Virtual Trip Lines for Distributed Privacy-Preserving Traffic Monitoring

Baik Hoh, Marco Gruteser
WINLAB / Dept. of Electrical & Computer Engineering, Rutgers Univ.
Piscataway, NJ USA
baikhoh,gruteser@winlab.rutgers.edu

Ryan Herring, Jeff Ban[a], Daniel Work, Juan-Carlos Herrera, Alexandre M. Bayen
Dept. of Civil & Environmental Engineering, UC Berkeley; [a]California Center for Innovative Transportation (CCIT)
Berkeley, CA USA
ryanherring,dbwork,jcherrera,bayen@berkeley.edu; [a]xban@calccit.org

Murali Annavaram[b], Quinn Jacobson[c]
[b]Ming Hsieh Dept. of Electrical Engineering, USC; [c]Nokia Research Center
[b]Los Angeles, CA USA; [c]Palo Alto, CA USA
[b]annavara@usc.edu; [c]quinn.jacobson@nokia.com

ABSTRACT

Automotive traffic monitoring using probe vehicles with Global Positioning System receivers promises significant improvements in cost, coverage, and accuracy. Current approaches, however, raise privacy concerns because they require participants to reveal their positions to an external traffic monitoring server. To address this challenge, we propose a system based on virtual trip lines and an associated cloaking technique. Virtual trip lines are geographic markers that indicate where vehicles should provide location updates. These markers can be placed to avoid particularly privacy sensitive locations. They also allow aggregating and cloaking several location updates based on trip line identifiers, without knowing the actual geographic locations of these trip lines. Thus they facilitate the design of a distributed architecture, where no single entity has a complete knowledge of probe identities and fine-grained location information. We have implemented the system with GPS smartphone clients and conducted a controlled experiment with 20 phone-equipped drivers circling a highway segment. Results show that even with this low number of probe vehicles, travel time estimates can be provided with less than 15% error, and applying the cloaking techniques reduces travel time estimation accuracy by less than 5% compared to a standard periodic sampling approach.

Categories and Subject Descriptors: K.4.1 [Computer and Society]: Public Policy Issues—Privacy; K.6 [Management of Computing and Information Systems]: Security and Protection

General Terms: Algorithms, Design, Experimentation, Security

Keywords: Privacy, GPS, Traffic, Data integrity

MobiSys'08, June 17–20, 2008, Breckenridge, Colorado, USA.

1. INTRODUCTION

Automotive navigation systems enable the effective delivery and presentation of fine-grained traffic information to drivers and have thus created demand for improved traffic data collection. Conventional traffic information services rely on eyewitness reports, traffic cameras, and loop detectors. More recently, traffic estimates are also generated from cellular base station hand off rates [24]. Due to their high cost, or low precision position information, these mechanisms are only available at selected traffic hotspots.

Probe vehicle systems promise to significantly improve coverage and timeliness of traffic information. Probe vehicle systems estimate traffic flow and congestion through speed reports collected from a set of GPS-equipped vehicles. Thus, with sufficient penetration (fraction of total traffic) this approach could potentially collect real-time traffic information over the complete road network at minimal cost for transportation agencies.

Several studies have demonstrated the feasibility of traffic flow estimation through analysis, simulations, and experiments [25, 22, 58]. Several open questions remain, however, before such a system is likely to be realized. First, it is unclear how such a system can quickly be bootstrapped since the service is only useful with sufficient participants. While telematics platforms or navigations system hardware is capable of performing these functions, these platforms are not openly programmable and thus hard to retrofit for this purpose. Second, it is not known how the quality of the obtained traffic information compares with those collected through conventional methods (e.g., loop detectors). Third, the system requires that cars reveal their positions to a traffic monitoring organization, raising privacy concerns. Our earlier work [37] has proposed privacy enhancing technologies that can alleviate concerns. These solutions, however, still require users to trust centralized privacy servers.

To address these challenges, we propose a novel traffic monitoring system design based on the concept of *virtual trip lines (VTLs)* and experimentally evaluate its accuracy. Virtual trip lines are geographical markers stored in the client, which trigger a position and speed update whenever a probe vehicle passes. Through privacy-aware placement of these trip lines, clients need not rely on a trustworthy server. The system is designed for GPS-enabled cell phones

to enable rapid software deployment to a large and increasing number of programmable smart phones. The key contributions of this work are:

- Arguing that sampling in space (through virtual trip lines) rather than in time leads to increased privacy because it allows omitting location samples from more sensitive areas.
- Describing a privacy-aware placement approach that creates the virtual trip line database.
- Demonstrating that the virtual trip line concept can be implemented on a GPS-enabled cellular phone platform.
- Evaluating accuracy and privacy through a 20 vehicle experiment on a highway segment.

The remainder of this paper is organized as follows. Section 2 summarizes the challenges in currently existing traffic monitoring techniques. In section 3, we describe the possible privacy risks and threat models based on highway traffic characteristics in probe vehicle based traffic monitoring. Section 4 introduces the traffic monitoring approach that uses the novel concept of virtual trip lines. We describe the privacy preserving trip line placement algorithms in section 5. Section 6 describes the system implementation on smartphones. We describe the experimental setup and privacy and accuracy results obtained through the 20 car experiment in section 7. We then discuss limitations and extensions in section 8, review related work in section 9, and conclude.

2. TRAFFIC MONITORING CHALLENGES

Inductive Loop Detection (ILD) systems are the most common highway traffic monitoring tool, and have been in use for decades. Current highway monitoring systems consist of wire inductive loops placed directly in the top layer of the pavement. When a vehicle passes over the sensor, it is recorded by a roadside controller. These sensors suffer from two fundamental drawbacks:

- They are accurate for flows (vehicle counts), but often generate inaccurate velocity measurements. California freeways are equipped with about 23,000 loop detectors embedded in the pavement, accounting for roughly 8,000 detector stations. Several of these stations feature a single inductive loop per lane, which cannot measure vehicle speed directly. Practitioners have attempted to create aggregate velocity estimates using the average length of a vehicle on the highway and the percentage time the sensor is occupied. Even when the sensor is working properly, these estimates are particularly noisy (some ranging from 20 mph to 120 mph) for traffic flowing at speeds greater than 50 mph [21]. This led researchers to develop algorithms to improve these single loop estimates [21], [47], [51], [32], [39]. In contrast, dual loops (composed of two successive inductive loops) compute velocity by matching the respective occupancy patterns. In practice, they also have been found to produce significant amounts of error [20].
- They are expensive to deploy and maintain. The sensors cost is roughly $900-$2000 based on the type of the loop. More importantly the deployment direct and indirect costs are significant (staff to install the sensors, and corresponding impact on traffic), and repairing a malfunctioning loop detector is also costly. According to the PeMS (PErformance Measurement System) [3], only 65% of the detectors in California are working properly, the main cause of malfunction being problems with the controller. In [50], malfunction rates of loop detectors and their causes were examined using data obtained from loops for the same stretch of I-880 as the present work is based. The average malfunction rate was 21%, despite significant efforts to maintain the system operational during the study.

New technologies have been developed to overcome the limitations of loop detectors for travel time estimation, including *RFID In-Vehicle Transponders* (IVTs) and *License Plate Readers* (LPRs). The IVTs (often used for automatic toll payment, such as *FasTrak* in California or *E-ZPass* in some states in the East coast) are read using roadside readers. The travel times are measured for segments of the highway between each reader the vehicle passes. This technology is successful only when drivers have an incentive to carry the transponder (such as shorter toll booth queues), and can only provide travel times between segments where the readers have been deployed. Similarly, license plate readers consist of high speed cameras which record the license plates of vehicles on the highway. As a vehicle passes multiple cameras, the travel time between the readers is computed. Although LPRs avoid the need for in-vehicle equipment, these systems are complicated to install, and require an additional camera for each lane of traffic to be monitored. The relatively high cost of the readers (in the $10,000 plus installation costs), have limited their widespread implementation. Additionally, both these technology are extremely privacy intrusive, since they link origin-destination information and average speed to transponder ID and plate number in the plate reader case. For an exhaustive description of the different in- and over-roadway sensor technologies, the reader is referred to [42].

Traffic monitoring through GPS-equipped vehicles promises to overcome the challenges mentioned before. This new sensing technology has greatly impacted the transportation field, constituting a novel way to monitor traffic. Some studies have already started to look in more detail at how this new source of traffic data can be used for monitoring purposes. Some of the studies use microsimulation data to generate the GPS measurement to estimate travel time [49] and [19], while some others make use of real vehicle trajectories to generate GPS measurement to estimate traffic state [34]. Note however that while GPS equipped phones have the potential of providing ubiquitous information for speed of vehicles, the problem of counts remains partially open for new. All these studies also assume that GPS-equipped vehicles broadcast their position and/or velocity in real time at a certain rate, which raises significant privacy concerns, as discussed in the next section.

3. PRIVACY RISKS AND THREAT MODEL

Traffic monitoring through GPS-equipped vehicles raises significant privacy concerns, however, because the external traffic monitoring entity acquires fine-grained movement traces of the probe vehicle drivers. These location traces might reveal sensitive places that drivers have visited, from which, for example, medical conditions, political affilitations, romantic relationship, speeding, or potential involvement in traffic accidents could be inferred.

Threat Model and Assumptions. This work assumes that adversaries can compromise any single infrastructure component to extract information and can eavesdrop on network communications. We assume that different infrastructure parties do not collude and that a driver's own handset is trustworthy. We believe this model is useful in light of the many data breaches that occur due to dishonest insiders, hacked servers, stolen computers, or lost storage media (see [6] for an extensive list, including a dishonest insider case that released 4500 records from California's FasTrak automated road toll collection system). These cases usually involve the

compromise of log files or databases in a single system component and motivate our approach of ensuring that no single infrastructure component can accumulate sensitive information.

We consider sensitive information any information from which the precise location of an individual at a given time can be inferred. Since traffic monitoring does not need to rely on individual node identities, only on the aggregated statistics from a large number of probe vehicles, an obvious privacy measure is to anonymize the location data by removing identifiers such as network addresses. This approach is insufficient, however, because drivers can often be re-identified by correlating anonymous location traces with identified data from other sources. For example, home locations can be identified from anonymous GPS traces [44, 36] which may be correlated with address databases to infer the likely driver. Similarly, records on work locations or automatic toll booth records could help identify drivers. Even if anonymous point location samples from several drivers are mixed, it can be possible to reconstruct individual traces because successive flow updates from the same vehicle inherently share a high spatio-temporal correlation. If overall vehicle density is low, location updates close in time and space likely originate from the same vehicle. This approach is formalized in target tracking models [52].

As an example of tracking anonymous updates, consider the following problem: given a time series of anonymous location and speed samples mixed from multiple users, extract a subset of samples generated by the same vehicle. To this end, an adversary can predict the next location update based on the prior reported speed $\hat{x}_{t+\Delta t} = v_t \cdot \Delta t + x_t$ of the actual reported updates, where x_t and $x_{t+\Delta t}$ are locations at time t and $t + \Delta t$, respectively, and v_t is the reported speed at t. The adversary then associates the prior location update with the next update closest to the prediction, or more formally with the most likely update, where likelihood can be described through a conditional probability $P(x_{t+1}|x_t)$ that primarily depends on spatial and temporal proximity to the prediction. The probability can be modelled through a probability density function (pdf) of distance (or time) differences between the predicted update and an actual update (under the assumption that the distance difference is independent of the given location sample).

Knowing speed patterns further helps tracking anonymous location samples if it is combined with map information. For example, consider the traffic scenarios depicted in figure 1. On straight sections (a) vehicles on high-occupancy vehicle (HOV) or overtaking lanes often experience less variance in speed. Vehicles entering at an on-ramp (b) or exiting after an off-ramp (c) usually drive slower than main road traffic. These general observations can be formally introduced into the tracking model by assigning an a priori probability derived from the speed deviations. For example, to identify the next location sample after an on-ramp for a vehicle that generated x_t on the main route before the ramp, an adversary could assign a lower probability to location updates with low speed. These low speed samples are likely generated by vehicles that just entered after the ramp.

Privacy Metrics. As observed in [37], the degree of privacy risk depends on how long an adversary successfully tracks a vehicle. Longer tracking increases the likelihood that an adversary can identify a vehicle and observe it visiting sensitive places. We thus adopt the *time-to-confusion* [37] metric and its variant *distance-to-confusion*, which measures the time or distance over which tracking may be possible. Distance-to-confusion is defined as the travel distance until tracking uncertainty rises above a defined threshold. Tracking uncertainty is calculated separately for each location update in a trace as the entropy $H = -\sum p_i \log p_i$, where the p_i are the normalized probabilities derived from the likelihood values

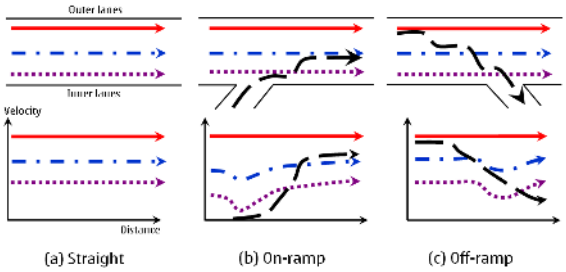


Figure 1: Driving Patterns and Speed Variations in Highway Traffic.

described later. These likelihood values are calculated for every location update generated within a temporal and spatial window after the location update under consideration.

These tracking risks and the observations regarding increased risks at certain locations further motivate the virtual trip line solution described next. Compared to a periodic update approach, where clients provide location and speed updates at regular time intervals, virtual trip lines can be placed in a way to avoid updates from sensitive areas.

4. TRAFFIC MONITORING WITH VIRTUAL TRIP LINES

We introduce the concept of virtual trip lines for privacy-preserving monitoring and describe two architectures that embody it. The first architecture seeks to provide probabilistic privacy guarantees with virtual trip lines. The extended second architecture demonstrates how virtual trip lines can help computing k-anonymous location updates via temporal cloaking, without using a single trusted server.

4.1 Design Goals

We aim to achieve both high quality traffic information and strong privacy protection in this traffic monitoring system. There exists an inherent tradeoff between these requirements, since privacy-enhancing technologies such as spatial cloaking [30] reduce accuracy of traffic monitoring.[1]

Privacy. We aim to achieve privacy protection by design so that the compromise of a single entity, even by an insider at the service provider, does not allow identifying or tracking users.

Data Integrity. The system should not allow adversaries to insert spoofed data, which would reduce the data quality of traffic information. This is especially challenging because it conflicts with the desire for anonymity.

Smartphone Client. The client software must cope with the resource constraints of current smartphone platforms.

In this article, we do not consider energy consumption because we assume that participants are using their phones in a charging dashboard mount to view navigation and traffic information (as illustrated in figure 11).

[1]One may expect, however, that a privacy-preserving design motivates more users to participate in such a system, which would improve the quality of traffic information.

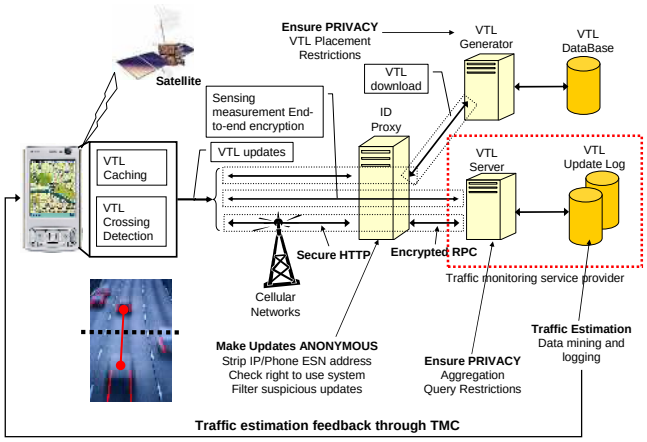


Figure 2: Virtual Trip Line: Privacy-Preserving Traffic monitoring System Architecture.

4.2 Virtual Trip Line Concept

The proposed traffic monitoring system builds on the novel concept of virtual trip lines and the notion of separating the communication and traffic monitoring responsibilities (as introduced in [36]). A *virtual trip line* (VTL) is a line in geographic space that, when crossed, triggers a client's location update to the traffic monitoring server. More specifically, it is defined by

$$[id, x1, y1, x2, y2, d]$$

where id, is the trip line ID, $x1$, $y1$, $x2$, and $y2$ are the (x, y) coordinates of two line endpoints, and d is a default direction vector (e.g., N-S or E-W). When a vehicle traverses the trip line its location update comprises time, trip line ID, speed, and the direction of crossing. The trip lines are pregenerated and stored in clients.

Virtual trip lines control disclosure of location updates by sampling in space rather than sampling in time, since clients generate updates at predefined geographic locations, compared to sending updates at periodic time intervals. The rationale for this approach is that in certain locations traffic information is more valuable and certain locations are more privacy-sensitive than others. Through careful placement of trip lines the system can thus better manage data quality and privacy than through a uniform sampling interval. In addition, the ability to store trip lines on the clients can reduce the dependency on trustworthy infrastructure for coordination.

4.3 Architecture for Probabilistic Privacy

To achieve the anonymization of flow updates from clients while authenticating the sender of flow updates, we split the actions of authentication and data processing onto two different entities, an ID proxy server and a traffic monitoring server. By separately encrypting the identification information and the sensing measurements (i.e., trip line ID, speed, and direction) with different keys, we prevent each entity from observing both the identification and the sensing measurements.

Figure 2 shows the resulting system architecture. It comprises four key entities: probe vehicles with the cell phone handsets, an ID proxy server, a traffic monitoring service provider, and a VTL generator. Each probe vehicle carries a GPS-enabled mobile handset that executes the client application. This application is responsible for the following functions: downloading and caching trip lines from the VTL server, detecting trip line traversal, and sending measurements to the service provider. To determine trip line traversals, probe vehicles check if the line between the current GPS position and the previous GPS position intersects with any of the trip lines in its cache. Upon traversal, handsets create a VTL update comprising trip line ID, speed readings, timestamps, and the direction of traversal and encrypt it with the VTL server's public key. Handsets then transmit this update to the ID proxy server over an encrypted and authenticated communication link set up for each handset separately. Each handset and the ID proxy share an authentication key in advance.

The ID proxy's responsibility is to first authenticate each client to prevent unauthorized updates and then forward anonymized updates to the VTL server. Since the VTL update is encrypted with the VTL server's key, the ID proxy server cannot access the VTL update content. It has knowledge of which phone transmitted a VTL update, but no knowledge of the phones position. The ID proxy server strips off the identifying information and forwards the anonymous VTL update to the VTL server over another secure communication link.

The VTL server aggregates updates from a large number of probe vehicles and uses them for estimating the real-time traffic status. The VTL generator determines the position of trip lines, stores them in a database, and distributes trip lines to probe vehicles when any download request from probe vehicles is received. Similar to the ID proxy, each handset and the VTL generator should share an authentication key in advance. The VTL generator first authenticates each download requester to prevent unauthorized requests and can encrypts trip lines with a key agreed upon between the requester and the VTL generator.[2] Both the download request message and the response message are integrity protected by a message authentication code.

Discussion. The above architecture improves location privacy of probe vehicle drivers through several mechanisms. First, the VTL server must follow specific restrictions on trip line placements that we will describe in section 5. This means that a handset will only generate updates in areas that are deemed less sensitive and not send any information in other areas. By splitting identity-related and location-related processing, a breach at any single entity would not reveal the precise position of an identified individual. A breach at the ID proxy would only reveal which phones are generating updates (or are moving) but not their precise positions. Similarly, a breach at the VTL server would provide precise position samples but not the individual's identities. Separating the VTL server from the VTL generator prevents active attacks that modify trip line placement to obtain more sensitive data. This is, however, only a probabilistic guarantee because tracking and eventual identification of outlier trips may still be possible. For example, tracking would be straightforward for a single probe vehicle driving along on empty roadway at night. The outlier problem in sparse traffic situations can be alleviated by changing trip lines based on traffic density heuristics. Trip lines could be locally deactivated by the client based on time of day or the clients speed. They could also be deactivated by the VTL generator based on traffic observations from other sources such as loop detectors. At the cost of increased complexity, the system can also offer k-anonymity guaratnees regardless of traffic density. We will describe this approach next.

4.4 Extensions for VTL-based Temporal Cloaking

We propose a distributed VTL-based temporal cloaking scheme that reduces timestamp accuracy to guarantee a degree of k-anonymity in the dataset accumulated at the VTL server. This provides

[2]While VTL positions are not highly sensitive, encryption reduces the possibility of timing analysis (see section 8.1).

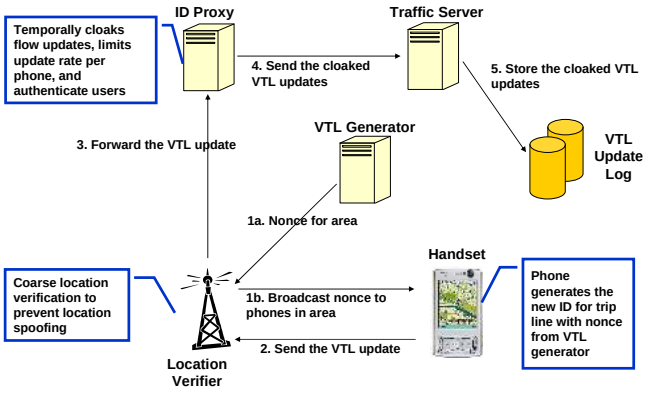


Figure 3: Distributed Architecture for VTL-based Temporal Cloaking.

a stronger privacy guarantee than probabilistic privacy, it prevents tracking or identification of individual phones based on anonymous position updates even if the density of phones is very low. The key challenge in applying temporal cloaking is concealing the location of probe vehicles from the cloaking entity. To calculate the time interval for nodes at the same location the cloaking entity typically needs access to the detailed records of each data subject [56, 30], which itself can raise privacy concerns.

Using virtual trip lines, however, it is possible to execute the cloaking function without access to precise location information. The cloaking entity can aggregate updates by trip line ID, without knowing the mapping of trip line IDs to locations. It renders each location update k-anonymous by replacing VTL timestamp with a time window during which at least k updates were generated from the same VTL (i.e., $k-1$ other phones passed the VTL). In effect k VTL updates are aggregated into a new update $(vtlid, \frac{s_1 \ldots s_k}{k}, \max(t_1 \ldots t_k))$, where s_i denotes the speed reading of each VTL update i. Since now k-phones generate the same update, it becomes harder to track one individual phone. The cloaking function can be executed at the ID proxy, if handsets add a VTL ID to the update that can be accessed by the ID proxy.

Beyond the cloaking function at the ID Proxy, two further changes are needed in the architecture to prevent an adversary from obtaining the mapping of VTL IDs to actual VTL locations. The system uses two techniques to reduce privacy leakage in the event of phone database compromises. First, the road network is divided into tiles and phones can only obtain the trip line ID to location mapping for the area in which the phone is located. This assumes that the approximate position of a phone can be verified (for example, through the cellular network). Second, the VTL server periodically randomizes the VTL ID for each trip line and updates phone databases with the new VTL IDs for their respective location.

This leads to the extended distributed architecture depicted in Figure 3, where again no central entity has knowledge of all three types of information: location, timestamp, and identity information. As before, VTL updates from phones to the ID proxy are encrypted, so that network eavesdroppers do not learn position information. It first checks the authenticity of the message and limits the update rate per phone to prevent spoofing of updates. It then strips off the identification information and forwards the anonymous update to the VTL server. Knowing the mapping of VTL IDs to locations, the VTL server can calculate road segment travel times. This architecture differs, in that the ID proxy server cloaks anonymous updates with the same VTL ID before forwarding to the VTL server. It also requires a Location Verifier entity, which can coarsely verify phone location claims (e.g., in range of a cellular base station) and distribute the VTL ID updates to only the phones that are actually present within a specified tile. Table 1 summarizes the roles of each entity and how information is split across them.

The temporal cloaking approach can be vulnerable to spoofing attacks unless it is equipped with proper protection. For instance, malicious clients can send many updates to shorten the cloaking time window. To prevent this denial of service attack, the ID proxy server limits the update rate per phone.

To reduce network bandwidth consumption of the periodic VTL updates, clients can independently update the VTL IDs based on a single nonce per geographic area (tile). The VTL server generates the nonces using a cryptographically secure pseudorandom number generator and distributes each nonce to the clients currently in the tile area. Both clients and server can then compute $VTLID_{new} = h(nonce, VTLID_{old})$, where h is a secure hash function such as SHA.

Discussion. Temporal cloaking fits well with the travel time estimation method used in the VTL system because the mean speed calculation does not depend on accurate timestamp information. To estimate the travel time, the VTL server calculates the mean speed for a trip line only based on the speed information in the flow updates. Typically, the travel time would be periodically recomputed. The use of temporal cloaking adaptively changes this update interval so that at least k phones have crossed the trip line. If k is chosen large, it reduces the update frequency. Even with temporal cloaking, however, the travel time algorithm would need speed reports from several vehicles to provide reliable estimates.

5. TRIP LINE PLACEMENT

This section describes placement algorithms that choose virtual trip line locations to maximize travel time accuracy and preserve privacy. A basic algorithm, the even placement approach, takes as input a partial road network graph. For each road segment, which refers to a stretch of road between two intersections or merges, the algorithm observes an exclusion zone at the beginning of the segment and then places equidistant trip lines orthogonally to the road. Key to preserving privacy are the procedures for determining appropriate trip line spacing parameters and exclusion zone sizes, which we discuss in the following subsection. Privacy is also significantly improved by selecting only higher traffic roads for trip line placement, such as highways and aterials, which usually are less sensitive areas.

5.1 Placement Privacy Constraints

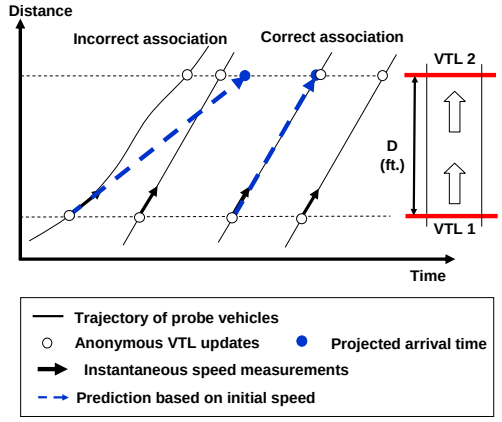
The algorithm considers two types of placement constraints, *exclusion areas* around sensitive locations and *minimum spacing* constraints to reduce tracking ability along straight roadways.

Determining Minimum Spacing. The minimum spacing constraint is particularly important on highways, where more regular traffic flows increase the tracking risks. Thus, we focus our derivations on straight highway scenarios. Minimum spacing for longer road segments is determined based on a tracking uncertainty threshold. Recall that to prevent linking compromises, an adversary should not be able to determine with high confidence that two anonymous VTL updates were generated by the same handset.

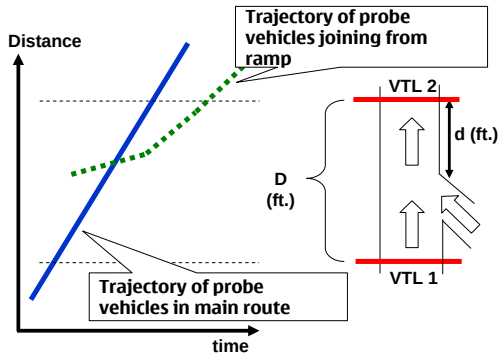
Tracking uncertainty defines the level of confusion that an adversary encounters when associating two successive anonymous flow updates to each other. We define tracking uncertainty as the entropy $H = -\sum p_i \log p_i$, where p_i denotes the probability (from the adversary's perspective) that anonymous flow update i at the next trip line was generated by the same phone as a given anony-

Entity	Role	Identity	Location	Time
Handset	Sensing	Yes	Accurate	Accurate
Location Verifier	Distributing VTL ID Updates	Yes	Coarse	Accurate
ID proxy	Anonymization and Cloaking	Yes	Not Available	Accurate
Traffic Server	Computing Traffic Congestion	No	Accurate	Cloaked

Table 1: Entity roles and splitting of sensitive information across entities



(a) Linking based on Prediction.



(b) Linking around On-Ramp.

Figure 4: Linking attack scenarios on straight highway section and on-ramp section.

mous flow update at a previous trip line. The probability p_i is calculated based on an empirically derived pdf model that takes into account the time difference between the predicted arrival time at the next trip line and the actual timestamp of flow update i. We fit an empirical pdf of time deviation with an exponential function, $\hat{p_i} = \frac{1}{\alpha} e^{-\frac{t_i}{\beta}}$, where we obtain the values of α and β by using unconstrained nonlinear minimization.

Consider the example scenario in Figure 4(a). In scenario (a) the adversary projects the arrival time at VTL 2 based on the phone's speed report at VTL 1. The projected arrival time is the endpoint of the dashed line (the solid lines indicate phones' actual paths). There are two actual flow updates at VTL 2 (indicated through points). The adversary would calculate the time difference between the projected arrival times, assign probabilities p_1 and p_2 using the pdf, and determine entropy. Compared to the second example in scenario (a) entropy is high indicating that an adversary cannot determine the correct VTL update with high confidence. In fact, the closest update would be incorrect in this scenario. Tracking uncertainty calculated through the entropy is maximized when there is more than one anonymous flow update with similar time-differences. In other words, lower values of H indicate more certainty or lower privacy. To reduce computational complexity, we only consider VTL updates within a time window w of the projected arrival time during the entropy calculation.

In general tracking uncertainty is dependent on the spacing between VTLs, the penetration rate, and speed variations of vehicles. If speed remains constant, as in the second example of figure 4(a), the projected arrival times match well and tracking uncertainty is low. Higher penetration rates lead to more VTL updates around the projected arrival time, which decreases certainty. As spacing increases, the likelihood that speeds and the order of vehicles remain unchanged decreases, leading to more uncertainty. Speed variations on highways are frequently caused by congestion—thus road segments with lower average speed tend to increase tracking uncertainty.

We empirically validate these observations through simulations using the PARAMICS vehicle traffic simulator [5]. Figure 5 depicts the minimum spacing required to achieve a minimum mean tracking uncertainty of 0.2 for different penetration rates and different levels of congestion (or mean speed of traffic). We choose a reasonably low uncertainty threshold, which ensures to an adversary a longer tracking that could have privacy events such as two different places (e.g., origin and destination).[3] The uncertainty value of 0.2 corresponds to an obvious tracking case in which the most likely hypothesis has a likelihood of 0.97. The penetration rates used were 1%, 3%, 5% and 10%. To evaluate different levels of congestion, we used traces from seven 15 min time periods distributed over one day. We also used three different highway sections (between the junction of CA92 and the junction of Tennyson Rd., between the junction of Tennyson Rd. and the junction of Industrial Rd., and between the junction of Industrial Rd. and the junction of Alvarado-Niles Rd.) to reduce location-dependent effects. The simulations show that the needed minimum spacing decreases with slower average speed and higher penetration rate.

The clear dependency of the tracking uncertainty on the penetration rate and the average speed allows creating a model that provides the required minimum spacing for a given penetration rate and the average speed of the target road segment.

Determining Exclusion Areas. Additional tracking risks are present at ramps and many intersections because of large speed variations. Vehicles leaving the main direction of travel move slower and vehicles joining the main direction of travel from ramps accelerate from a very low speed while vehicles staying on the main direction maintain their speed. Figure 4(b) depicts the latter case. If trip lines are placed immediately before or after intersections, an adversary may be able to follow vehicles paths based on speed differences.

In a case study, we determine the required size of an exclusion area using the 20 vehicle dataset described in the next section (PARAMICS does not model ramp trajectories in sufficient detail). Figure 6(a) shows the difference in speed between the merging traf-

[3]Two recent studies [44, 36] observe about 15 minutes as a median trip time.

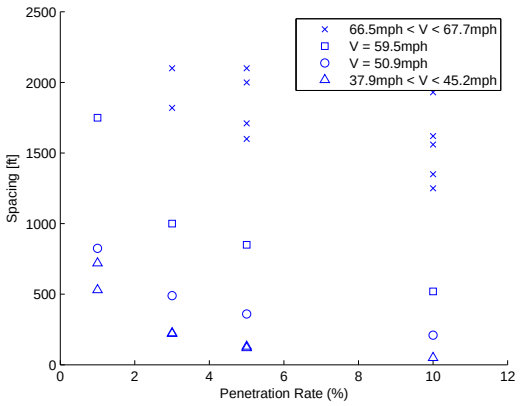
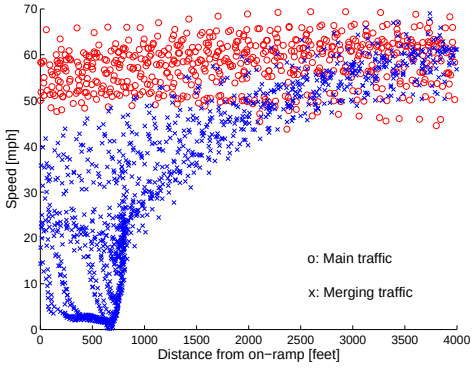
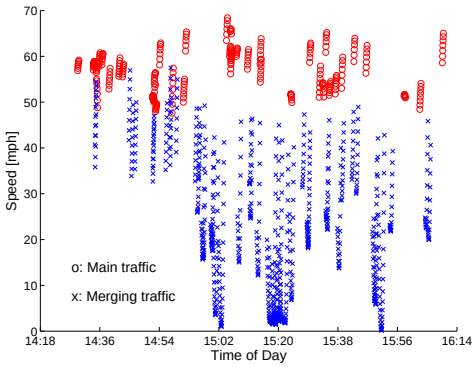


Figure 5: Minimum Spacing Constraints for Straight Highway Section.



(a) Speed Measurements over Distance.



(b) Speed Measurements over Time.

Figure 6: Exclusion Area Constraints for Highway On-ramp Section.

fic (red circles) and the main traffic (blue crosses) with increasing distance from the on-ramp. As expected, we observe that near the on-ramp an adversary can distinguish main and merging traffic through a simple speed threshold (about 40 mph in our scenario). The graph also shows that speeds converge at a distance of about 3000 ft from the on-ramp, which can be used as an exclusion zone size.

The figure 6(b) depicts a weekday time interval between 1:30 PM to 4:00 PM, during which the mean speed of the test road segment decreases due to the increasing congestion, as reported by the PeMS highway measurement database [3]. As evident, the speed variation between main route traffic and merging traffic increases with congestion. This may require longer exclusion during congestion.

In addition to merges and intersections, where detailed information would be especially important for an adversary to track which path a vehicle takes, exclusion zones can also be places around other sensitive places. These may be places that could allow sensitive inferences, such as a medical clinics, or locations where the driver of handsets may be identified (e.g., suburban home locations [44] or automatic toll booth plazas).

5.2 Optimal Placement for Traffic Flow Estimation Accuracy

Traffic flow estimation accuracy can be further improved by adjusting the positions of trip lines with the following optimal placement algorithm, which is described in more detail in [13]. To determine the optimal VTL locations for a given number K of VTLs on a given route, we first evenly divide the route into N segments, called sections. In practice, each section is assumed to be 50 to 100 ft long. We can reasonably assume that speed is approximately constant along this short section. Each VTL is associated with a link, which consists of one or multiple consecutive sections. We assume that the VTL is always in the middle of its link. This way, the studied route is divided into K links, where K is less than N. The travel time of the route is equal to the summation of all link travel times. Under the above assumptions, the optimal VTL location problem becomes how to determine the optimal starting and ending sections of each of the links.

Deploying VTLs can be solved optimally in a staged process, with one VTL deployed in each stage. This is a key feature in the formulation of the problem as a DP model. In particular, if we are to deploy K VTLs, the model will contain K stages. The state variable of each stage is the starting location of the link associated with the VTL of the stage, and the decision variable is the ending location of the link. The objective is to minimize the summation of mean squared errors of travel time estimates of all links for a given number of "representative" vehicles.

It is shown in [13] that such a DP model can be represented as an acyclic graph, and solving the optimal VTL location problem is equivalent to find the shortest path in the graph. The complexity of the algorithm is $O(KN^2)$, which is polynomial and can be used for solving large scale problems. Exclusion zones and minimum spacing constraints can be incorporated into the optimal placement algorithm by modifying the graph representation of the DP model, eliminating proper arcs. The complexity of solving the model with these constraints remains $O(KN^2)$.

6. IMPLEMENTATION

We have completely implemented the probabilistic privacy architecture and use this implementation for the experiments in the following section. The implementation uses Nokia N95 smartphone handsets, which include a full Global Positioning System receiver that can be accessed by application software.

6.1 Map Tiles and Trip Lines

In our system, we recursively divide the geographic region of interest into four smaller rectangles (or quadrants), and the minimum quadrant size is 1m by 1m. We convert the GPS location of a user into a Mercator projection using the WSG84 world model. Merca-

tor projects the world into a square planar surface. A *zoom* of 25 is assumed to be the maximum precision that location can be specified in. By default every GPS location is converted into 25 bit x and y values with *zoom* set to 25. By using the quadrant representation the mobile device can efficiently control the granularity by simply changing the *zoom* level. In this encoding, the world is treated as a square grid of four quadrants with *zoom* level 2, where x and y are the offsets from the top left corner of the world.

This reprentation makes it easy to specify the specific map tile. We define a map tile as a container that groups all trip lines within it. When a client wants to download all virtual trip lines within the San Francisco Bay Area, it sends the VTL server the triplet, $(zoom, x, y)$ for the corresponding region. In our implementation, we choose 12 as the default zoom level, which corresponds to an 8 km by 8 km square.

This representation also helps in reducing storage size and bandwidth consumption. Since the general area is identified by the quadrant, we only store the 13 least significant bits of the trip line end point coordinates instead of the full 25 bits used for typical UTM coordinates. This decreases storage consumption to 68bits (15 bit id, 1 bit direction, $4 \cdot 13$ bits coordinates) per trip line. As an example of required storage and bandwidth consumption, consider the section of the San Francisco Bay Area shown in figure 7. The total road network in the white titles shown in the left figure contains about 20,000 road segments, according to the Digital Line Graph 1:24K scale maps of the San Francisco Bay Area Regional Database (BARD [1], managed by USGS). Assuming that the system on average places one trip line per segment this results in 166KB of storage.

6.2 Client Device and Software

We implemented the client software using J2ME (Java Platform, Micro Edition) on an Nokia N95 handset. This Symbian OS handset uses an ARM11-based Texas Instruments OMAP2420 processor running at 330MHz, and it contains 64MB RAM and 160MB internal memory. Its storage can be expanded up to 8GB with flash memory. We use the JSR 179 library (Location API for J2ME) [2] for communicating with the internal TI GPS5300 NaviLink 4.0 single-chip GPS/A-GPS module to set the sampling period and retrieve the position readings. This setup did not provide speed information. Instead, we calculate the mean speed using two successive location readings (in our implementation, every 3 seconds). The client software registers the task for checking the traversal of trip lines as an event handler for GPS module location updates, which is automatically invoked whenever a new position reading becomes available.

The communication between the handset and the ID proxy server, to send VTL updates or to request VTL downloads, is implemented via HTTPS GET/POST messages. The client software encrypts the message content but not the handset identification information using the public key of the VTL server so that only the VTL server with the corresponding private key can decrypt the message. To save network bandwidth and to reduce delay, we cache the downloaded trip lines for the nine map tiles closest to the current position in local memory. When a vehicle crosses a tile boundary, it initiates VTL download background threads for the missing tiles.

6.3 Servers and Databases

VTL Server. At the bottom of the hierarchy of our server implementation is a backend database server. The database server contains two databases. First is a VTL database which holds GPS coordinates of all trip lines. In future we plan to enhance our trip line database to hold meta data associated with that trip line. For instance, the meta data for a trip line can contain the posted speed limit at that trip line which can be used by the client application to decide if it is going over the speed limit in which case the client application can disable VTL updates. Write access to this database is restricted only to traffic administrators who can add, delete or update a VTL.

Figure 7: Road networks extracted from Bay Area DLG files (Left) and Trip Lines per road segment in Palo Alto CA (Right).

The second database is the VTL update database. This database stores the VTL updates sent by the mobile device whenever the mobile device chooses to send an update after crossing a VTL. The update database simply appends every VTL update along with a time stamp on when the update was received. To sanitize bogus VTL updates from the clients, the VTL update database also keeps both the encrypted and decrypted versions of the VTL update for further investigation in collaboration with the ID proxy server. When bogus VTL updates are detected in the VTL update database, their encrypted versions are compared to the encrypted version stored in the ID proxy server to blacklist the originator of bogus VTL updates.

We use Microsoft SQL to implement the databases, and we develop the VTL server using J2EE (Java Platform, Enterprise Edition) and JDBC (Java Database Connectivity) to control the SQL databases that are connected to the VTL server. While we have used only a single DB server in this prototype, the two databases should ideally be implemented by different entities to prevent active trip line modification attacks by a compromised traffic monitoring entity.

ID Proxy Server. On top of the database server is the ID Proxy server. The identification proxy server is envisioned to be operated by an entity that is independent of the traffic service provider. We implement the ID proxy server as a servlet-based web server that takes in HTTPS GET/POST messages from clients and forwards messages to the VTL server. The HTTP message received by the proxy server from the client has two components. The first component contains the mobile device identification information, namely phone number of the message origin. This component of the message is required for all cell phone communications as operator needs to appropriately charge for data communication costs. The second component of the message contains information that is intended for the database server. The proxy server strips all the identification information from the message, namely the first component of the message, and passes on the second component of the message to the application server. We implemented the secure channel between ID proxy server and the VTL server using WSDL (Web Service Definition Language)-RPC (Remote Procedure Call) over J2EE Server.

7. EXPERIMENTAL EVALUATION

We evaluate our system first in terms of travel time estimation accuracy and then analyze privacy-accuracy tradeoffs.

7.1 Traffic Flow Estimation Accuracy

GPS Speed Accuracy. A first experiment was run to estimate the position and speed accuracy of a single cell phone carried onboard a vehicle. The experiment route consisted of a single 7 mile loop on I-80 near Berkeley, CA, and VTLs were placed evenly on the highway every 0.2 miles. Speed and position measurements were stored locally on the phone every 3 seconds, and speed measurements were sent over the wireless access provider's data network every time a VTL was crossed. The speed measurements were computed using two consecutive position measurements. In order to validate this calculation, the vehicle speed was also recorded directly from the speedometer on a laptop with a clock synchronized with the N95. In Figure 8, the speed measured directly from the vehicle speedometer is compared to the speeds measured by the VTLs and the speed stored in the phone log. Timestamp of each record denotes the elapsed time since midnight of the experiment day. On average, the vehicle odometer reported a speed 3 mph slower than the GPS. The position data was accurate enough to correctly place the vehicle on either the correct or neighboring lane of travel.

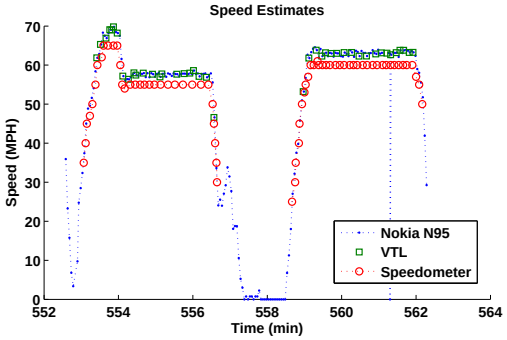


Figure 8: Comparison of the speed measurements recorded from the N95 (dots), the VTLs (boxes) and the vehicle speedometer (circles) as a function of time.

Figure 9: Satellite image of the first experiment site I-80 near Berkeley, CA. The red lines represent the locations of the VTLs, the blue squares show the speed recorded by the VTL, the green squares represent the position and speed stored in the phone log. The brown circles represent the readings from the vehicle speedometer.

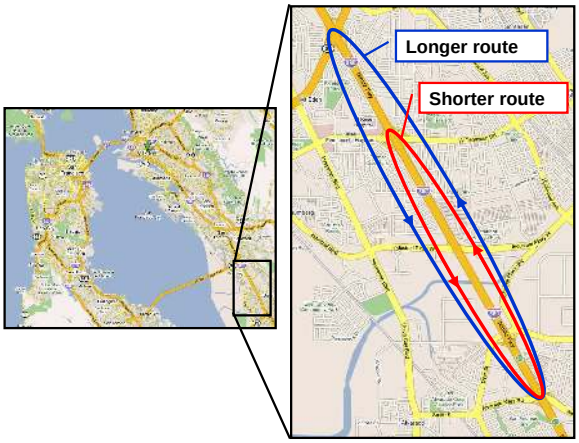


Figure 10: I880 Highway Segment for 20 Car Experiment.

To further validate the position accuracy of GPS enabled cell phones, the vehicle was driven on a frontage road along the highway, which poses a very important problem for cell phone based traffic monitoring. Frontage roads typically have slow moving traffic with speed limits of 25 or 35 mph and run alongside the freeway. Without high precision position accuracy, this traffic can be incorrectly identified as freeway traffic. In our study, the VTLs were only placed on the freeway, and they did not detect the vehicle as it traveled on the frontage road, despite the freeway an frontage road being separated in some locations by as little as 30 ft. Although the test was only conducted using a single phone, it presented promise that the technology can be used for advanced traffic monitoring applications.

Experimental Setup. A second experiment was conducted to demonstrate the feasibility of cell phone based travel time estimation in practice. Note that the accuracy experiments do not use placement constraints or temporal cloaking. We consider temporal cloaking separately in section 7.3. For two hours, twenty vehicles were driven back and forth on a 4 mile section of I-880 south of Oakland CA as shown in figure 10. The length (i.e., 4 miles) of test road segment was chosen to have 1% to 2% penetration rate given 20 participants and approximate round travel time. In order to observe a more natural mixing phenomena (in which vehicles pass each other) half of the drivers were instructed to drive a slightly shorter, 3 mile section of the highway (red circle) after the completion of the first lap. The location of this experiment was specifically selected because it featured both free flowing traffic at greater than 50 mph, and congested, stop and go traffic. An accident just north of the experiment site further added to the complexity of the northbound traffic flow. As observed by the drivers of the experiment, this accident created "shear" in the traffic flow, where vehicles in adjacent lanes of traffic were traveling at significantly different speeds.

We expect that actual users will place their phones into a dashboard car holder (depicted in the left of fig 11) to be able to view navigation and travel time information while driving. Since we did not have sufficient car holders available during the experiment, we placed the cell phones onto the dashboard as shown in the right of figure 11. For this experiment, 45 VTLs were first evenly placed to record the speed measurements from the 20 vehicles. Each phone also stored speed and position measurements to a local log every three seconds, following the same protocol as the first experiment. To estimate the travel time, the *instantaneous travel time* was com-

Figure 11: Experimental Setup in a Car for 20 Car Experiment.

puted, which assumes traffic conditions remain unchanged on every link[4] from the time the vehicle enters the link until it leaves the link. Therefore, the travel time of the section can be computed by simply summing those of the constituent links at the time a vehicle enters the route. The travel time of each link is computed with the length of a link and the mean speed that is obtained by averaging out speed readings from probe vehicles during *an aggregation interval*. The aggregation interval can vary from 10s of seconds to few minutes, depending on traffic condition. Its effect on travel time estimation accuracy will be examined in section 7.3.

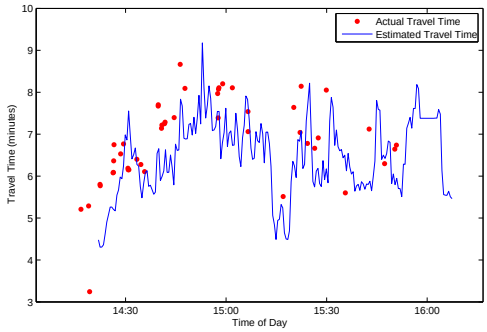


Figure 12: Actual travel times compared with an estimate given by the instantaneous method (30 second aggregation interval).

We then run the DP algorithm to compute the optimal VTL locations. Using *the instantaneous travel time method*, we plot actual travel times versus predicted travel times in figure 12. We obtain ground truth for actual travel time by checking logging data of each probe vehicle. Since variates between actual travel times and predicted travel times are positive and negative, we calculate Root Mean Squared (RMS) error between two sets to see the expected magnitude of a travel time estimation error. For a given 30 second aggregation interval, we achieved a RMS error of about 80 seconds.

7.2 Privacy-Accuracy Tradeoffs

This section analyzes the travel time estimation accuracy and privacy preservation of our probabilistic approach, the VTL-based placement algorithm.

To analyze privacy, we measure the *distance-to-confusion* with two different sets of anonymous flow updates from both the evenly spaced VTLs (with exclusion area) and the evenly spaced VTLs (without exclusion area). We call the latter *spatial periodic sampling*. We use the repeated south bound trips of the 20 probe vehicles, which contain the effect of merging traffic from the shorter loop (see figure 13). The south bound direction also has lower traffic volume than the north bound direction, providing a more challenging environment to protect privcay. On the experiment day, we verified from the PeMS [3] highway measurement database that our test road segment (on south bound from Route 92 to Alvarado) experienced about 5000 vehs/h as a traffic volume and an average speed of 55 mph. Because we have 88 traces from 20 probe vehicles during our 100 min test period, the penetration rate is about 1% to 2%. Based on the reported average speed and the penetration rate, we obtain the approximate value of the required minimum spacing (800 ft.) from the empirical result graph as shown in figure 5. At the on-ramp, we define a 1670 ft (500 meters) exclusion area. Given the fixed exclusion area, we generated different sets of equidistant trip lines with minimum spacing varying from 333 ft (100 meters) to 1670 ft (500 meters).

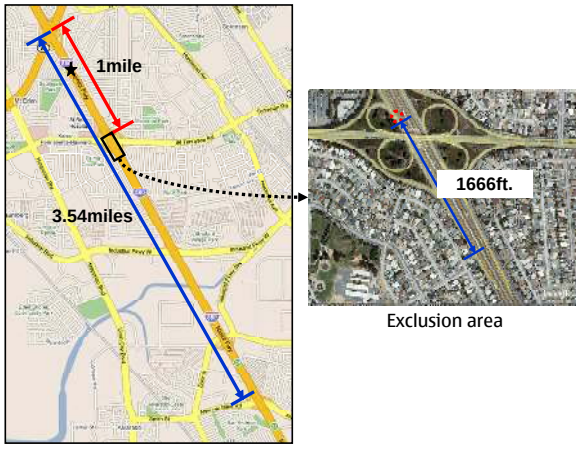
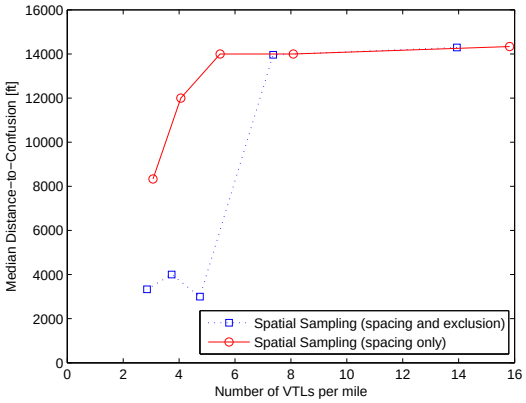


Figure 13: Exclusion Area on Test Road Segment. Tracking starts from the point marked by star.

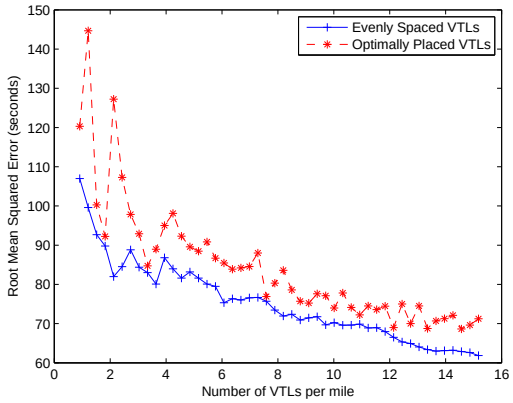
When we measure the *distance-to-confusion*, we use an uncertainty threshold of 0.2, meaning that tracking stops when it incorrectly links updates from different handsets, or when the uncertainty at any step rises above this threshold. We choose the probe vehicles of the main route as the test vectors. Among the set of anonymous updates that are reported at the same VTL, we measure the time deviation of each of them from the projected arrival time of the target probe vehicle, then we calculate the entropy using the empirically obtained probability distribution function of the time deviation between the projected arrival time and each timestamps of anonymous updates at the corresponding VTL. This empirical pdf was measured from the PARAMICS traffic dataset that have similar average speed and traffic volume. In linking anonymous flow updates that the spatial periodic sampling techniques generate, the adversary removes from the candidate set several anonymous flow updates that have speed measurements less than 40 mph within the 500 meter distance from the on-ramp, because an adversary has a knowledge on general trend around on-ramps as shown in figure 6(a). This leads to better tracking performance by reducing the number of likely hypotheses.

The results are shown in figure 14(a) which plots the median *distance-to-confusion* against the total number of anonymous flow updates for each case. The dotted curve shows the VTL-based placement cases, 1666, 1333, 1000, 666, and 333 feet from the left to the right. The solid curve shows the spatial periodic sampling techniques for the same spacings. We observe that the dotted line drops at spacing of 1000 feet. As we expect from the figure 5, two successive anonymous updates that are sampled longer than

[4]Each VTL is placed in the middle of its respective link and the conditions on the entire link are given by the VTL reading.



(a) Spatial Sampling (exclusion and spacing vs. spacing only)



(b) Evenly spaced vs. Optimally placed

Figure 14: Comparison of privacy and travel time accuracy over different VTL spacings. Spatial sampling with exclusion zones better preserves location privacy.

800 feet apart experience high tracking uncertainty. Another major reason for the drop in the curve is the existence of the exclusion area. The anonymous updates from the merging traffic can cause high uncertainty outside the exclusion area since the speed measurements look similar to those from the main route traffic. Note that the periodic sampling in time behaves similarly as the spatial periodic sampling by increasing a time interval, but it cannot support the location awareness in sampling. Thus, the location-aware sampling via trip lines better preserves privacy than the periodic sampling in time. Also, the spatial sampling based on trip lines naturally perturbs position and timestamp information since the reported measurements are actually sampled when probe vehicles already pass the position of trip lines. This noisy measurement also can cause the reduction of distance to confusion in high penetration rate.

To study the traffic flow estimation accuracy tradeoff incurred by larger VTL spacings, figure 14(b) shows the root-mean-square error over the same range of VTL spacings. The travel time estimation generally improves with an increasing number of VTLs, both for evenly spaced and for optimally placed VTLs. In particular, one can see that the error from optimally placed VTLs from the DP algorithm is lower than the naive approach of evenly spaced VTLs. Because of the previously mentioned shearing present on

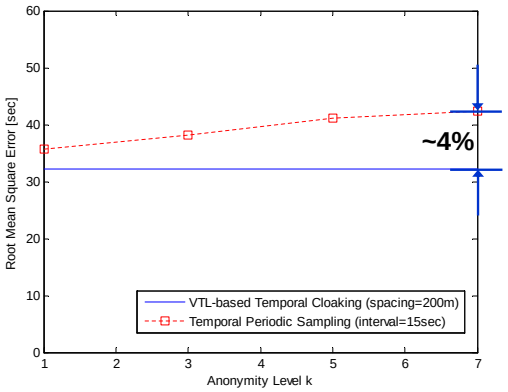


Figure 15: Travel Time Accuracy versus Anonymity k.

the highway, the error of travel time estimation algorithms, and the variability of the drivers themselves, there will always be some error present in travel time which cannot be removed by increasing the number of VTLs alone. Note that this variability is present in figure 12, where we plot actual travel times versus predicted travel times using the instantaneous method earlier. The graph is not monotonically decreasing because travel time error is inherently variable and sometimes by adding a VTL we actually produce worse estimates if the additional VTL does not precisely capture the correct speed for the long link that it covers.

7.3 Guaranteed Privacy via VTL-based Temporal Cloaking

To compare the travel time accuracy of temporal cloaking with that of a baseline temporal periodic sampling techniques, we measure the RMS error between estimated travel times and the actual travel times that are collected from 20 probe vehicles over the shorter route (north bound) in figure 10. The mean of actual travel time over this shorter route is 265.13 seconds and its 95% confidence interval is $(254.9; 275.3)$.

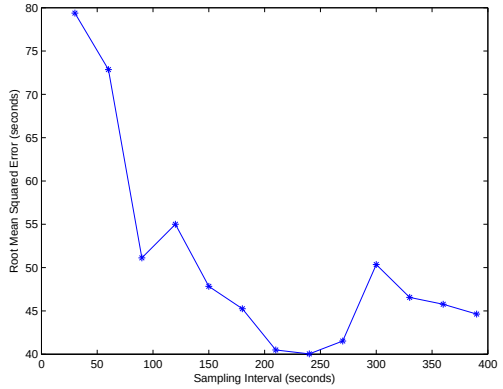


Figure 16: Travel time estimate errors by different aggregation intervals using 15 VTLs.

The travel time estimator divides the road into 200m intervals and separately calculates mean speed based on the periodic or VTL reports for each segment using an aggregation interval of a 200s window (parameters empirically chosen to provide good performance from figure 16). Our temporal cloaking method evenly place

a trip line every 200 meters without an exclusion area. The sampling interval of 15 seconds for the periodic sampling technique is chosen so that both the VTL approach and periodic sampling produce the same number of updates. This allows us to compare both techniques based on similar input information and network overhead. Figure 15 shows that temporal cloaking reduces accuracy by only up to 4% for a k value of 7.

Larger k-anonymity parameters lead to longer aggregation intervals. Figure 16 illustrates the effect on the quality of the travel time estimates in terms of the length of the aggregation interval. During the aggregation interval, we sum anonymous flow updates for the corresponding trip lines to calculate the average speed of the link that is specified by the trip line. The general trend is that a longer sampling interval provides more accurate travel time estimates. However, the RMS error increases again after 250 seconds, indicating that the aggregation interval should be shorter than the changing period of traffic conditions.

8. DISCUSSION

In addition to the privacy benefits, a key advantage of virtual trip lines over physical traffic sensors is the flexibility with which they can be deployed. For example, when roadwork is performed, VTLs can be deployed throughout the construction region, providing accurate travel time estimates in an area which often creates significant congestion. Because there is almost no additional cost to deploy the VTLs, and they do not interfere with the construction work or the highway traffic, they can be placed to adjust to the temporarily changed traffic patterns. One could even envision a VTL placement strategy which changes on a much shorter time period, with optimal placement strategies for the morning and evening rush hours, or holiday traffic patterns.

8.1 Security

This system significantly improves privacy protection over earlier proposals, by distributing the traffic monitoring functions among multiple entities, none of which have access to both location and identity records.

The system protects privacy against passive attacks under the assumption that only a single infrastructure component is compromised. One passive attack that remains an open problem for further study is timing analysis by network eavesdroppers or by the ID Proxy. Given knowledge of the exact trip line locations, which every handset could learn over time, and public travel time information on the road network an adversary could estimate the time needed to travel between any two trip lines. The adversary could then attempt to match a sequence of observed VTL update message inter-arrival times to these trip line locations. One may expect that the natural variability of driving times provides some protection against this approach. Protection could be further strengthened against network eavesdroppers by inserting random message delays. Under the temporal cloaking scheme, however, the ID proxy also obtains trip line identifier information. If they are used for extended durations, an adversary may match them to actual VTL positions based on the sequence in which probe vehicles have pass them. This threat can be alleviated through frequent VTL ID updates. Quantifying these threats and choosing exact tile size and update frquency parameters to balance privacy and network overhead concerns remain open research problems.

The system also protects the privacy of most users against active attacks that compromise a single infrastructure component and a small fraction of handsets. It does not protect user privacy against injecting malware directly onto users' phones, which obtains GPS readings and transfers them to an external party. This challenge remains outside of the scope of this paper, because this vulnerability is present on all networked and programmable GPS devices even without the use of a traffic monitoring system. Instead, the objective of the presented architecture is to limit the effect of such compromises on other phones. For the temporal cloaking approach, compromised phones result in two concerns. First, an adversary at the ID proxy can learn the current temporary trip line IDs. To limit the effectiveness of this attack, the architecture periodically changes trip lines and verifies the approximtate location of each phone so that a tile of trip line updates can only be sent to phones in the same location. Second, a handset could spoof trip line updates at a certain location to limit the effectivness of temporal cloaking. Our proposed architecture already eliminates updates from unauthorized phones and can easily limits the update rate per phone and verify that the approximate phone position matches the claimed update. This renders extended tracking of individual difficult because it would either require a large number of compromised phones spread around the area in which the individul moves, or set of compromised phones that move together with the indivdual. The system could also incorporate other sanity checks and blacklist phones that deliver suspicious updates.

The same methods also offer protection against spoofing attacks that seek to reduce the accuracy of traffic monitoring data. The system does not offer full protection against any active attack on traffic monitoring accuracy, however. For example, a compromised ID proxy could drop messages to reduce accuracy. These challenges remain an open problem for further work.

8.2 Involvement of Cellular Networks Operators

While this work was based on cellular handsets, the question of how to improve location privacy within cellular networks themselves is out of scope of this work. The Phase II E911 requirements [4] mandate that cellular networks can locate subscriber phones within 150-300m 95% of the time, and AGPS solutions could achieve similar accuracy as the traffic monitoring system on open-sky roadways. In addition, the phones are identified through IMSI (International Mobile Subscriber Identity, in the GSM system) and operators typically know their owner's names and addresses. While precise phone location information is accessible, to our knowledge, it is not widely collected and stored by operators at this level of accuracy.

To slow the further proliferation of such data, this work has investigated how traffic monitoring services can be offered without access to sensitive location information. It was primarily motivated by third party organizations that currently do not yet have access to identity and location information and want to implement privacy-preserving traffic monitoring services. The solution, however, is general enough so that in actual implementations different levels of involvement of network operators are possible. One case may be four separate organizations implementing each one infrastructure component of the system with no involvement of the network operator.[5] Another extreme case would be a cellular network operator creating separate entities within the company to protect itself against dishonest insiders and accidental data breaches of their customers records. Clearly, the first would be more preferable from a privacy perspective, but in the end both lead to a significant improvement in privacy over a naive implementation.

[5]The only limitation is that for temporal cloaking one of the identities needs to be able to approximately (at the level of a tile size) verify client location claims. This verification could be provided by a network operator but other forms of verification are also plausible.

9. RELATED WORK

Traffic monitoring applications based on a large number of probe vehicles have recently received much attention. The MIT CarTel project [38] proposed to use the unused bandwidth of open wireless hotspots to deliver sensor information, such as GPS-based location and speed measurements of probe vehicles, to a central server for traffic data processing. Previous studies using cell phone based traffic monitoring [17], [55], [57] investigate the use of trilateration-based positioning technology to locate phones, and because of the lower accuracy position estimate (100m accuracy), vehicle speeds could not be consistently determined. Yoon et al. [58] propose to use cellular network as a delivery method of GPS-based sensing information from probe vehicles. All these works have not addressed the location privacy concerns in such systems.

Since most traffic monitoring applications that have been proposed so far do not depend on the specific identification information about probe vehicles, the anonymization of sensing information has been a solution in practical deployments [8, 9, 7]. Not surprisingly, recent analyses of GPS traces [44, 36, 31] have shown that naive anonymization by simply omitting identifiers from location dataset does not guarantee anonymity. These studies use multi-target tracking [52] and k-means clustering [44, 36] algorithms to re-identify individuals from these traces. They exploit unique parts of the GPS traces, analogues to hardware and software fingerprinting approaches to identify computer systems [43].

Therefore, several stronger protection mechanisms have been investigated. The k-anonymity concept [53, 56] provides a guaranteed level of anonymity for a database, although some recent studies [46, 40] have identified weaknesses. For location services, the k-anonymity concept has lead to the development of centralized architectures that temporally and spatially cloak location-based queries [30, 48, 27]. The proposed schemes use a trustworthy proxy server to automatically determine the sizes of cloaked regions and cloaked time windows to achieve k-anonymity. This present work, in comparison, concentrates on providing privacy without requiring a single trustworthy entity.

There are many best effort approaches [14, 54, 45, 35, 15] that degrade information in a controlled way before releasing it. These approaches can be implemented in a centralized architecture or a decentralized approach. Many best effort approaches successfully preserve the privacy of users in high density areas, but they do not guarantee the privacy regardless of user density and user behavior pattern. Hoh et al. [37] propose the uncertainty-aware path cloaking algorithm to provide the guaranteed privacy regardless of user density, but this again requires the existence of a trustworthy privacy server.

Anonymous communication systems (e.g., onion routing [29, 28], tor [23]) use a similar approach of distributing knowledge over several mixes. Random perturbation approaches for privacy-aware data mining [11, 10], which perturbs the collected inputs from users to preserve privacy of data subjects while maintaining the quality of data, are not applicable for time-series location data since noise with large variance does not preserve sufficient data accuracy, while noise with small variance may be filtered by tracking algorithms due to the spatio-temporal nature of the data [41]. Access control methods [26, 59] restrict access to data to permitted users. However, these techniques do not fully address the dishonest insider challenge. Further, they are not applicable to business models where the aggregated data is transferred to third party.

Also, Hengartner [33] proposes to use private information retrieval techniques to prevent the service provider from being aware of location-based query. Also, Zhong et al. [60] issued an interesting problem, the nearby-friend problem which only lets users to share their location information when they are really nearby each other. In their implementation, they do not use a separate service provider that is aware of user's locations. Annavaram et al. [12] developed *HangOut*, a social networking application that shows where people with similar interests are likely to congregate. In HangOut, the mobile device decides on the granularity of its location update based on how many other users are already seen by the server in a given area. Furthermore, Hangout implements a function of separating anonymous data aggregation and authentication over different entities (as introduced in [36]). The same goal can be achieved by other schemes based on cryptography [18, 16]. However, those techniques still have not answered to the question of preventing algorithms from reconstructing private information from anonymous database samples.

10. CONCLUSIONS

This paper described an automotive traffic monitoring system implemented on a GPS smartphone platform. The system uses the concept of virtual trip lines to govern when phones reveal a location update to the traffic monitoring infrastructure. It improves privacy, through a system design that separates identity- and location-related processing, so that no single entity has access to both location and identity information. Virtual trip lines can be easily omitted around particularly sensitive locations. Virtual trip lines also allow the application of temporal cloaking techniques to ensure k-anonymity properties of the stored dataset, without having access to the actual location records of phones. We demonstrate the feasibility of implementing this system on a smartphone platform and conduct a 20 vehicle experiment on a highway segment. We show that the privacy techniques lead to less than 5% reduction in the accuracy of travel time estimates for k values less than 7.

Acknowledgement

This study was supported in part by the US National Science Foundation under grant CNS-0524475. The experiment and development of the optimal placement algorithm were supported by the California Department of Transportation (Caltrans). We thank the reviewers and our shepherd for their valuable comments, which improved the paper in a number of ways. We also would like to thank the twenty drivers that voluntarily participated in the experiment.

11. REFERENCES

[1] http://bard.wr.usgs.gov/.
[2] http://jcp.org/en/jsr/detail?id=179/.
[3] http://pems.eecs.berkeley.edu/public/.
[4] http://www.fcc.gov/bureaus/wireless/.
[5] http://www.paramics-online.com.
[6] http://www.privacyrights.org/ar/chrondatabreaches.htm.
[7] TeleNav. http://www.telenav.net/, 2004.
[8] Inrix. http://www.inrix.com/, 2006.
[9] Intellione. http://www.intellione.com/, 2006.
[10] D. Agrawal and C. C. Aggarwal. On the design and quantification of privacy preserving data mining algorithms. In *Symposium on Principles of Database Systems*, 2001.
[11] R. Agrawal and R. Srikant. Privacy-preserving data mining. In *Proc. of the ACM SIGMOD Conference on Management of Data*, pages 439–450. ACM Press, May 2000.
[12] M. Annavaram, Q. Jacobson, and J. P. Shen. Hangout: A privacy preserving social networking application. In *Proc. of International Workshop on Mobile Devices and Urban Sensing (to appear)*, St. Louis, USA, 2008.
[13] X. Ban, R. Herring, J. Margulici, and A. Bayen. Optimal sensor placement for freeway travel time estimation. Interim technical report (available at `http://www.calccit.org/resources/2008-pdf/OptSenDP_InterimReport_2008.pdf`, California Center for Innovative Transportation, 2008. Revised version to be submitted to the 18th International Symposium on Traffic and Transportation Theory (ISTTT).

[14] A. Beresford and F. Stajano. Mix zones: User privacy in location-aware services. In *IEEE PerSec*, 2004.

[15] C. Bettini, X. SeanWang, and S. Jajodia. Protecting privacy against location-based personal identification,. In *2nd VLDB Workshop SDM*, 2005.

[16] D. Boneh, X. Boyen, and H. Shacham. Short group signatures. In *CRYPTO 2004*, volume 3152 of *Lecture Notes in Computer Science*, pages 41–55. Springer, 2004.

[17] R. Cayford and T. Johnson. Operational parameters affecting the use of anonymous cell phone tracking for generating traffic information. *Transportation Research Board 82nd Annual Meeting*, 1(3):03–3865, 2003.

[18] D. Chaum and E. V. Heyst. Group signatures. In *Advances in Cryptology–EUROCRYPT '91*, volume 547 of *Lecture Notes in Computer Science*, pages 257–265. Springer, 1991.

[19] L. Chu, S. Oh, and W. Recker. Adaptive Kalman filter based freeway travel time estimation. In 84^{th} *TRB Annual Meeting*, Washington D.C., January 9-13 2005. Transportation Research Board.

[20] B. Coifman. Using dual loop speed traps to identify detector errors. *Transportation Research Board*, Transportation Research Record 1683(-1):47–58, 1999.

[21] B. Coifman. Improved velocity estimation using single loop detectors. *Transportation Research Part A*, 35(10):863–880, 2001.

[22] X. Dai, M. Ferman, and R. Roesser. A simulation evaluation of a real-time traffic information system using probe vehicles. In *Proceedings of the IEEE Intelligent Transportation Systems*, pages 475–480, 2003.

[23] R. Dingledine, N. Mathewson, and P. F. Syverson. Tor: The second-generation onion router. In *USENIX Security Symposium*, pages 303–320, 2004.

[24] A.-M. Elliott. Tomtom announces tomtom high definition traffic. http://www.pocket-lint.co.uk/news/news.phtml/11248/12272/TomTom-High-Definition-Traffic-announced.phtml, Nov 2007.

[25] M. Ferman, D. Blumenfeld, and X. Dai. A simple analytical model of a probe-based traffic information system. In *Proceedings of the IEEE Intelligent Transportation Systems*, pages 263–268, 2003.

[26] A. Gal and V. Atluri. An authorization model for temporal data. In *Proceedings of the 7th ACM CCS*, pages 144–153, New York, NY, USA, 2000. ACM Press.

[27] B. Gedik and L. Liu. Location privacy in mobile systems: A personalized anonymization model. In *Proceedings of the 25th IEEE ICDCS 2005*, pages 620–629, Washington, DC, USA, 2005.

[28] D. Goldschlag, M. Reed, and P. Syverson. Onion routing. *Communications of the ACM*, 42(2):39–41, 1999.

[29] D. Goldschlag, M. Reed, and P. Syverson. Onion routing for anonymous and private internet connections. *Communications of the ACM (USA)*, 42(2):39–41, 1999.

[30] M. Gruteser and D. Grunwald. Anonymous usage of location-based services through spatial and temporal cloaking. In *Proceedings of the ACM MobiSys*, 2003.

[31] M. Gruteser and B. Hoh. On the anonymity of periodic location samples. In *Proceedings of the Second International Conference on Security in Pervasive Computing*, 2005.

[32] F. Hall and B. Persaud. Evaluation of speed estimates made with single-detector data from freeway traffic management systems. *Transportation Research Board*, Transportation Research Record 1232:9–16, 1989.

[33] U. Hengartner. Hiding location information from location-based services. In *Proc. of International Workshop on Privacy-Aware Location-based Mobile Services (PALMS)*, Mannheim, Germany, 2007.

[34] J. Herrera and A.M.Bayen. Traffic flow reconstruction using mobile sensors and loop detector data. In 87^{th} *TRB Annual Meeting*, Washington D.C., January 12-16 2008. Transportation Research Board.

[35] B. Hoh and M. Gruteser. Protecting location privacy through path confusion. In *Proceedings of IEEE/Create-Net SecureComm*, Athens, Greece, September 2005.

[36] B. Hoh, M. Gruteser, H. Xiong, and A. Alrabady. Enhancing security and privacy in traffic-monitoring systems. *IEEE Pervasive Computing*, 5(4):38–46, 2006.

[37] B. Hoh, M. Gruteser, H. Xiong, and A. Alrabady. Preserving privacy in gps traces via uncertainty-aware path cloaking. In *Proceedings of ACM CCS 2007*, October 2007.

[38] B. Hull, V. Bychkovsky, Y. Zhang, K. Chen, M. Goraczko, A. K. Miu, E. Shih, H. Balakrishnan, and S. Madden. CarTel: A Distributed Mobile Sensor Computing System. In *4th ACM SenSys*, Boulder, CO, November 2006.

[39] Z. Jia, C. Chen, B. Coifman, and P. Varaiya. The PeMS algorithms for accurate, real-time estimates of g-factors and speeds from single-loop detectors. *IEEE Control Systems Magazine*, 21(4):26–33, 2001.

[40] P. Kalnis, G. Ghinita, K. Mouratidis, and D. Papadias. Preventing location-based identity inference in anonymous spatial queries. *IEEE Transactions on Knowledge and Data Engineering*, 19(12):1719–1733, 2007.

[41] H. Kargupta, S. Datta, Q. Wang, and K. Sivakumar. Random data perturbation techniques and privacy preserving data mining. In *IEEE ICDM*. IEEE Press, 2003.

[42] L. Klein, M. Mills, and D. Gibson. *Traffic Detector Handbook*, volume 1 and 2. Third edition, October 2006.

[43] T. Kohno, A. Broido, and K. C. Claffy. Remote physical device fingerprinting. In *SP '05: Proceedings of the 2005 IEEE Symposium on Security and Privacy*, pages 211–225, Washington, DC, USA, 2005. IEEE Computer Society.

[44] J. Krumm. Inference attacks on location tracks. In *Proceedings of the 5th International Conference on Pervasive Computing (Pervasive 2007)*, May 2007.

[45] M. Li, K. Sampigethaya, L. Huang, and R. Poovendran. Swing & swap: user-centric approaches towards maximizing location privacy. In *Proceedings of the 5th ACM WPES '06*, pages 19–28, New York, NY, USA, 2006. ACM Press.

[46] A. Machanavajjhala, J. Gehrke, D. Kifer, and M. Venkitasubramaniam. l-diversity: Privacy beyond k-anonymity. In *Proceedings of the 22nd International Conference on Data Engineering (ICDE'06)*, page 24, 2006.

[47] B. Mikhalkin, H. Payne, and L. Isaksen. Estimation of speed from presence detectors. *Highway Research Record*, 388:73–83, 1972.

[48] M. F. Mokbel, C.-Y. Chow, and W. G. Aref. The new casper: query processing for location services without compromising privacy. In *Proceedings of the 32nd VLDB'2006*, pages 763–774. VLDB Endowment, 2006.

[49] C. Nanthawichit, T. Nakatsuji, and H. Suzuki. Application of probe-vehicle data for real-time traffic-state estimation and short-term travel-time prediction on a freeway. *Transportation Research Record*, 1855:49–59, 2003.

[50] H. Payne and S. Thompson. Malfunction detection and data repair for induction-loop sensors using i-880 data base. *Transportation Research Board*, Transportation Research Record 1570(-1):191–201, 1997.

[51] A. Pushkar, F. Hall, and J. Acha-Daza. Estimation of speeds from single-loop freeway flow and occupancy detectors using cusp catastrophy theory model. *Transportation Research Board*, Transportation Research Record 1457:149–157, 1994.

[52] D. Reid. An algorithm for tracking multiple targets. *IEEE Transactions on Automatic Control*, 24(6):843–854, Dec 1979.

[53] P. Samarati and L. Sweeney. Protecting privacy when disclosing information: k-anonymity and its enforcement through generalization and suppression. In *Proceedings of IEEE Symposium on Research in Security and Privacy*, 1998.

[54] K. Sampigethaya, L. Huang, M. Li, R. Poovendran, K. Matsuura, and K. Sezaki. Caravan: Providing location privacy for vanet. In *3rd workshop on Embedded Security in Cars (ESCAR2005)*, 2005.

[55] B. Smith, H. Zhang, M. Fontaine, and M. Green. Cell phone probes as an ATMS tool. Research Report UVACTS-15-5-79, June 2003.

[56] L. Sweeney. Achieving k-Anonymity Privacy Protection Using Generalization and Suppression. *International Journal on Uncertainty, Fuzziness and Knowledge-based Systems*, 10(5):571–588, 2002.

[57] U. o. M. Transportation Studies Center. *Final Evaluation Report for the CAPITAL-ITS Operational Test and Demonstration Program*. Transportation Studies Center, University of Maryland, 1997.

[58] J. Yoon, B. Noble, and M. Liu. Surface street traffic estimation. In *MobiSys '07: Proceedings of the 5th international conference on Mobile systems, applications and services*, pages 220–232, New York, NY, USA, 2007. ACM.

[59] M. Youssef, V. Atluri, and N. R. Adam. Preserving mobile customer privacy: an access control system for moving objects and customer profiles. In *Proceedings of the 6th MDM '05*, pages 67–76, New York, NY, USA, 2005. ACM Press.

[60] G. Zhong, I. Goldberg, and U. Hengartner. Louis, lester and pierre: Three protocols for location privacy. In *Privacy Enhancing Technologies*, pages 62–76, 2007.