Peer-to-Peer Networking and Applications (2019) 12:1178–1193
https://doi.org/10.1007/s12083-019-00786-4

# Blockchain Meets VANET: An Architecture for Identity and Location Privacy Protection in VANET

Hui Li[1,2] · Lishuang Pei[1] · Dan Liao[1,2] · Gang Sun[1,3] · Du Xu[1]

Received: 30 October 2018 / Accepted: 3 July 2019 / Published online: 20 July 2019

## Abstract

With the breakthroughs in sensor technology and internet of things, Vehicular Ad Hoc Network (VANET) is developing into a new generation. The technical challenges of current VANET are decentralized architecture deployment and privacy protection. Since the blockchain owns the characteristics of being decentralized, distributed, collective maintenance and non-tampering, this paper designs a novel decentralized architecture using blockchain technology, which is called blockchain-based VANET. The blockchain-based VANET involves four major stages: blockchain set-up, registration of vehicles, SBMs upload, and blockchain record. It can effectively address the problems of centralization and mutual distrust between entities in current VANET. For protecting identity and location privacy, we propose UGG, IPP and LPP algorithms with the way of dynamic threshold encryption and *k*-anonymity unity in the stage of SBMs upload of blockchain-based VANET. To quantify the availability of *k*-anonymity unity, we propose two indicators: connectivity and average distance. Extensive simulations have been conducted to validate the effectiveness of blockchain-based VANET. We analyze the simulation results from four aspects: system time, average distance, connectivity, and privacy leakage. The simulation results show that our proposed architecture performs better in terms of processing time than current architectures. Furthermore, our proposed architecture shows its superior in the aspect of protecting identity and location privacy.

**Keywords** Blockchain · Identity privacy · Location privacy · Decentralized · VANET

## 1 Introduction

According to the latest research report by Counterpoint Internet of Things (IoT) servers, the global internet car market is expected to grow by 270% by 2022 [1]. As the number of vehicles increases exponentially [2], it brings people more convenient transportation. However, it also causes new problems, such as traffic safety, environmental protection, and privacy security [3]. Therefore, Vehicular Ad Hoc Network [4–6] (VANET) has become a hot research in the field of transportation. VANET is expected to change the existing problems of the current transportation system and realizes intelligent traffic management. In VANET, through the aggregation and constant exchange of Safety Beacon Messages (SBMs), all vehicles can receive safety information in a timely manner and be aware of the surrounding traffic environment, such as traffic flow, traffic congestion, etc.

The traditional architecture [7] of VANET is shown in Fig. 1, which mainly includes vehicles, RSU, CA, and core network server. SBMs are collected through the sensor on the vehicles,

✉ Dan Liao
dliao.uestc@gmail.com

Hui Li
huier.uestc@uestc.edu.cn

Lishuang Pei
peilisher@gmail.com

Gang Sun
gangsun@uestc.edu.cn

Du Xu
xudu@uestc.edu.cn

[1] Key Lab of Optical Fiber Sensing and Communications (Ministry of Education), University of Electronic Science and Technology of China, No. 2006 Xiyuan Road, hi-tech West District, Chengdu, Sichuan, China

[2] Chengdu Research Institute, University of Electronic Science and Technology of China, No. 2006 Xiyuan Road, hi-tech West District, Chengdu, Sichuan, China

[3] Center for Cyber Security, University of Electronic Science and Technology of China, No. 2006 Xiyuan Road, hi-tech West District, Chengdu, Sichuan, China

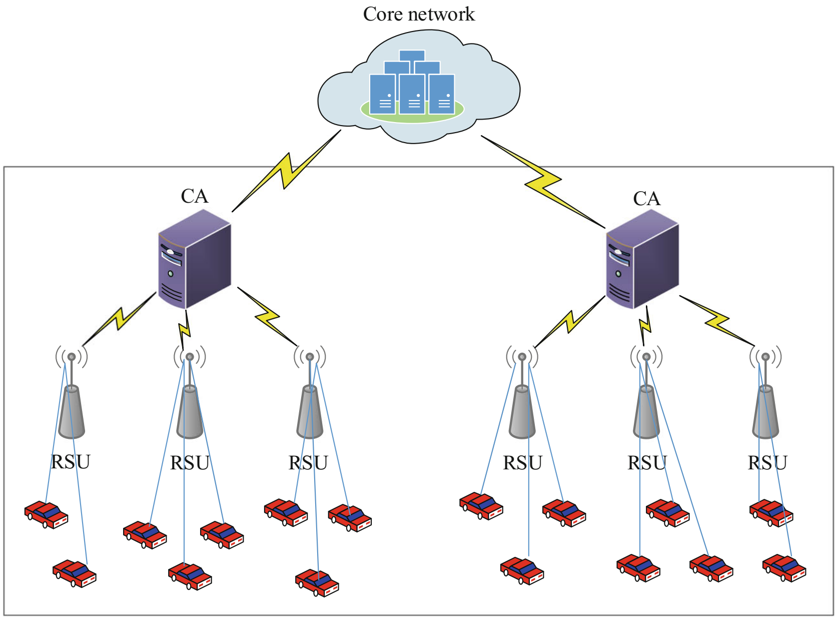


**Fig. 1** Traditional architecture of VANET

and transmitted to the upper layer by RSU. Finally, these SBMs are gathered to the core network for centralized processing. Therefore, the traditional architecture of VANET is based on centralized system, which is convenient for centralized management of vehicular information. However, traditional architecture must rely on trusted centralized entities (like CA in the Fig. 1). And it cannot guarantee that these centralized entities are completely credible. Once a centralized entity is attacked, it will bring serious data security risk. By mining SBMs, the attacker can get users' private information, such as identity, location, social status, etc., which will bring life disturbance and even life security to users. In addition, with the continuous development of sensor technology and IoT technology, the data volume increases dramatically. The centralized management of data in traditional architecture leads to excessive load on central entities and bottleneck problems. Traditional architecture of VANET is also at risk of a single point of failure. Therefore, it is urgent to study the decentralized architecture of VANET to ensure the safety of users' data.

In VANET, SBM contains vehicular identity, speed, location, content of request, etc. The identity and location information are particularly important [8–10]. The vehicular identity information is generally protected using pseudonym, which is generated by centralized entity CA in traditional VANET. However, once the CA is compromised, the vehicular identity privacy is threatened. Furthermore, all SBMs are generated based on location information. Only with accurate location information can VANET provide high-quality services to vehicles. The existing protection way of location information generally adopts encryption method, which can only guarantee that the location information will not be stolen or leaked easily in transmission process. However, it cannot guarantee that the location information will not be leaked by the CA or other servers in centralized architecture. Thus, the location privacy is threatened. Moreover, users are paying more and more attention to their private information in data era [11–14]. So it is very important to protect the vehicular identity and location privacy in VANET.

In this paper, to protect identity and location privacy, the decentralized architecture of VANET is constructed by using blockchain technology. Blockchain [15] technology is a kind of chain data structure, which is composed of data blocks connected sequentially according to time sequence. And blockchain owns a distributed ledger that cannot be tampered or forged using cryptography. Blockchain technology has typical features of decentralization, distribution, collective maintenance and non-tampering. It can effectively solve the problems of centralization, mutual distrust between entities and privacy leakage of traditional VANET. Therefore, a decentralized architecture is proposed to protect the identity and location privacy in VANET. The main contributions of this paper are as follows:

- To address the problem of centralization in traditional VANET. This paper brings in blockchain technology and designs a novel decentralized architecture, blockchain-based VANET. The hash of SBMs is recorded in the blockchain of our proposed architecture, which not only ensues the integrity and security of SBMs, but also saves the blockchain storage and processing time.
- To protect vehicular identity privacy, the identity of vehicle is divided into multiple sub-identities by the way of ($m$, $r$) threshold secret sharing scheme. And the sub-identities are updated periodically to prevent attacker from

acquiring at most $m$ sub-identities of vehicle to calculate vehicular real identity.

- To protect vehicular location privacy, vehicles are united together to upload SBMs by the way of $k$-anonymity unity. This paper maps the group of vehicles into an undirected graph and quantifies the undirected graph with the parameter of that connectivity$\Delta$ and average distance$\overline{D}$.

The remainder of this paper is organized as follows. In Section 2, we review related works about the privacy protection ways and blockchain applications in VANET Section 3 describes some relative definitions and basic concepts. Then Section 4 shows system model, threat model and system interactions of our proposed architecture. Section 5 gives the detailed description of privacy protection algorithms and their performance analysis. The simulation environment and results are given in Section 6. Finally, Section 7 makes a conclusion.

## 2 Related work

The related work presents two aspects: the privacy protection ways and blockchain applications in VANET.

### 2.1 VANET and privacy protection

In recent years, privacy protection has become a hot spot in study of VANET [16, 17]. According to the objects protected by researchers, we can divide the privacy protection into three types: location privacy protection [18–22], trajectory privacy protection [23–25] and identity privacy protection [26]. For example, the work in [22] dealt with the distribution of vehicular pseudonyms using fog computing technology. The edge resources of VANET were used to effectively manage pseudonyms, which can improve the ability of location privacy protection. In [25], the authors proposed a policy of trajectory privacy using the multiple mix zones. By constantly changing pseudonyms, it made the pseudonyms unlinkable and protected vehicular trajectory privacy.

From the perspective of protection means, the privacy protection is mainly divided into two types: anonymity-based and encryption-based privacy protection. Anonymity-based privacy protection often adopts $k$-anonymity mechanism [27, 28]. K-anonymity mechanism was originally created to protect users' location privacy in LBS applications [29]. Nowadays, many researchers use $k$-anonymity mechanism in VANET. By the principle of maximum entropy, we find $k$ appropriate vehicles with the closest historical request probability. Then the real vehicle is hidden in these $k$ vehicles, and vehicular privacy is protected. While the encryption-based privacy protection [30] is commonly used in the authentication of VANET. Authentication is an important guarantee to receive SBMs from legitimate vehicles in VANET. It plays a vital role in privacy protection in VANET. At present the authentication of VANET generally adopts asymmetric encryption authentication methods, such as Public Key Infrastructure (PKI) and Elliptic Curve Digital Signature Algorithm (EDSA), and symmetric encryption authentication methods, such as group signature authentication mechanism [31].

### 2.2 Blockchain and VANET

Blockchain technology originated from an article titled "bitcoin: a peer-to-peer electronic cash system" written by Nakamoto in 2008. From the birth of blockchain to the present, blockchain technology has experienced three generations from 1.0 to 3.0 [32]. Blockchain 1.0 focuses on the transactions of digital currency. Since the birth of bitcoin, more than 600 cryptodigital currencies have been generated correspondingly (litecoin, dogecoin, YBcoin, etc.). These cryptodigital currencies mainly be used in small payments, foreign exchange, gambling and money laundering; Blockchain 2.0 focuses on the registration, validation, and transfer of smart contract. Smart contracts rely on oracles prediction and are mainly used in the financial field, such as securities trading, bank bills, payment and clearing, and supply chain finance. Blockchain 3.0 transcends the economic field, and its application field is not limited to finance, commodity transaction. The scope of blockchain 3.0 extends to government, medical care, science, culture, transportation, etc.

In recent years, a few researchers have introduced blockchain into VANET, considering with the characteristics of decentralization, redundant storage, collective maintenance and tamper-proof of blockchain. For example, Joy et al. [33] proposed the concept of blocktree. And the vehicle embedded its signature into the blockchain. Greg et al. [34] verified the feasibility of blocktree and analyzed the end-to-end delay and the time to collect, write and update blockchain contents. To better combine the blockchain with the VANET, Dorri et al. [35] proposed a lightweight scalable blockchain. And based on this, the reference of [36] proposed a blockchain architecture with decentralized privacy protection for intelligent vehicle systems. While Lei et al. [37] proposed a new network topology based on blockchain structure to simplify distributed key management in heterogeneous vehicle communication systems. Furthermore, the use of the blockchain as a means for privacy-preserving proof of location has been proposed in the work of [38].

## 3 Preliminaries

This section gives some basic concepts and definitions used in this paper: blockchain technology, dynamic threshold encryption, and $k$-anonymity unity.

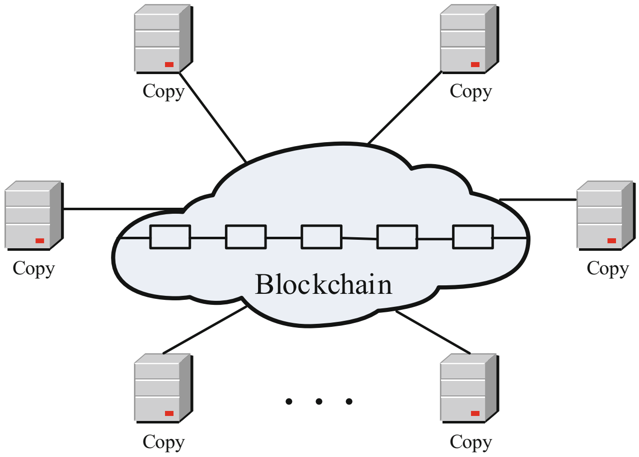


**Fig. 2** Blockchain system

### 3.1 Blockchain technology

Blockchain system adopts distributed structure, as shown in the Fig. 2. It consists of several decentralized nodes which cannot trust each other. Each node participates in data management. When a new block is confirmed by most of nodes (the number depends on different consensus mechanisms) in blockchain system, the new block is written into blockchain by miner. This is called as mining process. Then all nodes verify the content of that be newly added to blockchain, and have a complete copy of blockchain. Therefore, all nodes in the blockchain system jointly maintain the data information, which ensures the consistency, integrity and non-tampering of data.

In this paper, blockchain technology is applied to VANET. The distributed features of the blockchain are used to remove the single trust center in traditional VANET. In this way, a novel decentralized architecture of VANET is constructed to realize the security and irreversibility of SBMs. However, it is precisely because all nodes participate in the joint maintenance of data information in blockchain. SBMs of vehicles become open and transparent, which is not conducive to identity and location privacy protection. And the two concepts of information transparency and privacy protection are contradictory. The more transparent information is, the more difficult privacy protection is. Therefore, in this paper, we adopt dynamic threshold encryption and $k$-anonymity unity to achieve identity and location privacy protection respectively.

### 3.2 Dynamic threshold encryption

Dynamic threshold encryption was proposed by Amir Herzberg [39]. It is based on $(m, r)$ threshold secret sharing scheme. Thus, we present what $(m, r)$ threshold secret sharing scheme is before introducing dynamic threshold encryption. $(m, r)$ threshold secret sharing scheme first constructs a $m$-1 degree polynomial and takes secret $s$ as constant term of polynomial. Suppose a finite field is $GF(q)$, where $q$ is a large prime number. Thus, the $m$-1 degree polynomial is expressed as:

$$f(x) = s + a_1x + a_2x^2 + \ldots + a_ix^i + \ldots + a_{m-1}x^{m-1}, a_i \in GF(q) \tag{1}$$

Thus, we get that secret $s$ is $f(0)$ from formula (1). Select $r$ elements $\{x_1, x_2, x_i, \ldots, x_r\}$ form $GF(q)$ as the input of $m$-1 polynomial. Then we get $r$ values $\{f(x_1), f(x_2), \ldots, f(x_i), \ldots, f(x_r)\}$. As each $f(x_i)$ implies the information of secret $s$, $f(x_i)$ is called a sub-secret of $s$. These sub-secrets are distributed to $r$ different participants to keep. By Lagrange interpolation theorem, arbitrary $m$ points $\{(x_1, f(x_1)), (x_2, f(x_2)), \ldots, (x_m, f(x_m))\}$-can recover $f(x)$, as shown in formula (2):

$$f(x) = \sum_{i=1}^{m} f(x_i) \prod_{\substack{j=1 \\ j \neq i}}^{m} \frac{x - x_j}{x_i - x_j} \tag{2}$$

The secret $s$ can be obtained through $f(x)$:

$$s = f(0) = \sum_{i=1}^{m} f(x_i) \prod_{\substack{j=1 \\ j \neq i}}^{m} \frac{0 - x_j}{x_i - x_j} \tag{3}$$

Therefore, from the above analysis, it can be concluded that $(m, r)$ threshold secret sharing scheme is effective and secure as long as $m$ or more sub-secrets cannot be obtained. However, we cannot guarantee that an attacker can only obtain sub-secrets within $m$-1. Thus, Amir Herzberg et al. improved the $(m, r)$ threshold secret sharing scheme and made the sub-secret dynamic. The survival cycle of sub-secret is divided into different time periods. Then the sub-secret is updated at each time period. This way makes it difficult for an attacker to obtain $m$ or more sub-secrets in a short time. The update process of sub-secret is as follows.

First, we divide the survival cycle of sub-secret into $w$ periods $\{t_1, t_2, \ldots, t_w\}$. The sub-secret is assumed to be $f_{i-1}(x)$ at time $t_{i-1}$. Then the sub-secret will be updated to $f_i(x)$ at time $t_i$.

$$f_i(x) = f_{i-1}(x) + \delta_i(x) \tag{4}$$

$$\delta_i(x) = \sum_{w} \left(h_i(x/\bmod q)\right) \tag{5}$$

$$h_i(x) = a_{i1}x^1 + a_{i2}x^2 + \ldots + a_{i(m-1)}x^{m-1}, a_{ij} \in GF(q) \tag{6}$$

To keep secret $s$ invariable, the constant term of $\delta_i(x)$must be zero.

Before updating the sub-secret:

$$f_{i-1}(0) = s \tag{7}$$

After updating the sub-secret:

$$f_i(0) = f_{i-1}(0) + \delta_i(0) = f_{i-1}(0) + 0 = s \tag{8}$$

Therefore, the update of sub-secret does not affect the change of secret $s$. The updated sub-secret still conforms to Lagrange interpolation theorem, and the secret $s$ can be recovered by gathering $m$ or more sub-secrets.

In this paper, the vehicle's identity *id* is used as a secret. To protect *id*, it is divided into multiple sub-identities by threshold secret sharing scheme. When a vehicle uploads SBM, it joins other $k$-1 vehicles and collects their sub-identities as identity information using $k$-anonymity unity technology. Each transmission of SBM is regarded as a transaction, and the hash value of each transaction is recorded into the blockchain after being verified. If the verification fails, such as a vehicle intentionally falsifying sub-identity information, the corresponding vehicle will be removed from blockchain-based VANET.

Since the vehicular identity information is a combination of sub-identities of $k$ vehicles in each transaction, the attacker cannot infer the real sub-identity of the vehicle. Furthermore, the sub-identity of vehicle is updated dynamically with the vehicular driving trajectory. Therefore, it is more difficult for attacker to acquire $m$ or more valid sub-identities in a short time, and vehicular identity privacy can be protected.

## 3.3 *k*-anonymity unity

In this paper, to protect vehicular location privacy, the vehicle's SBM is uploaded with other $k$-1 vehicular SBMs by adopting the concept of $k$-anonymity. The format of $k$ united messages is shown in Table 1. From the Table 1, we can get that the format of $k$ united messages contains three basic types of information: sub-identity *id'*, location $l$, and request content $C$.

Therefore, it is difficult for an attacker to get vehicular real location information even if attacker steals SBMs of vehicles. The attacker guesses the true location information with the probability of at most $1/k$.

Generally, a first-come, first-served method is used to unite other $k$-1 vehicles. This method is easy to operate. However, there is not a standard to judge whether this unity is reasonable or not. To better protect the vehicular location privacy, this paper proposes two indicators to measure the effectiveness of unity.

**Table 1** The format of $k$ united messages

| |
|---|
| $id'_1, id'_2, \ldots, id'_k$ |
| $l_1, l_2, \ldots, l_k$ |
| $C_1, C_2, \ldots, C_k$ |

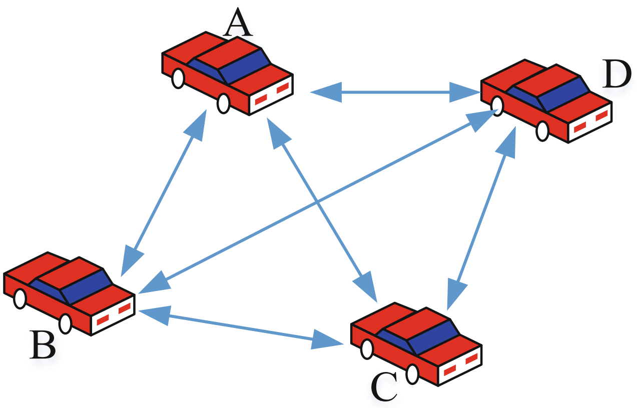


**Fig. 3** The united vehicles construct a complete graph

### 3.3.1 Connectivity Δ

To protect the vehicular location privacy, this unity is not valid when the united messages have the same information in each uploaded message of united vehicle. As shown in Fig. 3, assume that $k$ is 4 and the $k$ united vehicles are {A, B, C, D}.

The vehicles {A, B, C, D} are combined with each other. Then the united messages uploaded by each vehicle are the same, as shown in Table 2.

To analyze united model among vehicles, this paper proposes graph analysis method. The vehicles are regarded as the nodes of graph. If two vehicles are united with each other, there exists an edge between the corresponding nodes. Then we can construct an undirected graph for united vehicles. From Fig. 3, we can conclude that the undirected graph cannot be a complete graph, otherwise the united message is same in each vehicle. As this paper adopts the method of $k$-anonymity unity, the number of united vehicles must be larger than $k$ to ensure that the graph is not a complete graph, as show in Fig. 4. We also assume that $k$ is 4. The united vehicles are {A, B, C, D, E, F}. And the number of united vehicles in Fig. 4 is $n = 6 > k$.

Then the united message uploaded by each vehicle is different in Fig. 4. Table 3 presents the united messages of vehicles {A, B, C, D, E, F}.

This paper uses variable$\varDelta$to measure the connectivity of a vehicle $V$.

$$\varDelta = \frac{num.(V)}{n} \geq \frac{k}{n} \tag{9}$$

Where $n$ represents the total number of united vehicles in graph, and $num.(V)$ is the number of united vehicles with the

**Table 2** The same information for each vehicle

| |
|---|
| $id'_A, id'_B, id'_C, id'_D$ |
| $l_A, l_B, l_C, l_D$ |
| $C_A, C_B, C_C, C_D$ |

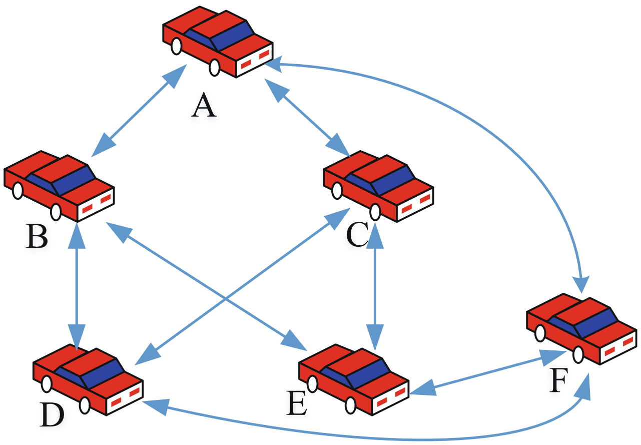


**Fig. 4** The united vehicles construct an uncomplete graph

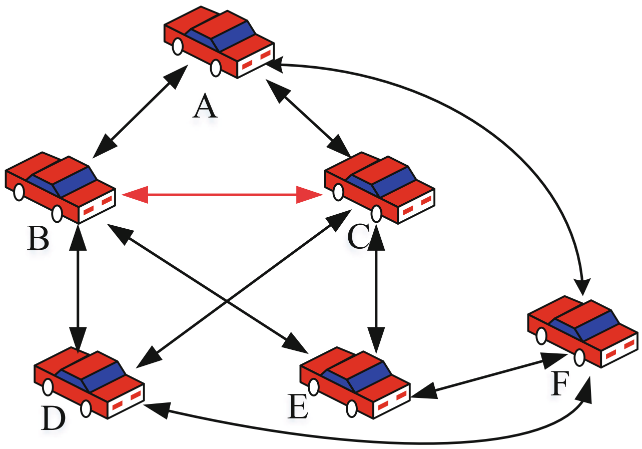


**Fig. 5** Certain vehicles construct a sub complete graph

target vehicle *V*. To realize *k*-anonymity, the *num*. (*V*) is more than *k*. If the constructed graph is an unconnected, the split subgraph must not be complete. Otherwise there will be a large number of vehicles that upload the same messages. If the constructed graph is connected, certain nodes can form sub complete graph. For example, if vehicle B and vehicle C unite with each other in Fig. 4, the vehicles {A, B, C}, {D, B, C}, {E, B, C} separately constitute a sub complete graph as show in Fig. 5. However, the connectivity may be different for the vehicles in sub complete graph.

The connectivity of vehicle A is calculated as:

$$\Delta_A = \frac{num.(A)}{n} = \frac{4}{6} \tag{10}$$

While the connectivity of vehicle B is represented by the symbol $\Delta_B$:

$$\Delta_B = \frac{num.(B)}{n} = \frac{5}{6} \tag{11}$$

Thus, the connectivity of vehicle B is greater than vehicle A.

$$\Delta_B > \Delta_A \geq \frac{k}{n} = \frac{4}{6} \tag{12}$$

**Table 3** The different information for each vehicle

| Vehicle A | Vehicle B | Vehicle C |
|---|---|---|
| $id'_{\mathrm{A}}$, $id'_{\mathrm{B}}$, $id'_{\mathrm{C}}$, $id'_{\mathrm{F}}$ | $id'_{\mathrm{A}}$, $id'_{\mathrm{B}}$, $id'_{\mathrm{D}}$, $id'_{\mathrm{E}}$ | $id'_{\mathrm{A}}$, $id'_{\mathrm{D}}$, $id'_{\mathrm{E}}$, $id'_{\mathrm{C}}$ |
| $l_{\mathrm{A}}$, $l_{\mathrm{B}}$, $l_C$, $l_F$ | $l_{\mathrm{A}}$, $l_{\mathrm{B}}$, $l_D$, $l_E$ | $l_{\mathrm{A}}$, $l_D$, $l_E$, $l_C$ |
| $C_{\mathrm{A}}$, $C_{\mathrm{B}}$, $C_{\mathrm{C}}$, $C_{\mathrm{F}}$ | $C_{\mathrm{A}}$, $C_{\mathrm{B}}$, $C_{\mathrm{D}}$, $C_{\mathrm{E}}$ | $C_{\mathrm{A}}$, $C_{\mathrm{D}}$, $C_{\mathrm{E}}$, $C_{\mathrm{C}}$ |
| Vehicle D | Vehicle E | Vehicle F |
| $id'_{\mathrm{D}}$, $id'_{\mathrm{B}}$, $id'_{\mathrm{C}}$, $id'_{\mathrm{F}}$ | $id'_{\mathrm{E}}$, $id'_{\mathrm{B}}$, $id'_{\mathrm{C}}$, $id'_{\mathrm{F}}$ | $id'_{\mathrm{A}}$, $id'_{\mathrm{E}}$, $id'_{\mathrm{D}}$, $id'_{\mathrm{F}}$ |
| $L_{\mathrm{D}}$, $l_{\mathrm{B}}$, $l_C$, $l_F$ | $L_{\mathrm{E}}$, $l_{\mathrm{B}}$, $l_C$, $l_F$ | $l_{\mathrm{A}}$, $l_{\mathrm{E}}$, $l_D$, $l_F$ |
| $C_{\mathrm{D}}$, $C_{\mathrm{B}}$, $C_{\mathrm{C}}$, $C_{\mathrm{F}}$ | $C_{\mathrm{E}}$, $C_{\mathrm{B}}$, $C_{\mathrm{C}}$, $C_{\mathrm{F}}$ | $C_{\mathrm{A}}$, $C_{\mathrm{E}}$, $C_{\mathrm{D}}$, $C_{\mathrm{F}}$ |

For the connectivity of a graph, it consists of the connectivity of all vehicles in the graph. And the greater the connectivity of the graph indicates the greater the similarity of the united messages for vehicles. Thus, the higher connectivity of a graph corresponds to better identity privacy protection for vehicles. This paper adopts average connectivity $\overline{\Delta}$ of all vehicles to represent the effect of identity privacy protection for a graph. Then the average connectivity is shown as follows:

$$\overline{\Delta} = \frac{\Delta_1 + \Delta_2 + \ldots + \Delta_n}{n} < 1 \tag{13}$$

When the graph constructed by united vehicles is a complete graph, the average connectivity is up to the maximum value of 1.

### 3.3.2 Average Distance $\overline{D}$

To protect location privacy, the unity is not valid when the locations of united vehicles are adjacent, or even the same. In this paper, the location is represented by (*x*, *y*), where *x* and *y* are horizontal and vertical coordinates respectively. The average distance between vehicles in a graph is defined as follows:

$$\overline{D} = \frac{\sum_{i \neq j} \sqrt{\left|x_i - x_j\right|^2 + \left|y_i - y_j\right|^2}}{C_n^2} \tag{14}$$

Where $(x_i, y_i)$ and $(x_j, y_j)$ respectively represent location information of any two vehicles in a graph. For the *n* vehicles in a graph, there are $C_n^2$ Euclidean distances between the *n* vehicles. Thus, the $\overline{D}$ describes the average distance of the $C_n^2$ Euclidean distances. And we can get that the larger the average distance $\overline{D}$ between vehicles, the better the effect of location privacy protection, and the more the effectiveness of *k*-anonymity unity. To prevent vehicles from approaching or being in the same location, a threshold value $\sigma$ is set in this paper. The $k$-anonymity unity is effective when $\overline{D} \geq \sigma$.

## 4 System model

We design an architecture of blockchain-based VANET in this section. Then we present components and interactions in blockchain-based VANET.

### 4.1 System model

For preserving the vehicular identity and location privacy, we adopt the construction approach of IoTchain architecture [40] to design a decentralized architecture of VANET, as shown in Fig. 6. We call it blockchain-based VANET, involving eight different components: vehicles, on board unit (OBU), road side unit (RSU), core network, blockchain network, smart contract, agent node, and miner.

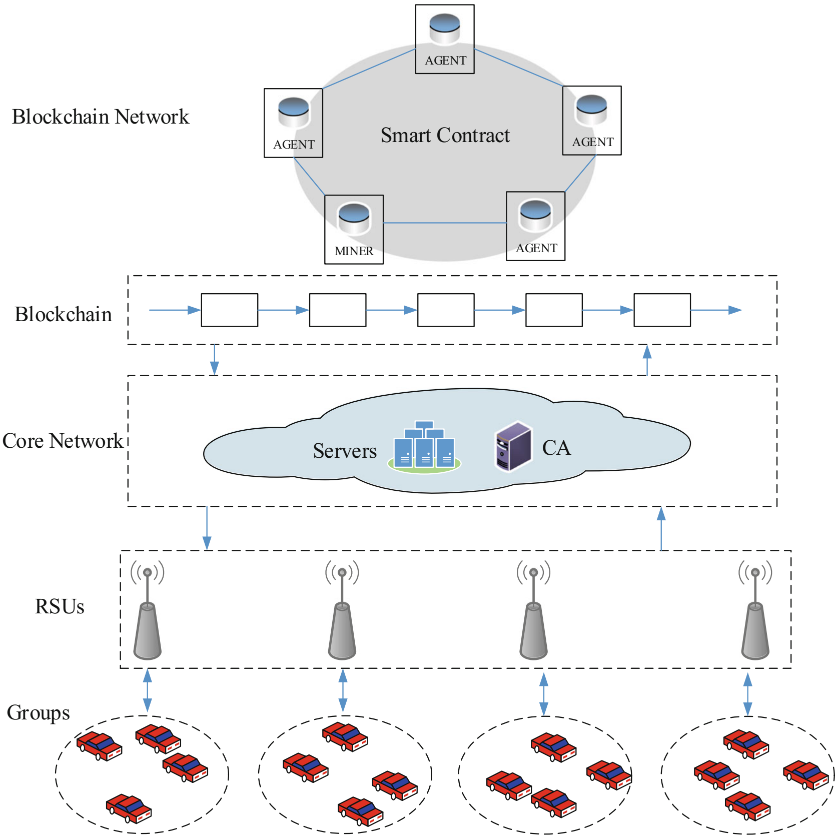
1) *Vehicles*: Vehicles are the moving entities in the blockchain-based VANET. They are equipped with an OBU to communicate with each other. When a vehicle uploads SBMs, the vehicle will unite with other vehicles.

2) *OBU*: OBU is a kind of sensor device mounted on vehicle. The OBU can sense the environment, speed and location information of vehicle. Moreover, the OBU can communicate with the adjacent RSU, and send SBMs to RSU.

3) *RSU*: RSU is deployed on the side of road, acting as an AP (Access Point). It collects SBMs from OBUs, and then upload these BSMs to the core network via wire or wireless. RSU can communicate with OBU in real time, and assist vehicles to from groups. At the same time, RSU is responsible for transmitting traffic and service information to vehicles through broadcasting, such as weather conditions, real-time road conditions, emergencies, control information, service facilities, etc.

4) *Core network*: It is composed of a large number of servers, such as CA server, data storage server, etc., with strong computing capacity, storage resources, and energy availability. All data is stored in the core network and processed by the core network. For the data security, these data stored in the core network is encrypted. In particular, it is pointed out that the CA proposed in this paper is a kind of weakened server compared with the traditional VANET. As the CA is

**Fig. 6** The decentralized architecture of VANET



just to generate a few parameters for dynamic threshold encryption, the CA server can be unreliable.

5) *Blockchain network*: Considering the privacy protection in the blockchain-based VANET, all participating nodes need to be strictly controlled. Therefore, this paper adopts private chain to build blockchain network. The hash values of all data in the core network are stored in blockchain network. This can guarantee data not to be tampered and manipulated by malicious attackers. Because, once the data has been changed, its hash value will be also changed. Blockchain is auditable so that the change will be discovered. Furthermore, storing its hash values can greatly save the storage resources of the blockchain network and prove the system response time.

6) *Smart contracts*: Smart contracts are pre-defined rules and terms that can be executed automatically. In this paper, we pre-write certain rules into smart contracts, such as authentication rules, *k*-anonymity unity rules, sub-identity generation rules, sub-identity dynamic change rules, SBMs recording rules, etc. Because smart contracts are defined entirely in code, they are automatically executed once the trigger condition is met, without human intervention. Therefore, smart contracts not only save transaction costs, but also improve accuracy and efficiency.

7) *Agent node*: Agent node is a participating node of the blockchain network. Each agent node should participate in the consensus and have the backup of data of blockchain network, so as to jointly maintain the correctness of transactions of the blockchain network. In this paper, agent nodes adopt the way of Proof of Work (PoW) for consensus.

8) *Miner*: If an agent node has solved the mathematical problem and has the rights of legal block keeping, this agent node is called miners. Miner involves in processing new block of blockchain, and writes verified data into new block. Then all agent nodes will update their backup of data in blockchain network. Thus, the miner node is a special agent node. It not only participates in the consensus of blockchain network but also is able to mine and validate the new blocks.

## 4.2 System interactions

The interactions of blockchain-based VANET is presented in Fig. 7. Here, we explain four mainly interactions between the different components in blockchain-based VANET.

1) *Blockchain set-up*: During this stage, the system is initialized. All agent nodes form blockchain network. Each agent node is equal and enjoys the same rights and obligations. Meanwhile, corresponding smart contract rules will be built in this stage. Agent node adds these rules of authentication, *k*-anonymity unity, sub-identity generation, sub-identity dynamic change, and SBMs recording to smart contract one by one. If rule is successfully formed, agent node will receive corresponding address from smart contract.

2) *Registration of vehicles*: Assume that a vehicle wants to enjoy the blockchain-based VANET. The vehicle first initiates a register request to CA server. Then the CA server will send the register request to smart contract. By the rules of authentication, smart contract validates its validity, and returns an address to the vehicle if the verification is successful. Otherwise, the unverified register messages will be recorded in the blockchain and broadcasted to blockchain network, making the blockchain-based VANET aware that the vehicle is invalid and illegal. After receiving the address, the vehicle constructs a *m*-1 degree polynomial $f(x) = id + a_1x + a_2x^2 + \ldots\ldots + a_{m-1}x^{m-1}$, and hides his/her identity *id* in the polynomial by $id = f(0)$. Final, the vehicle calculates *r* sub-identities based parameter information provided by CA server.

3) *SBMs upload*: For protecting vehicular identity and location privacy, *k*-anonymity unity is adopted in this paper. Assume that a vehicle wants to upload SBMs. The vehicle first forms a group automatically with other vehicles by the assistance of RSU. When the vehicle initiates request to upload SBMs, it selects other *k*-1 sub-identities in the group. Then the *k*-1 sub-identities and its own sub-identities are together constituted vehicle's identity information. Then the vehicle uploads *k* merged messages to core network using the constituted identity. From the merged information, we can get that it contains *k* sub-identities, *k* locations and *k* request contents. Therefore, the servers of the core network cannot infer the relationship between the vehicle and its real location and identity, so as to protect identity and location privacy of the vehicle.

4) *Blockchain record*: During this stage, we proceed according to the period *T*. In a period *T*, it completes the process of blockchain record including three phases: new transaction creation, transaction verification for miner and backup for agent nodes. Here, assume that the processing of each message is a transaction, and the hash value of messages of in the core network is recorded into blockchain network. Thus, the hash of new transactions in a period *T* are received by agent node.

New transaction creation: Agent node broadcasts the newly generated transaction data to all nodes in the blockchain network. Each node stores the transaction data in a new block.

Transaction verification: As our proposed blockchain-based VANET does not involve tokens, this paper selects the

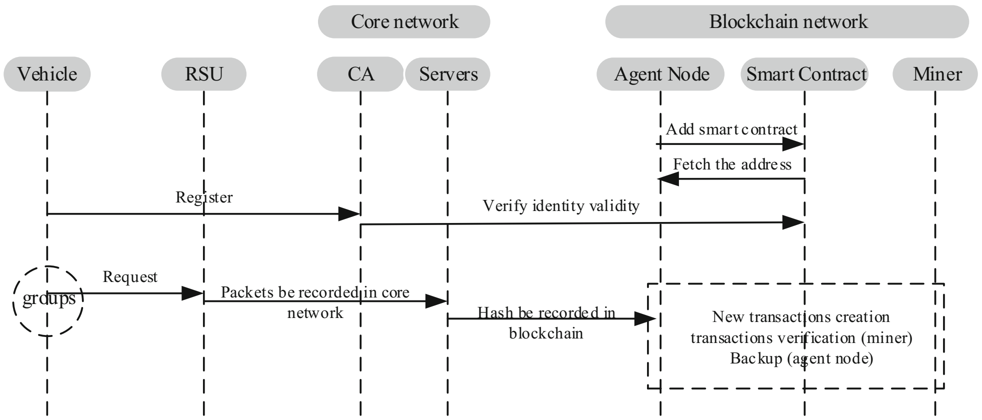


**Fig. 7:** The interactions of blockchain-based VANET.

consensus of Practical Byzantine Fault Tolerance to realize transaction verification. Each node participates in voting based on its own calculation force. When a node (that is miner) finds a proof of new block, it broadcasts the block to all nodes in the blockchain network.

Backup: The validity of the block is recognized by other nodes only if all transactions contained in the block are valid and have not existed before. And all nodes will make backup for the new block and update block chain.

### 4.3 Threat model

The CA and other servers are a semi-trusted entity. They can faithfully performs the work with other entities while probably being curious about vehicles' privacy. They may be compromised, and leak certain sensitive information for profit. Thus, although the blockchain technology ensures that the vehicles' SBMs are not tampered, the identity and location privacy are at risk of being compromised.

Identity privacy attack is one of the typical attack modes in vehicle registration process. When a vehicle wants to enjoy the blockchain-based VANET, the vehicle communicates with CA to get its legal identity information. Thus, the CA controls all identities of vehicles. The identity privacy is easy to be revealed.

Location privacy attack describes the fact that the $k$-anonymity unity is invalid, and an adversary could infer the location information of a target vehicle.

Therefore, for protecting identity and location privacy, we design corresponding privacy protection algorithms in the process of *Registration of vehicles* and *SBMs upload*. These algorithms will be introduced in detail in Section 5, including Undirected Graph Generation (UGG) algorithm, Identity Privacy Protection (IPP) algorithm based on dynamic threshold encryption, and Location Privacy Protection (LPP) algorithm based on $k$-anonymity unity.

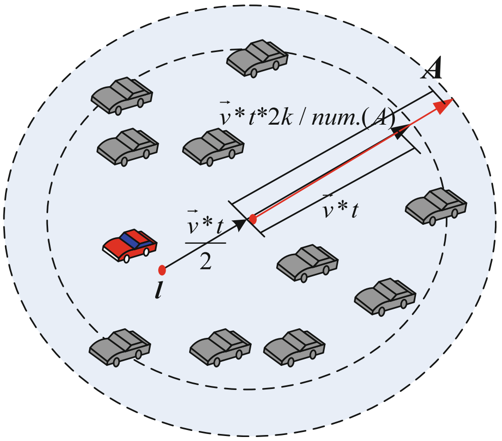


**Fig. 8** The generative process of undirected graph

## 5 Algorithm design

In this section, we detailedly introduce the three algorithms for identity and location privacy protection in blockchain-based VANET.

### 5.1 The undirected graph generation algorithm

To protect the privacy of vehicles, groups are built among vehicles with the assistance of RSUs. In this paper, we map a group into an undirected graph to quantitatively describe the group. Although the undirected graph is a static representation of the range of vehicles group at a given time, the generative process of undirected graph takes into that vehicles move with different velocities, as show the Fig. 8. The undirected graph generation is based on the location, speed and direction of the target vehicle.

*Algorithm* 1 presents how to generate the undirected graph $G$. The input parameters of *algorithm* 1 are: target vehicle U, location $l$, speed $v$, time $t$, and parameters $(m, r)$ for threshold secret sharing scheme. When a vehicle U initiates a request, the undirected graph $G$ must be completed within time $t$. Otherwise, vehicle U reinitiates request and the previous request is invalid.

Here, we describe the generative process of undirected graph $G$. First, we initialize the range of a group is $A\left(\overrightarrow{l}+\overrightarrow{v}*t/2,\overrightarrow{v}*t\right)$ basing on the location and speed of vehicle U, where $\overrightarrow{l}+\overrightarrow{v}*t/2$ is the center and $\overrightarrow{v}*t$ is radius. Using this vector pattern to construct the group can prevent attacker from guessing vehicle U is in the center of $A$. And it can make sure that vehicle U is in the group within time $t$.

Then we calculate the number of vehicles in range $A$, represented by *num.* ($A$). When *num.* $(A) > 2r$, the undirected graph $G$ will be built as shown at line 5–15 in *algorithm* 1. We initialize the $G$ and select vehicle U as the initial vertex of $G$. An array *visite*[] is set to store the vehicles in $G$. Then traverse each vehicle $U_i$ in *visite*[]. If new vehicle $U_j$ is united with $U_i$, the vehicle $U_j$ will be store in *visite*[]. We update the *visite*[] and continue to traverse until no new vehicles join *visit*[]. Thus, the undirected graph $G$ can be built based *visit*[]. If $G$ is not a complete graph, the $G$ is successfully constructed and the average connectivity of $G$ is calculated. Otherwise, we will change the size of the group range and modify the radius of range $A$ from $\overrightarrow{v}*t$ to $\overrightarrow{v}*t*2r/num.(A)$.

*Algorithm* 1 describes the pseudo code of UGG algorithm.

```
Algorithm 1: UGG Algorithm
Input: U, $l$, $v$, $t$, $(m, r)$
Output: undirected graph $G$
1: Initiates a request for vehicle U;
2: while ($t$);
3:   Initialize $A(\vec{l}+\frac{\vec{v}*t}{2},\vec{v}*t)$;
4:   for (num. ($A$) > $2r$)
5:     Initialize $G = \{U\}$;
6:     Set visit[];
7:     U -> visit[];
8:      for ($U_i \in$ visit[])
9:          if (new vehicle $U_j$ is united with $U_i$ && $U_j \notin$ visit[]);
10:          $U_j$ -> visite[];
11:         else
12:             Break;
13:          end if
14:     end for
15:     Construct $G$ by visit[];
16:      if ($G \notin$ complete graph)
17:          Calculate $\bar{\Delta}$;
18:           Return $G$;
19:      else
20:          Change $\vec{v}*t$ -> $\vec{v}*t*2r$/num. ($A$);
21:      end if
22:      Get new $A$;
23:  end for
24: end while
```

## 5.2 The identity privacy protection algorithm

For protecting vehicular identity privacy, this paper adopts the way of dynamic ($m$, $r$) threshold encryption. A target vehicle U uploads SBMs by using the united sub-identities of $r$ vehicles. Of course, these $r$ vehicles must contain the vehicle U. To make it difficult for attacker to acquire at most $m$-1 sub-identities of vehicle U and calculate real identity of vehicle U, the sub-identity is updated in two forms, as shown the two for-loops in *algorithm* 2. Consider the vehicle U is on a driving trajectory $Tr = \{l_1,\ldots, l_d\}$. In the first for-loop (at line 1), we generate an undirected graph $G_i$ for each location $l_i$ by calling UGG algorithm. When the vehicle U enters a different group, it will regenerate $r$ sub-identities with the assistance of the CA server (at line 3–5). As the CA server is responsible for generating certain parameter information $f(x)$ and $GF(q)$ for sub-identity. The calculation of the specific sub-identity is the vehicle itself. Therefore, the CA cannot gain vehicular identity information. In the second for-loop (at line 7), the vehicle U will randomly update certain sub-identity with the frequency $f$. That is to say, $f$ sub-identities of the $r$ sub-identities will be updated in unit time $R/v$, where $R$ is the moving distance and $v$ is the speed in group $G_i$ for vehicle U.

*Algorithm* 2 describes the pseudo code of UGG algorithm.

```
Algorithm 2: IPP Algorithm
Input: $Tr = \{l_1,\ldots, l_d\}$
Output: id'
1: for ($Tr$);
2:    Get $G_i$ = call UGG algorithm;
3:    Generate polynomial $f(x)$ and $GF(q)$;
4:     Select $\{x_1, x_2, x_i,\ldots, x_r\} \in GF(q)$;
5:     Calculate $r$ sub-identities;
6:      Broadcast $r$ sub-identities in $G_i$;
7:      for ($G_i$);
8:           Calculate update frequency $f$;
9:           Update id';
10:     end for
11: end for
```

## 5.3 The location privacy protection algorithm

To protect location privacy, this paper adopts the method of $k$-anonymity unity. When a vehicle uploads SBM, the vehicle unites other $k$-1 vehicles and uploads $k$ SBMs, which contains $k$ locations. Thus, it can blur the link between the vehicle and its location. And attacker cannot get which location belongs to which vehicle. The purpose of *algorithm* 3 is to gain the appropriate $k$ united vehicles set $M$. The average distance $\overline{D}$ of the $k$ vehicles must be greater than threshold value $\sigma$, preventing the $k$ locations is too adjacent or ever the same.

First, it is assumed that a target vehicle U receives $r$-1 sub-identities form other vehicles in group $G$. We store the corresponding $r$-1 vehicles and vehicle U into an alternative set $W$. Select $k$ vehicles from the set $W$, including vehicle U. Then we initialize the set $M$ and store the $k$ vehicles in $M$. We calculate the distance $D$ between each of two vehicles in set $M$. And the number of distance is $k(k$-1)/2. Then we sort these $k(k$-1)/2 distances from small to large, and calculate the average distance $\overline{D}$ of $k$ vehicles. If $\overline{D}\geq\sigma$, we succeed in finding the appropriate $k$ vehicles. Otherwise, we replace the two vehicles $\{u_i, u_j\}$ with the minimum distance. If the vehicle U belongs to one of the vehicles $\{u_i, u_j\}$ (assume that $U = u_i$), we select another vehicle from the set $W$ to replace vehicle $u_j$. If vehicle $U \notin \{u_i, u_j\}$, we select two vehicles from $W$ to replace the two vehicles $\{u_i, u_j\}$. Then the set $M$ is updated, and we calculate average distance until the appropriate $k$ vehicles are found.

*Algorithm* 3 describes the pseudo code of LPP algorithm.

***Algorithm* 3: LPP Algorithm**

```
Input: U, (m, r), l, k, G, t
Output: k united vehicles set M
for (t);
    Vehicle U receives other r-1 id';
    Get set W = {U, r-1 vehicles};
end for
for (W);
    Randomly select k vehicles including vehicle U;
    Initialize set M;
    for (k(k-1)/2--)
        Calculate distance D for $\forall$ two vehicles $\in M$ ;
    end for
    Sort k(k-1)/2 distances from small to large;
    Calculate $\overline{D}$;
    if $\overline{D} \geq \sigma$
        return M;
    else
        Select two vehicles {$u_i$, $u_j$} with the minimum distance from M;
        if ($U \in \{u_i, u_j\}$)//assume that U = $u_i$
            Replace vehicle $u_j$;
        else
            Replace {$u_i$, $u_j$};
        end if
        Get new M;
    end if
end for
```

## 5.4 Performance analysis

We analyze the performance of our proposed UGG, IPP and LPP algorithms in terms of the effectiveness of vehicle unity and privacy.

Vehicle unity: We first analyze the connectivity of vehicle unity. To get a high average connectivity $\overline{\Delta}$, the range of graph $G$ ensures *num*. $(A) > 2r$ in UGG algorithm. If the *num*. $(A) \leq 2r$, the center of $A$ will be changed with the vehicle's speed and direction. Moreover, the radius of $A$ is also changed with the vehicle density. Thus, the vehicle's location, speed, direction and density are taken into account for vehicle unity. Second, we analyze the average distance $\overline{D}$ of vehicle unity. By setting a threshold value $\sigma$, it ensure that the vehicles in a graph G are not lies in the same location. Thus, this paper guarantees the effectiveness of vehicle unity from average connectivity $\overline{\Delta}$ and average distance $\overline{D}$.

Privacy: We first analyze the identity privacy protection for the IPP algorithm. Using the threshold secret sharing scheme, the identity of vehicle is divided into $r$

**Table 4** simulation parameters

| Parameter | Setting |
|---|---|
| Vehicle Speed | 30 km/h |
| Heavy traffic | 0.4 v/h |
| Light traffic | 0.15 v/h |
| $k$ | 8 |
| $(m, r)$ | (16, 12) |
| $\sigma$ | 600 m |
| $Tr$ | $\{l_1, l_2, \ldots, l_{10}\}$ |

sub-identities. Thus, the vehicle communicates with the CA using the sub-identities so that the CA cannot easily gain complete identity of vehicle. Furthermore, the sub-identities are changed through two different ways.

1) Between groups: when a vehicle enters a new group, all the previous $r$ sub-identities are discarded, and the $r$ sub-identities are generated again with the assistance of CA.
2) Inside the group: As the vehicle moves within the group, it randomly updates certain sub-identities with frequency $f$.

Therefore, it is difficult for an attacker to collect $m$ sub-identity within a short period of time, thus unable to decrypt the real identity of the vehicle. Thus, the identity privacy can be effectively protected.

Second, we analyze the location privacy protection for the LPP algorithm. Using the method of $k$-anonymity unity, the vehicle real location is hide in $k$ locations. Moreover, the $k$ locations can hardly be identical as the average distance between vehicles must be greater than the threshold value $\sigma$. Thus, an adversary could not infer the location of vehicle, and the location information is effectively protected.

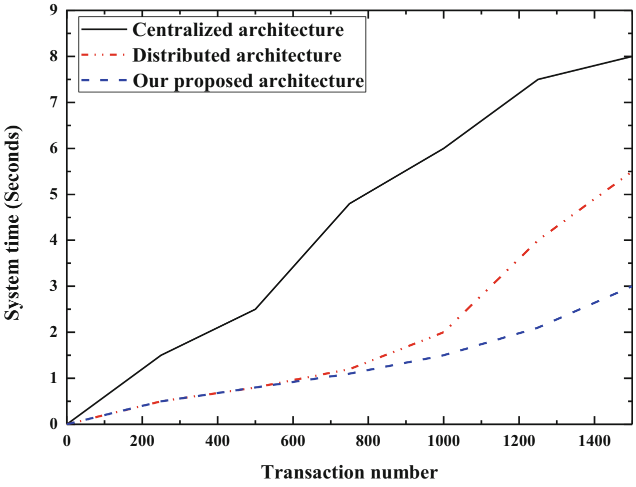


**Fig. 9** System time

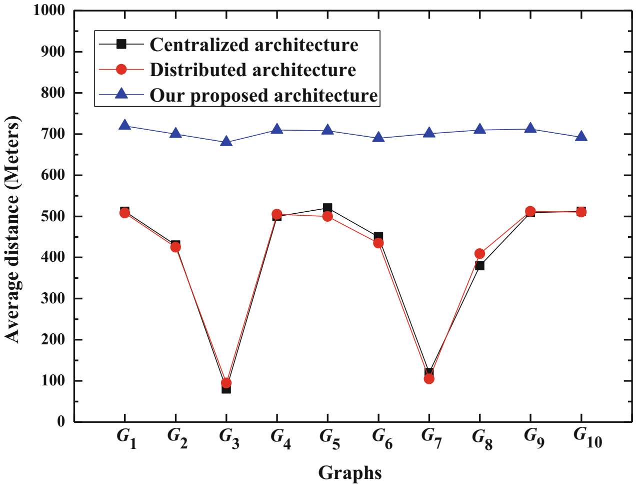


**Fig. 10** Average distance for heavy traffic scenario

## 6 Simulation and results

The performance evaluation of our proposed architecture (the blockchain-based VANET) is conducted in this section. The first part describes simulation environment. And the last part analyzes simulation results using four metrics: system time, average distance, connectivity, and privacy leakage.

To identify the effectiveness of our proposed architecture, we compare it with centralized architecture [7] and distributed architecture [37]. The biggest difference between the distributed architecture and our proposed architecture lies in the story data. In blockchain network of distributed architecture, it directly storages the SBMs data. While in the blockchain network of our proposed architecture, we storage the hash values of the SBMs data.

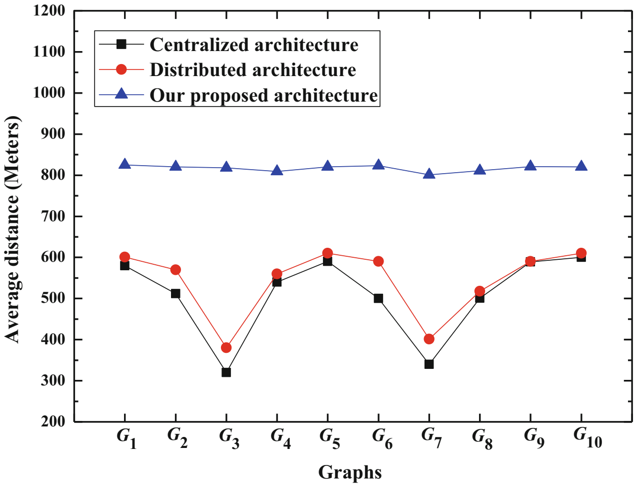


**Fig. 11** Average distance for light traffic scenario

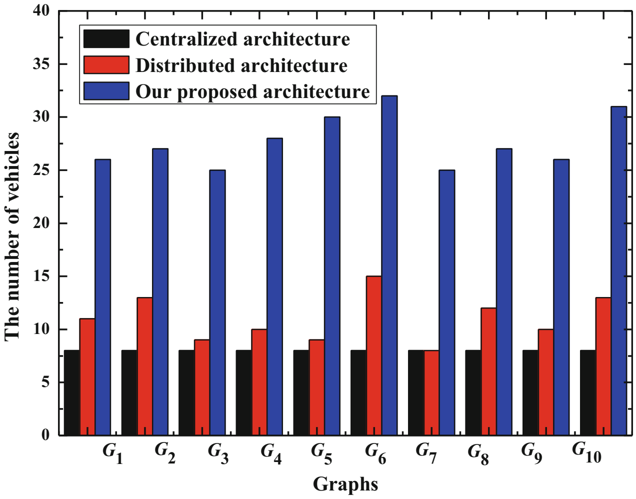


**Fig. 12** The number of vehicles

## 6.1 Simulation Environment

To simplify simulation experiment, we regard our proposed architecture as two parts: VANET network and blockchain network. The VANET network mainly do that vehicles upload SBMs and the blockchain network is responsible for the transaction record. We use OPNET [41] and Ethereum [42] to simulate the VANET network and blockchain network respectively. Ethereum is the blockchain-based computing platform, which provides the power of smarts contracts and the PoW consensus. Thus, in our experiment, we use Ethereum platform to write rules (e.g. authentication rule, *k*-anonymity unity rule, sub-identity generation rule, sub-identity dynamic change rule, SBMs recording rule) into smart contracts. For blockchain network, using the PoW consensus to verify the new data block.

In our experiment, we simulate that a target vehicle U drives on a trajectory *Tr*, which including 10 locations $\{l_1, l_2, \ldots, l_{10}\}$. For privacy protection, a corresponding undirected graph will be generated for each location. Thus, there exists 10 undirected graphs $\{G_1, G_2, \ldots, G_{10}\}$ for the vehicle U. To describe the performance accurately, we do the experiments for 8 times and calculate the average value including system time, average distance, connectivity, and privacy leakage.

Furthermore, to better present the performance of average distance $\overline{D}$, we simulate two scenarios: light and heavy traffic. We set the heavy traffic scenario as 0.4 v/h (that is, 4000 vehicles pass the *Tr* in one hour), while the light traffic scenario as 0.15 v/h. All the simulation parameters is shown in Table 4 in detail.

## 6.2 Simulation Results

**System time** For our proposed architecture and the distributed architecture [37], the system time mainly contains two parts: the blockchain processing time and SBMs processing time. However, the centralized architecture [7] only has the SBMs processing time.

Figure 9 plots the system time in terms of transactions number under different architectures. Transactions ranging from 1 to 1500 is set for testing the system time. The system time increases with the increase in the number of transactions. Although the system time of the centralized architecture has no blockchain processing time, it is the worst performance in the aspect of system time. Because the centralized architecture has many central entities, these central entities must wait for *k* SBMs for centralized process transactions. The process of waiting is time consuming. Comparing our proposed architecture with the centralized architecture, the system time of our proposed architecture is smaller triple than centralized architecture.

Comparing our proposed architecture with the distributed architecture, we can get that our proposed architecture and the distributed architecture have the same system time when transactions number is smaller than 600. However, our

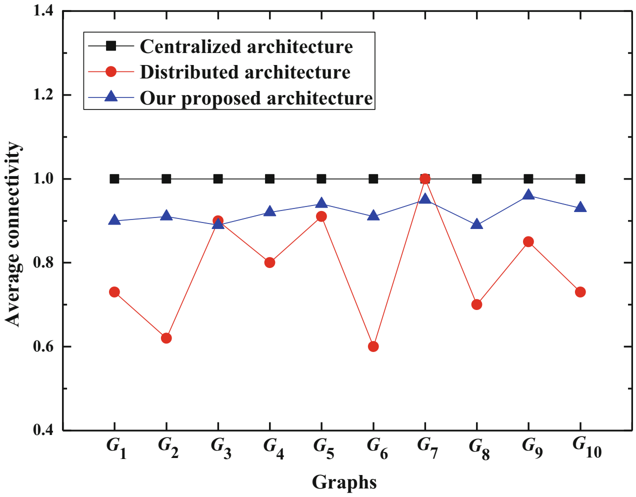


**Fig. 13** The average connectivity

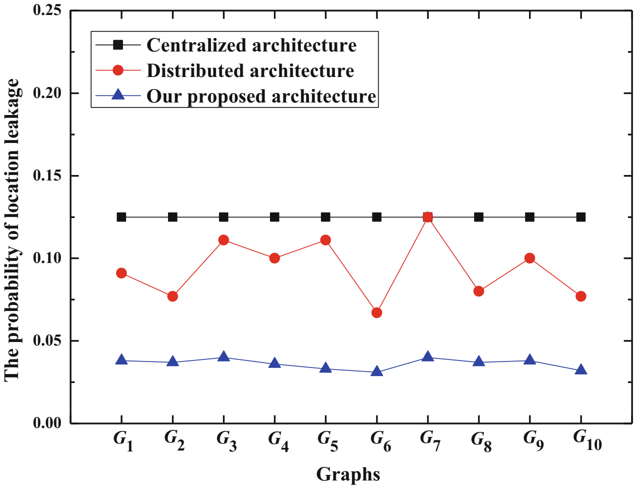


**Fig. 14** The probability of location leakage

proposed architecture have faster system time than distributed architecture when transactions number is larger than 600. The main reason is that we store the hash value of transactions to blockchain network in our proposed architecture. Thus the bigger the transactions number is, the more time will be saved.

In conclusion, our proposed architecture reduces the system time by using the blockchain technology.

**Average distance** Figure 10 and Fig. 11 respectively depicts the average distance $\overline{D}$ under the heavy and light traffic scenarios. Although the shape of average distance $\overline{D}$ in terms of the undirected graphs is very similar, there exists a difference about 100 m for average distance between heavy and light traffic scenarios. The average distance $\overline{D}$ is about 700 m in heavy traffic scenario, and 800 m in light traffic scenario. The average distance $\overline{D}$ in our proposed architecture is greater than both the centralized and distributed architectures. Thus, it's harder to get the same or adjacent locations in our proposed architecture. And our proposed architecture works well for location privacy protection.

In Fig. 10 and Fig. 11, there exist two lows at $G_3$ and $G_7$ in centralized and distributed architecture. We can get that the corresponding locations $\{l_3, l_7\}$ may be crossroads. Because vehicles easily gather at crossroads, leading to a sharp increase in vehicles number. Thus, the average distance $\overline{D}$ will drop dramatically at locations $\{l_3, l_7\}$. However, the line of average distance is smooth in our proposed architecture. Thus, whether in heavy or light traffic scenario, our proposed architecture is more stable and cannot be affected by the crossroads.

Figure 12 plots the number of vehicles at graphs $\{G_1, G_2,\ldots, G_{10}\}$. From the Fig. 12, we can get that the number of vehicles is greater than $2r$ ($2r = 2*12 = 24$). However, the number of vehicles is between 8 to 15 in distributed architecture, and 8 in centralized architecture. To get $k = 8$ appropriate united vehicles, the greater number of vehicles means better privacy protection. Thus, our proposed architecture shows its superior.

Furthermore, we validate the effectiveness of the privacy protection in terms of average connectivity $\overline{\varDelta}$ and the probability of location leakage, as shown in Fig. 13 and Fig. 14, respectively.

As shown in Fig. 13, compared with centralized architecture, the average connectivity $\overline{\varDelta}$ is about 0.9 in our proposed architecture, which is only a little smaller than in centralized architecture. Compared with distributed architecture, the average connectivity in our proposed architecture is greater than distributed architecture. $\overline{\varDelta} = 1$ means that the corresponding graph is complete graph and is not conducive to identity privacy protection. When $\overline{\varDelta}$ is not equal to 1, the bigger the average connectivity is, the better privacy protection is. Thus, the average connectivity $\overline{\varDelta}\approx 0.9$ presents that our proposed architecture is superior to that of the centralized architecture and distributed architecture.

As shown in Fig. 14, for each graph, the probability of location leakage is 0.125 using centralized architecture and 0.075–0.0125 using distributed architecture. Whereas our proposed architecture can reduce the probability to about 0.03%. Therefore, the ability of location privacy protection has been greatly improved using our proposed architecture.

## 7 Conclusion

This paper has studied the decentralized architecture using blockchain technology in VANET and designs a system model of blockchain-based VANET including blockchain set-up, registration of vehicles, SBMs upload, and blockchain record. Using the blockchain-based VANET can effectively address the problems of centralization and distrust among the entities in VANET. There is no third central entity in our proposed system model. The hash of SBMs is recorded in blockchain, which can ensure the integrity of SBMs and increase data processing time. Then to protect vehicular identity privacy in blockchain-based VANET, the identity is divided to more than $k$ sub-identities, which will be periodically updated adopting the way of dynamic threshold encryption. Moreover, this paper designs two indicators (connectivity and average distance) to measure the effectiveness of $k$-anonymity unity for location privacy preserving. The experimental results demonstrated that our blockchain-based VANET had high efficiency in system time and privacy protection.

**Acknowledgements** This research was partially supported by Natural Science Foundation of China (61571098), 111 project (B14039), Fundamental Research Funds for the Central Universities (002).

## References

1. Vehicle/5G, available on https://http://www. sohu. com/a/229846324_114835, accessed on Apr. 28, 2018
2. Dvir E, Strasser G (2018) Does Marketing Widen Borders? Cross-Country Price Dispersion in the European Car Market. Journal of International Economics, 134–149
3. Rasheed H, Donghyun K, Junggab S et al. (2018) Secure and Privacy-Aware Incentives-Based Witness Service in Social Internet of Vehicles Clouds, IEEE Internet of Things Journal, 2441–2448
4. Jie C, Jing Z, Hong Z et al. (2018) An efficient certificateless aggregate signature without pairings for vehicular ad hoc networks, Information Sciences, 1–15
5. Sun G, Yu M, Liao D et al. (2018) Analytical Exploration of Energy Savings for Parked Vehicles to Enhance VANET Connectivity. IEEE Transactions on Intelligent Transportation Systems, DOI: https://doi.org/10.1109/TITS.2018. 2834569
6. Sun G, Zhang Y, Liao D et al. (2018) Bus Trajectory-Based Street-Centric Routing for Message Delivery in Urban Vehicular Ad hoc Networks. IEEE Transactions on Vehicular Technology, 7550–7563
7. Nzouonta J, Rajgure N, Wang G et al. (2009) VANET Routing on City Roads Using Real-Time Vehicular Traffic Information. IEEE Transactions on Vehicular Technology, 3609–3626

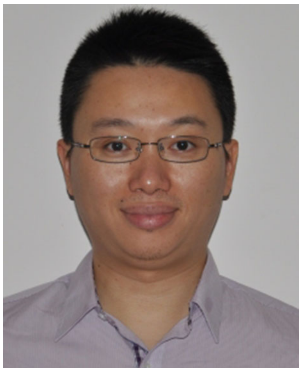
8. Li H Liao D, Sun G et al. (2017) Towards Location and Trajectory Privacy Preservation in 5G Vehicular Social Network. IEEE International Conference on Computational Science and Engineering (CSE) and IEEE International Conference on Embedded and Ubiquitous Computing (EUC), 63–69
9. Liao D, Li H, Sun G et al. (2018) Location and Trajectory Privacy Preservation in 5G-Enabled Vehicle Social Network Services. Journal of Network and Computer Applications, 108–118
10. Li H, Liao D, Sun G et al. (2018) Two-stage Privacy-preserving Mechanism for a Crowdsensing -based VSN. IEEE Access, 40682–40695
11. Liao D, Li H, Sun G et al. (2015) Protecting User Trajectory in Location-Based Services, IEEE GLOBECOM, 1–6
12. Liao D, Li H, Anand V et al. (2016) Using Location-labeling for Privacy Protection in Location-Based Services. International Conference on Internet of Things and Big Data, 299–306
13. Sun G, Liao D, Li H et al. (2017) L2P2: A location-label based approach for privacy preserving in LBS. Future Generation Computer Systems, 375–384
14. Liao D, Sun G, Li H et al. (2017) The Framework and Algorithm for Preserving User Trajectory while using Location-Based Services in IoT-Cloud Systems. Cluster Computing, 2283–2297
15. Novo O (2018) Blockchain Meets IoT: An Architecture for Scalable Access Management in IoT. IEEE Internet of Things Journal, 1184–115
16. Sun G, Chang V, Ramachandran M et al. (2017) Efficient Location Privacy Algorithm for Internet of Things (IoT) Services and Applications. Journal of Network and Computer Applications, 3–13
17. Sun G, Xie Y, Liao D et al. (2017) User-Defined Privacy Location-Sharing System in Mobile Online Social Networks. Journal of Network and Computer Applications, 34–45
18. Tyagi A, Sreenath N (2015) Location privacy preserving techniques for location based services over road networks. International Conference on Communications and Signal Processing (ICCSP), 1319–1326
19. Ullah I, Wahid A, Shah M et al. (2017) VBPC: Velocity based pseudonym changing strategy to protect location privacy of Vehicles in VANET. International Conference on Communication Technologies, 132–137
20. Au M, Liu J, Fang J et al. (2014) A New Payment System for Enhancing Location Privacy of Electric Vehicles. IEEE Transactions on Vehicular Technology, 3–18
21. Ying B, Makrakis D, Mouftah H (2013) Dynamic Mix-Zone for Location Privacy in Vehicular Networks. IEEE Communications Letters, 1524–1527
22. Kang J, Yu R, Huang X et al. (2018) Privacy-Preserved Pseudonym Scheme for Fog Computing Supported Internet of Vehicles. IEEE Transactions on Intelligent Transportation, 2627–2637
23. Zhang S, Wang G, Liu Q et al. (2018) A trajectory privacy-preserving scheme based on query exchange in mobile social networks. Soft Computing, 6121–6133
24. Zhu L, Xie H, Liu Y et al. (2018) PTPP: Preference-Aware Trajectory Privacy-Preserving over Location-Based Social Networks. Journal of Information Science and Engineering, 803–820
25. Memon I, Chen L, Arain Q et al. (2018) Pseudonym changing strategy with multiple mix zones for trajectory privacy protection in road networks. International Journal of Communication Systems. 1–18
26. Cui J, Xu W, Zhong H (2018) et al. Privacy-Preserving Authentication Using a Double Pseudonym for Internet of Vehicles. Sensors, 1–15
27. Afifi M, Zhou K, Ren J (2018) Privacy Characterization and Quantification in Data Publishing. IEEE Transactions on Knowledge and Data Engineering, 1756–1769
28. Takabi H, Joshi J, Karimi H (2009) A collaborative k-anonymity approach for location privacy in location-based services, International Conference on Collaborative Computing, 1–9
29. Niu B, Li Q, Zhu X et al. (2014) Achieving k-anonymity in privacy aware location-based services. IEEE INFOCOM, 754–762
30. Islam S, Obaidat M, Vijayakumar P (2018) et al. A robust and efficient password-based conditional privacy preserving authentication and group-key agreement protocol for VANETs. Future Generation Computer Systems, 217–227
31. Cui J, Zhang J, Zhong H et al. (2018) An efficient certificateless aggregate signature without pairings for vehicular ad hoc networks. Information Sciences, 1–15
32. Peters D, Wetzlich J, Thiel F et al. (2018) Blockchain applications for legal metrology. IEEE International Instrumentation and Measurement Technology Conference (I2MTC), 1–6
33. Joy J, Cusack M (2017) Internet of Vehicles and Autonomous Connected Car-Privacy and Security Issues. IEEE International Conference on Computer Communication and Networks, 1–9
34. Joy J, Cusack G, Gerla M (2017) Poster: Time Analysis of the Feasibility of Vehicular Blocktrees. ACM 3rd Workshop on Experiences with the Design and Implementation of Smart Objects, 25–26
35. Dorri A, Kanhere S, Jurdak R (2017) Towards an Optimized BlockChain for IoT. ACM 2nd International Conference on Internet-of-Things Design and Implementation, 173–178
36. Sharma P, Moon S, Park J (2017) Block-VN: A Distributed Blockchain Based Vehicular Network Architecture in Smart City. J Inf Process Syst 13(1):184–195
37. Lei A, Cruickshank H, Cao Y et al. (2017) Blockchain-Based Dynamic Key Management for Heterogeneous Intelligent Transportation Systems. IEEE Internet of Things Journal, 1832–1843
38. Amoretti M, Brambilla G, Medioli F et al. (2018) Blockchain-Based Proof of Location. IEEE International Conference on Software Quality, Reliability and Security, 146–153
39. Herzberg A, Jarecki S, Krawczyk H et al. (1995) Proactive Secret Sharing Or: How to Copy with Perpetual Leakage. Proc of Crypto, 339–352
40. Alphand O, Amoretti M, Claeys T et al. (2018) IoTChain: A Blockchain Security Architecture for the Internet of Things. IEEE Wireless Communications and Networking Conference (WCNC), 1–6
41. A. Said, M. Marot, A. Ibrahim, et al. (2016) Modeling interactive real-time applications in VANETs with performance evaluation. Computer Networks, 66–78
42. G. Wood. Ethereum: A Secure Decentralised Generalised Transaction Ledger. Available: https://www.ethereum.org/

**Publisher's note** Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

**Hui Li** received the BS and MS degree at the University of Electronic Science and Technology of China (UESTC) in 2012 and 2016, respectively. She is currently pursuing the Ph.D. degree with the Information and Communication engineering, UESTC.

**Lishuang Pei** received a bachelor's degree from Chengdu University of Technology (cdut) in 2018. He currently pursing a master's degree in electronics and communications engineering from UESTC.

**Gang Sun** is an associate professor of Computer Science at University of Electronic Science and Technology of China (UESTC). He is a Member of IEEE/IEEE Computer Society. He has also edited special issues at top journals.



**Dan Liao** is a professor at University of Electronic Science and Technology of China (UESTC). His research interests are in the area of next generation network, wired and wireless computer communication networks and protocols.

**Du Xu** is a professor at University of Electronic Science and Technology of China (UESTC) in Chengdu, China. He presided over many advanced research projects, including NSFC, National 863 Plans and National key Research and Development Program of China.