Verifiable and Privacy-Preserving Traffic Flow Statistics for Advanced Traffic Management Systems

Chuan Zhang, *Student Member, IEEE*, Liehuang Zhu, *Member, IEEE*, Jianbing Ni, *Member, IEEE*, Cheng Huang, *Student Member, IEEE*, and Xuemin Shen, *Fellow, IEEE*

***Abstract*—Crowdsourcing-based traffic monitoring plays an important role in advanced traffic management systems due to its high accuracy and low costs, but it may expose drivers real identities and sensitive locations that results in the privacy leakage of drivers. In this paper, we propose a crowdsourcing-based traffic monitoring scheme that enables a transportation management center (TMC) to achieve traffic flow statistics at road intersections in an efficient, verifiable, and privacy-preserving manner. Specifically, by integrating a homomorphic encryption primitive and a super-increasing sequence, traffic flow can be flexibly structured and encrypted by drivers, i.e., each drivers travel direction at T-junctions or crossroads is protected. As a middle-ware between drivers and TMC, roadside units (RSUs) are introduced to aggregate and further perturb the aggregated encrypted traffic flow based on a differential privacy mechanism. In this way, TMC is capable of acquiring the traffic flow statistics by decrypting the perturbed encrypted traffic flow, without disclosing each individual drivers traffic information. In addition, based on a lightweight commitment proof, the correctness of the encrypted drivers data can be guaranteed, i.e., a selfish driver cannot arbitrarily manipulate his data to poison the aggregated traffic flow. Finally, security analysis demonstrates that the proposed scheme satisfies all desirable security properties, including confidentiality, verifiability, unlinkability, and traceability. Extensive simulations are also conducted to show that the proposed scheme is efficient in terms of low computation and communication costs.**

***Index Terms*—Efficiency, privacy, traffic management, traffic flow statistics, verifiability.**

I. Introduction

ADVANCED traffic management systems (ATMS), which involve various infrastructures such as roadside units (RSUs) and transportation management center (TMC), have been considered as a primary part of intelligent transportation systems (ITS) [1]. By means of the widely deployed roadside traffic sensors and communication devices, ATMS can collect and analyze vehicles' real-time traffic information and take effective measurements to reduce traffic delays. In ATMS, smart traffic lights are considered as one of most components for traffic management [2]. Different from traditional traffic lights with fixed-cycles, smart traffic lights can dynamically control the traffic signal to reduce drivers' waiting delay and improve traffic efficiency. Since traffic lights are usually deployed at road intersections, how to collect real-time traffic information to monitor traffic flow at these intersections is an essential problem in ATMS.

Currently, there are many mechanisms that can be used to estimate traffic flow at intersections, such as video analysis [3], time-spatial image processing [4], UAV-empowered edge computing [5], and mobility trace data analysis [6]. Among them, crowdsourcing-based traffic flow collection, as a simple but effective way, has received considerable attention in recent years. Several applications, such as Google Maps [7], WAZA [8], and a series of research studies [9]–[14] have been developed and presented. Generally, a crowdsourcing-based traffic flow statistics at intersections approach works as follows: a driver intending to pass through a road intersection sends her travel direction to a nearby RSU. The RSU can cooperate with TMC to estimate the traffic flow for each direction in an aggregated way. Based on the aggregated results, traffic congestion points can be effectively obtained and potentially be mitigated or avoided by rescheduling the traffic lights. In spite of the appealing benefits, some new challenges are triggered that may impede the flourish of such a crowdsourcing-based traffic flow statistics system.

Privacy concern is one of the most serious issues. Since drivers' travel directions usually contain their next-step locations, knowing a driver's travel direction may not only violate the driver's location privacy, but may put the driver in a dangerous situation such as being stalking and robbery. As a result, drivers may be reluctant to upload their travel directions to non-fully trusted third parties (i.e., RSUs and TMC). Hence, appropriate privacy-preserving mechanisms should be in place to prevent drivers' sensitive identities and locations information from being

Manuscript received January 31, 2020; revised April 21, 2020; accepted May 27, 2020. Date of publication June 29, 2020; date of current version October 13, 2020. This work was supported in part by the National Natural Science Foundation of China Grants 61272512 and 61402037, in part by the Grants of Natural Sciences and Engineering Research Council of Canada (NSERC), and in part by the China Scholarship Council. The work of Chuan Zhang was supported by the colleagues from BBCR Lab, University of Waterloo. The review of this article was coordinated by Dr. Z. Cai. *(Corresponding author: Liehuang Zhu.)*

Chuan Zhang is with the Beijing Engineering Research Center of Massive Language Information Processing and Cloud Computing Application, School of Computer Science and Technology, Beijing Institute of Technology, Beijing 100811, China, and also with the Department of Electrical and Computer Engineering, University of Waterloo, Waterloo, ON N2L 3G1, Canada (e-mail: chuanz@bit.edu.cn).

Liehuang Zhu is with the Beijing Engineering Research Center of Massive Language Information Processing and Cloud Computing Application, School of Computer Science and Technology, Beijing Institute of Technology, Beijing 100811, China (e-mail: liehuangz@bit.edu.cn).

Jianbing Ni is with the Department of Electrical and Computer Engineering, Queens University, Kingston, ON K7L 3N6, Canada (e-mail: jianbing.ni@queensu.ca).

Cheng Huang and Xuemin Shen are with the Department of Electrical and Computer Engineering, University of Waterloo, Waterloo, ON N2L 3G1, Canada (e-mail: c225huan@uwaterloo.ca; sshen@uwaterloo.ca).

Digital Object Identifier 10.1109/TVT.2020.3005363

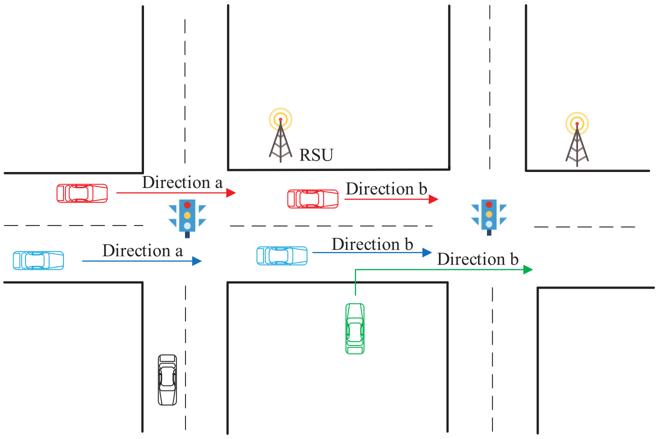


Fig. 1. An example of statistical attack.

publicly disclosed. Ideally, a traffic flow statistics scheme should not disclose any individual's privacy-aware information other than the aggregated traffic flow. Such requirements are also compliant with the strict personal data protection regulations law, such as the General Data Protection Regulation (GDPR) [15].

Intuitively, encrypting drivers' traffic information is a good choice to preserve their privacy. However, an individual driver's travel direction may be inferred by analyzing the traffic flows via a particular statistical attack [16]. As shown in Fig. 1, two vehicles (i.e., the red and blue vehicles) have been traveling in direction a, and they are going to turn to direction b. At the same time, another vehicle (i.e., the green vehicle) is also going to direction b. Although the green vehicle has encrypted its data, adversaries can still obtain the green vehicle's travel direction by analyzing the traffic flows of directions a and b between two consecutive intersections. By launching such an attack, a driver's trajectory may be reconstructed and the driver may be further identified from the trajectory information.

In addition, data correctness is also an important issue in crowdsourcing-based traffic flow statistics. Ideally, drivers are expected to provide truthful direction data for traffic monitoring. However, some drivers are likely to provide untruthful data for their selfish, malicious, or other unforeseeable purposes. For example, a driver may submit a valid report, say that there are 50 vehicles that will pass by the direction i, though the actual number is only 1. Such behaviors can seriously pollute the aggregated traffic information and thus need special consideration. Since the direction data is encrypted, how to verify the correctness of drivers' data in ciphertexts while not disclosing drivers' private information is a significant challenge.

In this paper, we propose a Verifiable and Privacy-preserving Traffic Flow Statistics (VPTS) scheme to cope with the above-mentioned issues simultaneously. Concretely, VPTS employs RSUs to collect drivers' travel directions where traffic lights are located and enables RSUs and TMC to cooperatively aggregate drivers' directions via an efficient and privacy-preserving way. To prevent the data pollution attack, a lightweight commitment proof is designed such that only the legitimate data can pass the verification. Specifically, our contributions are summarized as three-folds.

- Firstly, we consider crowdsourcing-based traffic flow statistics at road intersections and propose VPTS that incorporates pseudonyms, homomorphic encryption cryptosystem, and differential privacy to strictly protect drivers identities and locations privacy.
- Secondly, a well-designed lightweight commitment proof mechanism is proposed such that RSUs can verify the correctness of drivers data, without privacy leakage.
- Thirdly, through a detailed security analysis, we demonstrate that VPTS satisfies all desirable security properties, including confidentiality, verifiability, unlinkability, and traceability. Extensive simulations are also conducted to show that VPTS has low computation and communication overhead on both driver and server sides.

The remainder of this paper is organized as follows. In Section II, we present the system model, define the security threats, and identify the design goals. In Section III, we describe the building blocks, and propose VPTS along with the correctness and security analysis in Section IV. In Section V, we evaluate the performance. Finally, Section VI reviews related works and Section VII concludes the paper.

II. Models and Design Goals

A. System Model

The system model of VPTS consists of the following four parties, i.e., a trusted authority (TA), a TMC, a set of RSUs, and drivers, as depicted in Fig. 2.

- TA: TA is an entity with full trust. It initializes the system, generates system parameters, and assigns unique identity information for participating entities. After initialization, TA stays offline until there is a need to trace a driver.
- TMC: TMC collects and analyzes the traffic data submitted by distributed RSUs. After obtaining the traffic conditions, it will manage or control road infrastructures such as traffic lights to keep smooth traffic.
- RSUs: RSUs are subroutines of TMC. They are widely distributed on roads and act as a bridge between drivers and TMC. They collect data, process data, and cooperate with TMC to obtain traffic conditions.
- Drivers: Drivers are data providers and they have a desire for smooth travel. With this wish, they are willing to share their traffic data by using smart devices, such as smartphones or onboard units.

B. Security Threats

We state security threats by analyzing the trustworthiness of each entity.

First of all, TA is fully trusted. Since TA initializes the system and holds all drivers' personal information, it cannot be compromised. RSUs and TMC are semi-honest, which means they will honestly perform the designed protocol, but may try to infer drivers' private information, such as identities, locations, and trajectories for additional benefits, including but not limited to targeted advertising and data sales. Especially, since TMC obtains the aggregated results, it may launch *statistical attack* to

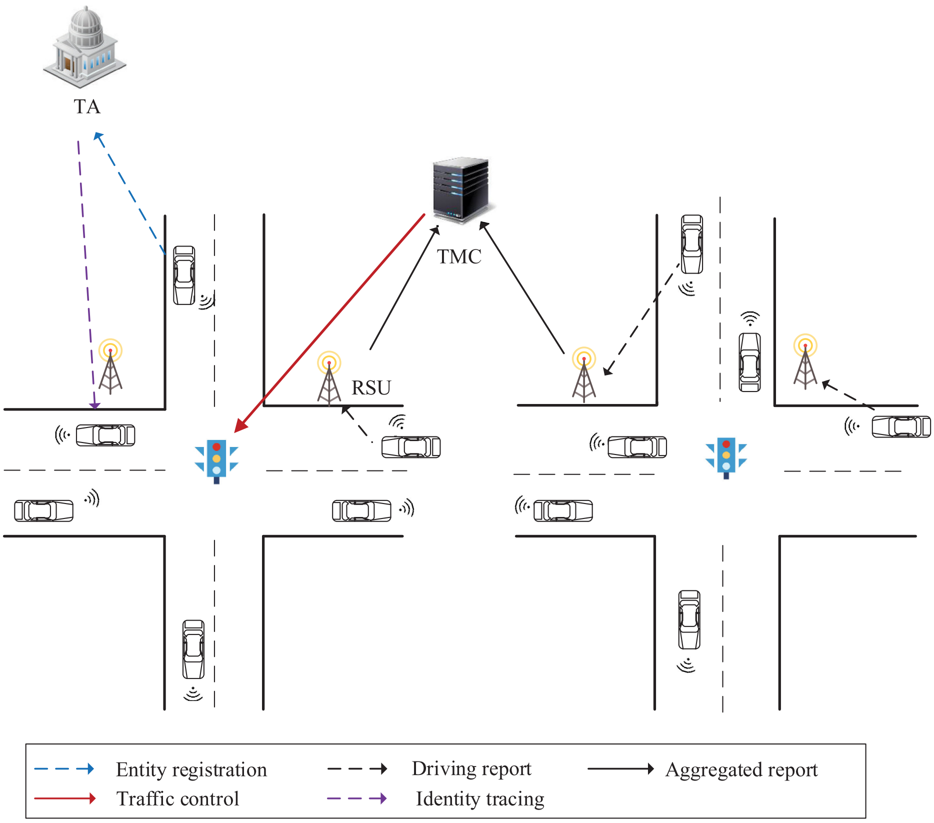


Fig. 2. System model.

infer an individual driver's location privacy. Meanwhile, since TMC holds the secret key, we assume there is no collusion between TMC and RSUs, which is similar to most of the existing traffic monitoring schemes [12], [13].

Although drivers are data providers, they may also bring threats to the system. In particular, some drivers may launch a *data pollution attack* by providing untruthful direction data. For instance, a selfish driver may report to an RSU that there will be more than 1 driver passing her direction. Note that, we do not consider the situation that a driver provides legitimate but untruthful data, e.g., reporting to the RSU that she will pass direction i but pass direction j. Since most of drivers want to reduce waiting time at crossroads, such behaviors will bring no benefits to the driver. Furthermore, similar to [11]–[13], the Sybil attack is not considered in this paper, as it can be effectively addressed by the authentication of physical devices [17], [18].

Besides the internal threats, external adversaries are also considered. They may compromise RSUs to obtain drivers' private information. In addition, they may violate drivers' privacy by eavesdropping on wireless communication channels and break the system by forging or modifying drivers' reports.

C. Design Goals

In response to the above security threats, the following designs goals should be captured in VPTS.

1) Security Goals:

- *Confidentiality.* Drivers' data should be strictly protected. RSUs and TMC cannot identify drivers' actual direction contents from their submitted reports. In addition, TMC cannot infer an individual driver's direction based on the aggregated results, i.e., the proposed scheme should withstand the statistical attack.
- *Verifiability.* Drivers' direction data should be in a reasonable range. Untruthful direction data that are out of the reasonable range should be detected by the RSU, i.e., the proposed scheme should withstand the data pollution attack.
- *Unlinkability.* The driver's identity should be protected. That is, given two reports, entities cannot identify if they are submitted by the same driver.
- *Authentication and data integrity.* The driver's data should be authenticated whether it is submitted by a registered driver. That is, the report that is forged or modified should be detected and rejected.
- *Traceability.* Even if a driver's real identity is hidden in her report, the system should be capable of tracking a driver's real identity.

2) Performance Goals:

- *Correctness.* The aggregated results should be guaranteed to be correct. That is, the final result for each direction should be aggregated by drivers' corresponding submitted direction data and the added noises should be generated from a correct distribution.
- *Efficiency.* Considering the resource limitation of smart devices and RSUs, the computation and communication costs introduced on the driver and RSU sides should be as minimal as possible.

III. Preliminaries

In this section, we review the cryptographic building blocks, including BGN encryption cryptosystem and bilinear groups, which serve as the basis of VPTS.

A. *BGN Encryption Cryptosystem*

The BGN encryption cryptosystem has the following algorithms.

- $\mathsf{Gen}(\kappa)$: Given a security parameter κ, two cyclic groups G, G_1 with the same order n are first generated, where $n = pq$ and p, q are two large prime numbers. Two random generators $g, u \in G$ are selected and $h = u^p \in G_q$ is calculated, where G_q is a subgroup of G with order q. The public key is represented as $\mathcal{PK} = (n, G, G_1, e, g, h)$ and the secret key is $\mathcal{SK} = q$.
- $\mathsf{Encrypt}(\mathcal{PK}, m)$: After picking a random value $r \xleftarrow{R} \{0, 1, \cdots, n-1\}$, the message $m \in \{0, 1, \cdots, T\}$ with $T < p$ is encrypted as $C = E(m) = g^m h^r \in G$.
- $\mathsf{Decrypt}(\mathcal{SK}, C)$: The ciphertext C is calculated by using the private key $\mathcal{SK} = q$ as $C^q = D(C) = (g^m h^r)^q = (g^q)^m$. With the Pollard's lambda algorithm, the message m can be recovered in the expected time of $O(\sqrt{T})$.

BGN cryptosystem holds the additive homomorphic property, which can be expressed as $E(m_1) * E(m_2) = E(m_1 + m_2)$. The security and homomorphic properties of the BGN cryptosystem have been proven in [19], we omit the details here.

B. *Bilinear Groups*

Given two (multiplicative) cyclic groups G, G_1 with the same order n. Let g_1 be a generator of the group G and $e(g_1, g_1)$ be a generator of G_1. There is a bilinear pairing map $e : G \times G \to G_1$ that has the following properties.

- Bilinear: For any $u, v \in G$ and all $a, b \in Z_q^*$, $e(u^a, v^b) = e(u, v)^{ab} \in G_1$.
- Non-degenerate: There exists $g \in G$ such that $e(g, g) \neq 1$.
- Computable: Given any $u, v \in G, e(u, v)$ can be efficiently computed.

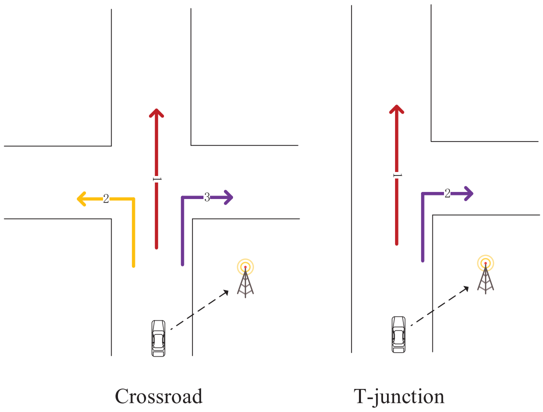
Referring to [20], [21], we have the following definitions.

Definition 1: Elliptic Curves Discrete Logarithm (ECDL) problem). Given random points g_1 and g_1^a, where $g_1, g_1^a \in G, a \in Z_n^*$, it is infeasible to recover a from g_1^a.

Definition 2: Computational Diffe-Hellman (CDH) Problem. Given elements $(g_1, g_1^a, g_1^b) \in G$, where g_1 is the generator of G and $a, b \in Z_n^*$ are unknown values, it is computationally intractable to compute g_1^{ab}.

IV. Proposed VPTS

In this section, we first consider a scenario where drivers are all honest but curious, i.e., the drivers will submit truthful data. Based on this assumption, we introduce the baseline scheme to depict the overall workflow and lay the foundation of our enhanced design. Then, we will give a refinement to the baseline scheme, which can guarantee the verifiability. At last, we give correctness and security analysis to demonstrate that VPTS is effective and is capable of achieving all desirable security properties.



Crossroad

T-junction

Fig. 3. Directions division of crossroad and T-junction.

A. *A Baseline Scheme of VPTS*

1) System Initialization: With a security parameter κ, TA first runs $\mathsf{BGN.Gen}(\kappa)$ to obtain BGN cryptosystem's public key and private key, i.e., $\mathcal{PK} = (n, G, G_1, e, g, h)$ and $\mathcal{SK} = q$. Then, TA chooses a cryptographic hash function as: $H : \{0, 1\}^* \to G$ and selects a generator $g_1 \neq g$ from G. After that, TA publishes $(\mathcal{PK}, H, g_1)$ to all entities and sends $\mathcal{SK}$ to TMC via a secure communication channel.

The road interaction where a traffic light is located has several lanes and each lane has several directions based on the travel direction. Here, we consider two common road interactions, i.e., crossroad and T-junction, and their directions division and representation are given in Fig. 3. For each lane l, an RSU $\mathcal{R}_l$ is deployed to collect the direction information submitted by drivers. In particular, we use a super-increasing sequence $\overrightarrow{a}$ to represent the travel directions as $a_1 \in Z_q^*, \sum_{i=1}^{j-1} a_i \cdot Q < a_j, \sum_{i=1}^{M} a_i \cdot Q < q$, where $j \in [2, M]$, M is the number of travel directions, and Q is a constant value that is greater than the maximum number of vehicles passing in one direction in a time period (e.g., a traffic light interval). After that, TMC publishes $\mathsf{Direction}(l) = (\mathcal{R}_l, \{a_i\}_{i=1}^{M})$.

2) Entity Registration: Before participating in the system, drivers are required to register themselves in TA. To preserve drivers' identities information, pseudonyms [22], short group signature [23], and conditional privacy-preserving authentication (CPPA) protocol [20] can be considered. In this work, similar to existing privacy-preserving traffic flow statistics schemes [10], [12], we also adopt the technique of pseudonyms. Specifically, a driver v_j submits her identity ID_j (e.g., license number) to TA. After verifying the validity of v_j's real identity, TA selects random values $k_1, k_2, \cdots, k_o$, calculates $K_1 = g_1^{k_1}, K_2 = g_1^{k_2}, \cdots, K_o = g_1^{k_o}$, and generates v_j's pseudonyms $PID_i = AES_{k_0}(ID_j || k_i)$, where the symmetric key is k_0 and $i \in [1, o]$. Then, TA sends $\{PID_i, K_i\}$ to the RSU $\mathcal{R}_l$ and returns $\{PID_i, k_i, K_i\}_{i=1}^{o}$ to v_j. $\mathcal{R}_l$ is also required to register itself to TA with its identity ID_l (e.g., location). Similar to v_j, $\mathcal{R}_l$ obtains its identity information as (ID_l, k_l, K_l).

3) Report Generation: In coverage area, $\mathcal{R}_l$ broadcasts the requirement of traffic flow statistics, which consists of the current time T and the direction information $\mathsf{Direction}(l)$. On receiving this requirement, a driver v_j which intends to pass through the road intersection will generate her report as follows.

- *Direction generation:* Based on the travel direction $b \in [1, M]$, v_j first calculates $a^j = \sum_{i=1}^{M} x_i^j \cdot a_i$, where $x_b^j = 1, \{x_i^j = 0\}_{i=1, i \neq b}^{M}$. v_j then encrypts a^j as $C_j = g^{a^j} h^{r_j}$ by selecting a random value $r_j \in [0, n-1]$.
- *Message signing:* v_j calculates a hash value as $H_j = H(PID_j || T_j || C_j)$, where T_j is the upload time. She then signs the value to obtain the signature as $\sigma_j = H_j^{k_j}$.

At last, v_j sends the report $Msg_j = (PID_j, T_j, C_j, \sigma_j)$ to $\mathcal{R}_l$.

4) Reports Aggregation: After receiving drivers' reports, $\mathcal{R}_l$ is expected to verify the validity of drivers' data and perform reports aggregation. For a driver v_j, $\mathcal{R}_l$ first checks if $|T - T_j|$ is less than a threshold and then checks if $e(g_1, \sigma_j)$ equals to $e(K_j, H_j)$. If it does hold, the report is legitimate and can be accepted, since $e(g_1, \sigma_j) = e(g_1, H_j^{k_j}) = e(g_1^{k_j}, H_j) = e(K_j, H_j)$. To improve efficiency, we use the technique of batch verification here to reduce the number of pairing operations, which is calculated as

$$e\left(g_1, \prod_{j=1}^{N} \sigma_j\right) = e\left(g_1, \prod_{j=1}^{N} H^{k_j}(PID_j || T_j || C_j)\right) = \prod_{j=1}^{N} e(K_j, H(PID_j || T_j || C_j)), \qquad (1)$$

where N is the number of reports $\mathcal{R}_l$ has received. Then, $\mathcal{R}_l$ performs ciphertexts aggregation to obtain $\prod_{j=1}^{N} C_j$. Note that, to defend against the statistical attack launched by TMC, $\mathcal{R}_l$ is also required to add a noise on the aggregated result for each direction, which is calculated as

$$C_l = g^{\sum_{i=1}^{M} a_i \gamma_i} \cdot \prod_{j=1}^{N} C_j, \qquad (2)$$

where γ_i is a random value (i.e., noise) generated by a geometric distribution $\mathsf{Geom}(\alpha)$ and α is a parameter that can control the privacy loss and data accuracy. Since the aggregated results are discrete in our scenario, similar to [24], [25], the geometric distribution $\mathsf{Geom}(\alpha)$ used in this paper is with $\alpha \in (0, 1)$ and the probability density function $\Pr[X = x] = \frac{1-\alpha}{1+\alpha} \cdot \alpha^{|x|}$.[1] At last, $\mathcal{R}_l$ sends the report $Msg_l = (ID_l, T_l, C_l, \sigma_l)$ to TMC, where $\sigma_l = H^{k_l}(ID_l || T_l || C_l)$.

5) Data Recovery: On receiving the report sent by $\mathcal{R}_l$, similar to the previous step, TMC checks the delivery time and validity of the report. After that, TMC performs the following steps to calculate the number of drivers that will pass each direction.

Algorithm 1: Recover DR_i.

```
Set $X_M = \sum_{i=1}^{M} a_i(\gamma_i + \sum_{j=1}^{N} x_i^j)$;
for $i = M$ to 2 do
    $X_{i-1} = X_i \bmod a_i$;
    $DR_i = (X_i - X_{i-1})/a_i$;
end for
$DR_1 = X_1/a_1$
return $(DR_1, DR_2, \cdots, DR_M)$
```

- *Ciphertext decryption:* With the secret key q and the ciphertext C_l, TMC runs $\mathsf{BGN.Decrypt}(q, C_l)$ and obtains $\sum_{i=1}^{M} a_i(\gamma_i + \sum_{j=1}^{N} x_i^j)$.
- *Data recovery:* With $\sum_{i=1}^{M} a_i(\gamma_i + \sum_{j=1}^{N} x_i^j)$, TMC runs the Algorithm 1 to recover the aggregated results $(DR_1, DR_2, \cdots, DR_M)$, where $DR_i = \gamma_i + \sum_{j=1}^{N} x_i^j$.

Based on $\{DR_i\}_{i=1}^{M}$, TMC will know the future traffic flow in each direction and then it can manage or control the infrastructures such as rescheduling the traffic lights to ensure the smooth flow of traffic.

B. The Extension of VPTS: Achieving Verifiability

Although the baseline scheme can achieve private traffic flow statistics, a valid value submitted by a selfish driver may spoil the final aggregated results. For example, if a driver v_j's direction is 1, she is supposed to submit direction data as g^{a_1}. To deceive TMC, v_j may submit the direction data as $g^{k \cdot a_1}$ where $k \gg 1$. On receiving the data, TMC will be misguided to consider that there will be congestion on direction 1, and thus reschedule the traffic light for v_l. We, therefore, need to check the data submitted by each driver and guarantee the system's robustness. To achieve this goal, we require that each driver should make a direction commitment to prove that her data is in a reasonable range.

We note that some protocols have been proposed to try to deal with the challenge of data verifiability, such as the non-interactive zero-knowledge (NIZK) proof [26], the zk-SNARKs [27], and the solution presented by Duan *et al.* [16]. However, these proofs have large proof generation costs and communication overhead, which is not friendly to the resource-limited smart devices. Thus we devise a new lightweight commitment proof mechanism. In particular, the commitment proofs are custom-designed for both crossroad and T-junction. Specifically, we first design a commitment proof mechanism for crossroads. This mechanism can be applied in all road interactions. We then further improve this mechanism for the scenario of T-junction to save the computational and communication costs. The detailed procedure of VPTS is given in Fig. 4 and the red contents are the improvements compared with the baseline scheme.

C. Correctness Analysis

The correctness of VPTS depends on whether the aggregated result for each direction can be correctly calculated and whether the direction data can be correctly verified. To proof that, we

[1] As explained in [24], [25], the $\mathsf{Geom}(\alpha)$ can be considered as a discrete approximation of Laplace distribution $Lap(\lambda)$, where $\lambda = \frac{\Delta f}{\epsilon}$, $\alpha \approx \exp(-\frac{1}{\lambda})$. Δf is the sensitivity and ϵ is the privacy budget.

A Verifiable and Privacy-Preserving Traffic Flow Statistics Scheme

★ **System Initialization**:

1: TA initializes the system: $\mathsf{BGN.Gen}(\kappa) \to \{\mathcal{PK} = ((n, G, G_1, e, g, h), \mathcal{SK} = q\},\ H : \{0,1\}^* \to G,\ g_1 \neq g \in G$

2: TMC generates the direction parameters: $\mathsf{Direction}(l) = (\mathcal{R}_l, \{a_i\}_{i=1}^{M})$

TA publishes $(\mathcal{PK}, H, g_1, \mathsf{Direction}(l))$, TA → TMC : $\mathcal{SK}$

★ **Entity Registration:**

1: v_j obtains her identity information: $ID_j \to (PID_j, k_j, K_j)$

2: $\mathcal{R}_l$ obtains its identity information: $ID_l \to (ID_l, k_l, K_l)$

★ **Report Generation**:

1: For the crossroad, v_j selects a direction (e.g., direction b) and generates the ciphertexts and corresponding commitments:

$$C_i^j = g^{a_i^j} h^{r_i^j}; \qquad \mathfrak{C}_i^j = (g^{2a_i^j - a_i} h^{r_i^j})^{r_i^j},$$

where $i \in [1, M], a_b^j = a_b, \{a_i^j = 0\}_{i=1, i\neq b}^{M}, r_i^j \in [0, n-1]$

For the T-junction, v_j selects a direction (e.g., direction b) and generates the ciphertext and corresponding commitment:

$$C_j = g^{a^j} h^{r_j}; \qquad \mathfrak{C}_j = (g^{2a^j - a_1 - a_2} h^{r_j})^{r_j},$$

where $a^j = \sum_{i=1}^{M} x_i^j \cdot a_i, x_b^j = 1, \{x_i^j = 0\}_{i=1, i\neq b}^{M}, r_j \in [0, n-1]$

2: v_j generates the signature: $\sigma_j = H^{k_j}(PID_j || T_j || \{C_j^i || \mathfrak{C}_j^i\}_{i=1}^{M})$ or $\sigma_j = H^{k_j}(PID_j || T_j || C_j || \mathfrak{C}_j)$

$v_j \to \mathcal{R}_l : Msg_j = (PID_j, T_j, \{C_j^i, \mathfrak{C}_j^i\}_{i=1}^{M}, \sigma_j)$ or $Msg_j = (PID_j, T_j, C_j, \mathfrak{C}_j, \sigma_j)$

★ **Report Aggregation:**

1: $\mathcal{R}_l$ verifies the signature: $e(g_1, \sigma_j) \overset{?}{=} e(K_j, H_j)$

2: $\mathcal{R}_l$ verifies the commitments:

Crossroad: $e(C_i^j, g^{-a_i} C_i^j) \overset{?}{=} e(h, \mathfrak{C}_i^j), i \in [1, M]$; T-junction: $e(g^{-a_1} C_j, g^{-a_2} C_j) \overset{?}{=} e(h, \mathfrak{C}_j)$

3: $\mathcal{R}_l$ aggregates the ciphertexts and adds noises:

Crossroad: $C_l = g^{\sum_{i=1}^{M} a_i \gamma_i} \prod_{j=1}^{N} \prod_{i=1}^{M} C_j^i$; T-junction: $C_l = g^{\sum_{i=1}^{M} a_i \gamma_i} \prod_{j=1}^{N} C_j$

4: $\mathcal{R}_l$ generates the signature: $\sigma_l = H^{k_l}(ID_l || T_l || C_l)$

$\mathcal{F}_l \to$ TMC : $Msg_l = (ID_l || T_l || C_l || \sigma_l)$

★ **Data Recovery:**

1: TMC verifies the signature: $e(g_1, \sigma_l) \overset{?}{=} e(K_l, H(ID_l || T_l || K_l || C_l))$

2: TMC decrypts the ciphertexts: $\mathsf{BGN.Decrypt}(q, C_l)$

3: TMC recovers the aggregated results: Algorithm 1.

Fig. 4. Details of a verifiable and privacy-preserving traffic flow statistics scheme.

demonstrate the Eq. 2, Algorithm 1, and the commitment proof here.

1) *Correctness of Formula (2):* From the homomorphic property of BGN cryptosystem, we can get

$$\begin{aligned} C_l &= \prod_{j=1}^{N} C_j = g^{\sum_{i=1}^{M} a_i \gamma_i} \cdot \prod_{j=1}^{N} g^{a^j} h^{r_j} \\ &= g^{\sum_{i=1}^{M} a_i \gamma_i + \sum_{j=1}^{N} a^j} h^{\sum_{j=1}^{N} r_j} \\ &\xrightarrow{\text{since } a^j = \sum_{i=1}^{M} a_i x_i^j,\ \text{where } x_b^j = 1, \{x_i^j = 0\}_{i=1, i\neq b}^{M}} \\ &= g^{\sum_{i=1}^{M} a_i \gamma_i + \sum_{i=1}^{M} \sum_{j=1}^{N} a_i x_i^j} h^{\sum_{j=1}^{N} r_j} \\ &= g^{\sum_{i=1}^{M} a_i (\gamma_i + \sum_{j=1}^{N} x_i^j)} h^{\sum_{j=1}^{N} r_j} \\ \Rightarrow \mathsf{Decrypt}(q, C_l) &= \sum_{i=1}^{M} a_i \left(\gamma_i + \sum_{j=1}^{N} x_i^j \right) \end{aligned}$$

2) *Correctness of Algorithm 1:* In Algorithm 1, we have $X_M = \sum_{i=1}^{M} a_i(\gamma_i + \sum_{j=1}^{N} x_i^j)$, which means,

$$X_M = a_1 \left(\gamma_1 + \sum_{j=1}^{N} x_1^j \right) + \cdots + a_M \left(\gamma_M + \sum_{j=1}^{N} x_M^j \right).$$

Since Q is much larger than $\sum_{j=1}^{N} x_i^j$, we could control $\triangle f, \epsilon$, i.e., α, to make $\gamma_i + \sum_{j=1}^{N} x_i^j < Q$, thus we will have

$$\begin{aligned} &a_1 \left(\gamma_1 + \sum_{j=1}^{N} x_1^j \right) + \cdots + a_{M-1} \left(\gamma_{M-1} + \sum_{j=1}^{N} x_{M-1}^j \right) \\ &= \sum_{i=1}^{M-1} a_i \left(\gamma_i + \sum_{j=1}^{N} x_i^j \right) < \sum_{i=1}^{M-1} a_i Q < a_M. \end{aligned}$$

Therefore, $X_{M-1} = \sum_{i=1}^{M-1} a_i(\gamma_i + \sum_{j=1}^{N} x_i^j) = X_M \mod a_M$, and $DR_M = (X_M - X_{M-1})/a_M$. With the same process, we can obtain $(DR_1, DR_2, \cdots, DR_M)$.

3) *Correctness of commitment proof:* As shown in Fig. 4, crossroad and T-junction are both considered in VPTS and their commitments are generated in different ways. Therefore, we prove the correctness of our presented commitment design in the following two cases.

Case 1: We first consider the crossroad, where $M = 3$. In this case, a driver generates M ciphertexts for all directions and generates M corresponding commitments for direction verifiability. With the ciphertext C_i^j and the commitment $\mathfrak{C}_i^j$, $\mathcal{R}_l$ calculates $e(C_i^j, g^{-a_i}C_i^j)$ and checks if $e(C_i^j, g^{-a_i}C_i^j)$ equals to $e(h, \mathfrak{C}_i^j)$. If it does hold, the ciphertext can be accepted, since $e(C_i^j, g^{-a_i}C_i^j) = e(g^{a_i^j}h^{r_i^j}, g^{a_i^j-a_i}h^{r_i^j}) = e(g^{a_i^j}, g^{a_i^j-a_i})e(h^{r_i^j}, g^{a_i^j-a_i})e(g^{a_i^j}, h^{r_i^j})e(h^{r_i^j}, h^{r_i^j}) = e(g,g)^{a_i^j(a_i^j-a_i)}e(h, (g^{2a_i^j-a_i} h^{r_i^j})^{r_i^j})$. We can see only if $a_i^j = 0$ or $a_i^j = a_i$, $e(C_i^j, g^{-a_i}C_i^j)$ will be equal to $e(h, \mathfrak{C}_i^j)$. Untruthful data that is out of $\{0, a_i\}$ will not pass the verification. Based on the above analysis, the correctness of our designed commitment proof when $M = 3$ is presented.

To reduce the number of time-consuming pairing operations, we also use the technique of batch verification in this case, which is calculated as follows.

$$
\begin{aligned}
&\prod_{j=1}^{N} e(C_i^j, g^{-a_i}C_i^j)\\
&\quad = \prod_{j=1}^{N} e(g,g)^{a_i^j(a_i^j-a_i)} e(h, (g^{2a_i^j-a_i}h^{r_i^j})^{r_i^j})\\
&\quad = e(g,g)^{\sum_{j=1}^{N} a_i^j(a_i^j-a_i)} \cdot \prod_{j=1}^{N} e(h, (g^{2a_i^j-a_i}h^{r_i^j})^{r_i^j})\\
&\quad = e(g,g)^{\sum_{j=1}^{N} a_i^j(a_i^j-a_i)} e\left(h, \prod_{j=1}^{N} \mathfrak{C}_i^j\right)\\
&\quad \xrightarrow{\text{If } \{a_i^j\in\{0,a_i\}\}_{j=1}^{N},\ \sum_{j=1}^{N} a_i^j(a_i^j-a_i)=0, \text{ then}}\\
&\quad \prod_{j=1}^{N} e(C_i^j, g^{-a_i}C_i^j) = e\left(h, \prod_{j=1}^{N} \mathfrak{C}_i^j\right). \qquad (3)
\end{aligned}
$$

Note: Although efficiency can be significantly improved by using batch verification, some drivers may deceive the TMC by colluding with each other. For example, two drivers (e.g., v_1, v_2) may try to pass the batch verification by providing untruthful direction data x_1, x_2, such that $x_1(x_1 - a_i) + x_2(x_2 - a_i) = 0$ and $x_1 + x_2 > 2a_i$. In the following, we demonstrate that even though two drivers collude with each other, their aggregated result will not be larger than $2a_i$.

Based on $x_1(x_1 - a_i) + x_2(x_2 - a_i) = 0$, we first have

$$
\begin{aligned}
&x_1(x_1 - a_i) + x_2(x_2 - a_i)\\
&= x_1^2 - x_1a_i + x_2^2 - x_2a_i\\
&= \left(x_1 - \frac{1}{2}a_i\right)^2 + \left(x_2 - \frac{1}{2}a_i\right)^2 - \frac{1}{2}a_i^2. \qquad (4)
\end{aligned}
$$

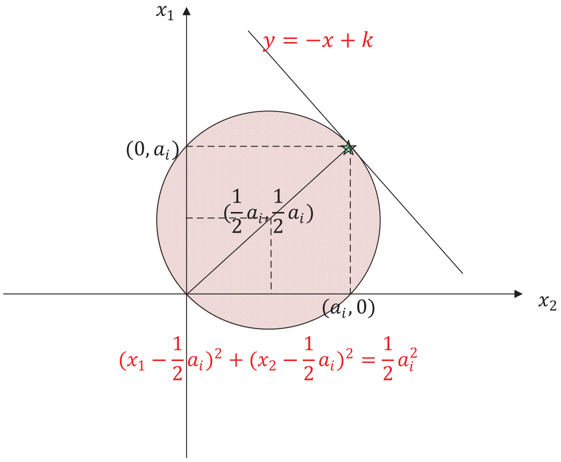

x_1
$y = -x + k$
$(0, a_i)$
$(\frac{1}{2}a_i, \frac{1}{2}a_i)$
$(a_i, 0)$
x_2
$(x_1 - \frac{1}{2}a_i)^2 + (x_2 - \frac{1}{2}a_i)^2 = \frac{1}{2}a_i^2$

Fig. 5. The maximum value of $x_1 + x_2$.

To achieve their selfish goal, v_1 and v_2 should guarantee $(x_1 - \frac{1}{2}a_i)^2 + (x_2 - \frac{1}{2}a_i)^2 = \frac{1}{2}a_i^2$ and $x_1 + x_2 > 2a_i$. However, from the equation $(x_1 - \frac{1}{2}a_i)^2 + (x_2 - \frac{1}{2}a_i)^2 = \frac{1}{2}a_i^2$, we can see it is a circle of radius $\frac{\sqrt{2}}{2}a_i$ and center coordinate $(\frac{a_i}{2}, \frac{a_i}{2})$, as shown in Fig. 5. According to the properties of plane geometry, only when (x_1, x_2) equals to (a_i, a_i) (i.e., the green point in Fig. 5), there is a maximum value of $x_1 + x_2$, which is $\text{Max}(x_1 + x_2) = 2a_i$. That is, if the two drivers want to pass the batch verification, the maximum aggregated result they obtain is $2a_i$, which means the selfish drivers cannot earn additional benefits by colluding with each other. With the similar analysis, we can prove the maximum value of $x_1 + x_2 + \cdots + x_N$ will be not larger than $N \cdot a_i$. Therefore, it is reasonable for using batch verification here to improve the efficiency of commitments verification.

Case 2: We then consider the T-junction, where $M = 2$. In this case, a driver only needs to perform one encryption and generate one corresponding commitment. With the ciphertext C_j and the commitment $\mathfrak{C}_j$, $\mathcal{R}_l$ calculates $e(g^{-a_1}C_j, g^{-a_2}C_j)$ and checks if $e(g^{-a_1}C_j, g^{-a_2}C_j)$ equals to $e(h, \mathfrak{C}_j)$. If it does hold, the ciphertext can be accepted, since

$$
\begin{aligned}
&e(g^{-a_1}C_j, g^{-a_2}C_j)\\
&= e(g^{a^j-a_1}h^{r_j}, g^{a^j-a_2}h^{r_j})\\
&= e(g^{a^j-a_1}, g^{a^j-a_2})e(g^{a^j-a_1}, h^{r_j})\\
&\times e(h^{r_j}, g^{a^j-a_2})e(h^{r_j}, h^{r_j})\\
&= e(g,g)^{(a^j-a_1)(a^j-a_2)}e(h, (g^{2a^j-a_1-a_2}h^{r_j})^{r_j})\\
&\xrightarrow{\text{if } a^j\in\{a_1,a_2\}, (a^j-a_1)(a^j-a_2)=0, \text{ then}}\\
&= e(h, \mathfrak{C}_j). \qquad (5)
\end{aligned}
$$

From Eq. 5, we can see only if $a^j = a_1$ or $a^j = a_2$, $e(g^{-a_1}C_j, g^{-a_2}C_j)$ will be equal to $e(h, \mathfrak{C}_j)$. Untruthful data that is out of $\{a_1, a_2\}$ will not pass the verification. Therefore, the correctness of the proposed commitment proof when $M = 2$ is presented.

Note: In this case, we integrate two directions data into one pairing operation, which can save more computation and communication overhead compared with Case 1, i.e., the pairing operation number can be reduced from 2 to 1. Nevertheless, we cannot use batch verification in this case, as two drivers may collude with each other to make $(x_1 - a_1)(x_1 - a_2) + (x_2 - a_1)(x_2 - a_2) = 0$, while $x_1 + x_2 > 2a_1$. Interested readers can prove this followed by a similar analysis as given in Case 1.

D. Security Analysis

We analyze how VPTS satisfies the security properties as described in Section. II-C.

Confidentiality. In VPTS, we use the BGN cryptosystem to protect the privacy of individual drivers' travel direction and use the differential privacy to defend against the statistical attack launched by TMC. We first prove the security of drivers' direction data on the RSU side. For the RSU $\mathcal{R}_l$, it knows a driver's ciphertext C_j and commitment $\mathfrak{C}_j$. Since the BGN cryptosystem has been proven to be semantic secure in [19] and the private key is only known by TMC, $\mathcal{R}_l$ cannot recover the direction data from the ciphertext. As for the commitment, the form of commitment can be expressed as $(A_j)^{r_j}$, where $A_j \in G$. Calculating r_j from $(A_j)^{r_j}$ is an ECDL problem. Therefore, RSU cannot recover the direction data from the commitment either.

For TMC, it knows the aggregated results $\gamma_i + \sum_{j=1}^{N} x_i^j$. However, TMC cannot recover private direction data of any other individual drivers due to the introduction of random noises. With the aggregated results, adversaries may infer individual drivers' direction information by performing sophisticated statistical analysis. In VPTS, differential privacy is applied to defend against such an attack. With the proofs given in [24], [25], we can see that $\gamma_i + \sum_{j=1}^{N} x_i^j$ achieves ϵ-differential privacy, i.e., given two perturbed aggregated results $\gamma_1 + \sum_{j=1}^{N} x_1^j, \gamma_1' + \sum_{j=1}^{N} x_{1'}^j$ for consecutive travel directions $\mathsf{Route}(1), \mathsf{Route}(1')$ that differ in at most one value, for any integer S, $\Pr[\gamma_1 + \sum_{j=1}^{N} x_1^j = S] \leq \exp(\epsilon) \cdot \Pr[\gamma_1' + \sum_{j=1}^{N} x_{1'}^j = S]$.

Based on the above analysis, the direction data of each driver will not be disclosed to RSUs and TMC under the proposed VPTS scheme.

Verifiability. To verify a driver's direction data, besides the ciphertexts, the driver is required to generate a corresponding commitment to prove her data is truthful. As analyzed in Eq. 5, only if a^j equals to one of the pre-designed direction data, the ciphertext can pass the verification. Untruthful data that are out of the range will be detected. Therefore, VPTS can defend against the data pollution attack and achieve verifiability.

Unlinkability. A driver's report is in a form of $(PID_j, T_j, C_j, \mathfrak{C}_j, \sigma_j)$. Due to the randomness of k_j, T_j, r_j, there are no same elements in two reports provided by the same driver. That is, no one can identify a driver based on her reports. Therefore, drivers' identity privacy is preserved.

Authentication and data integrity. In VPTS, driver's reports are signed by using the BLS short signature [23]. This technique has been proven to be secure in the random oracle model [28] if the CDH Problem is difficult to be solved. It is complete and unforgeable: 1) (*Completeness*) a registered user can obtain her valid credential and pass the verification; 2) (*Unforgeability*) a non-registered user cannot forge a valid credential and pass the verification. Therefore, the authentication and data integrity is achieved in VPTS.

Traceability: A driver's pseudonym is generated as $PID_j = AES_{k_0}(ID_j||k_j)$. With the symmetric key k_0, TA can recover a driver's real identity from the pseudonym.

V. Performance Analysis

In this section, we evaluate the performance of VPTS. We first analyze the aggregated results, and then conduct experiments to evaluate the system's efficiency in terms of computational costs and communication overhead.

A. Aggregation Analysis

According to Algorithm 1, the encryption of direction information will not influence the correctness of directions aggregation. Therefore, the correctness of the aggregated results depends on differential privacy. Here, we conduct experiments to observe the influence on the aggregated results with different ϵ choice, as shown in Fig. 6.

From Fig. 6(a), we can see there is a high deviation between the aggregated result and the truthful value when $\epsilon = 0.1$. In Fig. 6(b), the deviation decreases with the increase of ϵ. When ϵ increases to 1, the aggregated result is nearly the same as the truth value, as shown in Fig. 6(c). However, ϵ is inversely proportional to the privacy guarantee, which means a larger ϵ will cause much privacy loss [29]. Recall the aim of VPTS is to provide traffic control based on traffic flows, general aggregated results which can accurately reflect the future traffic flows would be sufficient in our scenario. Thus, to achieve a good trade-off between aggregation accuracy and privacy loss, ϵ is chosen as 0.5 in this paper.

B. Computational Costs

In this subsection, we first consider the computational complexity of VPTS and the result is summarized in Table. I. For simplicity of expression, we use $T_{G_m}, T_{G_e}, T_{G_p}$ to denote a multiplication operation, an exponentiation operation, and a pairing operation in G. In addition, we use T_h to denote a hash operation. We ignore the costs of addition and multiplication operations on plaintexts, as their costs are negligible compared to the operations on G. For efficiency comparison, a traditional scheme that also meets the requirement of data security and verifiability is considered. Specifically, the competing scheme adopts the BGN encryption system to encrypt the direction data x_i^j and adopts the zero-knowledge proof [26] to generate the commitment.

Then, we conduct experiments on an android phone with 6GB RAM and a laptop with 2.5GHz Intel Core i7, 16GB RAM. We use the android phone on the driver side and use the laptop on the RSU and TMC sides. The proposed VPTS and the traditional scheme are implemented by using Java with JPBC library,[2]

[2] http://gas.dia.unisa.it/projects/jpbc/

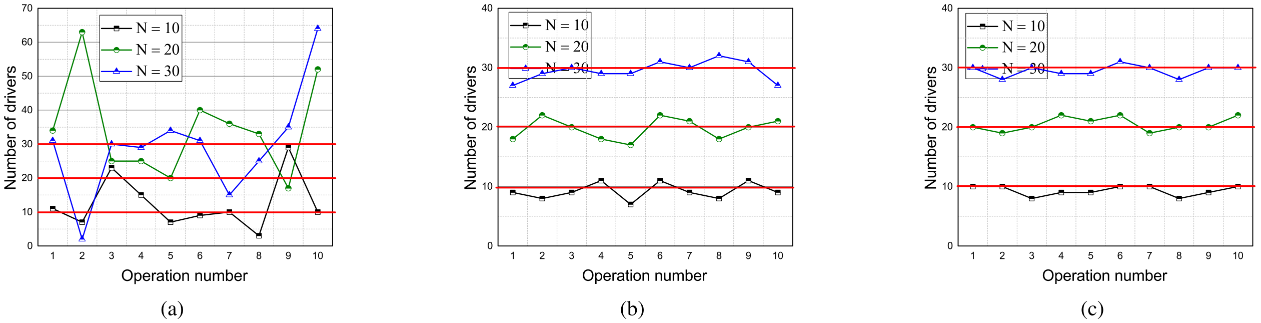


Fig. 6. Perturbed aggregated results, where (a) $\epsilon = 0.1$; (b) $\epsilon = 0.5$; (c) $\epsilon = 1$.

TABLE I
A SUMMARY OF COMPUTATIONAL COMPLEXITY

Entity	Phases	VPTS (Crossroad, $M = 3$)	VPTS (T-junction, $M = 2$)	Traditional scheme ($M = 2$ or 3)
Driver	Data encryption	$M(2T_{G_e} + T_{G_m})$	$2T_{G_e} + T_{G_m}$	$M(2T_{G_e} + T_{G_m})$
	Commitment generation	$M(3T_{G_e} + T_{G_m})$	$3T_{G_e} + T_{G_m}$	$M(5T_{G_e} + T_{G_m})$
	Signature generation	$T_h + T_{G_e}$	$T_h + T_{G_e}$	$T_h + T_{G_e}$
Fog	Report verification	$N(T_{G_m} + T_h + T_{G_p}) + T_{G_p}$	$N(T_{G_m} + T_h + T_{G_p}) + T_{G_p}$	$N(T_{G_m} + T_h + T_{G_p}) + T_{G_p}$
	Commitment verification	$M(N(2T_{G_m} + T_{G_p}) + T_{G_p})$	$N(2T_{G_m} + 2T_{G_p})$	$MN(T_{G_m} + 4T_{G_p})$
	Ciphertexts aggregation	$T_{G_e} + MNT_{G_m}$	$T_{G_e} + NT_{G_m}$	$3T_{G_e} + MNT_{G_m}$
TMC	Report verification	$2T_{G_p} + T_h$	$2T_{G_p} + T_h$	$2T_{G_p} + T_h$
	Decryption	T_{G_e}	T_{G_e}	T_{G_e}

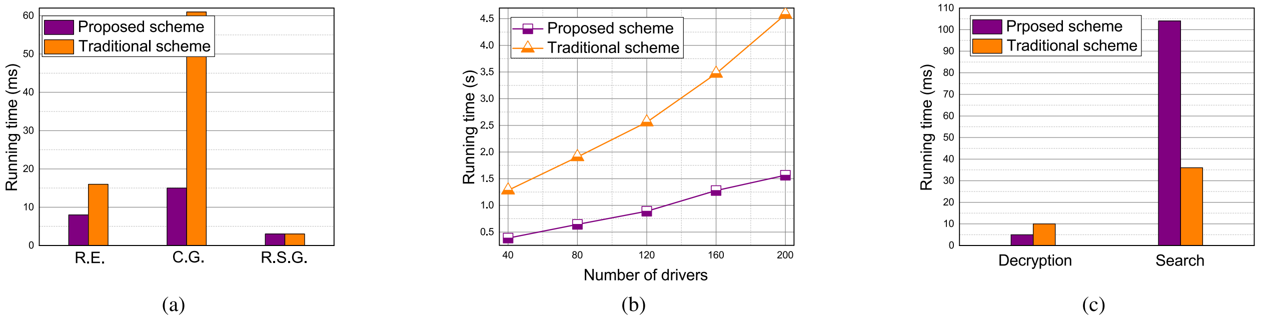


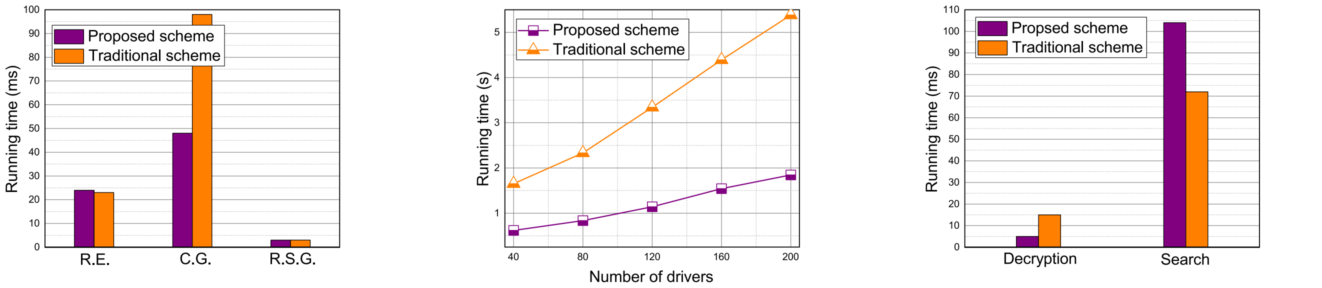
Fig. 7. Performance analysis between VPTS and the traditional scheme, when $M = 2$. (a)–(c) Computational costs on the driver side, RSU side, and TMC side, respectively.

where $|G| = 160$ bits and $Q = 50$. Each experiment is executed 10 times and we select the average result for comparison.

Case 1: We first consider the T-junction, where M is set as 2. In Fig. 7(a), we plot the running time of report generation on the driver side, where R.E. denotes direction encryption, C.G. denotes commitment generation, and R.S.G. denotes report signature generation. Thanks to the adoption of the well-designed super-increasing sequence, a driver can integrate multi-direction ciphertexts into one value, which can significantly reduce the computational costs on the driver side. With the enhanced commitment mechanism for T-junctions, compared with the basic commitment mechanism, a driver can perform fewer calculations and thus generate the commitment with less time. Specifically, a driver needs 8 ms to encrypt the direction, 15 ms to generate the commitment proof, and 3 ms to generate the signature in VPTS, while the traditional scheme needs 16 ms, 61 ms, and 3 ms, respectively. The experimental results confirm to the complexity analysis in Table I.

We then plot the computation time on the RSU side, as shown in Fig. 7(b). With the new design of the commitment and the utilize of super-increasing sequence, VPTS costs less time to perform commitment verification and ciphertexts aggregation. To illustrate, when the number of drivers reaches to 200, VPTS needs 1.565 s on the RSU side, while the traditional scheme requires 4.580 s.

We last consider the computational costs on TMC side. To recover the aggregated results, i.e., $X = \sum_{i=1}^{M} a_i(\gamma_i + \sum_{j=1}^{N} x_i^j)$, TMC needs to firstly conduct one exponentiation operation to calculate ${C_l}^q$, which costs 0.005s, and then find X from $(g^q)^X$. Since $X \in [0, \sum_{m=1}^{M} a_i Q]$, g^q, a_i, S are constant values, we can calculate and store all plaintext-ciphertext pairs in advance, so that to significantly reduce the recovery time. From Fig. 7(c),



(a)

(b)

(c)

Fig. 8. Performance analysis between VPTS and the traditional scheme, when $M = 3$. (a)–(c) Computational costs on the driver side, RSU side, and TMC side, respectively.

we can see VPTS needs less time to perform decryption, i.e., the the exponentiation operation, but consumes more time to find the aggregated results. The reason is that although the super-increasing sequence can integrate multi-users' ciphertexts into a single value, it also extends the scope of aggregated results, i.e., from Q to $\sum_{m=1}^{M} a_i \cdot Q$.

Case 2: We then consider the crossroad, where M equals to 3. We also plot the computational costs on the driver side, RSU side, and TMC side, as shown in Fig. 8(a), Fig. 8(b), and Fig. 8(c), respectively. On the driver side, since both schemes need to perform M encryption to preserve drivers' direction privacy, there is no obvious difference in the encryption time. Since VPTS is with a new commitment and fewer operations are conducted to generate the commitment proof, the commitment generation time is less than that of the traditional scheme, i.e., 48 ms *vs.* 98 ms. On the RSU side, since VPTS adopts the super-increasing sequence to integrate ciphertexts and uses batch verification to reduce pairing operations in reports verification and commitments verification, the consumed time is much less than that of the traditional scheme. For example, when the number of drivers reaches to 200, VPTS needs 1.846s on the RSU side, while the traditional scheme needs 5.383s. On the TMC side, VPTS performs better in the phase of decryption, but requires more time in the phase of aggregated results search.

C. Communication Overhead

The communication of VPTS includes two parts, i.e., driver-to-RSU communication and RSU-to-TMC communication. Since our scheme is designed for vehicles' future travel direction collection, the tolerance for communication delay is usually in seconds. Thus, the wireless communication technology applied for vehicle-to-RSU can be either cellular communications or dedicated short-range communications (DSRC) [30]. We first consider the driver-to-RSU communication, where drivers generate their direction reports and send the reports to the RSU. For the T-junction, the report a driver generates is in the form of $(PID_j, T_j, C_j, \mathfrak{C}_j, \sigma_j)$, which has a size of $|Msg_j| = |PID_j| + |T_j| + 160 * 2 + 160 = 0.071$KB if we set $|PID_j| + |T_j| = 100$ bits and $|G| = 160$ bits. For the crossroad, the form of a report is $(PID_j, T_j, \{C_i^j, \mathfrak{C}_i^j\}_{i=1}^{M}, \sigma_j)$ and the data size is 0.149KB. The RSU collects all N drivers' reports and the total size of driver-to-RSU communication should be $N * 0.071$KB and $N * 0.149$KB, respectively. We then consider the RSU-to-TMC communication. For both T-junction and crossroad, the message transmitted from RSU to TMC is in a form of $Msg_l = (ID_l, T_l, C_l, \sigma_l)$ and its seize is $|Msg_l| = |ID_l| + |T_l| + 160 + 160 = 0.051$ KB. Since we use a well-designed super-increasing sequence to aggregate the ciphertexts, the data size will not be increased with the direction number. We plot the communication overhead of VPTS and the traditional scheme by varying the number of drivers. As shown in Fig. 9 and Fig. 10, less communication overhead is introduced in VPTS in both driver-to-RSU communication and RSU-to-TMC communication.

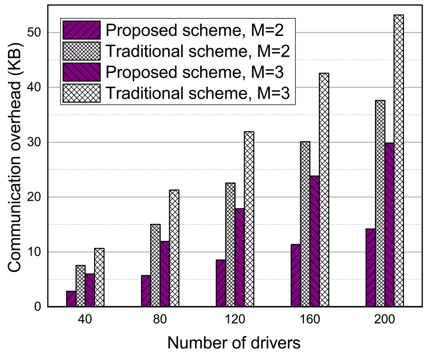


Fig. 9. Communication overhead on driver-to-RSU communication.

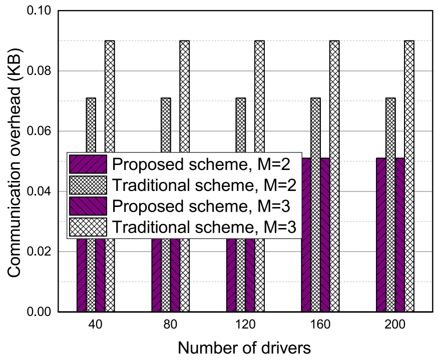


Fig. 10. Communication overhead on RSU-to-TMC communication.

VI. Related Works

In this section, we briefly review the existing works related to privacy-preserving crowdsourcing-based traffic management.

Alleviating traffic congestion is one of the world's greatest challenges and has received most attention from both academia and industry. Some traffic-related apps, such as Google Maps [7] and WAZA [8] have been developed and put into use. By collecting drivers' real-time traffic data, these apps can help drivers to know real-time traffic conditions and select optimal directions before their driving. However, further progress in these traffic-related apps may be impeded if drivers' private information cannot be well protected, especially considering that GDPR has raised strict legal requirements on personal privacy. To achieve privacy-preserving traffic monitoring, several solutions have been proposed. For example, Lu *et al.* [31] proposed an efficient and privacy-preserving traffic monitoring scheme, which not only achieves identity tracing but also supports local revocation verification. Based on temporal and spatial entropies of traffic samples, Du *et al.* [9] presented a new approach to realize efficient urban traffic monitoring. Ni *et al.* [11] presented a private vehicular crowdsourcing-based navigation scheme. By using the randomizable signature [32], their scheme can collect drivers' speed without disclosing drivers' identity information. Based on [11], Li *et al.* found that drivers may deceive the node by colluding with each other and they presented an effective solution to defend against this attack. Liu *et al.* [14] presented a fog-assisted intelligent traffic control scheme with drivers' privacy preservation. Although the above schemes can realize secure and private traffic monitoring, most of them rely on current real-time traffic data and they cannot predict future traffic prediction. Collecting drivers' directions is an effective way to know the future traffic conditions. Therefore, some recent researches focus on future traffic flows statistics, such as [10], [12], [33]–[36]. Among them, the works [10], [12], [36] are designed for privacy-preserving traffic flows statistics. Specifically, Florian *et al.* [10] presented to collect drivers' current and planned future locations in a privacy-preserving way. Rabieh *et al.* [12] further proposed two privacy-preserving route collection solutions for infrastructure-based vehicular ad hoc networks and self-organizing vehicular ad hoc networks. However, existing techniques in privacy-preserving traffic flow statistics cannot be directly applied in our scenario, since they cannot deal with the statistical and data pollution attacks.

Another related work is the secure data aggregation scheme using super-increasing sequences. For example, Xu *et al.* [37] proposed to use a super-increasing sequence to aggregate multidimensional sensory data to save communication overhead. Xue *et al.* [38] proposed to use a well-defined super-increasing sequence to aggregate multi-set data to improve computation efficiency and reduce communication overhead. Similar secure data aggregation schemes have also been presented for applications such as disease risk prediction [39] and smart grid [40].

VII. Conclusion

In this paper, we have considered crowdsourcing-based traffic flow statistics at road intersections and proposed a novel verifiable and privacy-preserving traffic flow statistics (VPTS) scheme for advanced traffic management systems. VPTS can provide strong protection of drivers privacy, verify the correctness of drivers data, and guarantee high efficiency for the traffic management system. We have also provided detailed analysis and extensive simulations to demonstrate its correctness, security, and efficiency. For the future work, we will investigate the fairness of the crowdsourcing-based traffic monitoring scenario, by applying the blockchain technology.

References

[1] L. Figueiredo, I. Jesus, J. T. Machado, J. R. Ferreira, and J. M. De Carvalho, "Towards the development of intelligent transportation systems," in *Proc. IEEE Intell. Transp. Syst.*, 2001, pp. 1206–1211.

[2] K. Zhang, J. Ni, K. Yang, X. Liang, J. Ren, and X. Shen, "Security and privacy in smart city applications: Challenges and solutions," *IEEE Commun. Mag.*, vol. 55, no. 1, pp. 122–129, Jan. 2017.

[3] N. H. C. Yung and A. H. Lai, "An effective video analysis method for detecting red light runners," *IEEE Trans. Veh. Technol.*, vol. 50, no. 4, pp. 1074–1084, Jul. 2001.

[4] E. Ua-areemitr, A. Sumalee, and W. H. Lam, "Low-cost road traffic state estimation system using time-spatial image processing," *IEEE Intell. Transp. Syst. Mag.*, vol. 11, no. 3, pp. 69–79, Jun. 2019.

[5] S. Garg, A. Singh, S. Batra, N. Kumar, and L. T. Yang, "UAV-empowered edge computing environment for cyber-threat detection in smart vehicles," *IEEE Netw.*, vol. 32, no. 3, pp. 42–51, May\Jun. 2018.

[6] N. Cheng *et al.*, "Big data driven vehicular networks," *IEEE Netw.*, vol. 32, no. 6, pp. 160–167, Nov.\Dec. 2018.

[7] Accessed: April 25, 2018. [Online]. Available: https://googleblog.blogspot.ca/2009/08/bright-side-of-sitting-intraffic.html

[8] Waze. Accessed: Jan. 2016. [Online]. Available: https://www.waze.com/

[9] 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.

[10] M. Florian, S. Finster, and I. Baumgart, "Privacy-preserving cooperative route planning," *IEEE Internet Things J.*, vol. 1, no. 6, pp. 590–599, Dec. 2014.

[11] J. Ni, X. Lin, K. Zhang, and X. Shen, "Privacy-preserving real-time navigation system using vehicular crowdsourcing," in *VTC Fall*, pp. 1–5, 2016.

[12] K. Rabieh, M. M. E. A. Mahmoud, and M. F. Younis, "Privacy-preserving route reporting schemes for traffic management systems," *IEEE Trans. Veh. Technol.*, vol. 66, no. 3, pp. 2703–2713, Mar. 2017.

[13] M. Li, L. Zhu, and X. Lin, "Privacy-preserving traffic monitoring with false report filtering via fog-assisted vehicular crowdsensing," *IEEE Trans. Services Comput.*, to be published, doi: 10.1109/TSC.2019.2903060.

[14] J. Liu *et al.*, "Secure intelligent traffic light control using fog computing," *Future Gener. Comput. Syst.*, vol. 78, pp. 817–824, 2018.

[15] S. Chakravarty, How GDPR will impact location data, Accessed on: May 23, 2018. [Online]. Available: https://www.geospatialworld.net/article/how-gdpr-impacts-location-data/

[16] H. Duan, Y. Zheng, Y. Du, A. Zhou, C. Wang, and M. H. Au, "Aggregating crowd wisdom via blockchain: A private, correct, and robust realization," in *Proc. PerCom*, 2019, pp. 43–52.

[17] G. Wang, B. Wang, T. Wang, A. Nika, H. Zheng, and B. Y. Zhao, "Defending against sybil devices in crowdsourced mapping services," in *Proc. 14th Annu. Int. Conf. Mobile Syst., Appl. Services*, 2016, pp. 179–191.

[18] J. Lin, D. Yang, K. Wu, J. Tang, and G. Xue, "A sybil-resistant truth discovery framework for mobile crowdsensing," in *Proc. ICDCS*, 2019, pp. 871–880.

[19] D. Boneh, E.-J. Goh, and K. Nissim, "Evaluating 2-dnf formulas on ciphertexts," in *Proc. Theory Cryptography Conf.*, 2005, pp. 325–341, Springer, 2005.

[20] N.-W. Lo and J.-L. Tsai, "An efficient conditional privacy-preserving authentication scheme for vehicular sensor networks without pairings," *IEEE Trans. Intell. Transp. Syst.*, vol. 17, no. 5, pp. 1319–1328, May 2016.

[21] D. Boneh and M. Franklin, "Identity-based encryption from the weil pairing," in *Proc. Annu. Int. Cryptology Conf.*, 2001, pp. 213–229.

[22] J. Xu, K. Xue, Q. Yang, and P. Hong, "PSAP: Pseudonym-based secure authentication protocol for NFC applications," *IEEE Trans. Consum. Electron.*, vol. 64, no. 1, pp. 83–91, Feb. 2018.

[23] D. Boneh, B. Lynn, and H. Shacham, "Short signatures from the weil pairing," *J. Cryptology*, vol. 17, no. 4, pp. 297–319, 2004.

[24] A. Ghosh, T. Roughgarden, and M. Sundararajan, "Universally utility-maximizing privacy mechanisms," *SIAM J. Comput.*, vol. 41, no. 6, pp. 1673–1693, 2012.

[25] R. Lu, K. Heung, A. H. Lashkari, and A. A. Ghorbani, "A lightweight privacy-preserving data aggregation scheme for fog computing-enhanced iot," *IEEE Access*, vol. 5, pp. 3302–3312, 2017.

[26] J. Groth, R. Ostrovsky, and A. Sahai, "Perfect non-interactive zero knowledge for np," in *Proc. Annu. Int. Conf. Theory Appl. Cryptographic Techn.*, 2006, pp. 339–358.

[27] E. Ben-Sasson, A. Chiesa, E. Tromer, and M. Virza, "Succinct non-interactive zero knowledge for a von neumann architecture," in *Proc. USENIX*, 2014, pp. 781–796.

[28] M. Bellare and P. Rogaway, "Random oracles are practical: A paradigm for designing efficient protocols," in *ACM, CCS*, 1993, pp. 62–73.

[29] C. Dwork, "Differential privacy: A survey of results," in *Proc. Int. Conf. Theory Appl. Models Comput.*, 2008, pp. 1–19, Springer.

[30] H. Peng, L. Liang, X. Shen, and G. Y. Li, "Vehicular communications: A network layer perspective," *IEEE Trans. Veh. Technol.*, vol. 68, no. 2, pp. 1064–1078, Feb. 2019.

[31] R. Lu, X. Lin, Z. Shi, and X. Shen, "A lightweight conditional privacy-preservation protocol for vehicular traffic-monitoring systems," *IEEE Intell. Syst.*, vol. 28, no. 3, pp. 62–65, May\Jun. 2013.

[32] D. Pointcheval and O. Sanders, "Short randomizable signatures," in *Proc. Cryptographers Track RSA Conf.*, 2016, pp. 111–126, Springer.

[33] M. Wang, H. Shan, R. Lu, R. Zhang, X. Shen, and F. Bai, "Real-time path planning based on hybrid-VANET-enhanced transportation system," *IEEE Trans. Veh. Technol.*, vol. 64, no. 5, pp. 1664–1678, May 2015.

[34] J. Lin, W. Yu, X. Yang, Q. Yang, X. Fu, and W. Zhao, "A real-time en-route route guidance decision scheme for transportation-based cyberphysical systems," *IEEE Trans. Veh. Technol.*, vol. 66, no. 3, pp. 2551–2566, Mar. 2017.

[35] M. Hajiahmadi, V. L. Knoop, B. De Schutter, and H. Hellendoorn, "Optimal dynamic route guidance: A model predictive approach using the macroscopic fundamental diagram," in *Proc. IEEE, ITSC*, 2013, pp. 1022–1028.

[36] Y. Zhang, Q. Pei, F. Dai, and L. Zhang, "Efficient secure and privacy-preserving route reporting scheme for vanets," in *Proc. J. Phys.: Conf. Ser.*, 2017, vol. 910, p. 012070.

[37] G. Xu, H. Li, C. Tan, D. Liu, Y. Dai, and K. Yang, "Achieving efficient and privacy-preserving truth discovery in crowd sensing systems," *Comput. Secur.*, vol. 69, pp. 114–126, 2017.

[38] K. Xue, B. Zhu, Q. Yang, D. S. L. Wei, and M. Guizani, "An efficient and robust data aggregation scheme without a trusted authority for smart grid," *IEEE Internet Things J.*, vol. 7, no. 3, pp. 1949–1959, Mar. 2020.

[39] X. Yang, R. Lu, J. Shao, X. Tang, and H. Yang, "An efficient and privacy-preserving disease risk prediction scheme for e-healthcare," *IEEE Internet Things J.*, vol. 6, no. 2, pp. 3284–3297, Apr. 2019.

[40] S. Li, K. Xue, Q. Yang, and P. Hong, "PPMA: Privacy-preserving multi-subset data aggregation in smart grid," *IEEE Trans. Ind. Inform.*, vol. 14, no. 2, pp. 462–471, Feb. 2018.