applied sciences

MDPI

*Article*

# SBTMS: Scalable Blockchain Trust Management System for VANET

**Fatemeh Ghovanlooy Ghajar [1]*, Javad Salimi Sratakhti [2,*] and Axel Sikora [1]**

[1] Institute of Reliable Embedded Systems and Communication Electronics (ivESK) , Offenburg University of Applied Sciences, 77652 Offenburg, Germany; axel.sikora@hs-offenburg.de
[2] Department of Electrical and Computer Engineering, University of Kashan, Kashan 8731753153, Iran
* Correspondence: fatemeh.ghovanlooy@hs-offenburg.de (F.G.G.); salimi@kashanu.ac.ir (J.S.S.)

**Abstract:** With many advances in sensor technology and the Internet of Things, Vehicle Ad Hoc Network (VANET) is becoming a new generation. VANET's current technical challenges are deploying decentralized architecture and protecting privacy. Because Blockchain features are decentralized, distributed, mass storage, and non-manipulation features, this paper designs a new decentralized architecture using Blockchain technology called Blockchain-based VANET. Blockchain-based VANET can effectively resolve centralized problems and mutual distrust between VANET units. To achieve this, it is needed to provide scalability on the blockchain to run for VANET. In this system, our focus is on the reliability of incoming messages on the network. Vehicles check the validity of the received messages using the proposed Bayesian formula for trust management system and some information saved in the Blockchain. Then, based on the validation result, the vehicle computes a rate for each message type and message source vehicle. Vehicles upload the computed rates to Roadside Units (RSUs) in order to calculate the net reliability value. Finally, RSUs using a sharding consensus mechanism generate blocks, including the net reliability value as a transaction. In this system, all RSUs collaboratively maintain the latest updated Blockchain. Our experimental results show that the proposed system is effective, scalable and dependable in data gathering, computing, organization, and retrieval of trust values in VANET.

**Keywords:** VANET; blockchain-based system; sharding algorithm; trust management system



**Citation:** Ghovanlooy Ghajar, F.; Salimi Sratakhti, J.; Sikora, A. SBTMS: Scalable Blockchain Trust Management System for VANET. *Appl. Sci.* **2021**, *11*, 11947. https://doi.org/10.3390/app112411947

Academic Editors: Pedro Valderas and Victoria Torres

Received: 9 November 2021
Accepted: 11 December 2021
Published: 15 December 2021



## 1. Introduction

Vehicular ad hoc network (VANET) is one of the most significant current discussion in the intelligent transport [1]. VANET provides a platform for vehicles to share road-related messages with other peers. This ability has greatly improved the management of urban traffic and road accidents. The point is event messages that dramatically increase in this network. Moreover, due to the numerous and diversity of vehicles in this network, vehicles are unable to fully trust each other. As a consequence, a trust management system and a method to coordinate the vehicle's interactions efficiently are required. The primary option, for the defined purpose, is a centralized system. In this kind of trust management system, all messages are stored and processed in a central server. Centralized systems can carry risks such as bottlenecks caused by increased layers of approvals and slower decision-making and single-point of failure. This challenge faces the system with efficiency and communication between node's problems. On the other hand, decentralized system is one of the most well-known approaches to overcome centralized issues. In decentralized systems, every node makes its own decision. One also should not overlook the fact that the final behavior of the system is an aggregation of the individual node's decisions. With this in mind, Blockchain is a one of the solutions to solve above problems. However, regarding the decentralized approach, it is a practicable tool for VANET. Blockchain enables distributed nodes to cooperate with each other and maintain a consist and tamper-proof ledger. Furthermore, there is no requires trusted third party (TTP) to

establish trust, the Blockchain technology can be seen as a feasible distributed platform. It is a distributed peer-to-peer (P2P) system for communicating and transferring transactions between nodes without TTP. Some Blockchain technology applications include digital currencies [2], health insurance system [3], health care system [4], Internet of things [5], social networks [6], intelligent contracts [7], voting system [8], authentication [9], notary [10], data storage [11], energy supply [12], supply chain management [13] and protection of intellectual property[14]. In all of these applications, Blockchain tries to eliminate the focus on a single entity. This has great potential for implementing centralized systems in a distributed way without the mentioned problems. A combination of Blockchain and IoT technology can greatly improve the field of security, reliability, storage of data, and immutability. However, due to the mining process, most of the Blockchain technologies have some shortcomings such as low transaction throughput and poor scalability [15]. In VANET the number of vehicles and their exchanged messages increase continuously, therefore the scalability is another important issue. Sharding as a consensus algorithm can make Blockchain's ledger more efficient, scalable, and sustainable by dividing large amounts of data into chunks. In this paper, a sharding consensus algorithm is employed to provide scalability. Sharding algorithm distributed mining tasks into committees, each of the committees processes a set of transactions [16]. Based on the distributed nature of Blockchain, trust management can be applied in distributed way among road side units (RSU), which can forbear the centralized issues. Moreover, Blockchain enables RSUs to work simultaneously together and patronize a consistent database. The main contribution of this paper can be summarized as follows:

1 We propose a scalable and tamper-proof distributed trust management system for VANET. In this system, RSUs response to vehicle's queries and calculate the trust by employing sharding consensus algorithm in several committees.
2 We propose a new Bayesian Inference-based validation procedure for vehicles to neutralize the effect of deceptive messages.
3 We simulate the proposed model to show that the proposed scalable Blockchain trust management system is efficient in VANET environment.

The rest of this paper is organized as follows. Section 2 introduces Blockchain, while the literature reviews are explained in Section 3. In Section 4, we propose our Blockchain-based VANET system in details. Performance evaluation and further discussion about the proposed system is given in Section 5. Finally, Section 6 concludes this paper.

## 2. Basic Concepts

*An Overview on the Blockchain Concepts*

Blockchain is defined as a distributed and consensus-based ledger that all successful transactions are stored in a list of blocks. In Blockchain network, communications are P2P [17]. In this technology, asymmetric cryptography and a consensus algorithm are used to authenticate the entities and to ensure the integrity of the Blockchain [18]. Blockchain structure is a back-linked record of transaction blocks, each block can be recognized by a hash value on the header of a block. The header contains index, previous hash, number of transactions, timestamp, nonce, and Merkel tree. The block body contains transactions that miners place in a block. Blockchain structure has been shown in Figure 1.

The key contribution of the Blockchain mechanism is consensus algorithm. The consensus algorithm has been designed to achieve reliability in a network that includes unreliable nodes [19]. The consensus algorithm within the Blockchain network ensures that all network agents have the same copy ledger. Confirmation of transactions and aggregation in Blockchain is accomplished through consensus protocols. Consensus protocols can be divided into two categories: proof-based algorithms and Byzantine algorithms [20]. Examples of proof options include proof of work, stocks proof, proof of capacity, proof of space, proof of elapsed time, proof of burning, proof of activity, proof of power, and proof of equity are examples of proof-based algorithms. Sharding algorithm and Byzantine algorithms include the tolerance of applied Byzantine error and Byzantine consensus.

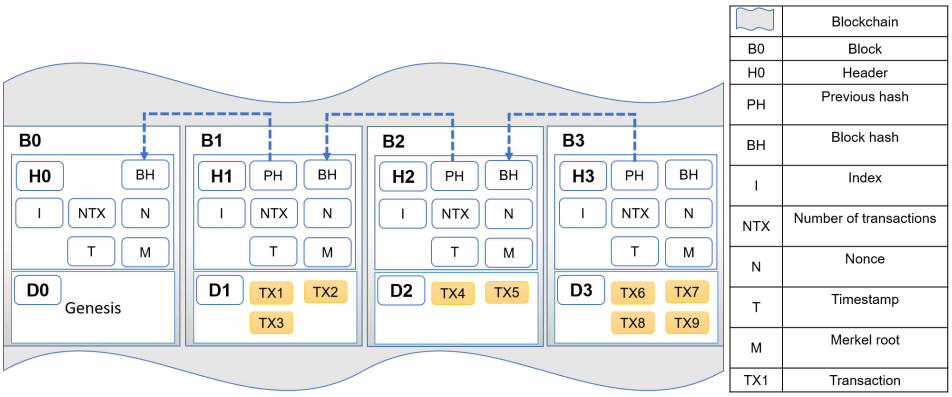



**Figure 1.** Blockchain structure.

## 3. Related Work

### *3.1. Trust Management Models*

On the Internet of Things, trust management and scalability are very important problems [21,22]. Most trust-based networks build a centralized confidential system that can examine whether a particular node is trustable or not [23]. VANET is an ad hoc network of vehicles that trust management is the critical issue for cooperating between the nodes. Only a few models consider sharing information on VANET [24,25]. Researchers focus on trust models that are dynamic and applicable to the network systems. In these models, nodes can cooperate based on the calculated trust and using exchanged messages. There are two type of trust management system for networks, specially VANET: central and decentralized systems. The benefit of central trust management systems is that they are easier to control and have reduced cost, but it has several big disadvantages. If the main system goes down, the whole system will fail. However, in the decentralized system, this problem has been fixed. Therefore, some researchers focused on the decentralized trust in VANET [26–28]. Many trust models have been used in P2P network to update node's belief based on the other node's trust value. For example, the system in [29] is one of the earlier distributed systems in VANET. They proposed a system for cars based on crowdsourcing in VANET. The system allows cars to hide votes, points, and lists of interacting vehicles under their own homogeneous encryption layer in order to stay out of reach of the malicious agents. It only uses the total credit of the cars at the end and uses the accumulated weight of the credits.

### *3.2. Possibility of Validating Reliability Using Artificial Intelligence*

One of the major challenges in VANET systems is the updating of automated decision-making processes performed during driving vehicles. The condition of roads or vehicles may in fact lead to inefficiency of planning, hence it needs to be held accountable to productivity. In order to address the current limitations of traditional scheduling methods, which is done by the traffic control center. Artificial intelligence can be helpful in predicting and preventing these cases. Maskey et al. studied this issue and proposed architecture [30]. They proposed ALICIA (AppLied Intelligence in bloCkchaIn vAnet) to be used with the Artificial Neural Network to select when and which node to exclude during the consensus process. Furthermore, they recommended an accident detection and validation system where to detect and validate an accident, and send the data to ALICIA for miner node selection. Castaño et al. proposed a quality control framework based on a model-based approach using embedded artificial intelligence strategies [31]. In this work, strategies are applied to monitor the microstructure process, with the aim of demonstrating the excellent performance of the framework in a very complex system. Another study regarding using AI for decision-making and reliability validation was done by Villalonga et al. They proposed a framework for decentralized and integrated decision-making for rescheduling a physical cyber-production system [32]. Furthermore, the decision-making process is supported on a

fuzzy inference system using the state or conditions of different assets and the production rate of the whole system.

### 3.3. Blockchain-Based Decentralized Data Management

Blockchain is a very good option for implementing distributed architecture. So far, a few works have been represented on Blockchain in VANET. Yang et al. proposed a distributed trust management system for vehicle network based on Blockchain technology. In this system, vehicles use Bayesian inference models to confirm received messages from other vehicles. Thus, each vehicle can produce report message for other vehicles and RSUs can calculate trust values. Finally, a trust value is stored in the blocks to improve traffic safety in an unreliable VANET [33]. Islam has employed Blockchain for Smart Internet of Things (SIoT) without any human intervention. SIoT categorizes and shares information based on social trust values. This network is fully distributed, and the Blockchain platform has been proposed to provide security for manufacturing operations [34]. Xiang explores how to use Blockchain in VANET and takes into account the distributed storage approach and security in bulk data generated by vehicles. He presents a model in which several types of nodes (i.e., cars and RSUs) are existing [35]. Kang et al. used the Blockchain consortium and intelligent contracts for storing and sharing secure data in VANET. Their system prevents the sharing of unverified data. In addition, they introduced a method for data sharing based on reputation, which is a logical triple model based on the frequency of interaction, real-time event, and the similarity of the vehicle's direction. Consequently, vehicles can achieve more qualitative data [36]. Barttholomy et al. have used of tangle technology to improve the safety and efficiency of cars in the VANET. They introduced and analyzed tangle as a solution to the Blockchain problems in the VANET [37]. Kang et al. proposed a Blockchain-based structure for VANET that miners are selected based on their cooperation in this system. Selecting miners in this system is done based on the level of reputation and availability. Furthermore, to encourage miners to participate in the block validation, a contract is set between the active and standby miners [38]. Azad et al. designed an M2M-REP system to protect the privacy of independent devices in the network. This system provides a condition that does not allow malicious nodes to manipulate people's message reliability and thus increases the overall trust of the computations [39]. Kanichet and Larrent have proposed a hierarchical identity-based encryption mechanism according to a Blockchain infrastructure, they provided a structured model based on Blockchain for data usage to more availability and privacy. Their model is based on audible contracts in the Blockchain infrastructure and can protect privacy and confidentiality of information [40]. In a different work, Gao et al. provided a safe payment mechanism in VANET based on Blockchain, which protects user information during period of data sharing time. Their proposed mechanism involves processes for storing data based on Blockchain technology and supporting anonymity [41].

The mentioned studies guarantee the security of storage data by Blockchain solution. Despite this advantage, one of their significant shortcoming is low transaction throughput and small scalability, while in VANET there are great number and high speed of generated transactions. In this study, we propose a novel method based on the Sharding algorithm to overcome this shortcoming.

## 4. The Proposed Scalable Blockchain Trust Management System (SBTMS)

This study aims to provide a tamper-proof, scalable, and trustable system for VANET. Figure 2 shows the VANET architecture that denotes normal workflow from observing an event until sending the message to RSU. In the proposed architecture (Figure 3), that hereafter is referred to as SBTMS, for the trust management each vehicle by observing an event or any kind of problem on their route, sends a message about the event to others. The vehicle that give the message, calculates message reliability between 0 and 1, then sends it to the RSU. Throughout this study, it is assumed that vehicles cooperate in sending and receiving the event messages. RSUs receive the message reliability from vehicles and

calculate net reliability (*OS*) for a given message type and a sender. The net reliability, as a transaction, adds to the Blockchain based on the sharding consensus algorithm. The Blockchain maintains the integrity of the network.

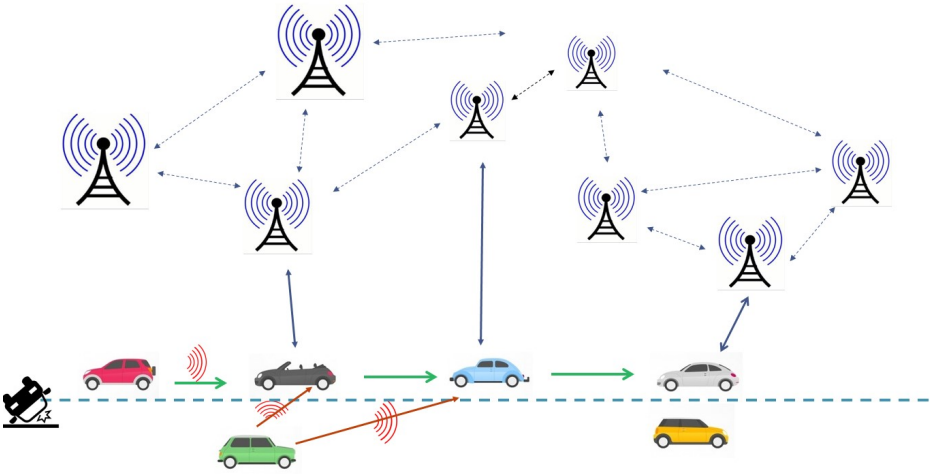


**Figure 2.** VANET architecture.

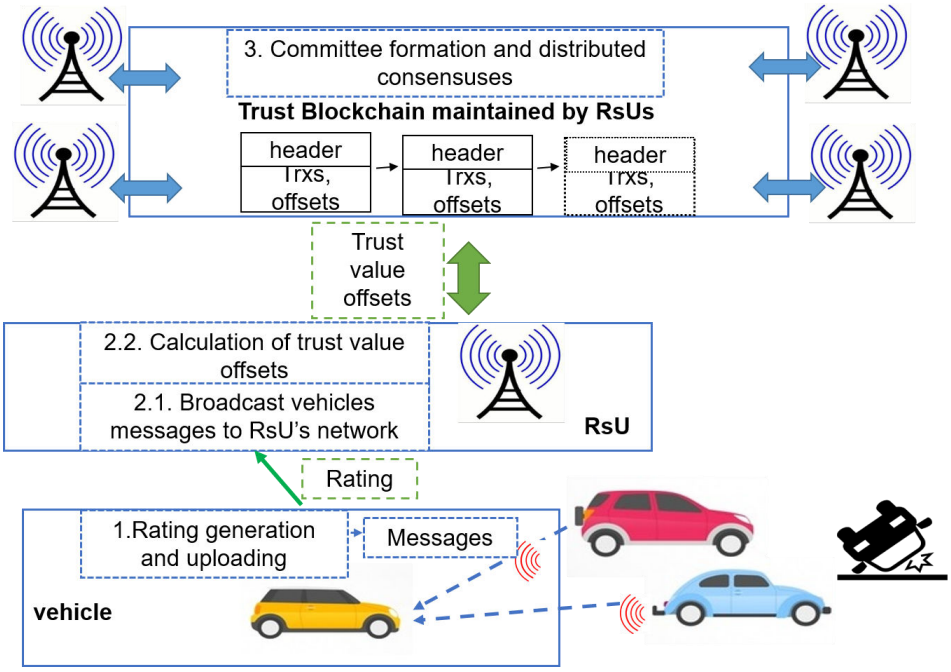



**Figure 3.** The proposed architecture: merging VANET with Blockchain.

Each received message should be validated by the receiver. As illustrated in Figure 3, the message validation in SBTMS including four main steps: (1) Calculating event confidence score and sending it to RSU. (2) Calculating the net reliability by RSUs for a given type of message and sender vehicle. (3) Forming committees by RSUs. (4) Running a consensus algorithm to add the transactions to Blockchain. In the following, we will discuss these steps. For ease of reference, Table 1 summarizes the model parameters and notations.

**Table 1.** Summary of parameters and notations.

| Symbols | Definition |
|---|---|
| $m$ | event message |
| $LL$ | Longitude and Latitude |
| $V_s$ | message sender |
| $d_s$ | The distance between the sender and the event |
| $c_s$ | The validity of the message m from the Vs sender calculated by each vehicle |
| $C$ | Vector of sum of values of Cs from different senders for a message m |
| $o_s$ | The amount of net reliability of Vs sender which is calculated by RSU |
| $O$ | Vector of sum of Os values from different senders for a message m |
| $N'$ | The number of vehicles participating in the scoring |
| IP | IP of each RSU |
| PK | External key of each RSU |
| $N$ | Total RSUs |
| $z$ | Maximum size of Committee |
| $2^s$ | Number of Committees |

### *4.1. Step 1: Calculating Confidence Score and Message Reliability*

A vehicle, by observing a given event, independently broadcasts the massage to the neighbor vehicles. This message is 5-tuple $(v_s, LL, e, m, t)$, where $V_s$ indicates identification of sender vehicle, $LL$ shows event location including longitude and latitude, $e$ indicates event type of message, $m$ is message's ID, and $t$ represents time stamp. The message reliability is calculated for a given message type and message sender, and is based on two effective cases: (1) the distance between message sender and event location (i.e., event confidence score) and (2) the level of reliability of the message sender for the given message type. The message that sent by the nearest vehicles to the event is more reliable than those sent by further vehicles. Therefore, the receiver can calculate a confidence score $C_s$ based on the distance for the event $e$ using Equation (1).

$$c_s = \frac{1}{1+e^{s\left(g-\left((d_s)^{-1}\right)\right)}} \tag{1}$$

Here, $C_s$ is confidence score for the event that has been sent by $V_s$, $d_s$ is distance between the sender message and the event location ($LL$). If $V_s$ does not send a message, then $C_s$ the value is equal to 0.

Due to the mobility of vehicles in the VANET, several vehicles may observe an event in the network. Thus, a vehicle may receive the same messages from different senders. Consequently, the receiver calculates several confidence scores for these same messages based on the distance of its sender and the event location. Vector $C = c_1, c_2, \ldots$ shows the confidence scores. After that, each vehicle requests the net reliability level for message sender $V_s$ from the nearest RSU. The RSU sends the net reliability level $Os$ to the requester. Therefore, the requester can calculate message reliability given by net reliability on a given type of message and confidence score of the message type (i.e., $P(m|O, C)$) as follows:

$$P(m|O, C) = \frac{P(m|C) \times P(C|m,O)}{P(m|C) \times P(C|m,O) + P(\overline{m}|C) \times P(C|\overline{m},O)} \tag{2}$$

where $P(m \mid C)$, $P(C \mid m, O)$, $P(C \mid m, O)$ are given by Equations (3)–(5).

$$P(m|C) = \frac{P(m) \times \prod_{s=1}^{N} P(c_s|m)}{P(m) \times \prod_{s=1}^{N} P(c_s|m) + P(\overline{m}) \times \prod_{s=1}^{N} P(c_s|\overline{m})} \tag{3}$$

$$P(C|m,O) = \prod_{s=1}^{N} P(c_s|m,O) \tag{4}$$

$$P(c_s|m,O) = P(c_s|O) \tag{5}$$

In Equation (5), $c_s$ is independent of message type $m$, so we have

$$P(c_s|O) = \frac{P(c_s) \times P(O|c_s)}{P(c_s) \times P(O|c_s) + P(\bar{c_s}) \times P(O|c_s)} \tag{6}$$

In Equation (6), it is assumed that $P(c_s)$ follows the normal distribution. Furthermore, $P = (\bar{c_s})$ is equal to $1 - P(c_s)$. In this Equation, $P(O|c_s)$ and $P(o \mid c_s)$ are given by Equations (7) and (8).

$$P(O|c_s) = \prod_{s'=1}^{N} P(o_{s'}|c_s) = \prod_{s'=1}^{N} o_{s'} \tag{7}$$

$$P(O|\overline{c_s}) = \prod_{s'=1}^{N} P(o_{s'}|c_s) = \prod_{s'=1}^{N} (1 - o_{s'}) \tag{8}$$

$P(\bar{m} \mid C)$ in Equation (2) is calculated using by Equation (9).

$$P(\overline{m}|C) = \frac{P(\overline{m}) \times \prod_{s=1}^{N} P(c_s|\overline{m})}{P(\overline{m}) \times \prod_{s=1}^{N} P(c_s|\overline{m}) + P(m) \times \prod_{s=1}^{N} P(c_s|m)} \tag{9}$$

where $P(m)$, $P(c_s \mid m)$ and $P(c_s \mid \bar{m})$ are equal to $1 - p(m)$, $c_s$, and $1 - c_s$, respectively. Furthermore, $P(C \mid \bar{m}, O)$ is given by

$$P(C|\overline{m},O) = P(C|O) = \prod_{s=1}^{N} P(c_s|O) \tag{10}$$

Finally, the receiver vehicle sends the calculated reliability to the nearest RSU as $(v_r, v_s, m, P(m|O,C))$, where $V_r$ is the receiver's ID, $V_s$ is the sender's ID, $m$ is the message's ID, and $P(m|O,C)$ is the calculated reliability level by $V_r$ for the message $m$. Algorithm 1 summarizes undertaken activities by each vehicle.

**Algorithm 1:** Vehicle's activities.

```
Input  : Event
Output : Rate value
while vehicle (running) do
    if (see an event) then
        Send message to other vehicles;
    end
    if ( Vehicle receive a message) then
        Forward message and set counter=counter+1;
        Calculate c_s;
        Send query for nearest RSU to get o_s;
        Calculate P(m | O,C);
        Send message reliability to RSU;
    end
end
```

### *4.2. Step 2: Calculating Net Reliability*

An RSU may receive different message reliability values generated by different vehicles. Each RSU should calculate net reliability based on the all received reliability values. In many studies, when an exact description of a phenomenon is lacking, researchers use an S-curve (sigmoid function) to approximate the phenomenon's behavior [42–44]. Net

reliability in our problem has such behavior. When the sum of received reliability values are low, net reliability will be near to zero. By increasing the sum, net reliability raises and approaches to its maximum level. In this study, net reliability is given by Equation (11).

$$o_{(u,s,m)} = \frac{1}{1 + e^{-k\left(\left(\frac{1}{N'}\sum P(m|O,C)\right) - x_0\right)}} \tag{11}$$

where $k$ is steepness, $x_0$ is the inflection point of the S-curve, and $N'$ indicates the number of received reliability values about a given type of the message by a given sender vehicle. Maximum value of $O(u, s, m)$ is equal to one.

### 4.3. Step 3: Formation of Committees by RSUs

After calculating the net reliability by RSUs, they should cooperate with each other to add the net reliability as a transaction to the Blockchain. Let assume a network of $N$ RSUs participate in adding block to the Blockchain. To prevent collusion, RSUs employ self-generated transient identity rather than a permanent identity or a public key infrastructure [45]. To provide a transient identity, RSU must find a Proof-of-work (PoW) solution. RSU's Committees will then compose corresponding to the same generated identities. For finding PoW, each RSU must generate a random string (i.e., EpochRandomness) and find a nonce that satisfies the following inequality:

$$O = H(EpochRandomness \parallel IP \parallel PK \parallel nonce) < 2^{\gamma - D} \tag{12}$$

where $O$ will be RSU's transient identity, $D$ is a predefined value on the network, IP is RSU address, and PK is RSU's public key. The value of $D$ determines the network's difficulty and can be set corresponding to the number of RSUs. EpochRandomness is generated to ensure that the PoW is not precomputed and consequently malicious RSUs cannot take over authority of the committees. Number of committees is given by $N = 2^s z$, where $N$ is the total number of RSUs in the network, $z$ is the maximum size of the committees, and $2^s$ is the number of committees. Based on the sharding algorithm and the last bit of its identifier, each RSU is assigned to a random committee. For more information, refer to [45], which provides a detailed discussion on sharding protocol.

### 4.4. Step 4: Consensus Process on Transactions

As mentioned in step one, calculated message reliability by the vehicles are sent to the nearest RSUs. Each RSU in a specific time calculates the net reliability and then broadcasts it as a transaction on the network of RSUs. Then, according to Step 3, RSUs generate a transient identity, and after that RSUs should attempt to find out transient identities of other RSUs in their committee by sharing or exchanging information with each other, and to compose a fully-connected overlay network for each committee in the RSUs's network. Next, in each committee, RSUs run Practical Byzantine Fault Tolerance (PBFT) protocol [46] to concur on a set (or shard) of transactions. Figure 4 shows the PBFT protocol in a committee to add a transaction in a shard.In PBFT consensus process. In the pre-prepare phase, a RSU sends a transaction to the committee members for validating. In the preparation phase, each of the committee's RSU verifies the transaction validity and sends the result to the others. In commit phase, each RSU that receives more than $\frac{2z}{3}$ of the prepare message, sends the commit message to the others. Finally, by receiving $\frac{2z}{3}$ a commit message by the others, the transaction is validated and added to the set (or shard).

The consensus shard in each committee, signed by at least $\frac{z}{2} + 1$ RSUs, is sent to the final committee, which is composed based on a given s-bit final committee identifier, to create a new block in the Blockchain. The final committee receives the consensus shards provided by the committees and computes a cryptographic digest. To do so, RSUs in final committee run again a PBFT and send shards to the network. Figure 5 presents the process of sharding protocol in our model.

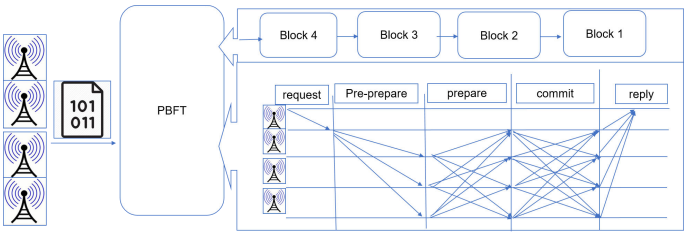



**Figure 4.** PBFT consensus process.

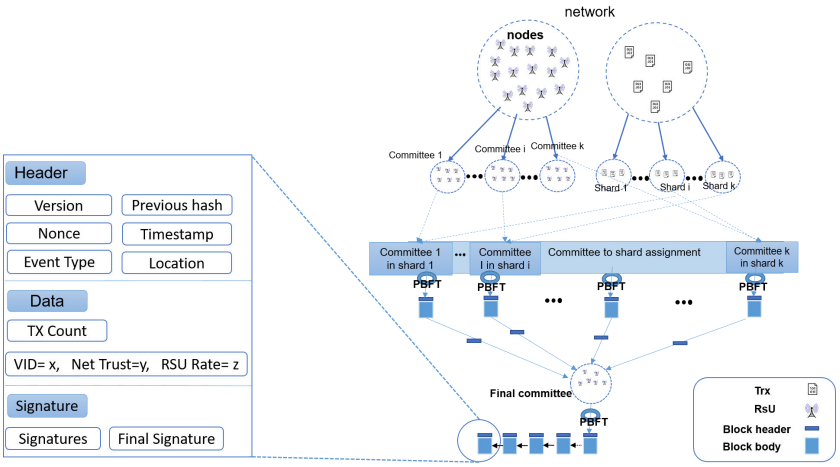



**Figure 5.** Sharding protocol in SBTMS.

## 5. Results

In this section, it is described how to simulate the proposed system in details. It is simulated the system with 20 RSUs, 10–3000 vehicles, and 3 events: accident, traffic, and road destruction. RSUs after computing net reliability run sharding protocol. It is assumed committees of size 4, and the number of RSU in each committee for consensus is 3. Each RSU in the system receives transactions that are corresponding to the committee that it belongs to. In the simulation, has been assumed that each committee confirms all transaction with net reliability greater than a threshold (70%). Another simulation's specifications are shown in Table 2.

Python language is used to simulate this infrastructure. The asynchronous simulation environment simulates network interactions between the vehicle and the blockchain. RSUs are stationary, and each vehicle is moving randomly and is moving in a two-dimensional environment. Events also occur randomly in the simulation environment.

**Table 2.** Table of symbols.

| Parameter | Value |
|---|---|
| $k$ | 10 |
| $x_0$ | 0.5 |
| $s$ | 4 |
| $z$ | 5 |
| $g$ | 0 |
| $s$ | 1 |
| Area | $800 \times 1000$ |
| event number | 3 |
| Transmission Radius | 50 |
| maximum speed | 200 |
| vehicular number | 10–3000 |
| simulation time | 3600 (s) |

### *5.1. Security Analysis*

In the proposed system, it is assumed that some vehicles and RSUs may be malicious or attacked. Therefore, the system should be able to tolerate these abnormalities. In accordance with the work in [33], malicious vehicles may send fake messages to others, which can lead to disturbance. In the proposed system, trust is calculated based on the Bayesian inference on the reputation of the vehicles and RSUs. Therefore, sending fake messages cannot disturb the system. Another scenario is that vehicles send fake message reliability to RSU, but due to Bayesian inference in RSU, this case cannot cause disturbance.

Trust to RSUs is the most important issue in this kind of VANET. All agents (i.e., RSUs and vehicles) in the network are selfish. This means that they are intending to maximize their own payoffs. Ignoring this concept can result in failing mechanisms. For example, a recent study by Yang et al. [33] ignored the RSU's selfishness. They have assumed each RSU mines a block to add to the Blockchain, with a network's difficulty determined by itself. The network's difficulty is calculated based on the received messages. However, due to the rationality (or hacking), a RSU can set a low network's difficulty in order to faster mining and getting more payoffs. In our proposed system, committees use a public random string epochRandomness to make certain that the members randomly (not precomputed) compose the committee.

### *5.2. Message Reliability and Net Reliability Analysis*

In the proposed system, by increasing number of the exchanged messages, message reliability increases. Figure 6 indicates message reliability for three types of the events over time in the simulated environment for a given vehicle.

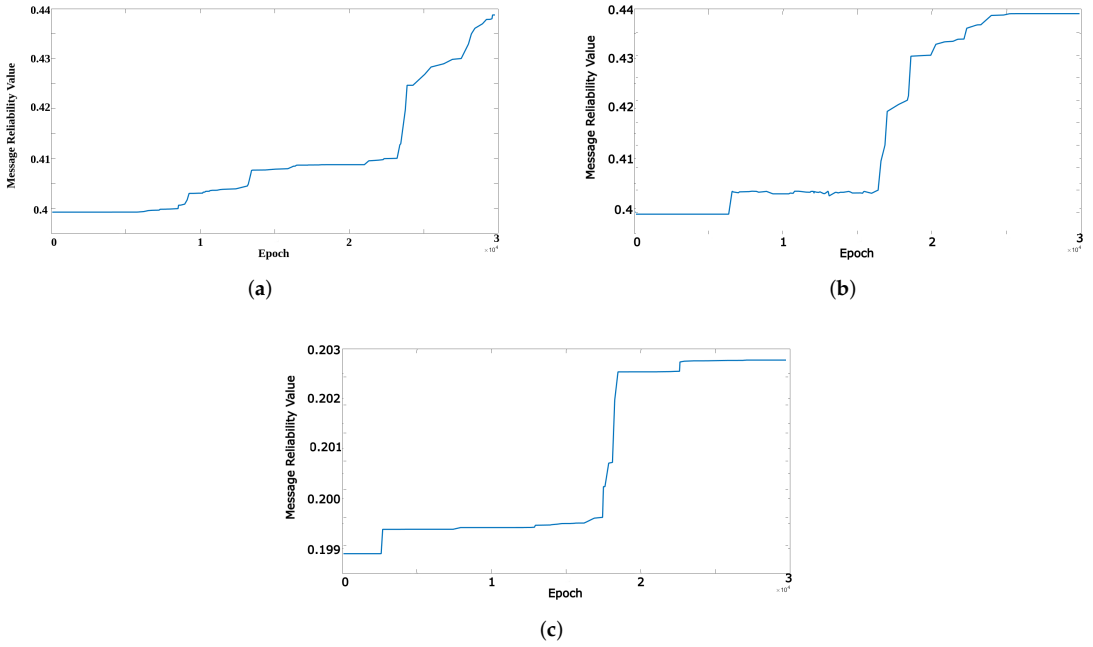



**Figure 6.** Changes of message reliability over time. (**a**) Event 1 (i.e., Accident); (**b**) Event 2 (i.e., Traffic); (**c**) Event 3 (i.e., Road Destruction).

As illustrated in Figure 6, by forwarding messages, their reliability increase. However, may exist several vehicles that do not forward event messages, hence the convergence speed of the message reliability reduce or fix (see Figure 6c Epoch 3000–17,000). Furthermore, if vehicles send fake messages, the message reliability can decrease (see Figure 6b Epoch 13,000).

Like as message reliability, the net reliability calculated by RSUs converges. This convergence has been shown in Figure 7 for three event messages originated from given vehicles. As shown in this figure, the net reliability increases over time.

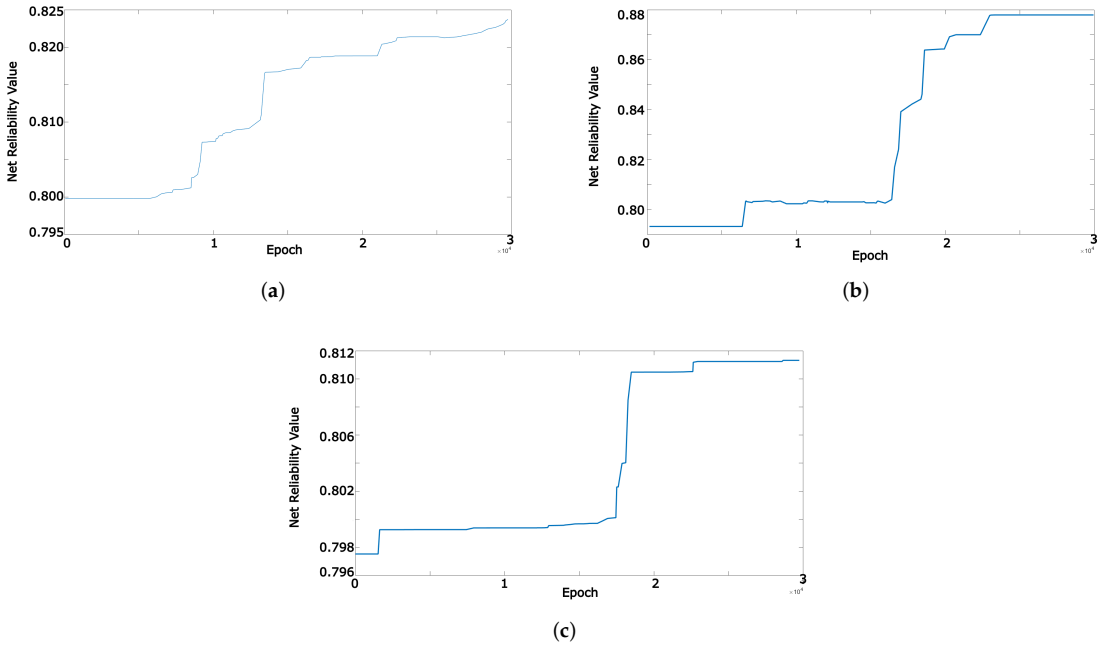



**Figure 7.** event OS value evaluation. (**a**) Event 1 (i.e., Accident); (**b**) Event 2 (i.e., Traffic); (**c**) Event 3 (i.e., Road Destruction).

In the above simulations, it is assumed that the event rates of accident, traffic, and road destruction are $\frac{2}{5}$, $\frac{2}{5}$, and $\frac{1}{5}$, respectively. In this subsection, it is shown the effect of event rates on message reliability and net reliability and their convergences. The event rates $P(m)$ had been employed in Equations (2) and (3). As shown in Figure 8 by increasing the event rate, net reliability for a message originated from a given vehicle is reduced.

Another criterion that may have a significant effect on the message and net reliability, and also on the performance of the system, is the radius that a vehicle is able to send or receive a message. As illustrated in Figure 9, increasing the vehicle's radius increases the message and net reliability. Growing the radius increases the number of vehicles that receive or send a given message, thus message and net reliability increase.

Number of vehicles is another determinant parameter. As the number of vehicles increases, a specific event message may be sent over and over again, which results in increasing message and net reliability. Furthermore, in this situation, fake messages have less impact on the reliability. Figure 10 shows reliability changes over different number of vehicles.

Number of blocks in the Blockchain and required time to mine a new block are important factors in the performance of the system. These factors depend on the number of times the messages are forwarded in the network. Figure 11 indicates the effect of number of messages in the network on number of blocks in the Blockchain and required time to mine.

The sharding consensus algorithm in the proposed system scales up the performance nearly linearly with the computational power of RSUs. It provides promising scalability in the simulation. To assess accuracy and precision of the proposed system, it is considered as a classification system in which any vehicle can classify the received message into two classes, correct and incorrect. In this way, it is defined: True positive (*TP*): If a received correct message is classified as a correct message by a vehicle; True negative (*TN*): If a received incorrect message is classified as an incorrect message by a vehicle; False positive (*FP*): If a received incorrect message is classified as a correct message by a vehicle; False

negative (*FN*): If a received correct message is classified as an incorrect message by a vehicle. Therefore, precision and accuracy of the system are given by

$$Precision = \frac{TP}{Tp+FP} \tag{13}$$

$$Accuracy = \frac{TP+TN}{Tp+FP+TN+FN} \tag{14}$$

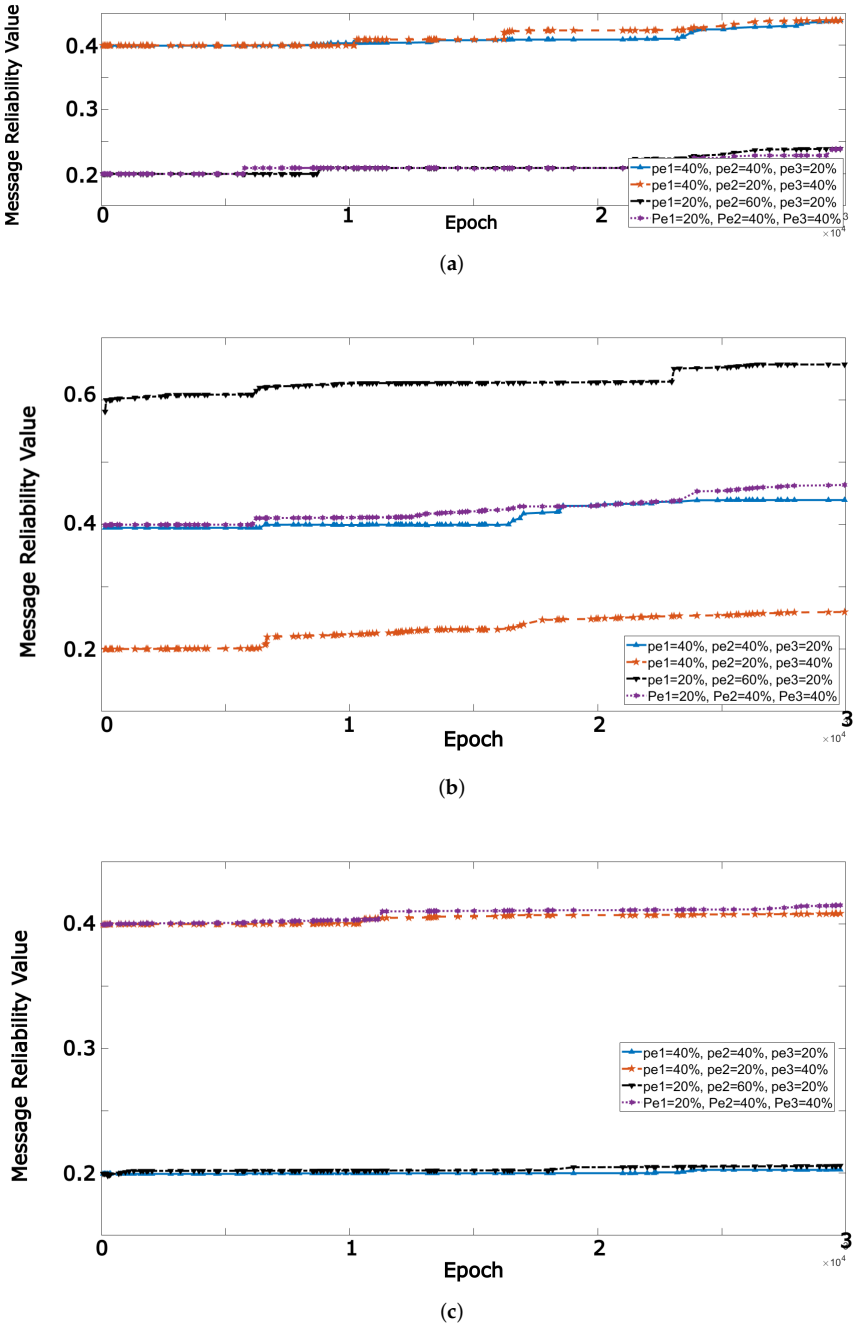



**Figure 8.** Message reliability values for different event rates. (**a**) Accident event; (**b**) Traffic event; (**c**) Road destruction event.

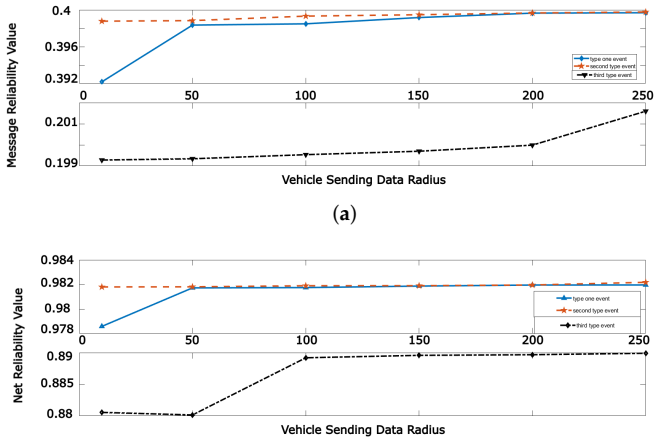



**Figure 9.** Message and net reliability values for different vehicle's radius. (**a**) Message Reliability; (**b**) Net Reliability.

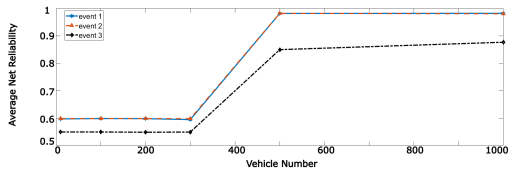



**Figure 10.** Net reliability with respect to the number of vehicles.

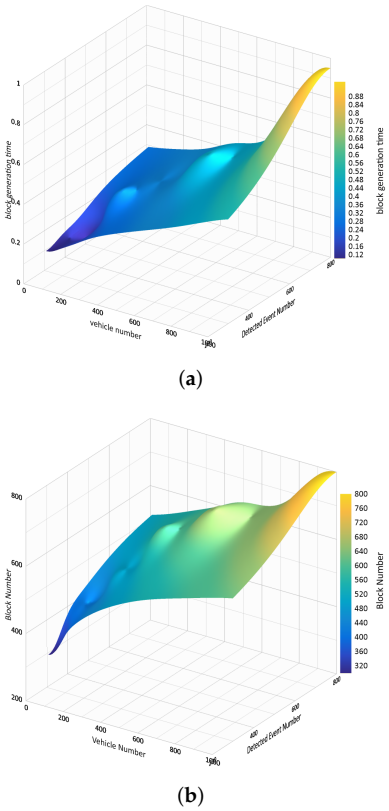
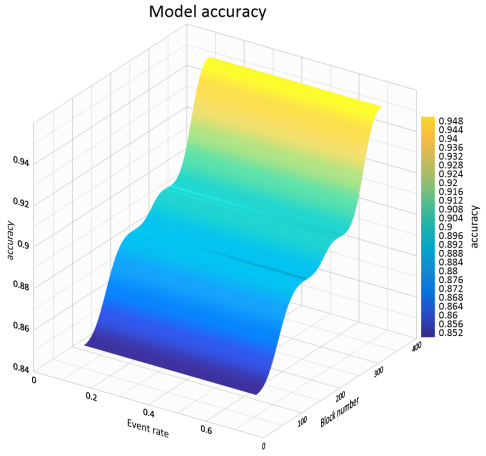
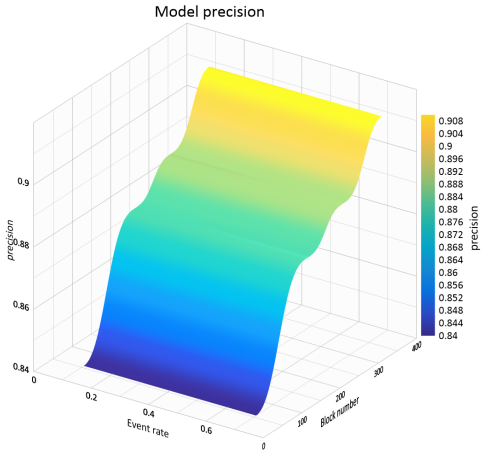



**Figure 11.** The effect of number of forwarded messages in the network on the number of blocks in the Blockchain and the required time to mine. (**a**) Required time to mine; (**b**) Number of blocks in the Blockchain.

Figure 12 shows the accuracy and precision for different event rate and number of generated blocks. As shown, by increasing number of generated block in the system, the accuracy and precision increase.




(**a**)




(**b**)

**Figure 12.** Accuracy and precision of the proposed system for different event rates and number of generated blocks. (**a**) Model accuracy; (**b**) Model precision.

### *5.3. Comparison to Proposed Model Computing Time*

To prove the optionality of the proposed model, the comparison result of computation times of the proposed model with the computational time of the vehicular network based on PoW is mentioned. The results can be seen in Figures 13 and 14.

Figure 13 compares generation block time in Sharding algorithm with POW as the most well-known consensus algorithm. As shown in this figure, by increasing number of vehicles, the required time to mine a new block in the proposed system, in contrast with POW, increases near linearly.

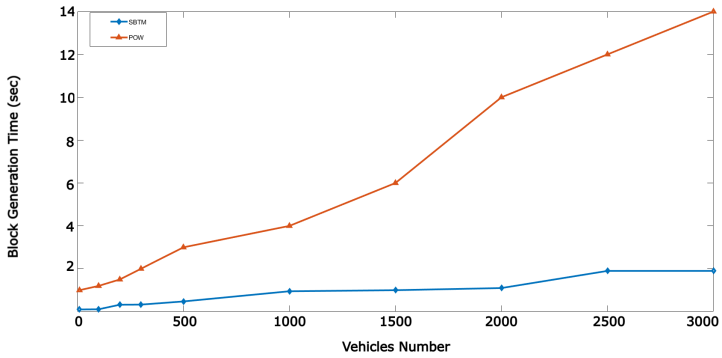



**Figure 13.** Comparison generation block time between Sharding and POW.

Figure 14 shows the calculation time for vehicles. It is clear from the diagram that the vehicle calculation time is less in our proposed model. As a result, it shows the success of this method compared to previous methods.

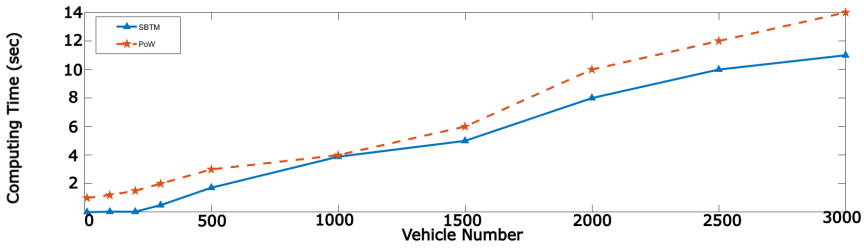



**Figure 14.** Comparison vehicles calculation time between Sharding and POW.

## 6. Comparison

As mentioned in [37], Barttholomy et al. have used tangle technology to improve the safety and efficiency of cars in the VANET. They introduced and analyzed tangle as a solution to the Blockchain problems in the VANET. Tangle technology uses the coordinator to manage the blockchain. However, as the coordinator is not distributed, it represents a single-point-of-failure. However, in this paper, the sharding algorithm is used, which is fully distributed. In [38], Kang et al. proposed a Blockchain-based structure for VANET that miners are selected based on their cooperation in this system. The concept of this work is based on POS blockchain. However, POS still has a sclability problem, which prevents vehicle transactions from being mined in near real time by the RSUs. However, with the proposed method, it is possible to add extra nodes to the network without having to worry about the delay in publishing blocks. Another study by Yang et al. [33] proposed a distributed trust management system for vehicle network based on Blockchain technology. The PoW algorithm was used for mining, which has a scalability problem. In addition, not all transactions are accessible to all RSUs, which means that there is a possibility of creating a fake transactions and blocks in order to receive a blockchain reward. In this paper, we proposed to broadcast transactions and using committees to solve this issue.

## 7. Conclusions

In this study, we propose a scalable trust management system based on Blockchain for VANET. In this system, vehicles can report the events to their neighbors, assess the message reliability received from neighbors, query the net reliability from the nearest RSU, and calculate trust on the RSUs. Net reliability values are calculated by RSUs and validating by RSUs committees. Each net reliability value, after approving by the committee, is added to the Blockchain. In the proposed system, all RSUs collaboratively maintain the latest updated Blockchain. Simulating demonstrated that malicious agents cannot disturb normal system behavior. Experimental results indicate the effectiveness and performance of proposed system in different situations. Further studies, such as a reward and punishment

method for malicious agents, can be applied to the proposed system as future work. We believe that the proposed system can be applied for decentralized trust management in the other applications. For future work, we plan to use Q-Learning to smarten vehicle behavior in the proposed model based on rewards and punishments.

**Author Contributions:** Conceptualization, F.G.G. and J.S.S.; methodology, F.G.G.; software, F.G.G.; validation, F.G.G., J.S.S. and A.S.; investigation, F.G.G.; writing—original draft preparation, F.G.G.; writing—review and editing, J.S.S. and A.S.; visualization, F.G.G.; supervision, J.S.S. and A.S. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research received no external funding.

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Not applicable.

**Data Availability Statement:** Not Applicable, the study does not report any data.

**Conflicts of Interest:** The authors declare no conflict of interest.

## References

1. Sikora, A. Privacy and Trust Management in Safety-Related Car2X Communication. In *Managing Trust in Cyberspace*; Chapman and Hall/CRC: Boca Raton, FL, USA, 2013; pp. 284–304.
2. Rocha, H.; Ducasse, S. Preliminary steps towards modeling blockchain oriented software. In Proceedings of the 2018 IEEE/ACM 1st International Workshop on Emerging Trends in Software Engineering for Blockchain (WETSEB), Gothenburg, Sweden, 27 May–3 June 2018; pp. 52–57.
3. Raikwar, M.; Mazumdar, S.; Ruj, S.; Gupta, S.S.; Chattopadhyay, A.; Lam, K.Y. A blockchain framework for insurance processes. In Proceedings of the 2018 9th IFIP International Conference on New Technologies, Mobility and Security (NTMS), Paris, France, 26–28 February 2018; pp. 1–4.
4. Mettler, M. Blockchain technology in healthcare: The revolution starts here. In Proceedings of the 2016 IEEE 18th International Conference on e-Health Networking, Applications and Services (Healthcom), Munich, Germany, 14–16 September 2016; pp. 1–3.
5. Huh, S.; Cho, S.; Kim, S. Managing IoT devices using blockchain platform. In Proceedings of the 2017 19th International Conference on Advanced Communication Technology (ICACT), Yeongchang, Korea, 19–22 February 2017; pp. 464–467.
6. Chen, Y.; Li, Q.; Wang, H. Towards trusted social networks with blockchain technology. *arXiv* **2018**, arXiv:1801.02796.
7. Wright, C.; Serguieva, A. Sustainable blockchain-enabled services: Smart contracts. In Proceedings of the 2017 IEEE International Conference on Big Data (Big Data), Boston, MA, USA, 11–14 December 2017; pp. 4255–4264.
8. Moura, T.; Gomes, A. Blockchain voting and its effects on election transparency and voter confidence. In Proceedings of the 18th Annual International Conference on Digital Government Research, Staten Island, NY, USA, 7–9 June 2017; pp. 574–575.
9. Jacobovitz, O. *Blockchain for Identity Management, The Lynne and William Frankel Center for Computer Science Department of Computer Science*; Ben-Gurion University: Beer Sheva, Israel, 2016.
10. Szalachowski, P. Blockchain-based tls notary service. *arXiv* **2018**, arXiv:1804.00875.
11. Shafagh, H.; Burkhalter, L.; Hithnawi, A.; Duquennoy, S. Towards blockchain-based auditable storage and sharing of IoT data. In Proceedings of the 2017 on Cloud Computing Security Workshop, Dallas, TX, USA, 3 November 2017; pp. 45–50.
12. Mengelkamp, E.; Notheisen, B.; Beer, C.; Dauer, D.; Weinhardt, C. A blockchain-based smart grid: towards sustainable local energy markets. *Comput. Sci.-Res. Dev.* **2018**, *33*, 207–214.
13. Abeyratne, S.A.; Monfared, R.P. Blockchain ready manufacturing supply chain using distributed ledger. *Int. J. Res. Eng. Technol.* **2016**, *5*, 1–10.
14. Holland, M.; Stjepandić, J.; Nigischer, C. Intellectual property protection of 3D print supply chain with blockchain technology. In Proceedings of the 2018 IEEE International Conference on Engineering, Technology and Innovation (ICE/ITMC), Stuttgart, Germany, 17–20 June 2018; pp. 1–8.
15. Zamani, M.; Movahedi, M.; Raykova, M. Rapidchain: Scaling blockchain via full sharding. In Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security, Toronto, ON, Canada, 15–19 October 2018; pp. 931–948.
16. Manshaei, M.H.; Jadliwala, M.; Maiti, A.; Fooladgar, M. A game-theoretic analysis of shard-based permissionless blockchains. *IEEE Access* **2018**, *6*, 78100–78112.
17. Yaga, D.; Mell, P.; Roby, N.; Scarfone, K. Blockchain technology overview. *arXiv* **2019**, arXiv:1906.11078.
18. Li, D.; Peng, W.; Deng, W.; Gai, F. A blockchain-based authentication and security mechanism for IoT. In Proceedings of the 2018 27th International Conference on Computer Communication and Networks (ICCCN), Hangzhou, China, 30 July–2 August 2018; pp. 1–6.
19. Bamakan, S.M.H.; Motavali, A.; Bondarti, A.B. A survey of blockchain consensus algorithms performance evaluation criteria. *Expert Syst. Appl.* **2020**, *154*, 113385.

20. Sankar, L.S.; Sindhu, M.; Sethumadhavan, M. Survey of consensus protocols on blockchain applications. In Proceedings of the 2017 4th International Conference on Advanced Computing and Communication Systems (ICACCS), Coimbatore, India, 6–7 January 2017; pp. 1–5.
21. Namal, S.; Gamaarachchi, H.; MyoungLee, G.; Um, T.W. Autonomic trust management in cloud-based and highly dynamic IoT applications. In Proceedings of the 2015 ITU Kaleidoscope: Trust in the Information Society (K-2015), Barcelona, Spain, 9–11 December 2015 ; pp. 1–8.
22. Nitti, M.; Girau, R.; Atzori, L.; Iera, A.; Morabito, G. A subjective model for trustworthiness evaluation in the social internet of things. In Proceedings of the 2012 IEEE 23rd International Symposium on Personal, Indoor and Mobile Radio Communications-(PIMRC), Sydney, NSW, Australia, 9–12 September 2012; pp. 18–23.
23. Minhas, U.F.; Zhang, J.; Tran, T.; Cohen, R. Towards expanded trust management for agents in vehicular ad-hoc networks. *Int. J. Comput. Intell. Theory Pract. IJCITP* **2010**, *5*, 3–15.
24. Gerlach, M. Trust for vehicular applications. In Proceedings of the Eighth International Symposium on Autonomous Decentralized Systems (ISADS'07), Sedona, AZ, USA, 21–23 March 2007; pp. 295–304.
25. Raya, M.; Papadimitratos, P.; Gligor, V.D.; Hubaux, J.P. On data-centric trust establishment in ephemeral ad hoc networks. In Proceedings of the IEEE INFOCOM 2008-The 27th Conference on Computer Communications, Phoenix, AZ, USA, 13–18 April 2008; pp. 1238–1246.
26. Dotzer, F.; Fischer, L.; Magiera, P. Vars: A vehicle ad-hoc network reputation system. In Proceedings of the Sixth IEEE International Symposium on a World of Wireless Mobile and Multimedia Networks, Taormina-Giardini Naxos, Italy, 16 June 2005; pp. 454–456.
27. Yu, B.; Singh, M.P. An evidential model of distributed reputation management. In Proceedings of the First International Joint Conference on Autonomous Agents and Multiagent Systems: Part 1, Bologna, Italy, 15–19 July 2002; pp. 294–301.
28. Yu, B.; Singh, M.P. A social mechanism of reputation management in electronic communities. In *Proceedings of the International Workshop on Cooperative Information Agents, Boston, MA, USA, 7–9 July 2000*; Springer: Berlin/Heidelberg, Germany, 2000; pp. 154–165.
29. Azad, M.A.; Bag, S.; Parkinson, S.; Hao, F. TrustVote: privacy-preserving node ranking in vehicular networks. *IEEE Internet Things J.* **2018**, *6*, 5878–5891.
30. Maskey, S.R.; Badsha, S.; Sengupta, S.; Khalil, I. ALICIA: Applied Intelligence in blockchain based VANET: Accident Validation as a Case Study. *Inf. Process. Manag.* **2021**, *58*, 102508.
31. Castaño, F.; Haber, R.E.; Mohammed, W.M.; Nejman, M.; Villalonga, A.; Lastra, J.L.M. Quality monitoring of complex manufacturing systems on the basis of model driven approach. *Smart Struct. Syst.* **2020**, *26*, 495–506.
32. Villalonga, A.; Negri, E.; Biscardo, G.; Castano, F.; Haber, R.E.; Fumagalli, L.; Macchi, M. A decision-making framework for dynamic scheduling of cyber-physical production systems based on digital twins. *Annu. Rev. Control* **2021**, *51*, 357–373.
33. Yang, Z.; Yang, K.; Lei, L.; Zheng, K.; Leung, V.C. Blockchain-based decentralized trust management in vehicular networks. *IEEE Internet Things J.* **2018**, *6*, 1495–1505.
34. Islam, A.; Kader, M.; Shin, S.Y. NEWSTRADCOIN: A Blockchain Based Privacy Preserving Secure NEWS Trading Network. *arXiv* **2019**, arXiv:1904.00184.
35. Jiang, T.; Fang, H.; Wang, H. Blockchain-based internet of vehicles: Distributed network architecture and performance analysis. *IEEE Internet Things J.* **2018**, *6*, 4640–4649.
36. Kang, J.; Yu, R.; Huang, X.; Wu, M.; Maharjan, S.; Xie, S.; Zhang, Y. Blockchain for secure and efficient data sharing in vehicular edge computing and networks. *IEEE Internet Things J.* **2018**, *6*, 4660–4670.
37. Bartolomeu, P.C.; Vieira, E.; Ferreira, J. IOTA Feasibility and Perspectives for Enabling Vehicular Applications. In Proceedings of the 2018 IEEE Globecom Workshops (GC Wkshps), Abu Dhabi, United Arab Emirates, 9–13 December 2018; pp. 1–7.
38. Kang, J.; Xiong, Z.; Niyato, D.; Ye, D.; Kim, D.I.; Zhao, J. Toward Secure Blockchain-Enabled Internet of Vehicles: Optimizing Consensus Management Using Reputation and Contract Theory. *IEEE Trans. Veh. Technol.* **2019**, *68*, 2906–2920.
39. Azad, M.A.; Bag, S.; Hao, F.; Salah, K. M2m-rep: Reputation system for machines in the internet of things. *Comput. Secur.* **2018**, *79*, 1–16.
40. Kaaniche, N.; Laurent, M. A blockchain-based data usage auditing architecture with enhanced privacy and availability. In Proceedings of the 2017 IEEE 16th International Symposium on Network Computing and Applications (NCA), Cambridge, MA, USA, 30 October–1 November 2017; pp. 1–5.
41. Gao, F.; Zhu, L.; Shen, M.; Sharif, K.; Wan, Z.; Ren, K. A blockchain-based privacy-preserving payment mechanism for vehicle-to-grid networks. *IEEE Netw.* **2018**, *32*, 184–192.
42. Gibbs, M.N.; MacKay, D.J. Variational Gaussian process classifiers. *IEEE Trans. Neural Netw.* **2000**, *11*, 1458–1464.
43. Bejan, A.; Lorente, S. Constructal law of design and evolution: Physics, biology, technology, and society. *J. Appl. Phys.* **2013**, *113*, 6.
44. Sartakhti, J.S.; Manshaei, M.H.; Basanta, D.; Sadeghi, M. Evolutionary emergence of angiogenesis in avascular tumors using a spatial public goods game. *PLoS ONE* **2017**, *12*, e0175063.

45. Luu, L.; Narayanan, V.; Zheng, C.; Baweja, K.; Gilbert, S.; Saxena, P. A secure sharding protocol for open blockchains. In Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security, Vienna, Austria, 24–28 October 2016; pp. 17–30.
46. Castro, M.; Liskov, B. Practical Byzantine fault tolerance. In Proceedings of the Third Symposium on Operating Systems Design and Implementation, New Orleans, LA, USA, 1 February 1999; Volume 99, pp. 173–186.