the trip effect E_i, so we consider dropping this term from the model. The other type is caused by the Markov model on $Q_{i,k}$, so we consider replacing this with an independence model, $\Pr(Q_{i,k} = q) = \gamma_{R_{i,k},b_{i,k}}(q)$ for all $k \in \{1, \ldots, \mathcal{Q}\}$.

Second, we compare the results to inferences from an adaptation of Clearflow (Microsoft Research, 2012). Clearflow models the distribution of travel time on each link in the network using Bayesian structure search and inference to integrate evidence from a large number of variables, including traffic flow sensor measurements, speed limit, road classification, road topological distinctions, time of day, day of week, and proximity to schools, shopping malls, city centers, and parks. It was designed for accurate prediction of mean real-time flows on all links of large metropolitan traffic networks. In practice, the Clearflow inferences about flows on individual links are used to guide route planning procedures that generate routes via summing the times of the set of links of a trip between starting point and destination. For this reason Clearflow was not targeted at modeling the distribution of travel time on entire routes. However, we can combine Clearflow with an assumption of independence across links, in order to produce distribution predictions on routes for the purposes of comparison.

Finally, we compare to a regression approach that models the travel time for the entire trip, as done in methods for predicting ambulance travel times (Budge et al., 2010; Westgate et al., 2016). In particular, we use a standard linear regression model where the outcome is the log of the trip travel time, and the predictor variables are: (a) the log of route distance; (b) the time bin $b_{i,1}$ in which the trip begins; and (c) the log of the travel time according to the speed limit. We include an interaction term between (b) and (c), meaning that the linear slope for (c) is allowed to depend on the value of (b). The assumptions of the linear regression model hold approximately; for example, scatterplots of the log travel time and the variables (a) and (c) show an approximately linear relationship.

As seen in Table 1, the predictions differ from the actual values by 9.6–10.4% for all the methods except linear regression, which has error of 12.8%. The same conclusions hold when considering mean absolute error instead of percentage error. The accuracy of TRIP is slightly better than Clearflow overall, despite the fact that Clearflow harnesses a larger number of variables than those considered by TRIP in its current form. The bias is small for all of the methods (the largest being.033, which corresponds to a factor of $\exp\{.033\} = 1.034$ in travel time, i.e. a bias of 3.4%).

In Table 2 we break down the accuracy of deterministic predictions according to whether the links in the route are heavily traveled; this is defined to mean that over half of the links in the route have at least 30 observations in the training data. Out of the 35,190 trips in the test data, all but 3197 are on such heavily traveled routes. The accuracy of TRIP is considerably better than linear regression on heavily traveled routes, and the same or better than linear regression on the sparsely traveled routes. TRIP performs slightly better than Clearflow on heavily traveled routes, whereas on sparsely traveled routes Clearflow is more accurate. We believe that this is due to the fact that Clearflow uses a large number of variables describing the link, which provide relevant information for predicting the speed. The bias is larger for the sparsely traveled routes than

for the heavily traveled ones, so we provide accuracy results both before and after bias correction; the qualitative conclusions are unaffected by this adjustment.

Fig. 7 shows that, out of the methods considered, only TRIP and linear regression have predictive interval coverage that is close to correct. The other methods have dramatically lower coverage, Clearflow's coverage of 95% intervals being 69.8%. Clearflow and the simplest version of TRIP have similar coverage because they both assume independence of travel time across the links of a trip. For the simplified versions of TRIP that incorporate one out of the two kinds of dependence, the coverage is much better than that of the methods that assume independence, but still well below the desired value. Fig. 7 also shows that the interval predictions from TRIP are 19–21% narrower on average than those from linear regression, providing evidence that TRIP is substantially better for interval prediction. Finally, Fig. 7 shows that TRIP's advantage relative to linear regression in interval prediction is greatest on heavily traveled routes; on sparsely traveled routes the performance of the two methods in interval prediction is similar.

The fact that TRIP provides little benefit relative to linear regression on the sparsest parts of the road network (which account for a small proportion of trips in the mobile phone data) is not surprising since linear regression is a simpler model with fewer parameters. This conclusion is consistent with previous results for emergency vehicles (Westgate et al., 2016). Indeed, regression-style approaches are typically used for emergency vehicles, due to the sparsity of data.

5. Conclusions

We have introduced a method (TRIP) for predicting the probability distribution of travel time on arbitrary routes in the road network, at arbitrary times. We evaluated TRIP on a case study using mobile phone data from the Seattle metropolitan region. Based on a complexity analysis and on the running time of the algorithms for this case study, we argue that TRIP is computationally feasible for the continental-scale road networks and high-volume data of commercial mapping services.

TRIP's deterministic predictions are more accurate on heavily traveled routes, although slightly less accurate on sparsely traveled routes, than Microsoft's commercially fielded system (Clearflow). The interval predictions from TRIP are much better. Clearflow's consideration of flows on a segment by segment basis is valuable for use with current route planning procedures, which consider the road speeds on separate segments in building routes. However, such independent handling of inferences about segments can lead to underprediction of route-specific variability. TRIP solves this issue by accurately capturing dependencies in travel time across the links of the trip. Although a linear regression approach yields reasonable accuracy of interval predictions, it gives worse deterministic predictions than TRIP. To our knowledge TRIP is the first method to provide accurate predictions of travel time reliability for complete, large-scale road networks.

Future work includes extending TRIP to incorporate additional variables, including those used in Clearflow learning and inference. For example, this would allow TRIP to take into account real-time information about traffic conditions, as measured using data from sensors installed in highways, or average measured GPS speeds from mobile phones during the current time period. This extension has the potential to provide narrower distribution forecasts and predictive intervals, and even more accurate deterministic estimates.

There is also opportunity to employ active information gathering methods to guide both selective real-time sensing of different portions of a road network and the bulk collection of data to reduce uncertainty about the flows over segments and routes. There has been related prior work on the use of active sensing for reducing uncertainty about the travel time on segments in a demand-weighted manner (Krause et al., 2008). The work considers the probabilistic dependencies across a city-wide traffic network and the value of sensing from different regions for reducing uncertainty across the entire road network. We foresee the use of similar methods in combination with TRIP to guide the optimal collection of data.

Acknowledgements

This work was supported by Microsoft Research.

References

Besag, J., 1986. On the statistical analysis of dirty pictures. J. Roy. Stat. Soc. Ser. B 48, 259–302.
Bilmes, J.A., 1997. A Gentle Tutorial of the EM Algorithm and its Application to Parameter Estimation for Gaussian Mixture and Hidden Markov Models. Technical Report TR-97-021. International Computer Science Institute.
Budge, S., Ingolfsson, A., Zerom, D., 2010. Empirical analysis of ambulance travel times: the case of Calgary emergency medical services. Manage. Sci. 56, 716–723.
Cousineau, D., Helie, S., 2013. Improving maximum likelihood estimation with prior probabilities: a tutorial on maximum a posteriori estimation and an examination of the Weibull distribution. Tutorials Quant. Methods Psychol. 9 (2), 61–71.
Delling, D., Goldberg, A.V., Pajor, T., Werneck, R.F., 2015. Customizable route planning in road networks, *Transp. Sci.* (in press).
Erkut, E., Ingolfsson, A., Erdogan, G., 2007. Ambulance location for maximum survival. Naval Res. Logist. (NRL) 55, 42–58.
Gelman, A., 2006. Prior distributions for variance parameters in hierarchical models (comment on article by Brown and Draper). Bayesian Anal. 1, 515–534.
Gelman, A., Carlin, J.B., Stern, H.S., Dunson, D.B., Vehtari, A., Rubin, D.B., 2013. Bayesian Data Analysis. CRC Press, Boca Raton, FL.
Gneiting, T., Balabdaoui, F., Raftery, A.E., 2007. Probabilistic forecasts, calibration, and sharpness. J. Roy. Stat. Soc. Ser. B 69, 243–268.
Grava, C., Gacsadi, A., Gordan, C., Grava, A.-M., Gavrilut, I., 2007. Applications of the Iterated Conditional Modes algorithm for motion estimation in medical image sequences. In: International Symposium on Signals, Circuits, and Systems. IEEE.
Hofleitner, A., Herring, R., Abbeel, P., Bayen, A., 2012a. Learning the dynamics of arterial traffic from probe data using a dynamic Bayesian network. IEEE Trans. Intell. Transp. Syst. 13, 1679–1693.

Hofleitner, A., Herring, R., Bayen, A., 2012b. Arterial travel time forecast with streaming data: a hybrid approach of flow modeling and machine learning. Transp. Res. Part B 46, 1097–1122.
Horvitz, E., Apacible, J., Sarin, R., Liao, L., 2005. Prediction, expectation, and surprise: methods, designs, and study of a deployed traffic forecasting service. In: Proceedings of the Conference on Uncertainty in Artificial Intelligence. AUAI Press, Arlington, Virginia, pp. 275–283.
Hunter, T., Abbeel, P., Bayen, A.M., 2014. The path inference filter: model-based low-latency map matching of probe vehicle data. IEEE Trans. Intell. Transp. Syst. 15 (2), 507–529.
Hunter, T., Das, T., Zaharia, M., Abbeel, P., Bayen, A., 2013. Large-scale estimation in cyberphysical systems using streaming data: a case study with arterial traffic estimation. IEEE Trans. Autom. Sci. Eng. 10, 884–898.
Hunter, T., Herring, R., Abbeel, P., Bayen, A., 2009. Path and travel time inference from GPS probe vehicle data. In: Neural Information Processing Systems, Vancouver, Canada.
Hunter, T., Hofleitner, A., Reilly, J., Krichene, W., Thai, J., Kouvelas, A., Abbeel, P., Bayen, A., 2013. Arriving on Time: Estimating Travel Time Distributions on Large-scale Road Networks. Technical Report. Available from: <1302.6617>.
James, G., Witten, D., Hastie, T., Tibshirani, R., 2013. An Introduction to Statistical Learning with Applications in R. Springer, New York.
Jenelius, E., 2012. The value of travel time variability with trip chains, flexible scheduling, and correlated travel times. Transp. Res. Part B 46, 762–780.
Jenelius, E., Koutsopoulos, H.N., 2013. Travel time estimation for urban road networks using low frequency probe vehicle data. Transp. Res. Part B 53, 64–81.
Jenelius, E., Koutsopoulos, H.N., 2015. Probe vehicle data sampled by time or space: implications for travel time allocation and estimation. Transp. Res. Part B 71, 120–137.
Krause, A., Horvitz, E., Kansal, A., Zhao, F., 2008. Toward community sensing. In: Proceedings of ISPN 2008, International Conference on Information Processing in Sensor Networks, St. Louis, Missouri.
Lee, S.X., McLachlan, G.J., 2013. EMMIXuskew: an R package for fitting mixtures of multivariate skew *t* distributions via the EM algorithm. J. Stat. Softw. 55 (12), 1–22.
Liu, C., 1997. ML estimation of the multivariate *t* distribution and the EM algorithm. J. Multivariate Anal. 63, 296–312.
Meng, X.-L., Rubin, D.B., 1993. Maximum likelihood via the ECM algorithm: a general framework. Biometrika 80, 267–278.
Microsoft Research, 2012. Predictive Analytics for Traffic: Machine Learning and Intelligence for Sensing, Inferring, and Forecasting Traffic Flows. <http://research.microsoft.com/en-us/projects/clearflow>.
Newson, P., Krumm, J., 2009. Hidden Markov map matching through noise and sparseness. In: 17th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems. Association for Computing Machinery, Seattle, WA, pp. 336–343.
Rahmani, M., Jenelius, E., Koutsopoulos, H.N., 2015. Non-parametric estimation of route travel time distributions from low-frequency floating car data. Transp. Res. Part C 58, 343–362.
Ramezani, M., Geroliminis, N., 2012. On the estimation of arterial route travel time distribution with Markov chains. Transp. Res. Part B 46, 1576–1590.
Russell, S., Norvig, P., 2009. Artificial Intelligence: A Modern Approach. Pearson Education, UK.
Samaranayake, S., Blandin, S., Bayen, A., 2012. A tractable class of algorithms for reliable routing in stochastic networks. Transp. Res. Part C 20, 199–217.
Texas Transportation Institute, 2015. Travel Time Reliability: Making it There on Time, Every Time. Technical Report. U.S. Department of Transportation. <http://www.ops.fhwa.dot.gov/publications/tt_reliability/index.htm>.
Westgate, B.S., Woodard, D.B., Matteson, D.S., Henderson, S.G., 2013. Travel time estimation for ambulances using Bayesian data augmentation. Ann. Appl. Stat. 7, 1139–1161.
Westgate, B.S., Woodard, D.B., Matteson, D.S., Henderson, S.G., 2016. Large-network travel time distribution estimation for ambulances. Eur. J. Oper. Res. 252, 322–333.
Woodard, D.B., Rosenthal, J.S., 2013. Convergence rate of Markov chain methods for genomic motif discovery. Ann. Stat. 41, 91–124.
Work, D.B., Blandin, S., Tossavainen, O.-P., Piccoli, B., Bayen, A.M., 2010. A traffic model for velocity data assimilation. Appl. Math. Res. eXpress 2010 (1), 1–35.