Privacy-Preserving Point-to-Point Transportation Traffic Measurement through Bit Array Masking in Intelligent Cyber-Physical Road Systems

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

Abstract—Traffic measurement is a critical function in transportation engineering. We consider privacy-preserving point-to-point traffic measurement in this paper. We measure the number of vehicles traveling from one geographical location to another by taking advantage of capabilities provided by the intelligent cyber-physical road systems that enable automatic collection of traffic data. The challenge is to allow the collection of aggregate point-to-point data while preserving the privacy of individual vehicles. We propose a novel measurement scheme which utilizes bit arrays to collect data and adopts maximum likelihood estimation (MLE) to obtain the measurement result. Both mathematical proof and simulation demonstrate the practicality and scalability of our scheme.

Keywords-transportation traffic measurement, privacy, cyber-physical systems, maximum likelihood estimation

I. INTRODUCTION

Traffic measurement is a critical function in transportation engineering [1]. There are two categories of traffic statistics, “point” statistics and “point-to-point” statistics. Point statistics tell the number of vehicles traversing a specific point (location). Various prediction models have been proposed to estimate them [2], [3]. Point-to-point statistics describe the number of vehicles traveling between two points. They are essential inputs to a variety of studies including estimation of traffic link flow distribution as part of investment plan, calculation of road exposure rates as part of safety analysis, and characterization of turning movements at intersections for signal timing determination. While point-to-point statistics may be inferred from point data [4], little work has been done on direct measurement of “point-to-point” traffic particularly when drivers’ location privacy is of concern.

This paper considers the important problem of privacy-preserving point-to-point transportation traffic measurement. The set of vehicles traveling from one geographical location to another is modeled as a traffic flow, whose size is the number of vehicles in the set. To enable automatic collection of traffic data, we take advantage of intelligent cyber-physical road systems (CPRS), which integrate the latest technologies in wireless communications and on-board computer processing to improve safety and efficiency of transportation systems [5] [6]. For example, IntelliDrive [7] from USDOT [8] envisions a nationwide system where vehicles communicate with roadside equipments (RSE) in real time via dedicated short range communications. In CPRS, vehicles may report their IDs to RSEs when they pass by, and that information can be used by the authority to measure traffic flows. However, if a vehicle keeps transmitting its unique identifier to RSEs, that information will enable others to track its entire moving history. As more and more people are concerned about their location privacy, the degree of privacy that a scheme preserves will directly affect its applicability.

To address the issue of privacy, there are many issues that we need to consider: First of all, we need a criteria to tell what is good privacy and what is not. In this paper, we capture the essence of privacy in traffic flow measurement, and quantify it as a probability that a potential tracker cannot identify any trace of any vehicle. Secondly, given that criteria, how can we preserve the optimal privacy? Apparently, the better the privacy preservation is, the more applicable the measurement scheme will be. Furthermore, to protect the privacy of vehicles, only randomized and de-identified information is collected.

We propose a novel scheme for privacy-preserving traffic flow measurement. The new scheme utilizes bit arrays to encode transportation traffic data sent from vehicles to RSEs, and adopt the maximum likelihood estimation (MLE) to obtain the measurement result. The measurement accuracy as well as privacy preserved by our proposed scheme are analyzed through both mathematical proof and simulation, which demonstrate the applicability of our scheme.

The rest of the paper is organized as follows: Section II gives the system and threat models, the problem statement, and the performance metrics that we consider. Section III discusses some straightforward solutions and their limitations, followed by Section IV which presents our novel solution and its performance analysis. Section V shows the simulation results. Section VI summarizes the related work. Finally, Section VII draws the conclusion.

II. PRELIMINARIES

A. System Model

We consider an intelligent cyber-physical road system involving three groups of entities: vehicles, roadside
equipments (RSE), and a central server. Each vehicle has a unique ID, e.g., its VIN, or other number chosen permanently or temporarily. Each RSE also has a unique ID. Both vehicles and RSEs are equipped with computing and communication capabilities, e.g., on-board computer chips and communication modules. Vehicles communicate with RSEs in real time via dedicated short range communications (DSRC) [8]. RSEs are connected to the central server through wired or wireless means. They collect information from vehicles and transfer it to the central server for further processing at the end of each measurement period.

B. Problem Statement

We define a traffic flow between one RSE-equipped location and another RSE-equipped location as the set of vehicles traveling between the two locations during a measurement period. The traffic flow size is the number of vehicles in the set. The problem is to measure the sizes of traffic flows in a road system between all pairs of locations where RSEs are installed while protecting vehicles’ privacy. To achieve the privacy-preserving end, we need a solution in which a vehicle never transmits any unique identifier. Ideally, the information transmitted by the vehicles to the RSEs looks totally random, out of which neither the identity nor the trajectory of any vehicle can be pieced with high probability.

We assume that a special MAC protocol is used to support privacy preservation such that the MAC address of a vehicle is not fixed. Vehicles may pick an MAC address randomly from a large space for one-time use.

C. Threat Model

We assume a semi-honest model for the RSEs. On the one hand, all RSEs are from trustworthy authorities, which 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. On the other hand, 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 unique identifier upon each query, that identifier can be exploited for tracking purpose. Note that there are also 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. In this paper, we focus on preventing tracking caused by the traffic flow measurement scheme itself.

D. Performance Metrics

In this paper, we use three performance metrics to evaluate a traffic flow measurement scheme: measurement accuracy, computation overhead, and preserved privacy.

1) Measurement Accuracy: Let $n_c$ be the true size of a traffic flow between a pair of locations and $\hat{n}_c$ be the corresponding measured result. We specify the measurement accuracy through a parameter $\beta$ which satisfies the following requirement: the probability for $n_c$ to fall into the interval $[\hat{n}_c \cdot (1 - \beta), \hat{n}_c \cdot (1 + \beta)]$ must be at least $\alpha$, where $\alpha$ is a pre-determined parameter in the range of $[0, 1]$. Given $\alpha$, a smaller value of $\beta$ means better measurement results. For example, when $\alpha = 95\%$, a solution with $\beta = 0.05$ is better than a solution with $\beta = 0.1$ because the former ensures the measured traffic flow size has a probability of 95\% to be within $\pm 5\%$ of the true value, while the latter only ensures the measured result to be within $\pm 10\%$ of the true value.

2) Computation Overhead: We consider the computation overhead for vehicles, RSEs, and the central server. For vehicles, we measure the computation overhead for each vehicle per RSE en route. For RSEs, we measure the computation overhead for each RSE per passing vehicle. For the central server, we measure the computation overhead to compute the traffic flow size of per pair of RSEs.

3) Preserved Privacy: We capture the essence of privacy preservation in point-to-point transportation traffic measurement, which is allowing the tracker only a limited chance to identify any partial or full trajectory of any vehicle. Accordingly, we quantify the privacy of a scheme through a parameter $p$ which satisfies the following requirement: the probability for any “trace” of any vehicle not to be identified must be at least $p$, where a trace of a vehicle is a pair of RSEs it has passed by. A larger value of $p$ means better privacy. Intuitively, a scheme with $p = 0.5$ is better than one with $p = 0.1$ in terms of privacy because the latter gives the tracker a better chance to link traces of a vehicle to obtain its trajectory since it allows the traces to be identified with a higher probability, i.e., $1 - p$.

III. STRAIGHTFORWARD APPROACHES AND THEIR LIMITATIONS

To measure the traffic flow sizes between all pairs of RSEs in the road system, a straightforward approach is making vehicles report their IDs to all RSEs that they pass by. RSEs collect the IDs from the passing vehicles. At the end of each measurement period, all RSEs send their collected ID sets to the central server, which then measures the traffic flow size between each pair of RSEs by simply comparing the two corresponding ID sets: if a vehicle ID appears in both ID sets, then the vehicle must have passed both RSEs. Thus, the number of IDs that appear in both ID sets equals the true traffic flow size between the two corresponding RSEs. However, this simple approach leads to serious privacy breaching as it reveals vehicles’ identities along the way.

A natural follow-up thinking is making vehicles report their encrypted IDs (EIDs) to the RSEs en route. The central server will compute traffic flow sizes based on
the EID sets collected by RSEs. To prevent the tracker from using fixed EIDs to identify vehicles, each vehicle has many EIDs encrypted by different keys. However, the EIDs of a vehicle must satisfy the following property: they will produce the same result after a certain procedure of computations, allowing the central server to find out they represent the same vehicle. In this scheme, although vehicles’ true identities are hidden, traces of each vehicle are still revealed and can be linked to obtain its full trajectory.

An alternative approach is having the RSEs broadcast their IDs (RIDs). Each vehicle will record the RIDs of all RSEs it has passed by, and transmit them to every RSE that it passes in the future. RSEs collect those RIDs from passing vehicles, and send them to the central server at the end of each measurement period. To compute the size of a traffic flow between two RSEs, denoted as \( R_x \) to \( R_y \), under the assumption that it passes in the future. RSEs collect those RIDs from passing vehicles, and transmit them to every RSE that it has passed by, and transmit them to every RSE that it passes in the future. RSEs collect those RIDs from passing vehicles, and send them to the central server at the end of each measurement period. To compute the size of a traffic flow between two RSEs, \( R_x \) and \( R_y \), the central server simply goes through the RID set collected by \( R_y \) (\( R_x \)), and count the number of times that \( R_x \) appears in this set. This is the size of the directional traffic flow from \( R_x \) to \( R_y \). The undirectional traffic flow between \( R_x \) and \( R_y \) is the sum of both directional flow sizes. Clearly, this approach also reveals a vehicle’s trajectory in the form of a list of RIDs that each RSE that it passes.

IV. PRIVACY-PRESERVING POINT-TO-POINT TRANSPORTATION TRAFFIC MEASUREMENT

In this section, we present our novel scheme for privacy preserving point-to-point transportation traffic measurement. There are two phases for each measurement period: online coding and offline decoding. Online coding is an interaction between vehicles and RSEs, where necessary information for traffic flow measurement are securely collected. Later in the offline decoding phase, the central server will use those information to compute traffic flow sizes. In the following, we first illustrate the two measurement phases, and then evaluate our scheme with respect to the three performance metrics described in Section II-D.

A. Online Coding Phase

In our scheme, each RSE \( R_x \) maintains a counter \( n_x \), which keeps track of the total number of vehicles passing by during the current measurement period. \( R_x \) also maintains a bit array \( B_x \) with a fixed length \( m \) \((m > 1)\) to mask vehicle identities. At the beginning of each measurement period, \( n_x \) and all the bits in \( B_x \) are set to zeros. In addition, each vehicle \( v \) has a logical bit array \( LB_v \), which consists of \( s \) \((1 < s < m)\) bits randomly selected from \( B_v \). The indices of these bits \( B_v \) are \( H(v \oplus K_v \oplus X[0]),... H(v \oplus K_v \oplus X[s-1]) \), where \( \oplus \) is the bitwise XOR, \( H(...) \) is a hash function whose range is \([0, m)\), \( X \) is an integer array of randomly chosen constants whose purpose is to arbitrarily alter the hash result, and \( K_v \) is the private key of \( v \) whose purpose is to protect the privacy of its logical bit array.

The online coding phase is quite simple. 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 RID and its public-key certificate. Suppose a vehicle, whose ID is \( v \), receives a query from an RSE, whose ID is \( R_x \). The vehicle first verifies the certificate, and then uses the RSE’s public key to authenticate the RSE. After verifying that \( R_x \) is from the trustworthy authority, the vehicle \( v \) will randomly select a bit from its logical bit array \( LB_v \) by computing an index \( b = H(v \oplus K_v \oplus X(H(R_x) \text{ mod } s)) \). The vehicle then sends the resulting index \( b \) to the RSE \( R_x \). Upon receiving the index \( b \), \( R_x \) will first increase its counter \( n_x \) by 1, and then set the \( b \)th bit in \( B_x \) to 1:

\[
B_x[b \oplus H(v \oplus K_v \oplus X(H(R_x) \text{ mod } s))] = 1. \tag{1}
\]

B. Offline Decoding Phase

At the end of each measurement period, all RSEs will send their counters and bit arrays to the central server, which then performs the offline measurement. We employ the maximum likelihood estimation (MLE) [9] to measure the sizes of traffic flows based on the counters and bit arrays.

Suppose the set of vehicles that pass RSE \( R_x \) (\( R_y \)) is denoted as \( S_x \) (\( S_y \)) with cardinality \( |S_x| = n_x, |S_y| = n_y \). Clearly, the set of vehicles that pass both RSE \( R_x \) and \( R_y \) is \( S_x \cap S_y \). Denote its cardinality as \( n_o \), which is the value that we want to measure. Furthermore, denote by \( S \) the subset of vehicles in \( S_x \cap S_y \) that happen to set the same bit in \( B_x \) and \( B_y \), where \( B_x \) and \( B_y \) are the bit arrays at \( R_x \) and \( R_y \), respectively. Let \( n_v \) be the cardinality of \( S \), i.e., \( n_v = |S| \). Clearly, \( S \subseteq S_x \cap S_y \) and \( 0 \leq n_o \leq n_v \). For any vehicle, it has the same probability \( 1/2 \) to set any bit in its \( s \)-bit logical bit array. As a result, the probability for an arbitrary vehicle \( v \) from \( S_x \cap S_y \) to select the same bit in both \( B_x \) and \( B_y \) is \( s \times \frac{1}{2} \times \frac{1}{2} = \frac{s}{4} \). Therefore, the number of such vehicles, \( n_v \), is binomially distributed according to \( B(n_v, \frac{1}{2}) \). Accordingly, the probability for \( n_o = z \) \((0 \leq z \leq n_v)\) is

\[
P(n_o = z) = \binom{n_v}{z} \left(\frac{1}{2}\right)^z \left(1 - \frac{1}{2}\right)^{n_v - z}. \tag{2}
\]

Given the counters \( n_x \) and \( n_y \), and bit arrays \( B_x \) and \( B_y \), we measure \( n_o \) as follows: First, take a bitwise AND of \( B_x \) and \( B_y \), and denote the resulting bit array as \( B_c \). Namely,

\[
B_c[i] = B_x[i] \land B_y[i], \forall i \in [0, m - 1]. \tag{3}
\]

Calculate the number of 0’s in \( B_c \), and denote it as \( U_c \). For an arbitrary bit \( b \) in \( B_c \), it is ‘0’ if and only if the following

\[
U_c[b] = B_c[b] \land \neg B_c[b]. \tag{4}
\]

Furthermore, the probability for \( n_o = z \) \((0 \leq z \leq n_v)\) is

\[
P(n_o = z) = \binom{n_v}{z} \left(\frac{1}{2}\right)^z \left(1 - \frac{1}{2}\right)^{n_v - z}. \tag{5}
\]

Given the counters \( n_x \) and \( n_y \), and bit arrays \( B_x \) and \( B_y \), we measure \( n_o \) as follows: First, take a bitwise AND of \( B_x \) and \( B_y \), and denote the resulting bit array as \( B_c \). Namely,

\[
B_c[i] = B_x[i] \land B_y[i], \forall i \in [0, m - 1]. \tag{6}
\]

Calculate the number of 0’s in \( B_c \), and denote it as \( U_c \). For an arbitrary bit \( b \) in \( B_c \), it is ‘0’ if and only if the following

\[
U_c[b] = B_c[b] \land \neg B_c[b]. \tag{7}
\]
two conditions are satisfied: First, it is not chosen by any vehicle in S. If b is chosen by a vehicle v ∈ S, we know that bit b in Sx and by are both set to ‘1’ (hence bit b in Bc is also ‘1’). Since each vehicle in Sx ∩ Sb has probability \( \frac{1}{m} \) to set bit b to ‘1’, the probability for the vehicle not to choose bit b is \( 1 - \frac{1}{m} \). There are \( n_c \) vehicles in S. Therefore, the probability for bit b not chosen by any vehicle in S is
\[
q_1 = (1 - \frac{1}{m})^{n_c}.
\]

Second, it is either not chosen by any vehicle in \( S_x - S \) or not chosen by any vehicle in \( S_y - S \). Otherwise, bit b in both \( B_x \) and \( B_y \) will be set to ‘1’ (hence bit b in \( B_c \) is also ‘1’). Similar to the previous analysis, we can obtain that the probability for bit b not chosen by any vehicle in \( S_x - S \) is \( (1 - \frac{1}{m})^{n_c-n_c} \), and the probability for bit b not chosen by any vehicle in \( S_y - S \) is \( (1 - \frac{1}{m})^{n_y-n_c} \). Therefore, the probability for the second condition to be satisfied is
\[
q_2 = 1 - (1 - (1 - \frac{1}{m})^{n_x-n_c}) \times (1 - (1 - \frac{1}{m})^{n_y-n_c})
\]
\[
= (1 - \frac{1}{m})^{n_x-n_c} + (1 - \frac{1}{m})^{n_y-n_c} - (1 - \frac{1}{m})^{n_x+n_y-2n_c}.
\]

Combining the above analysis, the conditional probability for bit b in \( B_c \) to remain ‘0’ given \( n_o = z \) is \( q_1 \times q_2 \), namely,
\[
q(n_c | n_o = z) = (1 - \frac{1}{m})^{n_c} + (1 - \frac{1}{m})^{n_y} - (1 - \frac{1}{m})^{n_x+n_y-z}.
\]

Given \( q(n_c | n_o = z) \) and the distribution of \( n_o \), the overall probability \( q(n_c) \) for bit b in \( B_c \) to remain ‘0’ is
\[
q(n_c) = \sum_{z=0}^{n_c} q(n_c | n_o = z) \times P(n_o = z)
\]
\[
= \sum_{z=0}^{n_c} q(n_c | n_o = z) \times (\frac{n_c}{z}) (\frac{1}{s})^{z} (1 - \frac{1}{s})^{n_c-z}
\]
\[
= (1 - \frac{1}{m})^{n_c} + (1 - \frac{1}{m})^{n_y} - (1 - \frac{1}{m})^{n_x+n_y}
\]
\[
\times (\frac{1}{s} + (1 - \frac{1}{s})(1 - \frac{1}{m}))^{n_c}.
\]

Since each bit in \( B_c \) has a probability \( q(n_c) \) to be ‘0’, and the observed number of ‘0’ bits in \( B_c \) is \( U_c \), the likelihood function for this observation to occur is:
\[
L = (q(n_c))^{U_c} \times (1 - q(n_c))^{m-U_c}.
\]

We follow the standard process of MLE to find the optimal value of \( n_c \) that maximizes the likelihood function in (8):
\[
\hat{n}_c = \arg \max_{n_c} \{L\}
\]

To find \( \hat{n}_c \), we take logarithm on both sides of (8):
\[
\ln L = U_c \times \ln q(n_c) + (m - U_c) \times \ln(1 - q(n_c)).
\]

Take the first order derivative of (10), we have:
\[
d\ln L \over dn_c = \frac{U_c}{q(n_c)} - \frac{m - U_c}{1 - q(n_c)} \times q'(n_c),
\]

where \( q'(n_c) \) can be computed from (7) as follows:
\[
q'(n_c) = \frac{dq(n_c)}{dn_c}
\]
\[
= - (1 - \frac{1}{m})^{n_c+n_y} \times \left( \frac{1}{s} + (1 - \frac{1}{s})(1 - \frac{1}{m}) \right)^{n_c}
\]
\[
\times \ln \left( \frac{1}{s} + (1 - \frac{1}{s})(1 - \frac{1}{m}) \right).
\]

To compute \( \hat{n}_c \), we set the right side of (11) to 0:
\[
\left( \frac{U_c}{q(n_c)} - \frac{m - U_c}{1 - q(n_c)} \right) \times q'(n_c) = 0.
\]

Observe from (12) that \( q'(n_c) \) cannot be 0 when \( m > 1 \) and \( s > 1 \). Therefore, we have:
\[
\frac{U_c}{q(n_c)} - \frac{m - U_c}{1 - q(n_c)} = 0.
\]

Substituting (7) to (14), we obtain the MLE estimator \( \hat{n}_c \) of the desired traffic flow size \( n_c \) as follows:
\[
\hat{n}_c = \ln((1 - \frac{1}{m})^{n_c} + (1 - \frac{1}{m})^{n_y} - \frac{U_c}{m}) / \ln \left( \frac{1}{s} + (1 - \frac{1}{s})(1 - \frac{1}{m}) \right)
\]
\[
\hat{n}_c \sim \text{Norm} \left( \hat{n}_c, \frac{1}{I(\hat{n}_c)} \right),
\]

where \( I(\hat{n}_c) \) is the fissher information of \( L \), defined as:
\[
I(\hat{n}_c) = -E \left[ \frac{d^2 \ln L}{dn_c^2} \right].
\]

According to (11), we compute the second-order derivative of \( \ln L \) in the following:
\[
\frac{d^2 \ln L}{dn_c^2} = \left( - \frac{U_c \cdot q'(n_c)}{q^2(n_c)} - \frac{(m - U_c) \cdot q'(n_c)}{(1 - q(n_c))^2} \right) \times q'(n_c)
\]
\[
+ \left( \frac{U_c}{q(n_c)} - \frac{m - U_c}{1 - q(n_c)} \right) \times q'(n_c) \times \ln C,
\]
where \( C = \frac{1}{2} + (1 - \frac{1}{2})(1 - \frac{1}{2}) \) and \( q'(n_c) \) is given in (12).

For an arbitrary bit \( b \) in \( B_c \), it has probability \( q(n_c) \) to remain ‘0’. \( U_c \) is the number of ‘0’s in \( B_c \). Therefore, \( U_c \) follows a binomial distribution \( B(m, q(n_c)) \). Accordingly,

\[
E(U_c) = m \cdot q(n_c).
\]  

(19)

Substituting (18) and (19) to compute (17), we have

\[
I(\hat{c}) = \frac{m \cdot q(n_c) + m \cdot q(n_c)}{1 - q(n_c)} \times q'(n_c)
\]

\[
= \frac{m(q'(n_c))^2}{q(n_c)(1 - q(n_c))},
\]  

(20)

According to (16), the variance of \( \hat{c} \) is

\[
Var(\hat{c}) = \frac{1}{I(\hat{c})} = \frac{q(n_c)(1 - q(n_c))}{m(q'(n_c))^2}.
\]  

(21)

Therefore, the confidence interval of our measurement is

\[
\hat{c} \pm Z_\alpha \times \sqrt{\frac{q(n_c)(1 - q(n_c))}{m(q'(n_c))^2}},
\]  

(22)

where \( \alpha \) is the confidence level parameter and \( Z_\alpha \) is the \( \alpha \) percentile for the standard Gaussian distribution [11].

D. Privacy Guarantee

Next, we evaluate the preserved privacy of our measurement scheme. Note that in our scheme, the only information that a vehicle \( v \) ever transmits to an RSE en route is an index of a bit \( b \) randomly selected from its \( s \)-bit logical bit array, \( LB_v \). From the tracker’s point of view, it can only identify the trace of a vehicle passing by two RSEs \( R_x \) and \( R_y \) through the observation of the bits that are set to ‘1’ in both \( B_x \) and \( B_y \); these bits will be ‘1’ in \( B_c \). Therefore, the preserved privacy of our scheme is actually a conditional probability which tells to what degree an observed ‘1’ in \( B_c \) does not represent a common vehicle passing by both \( R_x \) and \( R_y \). We derive this conditional probability in the following.

Firstly, consider the probability for the tracker to observe an arbitrary bit, \( b \), to be set to ‘1’ in both \( B_x \) and \( B_y \) (event \( A \), \( P(A) \)). Obviously, the probability \( P(A) \) equals 1 minus \( q(n_c) \) given our analysis in Section IV-B:

\[
P(A) = 1 - \left(1 - \frac{1}{m}\right)^{n_x} - \left(1 - \frac{1}{m}\right)^{n_y} + \left(1 - \frac{1}{m}\right)^{n_x+n_y}
\]

\[
\times \left(\frac{\frac{1}{2} + (1 - \frac{1}{2})(1 - \frac{1}{2})}{1 - \frac{1}{m}}\right)^{n_c}.
\]  

(23)

Secondly, consider the conditional probability for such a bit, \( b \), to not represent a common vehicle passing both \( R_x \) and \( R_y \) (event \( E \), \( P(E|A) \)). This is the privacy \( p \) that we want to derive. Note that event \( E \) happens if and only if bit \( b \) in \( B_x \) is set only by vehicles passing only RSE \( R_x \) (i.e., in set \( S_x - S_y \)), and bit \( b \) in \( B_y \) is set only by vehicles passing only RSE \( R_y \) (i.e., in set \( S_y - S_x \)). Denote these two events as \( E_x \) and \( E_y \), respectively. There are \( n_x \) (\( n_y \)) vehicles passing \( R_x \) (\( R_y \)), and \( n_c \) vehicles among them pass both \( R_x \) and \( R_y \). Since each vehicle has a probability \( \frac{1}{m} \) to set bit \( b \) to ‘1’, the probability for \( E_x \) (\( E_y \)) to happen is:

\[
P(E_x) = \left(1 - \left(1 - \frac{1}{m}\right)^{n_x}\right) \times \left(1 - \frac{1}{m}\right)^{n_c},
\]  

(24)

\[
P(E_y) = \left(1 - \left(1 - \frac{1}{m}\right)^{n_y}\right) \times \left(1 - \frac{1}{m}\right)^{n_c}.
\]  

(25)

Combining the above analysis, we have the formula for the preserved privacy of our scheme as follows:

\[
p = P(E|A) = \frac{P(E_x) \times P(E_y)}{P(A)}
\]

\[
= \frac{1}{P(A)} \times (\left(1 - \left(1 - \frac{1}{m}\right)^{n_x}\right) \times \left(1 - \frac{1}{m}\right)^{n_c})
\]

\[
\times (\left(1 - \left(1 - \frac{1}{m}\right)^{n_y}\right) \times (1 - \frac{1}{m})^{n_c}),
\]  

(26)

where \( P(A) \) is given in (23).

Observe that there are 2 parameters, \( s \) and \( m \), that determine the value of \( P(E|A) \). Among them, \( s \) only appears in the denominator \( P(A) \), and it influences \( P(E|A) \) through varying the value of \( P(A) \), \( m \) influences both the denominator and the numerator. In the following, we consider the influences of \( s \) and \( m \) on \( P(E|A) \) by first examining the influence of \( s \) on \( P(A) \) (hence that on \( P(E|A) \)) under various values of \( m \), and then analyzing how \( m \) determines the value of \( P(E|A) \) given values for \( s \).

1) Influence of \( s \) on \( P(A) \): To examine how \( s \) affects \( P(A) \), we take partial derivative of (23) with respect to \( s \)

\[
\frac{\partial P(A)}{\partial s} = -(1 - \frac{1}{m})^{n_x+n_y} \times \frac{n_c}{(m-1)s^2C_{m-1}}.
\]  

(27)

Clearly, \( \frac{\partial P(A)}{\partial s} < 0 \). Therefore, with the increment of \( s \), the value of \( P(A) \) decreases, and in turn, the value of \( P(E|A) \) increases. In other words, the preserved privacy will be better with a larger value of \( s \). The numerical results are shown in the first two plots of Figure 1 where \( n_x = n_y = n = 50,000, n_c = 5,000, \) and \( s = 2,5,10 \), corresponding to three curves in each plot. Clearly, as \( s \) increases, the probability \( P(A) \) decreases.

Another observation from the numerical results gives that when \( s > 5 \), the difference in probability \( P(A) \) under different \( s \) becomes quite small. For instance, when \( m \in [5n,20n] \), the difference in \( P(A) \) when \( s = 5 \) and \( s = 10 \) is smaller than 0.0005 (see the two lower curves in the second plot of Figure 1). When \( n > 10 \), that difference becomes negligible. Therefore, when we analyze the effect of \( m \) on \( P(E|A) \) in the following subsection, and set up the parameters for our simulations, we only consider the cases when \( s = 2,5,10 \), with established understanding that larger values of \( s \) will only make negligible differences.
2) Influence of \( m \) on \( P(E|A) \): To examine the effects of \( m \) on \( P(E|A) \), we take the partial derivative of (26) with respect to \( m \) and obtain the following

\[
\frac{\partial P(E|A)}{\partial m} = \frac{\partial P(E)}{\partial m} \times P(A) - \frac{\partial P(A)}{\partial m} \times P(E) \quad (28)
\]

where \( P(E) = P(E_x) \times P(E_y) \). \( P(E_x) \) and \( P(E_y) \) are given in (24) and (25), respectively. Therefore, the partial derivative of \( P(E) \) with respect to \( m \) is

\[
\frac{\partial P(E)}{\partial m} = \frac{m - 1}{m^3} \left[ 2n_x (1 - \frac{1}{m})^{n_y - 1} + (n_x + n_y) (1 - \frac{1}{m})^{n_y - 1} + (n_x + n_y - 1) - (n_x + n_y) \right] \left[ (1 - \frac{1}{m})^{n_y - 1} \cdot C_{n_x + n_y - 2} \cdot \binom{x + y - 1}{n_x + n_y} \right]
\]

In addition, from (23), we can compute the derivative of \( P(A) \) with respect to \( m \):

\[
\frac{\partial P(A)}{\partial m} = \frac{1}{m^2} \left[ -n_x (1 - \frac{1}{m})^{n_y - 1} - n_y (1 - \frac{1}{m})^{n_x - 1} + (1 - \frac{1}{m})^{n_x + n_y - 2} \cdot C_{n_x + n_y} \cdot \left( (n_x + n_y) (1 - \frac{1}{m}) + \frac{\ln C}{s} \right) \right]
\]

Both \( \frac{\partial P(E)}{\partial m} \) and \( \frac{\partial P(A)}{\partial m} \) are negative, meaning that both \( P(E) \) and \( P(A) \) decrease with the increment of \( m \). Intuitively, increasing \( m \) gives each vehicle a smaller chance \( \frac{1}{m} \) to set an arbitrary bit, \( b \). Hence, \( P(E) \) and \( P(A) \) also drop. The effects that \( m \) has on \( P(E|A) \) are twofold: on the one hand, the increment of \( m \) decreases the denominator \( P(A) \), which pulls the privacy up; on the other hand, the increment of \( m \) decreases the numerator \( P(E) \), which drags the privacy down. The combination of these two effects gives that the partial derivative of \( P(E|A) \) with respect to \( m \) can be positive, negative, or 0, according to (28). Also, given \( s \), we can choose an optimal \( m \) to achieve the best degree of privacy. The optimal \( m \) is obtained by setting the right side of (28) to 0. The third plot of Figure 1 shows the numerical results for the preserved privacy under different \( m \) when \( n_x = n_y = n = 50,000, n_c = 5,000 \), and \( s = 2, 5, 10 \).

Clearly, along each curve (controlled by \( s \)), there is an optimal value of \( m \) that gives the optimal privacy, \( p \). For instance, \( m = 3.8n \) gives the optimal privacy \( p = 0.7661 \) when \( s = 10 \). Another observation is, when \( s \) is large (5 or 10), there always exists a smooth interval of \( m \) near its extreme point that can achieve comparable privacy as the optimal. For example, when \( s = 10 \), the values of \( m \) within the interval \([3.8n, 13.2n]\) achieves privacy that is within 5\% of the optimal privacy 0.7661. This smooth interval for privacy allows us to adjust the value of \( m \) to achieve better measurement results while preserving comparable privacy.

E. Computation Overhead

In our scheme, when a vehicle \( v \) passes an RSE \( R_x \), the vehicle \( v \) only needs to compute two hashes to obtain an index of a random bit in its logical array \( LB_c \), and the RSE \( R_x \) only needs to set 1 bit in its bit array \( B_x \), as described in Section IV-A. Therefore, the computation overhead for the vehicle per RSE as well as the RSE per vehicle are both \( O(1) \). As for the central server, in order to compute a traffic flow size between a pair of locations, it only needs to do a bitwise AND over two \( m \)-bit bit arrays, count the number of ‘0’ in the resulting bit array, and use (15) to compute the MLE estimator. Therefore, the computation overhead for the central server is also \( O(1) \).

V. Simulation

In this section, we evaluate the performance of our measurement scheme through simulations. The simulations are performed under five system parameters, \( n_x, n_y, n_c, s, \) and \( m \). For a pair of RSEs, \( R_x \) and \( R_y \), \( n_x (n_y) \) is the number of vehicles passing by \( R_x \) (\( R_y \)). There are \( n_c \) vehicles passing both \( R_x \) and \( R_y \), which means the true traffic flow size is \( n_c \). \( s \) is the number of bits that each vehicle chooses in its logical bit array, and \( m \) is the number of bits in the bit array of each RSE. In the simulation, we choose the five parameters as follows: \( n_x = n_y = n = 50,000, 100,000, \) or 500,000, and \( n_c \) varies from 1\%\( n \) to 50\%\( n \), with step size of 0.1\%\( n \); \( s = 2, 5, 10 \), and \( m \).
Figure 2. Measurement accuracy controlled by $s$, $n_x = n_y = n = 50,000$, $n_c = [0.01n, 0.5n]$. The x-axis shows true traffic flow sizes, and the y-axis shows the corresponding measured traffic flow sizes. First Plot: $s = 2$; Second Plot: $s = 5$; Third Plot: $s = 10$.

Figure 3. Measurement accuracy controlled by $s$, $n_x = n_y = n = 100,000$, $n_c = [0.01n, 0.5n]$. The x-axis shows true traffic flow sizes, and the y-axis shows the corresponding measured traffic flow sizes. First Plot: $s = 2$; Second Plot: $s = 5$; Third Plot: $s = 10$.

Figure 4. Measurement accuracy controlled by $s$, $n_x = n_y = n = 500,000$, $n_c = [0.01n, 0.5n]$. The x-axis shows true traffic flow sizes, and the y-axis shows the corresponding measured traffic flow sizes. First Plot: $s = 2$; Second Plot: $s = 5$; Third Plot: $s = 10$.

<table>
<thead>
<tr>
<th>Table I</th>
</tr>
</thead>
<tbody>
<tr>
<td>Values for $m$ to achieve optimal $p$ under different $s$.</td>
</tr>
<tr>
<td>$s$</td>
</tr>
<tr>
<td>------</td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
</tbody>
</table>

is chosen to achieve the optimal privacy $p$, as determined in Section IV-D. Table I lists the values for $m$ to achieve optimal $p$ under different values of $s$.

Figure 2, 3, and 4 show our simulation results when $n = 50,000$, $100,000$, and $500,000$, respectively. For each figure, there are three plots, corresponding to the results of three sets of simulations controlled by parameter $s$, where $s = 2, 5,$ and $10$. Each plot shows the measured traffic flow sizes $\hat{n}_c$ (y-axis) with respect to different true traffic flow sizes $n_c$ (x-axis) under a given setting of $n$, $s$, and $m$, where $m$ is chosen as described in Table I so that the optimal privacy is achieved. We also draw the equality line $y = x$ in each plot for reference. Clearly, the closer a point is to the equality line, the more accurate the measurement result. From the figures, one can see that given other parameters, our MLE estimator produces almost perfect results when $s = 2$ (the first plot in Figure 2, 3, and 4). When $s$ becomes larger, the variant for our estimator also becomes larger, producing more points not close to the equality line (the third plot in Figure 2, 3, and 4). Recall that a larger value of $s$ brings better privacy (Table I). For example, the optimal privacy is $0.7661$ when $s = 10$, better than the optimal
privacy of 0.7258 when \( s = 2 \). This implies a tradeoff between the preserved privacy and measurement accuracy. From Section IV-D, we know when \( s \) is large, there always exists a smooth interval of \( m \) near its extreme point that can achieve comparable privacy as the optimal. In reality, one can choose a relatively large value for \( s \) (e.g., 5 or 10), and adjust the value of \( m \) to achieve better measurement results while still preserving comparable privacy as the optimal. Finally, the results are more accurate with larger values of \( n \), which is a natural phenomenon given that the measurement result is obtained through an MLE estimator.

VI. RELATED WORK

A. Transportation Traffic Measurement

In the area of transportation traffic measurement, various prediction models have been proposed to measure “point” traffic statistics, using data recorded by automatic traffic recorders (ATR) installed at road sections. For example, the multiple linear regression model in [2], and the artificial neural network in [3], etc. Those solutions, though elegant, are not appropriate for “point-to-point” transportation traffic measurement. While some “point-to-point” statistics may be inferred from “point” data [4], 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 [12], their approach cannot assure vehicle’s privacy since it uses GPS and Wi-Fi in phones to track locations [13].

B. Network Traffic Measurement

Another branch of research that relates to (but is also significantly different from) ours is network traffic measurement, where researchers have proposed various methods for traffic flow measurement. Though having the same name, their problem is different from ours: to measure the network traffic between two network routers. The solutions can be summarized into two categories. One is indirect estimation based on link load, and network routing, by employing statistical techniques [14] [15]. These methods cannot achieve high accuracy since their estimations are based on the unknown traffic volume. The other is direct measurement by different counting methods [16] [17]. In particular, Li et al. [17] develops a bitmap-based counting method for traffic flow measurement, which is most related to our work. However, all these solutions are not appropriate for our problem, since they measure traffic in network environment where the privacy of packets is not a concern, so counting can be done directly based on the packet IDs. In our problem, the privacy of vehicles is the major concern. Therefore, the solutions must incorporate randomization and de-identification techniques to protect vehicles’ privacy, and do counting based on information that looks totally random.

VII. CONCLUSION

In this paper, we focus on the problem of privacy-preserving “point-to-point” transportation traffic monitoring in intelligent cyber-physical road systems. We formalize “point-to-point” traffic as traffic flows, and quantify privacy as a probability. We propose a novel scheme which allows the collection of aggregate traffic flow data while preserving the optimal privacy of individual vehicles. The proposed scheme utilizes bit arrays to collect data and adopts maximum likelihood estimation (MLE) to obtain the measurement result. The feasibility and scalability of our scheme are shown by both mathematical proofs and simulations.

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