# Blockchain Enabled Trust-Based Location Privacy Protection Scheme in VANET

Bin Luo, Xinghua Li, *Member, IEEE*, Jian Weng, *Member, IEEE*, Jingjing Guo,
and Jianfeng Ma, *Member, IEEE*

***Abstract*—While enjoying the convenience brought by Location Based Service (LBS), the location privacy of vehicles in VANET may be disclosed. Distributed *k*-anonymity, as one of the most popular privacy protection methods, fails to take the trustworthiness of participants into account, resulting in malicious tracing of vehicles, which further leads to the sensitive information leakage, and even the safety threat of personal property. To address this issue, we propose a blockchain enabled trust-based location privacy protection scheme in VANET. Specifically, by analyzing the different requirements of the request vehicle and the cooperative vehicle during the process of constructing the anonymous cloaking region, as well as combining the characteristics of these two roles, we devise the trust management method based on *Dirichlet distribution*, such that both the requester and the cooperator will only cooperate with the vehicles they trust. Moreover, by employing blockchain, we also proposed the data structure to record the trustworthiness of vehicles on publicly available blocks timely, so that any vehicle can access the historical trust information of counterparties whenever necessary. Finally, the construction process of anonymous cloaking region is presented. Security analysis and extensive experiments indicate that the proposal is resilient to various trust model attacks, it can effectively detect the malicious vehicles, and preserve the location privacy of vehicles in the anonymous cloaking region construction, while the required time delay is limited.**

***Index Terms*—LBS, distributed *k*-anonymity, trust mechanism, blockchain, VANET.**

## I. Introduction

VEHICULAR Ad-hoc Network (VANET) is a new kind of Ad-hoc wireless network, which is formed by vehicle-to-vehicle, i.e., V2V communications and vehicle-to-roadside unit (RSU), i.e., V2R communications. It not only facilitates the establishment of a secure and reliable transportation system, but also provides an information-rich driving environment based on massive information sharing and corresponding decision making. Recently, as an essential part of Intelligent Transportation Systems (ITSs), it has attracted wide attentions of research institutes and scholars [1], [2].

With the rapid development of wireless communication, positioning technology, and smart mobile terminals, Location Based Service (LBS), which is on the basis of Geographic Information System (GIS) platform, enables vehicles to access various information related to spatial locations at any time, such as nearby gas stations, rest areas and so on. However, during the process of LBS, Location Based Service Provider (LSP) may collect and abuse location information submitted by vehicles, to further infer a large amount of sensitive information, including the drivers' home addresses, hobbies, religious faiths, and political tendencies [3]. Therefore, while enjoying the convenience brought by LBS, the location privacy leakage cannot be ignored [4].

As one of the most popular location privacy protection methods, *k*-anonymity can not only provide precise query results without any necessity for key sharing while guaranteeing the personalized privacy protection requirement, but also have low computational overhead. In particular, by utilizing the anonymous cloaking region that involved at least *k* participants instead of the requester's real location, it makes that any adversary cannot associate the query with a specific participant with a confidence greater than $\frac{1}{k}$. Existing *k*-anonymity privacy protection schemes are mainly divided into two categories: centralized and distributed, where the former requires an anonymous cloaking server, and the latter achieves anonymity by the union of participants themselves. Specifically, when a participant initiates the LBS query in distributed *k*-anonymity, he will construct the anonymous cloaking region with the help of at least $k-1$ participants nearby. Since it overcomes the performance bottleneck in centralized situation, distributed *k*-anonymity has become a hot research field in recent years [5]–[10]. Nevertheless, the existing works usually imply that there exist trust relationships among all participants, since the failure of taking the trustworthiness of them into account, if the works are applied to VANET directly, the following consequences will appear:

1) If the request vehicle that initiates the LBS query is malicious, it may disclose the received locations from cooperators to obtain the additional benefits;
2) If the cooperative vehicle is malicious, out of unwillingness to expose its location in a sensitive area or colluding with LSP to get extra benefits, it may provide an inappropriate location resulting in a disturbed anonymous cloaking region, which not only affects the query results seriously, but also leads that LSP can effectively shrink the anonymous cloaking region, as shown in Fig. 1, and even infer the request vehicle's real location in extreme cases.

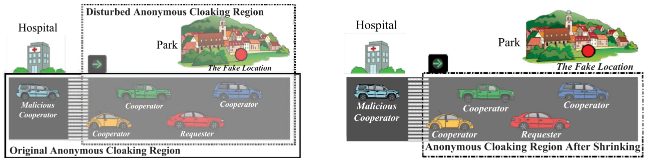
It is obvious that the aforementioned situations will cause malicious tracing of vehicles, enabling adversaries to infer the

Manuscript received March 13, 2019; revised July 5, 2019 and October 9, 2019; accepted November 28, 2019. Date of publication December 6, 2019; date of current version February 12, 2020. This work was supported by National Natural Science Foundation of China under Grants U1708262, U1736203, 61772173, and 61672413. The review of this article was coordinated by Dr. S. Misra. *(Corresponding author: Xinghua Li.)*

B. Luo, X. Li, J. Guo, and J. Ma are with the School of Cyber Engineering, Xidian University, Xi'an 710126, China (e-mail: luobin2127@hotmail.com; xhli1@mail.xidian.edu.cn; jjguo@xidian.edu.cn; jfma@mail.xidian.edu.cn).

J. Weng is with the College of Information Science and Technology, Jinan University, Guangzhou 510632, China (e-mail: cryptjweng@gmail.com).

Digital Object Identifier 10.1109/TVT.2019.2957744



(a) (b)

Fig. 1. The location privacy leakage caused by the malicious cooperative vehicle. *In this example, the malicious cooperative vehicle attempts to conceal its real location near the hospital, and provides a fake one in the park, thus, the original anonymous cloaking region is disturbed, and LSP will retrieve the querying results according to the disturbed region instead of the original one, which has an impact on the querying results. Furthermore, LSP is able to shrink the region from (a) to (b) because it is unusual for a vehicle to locate in a park.* (a) Before shrinking the anonymous cloaking region. (b) After shrinking the anonymous cloaking region.

sensitive information such as driving habits, and further threaten the safety of personal property.

To solve these problems, in this paper, we take the trustworthiness of vehicles into account during the process of constructing the anonymous cloaking region. Specifically, the malicious behaviors of the vehicle will lead to the reduction of the trust level, and once its trust level is lower than the defined threshold, it will be identified as a malicious one. On the premise of meeting the self-containment nature, our scheme makes that both the request vehicle and the cooperative vehicle will only cooperate with the ones they trust. Although there have been plenty of trust management models in VANET [11]–[23], they mainly focus on the trustworthiness of vehicles' behaviors during data transmission or the reliability of received events. Due to the different objectives, the factors that taken into account in these models do not adapt to the specific features of distributed anonymous cloaking region construction. Moreover, for the high mobility, a vehicle may move in wide geography scope and initiate LBS queries. It is difficult to establish a trusted server to completely store the historical trust information of all vehicles from a wide geography scope because of performance bottlenecks. Therefore, we aim to establish a distributed trust management system, making that vehicles can effectively obtain the trustworthiness of counterparties by storing their trust information in a secure and trusted database that can be shared globally.

To address the issues above, we first devise the trust management method for vehicles. Then, to build the desired distributed trust management system, blockchain technology, which has the characteristics such as data consistency, tamper-resistant and so on, is adopted to record vehicles' historical trust information. The detailed process of constructing the anonymous cloaking region is further presented. The contributions of this paper are summarized as follows:

1) During the process of constructing the anonymous cloaking region, by analyzing the factors reflecting the trust degree of the vehicle acts as a requester or a cooperator, as well as taking its historical trust information into account, we give the method to calculate the trust level of the vehicle. Moreover, to update the vehicle's historical trust information, the specific approach is also developed.
2) We establish the data structure to record the historical trust information of the vehicle on the blockchain in time. Furthermore, by introducing the proposed trust management method into distributed $k$-anonymity, the trust-based anonymous cloaking region construction procedure is presented.
3) Security analysis and extensive experiments indicate that our scheme is resilient to various trust model attacks (such as *bad-mouthing attack*, *on-off attack*), it can detect malicious vehicles and preserve vehicles' location privacy effectively during the anonymous cloaking region construction process. In addition, very limited time delay is required, making it feasible to be implemented in VANET.

The remainder of this paper is organized as follows. In Section II, we review the existing works related to distributed $k$-anonymity and trust management models in VANET. Section III presents the system architecture and preliminaries. The detailed scheme is proposed in Section IV, including the specific trust management method and the steps of constructing the anonymous cloaking region. Section V and Section VI give the theoretical analysis and experimental results respectively. Finally, we conclude the proposal in Section VII.

## II. Related Work

### A. Distributed k-Anonymity Schemes

The first distributed $k$-anonymity scheme was proposed by Chow *et al.* [5]. In their scheme, by means of point-to-point communication, the requester is advised to construct the anonymous cloaking region with at least $k-1$ nearby users. Later, Ghinita *et al.* [6] proposed MobiHide, which used a Hilbert Curve to map locations in 2-D space into 1-D values. When a user initiates the LBS query, including himself, $k$ sequential users in 1-D space will be selected randomly to achieve anonymity. To make the $k$ selected locations non-homogeneous, Sun *et al.* [7] introduced location labels, which distinguish between sensitive locations and ordinary ones of mobile users. Li *et al.* [8] proposed a credit based $k$-anonymity scheme to encourage users to participate in the construction of anonymous cloaking region. However, the works above require the requester to wait passively once he has not received the locations of at least $k-1$ cooperators. To address this issue, Zhong *et al.* [9] proposed a regional perception based location privacy protection scheme with the help of the mobile service provider. Recently, Ghaffari *et al.* [10] proposed a peer-to-peer privacy preserving query service scheme named $P^4QS$. In the proposal, a client will send the query content and its location to the anonymizer in the same region. After collecting $k$ queries, the anonymizer will construct the anonymous cloaking region.

However, the existing works fail to take the trustworthiness of participants into account while constructing the anonymous cloaking region. If they are applied to VANET directly, it will enable adversaries to track the request vehicle or the cooperative one, so as to infer the sensitive information, and even threat the safety of personal property.

### B. Trust Management Models in VANET

In VANET, researchers have been developing various trust management models which can be divided into three categories:

entity-oriented trust model, data-oriented trust model, and combined trust model [11].

- **Entity-oriented trust model.** Entity-oriented trust model focuses on predicting the probability of vehicles behave honestly based on historical experiences, it only concerns the vehicles themselves without any attention about the trustworthiness of the transmitted content or reported data.

To realize the communication between VANET and other infrastructure networks, Sharef *et al.* [12] explored how to select an appropriate gateway. They combined direct trust and indirect trust together, so as to measure the reliability of the communication path between nodes. By measuring the context-based parameters like message sending frequency, transmission channel features, Kerrache *et al.* [13] made it possible to select trusted vehicles for data transmission. In the scheme proposed by Minhas *et al.* [14], based on the combination of role-based trust, experience-based trust, and majority-based trust, the trustworthiness of vehicles are modeled.

In the course of driving, vehicles constantly receive beacon messages from nearby neighbors and make driving decisions accordingly, thus, it is crucial to guarantee the reliability of received messages. Li *et al.* [15] indicated the reliability of a message is based on the reputation of the vehicle reporting it. They introduced a reputation server in the proposed scheme. After receiving a message, it will be determined whether the source vehicle has the valid reputation certificate issued by the server. Similarly, by means of reputation certificate, Lu *et al.* [16] proposed a blockchain-based anonymous reputation system to prevent internal vehicles from broadcasting bogus messages. Yang *et al.* [17] also utilized blockchain technology. In their scheme, the reputation of vehicles is directly recorded on blocks, which are generated and maintained by adjacent vehicles of the evaluated one.

- **Data-oriented trust model.** Data-oriented trust model handles the messages or events released by vehicles, which emphasizes the trustworthiness of data itself.

While evaluating the reliability of beacon messages, Raya *et al.* [18] pointed out that entity-oriented trust model is not appropriate. In their scheme, a framework for data-oriented trust model is proposed. First, they measured the trustworthiness of messages provided by each vehicle. Then, using logical decision-making methods, the opinion of an event included in the messages derived from multiple vehicles is evaluated comprehensively. Golle *et al.* [19] assumed that each vehicle has a model of all knowledge about VANET. After receiving a message, it is evaluated against the vehicle's model. Recently, Gurung *et al.* [20] proposed a trust model named RMCV, which includes several factors such as content similarity, content conflict, and message routing path similarity. In the model, the messages related to the same event will first be classified using clustering algorithms, then the trustworthiness of the messages in each class is evaluated.

- **Combined trust model.** Combined trust model takes both entity-oriented trust model and data-oriented trust model into account. It has been proposed to model the trustworthiness of vehicles and utilize the modeling results to evaluate the reliability of received messages.

Based on opinion piggybacking, Dőtzer *et al.* [21] proposed a reputation system in which each forwarding peer appends its opinion about the trustworthiness of the event to the message. It is obvious that the former appended opinion will affect the latter ones. Patwardhan *et al.* [22] proposed an approach where a few nodes, named as anchor ones, are assumed to be pre-authenticated. By communication with them, it is enable to carry out message validation. However, the scheme only works well while there exist a sufficient number of reports related to the same event. Furthermore, the authors in [23] estimated the trustworthiness of the misbehavior detection nodes in VANET based on $k$-means consensus clustering algorithm.

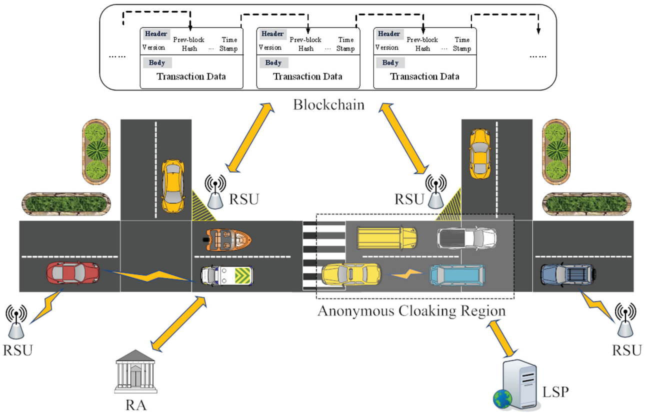


Fig. 2. System architecture.

Due to the high mobility of vehicles, their surrounding environments may be either familiar or strange when they initiate LBS queries. It is obvious that in the former, the vehicle can depend on the received messages to determine the counterparty's trustworthiness, while in the latter, it mainly refers to the reliability of the target vehicle itself. Thus, neither data-oriented trust model nor entity-oriented trust model can meet our requirement, which implies that combined trust model should be adopted. However, the existing works mainly focus on the vehicle's trustworthiness during data transmission or the reliability of reported events, where the factors considered in trust evaluation are not suitable to distributed $k$-anonymity. Moreover, to achieve trust management, the works above often depend on a third party [15], [16] or evaluate the trustworthiness of the received messages via direct experiences and other vehicles' recommendations [12]–[14], [18], [19], [21], [22]. Nevertheless, the third party may cause a single point of failure, meanwhile, inadequate direct experiences and recommendations will cause inaccurate trust evaluation result. Although there are some schemes that have introduced blockchain to record the trustworthiness of vehicles, they still rely on a third management center [16], or cannot create and maintain the blockchain feasibly because of vehicles' high mobility [17]. Therefore, the existing works are inapplicable to distributed $k$-anonymity in VANET.

## III. System Architecture and Preliminaries

### A. System Architecture

The system architecture of our scheme is depicted in Fig. 2, which consists of four parts, registration authority (RA), RSUs, vehicles and LSP.

In distributed $k$-anonymity, the identity privacy and the trust evaluation of vehicles are contradictory. The leakage of identity information will lead to the threat to vehicles' privacy. For example, once the identity of the cooperative vehicle is disclosed, based on the received locations, the requester may obtain its trajectories; and once the request vehicle's identity is leaked, its behavior pattern will be inferred by means of the methods such as intersection attack. To solve this problem, our scheme introduces pseudonym change method. After participating in the construction of anonymous cloaking region for $x$ times, the vehicle will change its pseudonym, the value of $x$ is presented in security analysis.

*Registration Authority (RA):* When a vehicle enters the LBS system, it will first generate the public-private key that is used during the process of constructing the anonymous cloaking region. Then, the vehicle will register with RA and send its public key through secure communication channels, which can be achieved by the existing authentication and key agreement protocols, such as [24]. As a trusted third party in VANET, RA can not only take the responsibility of assigning the vehicle's pseudonym and initial trust level, which will be delivered to RSUs in the form of the transaction bill to be recorded on the blockchain, but also act as a Certification Authority (CA) to create the public key certificate. Obviously, the certificate includes the assigned vehicle's pseudonym, the public key, and will be returned to the vehicle itself. In the subsequent construction process, if the vehicle's pseudonym is approaching expiration, it will initiate an update request including its pseudonym, the transaction index number that records its latest historical trust information, and the new generated public key to RA. After receiving and verifying the message, RA will assign a new pseudonym for the vehicle and create the new public key certificate. Moreover, the new pseudonym and the vehicle's latest historical trust information obtained by RA will again be delivered to RSUs with the form of the transaction bill and be recorded on the blockchain, which completes the update of the pseudonym.

*Roadside Units (RSUs):* RSUs are connected through wireless channels to vehicles. During the process of constructing the anonymous cloaking region, on one hand, the vehicles will query the counterparties' historical trust information from RSUs. On the other hand, RSUs are in charge of creating and maintaining the entire blockchain. We assume that RSUs are widely deployed on both sides of the roads, which not only ensures that vehicles can always obtain the required data but also makes it possible to record the new trust information to the block in time.

*Vehicles:* When a request vehicle initiates the LBS query, it first broadcasts the cooperation request. By means of calculating the trust level, both the request vehicle and the cooperative one will only select trusted counterparties to cooperate. After receiving the locations derived from at least $k-1$ cooperative vehicles, the requester constructs the anonymous cloaking region that contains its real location and submits the region along with the query content to LSP.

*Location-Based Service Provider (LSP):* After receiving the anonymous cloaking region and the query content, LSP retrieves the relevant information from the database, and further returns the results.

### B. Blockchain

As a new type of distributed infrastructure and computing method, blockchain, whose technology originated from Bitcoin, has been applied in various domains [3], [25]. Essentially, it is a public database that combines data blocks in a sequential manner according to time sequence. Each data block contains block head and block body, where head encapsulates the information such as current version number, previous block address, timestamp, etc., and body mainly records the detailed transactions, as shown in Fig. 3.

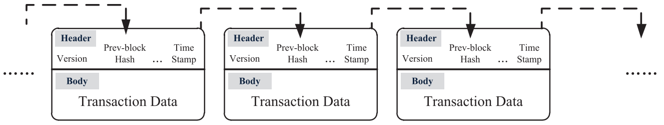


Fig. 3. Blockchain.

A transaction in blockchain can be regarded as an interaction between two or more participants, which will be recorded permanently. The complete life cycle of a transaction includes transaction generation, transaction propagation, block creation, block validation and blockchain update. During this process, the consensus algorithms are adopted to ensure the consistency of the recorded information.

In our scheme, we regard the construction of the anonymous cloaking region as a transaction and encourage the request vehicle and the cooperative ones to behave honestly by recording their historical trust information on the blockchain. Due to the facts that 1) in VANET, vehicles are required to transmit and handle a large amount of data while moving in high-speed, such as road condition monitoring, accident reporting and so on; 2) all transactions are required to be stored on the blockchain, and the record of a new block needs to run consensus algorithms to reach an agreement, it will become a huge burden for vehicles to update, maintain and store the blockchain directly. As shown in Fig. 2, we utilize RSUs to address this problem. Because the number of RSUs is much less than that of vehicles, and generally, honest RSUs are more than malicious ones, it will be great progress on performance.

There are three kinds of blockchain in existing works: public blockchain, consortium blockchain, and private blockchain. Since in the proposal, only RSUs are allowed to record transactions on the blockchain, and vehicles query the corresponding data merely, consortium blockchain is adopted. Compared with the other two categories, it has the characteristics of limited access, high efficiency, and good extendibility.

### C. Trust Calculation Method

According to the specific requirements, the trust calculation method based on *Dirichlet distribution* [26] is able to classify participants' behaviors into several ranks and regard the trust value as the probability that the behavior is at a specified rank. In contrast with the other trust calculation methods, such as *EnigenTrust* [27] and *Beta theory* based trust calculation method [28], the trust calculation method based on *Dirichlet distribution* enables finer-grained calculation.

*Dirichlet distribution: Dirichlet distribution* is a joint probability distribution of multiple random variables defined in the interval of [0,1]. Suppose there are $n$ random variables $\sigma_i$, which are denoted as $\overrightarrow{\sigma} = (\sigma_1, \sigma_2, \ldots, \sigma_n)$. $p(\sigma_i)$ is the probability distribution that $\sigma_i$ occurs. It is established that $p(\sigma_i) > 0$ and $\sum_{i=1}^{n} p(\sigma_i) = 1, \overrightarrow{p_\sigma} = (p(\sigma_1), p(\sigma_2), \ldots, p(\sigma_n))$. Under $p(\sigma_i)$, each $\sigma_i$ has a parameter $\alpha_i$ that can be regarded as the priori observation counts, $\overrightarrow{\alpha} = (\alpha_1, \alpha_2, \ldots, \alpha_n)$. Then the *probability density function of Dirichlet distribution* is:

$$f(\overrightarrow{p_\sigma}|\overrightarrow{\alpha}) = Dir(\overrightarrow{p_\sigma}|\overrightarrow{\alpha}) = \frac{\Gamma(\sum_{i=1}^{n} \alpha_i)}{\prod_{i=1}^{n} \Gamma(\alpha_i)} \prod_{i=1}^{n} p(\sigma_i)^{\alpha_i - 1},$$

where $\Gamma(x) = \int_0^\infty t^{x-1} e^{-t} dt$ is *Gamma function*.

As a common Bayesian inference method, *Dirichlet distribution* has the characteristic that its prior distribution and posterior distribution are conjugate. We adopt it in the trust level calculation process to predict the vehicle's behaviors.

## IV. Our Proposed Scheme

### A. Trust Management Method

To effectively identify the malicious vehicles during the process of constructing the anonymous cloaking region, in the proposed scheme, we evaluate the trustworthiness of vehicles. At this stage, a vehicle will intuitionally evaluate the counterparty's current behaviors based on its own knowledge, facing risks, and so on. However, due to the limitation of vehicle's self-knowledge and the obtained context information, the evaluation may be biased and relatively subjective. Therefore, besides the current behaviors, we also take the historical trust information of the counterparty recorded on the blockchain into account.

Specifically, the vehicle will first evaluate the counterparty's current behaviors, then, combining with the recorded historical trust information on the blockchain, which is maintained by all RSUs, the counterparty's trust level under current transaction is calculated. Obviously, the trust level is a reflection of the long-term behaviors of the vehicle. Consequently, it is adopted to predict the vehicle's behaviors, and further judge whether the vehicle is honest or malicious. In a specific transaction, once the vehicle's trust level is lower than a certain degree, it will be identified as a malicious one. To prevent the vehicle from increasing its trust level rapidly by means of recent honest behaviors, or maintaining a high degree of trust level through *on-off attack*, while calculating the trust level, we make that the historical trust information and the current behaviors evaluation have the same weight.

In this subsection, we first develop the means of evaluating the vehicle's current behaviors, then, along with the historical trust information, the calculation method of the trust level is given. Finally, we also explore the approach for updating the historical trust information recorded on the blockchain. Under the current transaction, symbol $v_a$ is used to represent the vehicle that acts as the requester and $v_b$ denotes the vehicle acting as the cooperator.

*1) Evaluation of Vehicle's Current Behaviors:* We respectively develop the means of evaluating the vehicle's current behaviors performed by $v_a$ to $v_b$, and $v_b$ to $v_a$, where the corresponding results are represented as $E_{rv_a,cv_b}$ and $E_{cv_b,rv_a}$. Without loss of generality, the range of evaluation result is defined as $[0, 1]$, where 1 means complete trust, 0 for distrust and 0.5 is neutrality.

- ***Evaluation of request vehicle's current behaviors***

It has been known that if $v_a$ is malicious, its purpose is to obtain and leak the locations of surrounding cooperative vehicles as much as possible, thus, comparing with an honest request vehicle, its querying behaviors will be different. Under a specific transaction, with the experiences and the help of RSUs, $v_b$ will mainly evaluate $v_a$'s reasonability to initiate a LBS query from two dimensions of space and time.

*Querying reasonability from space:* In the existing works, Niu *et al.* [29] pointed out that the probability of users initiating LBS queries varies with their locations. In their scheme, they divided the entire location map into $n \times n$ cells with equal size. According to the previous query history, each cell is defined to have a query probability that can be accessed by all users. In VANET, when vehicles communicate with each other, their locations are generally close. Therefore, according to its location information, $v_b$ can determine the probability of $v_a$ to initiate the LBS query, and further evaluate the querying reasonability from space. Using $p_q$ to denote the probability that $v_a$ initiates a query, it can be judged by the experiences of $v_b$ itself, or be estimated by the neighboring RSU through statistical analysis of all LBS queries initiated in its coverage. We define the querying reasonability from space as $r_s = f(p_q)$, which is divided into $\ell$ incremental levels $\{r_s^1, r_s^2, \ldots, r_s^\ell\}$ based on the probabilities $\{p_q^1, p_q^2, \ldots, p_q^\ell\}$, $1 \leq i \leq \ell$. In our scheme, we take the value of $r_s^i$ as follows as an example:

$$r_s^i = f(p_q^i) = \frac{1}{\ell - 1}(i - 1).$$

*Querying reasonability from time:* The researchers in [30] and [31] indicated that the query behaviors of users in LBS have certain regularity. In VANET, by counting the cooperation requests broadcasted by request vehicles, a RSU can learn the historical querying habits in its coverage area, which is enabled to be utilized by the cooperator. Specifically, from a time point of view, we divide a day into several time internal $\{I_1, I_2, \ldots, I_l\}$. Assuming $\tilde{q}$ and $q^i$ are the number of times that vehicles initiate LBS queries in a day and in $I_i$ on average, $1 \leq i \leq l$. Then, $p(q^i) = \frac{q^i}{\tilde{q}+\mu_1}$ is the probability that a LBS query is raised in $I_i$, where $\mu_1$ is the tuning parameter to restrain the influence of too small $\tilde{q}$. We use $r_t^i$ to represent the querying reasonability in $I_i$, it has:

$$r_t^i = \frac{1}{1 + e^{-\delta(p(q^i) - \mu_2)}},$$

where $\delta$ is the regulatory parameter, $\mu_2$ controls the partition of the reasonability, for example, $\mu_2 = \frac{1}{4}$ means that $v_a$'s LBS query is regarded reasonable while $p(q^i) \geq \frac{1}{4}$, which equals to $r_d^i \geq 0.5$.

Up to now, the evaluation of $v_a$'s current behaviors performed by $v_b$ can be obtained as follows, where $\eta \in [0, 1]$ balances the querying reasonability from space and the reasonability from time:

$$E_{cv_b,rv_a} = \eta r_s^i + (1 - \eta) r_t^i.$$

- ***Evaluation of cooperative vehicle's current behaviors***

During the construction process, $v_b$ will submit its location to $v_a$, and the *rationality of location information* will be determined first.

Since vehicles always drive along certain roads, if $v_b$ deviates from the current road, or its driving direction is opposite to the prescribed one, then it is considered to be malicious. We use $l_r$ to denote the rationality of location information, it relies on the side information grasped by $v_a$. In particular, $l_r$ is divided into 3 cases: rational, irrational and undetermined, which are represented by 1, $-1$ and 0 respectively.

*Authenticity of location information:* After receiving the cooperation request, if $v_b$ is malicious, it may provide a fake location. Generally, the fake location is expected not to be discovered by $v_a$. Comparing with the situation that $v_b$ provides its location directly, the time delay of generating a fake but undiscovered one is much greater [4], [29]. On the basis of taking rationality of location information into account, we also evaluate the authenticity of location information according to the hesitation time of the response. Assuming $t_q$ and $t_r$ are the timestamps that $v_a$ broadcasts the cooperation request and receives the response respectively, we define the authenticity of location information $l_g$ as:

$$l_g = 1 - \frac{(t_r - t_q) - trans(\Delta t_r)}{\Delta \tilde{t}_m},$$

where $trans(\Delta t_r)$ is the expected propagation delay speculated by $v_a$ based on its experiences. $(t_r - t_q) - trans(\Delta t_r)$ can be regarded as the hesitation time. The longer hesitation is, the lower authentic means. $\Delta \tilde{t}_m$ is the maximum waiting time preset by $v_a$.

Based on the above, $v_a$ can obtain the evaluation of $v_b$'s current behaviors, $\tilde{\eta} \in [0, 1]$ balances the rationality and the authenticity of location information, the value of $\tilde{\eta} \in [0, 1]$ under $l_r = 0$ is smaller than the case while $l_r = 1$:

$$E_{rv_a,cv_b} = \begin{cases} \tilde{\eta} l_r + (1 - \tilde{\eta}) l_g, & l_r = 1 \text{ or } l_r = 0 \\ 0, & l_r = -1. \end{cases}$$

*2) Calculation of Vehicle's Trust Level:* Combining the vehicle's current behaviors evaluation result with the recorded historical trust information on the blockchain, $v_a$ and $v_b$ can calculate each other's trust level.

As mentioned before, *Dirichlet distribution* classifies participants' behaviors into several ranks and regards the trust value as the probability that the behavior is at a specified rank. In our scheme, the vehicles' behaviors are divided into $n$ ranks, which are $b_i, i = 1, 2, \ldots, n$. With the increase of $i$, the corresponding reliability of $b_i$ also rises. We define the value interval of $b_i$ as $((i - 1)/n, i/n]$, $1 \le i \le n$. To make it more mathematically precise, it is defined that $0 \in b_1$. For the convenience to assess the required probability, we record the number of times that a vehicle has received each rank on the blockchain. In particular, assuming the current transaction is the *p-th* for $v_a$, and *q-th* for $v_b$, it is obvious that although $v_a$ acts as the requester and $v_b$ acts as the cooperator under the current transaction, both of them may act as either the cooperator or the requester under the other transactions. $a_{rv_a}^{i\_p-1}$, $a_{cv_a}^{i\_p-1}$, $a_{rv_b}^{i\_q-1}$, and $a_{cv_b}^{i\_q-1}$ are the numbers of times that $v_a$ and $v_b$ have received $b_i$ either as a requester or a cooperator. Thus, the recorded historical trust information on the blockchain for $v_a$ is $\overrightarrow{A_{rv_a}^{p-1}} = (a_{rv_a}^{1\_p-1}, a_{rv_a}^{2\_p-1}, \ldots, a_{rv_a}^{n\_p-1})$, $\overrightarrow{A_{cv_a}^{p-1}} = (a_{cv_a}^{1\_p-1}, a_{cv_a}^{2\_p-1}, \ldots, a_{cv_a}^{n\_p-1})$, and for $v_b$, we have $\overrightarrow{A_{rv_b}^{q-1}} = (a_{rv_b}^{1\_q-1}, a_{rv_b}^{2\_q-1}, \ldots, a_{rv_b}^{n\_q-1})$, $\overrightarrow{A_{cv_b}^{q-1}} = (a_{cv_b}^{1\_q-1}, a_{cv_b}^{2\_q-1}, \ldots, a_{cv_b}^{n\_q-1})$.

After obtaining the historical trust information, to calculate the trust level, both $v_a$ and $v_b$ will compute the local number of times of each rank that the other side receives with its current behaviors evaluation. Specifically, if the evaluation of $v_a$'s current behaviors performed by $v_b$ is $b_i$, then, $v_b$ can compute the local number of times of $b_i$ that $v_a$ receives as $\tilde{a}_{rv_a}^{i\_p} = a_{rv_a}^{i\_p-1} + \varepsilon^i$, while the number of times related to other ranks remains unchanged. $a_{rv_a}^{i\_p-1}$ is the global historical number of times of $b_i$ that $v_a$ has received as a requester recorded on the blockchain, and $\tilde{a}_{rv_a}^{i\_p}$ is the local variable computed by $v_b$ in combination with its own evaluation. Similarly, if $v_a$'s evaluation of the current behaviors of $v_b$ is $b_i$, while calculating the trust level, $v_a$ has $\tilde{a}_{cv_b}^{i\_q} = a_{cv_b}^{i\_q-1} + \varepsilon^i$, and except $b_i$, the other ranks' numbers of times still remain unchanged. To promote the honest behaviors of vehicles, with the increment of $b_i$'s reliability, the value of regulation parameter $\varepsilon^i$ is suggested to monotonously decrease, that is $\varepsilon^1 \ge \varepsilon^2 \ge \cdots \ge \varepsilon^n$.

Since the current behaviors evaluation made by the vehicle is not always accurate, we also take its ***authenticity*** into account, which reflects whether the evaluation can accurately describe the behaviors of vehicle. Assuming $p(b_i)$ is the predicted probability that the vehicle's behaviors are rated as $b_i$ based on the recorded historical trust information on the blockchain, for $v_a$, there is $p(b_i) = E(p_{rv_a}(b_i) | \overrightarrow{A_{rv_a}^{p-1}})$, and $p(b_i) = E(p_{cv_b}(b_i) | \overrightarrow{A_{cv_b}^{q-1}})$ for $v_b$. The computation of $E(p_{rv_a}(b_i) | \overrightarrow{A_{rv_a}^{p-1}})$ and $E(p_{cv_b}(b_i) | \overrightarrow{A_{cv_b}^{q-1}})$ will be given latter. If $p(b_i) \ge a\_Thre$, then the evaluation $b_i$ can be regarded as accurate, where $a\_Thre$ is the determination threshold. Taking $\tau \in (0, 1)$ as the tuning parameter, we have:

$$\varepsilon^i = \begin{cases} n + 1 - i, & p(b_i) \ge a\_Thre \\ \tau(n + 1 - i), & \text{else.} \end{cases}$$

However, during the process of LBS, the increase of the vehicle's trust level as a cooperator will only make other vehicles more prefer to cooperate with it, which has no benefits for the vehicle itself. Therefore, the following two situations may appear:

1) The vehicle may only maintain its trust level as a requester, while not paying any attention to the corresponding value as a cooperator.
2) In more extreme cases, the vehicle will only initiate the LBS query without providing any assistance.

To address these issues, for situation (1), we require that when a vehicle's trust level is calculated as a requester, its trust level as a cooperator has to be taken into account. Since vehicles' behaviors are divided into $n$ ranks $\{b_1, b_2, \ldots, b_n\}$, and the value interval of $b_i$ is $((i - 1)/n, i/n]$. When $\lceil n/2 \rceil \le i \le n$, meaning that the behaviors of vehicles are neutral or above it. Let $\hat{B}_c = \{b_{\lceil n/2 \rceil}, b_{\lceil n/2 \rceil + 1}, \ldots, b_n\}$, then when $v_b$ calculates $\tilde{a}_{rv_a}^{i\_p}$ of $v_a$, we have:

$$\tilde{a}_{rv_a}^{i\_p} = \begin{cases} a_{rv_a}^{i\_p-1} + \varepsilon^i T\_L_{cv_a}(\hat{B}_c), & \lceil n/2 \rceil \le i \le n \\ a_{rv_a}^{i\_p-1} + \varepsilon^i / T\_L_{cv_a}(\hat{B}_c), & 1 \le i < \lceil n/2 \rceil, \end{cases}$$

where according to $v_a$'s historical trust information $\overrightarrow{A_{cv_a}^{p-1}} = (a_{cv_a}^{1\_p-1}, a_{cv_a}^{2\_p-1}, \ldots, a_{cv_a}^{n\_p-1})$, $T\_L_{cv_a}(\hat{B}_c)$ is its trust level as a cooperator under the ranks $\hat{B}_c$. Due to the fact that $T\_L_{cv_a}(\hat{B}_c) \in [0, 1]$, the higher of $T\_L_{cv_a}(\hat{B}_c)$ is, the greater $\tilde{a}_{rv_a}^{i\_p}$ will be if the evaluation of $v_a$'s current behaviors is neutral or above it, otherwise, $\tilde{a}_{rv_a}^{i\_p}$ will get smaller.

For situation (2), by defining an additional variable $p\_token$ for each vehicle that reflects the number of times acting as a requester and a cooperator, we also enable vehicles to assist others actively. In particular, while the vehicle initiates a LBS query, the value of $p\_token$ will be reduced by 1, otherwise, if it acts as a cooperator, the corresponding value will be increased by 1. During the process of constructing the anonymous cloaking region, $p\_token$ will be checked as described in the subsequent content.

On the basis of the aforementioned, both $v_a$ and $v_b$ can obtain each rank's number of times of the other side locally. In particular, $v_b$ can obtain $(\tilde{a}^{1\_p}_{rv_a}, \ldots, \tilde{a}^{i\_p}_{rv_a}, \ldots, \tilde{a}^{n\_p}_{rv_a})$ of $v_a$, which is signed as $\overrightarrow{\tilde{A}^{p}_{rv_a}}$, and $v_a$ also has $(\tilde{a}^{1\_q}_{cv_b}, \ldots, \tilde{a}^{i\_q}_{cv_b}, \ldots, \tilde{a}^{n\_q}_{cv_b})$ of $v_b$, signed as $\overrightarrow{\tilde{A}^{q}_{cv_b}}$. Later, *Dirichlet distribution* is adopted to calculate the trust level.

Taking $v_a$ as an example, we assume its probability distribution of $b_i$ is $p_{rv_a}(b_i)$, which is a prediction of $v_a$'s behaviors. In the *p-th* transaction, the value of it refers to the proportion of $b_i$ in all ranks $\overrightarrow{\tilde{A}^{p}_{rv_a}}$, $\overrightarrow{p_{rv_a}} = (p_{rv_a}(b_1), p_{rv_a}(b_2), \ldots, p_{rv_a}(b_n))$. Then, the prediction of $v_a$'s behaviors obeys the *probability density function distribution* as:

$$f\left(\overrightarrow{p_{rv_a}} | \overrightarrow{\tilde{A}^{p}_{rv_a}}\right) = Dir\left(\overrightarrow{p_{rv_a}} | \overrightarrow{\tilde{A}^{p}_{rv_a}}\right)$$
$$= \frac{\Gamma(\sum_{i=1}^{n} \tilde{a}^{i\_p}_{rv_a})}{\prod_{i=1}^{n} \Gamma(\tilde{a}^{i\_p}_{rv_a})} \prod_{i=1}^{n} p_{rv_a}(b_i)^{\tilde{a}^{i\_p}_{rv_a} - 1}.$$

Based on this, the *expectation* of $p_{rv_a}(b_i)$ is:

$$E\left(p_{rv_a}(b_i) | \overrightarrow{\tilde{A}^{p}_{rv_a}}\right) = \frac{\tilde{a}^{i\_p}_{rv_a}}{\sum_{i=1}^{n} \tilde{a}^{i\_p}_{rv_a}}.$$

Obviously, $E(p_{rv_a}(b_i) | \overrightarrow{\tilde{A}^{p}_{rv_a}})$ means the belief that $v_b$ puts on $v_a$ after obtaining $\overrightarrow{\tilde{A}^{p}_{rv_a}}$, which can be viewed as $v_a$'s trust level to behave as $b_i$.

In practice, with the variety of privacy protection requirements, the vehicle's demands for the counterparty may be different. For example, for a vehicle with sensitive privacy, it may only cooperate with the vehicles that behave as $b_n$; otherwise, the vehicle may be willing to interact with the ones that behave as $b_n, b_{n-1}$, or even as $b_{n-i'}$, $1 < i' < n$. This characteristic can be met well by utilizing the *aggregation of Dirichlet distribution*.

*Aggregation of Dirichlet distribution:* It is assumed that random variables $\pi_1, \pi_2, \ldots \pi_n$ obey *Dirichlet distribution*, which is denoted as $(\pi_1, \pi_2, \ldots \pi_n) \sim Dir(\alpha_1, \alpha_2, \ldots, \alpha_n)$. If $(l_1, l_2, \ldots l_j)$ is a partition of $(1, 2, \ldots, n)$, then it holds that $(\sum_{i \in l_1} \pi_i, \ldots, \sum_{i \in l_j} \pi_j) \sim Dir(\sum_{i \in l_1} \alpha_i, \ldots, \sum_{i \in l_j} \alpha_j)$.

Still taking $v_a$ as an example, in the *p-th* transaction, its final trust level that under the ranks specified by $v_b$ is:

$$T\_L_{rv_a}(\tilde{B}_r) = E\left(p_{rv_a}(\tilde{B}_r) | \overrightarrow{\tilde{A}^{p}_{rv_a}}\right) = \frac{\sum_{i \in I_r} \tilde{a}^{i\_p}_{rv_a}}{\sum_{i=1}^{n} \tilde{a}^{i\_p}_{rv_a}},$$

where $\tilde{B}_r$ represents the specified ranks, $I_r$ is the corresponding subscript set, for example $\tilde{B}_r = \{b_n, b_{n-1}\}$, $I_r = \{n, n-1\}$, $p_{rv_a}(\tilde{B}_r) = \cup_{i \in I_r} p_{rv_a}(b_i)$ denotes the union of the probabilities related to $\tilde{B}_r$. Similarly, $v_a$ can also obtain $v_b$'s final trust level that under its specified ranks $\tilde{B}_c$ as below, where $I_c$ is $\tilde{B}_c$'s subscript set, and $p_{cv_b}(\tilde{B}_c) = \cup_{i \in I_c} p_{cv_b}(b_i)$:

$$T\_L_{cv_b}(\tilde{B}_c) = E\left(p_{cv_b}(\tilde{B}_c) | \overrightarrow{\tilde{A}^{q}_{cv_b}}\right) = \frac{\sum_{i \in I_c} \tilde{a}^{i\_q}_{cv_b}}{\sum_{i=1}^{n} \tilde{a}^{i\_q}_{cv_b}}.$$

*3) Update of the Recorded Historical Trust Information on the Blockchain:* After exchange of the cooperation request and response messages, the request vehicle and the cooperation one will evaluate each other's current behaviors, and publish the corresponding evaluation results to the nearby RSUs. The nearby RSUs will broadcast the receiving results to others. Upon receiving the evaluation results related to a vehicle, the specific accounting RSU will update the corresponding vehicle's historical trust information and generate a transaction bill that records the trust information of the vehicle. Once a certain number of bills are generated, these bills will be assembled into a new block by the accounting RSU, and then, PBFT algorithm is executed by all RSUs to reach a consensus. The approach of updating the recorded historical trust information on the blockchain is shown as follows.

It is assumed that the current anonymous cloaking region construction is the *p-th* transaction for $v_a$. After receiving its current behaviors evaluations derived from all cooperative vehicles including $v_b$, the specific accounting RSU will first compute the number of times of the rank that should be updated. It is worth noting that this is different from the calculated value of $\tilde{a}^{i\_p}_{rv_a}$ by $v_b$, since the recorded trust information on the blockchain is global, which means $a^{i\_p}_{rv_a}$, all counterparties' evaluations related to $v_a$'s current behaviors should be taken into account. Specifically, since there exist at least $k - 1$ cooperative vehicles in the construction, it is necessary to aggregate their evaluations first. Assuming $v_a$ receives $y_i$ evaluations of $b_i$ in total, then the final updated rank is $b_s$, where $s = \lfloor \sum_{i=1}^{n} (y_i \cdot i) / \sum_{i=1}^{n} y_i \rfloor$. The updated trust information for $v_a$ is $a^{s\_p}_{rv_a} = a^{s\_p-1}_{rv_a} + \varepsilon^s$, and $a^{i\_p}_{rv_a} = a^{i\_p-1}_{rv_a}, i \neq s$. Similarly, the calculation of $a^{s\_p}_{rv_a}$ will also take $v_a$'s trust level as a cooperator into account. Moreover, because in the *p-th* transaction, $v_a$ acts as a requester, its trust information acts as a cooperator remain unchanged, that is $a^{i\_p}_{cv_a} = a^{i\_p-1}_{cv_a}, 1 \leq i \leq n$.

Meanwhile, supposing the current anonymous cloaking region construction is the *q-th* transaction for $v_b$, since it only interacts with one requester $v_a$, the corresponding rank should be updated directly. If the evaluation of $v_b$'s current behaviors is $b_i$, then we have $a^{i\_q}_{cv_b} = a^{i\_q-1}_{cv_b} + \varepsilon^i$, and all other ranks' numbers of times, including the ones that $v_b$ acts as a requester, remain unchanged.

In our scheme, we make that all RSUs will act as an accounting node by taking turns. Due to the fact that not all RSUs can create the new block accurately while acting as the accounting node, we stipulate that once a RSU creates the wrong block (i.e. cannot reach a consensus), it will be blacklisted and unable to participate in the subsequent accounting work.

## B. Anonymous Cloaking Region Construction

According to the proposed trust management method, we establish the data structure of the transaction bill attached to each vehicle on the blockchain. As shown in Table I, $p\_id$ protects the real identity of the vehicle. $p\_token$ is initialized to $\lambda$, $\lambda \in N_+$. After the *m-th* transaction, there exist $\overrightarrow{A^{m}_{rv}} = (a^{1\_m}_{rv}, \ldots, a^{i\_m}_{rv}, \ldots, a^{n\_m}_{rv})$ and $\overrightarrow{A^{m}_{cv}} = (a^{1\_m}_{cv}, \ldots, a^{i\_m}_{cv}, \ldots, a^{n\_m}_{cv})$ for the vehicle.

The procedure of constructing the anonymous cloaking region is as follows.

TABLE I
THE DATA STRUCTURE RECORDED ON THE BLOCKCHAIN

| ***Notation*** | ***Definition*** |
|---|---|
| $p\_id$ | The vehicle's pseudonym |
| $p\_token$ | The motivation of the vehicle to provide assistance |
| $\overrightarrow{A_{rv}^{m}}$ | The historical trust information of the vehicle as a requester |
| $\overrightarrow{A_{cv}^{m}}$ | The historical trust information of the vehicle as a cooperator |
| $Tim_1$ | The update timestamp of $\overrightarrow{A_{rv}^{m}}$ |
| $Tim_2$ | The update timestamp of $\overrightarrow{A_{cv}^{m}}$ |

*Step 1:* When a request vehicle initiates the LBS query, it first broadcasts the cooperation request

$$Req = \{p\_id_{rv}, t_q, Cer_{p\_id_{rv}}, Num(Tran_{n_{rv}}),$$
$$Sig_{SK-p\_id_{rv}}(t_q || Num(Tran_{n_{rv}}))\}.$$

In particular, $p\_id_{rv}$ is the requester's pseudonym; $t_q$ is the timestamp; $Cer_{p\_id_{rv}}$ is the request vehicle's public key certificate; $Num(Tran_{n_{rv}})$ represents the transaction index number that records its latest historical trust information on the blockchain; $Sig_{SK-p\_id_{rv}}(t_q || Num(Tran_{n_{rv}}))$ is the signature of $t_q$ and $Num(Tran_{n_{rv}})$, which uses the requester's private key $SK - p\_id_{rv}$ to sign, "‖" denotes the concatenation operation.

*Step 2:* After receiving the cooperation request, the cooperative vehicle will first verify the signature. Then, according to $Num(Tran_{n_{rv}})$, the transaction bill $Tran_{n_{rv}}$ will be queried from the nearest RSU. If there holds $p\_token < n\_Thre$, the cooperator will discard the request, where $n\_Thre$ is the preset threshold. It means that the request vehicle is regarded as a selfish one. Otherwise, the cooperator will compute the requester's final trust level $T\_L_{rv}(\tilde{B}_r)$:

- if $T\_L_{rv}(\tilde{B}_r) \geq t_{c,r}\_Thre$, where $t_{c,r}\_Thre$ is the threshold preset by the cooperative vehicle, it will send the response
$$Res = \{p\_id_{cv}, p\_id_{rv}, t_r, Cer_{p\_id_{cv}}, Num(Tran_{n_{cv}}),$$
$$Enc_{PK-p\_id_{rv}}(Loc_{cv}), Sig_{SK-p\_id_{cv}}(t_r ||$$
$$Num(Tran_{n_{cv}}) || Enc_{PK-p\_id_{rv}}(Loc_{cv})\}$$
to the requester. The pseudonym of the cooperator is $p\_id_{cv}$; $t_r$ represents the response timestamp; $Cer_{p\_id_{cv}}$ is the cooperative vehicle's public key certificate; $Num(Tran_{n_{cv}})$ denotes the transaction index number that records the cooperator's latest historical trust information on the blockchain; using requester's public key $PK - p\_id_{rv}$, the cooperator encrypts the location information $Loc_{cv}$, which is further signed along with $t_r$ and $Num(Tran_{n_{cv}})$ by using its private key $SK - p\_id_{cv}$.
- otherwise, there will be no response from the cooperative vehicle.

*Step 3:* While receiving the response, the request vehicle will also query the transaction bill $Tran_{n_{cv}}$ from the nearest RSU. Later, the cooperative vehicle's final trust level $T\_L_{cv}(\tilde{B}_c)$ will also be calculated. Similar to the above,

- if $T\_L_{cv}(\tilde{B}_c) \geq t_{r,c}\_Thre$, where $t_{r,c}\_Thre$ is the requester's preset threshold, the request vehicle will view the cooperator as one of the candidates for constructing the anonymous cloaking region.
- otherwise, the cooperative vehicle will not be considered.

*Step 4:* After completing the evaluation of the other side's current behaviors, both the request vehicle and the cooperative one will publish the results, which are $Rat_{r,c} = \{p\_id_{rv}, p\_id_{cv}, b_i, t_a, Sig_{SK-p\_id_{rv}}(b_i || t_a)\}$ and $Rat_{c,r} = \{p\_id_{cv}, p\_id_{rv}, b_i,' t_a,' Sig_{SK-p\_id_{cv}}(b_i' || t_a')\}$ respectively. $b_i$, $b_i'$, $t_a$ and $t_a'$ are the evaluation results that derived from the requester, the cooperator and the corresponding generated timestamps. Once receiving the evaluations, RSUs will update the recorded historical trust information on the blockchain.

In the aforementioned steps, RSUs are only required to provide the vehicle with the recorded data on the blockchain, they do not participate in the negotiating between vehicles, meaning that all RSUs are acting as a distributed database together at this time. In addition, we assume that no matter the request vehicle or the cooperative one, its query request to the nearest RSU can always be replied in time. Specially, once the vehicle has crossed the coverage of the particular RSU, since according to the transaction index number provided in the request, no matter which RSU is queried from, the accessed trust information has always achieved consensus and been recorded on the blockchain as illustrated in the experiments, it will again query the trust information form the next nearest RSU. Under this case, the next nearest RSU is only required to execute a simple local search to return the result.

## V. SCHEME ANALYSIS

### A. Security

*The pseudonym life time of vehicle $x$:* During the process of constructing the anonymous cloaking region, since the cooperative vehicle provides the location information to the request vehicle, comparing with the requester, the facing risks of its privacy leakage are higher. We take the vehicle acts as a cooperator as an example to analyze the risk of privacy leakage. If the privacy can be protected at this condition, our scheme can always preserve the vehicle's privacy whether it acts as a requester, or a requester and a cooperator alternately. After the request vehicle obtains $h$ locations of the cooperator, it is assumed that the requester can infer the driving track of the cooperative vehicle with the probability $p_s$. Using $p_m$ and $p_t$ to denote the probabilities that the cooperative vehicle within the communication range of the requester and provides its location information respectively. Then, in the pseudonym life time $x$, the requester can infer the privacy of the cooperator with a confidence $P_{r,c} = C_x^h p_s (p_m p_t)^h$. To make $P_{r,c} \leq \xi, \xi \rightarrow 0$, we have $x(x-1)\cdots(x-h+1) \leq \frac{\xi h!}{p_s (p_m p_t)^h}$, which can be used to determine the value of $x$. For example, if $h = 4$, $p_s = 0.75$, $p_m = 0.3$, $p_t = 0.5$ and $\xi = 0.05$, then $x \leq 9$. Based on the fact that changing the pseudonym frequently will increase the processing burden of RA, it can be ordered $x = 9$.

*Resilience to bad-mouthing attack:* While submitting the evaluation result, both the request vehicle and the cooperative one may defame the other side. However, when calculating the vehicle's trust level or updating the recorded historical trust information on the blockchain, the authenticity of the received evaluation is measured. Therefore, the impact of *bad-mouthing attack* is very limited for a well-behaved vehicle; for a vehicle

TABLE II
THE ANALYSIS OF COMPUTATIONAL COMPLEXITY

| *Item* | *Computational Complexity* |
|---|---|
| The requester vehicle | $(z+1)\cdot\mathcal{O}\left(Sig\right)+z\cdot\mathcal{O}\left(Sig'\right)+z\cdot\mathcal{O}\left(Enc'\right)+z\cdot\mathcal{O}\left(1\right)$<br>$=\mathcal{O}\left(Sig\right)+\mathcal{O}\left(Sig'\right)+\mathcal{O}\left(Enc'\right)$ |
| The cooperative vehicle without response | $\mathcal{O}\left(Sig'\right)+\mathcal{O}\left(1\right)=\mathcal{O}\left(Sig'\right)$ *or* $\mathcal{O}\left(Sig'\right)+\mathcal{O}\left(1\right)+\mathcal{O}\left(Sig\right)=\mathcal{O}\left(Sig'\right)+\mathcal{O}\left(Sig\right)$ |
| The cooperative vehicle providing assistance | $\mathcal{O}\left(Sig'\right)+\mathcal{O}\left(1\right)+\mathcal{O}\left(Enc\right)+2\cdot\mathcal{O}\left(Sig\right)$<br>$=\mathcal{O}\left(Sig'\right)+\mathcal{O}\left(Enc\right)+\mathcal{O}\left(Sig\right)$ |

that behaves maliciously, clearly, its trust level will be further decreased under *bad-mouthing attack*.

*Resilience to on-off attack:* Utilizing the dynamic nature of trust information, vehicles may behave honestly and maliciously alternately. Generally, a malicious vehicle is likely to first behave honestly, and then become malicious once its trust level reaches a certain degree. In our scheme, there exist tuning parameters $\varepsilon^1 \geq \varepsilon^2 \geq \cdots \geq \varepsilon^n$ for evaluation ranks $b_1, b_2, \ldots, b_n$. Compared with honest behaviors, the impact of malicious behaviors is harder to forget, meaning the malicious behaviors will be magnified. Therefore, it is necessary for the vehicle to be cautious once behaving maliciously, which effectively restrains *on-off attack*.

*Resilience to whitewashing and Sybil attack:* A vehicle may implement *whitewashing* when its trust level is poor, which means it re-register with RA to reenter the LBS system. Moreover, while the vehicle registers multiple pseudonyms at the same time, *Sybil attack* also occurs, where the vehicle can raise its trust level under a specific pseudonym by utilizing the remaining ones. To address these problems, we can use the information such as driving license that associated with the real world to avoid repeated registration. In addition, by assigning the vehicle with a low initial trust level, our proposal makes *whitewashing* meaningless for it takes the vehicle long time to become trustworthy.

*The authenticity of the recorded information:* After the vehicle publishes the evaluation result about the other side's current behaviors, multiple RSUs nearby are able to receive and broadcast them to all RSUs. Due to the fact that RSUs are aware of the vehicle's current evaluation result as well as the historical information recorded on the blockchain, it is impossible for the accounting RSU to generate a new transaction bill with changed trust information. Therefore, the authenticity of the recorded information can be guaranteed.

## *B. Computational Complexity*

Since computation ability and the storage capacity of RSUs are generally strong, in this subsection, we mainly focus on the computational complexity of the vehicle. As depicted before, the construction process involves signature, verification, encryption, and decryption, the corresponding complexities are defined as $\mathcal{O}(Sig)$, $\mathcal{O}(Sig')$, $\mathcal{O}(Enc)$, and $\mathcal{O}(Enc')$.

When a request vehicle broadcasts the cooperation request, the signature of $t_q$ and $Num(Tran_{n_{rv}})$ will be generated, the required computational complexity is $\mathcal{O}(Sig)$. After receiving the cooperation request, the cooperative vehicle will first verify $Sig_{SK-p\_id_{rv}}(t_q||Num(Tran_{n_{rv}}))$ with a complexity of $\mathcal{O}(Sig')$. Then, according to the queried $Tran_{n_{rv}}$, it will determine whether there holds $p\_token < n\_Thre$. The transaction will be terminated if it is established, the computational complexity is $\mathcal{O}(1)$. Otherwise, the cooperator will evaluate the requester's current behaviors and calculate its trust level that meets the specified ranks. With a complexity of $\mathcal{O}(1)$, the former can be evaluated directly through the devised factors, and the latter can be acquired by using the formula of trust level, the corresponding computational complexity is still $\mathcal{O}(1)$. It is clear that if the requester's trust level does not meet the threshold preset by the cooperator, the transaction will be terminated, the judgment can be completed within the complexity of $\mathcal{O}(1)$. Conversely, the cooperative vehicle will calculate $Enc_{PK-p\_id_{rv}}(Loc_{cv})$ and further sign it along with $t_r$, $Num(Tran_{n_{cv}})$ to return the response, the computational complexity is $\mathcal{O}(Enc) + \mathcal{O}(Sig)$. Once the requester receives the response, it will decrypt and verify the message with a complexity of $\mathcal{O}(Enc') + \mathcal{O}(Sig')$. Similar to the above, the cooperator's trust level is also computed. Moreover, the request vehicle and the cooperative one will also calculate $Sig_{SK-p\_id_{rv}}(b_i||t_a)$ and $Sig_{SK-p\_id_{cv}}(b_i{}'||t_a{}')$ both with the complexity of $\mathcal{O}(Sig)$, so as to publish the evaluation result of the other side.

In summary, if there are $z$ cooperative vehicles in the construction, then, the computational complexity of our scheme is shown in Table II.

## *C. Convergence*

Assuming that there are $N$ vehicles within the acceptable area of the request vehicle, and through point-to-point communication, $N'$ vehicles can receive the broadcast cooperation request. After receiving the request, the corresponding vehicles will further transmit it to other $N'$ vehicles. For each transmission of the vehicle, it is supposed that $Q$ of $N'$ receivers are duplicated on average. Define $p_a$ as the probability that the received vehicle reaches an agreement with the requester. In the case of $p_a(N-1) \geq k-1$, the request vehicle will discover at least $k-1$ cooperators with at most $\lceil \log_{N'-Q}(1 - \frac{(N-1)(1-N'+Q)}{N'}) \rceil$ point-to-point communication. By enlarging the anonymous cloaking region size or reducing the privacy protection requirement k to $p_a(N-1)$, the request vehicle is still able to achieve anonymity while $p_a(N-1) < k-1$.

After the request vehicle and the cooperative vehicle publish the evaluation of the other side's behaviors, the specific accounting RSU will generate the new block which will be agreed among all RSUs through PBFT consensus algorithm. Because PBFT algorithm is usually efficient, it is possible to ensure the recorded information to be updated in time.

# VI. EXPERIMENTS

## *A. Experiment Environment*

As an open source technology platform, Hyperleader is pluggable, customizable, and has been widely adopted in the deployment of consortium blockchain. Referring to the blockchain system, it consists of five layers from bottom to top,

TABLE III
THE VALUE OF EACH PARAMETER IN TRUST EVALUATION

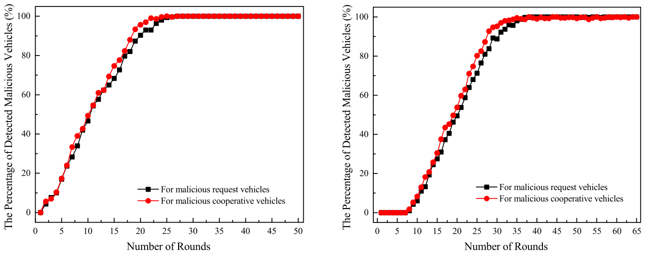
| *Notation* | *Definition* | *Value* |
|---|---|---|
| $r_s^i$ | Querying reasonability from space | **H**: $\ell = 5,\ i = 4, 5$; **M**: $\ell = 5,\ i = 1, 2, 3$ |
| $r_t^i$ | Querying reasonability from time | **H**: $\delta = 10,\ p(q^i) \in [0.6, 1],\ \mu_2 = 0.5$; **M**: $\delta = 10,\ p(q^i) \in [0, 0.4],\ \mu_2 = 0.5$ |
| $l_r$ | Rationality of location information | **H**: 1 (70%), 0 (30%); **M**: 0 (30%), -1 (70%) |
| $l_g$ | Authenticity of location information | **H**: $[0.7, 1]$; **M**: $[0, 0.3]$ |
| $\eta$, $\tilde{\eta}$ | Balancing factors | $\eta = 0.5,\ \tilde{\eta} = 0.5\ (l_r = 1),\ \tilde{\eta} = 0.25\ (l_r = 0)$ |
| $\tau$ | Tuning parameter | 0.25 |
| $a\_Thre$ | Authenticity determination threshold | 0.1 |

including data layer, network layer, consensus layer, contract layer and application layer. To adapt to our scheme, in the experiments, we modified the implementation of each layer if necessary. Specifically, in the data layer, we redefined the recorded data structure in the transaction bill to publish and update the historical trust information of vehicles. In the network layer, we still use the original code to collect transactions and achieve the block transmission and validation among nodes. Since the consensus algorithm in Hyperledger is pluggable, PBFT consensus algorithm is embedded in the consensus layer. Comparing with the consensus algorithms such as PoW, PoS in public blockchain, there is no need for tokens in PBFT. It is efficient with low resource overhead, and is suitable for non-financial environment. As for the contract layer and application layer, the function of chaincode in Hyperledger is employed to realize the interaction between the application layer and the consortium blockchain data, which means the query and the update of vehicles' trust information.

Comparing with the other public key cryptography algorithms such as RSA, since Elliptic Curves Cryptography (ECC) has the characteristics of low computational overhead and high security, we use ECC-secp256k1 and ECDSA-secp256k1 to encrypt/decrypt and sign/verify all messages generated in our scheme. It has been known that ECDSA-secp256k1 algorithm can protect the signature message with 32 bytes (256 bits), and after signature, the length of the message is usually 72 bytes.

All programs are implemented in JAVA language, and the value of each parameter is shown in Table III, where "**H**" and "**M**" denote the cases for the honest vehicles and the malicious ones respectively. It is worth noting that the concrete values of parameters have no substantial impact on the performance of our scheme. The experiment environment is: Intel(R) Core(TM) i5-4590 CPU @ 3.20 GHz, 4 GB DDR3-1600 RAM, and the Operating System is Windows 7.

In the experiments, we divide the behaviors of vehicles into 10 ranks, and the initial value of each rank is defined as 1, that is $a_{rv}^{i\_0} = 1,\ a_{cv}^{i\_0} = 1,\ 1 \leq i \leq 10$. Comparing with the situation that the request vehicle receives a fake location, once the cooperative vehicle provides its real location to a malicious requester, there will be no protection of its location privacy. It is assumed that the cooperator will only assist the requester whose behaviors rank is no less than $b_7$, while the requester is willing to achieve anonymity with the cooperator that ranks no less than $b_5$. Thus, the initial trust level of the requester is 0.3 while 0.5 for the cooperator. To prevent the vehicle from *bad-mouthing attack* just after entering the LBS system, and take the vehicle's personalized requirement into account, we set the thresholds as $t_{c,r}\_Thre \in [0.1, 0.3]$, and $t_{r,c}\_Thre \in [0.25, 0.5]$.



(a) (b)

Fig. 4. The percentage of detected malicious vehicles. (a) Malicious vehicles' behavior pattern: behave honestly first, then maliciously. (b) Malicious vehicles' behavior pattern: behave honestly and maliciously alternately.

## B. Effectiveness

To verify the effectiveness of our scheme, we first illustrate the accuracy in the prediction of vehicle behaviors on the considered scenario, which means the existence of malicious vehicles. It reflects the ability of our scheme to successfully identity the malicious vehicles.

We analyze the behavior pattern of malicious vehicles in detail. Generally, a malicious vehicle usually behaves maliciously after its trust level reaches a certain degree in practical application, it can be divided into two categories: 1) behave honestly in the early stage, and then behave maliciously once the trust level reaches a certain degree; 2) by utilizing the dynamic nature of trust information, the malicious vehicles may behave honestly and maliciously alternately. In the experiments, we assume that both the number of request vehicles and cooperative ones are 1000, where 30% of them are malicious, and the others are honest. Taking the first category into account, a malicious vehicle may start behave maliciously at $\tilde{n}$*-th* rounds, where $0 \leq \tilde{n} \leq n'$. It is assumed $n' = 15$. The results are as shown in Fig. 4(a).

From the results, it can be seen that no matter for a request vehicle or a cooperative one, if it behaves maliciously, it will be detected with the increment of rounds. For instance, when the number of rounds is 20, the percentages of detected malicious request vehicles and malicious cooperative vehicles are 90.33% and 95.67%, respectively. In particular, once a vehicle starts to behave maliciously, the evaluation of its current behaviors will be always low, which leads to the decrease of its trust level and the recorded historical trust information on the blockchain. Later, in a specific construction, the vehicle will be identified as a malicious one if its trust level is lower than the threshold.

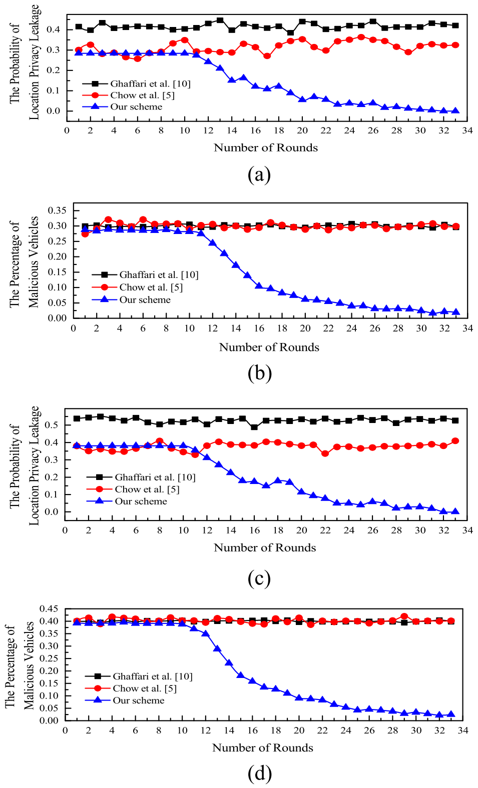
The results of the second category are as shown in Fig. 4(b), we assume that a malicious vehicle takes $5 \leq n'' \leq 20$ as a cycle to behave honestly and maliciously alternately. Once the vehicle behaves maliciously, even it is not detected and continues to behave honestly, the increase of its trust level will still be slow (as shown in Fig. 10). Under this condition, if the vehicle turns to be malicious in the subsequent rounds, its trust level will be decrease rapidly. The corresponding percentages of detected malicious request vehicles and malicious cooperative vehicles are respectively 88.75% and 95.25% when the number of rounds is 30. Obviously, since the malicious vehicles behave honestly and maliciously alternately, the corresponding percentage of detected malicious vehicles is lower than the first category.

We can see that our scheme can detect malicious vehicles with high-accuracy. After 20 or 30 rounds, the detected percentages of malicious request vehicles and malicious cooperative vehicles remain high, thus, the first 20–30 rounds can be regarded as the initialization process of the system. After the initial number of rounds, our scheme can eliminate the malicious vehicles and work stably, meaning constructing the anonymous cloaking region with trusted vehicles.

In addition, for the vehicle behaves honestly, no matter it acts as a requester or a cooperator, its initial trust level has been higher than the threshold, based on which its trust level will continuously increase (as shown in Fig. 6(a)). Thus, the vehicle will always be discovered as the honest one, meaning our scheme can also accurately identify the honest vehicles.

We also compare the proposal with the schemes proposed by Chow *et al.* [5] and Ghaffari *et al.* [10] respectively. The former is the first distributed $k$-anonymity scheme, while the latter is the latest representative solution. The experiments data is derived from Uppoor *et al.* [32], they collected 24 hours driving experiences in the 400 km$^2$ over the city of Cologne in Germany. Among the data, we treat some vehicles as requesters, and the others as cooperators. If the cooperative vehicle provides its location to a malicious requester, its location privacy will be leaked; if the request vehicle cooperate with the malicious cooperator, the percentage of malicious vehicles in the anonymous cloaking region will be high. We assume that the ratios of malicious vehicles are 30% and 40% respectively. For simplicity, it is supposed that the behavior pattern of malicious vehicles are under the first category, and once a vehicle is malicious as a requester (cooperator), then it always tends to be malicious while as a cooperator (requester).

In [10], Ghaffari *et al.* indicated that once the number of collected LBS queries is less than $k$, the anonymizer (viewed as requester) can generate fake ones or collaborate with the adjacent anonymizers. Since Chow *et al.* have pointed out that the adversary is able to estimate the locations of participants in the anonymous cloaking region, we choose the latter in the simulation, which means that once there exists a malicious anonymizer in the collaboration, the location privacy of cooperators will be leaked, resulting in the highest probability of privacy leakage among comparison. As shown in Fig. 5, because Chow *et al.* and Ghaffari *et al.* did not take the trustworthiness of participants into account, their curve values remain unchanged basically. In our scheme, the vehicle's malicious behaviors will lead to a decrease of its trust level. As a result, it cannot participate in the subsequent anonymity. With the increment of rounds, the probability of location privacy leakage and the percentage of malicious vehicles will decrease.



(a)

(b)

(c)

(d)

Fig. 5. The probability of location privacy leakage and the percentage of malicious vehicles in the anonymous cloaking region under different ratios of malicious vehicles. *Compared with [5] and [10], in our scheme, the probability of location privacy leakage and the percentage of malicious vehicles gradually decrease, and with the increment of rounds, both of them tend to be 0.* (a) The ratio of malicious request vehicles is 30%. (b) The ratio of malicious cooperative vehicles is 30%. (c) The ratio of malicious request vehicles is 40%. (d) The ratio of malicious cooperative vehicles is 40%.

In addition, [15], [18] and [21] are respectively the classical works of entity-oriented trust model, data-oriented trust model and combined trust model, they are highly cited in the works related to trust management under VANET. A lot of survey reviews, such as [11], [33], also illustrate this point. We also compare our scheme with them from three items, which are whether depends on the third party, whether the identity privacy of vehicles is preserved, and whether specific trust evaluation factors are devised. The first item determines that if the proposal is applicable to distributed $k$-anonymity; the second represents that if the proposed scheme can protect the vehicle's identity while evaluating trust, making it impossible for the malicious vehicles to further infer the privacy information of the counterparty, such as the driving trajectories; and the last one indicates if there is guidance while determining the trustworthiness of vehicles. These three items are relatively important for distributed $k$-anonymity, and they are usually used as metrics in many trust management works in VANET, such as [13], [33] and so on. [15], [18] and [21] focus on different aspects and solve important problems for trust management in different scenarios, however, they don't have a comprehensive consideration of those three important comparison items for distributed $k$-anonymity in VANET. The details are described below.

As mentioned before, Li *et al.* [15] introduced a third reputation server in the proposed scheme. They believed that once the

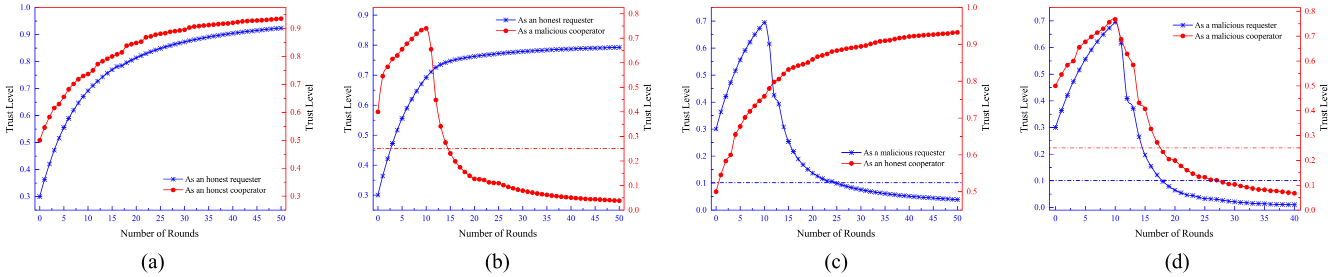


Fig. 6. The variation trend of the vehicle's trust level as a requester and a cooperator. *It can be seen that the vehicle's trust level varies with its behaviors, and the trust level as a cooperator has much effect on its trust level as a requester.* (a) Honest requester & Honest cooperator. (b) Honest requester & Malicious cooperator. (c) Malicious requester & Honest cooperator. (d) Malicious requester & Malicious cooperator.

vehicle has a valid reputation certificate issued by the server, the messages derived from it is reliable. This process does not involve the evaluation of the message itself, no trust evaluation factors are given. Moreover, once the vehicle broadcasts the message, it will also provide the reputation certificate containing its identity to the neighbors, resulting in the identity privacy reveal.

By means of the comprehensive consideration of the default trustworthiness, the event- or task-specific trustworthiness and so on, Raya *et al.* [18] first measured the trustworthiness of messages provided by each vehicle. Then, by adopting logical decision-making methods, the opinion of an event involved in multiple messages derived from different vehicles can be obtained. All these operations are performed by the messages receiving vehicle itself. Meanwhile, the evaluation such as the default trustworthiness of the message depends on the experiences of the receiving vehicle put on the publishing one, meaning vehicles are aware of the identity of each other, which is difficult for the proposal to preserve the identity privacy.

Based on opinion piggybacking, Dőtzer *et al.* [21] proposed a reputation system, where each forwarding peer will compute and append its opinion about the event's trustworthiness in terms of the attached opinions of previous vehicles and its own experiences before further forwarding. In their scheme, Dőtzer *et al.* assumed that there already existed the opinions of previous vehicles, and the vehicle's own experiences. Moreover, the forwarding peer's identity information is always contained in the attached opinion, thus, the identity privacy of vehicles cannot be protected.

In our scheme, we not only devise the specific factors reflecting the trust degree of vehicles in Section IV, but also present the pseudonym method and explore its life time in security analysis. Since the process of trust management mainly relies on the vehicle's experiences of the context and the historical trust information recorded on the blockchain, there is no need for the participation of the third party. The results are shown in Table IV. It can be seen that compared with the existing schemes, our scheme has obvious advantage in the trust management for distributed anonymous cloaking region construction in VANET.

### C. Trust Level

In this subsection, we explore the variation trend of the vehicle's trust level. In distributed $k$-anonymity, a vehicle can act as a requester and a cooperator, and it may have inconsistent performance under different roles. Therefore, we explore the values of trust level under four cases, in which the vehicle acts as 1) an honest requester and an honest cooperator; 2) an honest requester and a malicious cooperator; 3) a malicious requester and an honest cooperator; and 4) a malicious requester and a malicious cooperator, respectively. For a malicious vehicle, we still assume that it behaves like the first category as mentioned before, and $\tilde{n} = 10$.

TABLE IV
Comparison With the Existing Trust Management Schemes

| *Schemes* | *Non-requirement of the third party* | *Vehicles' identity privacy protected* | *Trust evaluation factors given* |
|---|---|---|---|
| Li *et al.* [15] | × | × | × |
| Raya *et al.* [18] | √ | × | √ |
| Dőtzer *et al.* [21] | √ | × | × |
| Ours | √ | √ | √ |

As shown in Fig. 6, no matter the vehicle acts as a requester or a cooperator, if it behaves honestly, the corresponding trust level will gradually increase, on the contrary, its trust level will decrease rapidly. The blue and red dotted lines are the lower bounds of $t_{c,r}\_Thre$ and $t_{r,c}\_Thre$ that we recommended to avoid the vehicle suffers *bad-mouthing attack* just after entering the LBS system. It is worth noting that even the vehicle is honest as a requester, its trust level still increases slowly once it behaves maliciously as a cooperator, as shown in Fig. 6(a) and Fig. 6(b). In particular, after 50 rounds, the vehicle's trust level as a requester in Fig. 6(b) is 0.83 and in Fig. 6(a), there only requires 22 rounds to reach the same degree. Similarly, in Fig. 6(d), the trust level of the vehicle as a requester decreases faster than the case that it behaves honestly as a cooperator in Fig. 6(c).

### D. Efficiency

In this subsection, we first explore the required time delay of constructing the anonymous cloaking region. Specifically, we import the driving experiences collected by Uppoor *et al.* [32] into OPNET Modeler 14.5 to simulate the driving behaviors of vehicles, meanwhile, the corresponding RSUs are also deployed. Assuming the ratio of malicious vehicles is 30%, the experimental results are shown in Fig. 7.

It can be seen that during the process of constructing the anonymous cloaking region, the required time delay of the

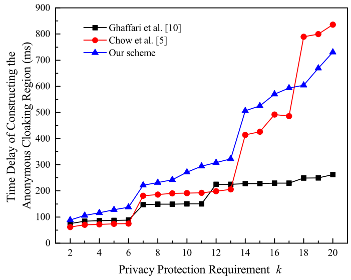


Fig. 7. The required time delay of constructing the anonymous cloaking region. *Comparing with the existing works, the required time delay of our scheme is acceptable.*

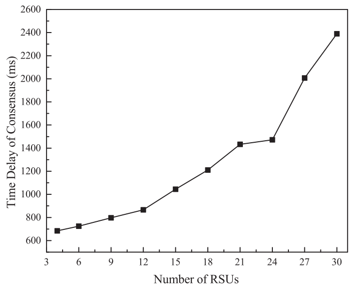


Fig. 8. The required time delay of reaching a consensus. *The more RSUs there are, the more required time delay of consensus process.*

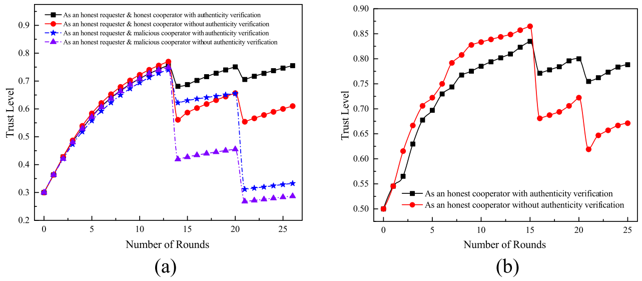


Fig. 9. The variation trend of trust level under *bad-mouthing attack. By means of the authenticity of the evaluation, our scheme is resilient to bad-mouthing attack.* (a) For the request vehicle. (b) For the cooperative vehicle.

scheme proposed by Ghaffari *et al.* [10] is the least. In their scheme, the query initiator will directly send its location information to the anonymizer in plaintext for anonymization. Chow *et al.* [5] makes the requester first broadcast the cooperation request of its one-hop neighbors. If the number of cooperators does not meet the privacy protection requirement, it will rebroadcast the cooperation request to seek cooperation involving nodes within two hops, and so on. Therefore, with the increment of $k$, the required time delay also increases rapidly. In our scheme, during the construction process, we evaluate the trustworthiness of vehicles. It is required to abandon unreliable cooperators. Meanwhile, the interaction process between the requester and the cooperator needs cryptographic operations. However, there is no significant difference of time delay from our scheme to the existing works, meaning that the required time overhead is acceptable.

We also simulate the process of RSUs to reach a consensus. It is assumed that by taking turns to act as the accounting node, every 100 transaction bills generated in the system, the specific accounting RSU will assemble them into a new block, and all RSUs involved will run PBFT consensus algorithm. The consensus process includes the accounting RSU's signature of the new block, the verification performed by the other RSUs, the propagation of the block among RSUs, and the required communication with each other to reach a consensus. Fig. 8 shows the relationship between the number of RSUs and the time delay of reaching a consensus. Obviously, the more RSUs there are, the more required time delay of consensus process. It is because that the more interaction rounds are needed to reach a consensus. In consensus process, there include several stages of communication among the accounting RSU and the others: *pre-prepare*, *prepare*, *commit* and *reply*. In *pre-prepare*, the accounting RSU broadcasts the new generated block to other RSUs. By means of two node-to-node interactions, all participating RSUs are reaching an agreement after the stages of *prepare* and *commit* (the accounting RSU does not participant in *prepare*). And the other RSUs will return their response to the accounting one in *reply*. Assuming the number of RSUs is $\tilde{N}$, and none of them is downtime. If we take the message delivery between two nodes as an interaction round, then the required number of interaction rounds to reach a consensus of all RSUs is: $(\tilde{N}-1)+(\tilde{N}-1)^2+\tilde{N}(\tilde{N}-1)+(\tilde{N}-1)=2\tilde{N}^2-\tilde{N}-1$.

Furthermore, to improve the efficiency of consensus process, Sharding Pattern [34] can be introduced to cope with the efficiency drop caused by the increase number of RSUs.

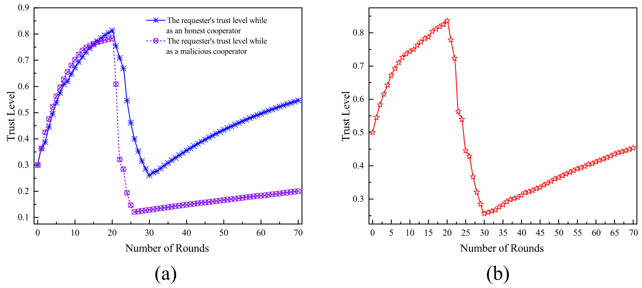
It is worth noting that once all RSUs have reached a consensus for the newly generated block, all of them will complete the update of the local storage blockchain. In our scheme, the historical trust information returned by the nearest RSU has always achieved consensus. Specifically, if a RSU receives the query request about the vehicle whose latest information does not reach a consensus, it will still return the corresponding result directly according to the transaction index number provided by the query vehicle as shown in *Step 1* and *Step 2*, which has achieved consensus.

### *E. Robustness*

While submitting the evaluation of current behaviors, both the request vehicle and the cooperative vehicle may defame the other side. Assuming the requester and the cooperator suffer from *bad-mouthing attack* in 14, 21 and 16, 22 rounds respectively. The results are as shown in Fig. 9. Since the ***authenticity*** of the evaluation is verified in our scheme, for an honest vehicle, even it suffers *bad-mouthing attack* occasionally, its trust level will not decrease remarkably. Meanwhile, if the vehicle continues to behave honestly, the trust level will rise steadily.

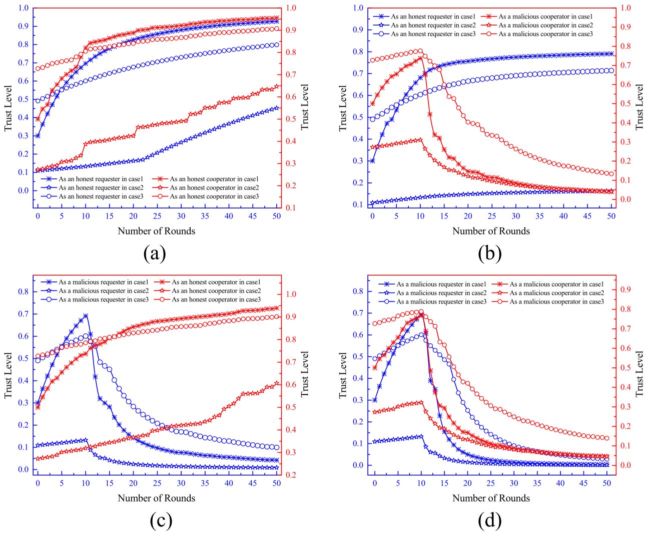
By utilizing the dynamic nature of trust information, the vehicle may also launch *on-off attack*. We assume that both the request vehicle and the cooperative vehicle behave honestly in the former 20 rounds, after the trust levels reach a certain degree, their behaviors become malicious. Later, once approaching the threshold, the behaviors will back to be honest. As shown in Fig. 10, if the vehicle launches *on-off attack*, its trust level decreases rapidly, and the recovery process is rather long.

Furthermore, we also explore the impact brought by the initial values of evaluation ranks. Specifically, we have three cases,



(a) (b)

Fig. 10. The variation trend of trust level under *on-off attack. Once the vehicle launches on-off attack, it takes long time to recover its trust level.* (a) For the request vehicle. (b) For the cooperative vehicle.



(a) (b) (c) (d)

Fig. 11. The variation trend of the vehicle's trust level under different initial values of evaluation ranks. *Different initial values lead to different rate of variation, while the corresponding trend is the same.* (a) Honest requester & Honest cooperator. (b) Honest requester & Malicious cooperator. (c) Malicious requester & Honest cooperator. (d) Malicious requester & Malicious cooperator.

where 1) the initial value of each rank is 1, that is $a_{rv}^{i\_0} = 1$, $a_{cv}^{i\_0} = 1$; 2) the initial value of each rank is decreasing in turn, meaning $a_{rv}^{i\_0} = 10 - i + 1$, $a_{cv}^{i\_0} = 10 - i + 1$; 3) the initial value of each rank is increasing, which equals to $a_{rv}^{i\_0} = i$, $a_{cv}^{i\_0} = i$, $1 \leq i \leq 10$.

As shown in Fig. 11, with the increment of the number of rounds, the variation trend of the vehicle's trust level in case1 is significant, while in case2 and case3, it is relatively gentle. According to the specific requirements, different initial values should be defined in practice. For example, the initial values in case2 can be adopted if the new vehicle is supposed to be malicious.

## VII. Conclusion

Due to the lack of consideration on vehicles' trustworthiness during the process of constructing the anonymous cloaking region, existing distributed *k*-anonymity schemes are unable to protect the location privacy in VANET. This leads that either the request vehicle or the cooperative one may be traced by adversaries, who will infer the sensitive information of their drivers as well as threaten the safety of personal property. To address the problem, we propose a blockchain enabled anonymous cloaking region construction scheme based on trust mechanism. It ensures that the requester and the cooperator will only cooperate with the vehicles they trust. In particular, by analyzing the various requirements of the request vehicle and the cooperative vehicle, we put forward the evaluation method of their behaviors. Moreover, with the help of blockchain, our scheme also establishes a secure and trusted distributed database, which records the trust information on publicly available blocks. Security analysis and extensive experiments indicate that the proposal is resilient to various attacks of trust models, and it does not violate the original intention of participant anonymity in distributed *k*-anonymity. Comparing with the existing works, our scheme can effectively manage the trustworthiness of vehicles, where the probability of location privacy leakage and the percentage of malicious vehicles in the anonymous cloaking region are reduced. During the construction process, the required time overhead is acceptable. Meanwhile, the update and maintenance of the blockchain can be achieved under limited time delay. Overall, the proposal enhances the reliability of LBS in VANET.

## References

[1] C. Guo, D. Li, G. Zhang, and M. Zhai, "Real-time path planning in urban area via VANET-assisted traffic information sharing," *IEEE Trans. Veh. Technol.*, vol. 67, no. 7, pp. 5635–5649, Jul. 2018.

[2] J.-S. Weng, J. Weng, Y. Zhang, W. Luo, and W. Lan, "Benbi: Scalable and dynamic access control on the northbound interface of SDN-based VANET," *IEEE Trans. Veh. Technol.*, vol. 68, no. 1, pp. 822–831, Jan. 2019.

[3] M. Yang, T. Zhu, K. Liang, W. Zhou, and R. H. Deng, "A blockchain-based location privacy-preserving crowdsensing system," *Future Gener. Comput. Syst.*, vol. 94, pp. 408–418, 2019.

[4] H. Liu, X. Li, H. Li, J. Ma, and X. Ma, "Spatiotemporal correlation-aware dummy-based privacy protection scheme for location-based services," in *Proc. IEEE Conf. Comput. Commun.*, 2017, pp. 1–9.

[5] C.-Y. Chow, M. F. Mokbel, and X. Liu, "A peer-to-peer spatial cloaking algorithm for anonymous location-based service," in *Proc. 14th Annu. ACM Int. Symp. Adv. Geographic Inf. Syst*, 2006, pp. 171–178.

[6] G. Ghinita, P. Kalnis, and S. Skiadopoulos, "Mobihide: A mobilea peer-to-peer system for anonymous location-based queries," in *Proc. Int Symp. Spatial Temporal Databases*, 2007, pp. 221–238.

[7] G. Sun, D. Liao, H. Li, H. Yu, and V. Chang, "L2p2: A location-label based approach for privacy preserving in lbs," *Future Gener. Comput. Syst.*, vol. 74, pp. 375–384, 2017.

[8] X. Li, M. Miao, H. Liu, J. Ma, and K.-C. Li, "An incentive mechanism for k-anonymity in lbs privacy protection based on credit mechanism," *Soft Comput.*, vol. 21, no. 14, pp. 3907–3917, 2017.

[9] G. Zhong and U. Hengartner, "A distributed k-anonymity protocol for location privacy," in *Proc. IEEE Int. Conf. Pervasive Comput. Commun.*, 2009, pp. 1–10.

[10] M. Ghaffari, N. Ghadiri, M. H. Manshaei, and M. S. Lahijani, "$P^4QS$: A peer-to-peer privacy preserving query service for location-based mobile applications," *IEEE Trans. Veh. Technol.*, vol. 66, no. 10, pp. 9458–9469, Oct. 2017.

[11] J. Zhang, "A survey on trust management for VANETS," in *Proc. IEEE Int. Conf. Adv. Inf. Netw. Appl.*, 2011, pp. 105–112.

[12] B. Sharef, R. Alsaqour, M. Alawi, M. Abdelhaq, and E. Sundararajan, "Robust and trust dynamic mobile gateway selection in heterogeneous VANET-UMTS network," *Veh. Commun.*, vol. 12, pp. 75–87, 2018.

[13] C. A. Kerrache, N. Lagraa, C. T. Calafate, and A. Lakas, "TFDD: A trust-based framework for reliable data delivery and dos defense in VANETS," *Veh. Commun.*, vol. 9, pp. 254–267, 2017.

[14] U. F. Minhas, J. Zhang, T. Tran, and R. Cohen, "Towards expanded trust management for agents in vehicular ad-hoc networks," *Int. J. Comput. Intell.: Theory Pract.*, vol. 5, no. 1, pp. 3–15, 2010.

[15] Q. Li, A. Malip, K. M. Martin, S. L. Ng, and J. Zhang, "A reputation-based announcement scheme for VANETS," *IEEE Trans. Veh. Technol.*, vol. 61, no. 9, pp. 4095–4108, Nov. 2012.

[16] Z. Lu, W. Liu, Q. Wang, G. Qu, and Z. Liu, "A privacy-preserving trust model based on blockchain for VANETS," *IEEE Access*, vol. 6, pp. 45655–45664, 2018.

[17] Z. Yang, K. Zheng, K. Yang, and V. C. Leung, "A blockchain-based reputation system for data credibility assessment in vehicular networks," in *Proc. IEEE 28th Annu. Int. Symp. Pers., Indoor, Mobile Radio Commun.*, 2017, pp. 1–5.

[18] M. Raya, P. Papadimitratos, V. D. Gligor, and J.-P. Hubaux, "On data-centric trust establishment in ephemeral ad hoc networks," in *Proc. IEEE 27th Conf. Comput. Commun.*, 2008, pp. 1238–1246.

[19] P. Golle, D. Greene, and J. Staddon, "Detecting and correcting malicious data in VANETS," in *Proc. 1st ACM Int. Workshop Veh. Ad Hoc Netw.*, 2004, pp. 29–37.

[20] S. Gurung, D. Lin, A. Squicciarini, and E. Bertino, "Information-oriented trustworthiness evaluation in vehicular ad-hoc networks," in *Proc. Int. Conf. Netw. Syst. Secur.*, 2013, pp. 94–108.

[21] F. Dőtzer, L. Fischer, and P. Magiera, "Vars: A vehicle ad-hoc network reputation system," in *Proc. 6th IEEE Int. Symp. World Wireless Mobile Multimedia Netw.*, 2005, pp. 454–456.

[22] A. Patwardhan, A. Joshi, T. Finin, and Y. Yesha, "A data intensive reputation management scheme for vehicular ad hoc networks," in *Proc. 3rd IEEE Annu. Int. Conf. Mobile Ubiquitous Syst. Workshop*, 2006, pp. 1–8.

[23] T. R. V. Krishna, R. P. Barnwal, and S. K. Ghosh, "CAT: Consensus-assisted trust estimation of MDS-equipped collaborators in vehicular ad-hoc network," *Veh. Commun.*, vol. 2, no. 3, pp. 150–157, 2015.

[24] G. Xu, J. Liu, Y. Lu, X. Zeng, Y. Zhang, and X. Li, "A novel efficient maka protocol with desynchronization for anonymous roaming service in global mobility networks," *J. Netw. Comput. Appl.*, vol. 107, pp. 83–92, 2018.

[25] L. Li *et al.*, "Creditcoin: A privacy-preserving blockchain-based incentive announcement network for communications of smart vehicles," *IEEE Trans. Intell. Transp. Syst.*, vol. 19, no. 7, pp. 2204–2220, Jul. 2018.

[26] Y. Zhang and Y. Fang, "A fine-grained reputation system for reliable service selection in peer-to-peer networks," *IEEE Trans. Parallel Distrib. Syst.*, vol. 18, no. 8, pp. 1134–1145, Aug. 2007.

[27] S. D. Kamvar, M. T. Schlosser, and H. Garcia-Molina, "The eigentrust algorithm for reputation management in p2p networks," in *Proc. 12th Int. Conf. World Wide Web*, 2003, pp. 640–651.

[28] A. Josang and R. Ismail, "The beta reputation system," in *Proc. 15th Bled Electron. Commerce Conf.*, 2002, vol. 5, pp. 2502–2511.

[29] B. Niu, Q. Li, X. Zhu, G. Cao, and L. Hui, "Achieving k-anonymity in privacy-aware location-based services," in *Proc. IEEE Conf. Comput. Commun.*, 2014, pp. 754–762.

[30] B. Niu, Q. Li, X. Zhu, G. Cao, and L. Hui, "Enhancing privacy through caching in location-based services," in *Proc. IEEE Conf. Comput. Commun.*, 2015, pp. 1017–1025.

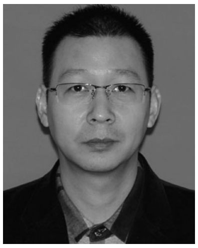
[31] S. Zhang, X. Li, Z. Tan, T. Peng, and G. Wang, "A caching and spatial k-anonymity driven privacy enhancement scheme in continuous location-based services," *Future Gener. Comput. Syst.*, vol. 94, pp. 40–50, 2019.

[32] S. Uppoor, O. Trullols-Cruces, M. Fiore, and J. M. Barcelo-Ordinas, "Generation and analysis of a large-scale urban vehicular mobility dataset," *IEEE Trans. Mobile Comput.*, vol. 13, no. 5, pp. 1061–1075, May 2014.

[33] S. A. Soleymani *et al.*, "Trust management in vehicular ad hoc network: A systematic review," *EURASIP J. Wireless Commun. Netw.*, vol. 2015, no. 1, pp. 1–22, 2015.

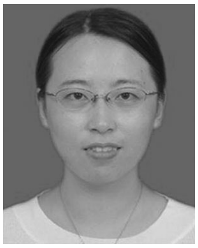
[34] E. Kokoris-Kogias, P. Jovanovic, L. Gasser, N. Gailly, E. Syta, and B. Ford, "Omniledger: A secure, scale-out, decentralized ledger via sharding," in *Proc. IEEE Symp. Secur. Privacy*, 2018, pp. 583–598.

**Bin Luo** received the B.S. degree in computer science from Xidian University, Xi'an, China, in 2015, where she is currently working toward the Ph. D. degree in security of cyberspace. Her research interests include privacy protection and trust management.



**Xinghua Li** received the M.E. and Ph.D. degrees in computer science from Xidian University, Xi'an, China, in 2004 and 2007, respectively. He is currently a Professor with the School of Cyber Engineering, Xidian University. His research interests include wireless networks security, privacy protection, cloud computing, and security protocol formal methodology.

**Jian Weng** received the Ph.D. degree with Shanghai Jiao Tong University, Shanghai, China, in 2008. He is currently a Professor and an Executive Dean with the College of Information Science and Technology, Jinan University, Guangzhou, China. His research interests include public key cryptography, cloud security, blockchain, etc. He served as a PC Co-Chairs or PC Member for more than 20 international conferences.



**Jingjing Guo** received the M.E. and Ph.D. degrees in computer science from Xidian University, Xi'an, China, in 2012 and 2015, respectively. She is currently a Lecturer with the School of Cyber Engineering, Xidian University. Her research interests include trust management, social networks, access control, and information security.

**Jianfeng Ma** received the M.E. and Ph.D. degrees in computer software and communications engineering from Xidian University, Xi'an, China, in 1988 and 1995, respectively. He is currently a Professor with the School of Cyber Engineering, Xidian University, China. His research interests include information and network security, coding theory and cryptography.