Privacy-Preserving Estimation of *k*-Persistent Traffic in Vehicular Cyber-Physical Systems

Yu-E Sun, He Huang, *Member, IEEE*, Shigang Chen, *Fellow, IEEE*,
You Zhou, Kai Han, *Member, IEEE*, and Wenjian Yang

***Abstract*—Traffic volume estimation is critical to the intelligent transportation engineering. Previous state-of-the-art studies mainly focus on measuring two types of traffic volume: "point" traffic (i.e., the number of vehicles passing a given location) and "point-to-point" traffic (i.e., the number of vehicles traversing between two given locations) during each measurement period. In this paper, we extend this line of research from single-period to multiple periods and study new problems of estimating the number of *k*-persistent vehicles that pass a location or two different locations in at least *k*-out-of-*t* predefined measurement periods. We propose two novel *k*-persistent traffic estimators with privacy-preserving for the point and point-to-point traffic models, respectively. Through theoretical analysis, we prove that our solution can solve more general traffic measurement problems and employ stronger privacy preserving, i.e., ϵ-differential privacy, than the existing studies. We also demonstrate the effectiveness and the accuracy of the proposed estimators through extensive experiments based on real transportation traffic flows in Shenzhen, China for five consecutive working days. The numerical results show that the estimators can achieve a trade-off between the estimation accuracy and privacy preservation through proper parameter setting.**

***Index Terms*—Persistent traffic, privacy, traffic measurement, vehicular networks.**

Manuscript received March 6, 2019; revised April 30, 2019; accepted May 4, 2019. Date of publication May 15, 2019; date of current version October 8, 2019. This work was supported in part by the National Science Foundation (NSF) under Grant NeTS 1719222 and Grant NSF NeTS 1718708, in part by the National Natural Science Foundation of China (NSFC) under Grant 61672369, Grant 61873177, Grant 61572342, and Grant 61772491, in part by the Natural Science Foundation of Jiangsu Province under Grant BK20161258 and Grant BK20161256, in part by the Florida Cybersecurity Center, and in part by the Open Project of Anhui Provincial Key Laboratory of Network and Information Security under Grant AHNIS2019003. *(Corresponding author: He Huang.)*

Y.-E. Sun is with the School of Rail Transportation, Soochow University, Suzhou 215131, China (e-mail: sunye12@suda.edu.cn).

H. Huang is with the School of Computer Science and Technology, Soochow University, Suzhou 215006, China, and also with the Anhui Provincial Key Laboratory of Network and Information Security, Wuhu 240002, China (e-mail: huangh@suda.edu.cn).

S. Chen is with the Department of Computer and Information Science and Engineering, University of Florida, Gainesville, FL 32611 USA (e-mail: sgchen@cise.ufl.edu).

Y. Zhou is with the Department of Computer and Information Science and Engineering, University of Florida, Gainesville, FL 32611 USA, and also with Google, Inc., Mountain View, CA 94043 USA (e-mail: youzhou@google.com).

K. Han is with the School of Computer Science and Technology, University of Science and Technology of China, Hefei 230027, China (e-mail: hankai@ustc.edu.cn).

W. Yang is with the School of Computer Science and Technology, Soochow University, Suzhou 215006, China (e-mail: 20164227034@stu.suda.edu.cn).

Digital Object Identifier 10.1109/JIOT.2019.2916349

I. Introduction

TRAFFIC volume measurement is crucial to intelligent transportation engineering. Accurate estimation of traffic data provides important input for the transportation authority to design cost-effective investment plans. In recent years, new vehicle-to-infrastructure communication technologies were proposed to support vehicular cyber-physical systems (VCPSs), which enable automatic traffic data collection for urban transportation [1]–[6]. An emerging trend in traffic analysis is to integrate wireless communications and computing devices into VCPS for better road safety and driving experience [7], [8]. For instance, the dedicated short range communications (DSRC) standard under IEEE 802.11p [9] enables wireless data exchanges between vehicles and road-side units (RSUs), making the traffic data collection more efficient and powerful. Some automobile industry magnates (such as Toyota and Lexus) plan to start deployment of DSRC systems on vehicles sold in the United States starting from 2021. By allowing VCPS to collaborate more broadly and effectively through DSRC technology, we can help drivers realize a future with zero fatalities from crashes, better traffic flow, and less congestion. To achieve these goals, accurate traffic volume estimation is necessary and critical [10].

There are two types of traffic measurement for "point" traffic and "point-to-point" traffic, respectively. The point traffic volume refers to the number of vehicles passing a specific location, while the point-to-point traffic volume refers to the number of (common) vehicles that traverse between two given locations. By measuring these two types of traffic during each measurement period (e.g., a day), some prior studies build mathematical models (e.g., the support vector regression model or the Bayesian model) based on historical data for traffic prediction [11]–[13]. Privacy is a big concern in traffic measurement, which means the information collected by RSUs can only be used to gather traffic statistics without tracking individual vehicles [14]. Any information that can serve the purpose of identification, such as vehicle IDs or fixed numbers, should not be contained in the collected data.

To protect the privacy of vehicles, many privacy-preserving traffic volume estimators were proposed [14]–[17]. These studies store the traffic information in a privacy-preserving data structure and can measure the point or multipoint traffic volume in one measurement period with high estimation accuracy. However, we need to find the "core" traffic volume in some real-world applications. The core traffic is the traffic persistently passes a location or two different locations in all or most

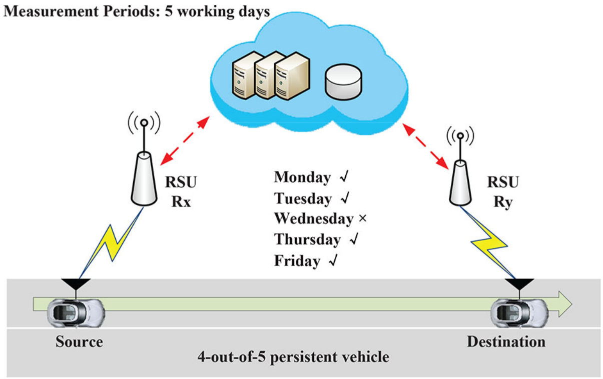


Fig. 1. Example of 4-persistent traffic.

measurement periods of interest. For instance, the solution for traffic congestion caused by core traffic may be different from the congestion caused by transient traffic. To measure the core traffic volume, we need to estimate the number of vehicles persistently pass the locations of interest, i.e., the persistent traffic volume. To the best of our knowledge, only the existing work [18] targets on the measurement of persistent traffic volume with privacy preserving. However, the previous work assumes the persistent traffic only consists of the vehicles that pass all the measurement periods, which is obviously too strong as a hypothesis.

In this paper, we study a more general problem of estimating the number of vehicles that pass a location or two different locations in at least k-out-of-t ($k \leq t$) periods of interest, where k and t are two parameters specified in user queries. An example query may be on how many vehicles travel between two locations in at least four days during Monday through Friday in a week, as shown in Fig. 1. The previous state-of-the-art work [18] only solves a special case of this problem, i.e., $k = t$, which is simpler to solve. For instance, the vehicle only absent in Wednesday (see the example in Fig. 1) is not a persistent vehicle based on the definition in [18], but a 4-persistent vehicle with the definition in this paper. Vehicles that appear in all periods of concern are certainly persistent traffic. However, the vehicles pass a location or two different locations in most of the periods of interest should be considered in the core traffic that persistently shows up, rather than belonging to the transient traffic that comes and goes. In general, such k-persistent traffic may be absent for any $(t-k)$ periods, but they are considered as part of the core traffic. Hence, the ability to answer any k-out-of-t queries, with arbitrarily chosen k and t, will provide users with great flexibility in investigating transportation traffic, which the method in [18] cannot support.

There exist efficient mechanisms to support differential privacy (DP), but none of them can be directly adopted in the context of transportation traffic measurement in this paper. In particular, RAPPOR is designed for protecting the reports from clients in crowdsourcing with DP [19]. It stores a noisy representation of each clients report, consisting of the client ID and a 0/1 value which has a certain probability of being flipped. In our context, we do not know the set of vehicles *a priori*, and we cannot let the passing vehicle transmit its ID to an RSU, which would break the location privacy. Therefore, the method of RAPPOR does not apply in our setting. In each measurement period, every RSU produces an anonymous bitmap (called traffic record) that encodes the set of passing vehicles. More specifically, when a vehicle encounters an RSU, with a certain probability, it sets a pseudo-randomly chosen bit in the RSU's bitmap, where the probability is controlled by a DP parameter ϵ. To estimate the volume of k-persistent point traffic, we derive an estimator through set theory based on different combinations of traffic records produced from a given RSU over multiple measurement periods. To estimate the point-to-point k-persistent traffic volume, we first perform bitmap expansion and then perform bitwise-AND over each pair of traffic records produced from two given RSUs during the same period, which results in a set of combined traffic records that contain the common traffic between the two locations. We then perform bitwise-OR over these combined traffic records and derive an estimator for k-persistent common (point-to-point) traffic through probabilistic analysis. The main contributions of this paper are listed as follows.

1) We propose two novel estimators to measure the k-persistent traffic point volume and point-to-point volume, separately. As far as we know, we are the first to study the k-persistent traffic volume estimation problem in transportation.
2) We prove that the proposed estimators can achieve ϵ-differential privacy (ϵ-DP), which means our estimators can provide stronger privacy preserving than the previous studies.
3) To evaluate the performance of our proposed estimators, we conduct extensive experiments by using the real transportation traffic traces in Shenzhen, China during five consecutive working days. The experimental results show the effectiveness of the proposed estimator in producing high accuracy measurement and meanwhile achieving privacy preservation.

II. Preliminaries

We consider an intelligent transportation system which includes RSUs, vehicles, and a central server. Vehicles and RSUs each have a unique ID. A vehicle communicates with an RSU through DSRC when they encounter each other [9]. RSUs are deployed at locations of interest, such as some major intersections. They record information of passing vehicles in data structures called *traffic records*, which will be sent to the central server periodically. The server can respond to the traffic volume queries.

A. Problems of k-Persistent Traffic Measurement

Consider a certain number t of measurement periods and a given RSU at location L. We define a *k-persistent vehicle* as one that passes location L in k or more out of the t periods of interest, where k is a user parameter specified in a query. For example, we may set each working day in a week as one measurement period. Thus, a 3-persistent vehicle is the car that passes location L in 3 or more working days during the week. The first problem called *k-persistent point traffic measurement* is formally defined as follows.

Definition 1 (k-Persistent Point Traffic Measurement): It is to estimate the volume (number) of k-persistent vehicles observed at location L based on the traffic records from the RSU at L.

Given an arbitrary pair of RSUs at locations L and L', we define a *common vehicle* as one that passes both L and L' during the same measurement period. We further define a *k-persistent common vehicle* as one that is a common vehicle in at least k-out-of-t measurement periods of interest. Our second problem, called *k-persistent common traffic measurement* is defined as follows.

Definition 2 (k-Persistent Common Traffic Measurement): It is to estimate the volume (number) of k-persistent common vehicles based on the traffic records from locations L and L'.

B. Attack Model

We consider three types of attacks. In this first type of attack, the adversary is assumed to be able to install fake RSUs at chosen locations to communicate with vehicles in order to obtain traffic records. To guard against these attackers, we must ensure that vehicles only interact with authorized RSUs. This is enforced through authentication based on PKI. Each RSU broadcasts beacons, each carrying its public-key certificate, which was obtained from a trusted third party. When a vehicle receives a beacon, it uses its preinstalled public key of the trusted third party to verify the certificate. If not successful, the vehicle will not communicate with the RSU further; otherwise, it performs authentication with the RSU using the latter's public key obtained from the verified certificate. After successful authentication, it communicates with the RSU for traffic measurement, with all data exchanges encrypted. Rogue RSUs may be deployed by nonauthorities; they will fail authentication through previous processes and the vehicles will not interact with them.

We adopt a semi-trusted model for transportation authorities, which are assumed to have good faith in implementing the proposed privacy-preserving methods since their goal is to gather traffic statistics, not to track people. Thus, the RSUs in the transportation system will communicate with passing vehicles and perform all required operations as proposed.

In the second type of attack, the adversary is assumed to have access to the traffic records from the authorized RSUs. For example, the transportation authority has such access, and the police may gain access after proper authorization. Our system design wants to protect drivers privacy against even these entities (who gain access to the records from the RSUs) from exploiting the information to track individual vehicles and the whereabouts of the ordinary drivers. If a hypothetical system design requires all vehicles to transmit their unique identifiers to each encountered RSU, then these recorded identifiers can be used to track the trajectory of any vehicle. In order to prevent this from happening, it is highly desirable that a vehicle should not transmit its unique ID, nor transmit any other fixed number to the RSUs. We assume that an anonymous MAC protocol, such as SpoofMAC [20] can be used to support privacy preservation such that the MAC address of a vehicle is not fixed. With such a protocol, before a vehicle communicates with an RSU, it picks a temporary MAC address randomly from a large space for one-time use, preventing the MAC address from serving as an identifier of the vehicle.

In the third type of attack, not only does the adversary (such as police) have access to all traffic records from RSUs, but also it has additional side information about the appearance of a vehicle at a certain location—for instance, the vehicle may be stopped by a police car for speeding at the location. If there are no other vehicles around, the police can associate the vehicle identity with the communication from the vehicle to the RSU at the location. Our system must prevent such association from revealing the presence of the vehicle at other locations. To meet this requirement, the data sent from the vehicle to different RSUs must differ probabilistically in order to present such data from being used as vehicle identifier.

C. Performance Metrics

We evaluate our traffic measurement methods based on two performance metrics: 1) estimation accuracy and 2) preserved privacy.

1. *Estimation Accuracy:* Let n^* be the actual traffic volume of k-persistent point/common traffic, and $\hat{n}^*$ be its estimated value. We use the absolute error, $|\hat{n}^* - n^*|$, to measure the estimation accuracy. The smaller, the better.

2. *Preserved Privacy:* For the second and third types of attackers, we provide two-level privacy preservation in our transportation traffic measurement. The first level is to protect the point privacy of vehicles, which is designed for the second type of attacker and supported by a ϵ-DP mechanism. The proposed DP mechanism ensures that any potential tracker has a very limited chance to identify any individual vehicle passing a location only based on the information of traffic records. However, the point privacy of a vehicle v may be leaked through other ways. For example, the vehicle may be stopped by a police car for speeding at the location where there are no other vehicles around. When the point privacy of a vehicle has been leaked, then another privacy concern is how much additional information the traffic records will leak to reveal the trajectory of v. Thus, the second level privacy preservation we provide is designed for the third type of attacker, which is to protect the trajectory of vehicles while the point privacy is leaked by accident.

First, we consider the first level privacy: point privacy. DP has recently emerged as a leading standard for privacy guarantees on statistical databases. Thus, we propose a randomized mechanism, which can achieve ϵ-DP, to protect the point privacy of vehicles. We say two sets S and S' are adjacent if their symmetric difference contains at most one element. The standard definition of ϵ-DP is given below [21], [22].

Definition 3 (ϵ-DP Privacy): A randomized mechanism $\mathcal{M} : \mathcal{S} \to \mathcal{R}$ provides ϵ-DP privacy if for all adjacent inputs $S, S' \in \mathcal{S}$ and all subsets $D \subseteq \mathcal{R}$, we have

$$\frac{\Pr[\mathcal{M}(S) \in D]}{\Pr[\mathcal{M}(S') \in D]} \leq e^{\epsilon} \quad (1)$$

where ϵ is a privacy parameter referred as the privacy budget. The smaller ϵ means the closer the distributions $\mathcal{M}(S)$ and $\mathcal{M}(S')$ are, and in turn a stronger privacy preservation.

Then, we consider the second level privacy: trajectory privacy.

TABLE I
MAIN NOTATIONS USED IN THIS PAPER

Symbol	**Symbol Meaning**
$n^*, \hat{n}^*$	the actual traffic volume, the estimated value of n^*
p''	the probability for a vehicle encoded by a passing RSU
K_v	the private key of vehicle v
B_i, S_i	the traffic record/encoded vehicle set at location L during the ith period
$m(m')$	the number of bits in bitmap B (B')
$n_{i,12\ldots t}$ $(\hat{n}_{i,12\ldots t})$	the number (estimated number) of vehicles that pass location L in exactly i periods out of period 1, 2,..., t
S_i^c	the set of encoded common vehicles that pass L and L' in period i
$S_{G,i}^c$	the set of common vehicles that encoded by bitmap G_i
S_i^v	the set of virtual vehicles in period i
$n_{i,12\ldots t}$ $(\hat{n}_{i,12\ldots t})$	the number (estimated number) of common vehicles that pass location L and L' in exactly i periods out of period 1, 2, ..., t
$n_{k,12\ldots t}^{*,c}$ $(\hat{n}_{k,12\ldots t}^{*,c})$	the number (estimated number) of common vehicles that pass location L and L during at least k periods out of period 1, 2, ..., t

Definition 4 (Trajectory Privacy): Consider two locations L and L'. If the point privacy of vehicle v has leaked in location L, a trajectory privacy-preserving method should ensure that there is only a limited chance to identify any part of the trajectory of vehicle v from the traffic records.

To achieve this, we introduce probabilistic noise-to-information ratio to characterize the level of trajectory privacy protection in this case. As defined in [18], the probabilistic noise-to-information ratio is $(p/[p' - p])$, where p is the probability that the traffic records will show that v has passed both locations even though v did not, and p' is the probability that the traffic records will show that v had passed both locations when v actually did so. Note that p is the noise term that is introduced by other vehicles and p' includes the noise contribution p. Thus, $p' - p$ is the information that can be used to identify the trajectory of vehicles. To increase the privacy-preserving level, we expect the noise p approaches to one and the information $p' - p$ approaches to zero. Thus, a larger probabilistic noise-to-information ratio can better protect the trajectory privacy of vehicles. To facilitate reading, we list some of the important notations in Table I.

III. Encoding Vehicles in Traffic Records

In this section, we describe how to encode vehicles in traffic records.

A. Traffic Records

Consider an arbitrary RSU installed at a certain location. It uses a bitmap B of m bits to store the traffic record. Each vehicle that passes the RSU has a certain probability p'' to be encoded by a bit, which is pseudo-randomly selected from B in a way that masks the identity of the vehicle yet records the presence of a vehicle for statistical analysis. We point out that the size m of the bitmap B may differ at different RSUs since they encounter different traffic volume.

There is a functional relationship between the number of ones (or zeros) in B and the number of vehicles encoded—the more the number of vehicles is, the more ones B will have. Through this relationship, we can estimate the number of vehicles from the number of ones (or zeros) in B. The problem of k-persistent traffic measurement will be more difficult because we must combine the traffic records from different RSUs or from different periods of one RSU to find out the number of k-persistent vehicles. Moreover, to support privacy, we want to mix the information from different vehicles in the traffic records: (S1) Different vehicles may be probabilistically encoded by the same bit in a traffic record B. (S2) The same vehicle may be encoded by bits of different indices at different locations (RSUs). Together, they break the one-to-one mapping between vehicles and bits in traffic records.

B. Encoding Procedure

Consider an arbitrary RSU. At the beginning of each measurement period, the RSU resets all bits in bitmap B to zeros. It then broadcasts beacons in preset intervals, such as ten times per second, ensuring that each passing vehicle is able to receive a beacon, which carries the RSU's location L, its public-key certificate, and it bitmap size m. After a vehicle v receives this beacon, it computes the hash output $h_v = H(K_v)$ mod r, where H is a cryptographic one-way hash function, K_v is a private key known only by the vehicle, r is a constant. If $h_v/r < p''$, v verifies the certificate and uses the public key to authenticate the RSU, where p'' is the system parameter controlled by the privacy parameter ϵ; otherwise, v will not communicate with the RSU (thus will not be encoded by the RSU). After verifying that the RSU is from a trusted authority, the vehicle computes the following hash output:

$$h'_v = H(K_v \oplus C[H(L) \oplus K_v \mod s]) \mod m$$

where C is an array of s randomly selected constants. The vehicle transmits h'_v to the RSU, which will encode the vehicle by the h'_vth bit in B, i.e., set $B[h'_v]$ to 1. At the end of each measurement period, the RSU will send the content of B as its traffic records to the central server, where users can submit queries for k-persistent traffic.

C. Design Considerations

The index h'_v produced from a vehicle is not predictable because the private key K_v is not known by others. Moreover, the array C are also only known to the vehicle. Therefore, it is not feasible for an eavesdropper to figure out the identity of a vehicle from the overhead value of h'_v.

On the one hand, h'_v is a function of L such that its value varies at different locations. Thus, the vehicle can be encoded by different locations at different RSUs, which conforms to the statement of (S2) in the previous section. The system parameter s controls the number of different values that h'_v can take. On the other hand, many vehicles may pass an RSU during a measurement period. Due to the random selection of bits to set, different vehicles may choose the same bit as a result of hash collision, which conforms to the statement of (S1). Such variation and mixing in vehicle encoding can help preserve privacy and The index h'_v produced from a vehicle is not predictable because the private key K_v is not known by others. Moreover, the array C are also only known to the vehicle. Therefore, it is not feasible for an eavesdropper to figure out the identity of a vehicle from the overheard value of h'_v.

On the one hand, h'_v is a function of L such that its value varies at different locations. Thus, the vehicle can be encoded by different locations at different RSUs, which conforms to the statement of (S2) in the previous section. The system parameter s controls the number of different values that h'_v can take. On the other hand, many vehicles may pass an RSU during a measurement period. Due to the random selection of bits to set, different vehicles may choose the same bit as a result of hash collision, which conforms to the statement of (S1). Such variation and mixing in vehicle encoding can help preserve privacy and make it harder for a tracker (including the authority) to determine the trajectory of any vehicle definitively.

IV. Measurement of k-Persistent Point Traffic

In this section, we provide the details of how to derive the estimator for the k-persistent point traffic at a given location.

A. Notations

Given a set of bitmaps $\mathcal{B} = \{B_1, \ldots, B_t\}$ that records the transportation traffic at a certain location L during t measurement periods, our goal is to estimate the k-persistent point traffic using this bitmap set $\mathcal{B}$. The size of the bitmap in each measurement is fixed to m, which is preset as $2^{\lceil \log_2(\bar{n}\times f)\rceil}$, where $\bar{n}$ is the expected traffic volume at L based on historical average, and f is a system-wide load factor that specifies the ratio of the bitmap size m and the expected traffic volume $\bar{n}$.

Let $n_{i,12\ldots t}$, $i \leq k$, be the number of vehicles that pass location L in exactly i periods out of period 1, period 2,..., and period t. As a special case, $n_{t,12\ldots t}$ denotes the number of persistent vehicles that pass L in all t periods. Then, the volume of k-persistent point traffic, denoted as $n^*_{k,12\ldots t}$, is equal to $\sum_{i=k}^{t} n_{i,12\ldots t}$. We also define a more general notation $n_{j,i_1 i_2\ldots i_k}$ as the number of persistent vehicles that pass L in exactly j periods out of period i_1, period i_2,..., and period i_k, where $1 \leq i_1 < i_2 < \cdots < i_k \leq t$ and $j \leq k$. As a special case, $n_{k,i_1 i_2\ldots i_k}$ is the number of persistent vehicles that pass L in period i_1, period i_2,..., and period i_k.

B. Derivation of $n_{k,i_1 i_2\ldots i_k}$

Before deriving an estimator for $n^*_{k,12\ldots t}$ on k-persistent point traffic, $1 \leq k \leq t$, we first give a method to estimate $n_{k,i_1 i_2\ldots i_k}$. Let S_i be the set of encoded vehicles that passes location L in the ith period (or referred as period i). Since each vehicle has a probability of p'' to be encoded by bitmaps, we have $n_{k,i_1 i_2\ldots i_k} = |S_{i_1} \cap S_{i_2} \cap \cdots \cap S_{i_k}|/p''$. To derive $n_{k,i_1 i_2\ldots i_k}$, we first join the information from the bitmaps by performing bitwise OR directly among them, and the combined bitmap is denoted as $F_{i_1 i_2\ldots i_k}$. Its lth bit is denoted as $F_{i_1 i_2\ldots i_k}[l]$, for $1 \leq l \leq m$. Then, we have $F_{i_1 i_2\ldots i_k}[l] = B_{i_1}[l] \vee B_{i_2}[l] \vee \cdots \vee B_{i_{k-1}}[l] \vee B_{i_k}[l]$.

For an arbitrary bit in $F_{i_1 i_2\ldots i_k}$, its value can be modeled as a random binary variable. Let $\Pr\{F_{i_1 i_2\ldots i_k}[l] = 0\}$ be the probability of $F_{i_1 i_2\ldots i_k}[l] = 0$, which can be derived recursively. Obviously, both $B_{i_k}[l]$ and $F_{i_1 i_2\ldots i_{k-1}}[l]$ are equal to zero if and only if $F_{i_1 i_2\ldots i_k}[l] = 0$, for $1 \leq j \leq k$. Then, we have

$$\begin{aligned}&\Pr\{F_{i_1 i_2\ldots i_k}[l] = 0\}\\&\quad = \Pr\{B_{i_k}[l] = 0 | F_{i_1 i_2\ldots i_{k-1}}[l] = 0\} * \Pr\{F_{i_1 i_2\ldots i_{k-1}}[l] = 0\}.\end{aligned} \tag{2}$$

Note that no vehicle exists in set $S_{i_1} \cup S_{i_2} \cup \cdots \cup S_{i_{k-1}}$ will hash to the lth bit when $F_{i_1 i_2\ldots i_{k-1}}[l] = 0$. Then, $\Pr\{B_{i_k}[l] = 0 | F_{i_1 i_2\ldots i_{k-1}}[l] = 0\}$ should be the probability for none of the vehicle that in set $S_{i_k} - (S_{i_1} \cup S_{i_2} \cup \cdots \cup S_{i_{k-1}})$ hashes to the lth bit of B_{i_k}. Based on the principle of *inclusion and exclusion* in set theory, we have

$$\begin{aligned}&|S_{i_k} - (S_{i_1} \cup S_{i_2} \cup \cdots \cup S_{i_{k-1}})|\\&\quad = |S_{i_k}| - \sum_{1\leq p_1<k} |S_{i_{p_1}} \cap S_{i_k}|\\&\qquad + \cdots + (-1)^q \sum_{1\leq p_1<p_2<\cdots<p_q<k} |S_{i_{p_1}} \cap S_{i_{p_2}} \cap \cdots \cap S_{i_{p_q}} \cap S_{i_k}|\\&\qquad + \cdots + (-1)^k |S_{i_1} \cap S_{i_2} \cap \cdots \cap S_{i_k}|.\end{aligned} \tag{3}$$

The probability for any vehicle in $S_{i_k} - (S_{i_1} \cup S_{i_2} \cup \cdots \cup S_{i_{k-1}})$ not being hashed to the lth bit of B_{i_k} is $1 - (1/m)$. Then, we have

$$\begin{aligned}&\Pr\{B_{i_k}[l] = 0 | F_{i_1 i_2\ldots i_{k-1}}[l] = 0\}\\&\quad = \left(1 - \frac{1}{m}\right)^{\left|S_{i_k} - \left(S_{i_1} \cup S_{i_2} \cup \cdots \cup S_{i_{k-1}}\right)\right|}.\end{aligned} \tag{4}$$

Substituting (4) to (2), we have

$$\begin{aligned}&\Pr\{F_{i_1 i_2\ldots i_k}[l] = 0\}\\&\quad = \left(1 - \frac{1}{m}\right)^{\left|S_{i_k} - \left(S_{i_1} \cup S_{i_2} \cup \cdots \cup S_{i_{k-1}}\right)\right|} * \Pr\{F_{i_1 i_2\ldots i_{k-1}}[l] = 0\}.\end{aligned} \tag{5}$$

Next, we illustrate a theorem as follows.

Theorem 1: $\Pr\{F_{i_1 i_2\ldots i_k}[l] = 0\} = E(V_{0,i_1 i_2\ldots i_k})$, where $V_{0,i_1 i_2\ldots i_k}$ is a random variable for the fraction of bits in $F_{i_1 i_2\ldots i_k}$ that are zeros, and $E(V_{0,i_1 i_2\ldots i_k})$ is the expected value of $V_{0,i_1 i_2\ldots i_k}$.

Proof: Since $V_{0,i_1 i_2\ldots i_k}$ is the fraction of bits in $F_{i_1 i_2\ldots i_k}$ that are zeros, we have

$$V_{0,i_1 i_2\ldots i_k} = \frac{1}{m}\sum_{l=1}^{m} I_{l,0} \tag{6}$$

where $I_{l,0}$ is an indicator variable, whose value is 1 when $F_{i_1 i_2\ldots i_k}[l] = 0$ and 0 otherwise. Clearly, $\forall 1 \leq l \leq m$, $E(I_{l,0}) = \Pr\{F_{i_1 i_2\ldots i_k}[l] = 0\}$. Hence,

$$\begin{aligned}E(V_{0,i_1 i_2\ldots i_k}) &= \frac{1}{m}\sum_{l=1}^{m} E(I_{l,0})\\&= \frac{1}{m}\sum_{l=1}^{m} \Pr\{F_{i_1 i_2\ldots i_k}[l] = 0\} = \Pr\{F_{i_1 i_2\ldots i_k}[l] = 0\}.\end{aligned} \tag{7}$$

Algorithm 1: Estimator for the t-Persistent Traffic

```
for $k = 1$ to $t$ do
    for each combination of $k$ measurement periods $i_1, i_2, ..., i_k$ do
        Measure $V_{0,i_1i_2\ldots i_k}$ by performing bitwise OR among all the $k$ bitmaps;
        Compute the value of $\hat{n}_{k,i_1i_2\ldots i_k}$ according to Equation (8);
    end for
end for
Return $\hat{n}_{t,12\ldots t}$;
```

Then, we have $E(V_{0,i_1i_2\ldots i_k}) = \Pr\{F_{i_1i_2\ldots i_k}[l] = 0\}$, which finishes our proof. ■

Combining (3), (5), (7), and replacing $E(V_{0,i_1i_2\ldots i_k})$ by the instant value $V_{0,i_1i_2\ldots i_k}$ measured from $F_{i_1i_2\ldots i_k}$, we have

$$\hat{n}_{k,i_1i_2\ldots i_k} = \frac{a\ln\left(1-\frac{1}{m}\right) + \ln V_{0,i_1i_2\ldots i_{k-1}} - \ln V_{0,i_1i_2\ldots i_k}}{(-1)^k \ln\left(1-\frac{1}{m}\right)p''} \qquad (8)$$

where $a = \sum_{q=0}^{k-2}(-1)^q \sum_{1\le p_1<\cdots<p_q\le k-1} n_{q+1,i_{p_1}\ldots i_{p_q}i_k}p''$.

We invoke the above estimator in the order of $k = 1, 2, \ldots,$ and t. For a specific value of k, the computation of $\hat{n}_{k,i_1i_2\ldots i_k}$ requires the values of $n_{q+1,i_{p_1}i_{p_2}\ldots i_{p_q}i_k}$, $1 \le p_1 < p_2 < \cdots < p_q \le k-1, 0 \le q \le k-2$, which are computed through (8). We also need the value of $V_{0,i_1i_2\ldots i_{k-1}}$, which can be measured earlier when we estimate the value of $\hat{n}_{k-1,i_1i_2\ldots i_{k-1}}$. For a given combination of k bitmaps that record the transportation traffic at L in periods $i_1, i_2, \ldots, i_k$, we first join the information of these bitmaps by performing bitwise OR to measure the value of $V_{0,i_1i_2\ldots i_k}$. Then, we estimate the value of $\hat{n}_{k,i_1i_2\ldots i_k}$ through (8). This iterative process is carried out by Algorithm 1 to estimate the volume of t-persistent point traffic.

C. Estimator for k-Persistent Point Traffic

Consider an arbitrary vehicle that passes location L in exactly k measurement periods, it must be recorded by one subset of k bitmaps from $\mathcal{B}$. There are C_t^k ways to form such a subset. For each subset of k bitmaps from $\mathcal{B}$, we can measure the k-persistent traffic volume by Algorithm 1. However, $\sum_{1\le i_1<i_2<\cdots<i_k\le t} n_{k,i_1i_2\ldots i_k}$ is larger than the actual volume of k-persistent traffic since all the $(k+1)$-persistent vehicles were double counted. For a vehicle passing location L in k measurement periods, it will be double counted C_i^k times. Hence, we have

$$\sum_{1\le i_1<i_2<\cdots<i_k\le t} n_{k,i_1i_2\ldots i_k} = n_{k,12\ldots t} + \sum_{i=k+1}^{t} C_i^k n_{i,12\ldots t}. \qquad (9)$$

Replacing $n_{k,i_1i_2\ldots i_k}$ and $n_{i,12\ldots t}$ with their estimated values $\hat{n}_{k,i_1i_2\ldots i_k}$ and $\hat{n}_{i,12\ldots t}$ separately, we have the following estimator:

$$\hat{n}_{k,12\ldots t} = \sum_{1\le i_1<i_2<\cdots<i_k\le t} \hat{n}_{k,i_1i_2\ldots i_k} - \sum_{i=k+1}^{t} C_i^k \hat{n}_{i,12\ldots t}. \qquad (10)$$

Algorithm 2: Estimator for the k-Persistent Point Traffic

```
for $j = t$ to $k$ do
    for each combination of $j$ measurement periods do
        Compute the value of $\hat{n}_{j,i_1i_2\ldots i_j}$ by Algorithm 1;
    end for
    if $j == t$ then
        $\hat{n}_{t,12\ldots t} = \hat{n}_{t,i_1i_2\ldots i_t}$;
    else
        Compute the value of $\hat{n}_{j,12\ldots t}$ by Equation (10);
    end if
end for
Set $\hat{n}^*_{k,12\ldots t} = \sum_{j=k}^{t} \hat{n}_{j,12\ldots t}$;
```

Algorithm 2 carries out an iterative process to estimate the k-persistent traffic volume, denoted as $\hat{n}^*_{k,12\ldots t}$. As j decreased from t to k, we measure the traffic volume $\hat{n}_{j,12\ldots t}$ in each iteration. For a specific j, we first compute the j-persistent traffic volume for each combination of j periods. Then, we compute $\hat{n}_{j,12\ldots t}$ through (10), which is in turn used in the next iteration to compute $\hat{n}_{j-1,12\ldots t}$ by (10). Finally, we get the k-persistent traffic volume by computing $\hat{n}^*_{k,12\ldots t} = \sum_{j=k}^{t} \hat{n}_{j,12\ldots t}$.

V. Measurement of k-Persistent Common Traffic

In this section, we derive the estimator for k-persistent common traffic.

A. Notations

Consider two locations of interest, L and L'. The two sets of bitmaps measured at these two locations during the t periods are denoted as $\mathcal{B} = \{B_1, \ldots, B_t\}$ and $\mathcal{B}' = \{B'_1, \ldots, B'_t\}$, respectively. We want to estimate the k-persistent common traffic as defined in Section II-A. Let m (m') be the size of the bitmaps in $\mathcal{B}$ ($\mathcal{B}'$), $n^c_{i,12\ldots t}$ be the number of common vehicles that pass location L and L' in exactly i periods out of period 1, period 2,..., period t. Then, the volume of k-persistent common traffic, denoted as $n^{*,c}_{k,12\ldots t}$, is equal to $\sum_{i=k}^{t} n^c_{i,12\ldots t}$. We also define a more general notation $n^c_{j,i_1i_2\ldots i_k}$ as the number of persistent common vehicles that pass L in exactly j periods out of period i_1, period i_2,..., period i_k, where $1 \le i_1 < i_2 < \cdots < i_k \le t$ and $j \le k$. As a special case, $n^c_{k,i_1i_2\ldots i_k}$ is the number of persistent vehicles that pass L in period i_1, period i_2,..., and period i_k.

B. Joining Bitmaps of Different Locations Through Expansion

To find the common traffic encoded by bitmaps of two locations, we need to join the information from the bitmaps in $\mathcal{B}$ and $\mathcal{B}'$. If all bitmaps have the same size, we can combine them by performing bitwise operation directly on them. However, the bitmap size is determined by the expected traffic volume at each location based on the historical average, which may be varied widely at different locations. Without loss of generality, we assume that $m \le m'$. To circumvent this problem, we expand the size of a bitmap B_i by replicating it multiple times until its size reaches m', i.e., expand the size of B_i from m to m' by replicating it (m'/m) times. Since the size of bitmaps is powers of 2, such expansion is always possible. The expanded bitmap is denoted as E_i. If $m = m'$, the expanded bitmap is simply B_i. After bitmap expansion, we join E_i and B'_i $(1 \leq i \leq t)$ by performing bitwise AND, and the result bitmap is denoted as G_i.

C. Derivation of $n^c_{k,i_1 i_2 \ldots i_k}$

Similar with the point k-persistent traffic measurement, we first give a method to estimate $n^c_{k,i_1 i_2 \ldots i_k}$. Let S_i (S'_i) be the set of encoded vehicles that pass location L (L') in period i. Then, $S^c_i = S_i \cap S'_i$ is the set of encoded common vehicles that pass both location L and L' in period i. Since each common vehicle has a probability of p'' to be encoded by bitmaps, we have $n^c_{k,i_1 i_2 \ldots i_k} = |S^c_{i_1} \cap S^c_{i_2} \cap \cdots \cap S^c_{i_k}|/p''$.

For point-to-point traffic measurement, a common vehicle $v \in S^c_i$ may set bits in E_i and B'_i at different indices. It only has a certain probability to set bits in E_i and B'_i at the same index, which makes this problem much harder than the point traffic measurement problem. Due to hash collision, two different vehicles may choose the same index at different locations, which will introduce noise of ones in G_i. If we abstract two different vehicles that introduce a noise of one in G_i as a virtual vehicle, G_i encodes both virtual vehicles and part of common vehicles. We use S^v_i to denote the set of virtual vehicles in period i. Based on (11), we compute the number of independent vehicles that would have produce the bitmap G_i

$$N^c_i = \frac{\ln V_{G_i,0}}{\ln\left(1 - \frac{1}{m'}\right)} \tag{11}$$

where $V_{G_i,0}$ is the fraction of zeros in G_i. Since we perform bitwise AND among E_i and B'_i to get G_i, a common vehicle v will be encoded by G_i only when it sets bits in E_i and B'_i at the same index. v has a probability of $(1/s)$ to choose the same bit from its logical bit array L_v in both L and L'. The probability for v to choose a separate bit randomly from C in L' is $(1 - [1/s])$, and further choose the same bit in B'_i is $(1/m')$ due to hash collision. Thus, a common vehicle v has a probability of $(1/s) + (1 - [1/s])[1/m']$ to choose the same bit in both locations, and introduce a bit of one in G_i. The number of common vehicles in S^c_i is $n^c_{1,i}$. Then, we have

$$N^c_i = n^c_{1,i} p'' \left(\frac{1}{s} + \left(1 - \frac{1}{s}\right)\frac{1}{m'}\right) + n^v_{1,i} \tag{12}$$

where $n^v_{1,i}$ is the number of virtual vehicles in period i. In the following, we will show how to filter the noise introduced by virtual vehicles from G_i, which will in turn an estimator for $n^c_{1,i}$.

Consider an arbitrary bit $G_i[l]$. The probability for a virtual vehicle to choose this bit is $(1/m')$. Suppose there are $n^v_{1,i}$ virtual vehicles in period i. The probability for none of them to choose this bit to set is $(1 - [1/m'])^{n^v_{1,i}}$, and the probability for at least one of them to choose this bit is $1 - (1 - [1/m'])^{n^v_{1,i}}$.

We say a common vehicle v is encoded by bitmap G_i if v chooses the same index separately in location L and L', and use $S^c_{G,i}$ to denote the set of common vehicles encoded by bitmap G_i. Since any common vehicle v has a probability of $([1/s] + (1 - [1/s])[1/m'])p''$ to choose the same bit in both locations, $|S^c_{G,i}| = n^c_{1,i} p''([1/s] + (1 - [1/s])[1/m'])$. If two different vehicles, except the common vehicles in $S^c_{G,i}$, choose the same index separately in location L and L' will introduce a virtual vehicle in S^v_i. Based on (13), we compute the number of independent vehicles that would have produced the bitmap E_i and B'_i

$$N_i = \frac{\ln V_{E_i,0}}{\ln\left(1 - \frac{1}{m'}\right)} \quad N'_i = \frac{\ln V_{B'_i,0}}{\ln\left(1 - \frac{1}{m'}\right)} \tag{13}$$

where $V_{E_i,0}$ is the fraction of zeros in E_i, and $V_{B'_i,0}$ is the fraction of zeros in B'_i. We use an abstract set of N_i (N'_i) vehicles to produce the same effect as what all the vehicles in S_i (S'_i) jointly produce in E_i (B'_i). The probability P_a (P_b) for at least one vehicle in $S_i - S^c_{G,i}$ $(S'_i - S^c_{G,i})$ to set the lth bit of E_i (B'_i) is

$$\begin{aligned} P_a &= 1 - \left(1 - \frac{1}{m'}\right)^{N_i - n^c_{1,i} p'' \left(\frac{1}{s} + \left(1 - \frac{1}{s}\right)\frac{1}{m'}\right)} \\ P_b &= 1 - \left(1 - \frac{1}{m'}\right)^{N'_i - n^c_{1,i} p'' \left(\frac{1}{s} + \left(1 - \frac{1}{s}\right)\frac{1}{m'}\right)}. \end{aligned} \tag{14}$$

The probability P_c for at least two different vehicles, which are not in $S^c_{G,i}$, to set the lth bit separately in location L and L' is

$$P_c = \left(1 - \left(1 - \frac{1}{m'}\right)^{N_i - |S^c_{G,i}|}\right)\left(1 - \left(1 - \frac{1}{m'}\right)^{N'_i - |S^c_{G,i}|}\right). \tag{15}$$

Notice that P_* is equal to the probability for at least one of the virtual vehicles to choose the lth bit to set. Thus, we have

$$\begin{aligned} 1 - \left(1 - \frac{1}{m'}\right)^{n^v_{1,i}} &= \left(1 - \left(1 - \frac{1}{m'}\right)^{N_i - n^c_{1,i} p'' \left(\frac{1}{s} + \left(1 - \frac{1}{s}\right)\frac{1}{m'}\right)}\right) \\ &\quad \times \left(1 - \left(1 - \frac{1}{m'}\right)^{N'_i - n^c_{1,i} p'' \left(\frac{1}{s} + \left(1 - \frac{1}{s}\right)\frac{1}{m'}\right)}\right). \end{aligned} \tag{16}$$

Now, we have two independent equations, i.e., (12) and (16), which can be used to solve for the values of unknowns $n^c_{1,i}$ and $n^v_{1,i}$.

When there are multiple periods, we can follow a similar way to derive an estimator for $n^c_{k,i_1 i_2 \ldots i_k}$, where $1 \leq k \leq t$. First, we join the information from bitmaps of k periods by performing bitwise AND among them, and the result bitmap is denoted as $W^c_{i_1 i_2 \ldots i_k}$. Its lth bit is denoted as $W^c_{i_1 i_2 \ldots i_k}[l]$, for $1 \leq l \leq m$, which is equal to $G_{i_1}[l] \wedge G_{i_2}[l] \wedge \cdots \wedge G_{i_k}[l]$. Based on (17), we compute the number of independent vehicles that would have produced the bitmap $W^c_{i_1 i_2 \ldots i_k}$

$$N^c_{i_1 i_2 \ldots i_k} = \frac{\ln V^w_{i_1 i_2 \ldots i_k,0}}{\ln\left(1 - \frac{1}{m'}\right)} \tag{17}$$

where $V^w_{i_1 i_2 \ldots i_k,0}$ is the fraction of zeros in $W^c_{i_1 i_2 \ldots i_k}$. Because the bits of ones in $W^c_{i_1 i_2 \ldots i_k}$ retain all the information from k-persistent common vehicles in $S^c_{G,i_1} \cap S^c_{G,i_2} \cap \cdots \cap S^c_{G,i_{k-1}}$ and virtual vehicles in $S^v_{i_1} \cap S^v_{i_2} \cap \cdots \cap S^v_{i_{k-1}}$, we can build an equation that relates $n^c_{k,i_1 i_2 \ldots i_k}$ to $n^v_{k,i_1 i_2 \ldots i_k}$

$$N^c_{i_1 i_2 \ldots i_k} = n^c_{k,i_1 i_2 \ldots i_k} p'' \left(\frac{1}{s} + \left(1 - \frac{1}{s}\right)\frac{1}{m'}\right) + n^v_{k,i_1 i_2 \ldots i_k} \tag{18}$$

where $n^{v}_{k,i_1 i_2 \ldots i_k} = |S^{v}_{i_1} \cap S^{v}_{i_2} \cap \cdots \cap S^{v}_{i_{k-1}}|$. There are two unknowns in (18). To derive an estimator for $n^{c}_{k,i_1 i_2 \ldots i_k}$, we need to build another independent equation that relates $n^{c}_{k,i_1 i_2 \ldots i_k}$ to $n^{v}_{k,i_1 i_2 \ldots i_k}$. Similar to the one period case, we will analyze the probability for at least one virtual vehicles in $S^{v}_{i_2} \cap \cdots \cap S^{v}_{i_{k-1}}$ to set the lth bit to build this equation.

Consider an arbitrary bit $W^{c}_{i_1 i_2 \ldots i_k}[l]$. The probability for a virtual vehicle $v \in S^{v}_{i_1} \cap S^{v}_{i_2} \cap \cdots \cap S^{v}_{i_k}$ to choose this bit to set is $(1/m')$. There are $n^{v}_{k,i_1 i_2 \ldots i_k}$ virtual vehicles in set $S^{v}_{i_1} \cap S^{v}_{i_2} \cap \cdots \cap S^{v}_{i_k}$. The probability for none of them to choose this bit is $(1-[1/m'])^{n^{v}_{k,i_1 i_2 \ldots i_k}}$, and the probability for at least one of the virtual vehicles choose this bit to set is $1-(1-[1/m'])^{n^{v}_{k,i_1 i_2 \ldots i_k}}$.

Note that $W^{c}_{i_1 i_2 \ldots i_k}$ is the result bitmap by performing bitwise AND among all the expanded bitmaps record in period $i_1, i_2, \ldots, i_k$. We can also first perform bitwise AND among bitmaps of each location, and then perform bitwise AND among the result bitmaps to get $W^{c}_{i_1 i_2 \ldots i_k}$. Let E_* (E'_*) be the result bitmap of performing bitwise AND among $E_{i_1}, E_{i_2}, \ldots, E_{i_k}$ $(B'_{i_1}, B'_{i_2}, \ldots, B'_{i_k})$. Then, $W^{c}_{i_1 i_2 \ldots i_k}$ should be the result bitmap of E_* and E'_* by performing bitwise AND. For each k-persistent common vehicle passes L and L' in period i_1, period i_2,…, period i_2, it has a probability of $((1/s) + (1-[1/s])[1/m'])p''$ to introduce a bit of one in $W^{c}_{i_1 i_2 \ldots i_k}$. Other ones can be abstracted as introduced by virtual vehicles. Based on (19), we compute the number of independent vehicles that would have produce the bitmap E_* and E'_*

$$N_* = \frac{\ln V_{E_*,0}}{\ln\left(1-\frac{1}{m'}\right)} \quad N'_* = \frac{\ln V_{E'_*,0}}{\ln\left(1-\frac{1}{m'}\right)} \tag{19}$$

where $V_{E_*,0}$ $(V_{E'_*,0})$ is the fraction of zeros in E_* (E'_*). Essentially we use an abstract set S_* (S'_*) of N_* (N'_*) vehicles to produce the same effect as what all encoded vehicles pass L (L') in period i_1, period i_2,…, period i_k jointly produce in E_* (E'_*). If two abstracted vehicles, except the encoded k-persistent common vehicles, choose the same index separately in E_* and E'_* will introduce a virtual vehicle in $S^{v}_{i_1} \cap S^{v}_{i_2} \cap \cdots \cap S^{v}_{i_k}$. We use $S^{c}_{w,i_1 i_2 \ldots i_k}$ to denote the set of k-persistent vehicles that choose the same index in E_* and E'_*. Then, we have $|S^{c}_{w,i_1 i_2 \ldots i_k}| = ([1/s] + (1-[1/s])[1/m'])n^{c}_{k,i_1 i_2 \ldots i_k}p''$. The probability $P_{a,*}$ $(P_{b,*})$ for at least one vehicle in $S_* - S^{c}_{w,i_1 i_2 \ldots i_k}$ $(S'_* - S^{c}_{w,i_1 i_2 \ldots i_k})$ set the lth bit of E_* (E'_*) is

$$\begin{aligned} P_{a,*} &= 1-\left(1-\frac{1}{m'}\right)^{N_* - n^{c}_{k,i_1 i_2 \ldots i_k} p''\left(\frac{1}{s}+\left(1-\frac{1}{s}\right)\frac{1}{m'}\right)} \\ P_{b,*} &= 1-\left(1-\frac{1}{m'}\right)^{N'_* - n^{c}_{k,i_1 i_2 \ldots i_k} p''\left(\frac{1}{s}+\left(1-\frac{1}{s}\right)\frac{1}{m'}\right)}. \end{aligned} \tag{20}$$

The probability $P_{c,*}$ for at least two different abstracted vehicles, which is not in $S^{c}_{w,i_1 i_2 \ldots i_k}$, to set the lth bit separately in location L and L' is equal to $P_{a,*} * P_{b,*}$.

Notice that $P_{c,*}$ is equal to the probability for at least one of the virtual vehicles in $S^{v}_{i_1} \cap S^{v}_{i_2} \cap \cdots \cap S^{v}_{i_k}$ to choose the lth bit to set. Thus, we have

$$\begin{aligned} 1-\left(1-\frac{1}{m'}\right)^{n^{v}_{k,i_1 i_2 \ldots i_k}} &= \left(1-\left(1-\frac{1}{m'}\right)^{N_* - |S^{c}_{w,i_1 i_2 \ldots i_k}|}\right) \\ &\quad \times \left(1-\left(1-\frac{1}{m'}\right)^{N'_* - |S^{c}_{w,i_1 i_2 \ldots i_k}|}\right). \end{aligned} \tag{21}$$

Solving (18) and (21), we can get the estimated value of $n^{v}_{k,i_1 i_2 \ldots i_k}$ and $n^{c}_{k,i_1 i_2 \ldots i_k}$.

Similar as the k-persistent point traffic measurement, we can join the information from bitmaps of different periods by performing bitwise OR among them, i.e., join G_{i_1}, G_{i_2},…, G_{i_k} to get the result bitmap $F^{c}_{i_1 i_2 \ldots i_k}$. Then, we can build another equation that relates $n^{v}_{k,i_1 i_2 \ldots i_k}$ to $n^{c}_{k,i_1 i_2 \ldots i_k}$ based on the principle of *inclusion and exclusion*

$$V^{c}_{i_1 i_2 \ldots i_k} = V^{c}_{i_1 i_2 \ldots i_{k-1}} \left(1-\frac{1}{m}\right)^{Ap''\left(\frac{1}{s}+\left(1-\frac{1}{s}\right)\frac{1}{m'}\right)+B} \tag{22}$$

where $V^{c}_{i_1 i_2 \ldots i_k}$is the fraction of zeros in $F^{c}_{i_1 i_2 \ldots i_k}$, $A = \sum_{q=0}^{k-1}(-1)^q \sum_{1 \le p_1 < \cdots < p_q < k} n^{c}_{i_{p_1} \ldots i_{p_q} i_k}$, $B = \sum_{q=0}^{k-1}(-1)^q \sum_{1 \le p_1 < \cdots < p_q < k} n^{v}_{i_{p_1} \ldots i_{p_q} i_k}$.

Solving (21) and (22), we can also estimate the value of $n^{v}_{k,i_1 i_2 \ldots i_k}$ and $n^{c}_{k,i_1 i_2 \ldots i_k}$. To decrease the relative error, our estimator first estimate $n^{c}_{k,i_1 i_2 \ldots i_k}$ by solving (18) and (21), the result is denoted as $\hat{n}^{c,1}_{k,i_1 i_2 \ldots i_k}$. Then, we estimate it again by solving (21) and (22), the result is denoted as $\hat{n}^{c,1}_{k,i_1 i_2 \ldots i_k}$. Finally, we take the average as the final estimation result $\hat{n}^{c}_{k,i_1 i_2 \ldots i_k}$

$$\hat{n}^{c}_{k,i_1 i_2 \ldots i_k} = \frac{\hat{n}^{c,1}_{k,i_1 i_2 \ldots i_k} + \hat{n}^{c,2}_{k,i_1 i_2 \ldots i_k}}{2}. \tag{23}$$

D. Estimator for k-Persistent Common Traffic

We can obtain an estimator for $\hat{n}^{*,c}_{k,12\ldots t}$ by using a similar way as our point model. A common vehicle that passes location L and L' exactly during k measurement periods must be recorded by all the bitmaps of one subset of k bitmaps from $\mathcal{G} = \{G_1, G_2, \ldots, G_t\}$. For each subset of k bitmaps from $\mathcal{G}$, we can measure the k-persistent traffic volume by using the estimator proposed in Section V-C. A common vehicle that passes location L and L' exactly during $i(k+1 \le i \le t)$ measurement periods will be double counted C^{k}_{i} times if we use $\sum_{1 \le i_1 < i_2 < \cdots < i_k \le t} n^{c}_{k,i_1 i_2 \ldots i_k}$ as the number of common vehicles that pass location L and L' exactly during k measurement periods. Thus, we have

$$\hat{n}^{c}_{k,12\ldots t} = \sum_{1 \le i_1 < i_2 < \cdots < i_k \le t} \hat{n}^{c}_{k,i_1 i_2 \ldots i_k} - \sum_{i=k+1}^{t} C^{k}_{i} \hat{n}^{c}_{i,12\ldots t} \tag{24}$$

where $n^{c}_{i,12\ldots t}$ $(\hat{n}^{c}_{i,12\ldots t})$ is the number (estimated number) of common vehicles that pass location L and L' exactly during i measurement periods, and $\hat{n}^{c}_{k,i_1 i_2 \ldots i_k}$ is the estimated number of k-persistent traffic volume for a given k measurement periods $i_1, i_2, \ldots, i_k$. Since the k-persistent common traffic include all the common vehicles that pass location L and L' during at least k measurement periods, we have our estimator for $\hat{n}^{*,c}_{k,12\ldots t}$

$$\hat{n}^{*,c}_{k,12\ldots t} = \sum_{1 \le i_1 < i_2 < \cdots < i_k \le t} \hat{n}^{c}_{k,i_1 i_2 \ldots i_k} - \sum_{i=k+1}^{t} \left(C^{k}_{i} - 1\right) \hat{n}^{c}_{i,12\ldots t}. \tag{25}$$

Based on (24), we can estimate the value of $\hat{n}^c_{t,12\ldots t}, \ldots, \hat{n}^c_{k,12\ldots t}$ one by one. Then, we obtain the value of $\hat{n}^{*,c}_{k,12\ldots t}$ according to (25).

VI. Privacy Analysis

When a vehicle passes an RSU, the only thing that a vehicle may do is to set a bit in this RSU's bitmap to one at an index that may vary from location to location. Moreover, different vehicles may choose the same indices. What each RSU gathers is a bitmap, with each bit of one suggesting the passage of at least one vehicle. Therefore, the tracker may possibly identify a vehicle through the observation that a bit at one location is one, or identify the trajectory of a common vehicle through the observation that the bits with the same index at two different locations are both ones. Below, we will analyze the privacy preservation of our persistent-traffic measurement design in terms of ϵ-DP and the probabilistic noise-to-information ratio as defined in Section II-C.

Theorem 2: The random encoding mechanism of our estimator satisfies ϵ-DP with the following value of p'':

$$p'' \leq \frac{(e^{\epsilon}-1)\left(1-e^{-\frac{np''}{m}}\right)}{e^{-\frac{np''}{m}}}. \tag{26}$$

Proof: Consider two adjacent sets of vehicles S and S', and there is only one vehicle difference between S and S'. Without loss of generality, we assume that the only different vehicle $v \in S$. Suppose v will be mapped to the lth bit if it is sampled and encoded in the bitmap. Then the bitmaps encode the vehicles separately in S and S' can only differ in the lth bit.

Let B (B') be the bitmap encoding the vehicles in set S (S'), $\Pr[B[l]=1]$ ($\Pr[B'[l]=1]$) be the probability for $B[l]=1$ ($B'[l]=1$). Since v is in set S, it has a probability p'' of setting $B[l]=1$. If v is not encoded by B, the probability for the lth bit of B is set to one by other vehicles in S is $1-(1-[1/m])^{np''}$, where m is the bitmap size and $n=|S|-1$. Then, we have $\Pr[B[l]=1] = p'' + (1-p'')(1-(1-[1/m])^{np''})$. The probability for vehicles in S' to set the lth of B' is $1-(1-[1/m])^{np''}$, then

$$\frac{\Pr[B[l]=1]}{\Pr[B'[l]=1]} = \frac{p''+\left(1-p''\right)\left(1-\left(1-\frac{1}{m}\right)^{np''}\right)}{1-\left(1-\frac{1}{m}\right)^{np''}}. \tag{27}$$

By applying the approximation $(1-[1/m])^{np''} \approx e^{-[np''/m]}$ that works when m is large, we have

$$\frac{\Pr[B[l]=1]}{\Pr[B'[l]=1]} \approx \frac{p''+\left(1-p''\right)\left(1-e^{-\frac{np''}{m}}\right)}{1-e^{-\frac{np''}{m}}} \leq e^{\epsilon}. \tag{28}$$

Similarly, we have

$$\frac{\Pr[B[l]=0]}{\Pr[B'[l]=0]} = \frac{\left(1-p''\right)\left(1-\frac{1}{m}\right)^{np''}}{\left(1-\frac{1}{m}\right)^{np''}} = 1-p'' \leq e^{\epsilon}. \tag{29}$$

In summary, $[(\Pr[\mathcal{M}(S)=X])/(\Pr[\mathcal{M}(S')=X])] \leq e^{\epsilon}$ always stands for any output X, which finish our proof. ■

Next, we give the analysis of probability noise-to-information ratio. Each vehicle v that pass location L has a probability p'' to send an index i to the RSU and set $B[i]=1$, where the index $i = H(K_v \oplus C[H(L) \oplus K_v \mod s]) \mod m$. However, the authority may associate the index i with the vehicle v at L by accidental events, such as v is stopped by a police for speeding when there is no other vehicle around, and the police informs the authority. In this case, the authority may derive the partial trajectory of vehicle v by observing the bit at the same index in the bitmap of other location.

Let p be the probability that $B'[i]$ is set to one by other vehicles even if v does not pass L', and n' be the number of vehicles passing L'. Note that only $n'p''$ vehicles can be recorded by the RSU. Thus, we have

$$p = 1-\left(1-\frac{1}{m'}\right)^{n'p''}. \tag{30}$$

Let p' be the probability that $B'[i]$ is set to one when v does pass L'. As the analysis in [18], we have

$$p' = p + (1-p)\frac{1}{s}. \tag{31}$$

Further, the probabilistic noise-to-information ratio is

$$\frac{p}{p'-p} = \frac{1-\left(1-\frac{1}{m'}\right)^{n'p''}}{\left(1-\frac{1}{m'}\right)^{n'p''}\frac{1}{s}} \approx s\left(e^{\frac{np''}{m}}-1\right). \tag{32}$$

VII. Experiment

We conduct extensive experiments to evaluate the performance of our estimators in this section. Our experiments are based on the real traffic flows in Shenzhen, China for five consecutive working days from May 15th to 19th, 2017. This data set includes the passing vehicle IDs of 475 locations, which are recorded by the surveillance cameras. Only 210 of them have exact geographical position information, which is shown in Fig. 2. We measure the k-persistent point and common traffic volume in the rush hours from 7 A.M. to 10 A.M., i.e., setting the time interval [7 A.M., 10 A.M.] as a measurement period. For any given location (or two locations) of interest, we measure the actual k-persistent traffic volume by removing the duplicate vehicles (common vehicles) of each period of interest, hashing the vehicles of each location in the duplicate vehicle sets gathered from those five periods to a hash table and counting the number of measurement periods that each vehicle passes. In our experiments, we compare the actual traffic volume and the estimated one to show the efficiency of our estimators.

In each measurement period, we use a bitmap to record the traffic for each location. Note that the bitmap size $m = 2^{\lceil \log_2(\bar{n}\times f)\rceil}$, which means $(m/2\bar{n}p'') < f \leq (m/\bar{n}p'')$. In our experiments, we choose the average traffic volume of each location as the measured value of the expected traffic volume $\bar{n}$. Combining (28), we have

$$\frac{p''+\left(1-p''\right)\left(1-e^{-\frac{np''}{m}}\right)}{1-e^{-\frac{np''}{m}}} \leq \frac{p''+\left(1-p''\right)\left(1-e^{-\frac{1}{2f}}\right)}{1-e^{-\frac{1}{2f}}}. \tag{33}$$

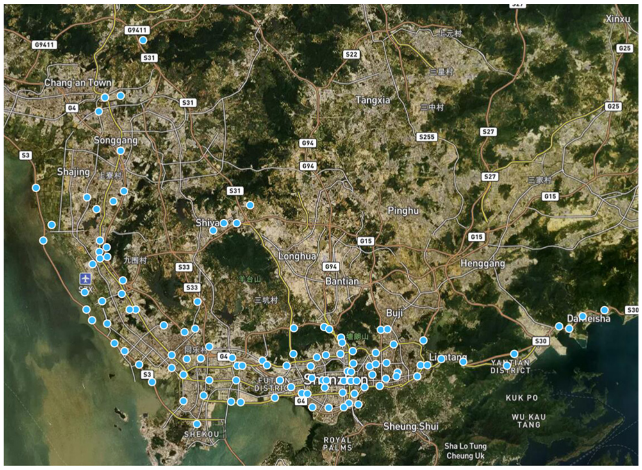


Fig. 2. RSU map in Shenzhen, China. Each blue dot represents an RSU location.

The inequality $([p'' + (1-p'')(1-e^{-[1/2f]})]/[1-e^{-[1/2f]}])$ $s \le e^{\epsilon}$ will always stand when p'' satisfies

$$p'' \le \frac{(e^{\epsilon}-1)\left(1-e^{-\frac{1}{2f}}\right)}{e^{-\frac{1}{2f}}}. \tag{34}$$

Therefore, to preserve the point privacy in each location, we set $p'' = ([(e^{\epsilon}-1)(1-e^{-[1/2f]})]/[e^{-[1/2f]}])$ in our experiments.

A. Experimental Results of k-Persistent Point Traffic Estimation

The experimental results of our estimator for the single-point model are presented in Fig. 3, where the horizontal axis represents the actual k-persistent traffic volume and the vertical axis represents the estimated k-persistent traffic volume. Note that the total number of measurement period t is set to 5. The plots from left to right in Fig. 3 show the results of 5, 4, 3, and 2-persistent traffic measurement of all 475 locations, respectively. We draw the equality line $y = x$ for reference. Each point represents k-persistent traffic measurement in a location. The closer the data points are to this line, the more the measurement accuracy is. Clearly, in Fig. 3, the points are clustered to the equality line in all plots, which indicates that the proposed single-point estimator has good measurement accuracy for different k-persistent point traffic estimator. Next, We investigate how different parameters affect the measurement accuracy of our k-persistent point traffic estimator. We employ a metric called the average absolute error to evaluate the measurement accuracy of our estimator. It is defined as $(1/N)\sum_{i=0}^{N-1}|\hat{n}-n|$, where N is the total number of flows, n is the true value, and $\hat{n}$ is its estimated value.

First, we evaluate the impact of the privacy budgets ϵ. The system-wide load factor f is fixed to 3. We repeat the experiments with p'' = 0.0635, 0.1491, and 0.3116, which translates to 0.3, 0.6, and 1-DP privacy, respectively. The estimation results are shown in Fig. 4. From left to right, the plots show the results of 5, 4, 3, and 2-persistent point traffic measurement, respectively. We observe that our estimator becomes more accurate as ϵ increase in all four plots. This is because a larger ϵ indicates more vehicles will be encoded (i.e., a smaller sampling probability), leading to a higher estimation accuracy due to a lower sampling error. From the results of the experiment, we also observe that there is a tradeoff between accuracy and privacy. As ϵ decreases, the DP becomes better but measurement accuracy decreases.

Second, we evaluate the impact of the system-wide load factor f on estimation accuracy. The sampling probability p'' is set to 1 such that all vehicles will be recorded. We repeat the experiments for k-persistent point traffic measurement with $f = 2, 3,$ and 5, respectively. The results are shown in the three plots in Fig. 5. Clearly, as f grows, the accuracy increases since each location has a larger bitmap to record traffic. Note that our estimator can still achieve 1.5087, 1.8739, 2.3522-DP privacy when $p'' = 1$, separately for $f = 2, 3,$ and 5.

B. Experiment Results of k-Persistent Common Traffic Estimation

General point-to-point traffic measurement mainly focuses on two relative close locations (or called relative locations), which produces more useful information for traffic engineering. Therefore, to evaluate our estimator for k-persistent common traffic measurement, we only consider the point-to-point traffic from two relative locations. In our experiments, we assume two locations are relative locations if their 1-persistent common traffic volume is no less than 2000, and 5-persistent common traffic volume is larger than zero. In our data set, there are 1585 relative location pairs that satisfy such requirements. The experiment results of our point-to-point estimator are presented in Fig. 6, where the plots from left to right show the results of 5, 4, 3, and 2-persistent common traffic measurement of all 1585 relative pair locations, respectively. Clearly, in each plot, the points persistently follow the equality line. Hence, the proposed our point-to-point estimator has good measurement accuracy for k-persistent common traffic estimator.

Then, we evaluate he impact of privacy budge ϵ on the measurement accuracy. The number s of indices that a vehicle can map is set to 3 and the load factor f is fixed to 3. We vary the value of ϵ from 0.3 to 0.6 to 1. The corresponding estimation results are shown in Fig. 7. Similar to the experiments of point traffic measurement, there is a tradeoff between estimation accuracy and point privacy in our k-persistent common traffic measurement.

Next, we evaluate the impact of parameters s and f setting on measurement accuracy when $p'' = 1$ and $t = 5$. From the simulation results as shown in Figs. 8 and 9, we found that the absolute error of our estimator for point-to-point model increases as s and decreases as f. This is mainly because a larger s or a smaller f will introduce more noise to traffic records, which will return a worse estimation accuracy.

We also consider the trajectory privacy of vehicles when their point privacy have been leaked. In Table II, we evaluate the trajectory privacy protection by measuring the probabilistic noise-to-information ratio with respect to f and s. Note that a larger probabilistic noise-to-information ratio indicates better trajectory privacy protection since it increases uncertainty to

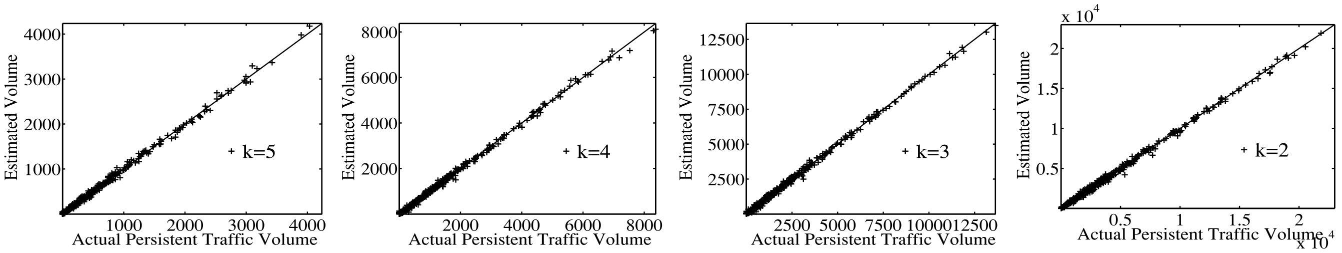


Fig. 3. Estimation accuracy of k-persistent point traffic volume when $t = 5, f = 3, p'' = 0.1491,$ and $\epsilon = 0.6$.

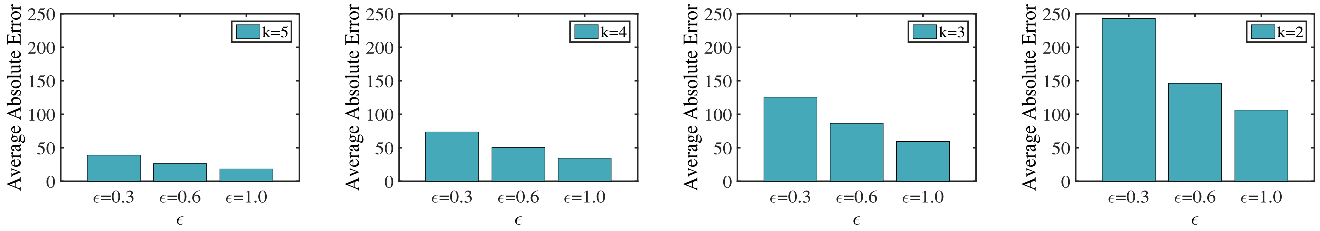


Fig. 4. Average absolute error of k-persistent point traffic volume when $t = 5,\ f = 3, p'' = \{0.0635, 0.1491, 0.3116\},$ and $\epsilon = \{0.3, 0.6, 1\}$.

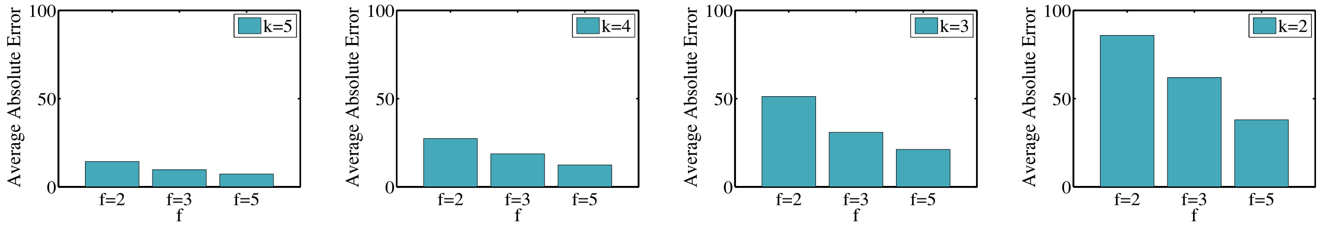


Fig. 5. Average absolute error of k-persistent point traffic volume when $p'' = 1, t = 5,\ f = \{2, 3, 5\},$ and $\epsilon = \{1.5087, 1.8739, 2.3522\}$.

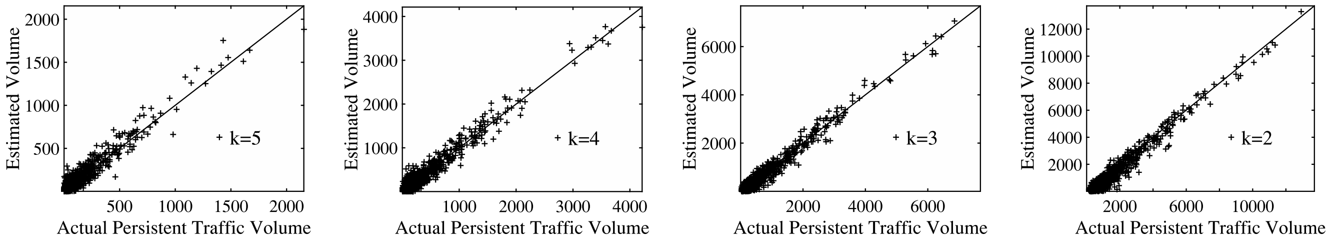


Fig. 6. Estimation accuracy of k-persistent common traffic volume when $t = 5,\ f = 3, s = 3, p'' = 0.1491,$ and $\epsilon = 0.6$.

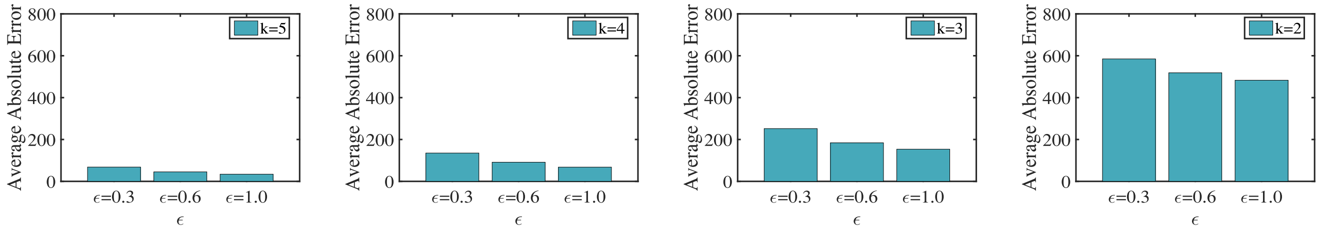


Fig. 7. Mean absolute error of k-persistent common traffic volume when $t = 5,\ f = 3, s = 3, p'' = \{0.0635, 0.1491, 0.3116\},$ and $\epsilon = \{0.3, 0.6, 1\}$.

track the trajectory of individual vehicles. In Table III, we evaluate the estimation accuracy with respect to f and s. Clearly, when decreasing f or increasing s, the noise-to-information ratio grows, while it leads to a worse estimation accuracy as we mentioned before. To balance the accuracy and trajectory privacy, we set $f = 3$ and $s = 3$ in Fig. 7. We point out that, under these parameters, our accuracy evaluation has consistently produced good results, and the probabilistic noise-to-information ratio is relatively large as shown in the table.

C. Computational Overhead

An RSU uses the same method to collect information from the passing vehicles in the point estimator or the point-to-point estimator. The time complexity of a passing vehicle is

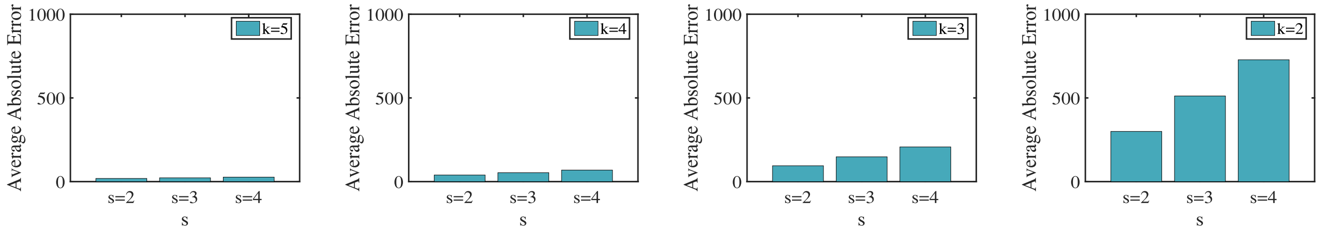


Fig. 8. Mean absolute error of k-persistent common traffic volume when $t = 5,\ f = 3, p'' = 1, \epsilon = 1.8739,$ and $s = \{2, 3, 4\}$.

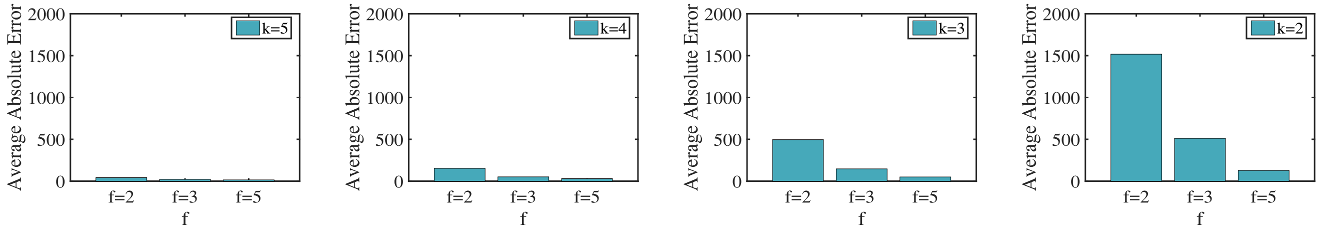


Fig. 9. Mean absolute error of k-persistent common traffic volume when $t = 5, s = 3, p'' = 1, f = \{2, 3, 5\},$ and $\epsilon = \{1.5087, 1.8739, 2.3522\}$.

TABLE II
PROBABILISTIC NOISE-TO-INFORMATION RATIO WITH RESPECT TO f AND s

f / s	$f=1$	$f=1.5$	$f=2$	$f=2.5$	$f=3$	$f=3.5$	$f=4$
$s=2$	3.44	1.90	1.30	0.98	0.79	0.66	0.57
$s=3$	5.16	2.84	1.95	1.48	1.19	0.99	0.85
$s=4$	6.88	3.79	2.59	1.97	1.58	1.32	1.14
$s=5$	8.59	4.74	3.24	2.46	1.98	1.65	1.42

TABLE III
AVERAGE ABSOLUTE ERROR OF COMMON TRAFFIC ESTIMATION WITH RESPECT TO f AND s, $\epsilon = 0.6$

f / s	$f=1$	$f=1.5$	$f=2$	$f=2.5$	$f=3$	$f=3.5$	$f=4$
$s=2$	225	67	43	36	33	34	34
$s=3$	345	100	57	46	47	45	44
$s=4$	483	137	73	61	58	54	57
$s=5$	624	180	93	74	73	68	69

TABLE IV
RUNNING TIME OF OUR ESTIMATORS

t / estimator	$t=4$	$t=5$	$t=6$	$t=7$	$t=15$
point	13ms	28ms	62ms	138ms	65s
point-to-point	45ms	95ms	211ms	461ms	120s

$O(1)$, including authentication with the RSU and several hash computations; the complexity of the RSU is also $O(1)$, including authentication with the vehicle and setting a bit to one in its bitmap. The computation of the offline k-persistent traffic estimation performed at the traffic measurement server is heavier. Under the point traffic model, we need to compute $\hat{n}_{j,i_1,i_2,\ldots,i_j}$ in $O(tm')$, for all subsets of the t bitmaps, where m is the bitmap size and t is the number of periods under consideration, which is a preset, small constant (such as 5 for a working week in our simulations). The total time complexity is therefore $O(2^t tm')$, for calculating the value of k-persistent point traffic volume based on Algorithm 2. Similarly, we need to compute $\hat{n}^c_{j,i_1,i_2,\ldots,i_j}$, for all combinations of j measurement periods $i_1, i_2, \ldots, i_j$, $k \le j \le t$, under the point-to-point traffic model. Therefore, the overall time complexity for the central server to calculate the k-persistent common traffic volume is also $O(2^t tm')$.

We perform experiments to evaluate the time efficiency of the proposed estimators. Because the online components take very little time (much less than 1 s), our focus is on the offline traffic estimation, which is performed on a desktop with Intel Core i7-4720HQ CPU at 2.6 GHz and 8-GB DDR memory. The experimental results are presented in Table IV. When $t = 5$ for persistent traffic in a working week with each period being a day, the computation times are just 28 ms for point traffic and 95 ms for point-to-point traffic. When $t = 15$ for half a month, the computation times are 65 and 120 s for point traffic and point-to-point traffic, respectively. We stress that this is offline traffic estimation, which is not subject to the same time constraint as online operations.

VIII. RELATED WORK

A. *Transportation Traffic Measurement*

With the development of the Internet of things and cloud computing [23]–[26], new technologies become available for data collection, such as VCPS that enables automatic monitoring of urban transportation traffic. In this paper, we study how to estimate the k-persistent traffic volume with information collected from VCPS.

There are two categories in the field of transportation traffic measurement: *single-period* traffic measurement and *multiperiod* traffic measurement. Single-period traffic measurement refers to measure the volume of point traffic or common (point-to-point) traffic during a particular measurement period. Note that, we use point traffic to denote the vehicles that traversing a particular geographical location (i.e., road intersection). Hence, a single-period point traffic measurement often returns in the form of annual average daily traffic (AADT). Various prediction methods [11]–[13] have been proposed to measure single-point traffic volume based on the data recorded by automatic traffic recorders (ATRs) installed at road intersections. For example, [11] employs a support vector regression model to evaluate the volume, [12] introduces an absolute deviation penalty procedure, and [13] uses the regression and Bayesian model to solve this problem. Different from point traffic, point-to-point traffic refers to the common vehicles that travel through 2 particular independent points. In fact, the passing points of a specific vehicle are part of its trajectory. Therefore, we should consider the privacy protection issues in the point-to-point traffic measurement. Various methods are proposed to estimate the single-period point-to-point traffic. Zhou *et al.* [14]–[17] studied the point-to-point traffic measurement issue by employing the data structure of bitmap in CPRS. In [14], they proposed a model for point-to-point traffic measurement, which preserves vehicles privacy by an encryption method and measures point-to-point traffic using the encrypted vehicle IDs. This model improves the computation efficiency to $O(n_x n_y)$ for any two RSUs, where n_x and n_y denote the number of vehicles traversing them, respectively. However, the computation efficiency of this model is still unacceptable for todayąŕs large-scale road networks. To solve the problem of high computation overhead, Zhou *et al.* [15], [17] introduced the fixed-length bit arrays and improves the computation efficiency to $O(n_x + n_y)$. However, different RSUs may observe different traffic in the real world. Thus, the performance of the method proposed in [15] and [17] dramatically decreases regarding both vehicle privacy and measurement accuracy when considering a realistic situation. To adapt to the real situation better, Zhou *et al.* [16] further designed a variable-length bit array masking method.

Different from single-period traffic, multiperiod traffic refers to the persistent vehicles during two or more measurement periods. Huang *et al.* [18] and Sun *et al.* [27] proposed two estimators, respectively for the point and point-to-point persistent traffic measurement. However, this research is based on an overly strict assumption that the persistent traffic passes the interested point or multipoints at all measurement periods, and cannot provide the DP for point location privacy.

B. Network Traffic Measurement

Network traffic measurement is another branch of the traffic measurement, which is similar to transportation traffic measurement. Various methods [28]–[32] have been proposed for network traffic measurement, which is to measure the network traffic in a network router. However, none of these network traffic measurement methods can be directly employed in the scenario of transportation traffic measurement since the passing points of a vehicle in fact indicate its trajectory, while the passing points of packets are not likely to reveal their privacy (i.e., packet contents, packet source, and packet destination).

There have been many prior privacy-preserving studies for counting/histogram problem, such as RAPPOR [19] from Google that can provide strong DP without any trusted entity. However, none of them can be directly used in our context. To protect the privacy of items in a set, most existing methods assign one bit (0/1) to each item. The whole bit array stores the membership of the set, with the identifiers of all items preknown. However, we cannot assign each possible passing vehicle a bit since we do not know the set of possible vehicles beforehand. If we let vehicles report their IDs, their point privacy will be breached once an RSU knows the IDs of passing vehicles. Therefore, these prior methods that are tailored for other applications cannot solve the problem we study here.

IX. Conclusion

This paper studies privacy-preserving k-persistent traffic measurement in the context of intelligent cyber-physical road systems (CPRS). As far as we know, this paper is the first to study the k-persistent traffic measurement problem in transportation. We propose two novel estimators for k-persistent point and point-to-point traffic measurements, respectively. We show that the proposed estimators can provide stronger privacy preservation than previous studies—the new approaches protect not only the point DP of vehicles, but also the trajectory privacy of vehicles even when their point privacy is compromised due to external means outside of our approaches control. The efficiency of our estimators is demonstrated by extensive experiments based on real transportation traffic data set.

As future work, we will extend our study beyond point-to-point traffic measurement for estimating k-persistent traffic volume along a path through more than two specified locations, and we plan to discuss and work with the local transportation authority to explore real-world experimentation of the proposed estimators.

References

[1] N. Lu, N. Cheng, N. Zhang, X. Shen, and J. W. Mark, "Connected vehicles: Solutions and challenges," *IEEE Internet Things J.*, vol. 1, no. 4, pp. 289–299, Aug. 2014.

[2] M. Pan, P. Li, and Y. Fang, "Cooperative communication aware link scheduling for cognitive vehicular ad-hoc networks," *IEEE J. Sel. Areas Commun.*, vol. 30, no. 4, pp. 760–768, May 2012.

[3] J. Sun, C. Zhang, Y. Zhang, and Y. Fang, "An identity-based security system for user privacy in vehicular ad hoc networks," *IEEE Trans. Parallel Distrib. Syst.*, vol. 21, no. 9, pp. 1227–1239, Sep. 2010.

[4] Y. Zhu, Y. Wu, and B. Li, "Vehicular ad hoc networks and trajectory-based routing," in *Internet of Things*. Cham, Switzerland: Springer, 2014, pp. 143–167. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/7111363/references#references

[5] X. Zhu, S. Jiang, L. Wang, and H. Li, "Efficient privacy-preserving authentication for vehicular ad hoc networks," *IEEE Trans. Veh. Technol.*, vol. 63, no. 2, pp. 907–919, Feb. 2014.

[6] R. Du, C. Chen, B. Yang, N. Lu, X. Guan, and X. Shen, "Effective urban traffic monitoring by vehicular sensor networks," *IEEE Trans. Veh. Technol.*, vol. 64, no. 1, pp. 273–286, Jan. 2015.

[7] J. Eriksson, H. Balakrishnan, and S. Madden, "CaberNet: Vehicular content delivery using WiFi," in *Proc. 14th ACM Int. Conf. Mobile Comput. Netw. (MOBICOM)*, 2008, pp. 199–210.

[8] U. Lee, J. Lee, J.-S. Park, and M. Gerla, "FleaNet: A virtual market place on vehicular networks," *IEEE Trans. Veh. Technol.*, vol. 59, no. 1, pp. 344–355, Jan. 2010.

[9] Y. L. Morgan, "Notes on DSRC & WAVE standards suite: Its architecture, design, and characteristics," *IEEE Commun. Surveys Tuts.*, vol. 12, no. 4, pp. 504–518, 4th Quart., 2010.

[10] (2018). *Toyota and Lexus to Launch Technology to Connect Vehicles and Infrastructure in the U.S. in 2021*. [Online]. Available: https://corporatenews.pressroom.toyota.com/releases/toyota+and+lexus+to+launch+technology+connect+vehicles+infrastructure+in+u+s+2021.htm

[11] M. Castro-Neto, Y. Jeong, M. K. Jeong, and L. D. Han, "AADT prediction using support vector regression with data-dependent parameters," *Expert Syst. Appl.*, vol. 36, no. 2, pp. 2979–2986, 2009.

[12] B. Yang, S.-G. Wang, and Y. Bao, "Efficient local AADT estimation via SCAD variable selection based on regression models," in *Proc. Chin. Control Decis. Conf. (CCDC)*, 2011, pp. 1898–1902.

[13] I. Tsapakis, W. H. Schneider, IV, and A. P. Nichols, "A Bayesian analysis of the effect of estimating annual average daily traffic for heavy-duty trucks using training and validation data-sets," *Transp. Plan. Technol.*, vol. 36, no. 2, pp. 201–217, 2013.

[14] Y. Zhou, S. Chen, Z. Mo, and Y. Yin, "Privacy preserving origin-destination flow measurement in vehicular cyber-physical systems," in *Proc. IEEE 1st Int. Conf. Cyber Phys. Syst. Netw. Appl. (CPSNA)*, 2013, pp. 32–37.

[15] Y. Zhou, Q. Xiao, Z. Mo, S. Chen, and Y. Yin, "Privacy-preserving point-to-point transportation traffic measurement through bit array masking in intelligent cyber-physical road systems," in *Proc. IEEE CPSCom*, 2013, pp. 826–833.

[16] Y. Zhou, S. Chen, Z. Mo, and Q. Xiao, "Point-to-point traffic volume measurement through variable-length bit array masking in vehicular cyber-physical systems," in *Proc. IEEE 35th Int. Conf. Distrib. Comput. Syst. (ICDCS)*, 2015, pp. 51–60.

[17] Y. Zhou, Z. Mo, Q. Xiao, S. Chen, and Y. Ying, "Privacy-preserving transportation traffic measurement in intelligent cyber-physical road systems," *IEEE Trans. Veh. Technol.*, vol. 65, no. 5, pp. 3749–3759, May 2016.

[18] H. Huang, Y.-E. Sun, S. Chen, H. Xu, and Y. Zhou, "Persistent traffic measurement through vehicle-to-infrastructure communications," in *Proc. IEEE 37th Int. Conf. Distrib. Comput. Syst. (ICDCS)*, 2017, pp. 394–403.

[19] Ú. Erlingsson, V. Pihur, and A. Korolova, "RAPPOR: Randomized aggregatable privacy-preserving ordinal response," in *Proc. ACM SIGSAC Conf. Comput. Commun. Security*, 2014, pp. 1054–1067.

[20] SpoofMAC. (2015). *Spoof Your MAC Address*. [Online]. Available: https://github.com/feross/SpoofMAC

[21] R. Stanojevic, M. Nabeel, and T. Yu, "Distributed cardinality estimation of set operations with differential privacy," in *Proc. IEEE Symp. Privacy Aware Comput. (PAC)*, 2017, pp. 37–48.

[22] C. Dwork and A. Roth, "The algorithmic foundations of differential privacy," *Found. Trends Theor. Comput. Sci.*, vol. 9, nos. 3–4, pp. 211–407, 2014.

[23] C. Stergiou, K. Psannis, A. Plageras, Y. Ishibashi, and B.-G. Kim, "Algorithms for efficient digital media transmission over IoT and cloud networking," *J. Multimedia Inf. Syst.*, vol. 5, no. 1, pp. 27–34, 2018.

[24] V. A. Memos, K. E. Psannis, Y. Ishibashi, B.-G. Kim, and B. B. Gupta, "An efficient algorithm for media-based surveillance system (EAMSuS) in IoT smart city framework," *Future Gener. Comput. Syst.*, vol. 83, pp. 619–628, Jun. 2018.

[25] A. P. Plageras, K. E. Psannis, C. Stergiou, H. Wang, and B. B. Gupta, "Efficient IoT-based sensor big data collection—Processing and analysis in smart buildings," *Future Gener. Comput. Syst.*, vol. 82, pp. 349–357, May 2018.

[26] K. E. Psannis, C. Stergiou, and B. B. Gupta, "Advanced media-based smart big data on intelligent cloud systems," *IEEE Trans. Sustain. Comput.*, vol. 4, no. 1, pp. 77–87, Jan./Mar. 2019.

[27] Y.-E. Sun, H. Huang, S. Chen, H. Xu, K. Han, and Y. Zhou, "Persistent traffic measurement through vehicle-to-infrastructure communications in cyber-physical road systems," *IEEE Trans. Mobile Comput.*, to be published.

[28] Y. Zhang, M. Roughan, C. Lund, and D. Donoho, "An information-theoretic approach to traffic matrix estimation," in *Proc. SIGCOMM*, 2003, pp. 301–312.

[29] Y. Zhang, M. Roughan, N. Duffield, and A. Greenberg, "Fast accurate computation of large-scale ip traffic matrices from link loads," in *Proc. SIGMETRICS*, vol. 31, 2003, pp. 206–217.

[30] J. Cao, A. Chen, and T. Bu, "A quasi-likelihood approach for accurate traffic matrix estimation in a high speed network," in *Proc. INFOCOM*, 2008, pp. 21–25.

[31] T. Li, S. Chen, and Y. Qiao, "Origin-destination flow measurement in high-speed networks," in *Proc. INFOCOM*, 2012, pp. 2526–2530.

[32] H. Huang *et al.*, "You can drop but you can't hide: K-persistent spread estimation in high-speed networks," in *Proc. IEEE INFOCOM*, 2018, pp. 1889–1897.

Yu-E Sun received the Ph.D. degree in computer science and technology from the Shenyang Institute of Computing Technology, Chinese Academy of Sciences, Shenyang, China, in 2011.

She is currently an Associate Professor with the School of Rail Transportation, Soochow University, Suzhou, China. Her current research interests include span network traffic measurement, spectrum auction, privacy preserving, and wireless networks.

He Huang (M'16) received the Ph.D. degree from the Department of Computer Science and Technology, University of Science and Technology of China, Hefei, China, in 2011.

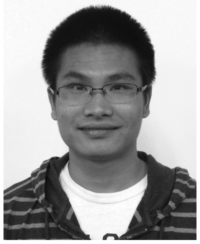
He is a Professor with the School of Computer Science and Technology, Soochow University, Suzhou, China. He is also with the Anhui Provincial Key Laboratory of Network and Information Security, Wuhu, China. His current research interests include network traffic measurement, spectrum auctions, privacy preserving, crowdsourcing, software defined networks, and satellite networks.

Dr. Huang is a member of the ACM.

Shigang Chen (A'03–M'04–SM'12–F'16) received the B.S. degree in computer science from the University of Science and Technology of China, Hefei, China, in 1993, and the M.S. and Ph.D. degrees in computer science from the University of Illinois at Urbana–Champaign, Champaign, IL, USA, in 1996 and 1999, respectively.

He was with Cisco Systems, San Jose, CA, USA, where he was involved with network security for three years and helped start up the network security company, Protego Networks, Milpitas, CA, USA. He joined the University of Florida, Gainesville, FL, USA, as an Assistant Professor in 2002, and became an Associate Professor in 2008, and where he he has been a Professor with the Department of Computer and Information Science and Engineering since 2013. He has authored or coauthored 190 peer-reviewed journal/conference papers and holds 12 U.S. patents.

Prof. Chen was a recipient of the IEEE Communications Society Best Tutorial Paper Award in 1999, the NSF CAREER Award in 2007, and the Cisco University Research Award in 2007 and 2012. He holds the University of Florida Research Foundation Professorship in 2017–2020 and the University of Florida Term Professorship in 2017–2020. He is an ACM Distinguished Member and an IEEE ComSoc Distinguished Lecturer.



You Zhou received the B.S. degree in electronic information engineering from the University of Science and Technology of China, Hefei, China, in 2013. He is currently pursuing the Ph.D. degree in computer and information science and engineering at the University of Florida, Gainesville, FL, USA, under the guidance of Prof. S. Chen.

He is currently with Google, Inc., Mountain View, CA, USA. His current research interests include network security and privacy, big network data, and Internet of Things.

Kai Han (M'05) received the B.S. and Ph.D. degrees in computer science from the University of Science and Technology of China, Hefei, China, in 1997 and 2004, respectively.

He is currently a Professor with the School of Computer Science and Technology, University of Science and Technology of China. His current research interests include mobile and social computing, graph data analysis, and data privacy.

Wenjian Yang received the B.S. degree from the Department of Mathematics and Computer Science, Anhui Normal University of China, Wuhu, China, in 2016. He is currently pursuing the M.S. degree at the School of Computer Science and Technology, Soochow University, Suzhou, China.

His current research interest includes traffic measurement, including transportation traffic measurement and network traffic measurement.