Dynamic Departure Time Estimation

Sudatta Mohanty*
University of California, Berkeley

Alexey Pozdnukhov †
University of California, Berkeley

Abstract

Traffic volumes and congestion patterns on a network are very sensitive to the distribution of departure times. In activity-based travel demand modeling, the assignment of departure times of agents from various activity locations usually relies on the travel surveys and accessibility heuristics and is very imprecise. It leads to high variability of the modeled traffic volumes, whether in agent-based simulation or numerical DTA solvers, and translates into an inability to reliably identify the source of errors in travel models. This work mitigates this uncertainty by integrating several emerging sources of travel data into a novel numerical optimization framework for departure time inference. We extend an Entropy Maximization approach (Janson et al., 1992) [1] that assumes a fixed link use probability matrix and solves for departure time distributions from each zone using observed link volume counts. This method is extended in two respects. First, we use a large sample of real-time mobility traces from a cellular communication network logs to obtain lower bound constraints for observed departures. Second, we assume that departure time choices are increasingly governed by online Advanced Traveler Information Systems and incorporate travel times available from online services into the design of the link use matrix. Since this matrix relies in turn on the routing, we apply the convex optimization framework developed by Wu et al. (2015) [2] to infer route flows as a sub-routine in travel time inference. We then introduce the optimization problem formulations to compute departure time and solve the problem using a primal-dual interior-point algorithm with a filter line-search method proposed by Wachter et al. (2006) [3]. Finally, we dynamically update the estimates through Batch Gradient Descent algorithm which allows the framework to be used in Advanced Traffic Management, in a scenario when the above mentioned data is available as a live stream. The framework is tested and validated on a detailed agent-based micro-simulation calibrated from the cell phone data for a simplified freeway network spanning over nine counties of the San Francisco Bay Area. Finally, the scalability of the algorithm is tested on a full-scale network simulation of the same area.

Keywords: entropy maximization, travel times estimation, cellular data, traffic assignment

*Corresponding Author
Tel: +1-510-717-5461
Email: sudatta.mohanty@berkeley.edu
Address: 116, McLaughlin Hall, University of California, Berkeley CA 94720
†Tel: +1-510-642-1008
Email: alexeip@berkeley.edu
Address: 115, McLaughlin Hall, University of California, Berkeley CA 94720
1 Introduction

1.1 Background and Motivation

It has been well established that traffic volumes and congestion patterns on a network are highly dependent on departure times and route choice decisions of agents (Henrickson et al., 1981, 1984; Arnott et al., 1990; Carey et al., 1993; Bhat, 1998; Huang et al., 2002; Ettema et al., 2003; Gaver (1968); De Palma (1983)). Departure time estimation has been an area of considerable research in the past. Several models have been built to model departure time choices and their effect on traffic flow patterns. Gaver (1968) assumed that agents update their estimate of distribution of delays from previous experiences. One of the first models incorporating the effect of departure time decisions on congestion patterns was developed by Vickrey (1969). The Vickrey Model was a point-queue model with a single Origin-Destination (O-D) pair assumption and utility maximization decision protocol. It proved the existence of an equilibrium condition wherein no agent gains by changing its departure time. This formulation was subsequently extended by Smith (1983, 1984) and Arnott et al. (1990, 1993). De Palma (1983) first proposed a deterministic queue and a random utility function to determine departure time choice. This model was further extended to a number of bottlenecks in parallel with route choice by Ben Akiva et al. (1984). Closed-form idealized analytic solutions for departure time choice in simple idealized networks were proposed by Mahmassani et al. (1986) and Ben-Akiva et al. (1987). Although these models explain choice scenario for the agent and the effect on bottlenecks, they fail to generalize the algorithm for a large network. When extended to a larger network, these models usually implement restricted route choice to allow for an analytical solution. Another drawback of these models is the static nature of the user equilibrium assumption. For such models to be useful for dynamic control strategies, they should be sensitive to daily fluctuations in demand. To capture the effect of situational variations like incidents or gridlocks, there is a need to develop a data-driven algorithm which can respond to such variations. In the past few decades, several techniques have been developed to estimate the state of traffic in a network using real-time data. Van Zuylen et al. (1980) presented one of the first formulations to estimate Origin-Destination (O-D) matrices from traffic counts through Entropy Maximization algorithm. To allow for time-varying travel demands, Dynamic Traffic Assignment (DTA) algorithm (Merchant et al., 1978) has been implemented in the past. Janson (1991a) was the first to attempt a complete formulation of DTA user equilibrium for multiple origins and destinations using mathematical programming. Variational-inequality formulations of DTA allowed for a unified mechanism to address equilibrium and optimization formulations. Due to unavailability of analytical representations of traffic flow that adequately replicate traffic theoretic relationships or yield well-behaved mathematical formulation (Peeta et al., 2001), simulation-based DTA techniques were later developed which were found to have greater acceptability for real-world deployment (Mahmassani, 1998b; Ben-Akiva, 1998). The application of DTA to estimate dynamic departure times on a network has been demonstrated by Janson et al. (1992) and Ben-Akiva et al. (2001). However, the major drawbacks in the use of DTA algorithm are the model complexity, computational burden, and uncertainties regarding both the properties and validity of solutions (Boyce, 1989). Therefore, this study implements a data-driven convex optimization technique for route flow estimation developed by Wu et al. (2015) as a sub-routine in travel time inference. This approach is model agnostic and thus compatible with user equilibrium, system-optimum, Stackelberg concepts, and other models.
1.2 Applications of Dynamic Departure Time Estimation

An accurate and dynamic estimation technique for departure times not only helps in understanding and modelling traffic patterns in a network but also allows devising of dynamic control strategies to mitigate congestion-related costs such as time delay, greenhouse gas emission, infrastructure cost, etc. Data requirements and some of the applications enabled by the framework proposed by this paper are discussed below.

1.2.1 Data Requirements

The range of applications we outline below, and the advances in the methodology we present in this paper are enabled by the increasing availability of mobility data. Data requirements for the algorithms introduced in this paper are as follows (also refer to Figure 1).

- Traffic volume counts on a subset of links.
  It is assumed that real-time counts \( v_t^k \) at small time intervals \( t \) (in the order of few minutes) are available on a subset of links \( k \) in the traffic network. The accuracy of the algorithm depends not only on the coverage of the count data, but also on the proportion of trips which traverse through links where counts are available.

- Partly observed aggregated location data.
  It is assumed that data is available regarding the number of individuals \( N^d_r \) observed in each zone \( r \) at small time intervals \( d \) (in the order of few minutes). It is further assumed that this data is only available for a proportion of the total population. The effect of data coverage on the accuracy of the algorithm is described in section 4.2.

- Travel times on the network.
  It is assumed that travel time data for each trip is recorded at the link level, and is available for dynamic departure time estimation. This data are used as part of a sub-routine for calculating the probability \( p_{t,d}^{k,r} \) of a trip departing from zone \( r \) in time interval \( d \) and arriving at link \( k \) at time interval \( t \). The effect of noise in calculation of \( p_{t,d}^{k,r} \) on the algorithm accuracy is discussed in section 4.4.

- Partly observed waypoints data.
  It is assumed that individual-level data in the form of cellpaths, defined as a time-stamped sequence of discrete zones within which a user can be located during the trip, is available. It is further assumed that this data is available for a proportion of the total population. This information is used as part of a subroutine for route flow estimation through the framework proposed by Wu et al (2015) [2]. It is further assumed that the total number of individuals in each zone of the study area is known.

The data listed above are becoming available in industry, and the practical obstacles in using them are in the immaturity of business models for data sharing rather than sensing or data processing technology. This paper removes yet another, algorithmic, obstacle in the way of building and deploying Advanced Traffic Management Systems (ATMS). We foresee it enabling advances in the following application areas.
1.2.2 Agent-Based Traffic Simulations

Traditional four-step travel demand model is not able to capture the dependencies within various activity chains on an individual (Bhat et al., 1999 [27]). Thus, an activity-based approach to travel demand modelling is gaining popularity. Activity-based demand generation produces a standard origin-destination (O-D) matrix, which can then be fed into the network assignment model. However, inaccuracies in activity departure time estimation may cause incorrect coupling between activities. An example of incorrect coupling was provided by Balmer et al. (2005) [28] wherein agents may be assigned departure time from a destination even before their actual arrival. To prevent such scenarios, it is essential to follow an agent-based approach (i.e., tracking each individual’s activity chain during simulation). To develop accurate and realistic activity chains for each individual, it is important to have a fine-grained departure time choice information. The proposed framework provides such estimates. The incremental optimization algorithm also enable modifying departure times in activity sequences dynamically in real-time to achieve more accurate within-day scenario simulations.

1.2.3 Dynamic Control Strategies

Transportation network control strategies to mitigate congestion may include staggered work hours (Henderson, 1981 [29]), congestion/cordon pricing, traffic signal coordination, ramp metering and license-plate lottery turn taking system (Yang et al, 2014 [30]). These strategies are very sensitive to demand data. Due to queue spillover effects, small perturbations in O-D and departure times can lead to a large aggregated impact on a network (Daganzo, 1998 [31]). Therefore, network models intended for traffic management during daily time-varying travel demand cannot be restricted to the steady-state assumptions of most models (Janson, 1992 [1]). A robust and dynamic departure time prediction model can thus go a long way in developing ATMS.

2 Methodology

2.1 Traditional Entropy Maximization Problem Formulation

The traditional entropy maximization problem proposed by Janson et al. (1992) [1] tries to estimate the number of trips departing from each zone $r$ in each time interval $d$. Zones could be defined as the area units where the number of departures are to be established. Time intervals can be defined as the smallest time unit within which departures could be estimated. Typically, this is determined through the frequency of the availability of data. The objective function utilizes Sterling’s approximation to derive the maximum entropy for all trip departures from all zones $Z$ across all time intervals $D$ (Wilson, 1970 [32]). The formulation stands as follows:

$$ \min_{q^d_r} \sum_{r \in Z} \sum_{d \in D} \{q^d_r \log(q^d_r) - q^d_r\} $$

s.t. $v^k_t = \sum_{r \in Z} \sum_{d \in D} \{q^d_r p^k_t r d\}, \forall k \in L, t \in D$ (1)

$q^d_r \geq 0, \forall r \in Z, d \in D$
Figure 1: Overview of the problem: At the zonal level, we observe at least $N^d_r$ individuals present in zone $r$ at the end of each time interval $d$. This is a lower bound of the total number of individuals present in zone $r$ at time $d$. At the link level, we observe a total of $v^t_k$ trips crossing link $k$ during each time interval $t$. This is an aggregation of trips with possibly different O-Ds and different departure times. It is also assumed that perfect knowledge is available regarding link travel times across the whole network at any instance. Our goal is to estimate the total number of trips departing zone $r$ during time interval $d$, $q^d_r$.

where:

$q^d_r$ is a number of trips departing from zone $r$ to all destinations in time interval $d$,

$v^t_k$ is the observed volume on link $k$ during time interval $t$,

$p^{t,d}_{k,r}$ is a probability that a trip departing zone $r$ in time interval $d$ uses link $k$ in time interval $t$.

The inputs for the optimization problem formulation are a vector of volume counts data for each link $k$ in each time interval $t$ and the link-use probability matrix. The link-use probability matrix provides information about the conditional probability for using a particular link $k$ in a particular time interval $t$ given the origin of a trip from a zone $r$ in a particular time interval $d$. This information is required across all links $L$ in all time intervals $D$. The traditional procedure employs Dynamic Traffic Assignment (DTA) to extract this information and only includes non-negativity constraint on the zone trip departures for all time intervals.

Two major drawbacks are observed regarding this formulation. Firstly, the conventional DTA algorithm might not be convergent (Janson et al., 1992 [1]) and none of the methods provides a universal solution for a general network (Peeta et al., 2001 [22]). There is a need for a more robust procedure to determine the link-use probability matrix. Secondly, the formulation assumes that the route choice decision is made at the time of departure and there is no change or update to that decision. While this might be true for short or frequently travelled trips, it may not hold true for any other kind of...
trip. Therefore, there is a need to update the route choice dynamically as new data become available. A third possible improvement to this formulation is to get a better lower bound on $q^d_r$ using data on the presence of travellers within the travel zones. We assume these data to be available in real-time and refer to it below as mobility trace data.

2.2 Dynamic Departure Time Estimation Formulation

2.2.1 Partly Observed Departures Constraints

Here, the standard entropy maximization problem is modified to include real-time information and overcome drawbacks of the traditional formulation. First, mobility trace data from each zone is utilized to generate partly observed departures from each zone which provide lower bounds on $q^d_r$. The number of trips departing from a zone $r$ in a particular time interval $d$ is greater than or equal to the difference of the number of observed individuals in zone $r$ at time interval $d$ and the number of observed individuals at time interval $(d-1)$ (2).

$$q^d_r \geq \max\{N^d_r - N^{d-1}_r, 0\}, \quad (2)$$

where:

$q^d_r$ is a number of trips departing from zone $r$ to all destinations in time interval $d$,

$N^d_r$ is number of observed individuals at zone $r$ during time interval $d$ from mobility trace data.

2.2.2 Link Counts and Partly Observed Waypoints Constraints

Next, data-driven convex optimization algorithm for route flow estimation using partly observed waypoints proposed by Wu et al. (2015) is applied to get probability for a trip originating from zone $r$ to use route $r^*$ (3). It assumes the availability of individual-level data in a form of cellpaths, defined as a time-stamped sequence of discrete zones within which a user can be located during a trip. It is a common format of mobility data available from cellular network carriers and IT service providers. The algorithm tries to minimize the squared error between observed link counts in each time bin $v_k^d$, and predicted link counts using Origin-Destination (O-D) flows $q^d_{r^*}$ and the link-route incidence matrix $A_{kr^*}^d$. That is, at any $d \in D$:

$$\min_{q^d_{r^*}} \sum_{k \in L} \sum_{r^* \in Z^*} A_{kr^*}^d q^d_{r^*} - v_k^d)^2$$

s.t. \[ \sum_{c \in C} \sum_{r \in Z^*} U_{cr}^d q^d_{r^*} = f^d \]

$q^d_{r^*} \geq 0, \forall r \in Z, d \in D$

where:

$$A_{kr^*}^d = \begin{cases} 1 & \text{if link } k \text{ lies along route } r^* \text{ at time } d, \\ 0 & \text{otherwise}, \end{cases}$$

$q^d_{r^*}$ is a number of trips departing along OD $r^*$ in time interval $d$,
\( v^d_k \) is an observed volume on link \( k \) during time interval \( d \),

\[
U^d_{cr} = \begin{cases} 
1 & \text{if user crosses cellpath } c \text{ while travelling along route } r^* \text{ at time } d, \\
0 & \text{otherwise}, 
\end{cases}
\]

\( f^d_c \) is the observed cellpath volume on cellpath \( c \) during time interval \( d \).

Now, the traditional DTA solution to the link-use probability matrix can be modified as shown in equations (4) and (5). First, the route choice probabilities at any time interval \( d \), \( p^{r^*,d}_{r^*} \), are calculated as a ratio of predicted O-D trips \( q^d_{r^*} \) and all trips originating from that zone in the time interval (4).

\[
p^{r^*,d}_{r^*} = \frac{q^d_{r^*}}{\sum_{r^* \text{ with origin } r} q^d_{r^*}} 
\] (4)

where:

- \( p^{r^*,d}_{r^*} \) is a probability that a trip with origin \( r \) at time \( d \) is along OD \( r^* \),
- \( q^d_{r^*} \) is a number of trips departing along route \( r^* \) in time interval \( d \), derived from the solution to (3).

Next, this information is combined with travel time data to obtain a robust and dynamic estimate for the link-use probability matrix for a trip:

\[
p^{t,d}_{k,r} = \sum_{r^* \in Z^*} p^{r^*,d}_{r^*} p^{t,d}_{k,r^*}, 
\] (5)

where:

- \( p^{t,d}_{k,r} \) is a probability that a trip departing zone \( r \) in time interval \( d \) uses link \( k \) in time interval \( t \),
- \( p^{r^*,d}_{r^*} \) is a probability that a trip with origin \( r \) at time \( d \) is along route \( r^* \),
- \( p^{t,d}_{k,r^*} \) is a probability that a trip along route \( r^* \) in time interval \( d \) uses link \( k \) in time interval \( t \),
- \( A^d_{k,r^*} \) is a deterministic binary indicator function that equals 1 if link \( k \) lies along route \( r^* \) at time \( d \), and 0 otherwise,
- \( A^{t,d}_{k,r^*} \) is a deterministic binary indicator function that equals 1 if link \( k \) is traversed during time interval \( t \) for trip along route \( r^* \) at time \( d \), and 0 otherwise.

This step assumes availability of real-time travel time inference data from advanced travel information systems operators\(^1\). Since the data sources carry single valued travel times with unknown error term, the proposed formulation treats them as deterministic. We provide an empirical evaluation of the sensitivity of the algorithms to multiple sources of uncertainties in the experimental section below. Note that the link-route incidence matrix \( A^d_{k,r^*} \) is modelled as varying over time \( d \). This allows us the flexibility to accommodate sudden changes in the network structure due to road closure or changes in the route choice behavior of individuals during a trip.

\(^1\)Google Directions API, Nokia HERE API, INRIX Drive Time, etc.
2.2.3 Combined Optimization Framework

The combined optimization framework including partly observed departures constraints, partly observed waypoints and observed link count constraints is described in equation (6). Interior point optimization or barrier method is proposed for this problem. The dual variable from the equality constraint in (1) is moved to the cost function as a lagrange multiplier scale parameter $\beta^t_k$ and the lower bound constraint is added as a logarithmic barrier function (6). The logarithmic barrier function ensures that the penalty for deviating from the lower bound is not too high in cases where there is insufficient data coverage in a particular zone. Since each summation term in the optimization problem is convex in nature, the combined optimization problem is also convex.

$$\min_{q^d_r} \sum_{r \in Z} \sum_{d \in D} \{q^d_r \log(q^d_r) - q^d_r\} + \sum_{k \in L} \sum_{t \in D} \left\{ \beta^t_k (v^t_k - \sum_{r \in Z} \sum_{d \in D} q^d_t p^t_{k,r}) \right\} + \sum_{r \in Z} \sum_{d \in D} \phi(q^d_r)$$

s.t. \(q^d_r \geq 0, \forall \ r \in Z, d \in D\)

where:

- $q^d_r$ is a number of trips departing from zone $r$ to all destinations in time interval $d$,
- $\beta^t_k$ is a relative cost parameter for link constraint for link $k$ at time interval $t$ (depends on the amount and accuracy of link count information available as well as the accuracy of the link-use probability matrix),
- $v^t_k$ is an observed volume on link $k$ during time interval $t$,
- $p^t_{k,r}$ is a probability that a trip departing zone $r$ in time interval $d$ uses link $k$ in time interval $t$ (calculated from (5)),
- $\phi(q^d_r) = -\log(q^d_r - \max(N^d_r, -N^{d-1}_r, 0))$, where $N^d_r$ is a number of individuals at zone $r$ during time interval $d$ from mobility trace data.

The proposed algorithm for solving the optimization problem described in equation (6) is a primal-dual interior-point filter line-search algorithm by Wachter et al. (2006) [3].

2.2.4 Incremental Estimation and Prediction

With near real-time data available at the time of estimation, it becomes possible to predict the number of departures from a given zone more accurately. Therefore, the final step in the departure time estimation framework is to dynamically update the optimal solution with newly available incoming data. There are two proposed algorithms for incremental estimation:

- Using the primal-dual interior-point filter line-search algorithm [3] mentioned in the previous section.
Using Batch Gradient Descent with the gradient $\nabla$ for the objective function of the optimization framework (6) provided as:

$$\nabla = q^d_r - \sum_{r \in Z} \sum_{d \in D} p^t_{k,r} - 1/(q^d_r - N^d_r),$$

(7)

where:

$q^d_r(k)$ is an estimated $q^d_r$ for iteration $k$,

$p^t_{k,r}$ is a probability that a trip departing zone $r$ in time interval $d$ uses link $k$ in time interval $t$,

$N^d_r$ is a number of observed individuals at zone $r$ during time interval $d$ from mobility trace data.

Batch gradient descent is preferred over the much faster Stochastic Gradient Descent (SGD) algorithm because it ensures convergence for a relatively small training data size. The initial feasible solution for the gradient descent algorithm can chosen as $\max(N^d_r - N^{d-1}_r, 0) + 1$. The step size may be chosen empirically or derived from line search algorithm. The convergence criteria proposed is based on the $L_2$ norm of matrix $q^d_r$ as shown in equation (8).

$$\sqrt{\sum_{r \in Z} \sum_{d \in D} (q^d_r(k) - q^d_r(k-1))^2} < \epsilon$$

(8)

where:

$q^d_r$ is a number of trips departing from zone $r$ to all destinations in time interval $d$,

$\epsilon$ is an empirically chosen precision parameter for convergence of batch gradient descent.

For the update framework, batch gradient descent is also preferred over the primal-dual interior-point filter line-search algorithm. This is because it is much faster since the previous solution is chosen as a starting point for the next optimization problem.

3 Experimental Setting

3.1 Test Scenario

The optimization framework for dynamic departure time estimation was tested and validated on an agent-based micro-simulation calibrated from the cell phone data spanning over nine counties of the San Francisco Bay Area (Yin et al, 2017 [33]). We present the results and share the data and software used to reproduce the findings adapted for a simplified freeway network, and then report on the implementation of the full-scale scenario and share a software implementation for the latter. The network was extracted from well known Open Street Map (OSM) of the test area using open source Java application OSMOSIS\(^2\). The freeways and highways were extracted by querying the OSM data

\(^2\)http://wiki.openstreetmap.org/wiki/Osmosis
and filtering over the road properties, including the road types, the number of lanes and speed limits. Single links were manually constructed along unidirectional paths to further simplify the network. The resulting network comprised of 39 nodes and 54 links. The Bay Area region has been classified into 1454 Traffic Analysis Zones (TAZs) by the regional transportation planning commission. However, for the purpose of this example, the TAZs were merged and grouped depending on the closest link in the simplified network. Hence, the analysis scenario included 54 aggregated TAZs (Figure 2). We preferred using this procedure in order to apply demand to the simplified network to an existing division of the area into 34 super-districts. The typical working day activity-based travel demand model for the area was calibrated from the cellular data and validated in [33]. The daily traffic flow scenario was produced using a micro-simulation software MATSim. MATSim is a well-known open-source activity-based traffic simulation software which uses a set of individual agent activity chains to output a set of executed trajectories that each agent performs [34]. MATSim searches for a user equilibrium in terms of utility functions defined for the agents, that includes utilities of performing the prescribed set of activities and dis-utilities related to the experienced travel time.

![Figure 2: Simplified freeway network and aggregated TAZs representing the San Francisco Bay Area for testing and validating the Dynamic Departure Time Estimation Framework.](image)

To test the framework, the estimated departure times must be compared to the actual departure times in the simulation. The actual departure times during the simulation are extracted from the Events File, which is generated as a standard MATSim simulation output, using an Agent Departure EventHandler [35]. To formulate the optimization framework described in equation (6), the two required pieces of information are link counts in small time bins $d$ (5 minutes for this scenario) and partly observed departures from each zone. The link count information from the simulation is determined from the Events File using a Link Leave EventHandler [35]. The partly observed departure counts depend on two main factors. Firstly, it depends on the percentage of population for which mobility trace data, for example cell-phone Call Detail Records (CDRs), location-based social media records, or GPS probe data, are available. Secondly, it depends on the number of arrivals occurring in the
particular zone. Therefore the number of observed departures from each zone in the corresponding
time bins are simulated as random numbers between zero and $k$ times the actual number of departures,
where $k$ represents the percentage of population for which mobility trace data is available.
The optimization step is performed using a MATLAB extension for optimization software IPOPT
(Interior-Point Optimizer) which implements a primal-dual interior-point algorithm with a line-search
method proposed by Wachter et al. (2006) [3]. IPOPT is an open-source software package for large-scale
non-linear optimization developed as part of COIN-OR (The Computational Infrastructure for
Operations Research) initiative. To the knowledge of the authors, this is the fastest open-source package
for large-scale non-linear optimization. For computational ease, it was assumed that all trips made
were within 1 hour in duration. This assumption was justified since even after considering peak-hour
velocities on all links, $\sim 90\%$ of all Origin-Destination (O-D) combinations could be covered within 1
hour.
The incremental update step in the framework was performed using both primal-dual interior-point
filter line-search algorithm [3] and Batch Gradient Descent. The step size for gradient descent algo-
rithm is empirically chosen as a constant value of 0.02. The precision parameter in equation (7) $\epsilon$
is empirically chosen as 0.01. The interior-point algorithm produced more accurate result (RMSLE value
of 0.479 as compared to 0.683 of the gradient descent), but the gradient descent algorithm converges
around 10 times faster (10 seconds on average as compared to 2 minutes on average on a commodity
desktop computer). An open source implementation of the algorithm is made available\(^3\).

### 3.2 Calibration of $\beta_k^t$

The proposed dynamic departure time estimation procedure requires the calibration of the relative
cost parameter $\beta_k^t$ in equation (6). This parameter represents the amount and accuracy of link-count
information available for the network. To perform calibration of the parameter, the overall root mean
squared logarithmic error (RMSLE) is minimized for varying values of $\beta_k^t$ for 5-minute departure count
predictions over one hour.

$$
RMSLE = \frac{1}{|Z||D|} \sum_{r \in Z} \sum_{d \in D} \left\{ \log (q_r^d + 1) - \log (a_r^d + 1) \right\}^2,
$$

where:

$q_r^d$ is an estimated number of trips departing from zone $r$ to all destinations in time interval $d$,

$a_r^d$ is an actual number of trips departing from zone $r$ to all destinations in time interval $d$.

The plot for RMSLE with several values of $\log \beta_k^t$ is shown for 5-minute departure counts in a 1
hour period between 5AM-6AM (shown in Figure 3). This plot was reproduced for both morning
and evening peaks during the day with similar trends. This clearly shows that the overall error is
minimized when $\beta_k^t = 1.0$. This value of $\beta_k^t$ was chosen for all further analysis presented below.

\(^3\)https://github.com/ucb-smartcities/Tutorials-General-Info/tree/master/Network-Analysis-Tutorials
Figure 3: Calibration of $\beta_k$. The Root Mean Squared Logarithmic Error (RMSLE) vs $\log(\beta_k)$ for 5AM-6AM time interval.

4 Analysis of the Results

4.1 Departure Time Estimation Results

The comparison between the actual and predicted number of departures across all zones in the study area is performed. The histogram and cumulative plots for the number of departures in 5-minute intervals from all zones in the test scenario across 24 hours are shown in figures (4) and (5) respectively for three cases:

- Actual number of departures,
- Estimated number of departures with no partly observed departures information,
- Estimated number of departures with partly observed departures constraints derived from mobility trace data.

Figures 4 and 5 give preliminary indication that the cumulative number of departures are predicted much better with the addition of partly observed departure constraints using mobility traces.

The absolute error in the cumulative number of departures across all zones is displayed in Figure 6 for two cases:

- With no partly observed departures information,
- With partly observed departures constraints derived from mobility trace data.

Figure 6 shows that addition of partly observed departures constraints significantly reduces errors in departure time estimation.

4.2 The Effect of Partly Observed Departures Constraints

Partly observed departures through mobility trace data provide lower bound constraints on the values of $q_r$. To test the effect of partly observed departures constraints on the optimal solution, the mobility
**Figure 4:** Plot showing histogram of number of departures in 5-minute intervals across all zones with time of day for 1) Actual number of departures, 2) Estimated number of departures with no partly observed departures information and 3) Estimated number of departures with partly observed departures constraints derived from mobility trace data.

**Figure 5:** Plot showing cumulative number of departures in 5-minute intervals across all zones with time of day for: 1) actual number of departures, 2) estimated number of departures with no partly observed departures information, and 3) estimated number of departures with partly observed departures constraints derived from mobility trace information.
trace data coverage was varied from 0% to 100% in intervals of 10%. Then, the number of observed departures for each zone \( r \) and time bin \( d \) was randomly selected between zero and the number of actual departures within coverage. Figure 7 shows the plot of RMSLE values for varying cell-phone data coverage in the population using optimal \( \beta_t^k \) of 1.0. Overall, the optimal solutions obtained without any constraints differs by 3.8 times of the actual one. This error reduces drastically to 1.8 times the actual solution given the 100% mobility trace data coverage. This shows that partly observed departures provide a much tighter bound on the estimated value of \( q_d^t \).

### 4.3 Effect of incremental estimation

An important analysis in this study is to see the evolution of the estimation error in the number of departures with more incoming streaming data from various sources. As more link-count and mobility trace information is attained with time, more accurate predictions can be made about trips departing up to 1 hour prior to receiving the information. This allows constant update of the \( q_d^t \) values over twelve 5-minute bins (Figure 8).

As hypothesized, the plots show that there is a general negative trend in the RMSLE values as one obtains more link-count and mobility trace information over time. However, in some zones (like Zone 2 and Zone 15 in Figure 8), there might be temporary fluctuation in the error term as the values are converging. This is because of relatively small volumes of flow in these zones during the 5-minute bin analyzed where a small fluctuation may greatly affect the RMSLE value.
Figure 7: Plot of Root Mean Squared Logarithmic Error (RMSLE) vs Percentage of Cell Phone Coverage for number of departures between 5-6 AM.

Figure 8: Plot of evolution of Root Mean Squared Logarithmic Error (RMSLE) in the number of departures between 5:55-6:00 AM with incoming information becoming available at 5 minute intervals from 6:00-6:55 AM for 1) Zone 28, 2) Zone 2, 3) Zone 15 and 4) All zones.
4.4 Effect of uncertainty in link-use probability matrix

An important assumption in the dynamic departure time estimation framework is the availability of sufficient link-count sensor data and link travel time data to accurately estimate the link-use probability matrix. However, in reality such data might either not be available or its quality may be low. In the absence of such data, the estimate for the values of $p_{td}^{kr}$ would contain error, which might propagate to the estimate of $q_{td}^r$. Hence, it is critical to discuss the effect of uncertainty in $p_{td}^{kr}$ on solution for $q_{td}^r$. As per equation (5), two kinds of error are expected in the calculations of $p_{td}^{kr}$:

- Error in probability that a trip with origin $r$ is along route $r^*$ ($p_{r^*}$),
- Error in probability that a trip along route $r^*$ in time interval $d$ uses link $k$ in time interval $t$ ($p_{k,r^*td}^t$)

4.4.1 Error in $p_{r^*}$

As shown in equation (4), $p_{r^*}$ depends on the values of $q_{td}^r$ estimated from equation (3). As shown by Wu et al. (2015) [2], the error in estimation varies with the number of cell paths, coverage of link count information and number of routes considered between each O-D. For the test study (Los Angeles highway network), high level of accuracy (>95%) was achieved with sufficient number of cell paths (80-120) and reasonable link data coverage (40-70%). The authors claim that with a sufficient selection of cells, route flow estimation may be possible without any other kinds of sensor data. In case the data is noisy, it was shown that curating 20-50 routes (per OD) was sufficient for achieving a low (<15%) route flow error. This proves that both absence of data and presence of noise in data have reasonably low effect if the degrees of freedom are kept fairly low and if there is sufficient fine grained data from other sources like cell paths.

To test the effect of noise on the predicted departures, two scenarios were introduced - a normally distributed error term, with mean 0 and standard deviation 0.1 and 1 respectively, was added to each $p_{r^*}$ term. The comparison of root mean squared logarithmic error (RMSLE) values, both with and without partially observed departure constraints, is shown in table 1.

<table>
<thead>
<tr>
<th>Partially observed departures</th>
<th>Standard error</th>
<th>RMSLE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Yes</td>
<td>0</td>
<td>0.479</td>
</tr>
<tr>
<td>Yes</td>
<td>0.1</td>
<td>0.495</td>
</tr>
<tr>
<td>Yes</td>
<td>1</td>
<td>0.496</td>
</tr>
<tr>
<td>No</td>
<td>0</td>
<td>1.08</td>
</tr>
<tr>
<td>No</td>
<td>0.1</td>
<td>1.36</td>
</tr>
<tr>
<td>No</td>
<td>1</td>
<td>1.50</td>
</tr>
</tbody>
</table>

Table 1: Table comparing effect of error in $p_{r^*}$ on Root Mean Square Logarithmic Error (RMSLE) with and without partially observed departures constraints.

As shown by Table 1, there is very low increase in RMSLE values between scenarios with standard error 0.1 and 1 when partly observed departures constraints are added. This further corroborates the fact that availability of fine grained data from other sources allows to keep effect of noise in $p_{r^*}$ values on predicted departures reasonably low.
4.4.2 Error in $p_{k,r_s}^{t,d}$

The values of $p_{k,r_s}^{t,d}$ are derived using link-route incidence indicator and real-time traffic-based link travel time information. The link-route incidence matrix depends on the number of routes considered between each O-D. It must be ensured that the routes considered cover the maximum possible number of trajectories between each O-D. However, due to computational limitations only the shortest path between each O-D was considered for this study. This assumption is valid for the our test network since it contains only major freeways. However, for a more detailed study network, it is important to consider more routes between each O-D. Simulation-based techniques can be employed to determine routes which maximum possible number of trajectories between each O-D [2]. The second piece of information required for calculating $p_{k,r_s}^{t,d}$ is real-time traffic-based link travel time information. In the absence of current link travel time data, typical link travel times for each day may be used for calculation. However, this would make the departure time estimation framework less sensitive to daily fluctuations in traffic.

Once again, to test the effect of noise on the predicted departures, two scenarios were introduced - a normally distributed error term, with mean 0 and standard deviation 0.1 and 1 respectively, was added to each $p_{k,r_s}^{t,d}$ term. The comparison of root mean squared logarithmic error (RMSE) values, both with and without partially observed departure constraints, is shown in table 2.

<table>
<thead>
<tr>
<th>Partially observed departures</th>
<th>Standard error</th>
<th>RMSLE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Yes</td>
<td>0</td>
<td>0.479</td>
</tr>
<tr>
<td>Yes</td>
<td>0.1</td>
<td>0.483</td>
</tr>
<tr>
<td>Yes</td>
<td>1</td>
<td>0.493</td>
</tr>
<tr>
<td>No</td>
<td>0</td>
<td>1.08</td>
</tr>
<tr>
<td>No</td>
<td>0.1</td>
<td>1.13</td>
</tr>
<tr>
<td>No</td>
<td>1</td>
<td>1.28</td>
</tr>
</tbody>
</table>

Table 2: Table comparing effect of error in $p_{k,r_s}^{t,d}$ on Root Mean Square Logarithmic Error (RMSE) with and without partially observed departures constraints.

Unlike the case with error in $p_{r_s}^r$ values, the RMSLE values between scenarios with standard error 0.1 and 1 are significant, both with and without partially observed departures constraints. Hence, this motivates employment of simulation-based techniques to get more accurate values of $p_{k,r_s}^{t,d}$.

4.5 Scalability of Algorithm

To test the scalability of the dynamic departure time algorithm, it was tested on the full scale network of the nine counties in the Bay Area. The network contains 564,367 links and 352,011 nodes (Figure 9). The zones used for this implementation are 1454 Traffic Analysis Zones (TAZs) defined by the Metropolitan Transportation Commission (MTC) in 2002. Link counts for every 5-minute interval are available for 1045 out of the 564,367 links through sensors placed on certain freeways by the Caltrans Performance Measurement System (PeMS).

The methodology employed for dynamic departure time estimation on the full-scale network was the same as that for the simplified freeway network with the following exceptions:
Extraction of link-route incidence matrix through simulation-based methods.
Since the previous shortest path assumption is no longer valid, link-route incidence matrix was extracted by tracking exact agent routes during MATSim simulation.

Optimization algorithm.
The interior-point algorithm using MATLAB interface of optimization tool IP-OPT (Wachter, 2006) [3] was found to be inadequately slow for dealing with a full-scale optimization problem. Therefore, the gradient descent algorithm was employed to solve the optimization problem. Considerations were made for the large size of the link-use probability matrix $p_{k,r}^{t,d}$ given memory constraints. An implementation of the gradient descent algorithm using Python2.7 for the full scale network is available.

**Figure 9:** Full scale network representing the San Francisco Bay Area for testing scalability of the Dynamic Departure Time Estimation Framework.

The differences in optimization problem setup and evaluation for the full-scale implementation of the dynamic departure time estimation algorithm can be summarized as follows:

- **Calculating RMSE instead of RMSLE.**
  Since the actual number of departures in 5-min bins from each of the 1454 zones are much fewer than the previous 54 super-districts, the absolute values of error terms are also low. Hence, it is more appropriate to calculate the root mean squared error (RMSE) instead of the root mean squared logarithmic error (RMSLE).

- **Optimal $\beta_k$.**
  Unlike the case with 54 zones (where optimal value of beta was 1), for 1454 zones the optimal

---

The value of $\beta^t_k$ was found to be 10 (Figure 10). This makes sense because there is more spatial variation in number of departures now which is captured by link counts. Therefore, more weight should be given to link count constraints.

- **Value of RMSE.**
  The optimal RMSE value was found to be around 1.55. This compares unfavorably with the small-scale example of freeway network. This drop in accuracy is expected and is due to the sparsity of counts, that is, the data available for only 1045 out of 564,000 links. With higher sensor coverage, the error is expected to be reduced.

- **Run times.**
  The run time of the algorithm was found to be around 40ms per iteration of gradient descent step (7), and around 60s up to convergence with tolerance parameter $\epsilon$ in Eq. (8) set as 0.01.

![Figure 10: Calibration of $\beta^t_k$ - Plot of Root Mean Squared Error (RMSE) vs $\log(\beta^t_k)$ for full scale network.](image)

### 5 Conclusions and Future Work

This paper proposes a data-driven algorithm for estimating departure times dynamically on a traffic network. It extends previous research by Janson et al (1992) [1] proposing an Entropy Maximization approach to infer departure times from link counts. It also incorporates partially observed departures through data sources available in a form of mobility traces (like Call Detail Records) as lower bound constraints and partially observed waypoints to infer route choice probabilities (Wu et al, 2015 [2]). The entire problem is formulated as a convex optimization problem in Equation (6) which can be solved by a primal-dual interior-point algorithm with a filter line-search method proposed by Wachter et al (2006) [3]. Finally, batch gradient descent approach is adopted to solve the problem incrementally with incoming data which allows the framework to be adopted for Advanced Traffic Management, in a scenario when data is available as a live stream.

The suggested algorithm is employed to predict departure times in a simulated scenario representing a simplified freeway network for the nine counties of the San Francisco Bay Area. Agent-based traffic simulation software MATSim was used to produce a realistic synthetic evaluation dataset, based on the travel model developed in [33]. The parameters of the combined optimization problem are calibrated
by minimizing the Root-Mean Squared Logarithmic Error (RMSLE) in number of departures across all zones for a particular time interval. Results show that inclusion of partially observed departures and higher data coverage greatly reduces the absolute error in number of estimated departures across all zones. Incremental estimation allows one to minimize error with newly incoming data. It was shown that error in measurement of Origin-Destination (O-D) probability matrix can be overcome through availability of fine grained data from other data sources while errors in measurement of link-route incidence matrix can be reduced through use of simulation-based techniques. Lastly, scalability of the algorithm was proved for a full-scale network representing the nine counties of the Bay Area. Empirical evaluation of the runtime of our current implementation was found to be around 60s, which satisfies the requirement for being able to update the solution at 5 minute intervals with incoming streaming data, under the assumption that the required route inference step of [2] is on par in terms of the performance.

An extended empirical evaluation have shown that the algorithm could prove useful in estimating departure times and deliver a crucial input for agent-based traffic simulations as well as for informing dynamic control strategies aimed at mitigating congestion on a traffic network. We expect that as data become available with the increasing penetration of IT services into transportation systems, it will find its use in practice.

5.1 Future Work

Finally, we would like to highlight a promising extension we plan to elaborate in the future, besides solving multiple engineering problems to ease the real-life deployment of the method in industry. Assume that all trips in the system are unimodal (multi-modal trips may be segregated into several unimodal components), and that there is a complete link traffic count and travel-time information for the mode over all links in the network in real-time. Under the validity of these assumptions, the proposed dynamic departure time estimation framework may be directly extended to a mixture of multiple modes in the network, as shown in Equation (10):

\[
\min_{q^d_r(m)} \sum_{r \in Z} \sum_{d \in D} \{q^d_r(m) \cdot \log(q^d_r(m)) - q^d_r(m)\} + \sum_{k \in L} \sum_{t \in D} \{\beta^t_k(m) \cdot (v^t_k(m) - \sum_{r \in Z} \sum_{d \in D} q^d_r(m) \cdot p^t_k,m_r(m))\} + \sum_{r \in Z} \sum_{d \in D} \phi(q^d_r(m)) \\
\text{subject to } q^d_r(m) \geq 0, \forall r \in Z, d \in D
\]

where:

\(q^d_r(m)\) is a number of trips departing from zone \(r\) to all destinations in time interval \(d\) for mode \(m\),

\(\beta^t_k(m)\) is a relative cost parameter for link constraint for link \(k\) at time interval \(t\) for mode \(m\),

\(v^t_k(m)\) is the observed volume on link \(k\) during time interval \(t\) for mode \(m\),

\(p^t_k,m_r,d(m)\) is a probability that a trip departing zone \(r\) in time interval \(d\) uses link \(k\) in time interval \(t\) for mode \(m\),

\(\phi(q^d_r(m)) = -\log(q^d_r(m)) - \max(N^d_r(m) - N^{d-1}_r(m), 0))\), where
$N_r^d(m)$ is a number of observed individuals at zone $r$ during time interval $d$ for mode $m$.

An advancement of this direct extension to the case of highly multi-modal trips with multiple modes accounting for waiting times and coordinated scheduling of modes presents further challenge.

**Acknowledgement**

This work was partly funded by NSF CRISP program, award number 1541181, the State of California Department of Transportation (CalTrans) through UCCONNECT faculty research grant program, agreement 65A0529, and a research gift from AT&T. The authors would like to acknowledge the research assistance provided by Emin Arakelianan, Madeline Sheehan and Mogeng Yin. We would also like to thank Andrew Campbell, Sid Feygin and Danqing Zhang for their support and comments on the research.

**References**


