Privacy Preserving Origin-Destination Flow Measurement in Vehicular Cyber-Physical Systems

Yian Zhou † Shigang Chen † Zhen Mo † Yafeng Yin ‡
†Department of Computer & Information Science & Engineering
‡Department of Civil and Coastal Engineering
University of Florida, Gainesville, FL 32611, USA

***Abstract*—Traffic volume measurement is one of the most basic functions of road planning and management. In this paper, we investigate an important problem of privacy preserving "point-to-point" traffic volume measurement. We formalize "point-to-point" traffic as an origin-destination (O-D) flow, which represents the set of vehicles traveling from one geographical location (origin) to another location (destination). We take advantage of vehicular cyber-physical systems (VCPS) to exploit the potential for a fundamental shift in the way how O-D data are collected. The challenge is to allow the collection of statistical O-D flow information, yet protect identities of individual vehicles. To address that, we design two novel schemes which utilize both the latest technological advance in VCPS and the nice properties of a family of commutative one-way hash functions. Furthermore, we adopt statistical methodology and use sampling to achieve far better efficiency with graceful degradation in measurement accuracy. We perform simulations to demonstrate the feasibility and scalability of our schemes.**

I. INTRODUCTION

Traffic volume measurement is one of the most basic functions of road planning and management. Today the widely used traffic statistic is annual average daily traffic (AADT) [1], which describes the number of vehicles traversing a specific *point* in the road system annually. Although AADT is very useful, it is only *"point"* information. To gain better understanding of road usage, we need *"point-to-point"* statistics that measure traffic volumes between distinct locations. Prior research has made steady advance in estimation of *"point"* statistics like AADT (e.g. [2], [3], [4], [5]). While point-to-point statistics may be inferred from point data [6], little work has been done on direct measurement of *"point-to-point"* traffic volume particularly when drivers' location privacy is of concern.

In this paper, we investigate the problem of privacy preserving *point-to-point* traffic volume measurement. We formalize point-to-point traffic as an origin-destination (O-D) flow, whose size is the number of vehicles traveling from one geographical location (origin) to another (destination). Like AADT, O-D flow data is an essential input to a variety of studies including estimation of transportation link flow distribution as part of investment planning, calculation of road exposure rates as part of safety analysis, and characterization of turning movements at intersections for signal timing determination, etc. However, very few techniques have been developed to directly measure O-D data, not to mention preserving traveler's privacy at the same time.

Vehicular cyber-physical systems (VCPS) utilize the latest technologies in wireless communications, on-board computer processing, sensors, GPS navigation, etc., to improve safety, efficiency, and resiliency of transportation systems [7] [8]. For example, IntelliDrive [9] from USDOT [10] envisions a nationwide system where vehicles communicate with roadside equipments (RSE) in real time via dedicated short range communications (DSRC). VCPS provides the potential for a fundamental shift in how O-D data are collected: When a vehicle passes by an RSE, it can report its unique ID (e.g., vehicle identification number or VIN). The O-D flow between two RSEs is simply the set of common IDs stored in them. However, this straightforward approach leads to serious privacy breaching as it also tracks the entire moving history of vehicles, which is against the "anonymity by design" principle for privacy protection required by IntelliDrive. Hence, the challenge is to allow the collection of statistical O-D flow data, yet protect information about individual vehicle.

The objective of our work is to allow transportation authorities to collect aggregate O-D flow data from VCPS without learning information about individual vehicles. Since globally unique IDs like VINs and other permanent or temporary numbers that are transmitted repeatedly by a vehicle can be exploited for the tracking purpose, IDs (or other fixed numbers) should be preprocessed and protected by keys before transmission. In other words, RSEs will only be able to collect Keyed signatures of vehicles' IDs (referred to as KIDs). To measure O-D flow sizes, we introduce a family of commutative one-way hash functions, and propose two novel O-D measurement schemes, which can protect the identities of vehicles. The first scheme is more efficient, but it is vulnerable to an identical-key attack. The second scheme prevents this attack at the cost of increased computation overhead. To make it practical, we adopt statistical methods with sampling to construct a maximum likelihood estimation formula for the O-D flow size. The sampling can gracefully control the tradeoff between computation efficiency and measurement accuracy. We perform simulations, and the results demonstrate the feasibility and scalability of our schemes.

II. PRELIMINARIES

A. System Model

We consider a vehicular cyber-physical system (VCPS) model involving three entities: vehicles, roadside equipments

978-1-4799-0798-4/13/$31.00 ©2013 IEEE

(RSE), and a central server. Each vehicle has a unique ID, e.g., its VIN or other number chosen permanently or temporarily. We assume that each vehicle randomly picks its ID (from a large space) at the beginning of a day. The set of RSEs is denoted as $S = \{s_1, s_2, ..., s_N\}$. Both vehicles and RSEs are equipped with computing and communication capabilities, such as on-board computer chips and communication modules. Vehicles communicate with RSEs in real time via dedicated short range communications (DSRC) [10]. RSEs are connected to the central server through wired or wireless means. They collect information from vehicles and transfer it at the end of each measurement period (such as a day) to the central server for further processing.

B. Problem Statement

We define an origin-destination (O-D) flow as the set of vehicles traveling between one RSE-equipped location (origin) and another RSE-equipped location (destination) during a measurement period. [1] The O-D flow size is the number of vehicles in the set. The problem is to design a privacy-assured scheme that measures the sizes of O-D flows in a road system between all pairs of origin/destination locations where RSEs are installed. To protect the identities of vehicles, we need a solution in which a vehicle never transmits its ID or any fixed number. Ideally, the information transmitted by a vehicle to any RSE is different each time and looks totally random.

An alternative approach is having the RSEs broadcast their IDs. Each vehicle will record the IDs of the RSEs that it has passed by, and transmit them to every RSE that it passes in the future. But this is not a good solution, because the vehicle is giving its trajectory (i.e., the driver's entire moving history).

We assume that a special MAC protocol is used to support privacy preservation such that the MAC address of a vehicle is not fixed. For instance, when responding to an RSE, the vehicle may pick an MAC address randomly from a large space for one-time use. Since the number of vehicles in the vicinity of the RSE is limited, the probability for two vehicles to choose the same MAC address can be made negligibly small when the address space is sufficiently large.

C. Threat Model

We use a semi-trust model for the RSEs. We assume that all RSEs are from trustworthy authorities. This assumption can be enforced by authentication based on PKI. Each vehicle is pre-installed with the public keys of the trusted third parties. Each RSE must have a public-key certificate from them. It broadcasts the certificate in each query that it sends out. When receiving a query, the vehicle verifies the certificate, and then uses the RSE's public key to authenticate it. We also assume that the authorities may exploit the information collected by RSEs to track individual vehicles when they need to do so. For instance, if a vehicle transmits any fixed number upon each query, that number can be exploited for tracking purpose.

It is important to note that there are many other ways to track a vehicle, for example, tailgating the vehicle, or setting cameras near RSEs to take photos and using image processing to recognize it. These methods are beyond the scope of this paper. We focus on preventing automatic real-time tracking caused by vehicle identity leakage via RSEs.

D. Design Goals

To enable privacy preserving O-D flow measurement under the aforementioned model, our scheme should achieve the following design goals.

1) Correctness: the scheme should correctly measure the O-D flow size for arbitrary pair of RSEs, or with a measurement error that is probabilistically bounded.
2) Privacy guarantee: the proposed scheme should be able to protect the identity information of vehicles from unauthorized leakage and inference.
3) Efficiency: the scheme should have means to control its overhead for scaling to a large road system.

III. SOLUTION USING COMMUTATIVE ONE-WAY HASH FUNCTIONS

In this section, we propose a solution for privacy preserving O-D flow measurement based on a family of commutative one-way hash functions (COHF). A common COHF is deployed to all RSEs and vehicles, and vehicles apply the hash function to produce Keyed signatures of their IDs (referred to as KIDs) using the keys obtained from RSEs that they pass by. The KIDs, instead of real IDs, are reported to RSEs for O-D flow measurement. Before describing the full solution, we first introduce the family of commutative one-way hash functions.

A. Commutative One-Way Hash Functions

Consider a hash function $h : A \times B \to C$, where the two arguments are a hash input and a hash key, respectively. A commutative one-way hash function, as its name suggests, satisfies both one-wayness and commutativity. The definitions of the properties below are collated from [11] and [12].

Definition 1: A family of *one-way hash functions (OHF)* is a set of functions $h_n : V_n \times K_n \to Z_n$, which satisfy the following three properties:

- *Ease of computation*: there exists a polynomial P such that for each integer n, $h_n(v, k)$ is computable in time $P(n, |v|, |k|)$ for all $v \in V_n$ and all $k \in K_n$.
- *Preimage resistance*: there is no polynomial P such that, given n, $k \in K_n$, and $z \in Z_n$, there exists a probabilistic polynomial time algorithm which can find $v \in V_n$ satisfying $h_n(v, k) = z$ with probability greater than $1/P(n)$ for sufficiently large n, when k is chosen uniformly from K_n and z is chosen uniformly from Z_n.
- *2nd-preimage resistance*: there is no polynomial P such that, given n, $(v, k) \in V_n \times K_n$, and $k' \in K_n$, there exists

[1] Our proposed solutions can handle both directional O-D flows (from a location x to a location y) and undirectional O-D flows (including two directional flows from x to y and from y to x). For an O-D flow between location x and location y, our definition includes all vehicles that pass both locations, comparing with a narrower definition often used in the transportation literature that includes only vehicles starting their trips from x and ending their trips at y.

a probabilistic polynomial time algorithm which can find $v' \in V_n$ satisfying $h_n(v,k) = h_n(v',k')$ with probability greater than $1/P(n)$ for sufficiently large n, when (v,k) is chosen uniformly among all elements of $V_n \times K_n$ and k' is chosen uniformly from K_n.

In this case, h_n is said to have the *one-wayness* property.

In Definition 1, the first property requires that OHF is relatively easy to compute (in polynomial time). The second property requires that it is computationally infeasible to find an input which can be hashed to an arbitrarily pre-specified output. The third property requires that it is computationally infeasible to find a second input that can be hashed to the same output as arbitrarily pre-specified input and key.

Definition 2: A *commutative hash function (CHF)* is a hash function $h_n : V_n \times K_n \rightarrow V_n$, which satisfies the *commutativity* property: for all $v \in V_n$ and for all $k, k' \in K_n$, $h_n(h_n(v,k),k') = h_n(h_n(v,k'),k)$.

One can see that commutativity lies in the hash keys: given any input and two keys, commutativity tells that changing the order in which the two keys are applied to the input won't change the hash result. Further observed, if the range of h_n equals the domain of its first argument, we can exploit a new family of commutative one-way hash functions which shall satisfy both one-wayness and commutativity.

Definition 3: Commutative one-way hash functions (COHF) are a family of hash functions which have both one-wayness property and commutativity property.

We will see shortly one crucial benefit of utilizing this hash function family: Vehicles can transmit their KIDs by hashing their IDs under totally different keys, and be sure that no one will be able to get their IDs, even knowing the keys used by the vehicles (one-wayness). Yet the KIDs allow O-D flow measurement as demanded (through commutativity).

B. The Proposed Scheme

Using the COHFs, we propose the following scheme for privacy preserving O-D flow measurement. Each measurement period consists of three phases: initialization, online reporting, and offline measurement. Before describing the three measurement phases, we first construct the COHFs.

1) Construction of Commutative One-Way Hash Functions: According to Definition 3, a COHF is a hash function that satisfies both one-wayness and commutativity. There can be different constructions of COHFs given different types of hash functions, and the one that we adopt is based on the exponentiation modulo n function, $h_n(v,k) = v^k \ mod\ n$. We claim that h_n is a COHF with some restrictions on n.

Definition 4: A prime p is defined to be *safe* if $p = 2p' + 1$ where p' is an odd prime. A number n is defined to be a *rigid integer* if $n = pq$ where p and q are distinct large safe primes.

Theorem 1: The function $h_n(v,k) = v^k \ mod\ n$ is a commutative one-way hash function if n is a rigid integer.

Proof: Because of space limitations, here we only give the proof skeleton. Clearly, h_n is commutative. As to one-wayness, h_n satisfies *ease-of-computation* since there are efficient methods to perform exponentiation of a base to an exponent in polynomial time (e.g., [13]). Note that the selection of n and h_n follows the RSA cryptosystem [14]. Therefore, the *preimage resistance* of h_n also follows the cryptographic security of RSA [15]. The third property, *2nd-preimage resistance*, is rooted in the characteristics of rigid integers. It is demonstrated in [12] that if n is a rigid integer, finding collisions with specific constraints (i.e., 2nd-preimage) cannot done in polynomial time. This completes the proof. □

2) Initialization: A common commutative one-way hash function h_n must be pre-distributed to all vehicles and RSEs. The hash function is determined by a large rigid integer n. There is a practical method to construct it, and the basic idea is that for $n = pq$ to be a rigid integer, each of p, q, $\frac{(p-1)}{2}$ and $\frac{(q-1)}{2}$ must be primes congruent to 5 modulo 6. Therefore, the process is to first select a "random" integer p' that is congruent to 5 modulo 6 until one is found such that p' and $2p' + 1$ are both prime, and then choose a suitable q' similarly. After that, n can be easily constructed by $n = pq = (2p' + 1)(2q' + 1)$.

All RSEs and vehicles are pre-configured with a suitable value of n, and clocks of RSEs are loosely synchronized as they are all connected to the central server through wired or wireless means. Every RSE generates a random number as its hash key for the current measurement period. With the server's assistance, all hash keys are unique: Let k_x be the hash key generated by RSE s_x. We require that, for any two RSEs s_x and s_y, their keys k_x and k_y be different. If the server finds two hash keys reported from RSEs are the same, it will inform one of them to regenerate a key. The key uniqueness requirement serves an important purpose, which will be explained later.

3) Online Reporting: The online reporting phase securely collects information for O-D flow measurement. The RSEs broadcast queries in pre-set intervals (e.g., once a second), ensuring that each passing vehicle receives at least one query and meanwhile giving enough time for the vehicle to reply. Collisions can be resolved through well-established CSMA or TDMA protocols, which are not the focus of this paper. Every query that an RSE sends out includes the RSE's ID, public-key certificate, as well as its current hash key. When a vehicle, whose ID is v, receives a query from an RSE s_x, it first verifies the certificate, and then uses the RSE's public key to authenticate the RSE. After verifying that s_x is from the trustworthy authority, the vehicle generates a KID based on its ID v and the RSE's key k_x by computing a hash $c = h_n(v, k_x) = v^{k_x} \ mod\ n$. After that, it reports the KID c to the RSE, which then stores c in its local storage.

4) Offline Measurement: At the end of each measurement period, the O-D flow sizes between pairs of RSEs are computed based on the KIDs collected by RSEs during the online reporting phase. Specifically, every RSE will send its key as well as the collected KID set to the central server, which will be in charge of the offline O-D flow size computation.

Thanks to the commutativity property of h_n, given two sets of KIDs, $H_x = \{h_n(\cdot, k_x)\}$ and $H_y = \{h_n(\cdot, k_y)\}$, collected by two RSEs s_x and s_y respectively, and the two corresponding keys, k_x and k_y, it is easy for the central server to determine the O-D flow size between s_x and s_y. In principle, changing the order in which two keys are applied to the same vehicle ID using COHFs won't change the final hash result. Therefore, the central server simply further hashes each RSE's KID set by the other RSE's key to obtain two double-hashed sets $H_{x,y} = \{h_n(h_n(\cdot, k_x), k_y)\}$ and $H_{y,x} = \{h_n(h_n(\cdot, k_y), k_x)\}$, and the O-D flow size between s_x and s_y simply equals the number of common elements in $H_{y,x}$ and $H_{x,y}$ according to Theorem 2 in the following. [2]

Theorem 2: Given a commutative one-way hash function $h_n(v, k) = v^k \, mod \, n$, for arbitrary vehicle IDs v and v', and arbitrary keys k and k', $h_n(h_n(v, k), k') = h_n(h_n(v', k'), k)$ holds if and only if $v = v'$ holds.

Proof: The sufficiency is clearly established given the commutativity of h_n. The necessity is granted through two facts. First, h_n is commutative. Second, since the number of vehicles in the vicinity of two RSEs is limited, and the hash space is sufficiently large, the probability for two distinct vehicle IDs to be hashed under the same key to the same value is negligibly small. This completes the proof. □

C. Scheme Analysis

The proposed scheme preserves vehicles' privacy. As vehicles only transmit their KIDs to RSEs, no one can obtain their real IDs thanks to the one-wayness of the COHF h_n. Vehicles are further protected from being tracked since no fixed information of them is transmitted because of the key uniqueness requirement. The scheme is also efficient. Each vehicle only needs to compute one hash for each passing RSE, so the time overhead for each vehicle is bounded by $O(N)$, where N is the number of RSEs. As for the central server, to compute an O-D flow size between two RSEs, it needs to perform a hash for each KID value from the two KID sets, so the number of hash operations is bounded by $O(M)$, where M is total number of vehicles. Further, to find the common double-hashed values, it needs to sort the two double-hashed sets, which takes $O(M \log M)$ comparison operations.

D. Identical-key Attack

The above analysis assumes the transportation authority (who owns RSEs and the central server) is trustworthy. But this assumption also allows the transportation authority an easy way of tracking vehicles. It may simply set all or a portion of RSEs with the same key. When a vehicle passes these RSEs, its KID stays the same and therefore may be exploited for tracking purpose. To avoid transmitting the same number (KID), a vehicle may keep record of the RSE keys that it has seen before, and will not respond to an RSE if the key from that RSE is already in the vehicle's record.

This solution however causes an under-measurement problem. Suppose during a measurement period (e.g., a day), a vehicle passes by an RSE for two or more times. This is not uncommon in reality. For example, people driving to work are likely to follow the same route back home. While the vehicle contributes twice to traffic volume between home and workplace, it is counted only once (since the vehicle does not respond to the same key). To fully address this concern, we need to make a shift in who is responsible for key generation. We shall move that responsibility from RSEs to the vehicles in order to ensure that the key uniqueness requirement is met.

IV. ENHANCED SCHEME FOR PRIVACY PRESERVING O-D FLOW MEASUREMENT

Instead of using the keys generated by RSEs, our second scheme lets vehicles choose their own keys to protect their IDs. Still, vehicles and RSEs are pre-configured with a common commutative one-way hash function h_n. RSEs will collect KIDs from vehicles, and a central server will compute O-D flow sizes based on the collected KID sets. The difference is that, RSEs will not just record the KIDs. Instead, it will store a set of ⟨key, KID⟩ pairs obtained from passing vehicles for measurement purpose. The enhanced scheme has two phases: online reporting and offline measurement.

A. The Enhanced Scheme

1) Online Reporting: During the online reporting phase, ⟨key, KID⟩ pairs are securely collected by RSEs from the passing vehicles. More specifically, when a vehicle v passes by an RSE s_x, the vehicle will first verify that the RSE comes from trusted authorities based on the public-key certificate received from the RSE's periodic broadcast. Then the vehicle will randomly choose a hash key k, and compute a hash $c = h_n(v, k) = v^k \; mod \; n$, which serves as a KID of v. After that, the vehicle reports the KID c and the key k to s_x, which stores this ⟨key, KID⟩ pair in its local storage.

2) Offline Measurement: At the end of each measurement period, all RSEs will send their collected data to the central server. Given two sets of ⟨key, KID⟩ pairs collected by two RSEs s_x and s_y, the central server can compute the size of the corresponding O-D flow based on the hash function h_n's commutativity. The process is to go through these two sets, and for each pair $\langle k_x, c_x \rangle$ collected by s_x, check if there is a pair $\langle k_y, c_y \rangle$ collected by s_y such that $h_n(c_y, k_x) = h_n(c_x, k_y)$; we say the two pairs share a common double-hashed value in this case. If so, a vehicle is found to pass both RSEs. One can easily verify its correctness through Theorem 2.

3) Scheme Analysis: The enhanced scheme eliminates the under-measurement problem that is encountered by the previous scheme. Even if a vehicle may pass an RSE for several times, each time it uses a different key to produce a new KID, which will be recorded and counted towards the final measurement result. Therefore, the measured O-D flow sizes should always be equal to the real ones. Observe that the

[2] Note that if we take the timestamps of the KIDs into consideration, we can easily determine the size of a directional O-D flow for vehicles that appear at s_x first and then appear at s_y at a later time.

enhanced scheme improves the measurement accuracy at the cost of increased computation overhead. In order to compute the O-D flow size between two RSEs, s_x and s_y, the central server needs to perform a re-hash for each pair collected by s_x under every key from s_y, and do the same thing for s_y. Suppose the two RSEs have collected n_x and n_y pairs of ⟨key, KID⟩, respectively. The time complexity for the central server to compute the corresponding O-D flow size will be $O(n_x \cdot n_y)$.

B. Sampling

To address the efficiency problem, we propose to use sampling to estimate the O-D flow sizes. Given two sets of ⟨key, KID⟩ pairs collected by two RSEs s_x and s_y, $D_x = \{\langle k_{ix}, c_{ix}\rangle\}_{i=1}^{n_x}$, $D_y = \{\langle k_{iy}, c_{iy}\rangle\}_{i=1}^{n_y}$, it takes $O(n_x \cdot n_y)$ time to calculate the O-D flow size. To reduce computation overhead, we randomly select n'_x elements from D_x and n'_y elements from D_y, denoting them as D'_x and D'_y, respectively. It only takes $O(n'_x \cdot n'_y)$ time to compute the O-D flow size n'_{xy} from such a sample. Based on n'_{xy} and the sampling probabilities, we can construct the maximum likelihood estimate (MLE) of n_{xy} as

$$\hat{n}_{xy} = n'_{xy} \times \frac{n_x}{n'_x} \times \frac{n_y}{n'_y}, \tag{1}$$

which is derived as follows: The idea is that if two pairs from D_x and D_y share a common double-hashed value, we treat them as a common element in these two sets. So our problem is equivalent to the set-intersection estimation problem: Let X and Y be two sets with $|X| = a$, $|Y| = b$, $|X \cap Y| = c$. We randomly choose two subsets of elements, X' and Y', with cardinalities a' and b', from X and Y. We find the number of common elements in X' and Y', denoted by c'. The problem is to construct the MLE of c based on c', a, b, a', and b'.

For a randomly selected $e \in X'$, the probability for $e \in X \cap Y$ is $\frac{c}{a}$. Under this condition $e \in X \cap Y$, the probability for $e \in Y'$ is $\frac{b'}{b}$. Combining them, we have $P(e \in Y' | e \in X') = \frac{cb'}{ab}$. There are a' elements in X', so the likelihood function for observing c' common elements in X' and Y' is

$$\mathcal{L} = (\frac{cb'}{ab})^{c'}(1 - \frac{cb'}{ab})^{a'-c'}. \tag{2}$$

We want to find the MLE of c, denoted as $\hat{c}$, which maximizes $\mathcal{L}$. To find $\hat{c}$, we take logarithm on both sides of (2):

$$\ln \mathcal{L} = c' \times \ln(\frac{cb'}{ab}) + (a' - c')\ln(1 - \frac{cb'}{ab}) \tag{3}$$

Take the first order derivative of (3) and let it be zero. We have $\hat{c} = c' \times \frac{a}{a'} \times \frac{b}{b'}$. By changing the notations to those for our problem, we have $\hat{n}_{xy} = n'_{xy} \times \frac{n_x}{n'_x} \times \frac{n_y}{n'_y}$, which is the MLE of n_{xy}. By adopting the sampling method, the computation overhead is reduced from $O(n_x \cdot n_y)$ to $O(n'_x \cdot n'_y)$.

V. SIMULATION RESULTS

We evaluate the performance of our two schemes through simulations. The programs are written in Matlab, and the experimental platform is a PC featured with an Intel Core 2 E8400 CPU and 4GB RAM, running Windows XP. However, we expect the central server in practice to be much more powerful. The offline measurement may also be outsourced to cloud servers and benefit from parallel work. The datasets used in the simulations are generated such that each vehicle ID or key is a 32-bit number, and two RSEs, s_x and s_y, each store 3,000 vehicle records. There are 500 vehicles that pass both s_x and s_y, i.e., the actual O-D flow size n_{xy} is 500.

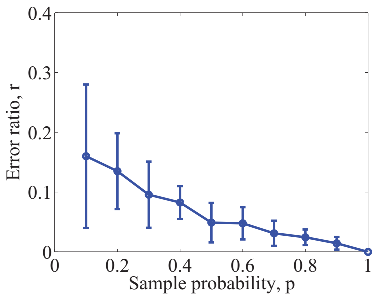

Fig. 1. Mean and standard deviation of error ratios for O-D flow measurement

In the simulations, we consider two performance metrics. One is *measurement accuracy*, represented by error ratio r:

$$r = \frac{|\hat{n}_{xy} - n_{xy}|}{n_{xy}} \times 100\%, \tag{4}$$

where $\hat{n}_{xy}$ is the measured O-D flow size. Clearly, smaller r represents more accurate measurement result, and vice versa. The other is *computation overhead*, measured by time consumed for the central server to obtain $\hat{n}_{xy}$.

Our first scheme has an error ratio of 0% unless it does not respond to the keys that it has seen before (for privacy purpose as we have discussed in Section III-D). Hence, we only measure its time cost. The enhanced scheme addresses the identical-key attack at the cost of higher computation overhead. It has an error ratio of 0% only when the sampling probability p is 1. In our simulations, we vary p from 0.1 to 1, with a step size of 0.1. For each p, we randomly draw a fraction p of all records from s_x and do the same for s_y. The offline measurement is performed over the sampled subsets and the O-D flow size are estimated by (1). The time cost is measured and the error ratio is computed from (4). The process is repeated 10 times to show the statistic effect.

Table 1 and Figures 1-2 present our simulation results. Table 1 shows the computation overhead of the first scheme and the second scheme under varied sampling probabilities p. The two figures are drawn from the simulation results of the second scheme. Figure 1 shows the mean and standard deviation of the error ratio r under varied p. The length of each error bar is two times the standard deviation of r, whose mean is at the center of the bar. We see that both mean and standard deviation of r decrease with the increment of p. Intuitively, when we increase the sample size, the measurement result is likely to be more accurate. When p equals 1, the error ratio is 0% (the rightmost of the figure), which agrees with our theoretical prediction. Figure 2 shows the average time taken by the central server to measure the O-D flow size under each sampling probability. It

TABLE I
AVERAGE COMPUTATION OVERHEAD FOR THE TWO PROPOSED SCHEMES.

	First Scheme	**Second Scheme with Different Sampling Probabilities**									
		0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1.0
time ($\times 10^3$**secs**)	0.001125	0.0173	0.0692	0.1555	0.2755	0.4303	0.6196	0.8464	1.1016	1.3938	1.7204

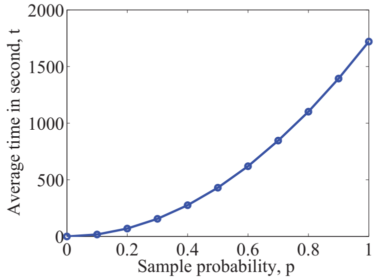

Fig. 2. Average time overhead for offline measurement.

is clear that the computation overhead increases quadratically with p, which is also consistent to our analysis in Section IV-B. We stress that this is offline computation.

VI. RELATED WORK

A. *Traffic Volume Measurement*

Various prediction models have been proposed using data recorded by automatic traffic recorders (ATR) installed at road sections. For example, the multiple linear regression model in [2] addresses the scarce of APRs for county roads, and the artificial neural network in [3] addresses traffic in cities. Other predication approaches include spatial regression [4] and support vector machines [5], etc. These solutions, though elegant, are not appropriate for "point-to-point" traffic volume measurement. While some "point-to-point" statistics may be inferred from "point" data [6], we prefer a more accurate direct-measurement approach that should also address the privacy concern. Although Google recently announced to provide real-time traffic data service in Google maps [16], their approach cannot assure vehicle's privacy since it uses GPS and Wi-Fi in phones to track locations [17].

B. *Privacy Preserving Data Mining*

Another branch of research that relates to (but is also significantly different from) ours is privacy preserving data mining (PPDM). Solutions can be summarized into two categories. One is to "randomly" perturb the data by adding "noise" before mining, and mitigate their impact afterwards [18] [19]. The other is to use cryptographic techniques [20] [21]. Though they are motivated by the same need to both protect privileged information and enable its use, directly applying PPDM methods to O-D flow measurement can still be problematic. In PPDM tasks, schemes can be quite efficient since they assume "untouched" data are gathered and shared among only a few data collectors. However, in our context, no one should know real vehicle ID ("untouched" data) except itself, which demands privacy preservation from the beginning and incurs much higher computation overhead, motivating us to seek statistical methods to improve its efficiency.

VII. CONCLUSIONS

In this paper, we focus on privacy-assured "point-to-point" traffic volume monitoring. Two novel schemes are proposed, utilizing VCPS and the nice properties of COHFs. Sampling is applied to improve efficiency. Simulations are performed to evaluate the feasibility and scalability of our schemes.

VIII. ACKNOWLEDGMENT

This work is supported in part by the National Science Foundation under grant CPS 0931969 and by a grant from the Center for Multimodal Solutions for Congestion Mitigation sponsored by the US Department of Transportation.

REFERENCES

[1] USDOT, "Traffic Monitoring Guide," *Section 3, Traffic Volume Monitoring, http://www.fhwa.dot.gov/ohim/tmguide/tmg3.htm*, 2001.

[2] D. Mohamad, K. C. Sinha, T. Kuczek, and C. F. Scholer, "Annual Average Daily Traffic Prediction Model for County Roads," *Journal of the Transportation Research Board*, vol. 1617/1998, pp. 69–77, 1998.

[3] W. Lam and J. Xu, "Estimation of AADT from Short Period Counts in Hong Kong – A Comparison Between Neural Network Method and Regression Analysis," *Journal of Advanced Transportation*, 2000.

[4] J. K. Eom, M. S. Park, T. Heo, and L. F. Huntsinger, "Improving the Prediction of Annual Average Daily Traffic for Nonfreeway Facilities by Applying a Spatial Statistical Method," *Journal of the Transportation Research Board*, vol. 1968/2006, pp. 20–29, 2006.

[5] M. C. Neto, Y. Jeong, M. K. Jeong, and L. D. Han, "AADT prediction using support vector regression with data-dependent parameters," *Expert Systems with Applications*, vol. 36, pp. 2979–2986, March 2009.

[6] Y. Lou and Y. Yin, "A Decomposition Scheme for Estimating Dynamic Origindestination Flows on Actuation-controlled Signalized Arterials," *Transportation Research Part C*, vol. 18, 2010.

[7] J. Eriksson, H. Balakrishnan, and S. Madden, "Cabernet: Vehicular Content Delivery Using WiFi," *Proc. of MOBICOM*, 2008.

[8] U. Lee, J. Lee, J. Park, and M. Gerla, "FleaNet: A Virtual Market Place on Vehicular Networks," *IEEE Trans. on Vehicular Technology*, 2010.

[9] [Online]. Available: http://www.its.dot.gov/press/2010/vii2intellidrive

[10] [Online]. Available: http://www.dot.gov/

[11] A. J. Menezes, P. C. Oorschot, and S. A. Vanstone, *Handbook of Applied Cryptography*. CRC Press, 1996.

[12] J. Benaloh and M. D. Mare, "One-way Accumulators: a Decentralized Alternative to Digital Signatures," *Proc. of EUROCRYPT*, 1993.

[13] C. Kaufman, R. Perlman, and M. Speciner, *Network Security, Private Communication in a Public World*, 2nd ed. Prentice Hall, 2002.

[14] R. Rivest, A. Shamir, and L. Adleman, "A Method for Obtaining Digital Signatures and Public-Key Cryptosystems," *Communications of the ACM*, vol. 21, pp. 120–126, 1978.

[15] A. Shamir, "On the generation of cryptographically strong pseudorandom sequences," *ACM Trans. on Computer System*, 1983.

[16] "Google map's time-in-traffic feature." [Online]. Available: http://mashable.com/2012/03/29/google-maps-traffic-data/

[17] T. Jeske, "Floating Car Data from Smartphones: What Google and Waze Know About You and How Hackers Can Control Traffic," *Proc. of the BlackHat Europe*, 2013.

[18] R. Agrawal and R. Srikant, "Privacy-Preserving Data Mining," *Proc. of the ACM SIGMOD*, pp. 439–450, 2000.

[19] A. Evfimievski, J. Gehrke, and R. Srikant, "Limiting Privacy Breaches in Privacy Preserving Data Mining," *Proc. of PODS*, 2003.

[20] H. Yu, X. Jiang, and J. Vaidya, "Privacy Preserving SVM using Nonlinear Kernels on Horizontally Partitioned Data," *Proc. of the 2006 ACM symposium on Applied Computing*, pp. 603–610, 2006.

[21] S. Narmadha and S. Vijayarani, "Protecting Sensitive Association Rules in Privacy Preserving Data Mining using Genetic Algorithms," *Journal of Computer Applications*, vol. 33, no. 7, pp. 36–43, 2011.