Hindawi
Wireless Communications and Mobile Computing
Volume 2022, Article ID 1644851, 17 pages
https://doi.org/10.1155/2022/1644851

Research Article

MPoR: A Modified Consensus for Blockchain-Based Internet of Vehicles

Bochuan Hou[1] **Hongliang Zhu**[1] **Yang Xin,**[1] **Jianyu Wang**[2] **and Yixian Yang**[1]

[1] *National Engineering Laboratory for Disaster Backup and Recovery, Information Security Center, School of Cyberspace Security, Beijing University of Posts and Telecommunications, Beijing 100876, China*
[2] *Beijing University of Posts and Telecommunications, Beijing 100876, China*

Correspondence should be addressed to Hongliang Zhu; zhuhongliang@bupt.edu.cn

Received 29 May 2022; Revised 27 June 2022; Accepted 5 July 2022; Published 12 August 2022

Academic Editor: Zhiguo Qu

Copyright © 2022 Bochuan Hou et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Ride-sharing services, such as ride-hailing and carpooling, have become attractive travel patterns for worldwide users. Due to the high dynamic topology, heterogeneous wireless communication mode, and centralization, the Internet of Vehicles (IoV) is much more vulnerable to security issues such as privacy theft, single point of failure, data island, and unauthorized access, resulting in great security risks, while ride-sharing services provide convenience. Blockchain technology used to solve the security problems of the IoV has become a current research hotspot, including authentication and privacy protection. Nevertheless, the existing algorithms still face challenges such as large amount of computation, low throughput, low scalability, consensus, and node security. Achieving an efficient, lightweight, and scalable secure blockchain–based IoV system still needs to be solved urgently. In this paper, we propose an effective consensus algorithm called Modified Proof of Reputation (MPoR). Firstly, by using the average network access time of the whole network nodes as the filtering threshold, the number of consensus nodes can be controlled adaptively. Then, a new multiweight reputation algorithm is proposed to quantify the reputation value of nodes, so as to detect and eliminate malicious nodes in the consensus node pool. Theoretical analysis and extensive simulation experiments reflect that under the IoV scenario, MPoR can adaptively select the number of consensus nodes, to effectively improve the consensus efficiency. When malicious nodes are less than 1/3 of the total nodes in the network, MPoR can effectively resist latent attack and collusive attack and has strong robustness.

1. Introduction

IoV is a typical application scenario for IoT technology. As an emerging concept, IoV is considered to help realize the vision of intelligent transportation system (ITS). Vehicles are equipped with high-tech equipment, such as GPS and radar, which can help realize the interconnection between vehicle and everything (V2X) through a variety of communication methods, to form a self-organizing network called Vehicular Ad-Hoc Network (VANET). With the development of the sharing economy and VANET, ride-sharing services such as ride-hailing and carpooling have become a common travel method. Drivers can find and provide driving services either to passengers with requirements precisely or to passengers with similar travel plans. The ride-sharing service has significant social efficiency: it can not only alleviate traffic congestion but also reduce the economic burden of people, as well as achieve environmental gain by reducing vehicle emissions and noise [1]. Currently, almost all ride-sharing service platforms are designed and operated on centralized systems, which rely on a trusted third party to store and process user travel information and transactions. Management is not transparent; the central node has all the control rights to decide policies and service conditions [2]. Once the central node has been attacked, the whole system will no longer ensure security. It is reported that many safety issues have occurred: carpool data was leaked to unauthorized and untrusted third party through Elastic Compute Service. A company has been exposed to serious violations of using the cloud platform to collect and disclose users' private data to certain organizations. Various facts indicate that existing service platforms have potential safety hazards. Blockchain is a novel exploration of network world operating rules and technologies [3]. After a new typical cryptocurrency, Bitcoin was proposed [4], and scholars have noticed the advantages of blockchain in the realization of distributed security. In blockchain-based networks, data is stored in the form of distributed ledger. Each trusted node has its own copy and cooperate with each other to maintain secure operation of the network. A lot of blockchain-based researches have been carried out in order to achieve the vision of realizing the smart city, including P2P transaction [5], medical care [6], supply chain management [7], smart grid [8], federal learning [9], and IoV [10]. In IoV network, with the help of blockchain technology, each node can save a copy of travel transaction records. The central system decentralizes the control tight, so that passengers and vehicles can communicate directly, decreasing the management cost. Transaction records are kept in the chain and can ensure security. Meanwhile, it can greatly improve the performance of the ride-sharing system in terms of computational and communication overhead. In ride-sharing system in existence, passengers send riding request to the central management platform, which helps matching passengers with appropriate drivers. After the matching, both the passenger and the driver will receive detailed information of each other, including pick-up location, planned route, expected delivery time, and total amount. When the sharing service is completed, passenger pays to the central system and then transfers to the driver's wallet account. It can be seen that there are massive issues existing. The central node has all the information from passengers and drivers, which makes it easy to suffer from a single point of failure. The opacity of matching and charging operations makes it prone to information monopoly or information island. Meanwhile, the system is computationally intensive and inefficient. There are unresolved challenges for the blockchain-based ridesharing network model as follows:

(i) Blockchain is essentially a circular system, with nodes accessing to the network, uploading transaction records to blocks, block consensus, and block chaining. Adaptability can monitor the major changes in each cycle of the system and take actions to implement the decision [11], so as to reduce management complexity, dynamically adjust operation process, and increase robustness. The data communicated in V2X include not only individual information, such as transaction information and service rating, but also traffic events, such as observation of serious traffic events. Before making emergency response, it is necessary to conduct consensus verification at the edge layer, evaluating the severity and site status of the event on the premise of ensuring the authenticity and accuracy of the event. In this case, the algorithm that can help adaptively adjust the size of the consensus group is meaningful and essential. Existing consensus algorithms are either to find a single miner (such as Proof of Work, Proof of Stake, and Proof of Elapsed Time) or to find a group of miner that meet certain conditions (such as Practical Byzantine Fault Tolerance and zyzzyva). The size of the consensus group is not controllable; therefore, the system cannot dynamically balance efficiency and security

(ii) Most of the existing consensus filtering mechanisms are based on information that can be eavesdropped, such as the distance traveled [12] and the number of digital signatures [13]. If there are collusive adversaries who can exchange information with each other in the network, these adversaries can share and unify information they have got and try to get selected in a consensus simultaneously and then implement destructions

(iii) The consensus algorithm based on simple stochastic filtering may reduce the initiative of honest users to participate in consensus (some participants may have been stochastic filtered several times without receiving any benefits). At the same time, current consensus algorithms have low selection rate mostly. In the selection stage, it will cause plenty of waste of computing power and, consequently, reduce the desire of users to participate. How to improve the initiative of legitimate users under the premise of ensuring security remains an unresolved issue

In view of the above-mentioned challenges, we propose a new consensus algorithm for secure ride-sharing service, called Modified Proof of Reputation (MPoR). When designing MPoR, we consider the following features: (1) all nodes in the network have opportunities to participate in the consensus; (2) absolute fairness for miners in the selection stage; (3) lightweight and low computation overhead; and (4) high scalability. In summary, our specific constructions mainly include as follows:

(i) Proposed a new consensus algorithm called MPoR for the consortium blockchain-based ride-sharing network system. Use the hash value as the quantization standard for stochastic filtering, which significantly reduces the amount of consensus nodes, as well as achieves self-adaptive control of the size of consensus pool. Compared with existing algorithm, MPoR is fairer and more scalable

(ii) Designed a new reputation rating algorithm based on multifactors, including miles traveled as established route, observation of interactions, service rating by passengers, and accumulated reputation of validate blocks. Multiweight reputation algorithm can make up for the deficiency of single-factor evaluation criteria, improve the performance of the algorithm, eliminate low reputation nodes (could be adversaries or broken nodes), therefore optimize the consensus node pool, and encourage nodes to provide positive activities for the system

(iii) Implemented MPoR systematically in GoLand. Several simulation experiments and security analysis are carried out, indicating that the proposed algorithm is effective, is efficient, and has higher security and scalability

2. Related Work

Due to the high mobility, frequently changing topology, and open and diverse communication environment, compared to other IoT network models, IoV needs more edge computing to assist content delivery. Roadside unit (RSU) and vehicle can be considered edge nodes. Instead of uploading data to a cloud server, they can directly participate in data verification and transmission, so as to realize decentralization, lower delay, and utilization of idle resources. A large number of researchers have conducted relevant research on edge computing in lightweight [14, 15], efficiency [16], and improving QoS [17, 18].

The characteristics of blockchain include decentralization, irrevocability, traceability, transparency, autonomy, and anonymity, which can meet the requirements for implementing edge computing in IoV. In an effort to achieve security, scalability, and efficiency, a cohort of scholars has proposed new blockchain-based IoV application methods based on the research directions of validation, consensus, and reputation. The MPoR consensus algorithm proposed in this paper is applied to consortium blockchain-based IoV network; the study involved consensus mechanism and reputation-based incentive mechanism.

For blockchain-based secure ride-sharing service model, Wang and Zhang [24] proposed a secure data sharing scheme based on the consortium blockchain, guarantee the confidentiality and privacy of data interaction through the attribute-based proxy reencryption algorithm, and reveal the true identity of malicious users through the reputation rating. Renu and Banik [25] implement the minimum matching algorithm to match the ride-sharing request through the smart contract. Li et al. [26] use the blockchain to assist in fog calculation to store carpool records. Zhang et al. [27] proposed a smart contract-based secure billing protocol to negotiate pick-up location, route, and price in advance. Li and Wang [19] proposed a location privacy protection scheme called MinHash to hide the user's actual geographical location and optimize the similarity between user and driver feature vectors. The disadvantage of the above methods is that the consensus adopts traditional PoW or PoS, which is inefficient.

In [12], the authors proposed a new consensus algorithm Proof of Driving (PoD), which is applied to the VANET. This algorithm can ensure the security of the system on the premise that the penetration attacker is less than 1/3 and the total reputation of honest nodes is greater than the total reputation of malicious nodes, but it cannot resist collusive attacks. Meanwhile, because the reputation resets to zero after each block generation, the system lacks an incentive mechanism and continuity.

Suo et al. [13] proposed a Proof of Travel (PoT) certification protocol based on Verifiable Vehicle Miles Traveled (VVMT). It is mainly aimed at the early deployment stage of IoV network, where there are not enough benign vehicles. Vehicles obtain VVMT by interacting with RSU during driving, and vehicles whose reputation values exceed the threshold can participate in consensus, thus increasing the attack cost of malicious vehicles. The inadequacy is that vehicles maintaining VVMT require a massive amount of communication overhead, and the feasibility of PoT depends on whether the RSUs are densely distributed.

To ensure security through reputation, Wang et al. [20] designed a reputation evaluation model based on blockchain to solve the security vulnerabilities and privacy problems in Autonomous Vehicle Social Network (AVSN) and stimulate the legal behavior and content delivery of vehicles. In [21], Yang et al. proposed a system to evaluate the credibility of vehicle network data based on blockchain. The reputation value is based on its historical information rating. Both scheme adopted PoW consensus, which is inefficient, so there are obstacles in the implementation.

In [22], Yuan and Wang proposed a contract-based security block verification incentive mechanism, which can encourage more miners to participate in block verification. The weighted subjective logic model is used to introduce a safe and efficient reputation management scheme, improve the DPoS consensus, and reduce collusion between stakeholders and mining candidates. This scheme can prevent attacks by less than 1/3 malicious miners. The disadvantage is that the accuracy of weight allocation is not considered.

In [23], the authors proposed a reputation system applied in ITS. Users interested in traffic information are regarded as the main participants of the architecture. Data are verified by crowd-sourcing and securely shared among legitimate users. The consensus is verified by cluster, and the consensus threshold is set. The disadvantage is that the network has a lower scalability and the consensus delay will increase significantly as the number of nodes increases.

The summary of related works based on several factors is listed in Table 1. We can see from the above summaries and Table 1 that existing works still have deficiency on incentive, performance optimization, or security optimization. Through MPoR consensus and multiweight reputation, our study can achieve improvements on the above concepts.

3. Basic Concept and Initial Setting for Blockchain-Based Ride-Sharing Network

3.1. System Components. Firstly, we clarify our purpose again: design a self-adaptive consensus algorithm for consortium blockchain-based ride-sharing network system. The key point is that the system can adaptively adjust the consensus scale according to the importance and requirements of event to be validated, so as to reach the balance of security and efficiency. We adapt a typical VANET distributed network model, including trusted authority (TA), road-side unit (RSU), vehicle node (full node), and passenger node (lightweight node). These infrastructures and devices communicate with each other through dedicated short-range communication (DSRC), long-term evolution (LTE), 4G, 5G, and other communication methods to share information. Figure 1 shows the composition and connections of

Table 1: Summarize of related works.

Reference	Consensus	Incentives	Performance optimization	Security optimization
[19]	PoW	—	—	Location privacy
[12]	PoD	—	Scalability	Expect collusive attack
[13]	PoT	Travel distance	—	Several kinds of attacks
[20]	PoW+PoS	Reputation	—	User privacy
[21]	PoW	Reputation	Efficiency	—
[22]	DPoS	Reputation	—	1/3 malicious nodes
[23]	PBFT	—	—	External attack
Our study	MPoR+PBFT	Multiweight reputation	Scalability, latency	Several kinds of attacks

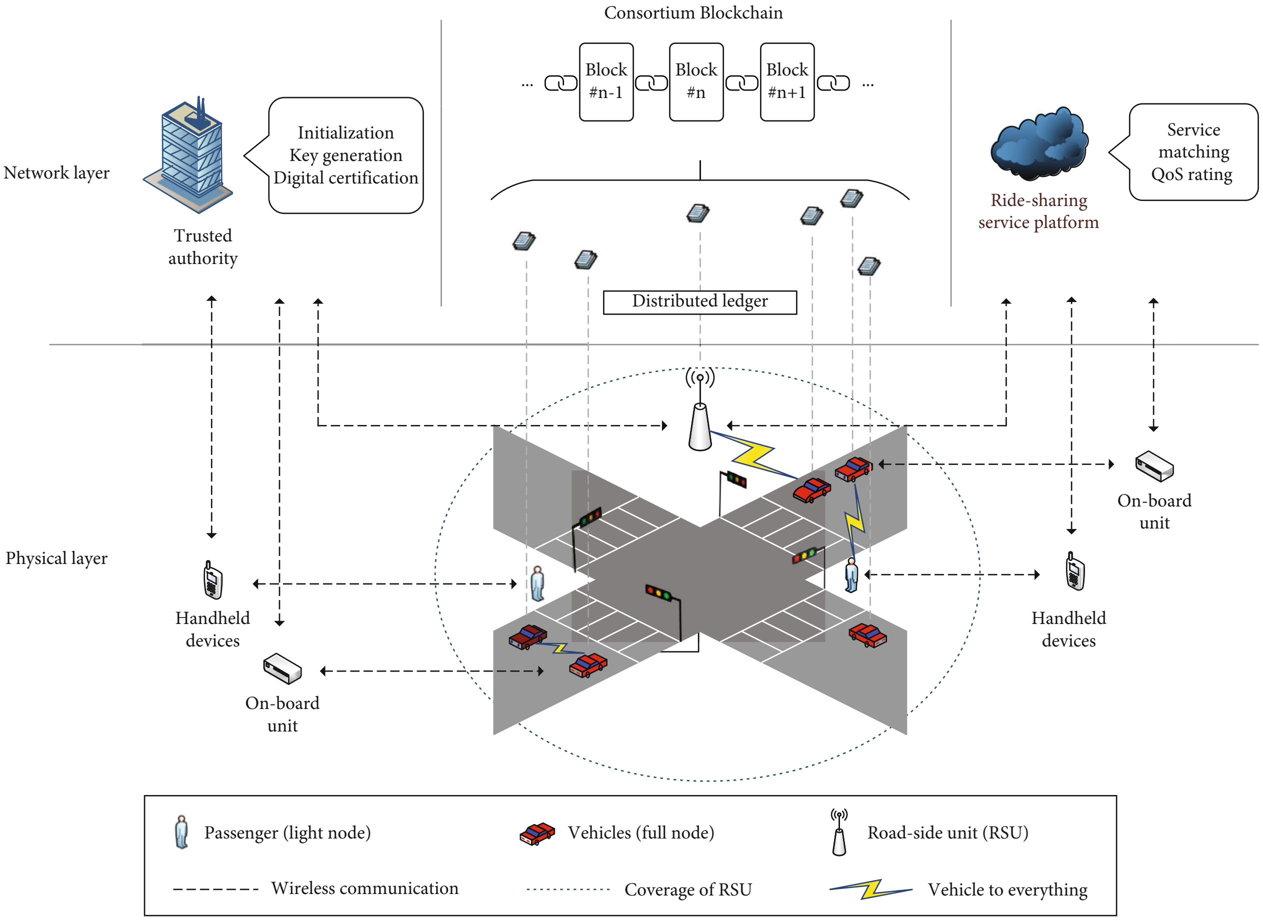


Figure 1: Overview of blockchain-based ride-sharing network.

the network between various components. The components are described in detail as follows.

TA: TA is a kind of infrastructure to ensure security, whose main responsibility is system initialization. Its functions include registration, certificate issuance, and key management. When RSU/users enter the network for the first time, it submits the registration request through secure channels (such as offline), and then, TA will generate a public-private key pair and digital certification. Users save the key pair and use their private keys to sign all transactions in a subsequent process. TA stays immune to all kinds of attack, and the contents given are reliable and trustworthy. Most of the time, TA remains offline. Only when the network finds malicious behaviors, TA will disclose the true identity of those nodes.

RSU: RSU is responsible for providing trustworthy identification as well as helping create communication between all vehicle nodes and passenger nodes under its coverage. It has sufficient computing power, network communication capability, and storage space and is well equipped with a Trusted Execution Environment (TEE). When the system needs to generate blocks, RSU will broadcast consensus invitation to the vehicle nodes within its coverage to participate. In our scheme, RSU remains trustworthy and participates in the whole consensus process, recording the access timestamp of all nodes, giving security identification, putting forward consensus requirements, storing and updating reputation ratings, and providing incentives.

Vehicle node: vehicles are equipped with an advanced communication device, wireless transmission module, and TEE and have rational computing power to execute simple calculations, such as verifying digital signatures and calculating travel routes. Vehicles work as full nodes for storing distributed ledger of blockchain. Vehicles use positive activities to accumulate reputation; when the system generates new blocks, vehicles can be selected for consensus as candidate miners based on its reputation and then improve their reputation ratings and obtain corresponding rewards through election.

Passenger node: passengers use applications in portable Internet-enabled devices to obtain ride-sharing services. These devices belong to the lightweight node and cannot save the distributed ledger. In this system, passenger nodes can only demand for ride-sharing service as well as give service rating after the ride is completed, unable to participate in consensus as candidates.

Consortium blockchain: blockchain stores individual pseudonyms, transaction records, and event information in the network. Under the ride-sharing service model, content stored in the block includes detailed ride records, transaction bills, and vehicle reputation values. After registration, any user can view nonprivate information on the block and obtain services [12].

Ride-sharing service platform: ride-sharing service platform implements as a decentralized application based on blockchain. It is an interface for both vehicles and passengers to transact automatically, including service matching and QoS rating.

3.2. Threat Model. Assuming that adversaries work as a group and driven by interests, its main purpose is to gain economic benefits through various means and ruin the security of the system comes second. Whether it is an internal attacker with credentials or a controlled broken node, if they enter the final step of consensus, they would send forged information and give opposite validation results to do evil. Meanwhile, it is assumed that those adversaries are economically rational; that is, they will pay attention to their attack cost. They will not spend massive cost to do evil for a long period, and due to the self-check security of the system model, adversaries will be found immediately after doing evil. Depending on the means and purpose used, adversaries can

(i) control most vehicles to control the entire network using overwhelming computing power

(ii) eavesdrop the communication channel between the target vehicle and RSU, record the information content, and then modify its own information to the range of normal nodes to penetrate the system

(iii) collude with other malicious vehicles, exchange each other's vouchers and valid information, and obtain the same identification through same network behavior, through the stochastic filtering, enter the final consensus step, and then break the system as their wishes

(iv) lurk in the system, behave as normal nodes, and provide positive activities to accumulate reputation. Only after being selected into the consensus group will it start to do evil in order to gain benefits

In Section 4.5, we will describe in detail the impact of various attack methods of adversaries and give the defense methods of our algorithm as well.

3.3. Reputation Calculation Model. The reputation rating of each vehicle node can be composed of multiple factors, which are recorded in each communication with RSU as a part of vehicle status information. In this paper, we use miles traveled as established route, observation of interactions, service rating by passengers, and accumulated reputation of validate blocks as the value of vehicles' reputation. Only a part of vehicles with large reputation values are eligible to participate consensus. The specific description of each reputation factor is as follows:

Miles traveled as established route: when providing a ride-sharing service, the vehicle will report its location information in the continuous communication process with the RSU. The RSU checks whether the vehicle travels as the established route and then quantifies mileage into reputation value. With the same mileage, vehicle passes more RSUs will gain more reputation. The advantage of using mileage as reputation rating factor is that it can promote the network participation of vehicles. Disadvantages arise because it does not take into account that legal vehicles may have a short travel route due to practical factors and cannot obtain many reputation values, and malicious nodes can easily accumulate a large amount of reputation by this means.

Observation of interactions: when in the process of driving, vehicles will share each other's information with adjacent vehicles through V2V channel, including but not limited to driving status, road condition status, and adjacent geographic information, such as gas stations. Adjacent interactions will evaluate the status of the vehicle nodes during communications and then upload it to RSU. RSU will obtain the vehicle node observation rating using the subjective logic model [28]. Since it is impossible to predict whether the vehicles with which the current node interacts are malicious, the node may have incorrect ratings. Although the malicious nodes can be found after consensus immediately, it still has the risk of being attacked.

Service rating by passengers: passengers will rate vehicle service quality after each ride-sharing service. Since the rating is highly subjective, it is not advisable to use the service rating alone as a standard for evaluating reputation.

Accumulated reputation of validate blocks: after each round of block generation, the RSU will reward the nodes in the final consensus group with reputation points. This factor could incentivize nodes to participate consensus. The disadvantage is that since the selection of consensus miners is based on stochastic filtering, if a node is randomly filtered out several times at previous rounds of block generation, even if it is selected in the subsequent block generation stage, it will still be shaved because it does not have enough accumulated reputation. Therefore, using this as a sole factor is not advisable as well.

It can be seen that all kinds of current reputation evaluation have limitations in quantitative standards. Therefore, in this paper, we accumulate and normalize the overall reputation of vehicles through the four factors mentioned above; see Section 4.3 for specific algorithm.

3.4. System Model

3.4.1. System Initialization. Vehicle and passenger nodes that join the system for the first time submit their identity information to the TA. After TA validates their identity, it issues pseudonym, digital certificate, and generate private-public key pair through elliptic curve to complete identity registration. The details of the nodes are stored in the ledger of the blockchain in the form of {public key | private key | pseudonym | network access time | hash value| signed (public key |private key|pseudonym|network access time|hash value)} as a transaction.

3.4.2. Joining the System for the First Time. Whenever any node enters the current RSU coverage, it communicates with RSU; sends its pseudonym, current timestamp, and travel record; and signs it. When the RSU receives the message, it verifies the signature. After confirming the identity information of the vehicle node, it assigns a unique ID to it as the sole special mark in the consensus stage. Meanwhile, it generates a block to record the information of nodes.

3.4.3. Activities in the Network. The activities of nodes in the network can be divided into three parts, including (1) vehicle and passenger nodes record their travel records, including departure and arrival location, time, driving route, and transaction amount. In addition, passenger nodes will also rate the service quality when service has finished. (2) When a vehicle observes events like traffic jams, traffic accidents, and improper behaviors of other vehicles, it records and then broadcasts the information to RSU and other nearby vehicle nodes. (3) When a vehicle node is about to leave the current RSU coverage, it communicates with the current RSU, records its activities within the coverage of the RSU, and records including ride-sharing service, consensus participation, and accumulated reputation. RSU signs and replies after verification. When the node enters coverage of new RSU, it communicates to the current RSU and RSU verifies the signature and then assigns a new unique ID to vehicle node.

3.4.4. Miner Selection. When the system collects enough records and needs to generate a new block, the RSU broadcasts the invitation to the vehicle nodes within its coverage, and vehicle nodes that respond within a limited time will become miner candidates. All mining candidates use the improved scheme proposed in this study to select the final consensus miners. The details of the algorithm will be given in Sections 4.2 and 4.3.

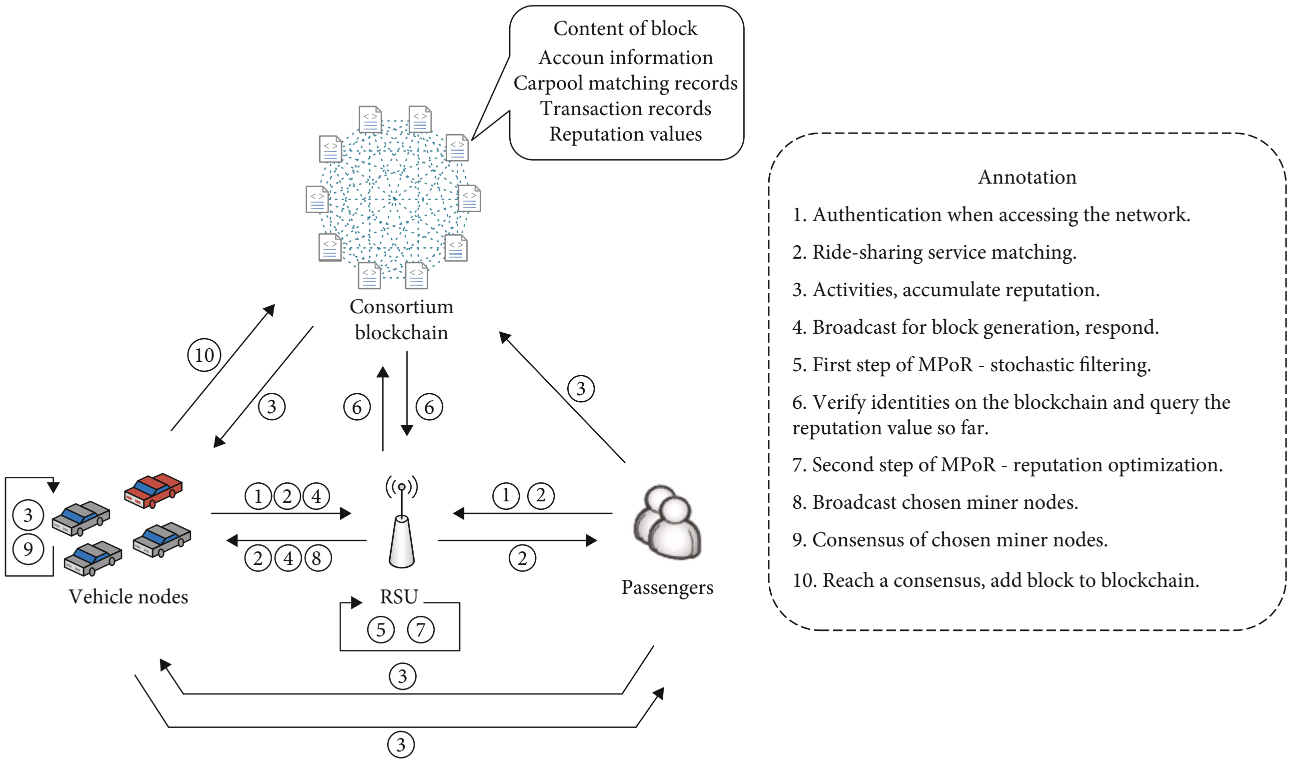
3.4.5. Consensus and Block Generation. The final consensus uses PBFT as the basic protocol. After the miners are obtained through MPoR, the PBFT protocol is used to generate and broadcast blocks. When the generation is complete, the reputation of the miners will be updated based on the contribution. Reputation will have an impact on the next mining selection. The specific implementation process of PBFT algorithm is also given in Section 4.4.

3.4.6. Termination. There are two type of terminations, including (1) network will reset reputation values of all nodes every time after several blocks generated (the number depends on the network status). The purpose of a reset is to prevent malicious nodes from accumulating reputation values and to promote the continuity of legal behaviors. Historical reputation values will be retained on the blockchain and can be used as currency to participate in real transactions to reward positive activities. (2) No matter whether a vehicle node goes off autonomously or is found to be a faulty node and forced to go off by the system in the consensus. At this time, its key pair and digital certificate must be revoked, and the node cannot participate in network activities. When a node wants to reenter the network, it needs to reregister with TA. Vehicle information will be permanently retained on the blockchain.

4. Design of Modified Proof of Reputation Algorithm

4.1. System Specific Settings. The data communicated in V2X include not only individual information, such as transaction information and service rating, but also traffic events, such as observation of serious traffic events. Before making emergency response, it is necessary to conduct consensus verification at the edge layer, evaluating the severity and site status of the event on the premise of ensuring the authenticity and accuracy of the event. In this case, the algorithm that can help adaptively adjusting the size of the consensus group is meaningful and essential. Our proposed algorithm MPoR can encourage nodes to accumulate reputation through legitimate activities in the network to compete for nominations on the premise of ensuring fairness and randomness. Everyone in the network is eligible to participate in miner competition, but only nodes chosen by MPoR can participate in consensus. The size of miner's pool of consensus is adaptively adjusted by MPoR, balancing efficiency, and accuracy according to the event state. MPoR includes two parts: stochastic filtering and reputation optimization. Figure 2 shows the processes for implement MPoR; specific process is discussed in Sections 4.2 and 4.3 separately. Firstly, we clarify the initial assumptions and settings of MPoR algorithm as follows:

(i) In our proposed system, TA and RSU are trustworthy and will not be attacked. In this scheme, the security of RSU will not be considered. The secure



Annotation

1. Authentication when accessing the network.
2. Ride-sharing service matching.
3. Activities, accumulate reputation.
4. Broadcast for block generation, respond.
5. First step of MPoR - stochastic filtering.
6. Verify identities on the blockchain and query the reputation value so far.
7. Second step of MPoR - reputation optimization.
8. Broadcast chosen miner nodes.
9. Consensus of chosen miner nodes.
10. Reach a consensus, add block to blockchain.

Figure 2: Frameworks for ride-sharing services based on MPoR.

consensus of RSU will be the subsequent research goal

(ii) Assuming that the transmission efficiency of all transmission channels are 100%, all information sent by each node can be reliably transmitted, and adversaries can only eavesdrop and cannot modify the broadcast information

(iii) The identification vehicle node obtained from RSU must be accurate and unique. It is stored in the TEE and cannot be modified by anyone

(iv) RSU will adaptively adjust the size of the miner's pool for the next time of consensus according to the current generation status. Adjustment conditions involve time, computing power, event coverage, and degree of impact. Our scheme will not discuss how it implement, but only the possibility that the proposed algorithm can adaptively adjust the size of the consensus miner's pool

(v) There are latent adversaries in the system. Their goal is to participate in consensus and then obtain economic benefits through giving false information. If the economic benefit is less than the cost incurred, the attack will be abandoned at once

(vi) There are malicious nodes in the system. Their goal is to destroy the system. Since our scheme still adopts PBFT as the consensus underlying algorithm, it is assumed that the number of malicious nodes should not exceed 1/3 of the total nodes. In subsequent section, we will theoretically prove that when the total number of attackers in the network does not exceed 1/3, the number of malicious nodes in the final consensus group will not exceed 1/3 as well, meeting the premise of safe operation of PBFT

Table 2: Key notations.

Notation	Definition
$\mathrm{PK}_{\mathrm{RSU}_i}, \mathrm{SK}_{\mathrm{RSU}_i}$	Public-private key pair of RSU_i
$\mathrm{PK}_{v_i}, \mathrm{SK}_{v_i}$	Public-private key pair of v_i
Add_i	Wallet address of v_i
$\mathrm{Hash}(x)$	Hash function of x
$M\{x\}$	Digital signature of x
U_{id_i}	Unique identification of v_i
N, N_a, N_n	Amount of all/adversary/normal nodes
α	Percentage of nodes required for block generation
T_i, T_{ave}	Timestamp/average timestamp of accessing the network
$L_{\mathrm{abs}}, L_{\mathrm{sf}}, L_{\mathrm{chosen}}$	List of certain pool
d_i, o_i, s_i, p_i	Reputation value
σ, ρ	Normalization factor
$\gamma, \epsilon, \zeta, \eta, \tau$	Weight factor

The key notations used in this paper are listed in Table 2.

```
Input: List of vehicles $L_N$, Percentage parameter $\alpha$
Output: List of vehicles after stochastic filtering $L_{sf}$
Calculate average timestamp from Equation (2)
Calculate target hash from Equation (3)
for $i = 0 ; i < N ; i + +$ do
    Calculate hash of $V_i$:
        $HashV_i = Hash(T || U_{id_i})$
    Calculate absolute difference value between
        $Targethash$ and $HashV_i$
    Add in list $L_{abs}$
end for
Sort $L_{abs}$ in ascending order
for $i = 0 ; i < \alpha N ; i + +$ do
    Add $L_{abs}(i)$ into $L_{sf}$
end for
Broadcast chosen result
return List $L_{sf}$
```

Algorithm 1: Stochastic filtering.

4.2. First Step of MPoR: Stochastic Filtering. When the vehicle node V_i enters the current RSU coverage, it communicates with the RSU and provides status information S_{it} at the access time t, which is defined as follows:

$$S_{it} = \mathrm{PK}_{\mathrm{RSU}_i} || \mathrm{PK}_{v_i} || \mathrm{Add}_i || \mathrm{hash}(S_i) || M\left\{\mathrm{PK}_{\mathrm{RSU}_i} | \mathrm{PK}_{v_i} | \mathrm{Add}_i | \mathrm{hash}(S_i)\right\}. \tag{1}$$

$\mathrm{PK}_{\mathrm{RSU}_i}$ can be ignored if it is the first communication after accessing the network firstly or network reset. S_{it} includes the historical cumulative reputation and the real-time reputation of v_i. After the RSU verifies S_{it}, a special identification U_{id_i} corresponding to v_i is generated according to the timestamp and access order when entering the network. There is no association between any two U_{id_i}, and it cannot be derived.

At certain time, RSU broadcasts the block generation request to all nodes within its coverage, and vehicle nodes that reply within a limited time will become miner candidates. RSU then calculates the average network access time T_{ave},

$$T_{\mathrm{ave}} = \frac{\sum_{i \in N} T}{N}, \tag{2}$$

and the percentage α of miner required for this time of generation in all candidate miners as the target threshold of stochastic filtering, that is,

$$\mathrm{Targethash} = \mathrm{Hash}(T_{\mathrm{ave}} || \alpha). \tag{3}$$

Hash() denotes hash function, which transforms any length of input into fixed length of output, and output is the hash value. Hash function is a kind of compressed mapping, and it is impossible for an attacker to obtain the specific contents of the input content from the hash value. In this paper, we adopted SHA256 (Secure Hash Algorithm 256) as the hash function. SHA256 transforms plain content to a 32-byte hash value. Then, we can quantizes hash values for comparison.

The number of miners required is determined according to the content of the block. As adversaries cannot control all online vehicles and RSU is secure, neither T_{ave} nor α can be predicted; that is, adversaries cannot know the target threshold of stochastic filtering.

Next, RSU calculates the hash value of U_{id_i}, then quantizes $\mathrm{Hash}(U_{\mathrm{id}_i})$ and Targethash into a computable number, compares the difference, records the absolute value of the difference, and stores it in L_{abs} in an ascending order. Then, select the first α part of vehicle nodes in L_{abs} according to the required amount and stored in L_{sf}. This part of the nodes will participate in the next step, while other nodes will end this selection time. It can be seen that the complexity of the stochastic filtering scheme is $O(N)$. The pseudocode of this step is provided in Algorithm 1.

Assuming the total number of malicious nodes f not greater than $(1/3)N$, so number of honest nodes a is $N - f$. Because filtering is completely stochastic and Targethash cannot be predicted by any node, the probability P_f of selecting malicious nodes is $(f * N_{\mathrm{sf}})/N$, which is less than $(1/3)N_{\mathrm{sf}}$. Therefore, when malicious nodes in the network do not exceed 1/3 of the total participating nodes, the security of the system can be guaranteed.

4.3. Second Step of MPoR: Reputation Optimization. When the stochastic filtering is completed, calculate the reputation value of the nodes in L_{sf} and select half of the nodes with the larger reputation value, so as to maximize the total reputation value of the final consensus group by limiting the number of nodes. Selected nodes will be acting as miners and generate block through PBFT. The pseudocode of this step is provided in Algorithm 2.

The reputation value can be composed of multiple factors. It is recorded in each communication with RSU as a part of vehicle status information and can be accumulated across different RSUs. After verification by the consensus group, it is added to the blockchain as a transaction. After the RSU completes several generations (recorded as one round), the system will reset the reputation value of each vehicle node so far (the reputation that has been added on the blockchain will be permanently retained as the historical reputation value) and start accumulating again.

In this study, we limit the range of values of each reputation value to normalize to [0,1], and the overall reputation of the vehicle can be calculated as

$$R_i = \sigma * (\gamma d_i + \epsilon o_i + \zeta s_i + \eta p_i), \tag{4}$$

where σ is the normalization factor, $\gamma, \epsilon, \zeta, \eta$ is the weight factor, and d_i, o_i, s_i, p_i represent miles traveled as established route, observation of interactions, service rating by passengers, and accumulated reputation of validate blocks separately.

Miles traveled as established route adopts Verifiable Vehicle Miles Traveled (VVMT) proposed in [13]. The vehicle moves along the predetermined path, obtains the

```
Input: List $L_{sf}$, Weight factor $\gamma, \epsilon, \zeta, \eta$
Output: List of vehicles after reputation optimization
    $L_{chosen}$
for $i = 0\,; i < len(L_{sf})\,; i + +$ do
    Calculate $d_i, o_i, s_i, p_i$ From equation (5), (9), (10), (11)
    Calculate normalized reputation from Equation (4)
    Add $R_i$ into List $L_{rep}$
    $R_{sum} + = R_i$
end for
Sort $L_{rep}$ in descending order
Calculate target reputation $TargetSum$
for $i = 0\,; i < len(L_{rep})\,; i + +$ do
    if $Sum_{current} > TargetSum$ then
        break
    else
        $Sum_{current} + = R_i$
        Add $v_i$ into $L_{chosen}$
    end if
end for
Broadcast $L_{chosen}$ to all nodes
return List $L_{chosen}$
```

Algorithm 2: Reputation optimization.

location certificate with digital signature from each RSU, and then quantifies the reputation through the location signature chain. The specific calculation is shown as follows:

$$d_i = f\left(\mathrm{LS}_i^T, n\right) = \rho \sum_t = 1^T d\left(ls_i^t - 1, ls_i^t\right) + (1-\rho)\frac{n}{n_{\max}}\bar{d}, \quad (5)$$

where ρ is the quantization factor, set to 0.5; d is the Euclidean distance; n is the total number of location signatures in the proof set ls_i^t held by v_i; $n_{\max}$ is the maximum number of location signatures that the vehicle can obtain in a certain area; and $\bar{d}$ is the average distance.

Observation of interactions adopts the subjective logic model proposed in [28]. The interactive vehicle v_j adjacent to v_i will evaluate according to the state of v_i during communication and then broadcast it to RSU. RSU obtains the observation rating of v_i through the subjective logic model combined with the observation opinions. Subjective logic uses a tuple $w_{j:i}$ to denote the degree of belief from j to i, defined as

$$w_{j:i} \equiv \left(b_{j:i}, d_{j:i}, u_{j:i}\right), \quad (6)$$

where$b_{j:i}, d_{j:i}, u_{j:i} \in [0, 1]$represent node$j$'s belief, disbelief, and uncertainty on node$i$.

Suppose RSU receives x interaction opinions on v_i; δ_x identify the weight factor of the recommender x, for each recommender $x \in X$, as follows:

$$\delta_x = \frac{b_{\mathrm{RSU}:j} \cdot c_{j:i}}{\sum_{x\in X} b_{\mathrm{RSU}:x} \cdot c_{x:i}}, \quad (7)$$

where $c_{j:i} = b_{j:i} + d_{j:i}$represents the degree of familiarity that$j$ is with$i$.

According to δ_x, subjective opinions of different recommenders are combined into a single opinion, which is called observation of interactions. First, calculate the interaction opinions of v_i:

$$\begin{cases} b_{x:i}^{\mathrm{int}} = \dfrac{1}{\sum_{x\in X}\delta_{x:i}} \sum_{x\in X} \delta_{x:i} b_{x:i}, \\ d_{x:i}^{\mathrm{int}} = \dfrac{1}{\sum_{x\in X}\delta_{x:i}} \sum_{x\in X} \delta_{x:i} d_{x:i}, \\ u_{x:i}^{\mathrm{int}} = \dfrac{1}{\sum_{x\in X}\delta_{x:i}} \sum_{x\in X} \delta_{x:i} u_{x:i}. \end{cases} \quad (8)$$

Then, calculate the final observation rating of v_i by

$$o_i = b_{x:i}^{\mathrm{int}} + \kappa u_{x:i}^{\mathrm{int}}, \quad (9)$$

where κ is the uncertainty influence level, set to 0.5.

The initial value of service rating is 0. After the vehicle completes a ride-sharing service, passengers will rate the service quality as sq_n. The rating is uploaded to RSU and broadcast to all nodes after verification. When generating a new block, calculate the average value of all ratings obtained so far as the service rating:

$$s_i = \frac{\sum_{n\in N} \mathrm{sq}_n}{N}. \quad (10)$$

Accumulated reputation of validate blocks of vehicle v_i is the reward for successfully generating blocks in each round of consensus, reflecting the positive contribution of v_i to the network. Denote m as the number of participants in v_i, and N_m is the total number of miners, and B_m is the total reward. The accumulated reputation p_i is the sum of the rewards obtained from all activities of the block generation that the node participated in the current round. The calculation is as follows:

$$p_i = \sum_{m\in M} \frac{\tau_i B_m}{N_m}, \quad (11)$$

where τ_i is the weight factor that is related to whether v_i is a miner or a leader, set to 0.8 or 0.2.

After completion of the selection of L_{sf}, RSU calculates the total reputation value R_{sum} according to the nodes in L_{sf} and selects $(1/2)R_{\mathrm{sum}}$ as the target reputation threshold. Order the reputation value from high to low, select the node until the total reputation value of the selected node exceeds the target reputation threshold, and the selected node is stored in L_{chosen}, which is the consensus group for the current block generation.

It can be formulated as a Bayesian game when the adversaries and normal vehicles run in the network as two groups, and each group has incomplete acquaintance about the resource other group uses for defense. Assume that the

Table 3: Parameters of simulation.

Parameter	Value
Number of vehicles N	[20, 700]
Time interval of vehicles accessing the network	[2 s, 60 s]
Proportion of miner adaptively selected α	[15%, 50%]
Travel distance d	[10 km, 50 km]
Maximum of location signatures obtained $n_{\max}$	10
Belief by normal/malicious vehicles $b_{j:i}$	[0.8, 0.9]/[0.4, 0.5]
Distrust by normal/malicious vehicles $b_{j:i}$	[0, 0.1]/[0.4, 0.6]
Uncertain by normal/malicious vehicles $b_{j:i}$	[0, 0.1]
Number of interacted vehicles	10
Number of ride-sharing services provided	[1, 5]
Weight factor for reputation factor $\gamma, \epsilon, \zeta, \eta$	0.25
Proportion of malicious nodes f	$1/3N$
Times before reputation reset	10

strategy space of the adversary group follows a uniform distribution Sadv ~ $U(0, N_a)$, N_a represents the maximum number of adversaries. Similarly, normal vehicles follow uniform distribution Snor ~ $U(0, N_n)$ and N_n denotes the maximum number of normal vehicles that respond to participate in the selection.

In our algorithm, assuming the worst case, the malicious node makes a part in L_{chosen} and then becomes the final consensus group member or even the leader node through reputation optimization. The malicious node must have at least a reputation value greater than the average reputation. Assuming that the acquisition and contribution of all four factors of reputation value are linear. The contribution value is denoted as θ for each factor to gain the same reputation value, and the cost of each θ is C_m. If the reputation threshold of L_{chosen} is R_t, the contribution of malicious vehicles in this round of consensus election is at least R_t/θ. As the first step of filtering is stochastic, α can be seen as the probability of being selected. The total cost of adversary group can be computed as follows:

$$C_{\text{adv}} = \frac{N_a \cdot C_m \cdot R_t}{\theta \cdot \alpha}. \tag{12}$$

According to the PBFT rules, if the adversary group colludes and destroys the system, N_a must be greater than 1/3 of the total number N in the network; that is, N_a must be greater than $(1/2)/N_b$. At this time, the total profit U_{adv} of the adversary group is

$$\begin{cases} U_{\text{adv}} = B_{\text{adv}} & S_n < 2S_a, \\ U_{\text{adv}} = -C_{\text{adv}} & \text{otherwise}. \end{cases} \tag{13}$$

B_{adv} denotes as the reward for successful attack. S_n and S_a represent the total resources N_n and N_a paid to win the game. If attack is given up, the return of the adversary group is 0. If participating in the consensus, the return depends on the difference between theB_{adv}andC_{adv}, that is,

$$E[U_{\text{adv}}] = \int_0^{2S_a} B_{\text{adv}} dS_n + \int_{2S_a}^{S_n} - C_{\text{adv}} dS_n. \tag{14}$$

To make $E[U_{\text{adv}}]$ greater than 0, we have the total resources paid by adversary group which is

$$S_{\text{adv}} > \frac{N_b}{2} - \frac{\alpha \cdot \theta \cdot B_{\text{adv}}}{R_t \cdot C_m}. \tag{15}$$

From Equation (15), we can see that when the number of nodes in the network is fixed, we can control the size of the consensus group α and the credit value of each contribution θ to control the resources that the adversary group need to pay, so as to defend the attack of rational adversaries.

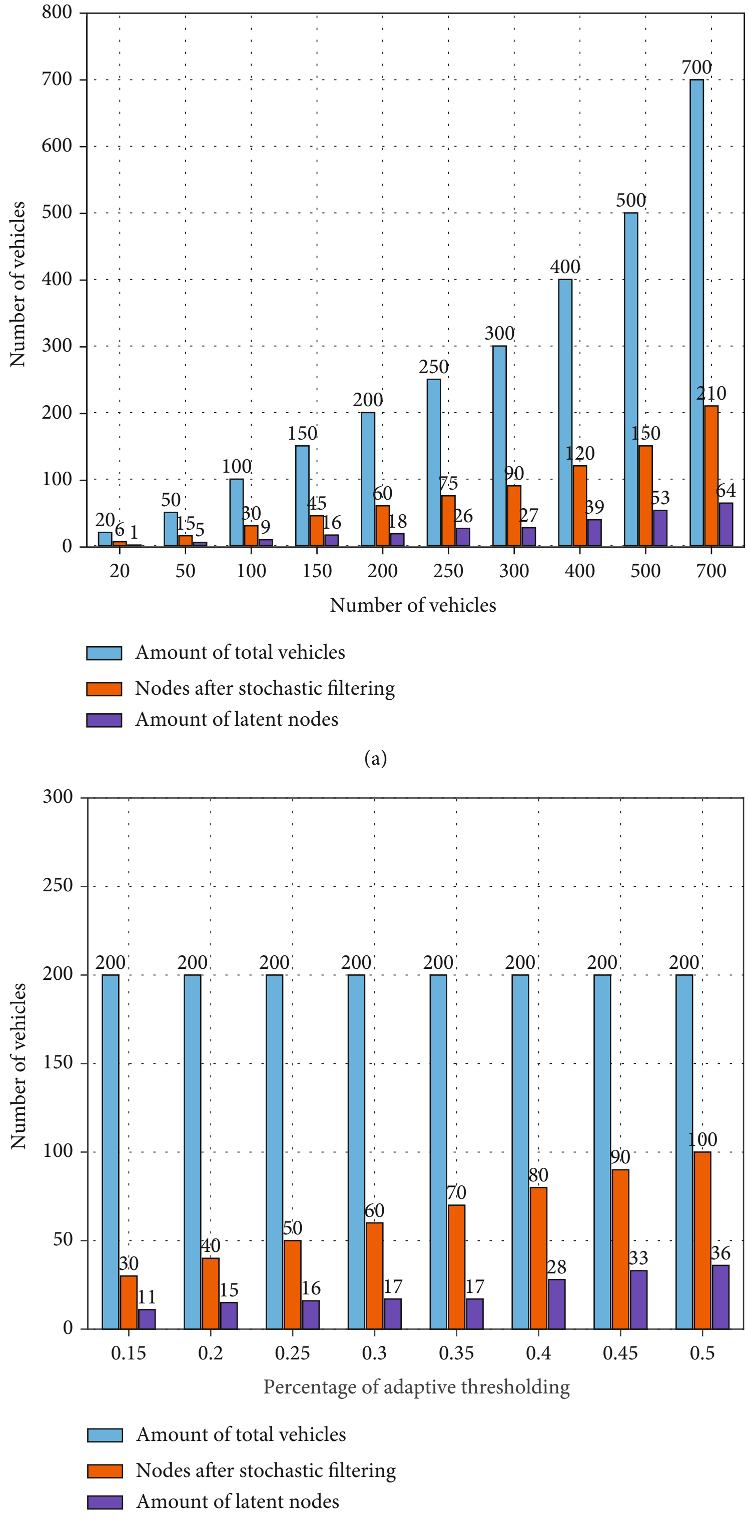
4.4. Basic Consensus Protocol: Practical Byzantine Fault Tolerance. Finally, the selected miner group L_{chosen} uses PBFT to determine what transactions and information are included in current block. All nodes in L_{chosen} become leader node in turn, and others become verification nodes. The complexity of PBFT is $O(N^2)$. As the number of nodes grows rapidly in the IoV, the complexity of the network increases exponentially. Therefore, it is necessary to use MPoR to adjust the size of the consensus group first.

4.5. Analysis of MPoR's Security. In this section, we provide theoretical proofs of capability against several kinds of attacks of MPoR.

(i) Internal attack: when majority of the internal nodes of the network are controlled by malicious adversaries, adversaries will temporarily own more than half of the computing power of the whole network, and the consensus algorithm of the blockchain system based on computing power (such as PoW) will be damaged. Since the initial filtering of the algorithm depends entirely on the specific timestamp and adaptive identification that are different every time, no one can predict how many active nodes are in the network. Therefore, even if adversary has more than half of the entire computing power of entire network, it is impossible to ensure that he will become a miner

(ii) External attack: when an adversary owns more than 50% of the total network stakes, the external adversary can control the miner selection in this way (such as PoS). Since the MPoR algorithm does not depend on the node with the highest stake in the network and the node with the highest stake cannot always pass stochastic filtering, so the effect of the external adversary by occupying majority stakes is futile

(iii) Collusive attack: in MPoR, whether through stochastic filtering or not depends on vehicle's individual ID, which is guaranteed unique, given by RSU. As a result, malicious nodes cannot collude to exchange information with each other to gain the



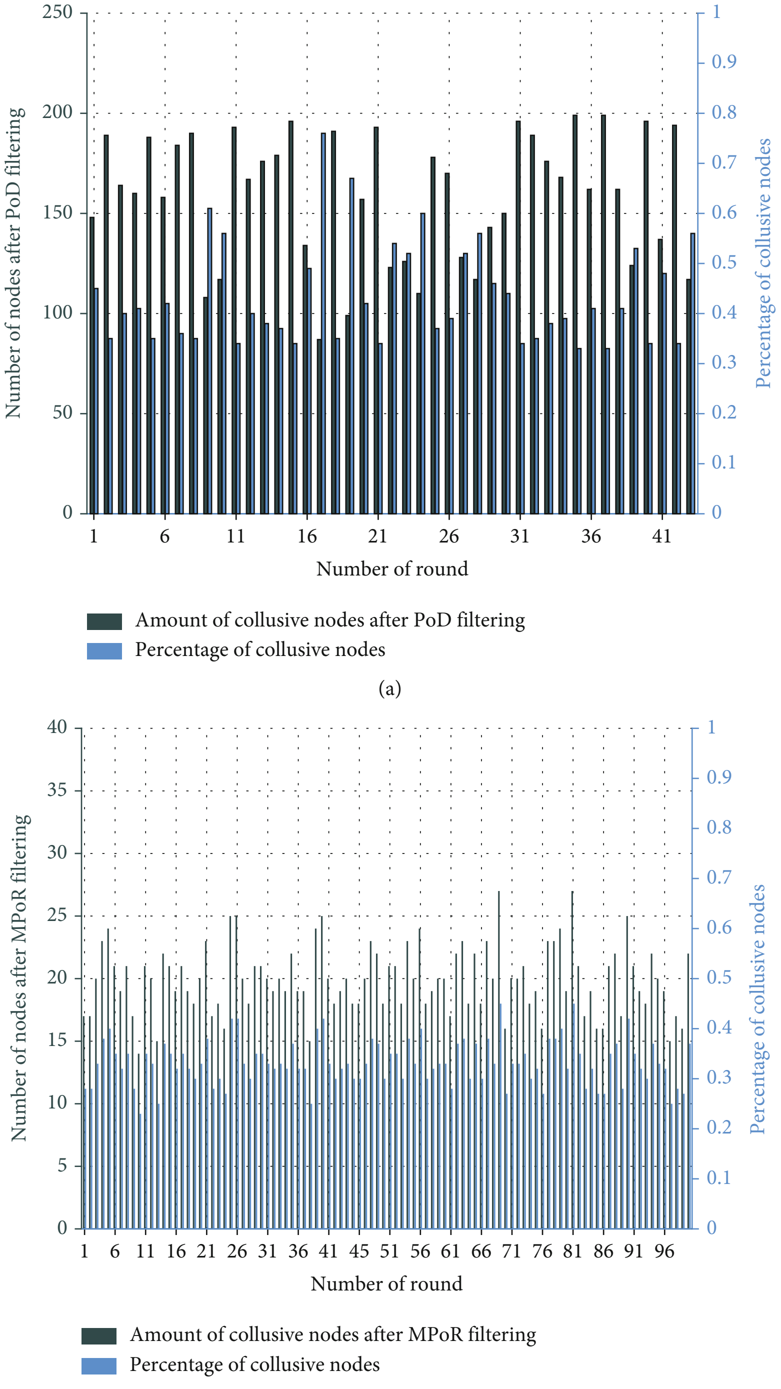
(a)

(b)

Figure 3: Stochastic filtering simulation under latent attack. Set latent nodes as 1/3 of total nodes. (a) Fix α as 30%; set N from 20 to 700. (b) Fix N as 200; set α from 15% to 50%.

same identification, and they cannot be selected or unselected at the same time. This completely avoids the possibility of collusive attack

(iv) Latent attack: we have discussed in Section 4.3 that the latent malicious node has no higher filtering probability than other nodes. Even through filtering,



(a)

(b)

Figure 4: Stochastic filtering simulation under collusive attack. Set the collusive nodes as 1/3 of total nodes. (a) Performance of PoD under collusive attack. (b) Performance of MPoR under collusive attack.

one needs a lot of positive contribution cost to enter the final consensus group. Therefore, our algorithm can alleviate the attack desire of the latent under reasonable assumptions

5. Experimental Analysis

5.1. Basic Settings of Simulation. The simulation content carries out reasonable parameter configuration according to the research objectives. The feasibility, scalability, and attack resistance of the algorithm are discussed and answered through the simulation data and detailed analysis of the simulation experiment. The experimental platform is GoLand2020 3.4. The main parameters used in the simulation are given in Table 3.

5.2. Security Analysis of Stochastic Filtering. First, we verify the security of the stochastic filtering step. We assume two

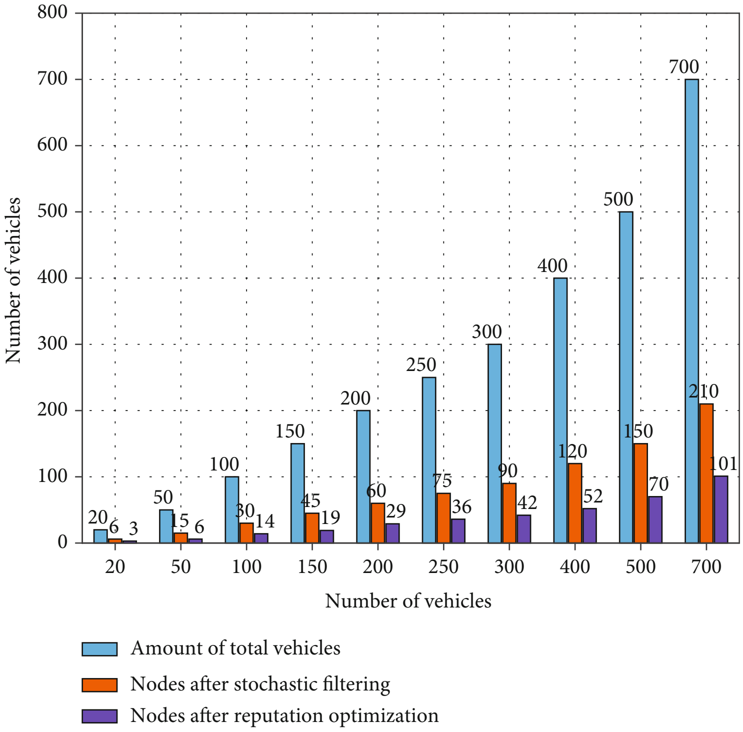


Figure 5: Performance of reputation optimization.

different types of adversaries: latent attacker and collusive attacker, simulate, respectively.

Latent attackers operate as legal vehicles, they complete ride-sharing service, communicate with RSU and other nodes normally, and accumulate reputation. Only after entering the final consensus group will it start to do evil. The simulation is carried out based on a different number of vehicle nodes N or different proportion of miner adaptively selected α, as shown in Figure 3.

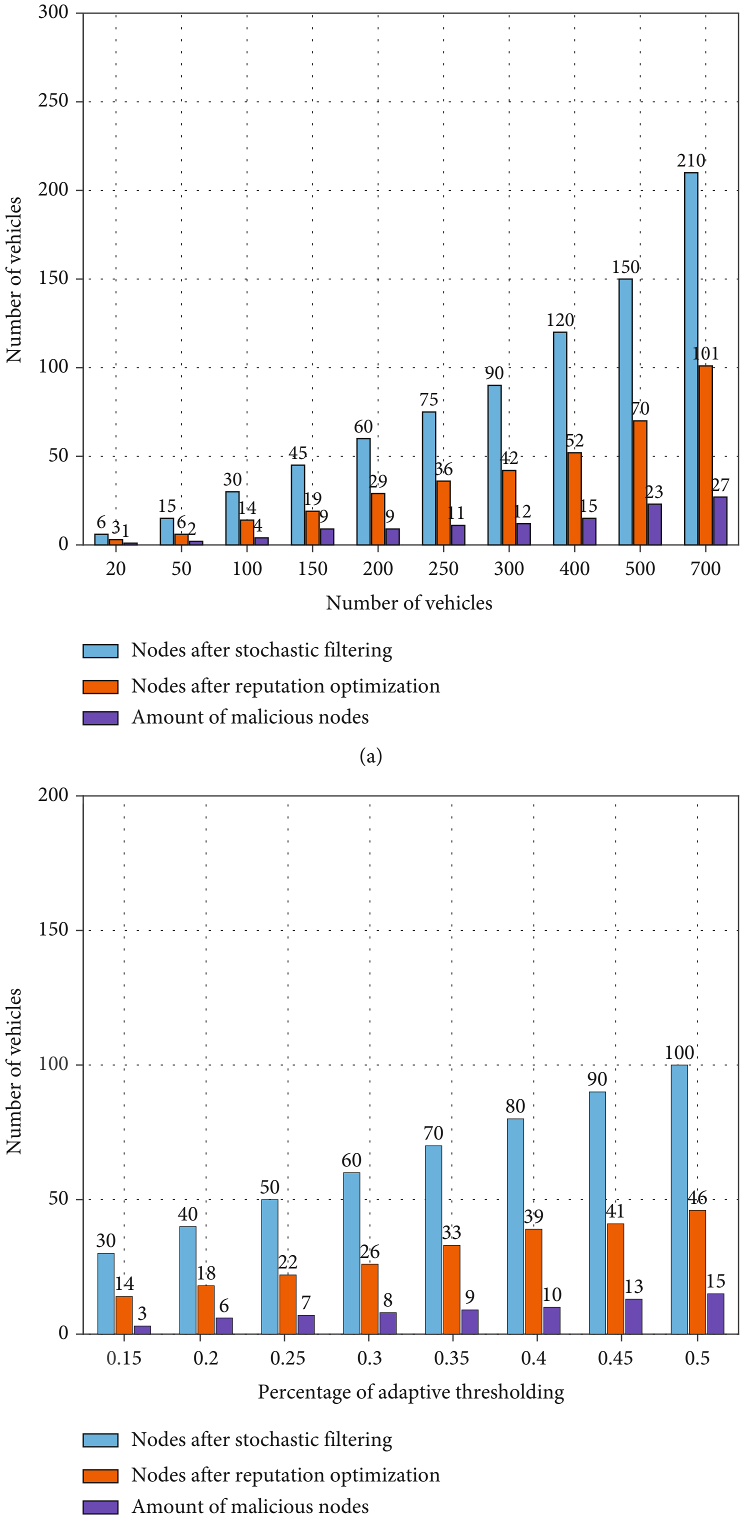
Figure 3(a) shows the stochastic filtering results of different N, and α is 30%; In Figure 3(b), we fix the vehicle node as 200 and then select different proportions of the number of miners, from 15% to 50%. As can be seen in the figure, under the two types of parameter configurations, the proportion of stochastically filtered malicious nodes in the total number of nodes after filtering is maintained at about 1/3, which is in line with our mathematical theoretical expectations discussed in Section 4.2.

Then, we assume that there are 1/3 collusive nodes in the network. Although they cannot predict the total number of nodes in the network and the average driving distance or access timestamp of the network, they can exchange information with each other and obtain the same hash value in the stochastic filtering step, so as to pass/not pass the stochastic filtering at the same time. Once they completely pass the stochastic filtering at a certain time, then there is a lot of room and possibility to do evil.

The previous typical algorithm PoD [12] based on stochastic filtering does not solve the problem of collusive attack. In PoD, the travel distance is used as the standard of the filter, and the network average travel distance is used as the target hash threshold. Set N to 200 and the collusive nodes to 66 (nearly 1/3). The nodes whose hash value of travel distance is less than the target threshold pass stochastic filtering. Collusive nodes share data with each other and travel the same distance. 100 simulations were carried out, of which 43 times collusive nodes were all selected after stochastic filtering. The percentage of collusive nodes after random filtering in all filtered nodes in the 43 simulations is given in Figure 4(a). Under PoD, the number of nodes after random filtering is completely related to the target hash threshold and unpredictable, so the percentage difference is large, but it can be seen that the proportion of collusive nodes in all rounds exceeds 1/3. It can be proved that PoD cannot resist collusive attack.

In MPoR, instead of using the information that can be predicted and exchanged by the vehicles themselves as the target hash threshold, we use the vehicle access timestamp and access sequence order to generate a unique identification, so as to prevent the possibility of collusive attack from the source. We simulate 100 times as well, set simulation parameters the same as PoD simulation, α set to 30% additionally. The results are shown in Figure 4(b). After MPoR stochastic filtering, the proportion of collusive nodes is maintained at about 33%, and the simulation is in line with our theoretical proof. Thus, when the amount of data is large enough, MPoR algorithm can resist collusive attack.

From the above simulation results for two types of adversaries, stochastic filtering can filter a large number of nodes on the premise of ensuring security. However, due to inevitable accidents in randomness, simple randomness is not enough to fully ensure the safe operation of the system. Therefore, we need to optimize the consensus group based on reputation.

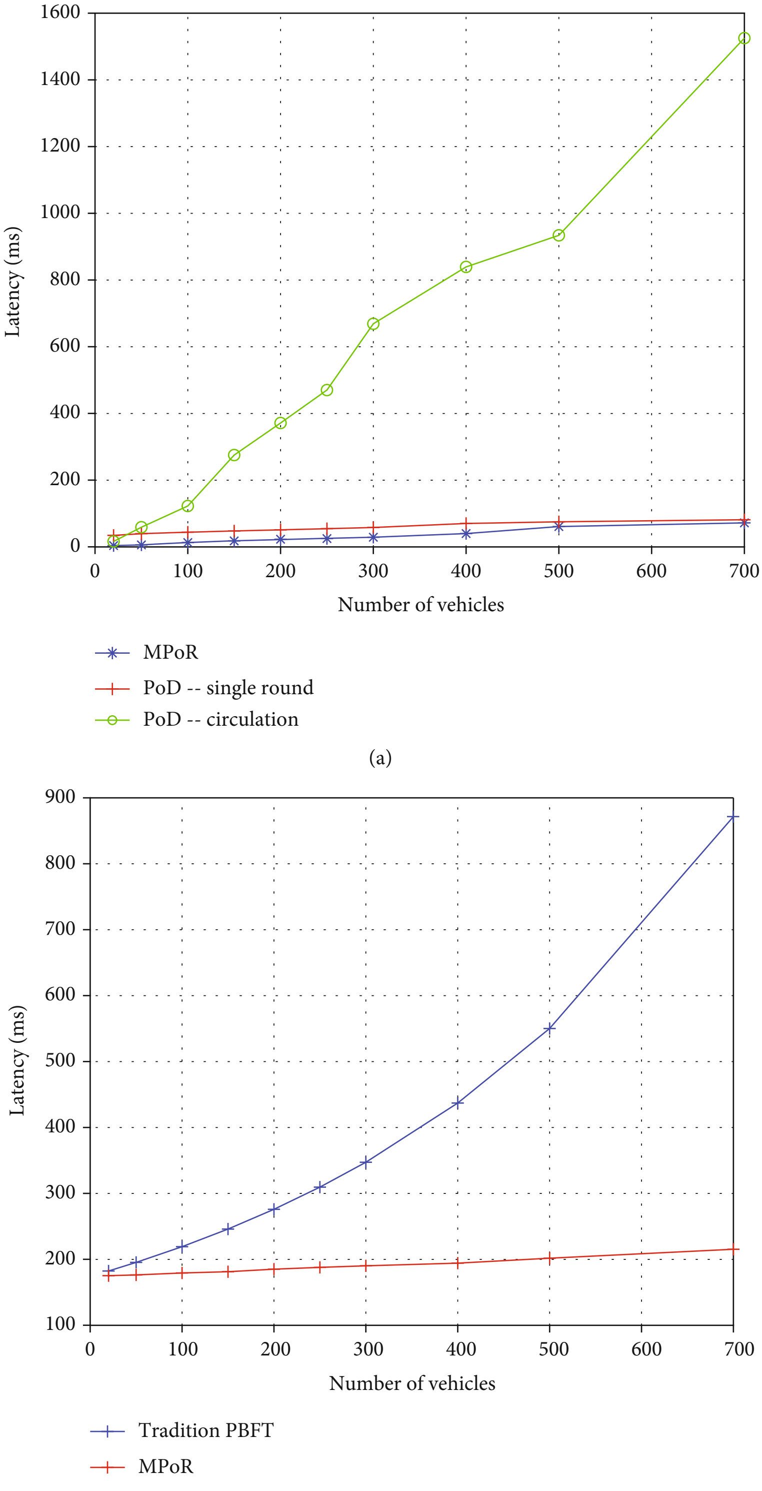


(a)

(b)

Figure 6: Reputation optimization simulation under 1/3 malicious nodes. (a) Fix α to 30%; set N from 20 to 700. (b) Fix N to 200; set α from 15% to 50%.

5.3. Security Analysis of Reputation Optimization. Here, we verify the performance the second step of MPoR, reputation optimization. Figure 5 shows the performance result under different numbers of vehicle nodes N. α is still 30%, and the reputation optimization selects half of the nodes with a greater reputation value. It can be seen from the figure that the size of the consensus group after reputation optimization basically accounts for less than 1/2 of the size of the



(a)

(b)

Figure 7: Latency performance simulation. (a)Time overhead of MPoR and PoD under stochastic filtering. (b)Time overhead of MPoR and PBFT to reach consensus.

consensus group after stochastic filtering, and there will be no significant difference. The reason is that, whether it is a normal node or a malicious node disguised as a normal node, the cumulative reputation value difference is not obvious under our algorithm. Therefore, in this step of filtering, it can be expected that the number of filtered nodes will basically remain below half. Meanwhile, through this reputation calculation method, RSU can quickly find nodes with abnormal reputation value (too high or too low) and pay extra attention to check whether they are malicious nodes.

Then, we assume that there are 1/3 malicious nodes in the network. The simulation is also carried out based on different vehicle nodes or different adaptive values, as shown in Figure 6.

Figure 6(a) shows the results of different vehicle nodes, and the proportion of miners is 30%; In Figure 6(b), we fix N to 200 and then vary α from 15% to 50%. In the simulation, the reputation value of malicious nodes is slightly lower than that of normal nodes. As in our scheme, the malicious node will be found at once when it does evil, so the cumulative reputation of the block is lower. As can be seen in the figure, under two types of parameter configuration, the proportions of malicious nodes after reputation optimization are always less than 1/3. According to the security proof in Section 4.3, the main function of reputation optimization is to increase the attack cost of malicious nodes so as to reduce their attack desire. Through simulation, we can see that our algorithm can filter a large number of consensus nodes under reasonable fault tolerance and improve the efficiency of consensus.

5.4. Latency Analysis. The time overhead of MPoR and PoD is compared in Figure 7(a). α is set to 30%. Two experiments of PoD are single round simulation (only complete one round of hash comparison, and the size of the filtered consensus pool cannot be controlled), and circular simulation (stipulate that only when the proportion of miners after filtering is 30%, it can be output). As can be seen from the figure, if we want to meet the adaptive selection of the number of miners, the MPoR algorithm can greatly reduce the latency. Meanwhile, in a single round of simulation, the latency overhead of the MPoR algorithm is also relatively better.

Then, we compare the time overhead between traditional PBFT and consensus after MPoR filtered in Figure 7(b). α is set to 30% as well; N ranges from 20 to 700. From the simulation results, it can be seen that after MPoR, the latency required for the network to reach a consensus is greatly reduced. This is because after filtering, the size of consensus group can be significantly reduced on the basis of ensuring fairness and security, so as to improve the efficiency of the whole network.

6. Conclusion

Aiming at the security challenges existing in ridesharing services, we propose a new consensus algorithm MPoR applied to the consortium blockchain-based Internet of Vehicles. MPoR takes the access time of network and order of vehicle nodes as the evaluation standard of stochastic filtering and sorts the nodes through the hash value, so as to realize the adaptive selection of the size of the consensus pool. In the filtering step, each node has the same selected probability, which is fairer and more scalable than the existing consensus algorithms. Meanwhile, in order to resist malicious attacks, a reputation rating algorithm based on multifactors is designed to eliminate the shortcomings of the existing single reputation evaluation criteria. It can eliminate low-reputation nodes, optimize the consensus node pool, and encourage nodes to participate in consensus. Finally, we conduct a large number of simulations and security analysis to prove the effectiveness, scalability, and ability to resist various attacks of the proposed algorithm.

Data Availability

The raw data supporting the conclusions of this article will be made available by the authors, without undue reservation.

Conflicts of Interest

The authors declared that there are no conflicts of interest in this study.

Acknowledgments

This research is funded by grants from the National Key R&D Program of China (No. 2020YFB1805403) and Foundation of Guizhou Provincial Key Laboratory of Public Big Data (No. 2017BDKFJJ015, No. 2018BDKFJJ008, No. 2018BDKFJJ020, and No. 2018BDKFJJ021).

References

[1] B. Caulfield, “Estimating the environmental benefits of ridesharing: a case study of Dublin,” *Transportation Research Part D: Transport and Environment*, vol. 14, no. 7, pp. 527–531, 2009.

[2] R. Calo and A. Rosenblat, “The taking economy: Uber, information, and power,” *Columbia Law Review*, vol. 117, p. 1623, 2017.

[3] S. Zou, J. Xi, S. Wang, Y. Lu, and G. Xu, “Reportcoin: a novel blockchain-based incentive anonymous reporting system,” *IEEE Access*, vol. 7, pp. 65544–65559, 2019.

[4] S. Nakamoto, “Bitcoin: a peer-to-peer electronic cash system,” *Decentralized Business Review*, 2008, https://bitcoin.org/bitcoin.pdf14.

[5] Y. Zhang and J. Wen, “The IoT electric business model: using blockchain technology for the internet of things,” *Peer-to-Peer Networking and Applications*, vol. 10, no. 4, pp. 983–994, 2017.

[6] S. Wang, J. Wang, X. Wang et al., “Blockchain-powered parallel healthcare systems based on the ACP approach,” *IEEE Transactions on Computational Social Systems*, vol. 5, no. 4, pp. 942–950, 2018.

[7] S. Saberi, M. Kouhizadeh, J. Sarkis, and L. Shen, “Blockchain technology and its relationships to sustainable supply chain management,” *International Journal of Production Research*, vol. 57, no. 7, pp. 2117–2135, 2019.

[8] K. Gai, Y. Wu, L. Zhu, L. Xu, and Y. Zhang, “Permissioned blockchain and edge computing empowered privacy-preserving smart grid networks,” *IEEE Internet of Things Journal*, vol. 6, no. 5, pp. 7992–8004, 2019.

[9] B. Jia, X. Zhang, J. Liu, Y. Zhang, K. Huang, and Y. Liang, *Blockchain-enabled federated learning data protection aggregation scheme with differential privacy and homomorphic encryption in IIoT*, IEEE Transactions on Industrial Informatics, 2022.

[10] C. Peng, C. Wu, L. Gao, J. Zhang, K.-L. Alvin Yau, and Y. Ji, “Blockchain for vehicular internet of things: recent advances and open issues,” *Sensors*, vol. 20, no. 18, p. 5079, 2020.

[11] M. Salehie and L. Tahvildari, “Self-adaptive software,” *ACM transactions on autonomous and adaptive systems (TAAS)*, vol. 4, no. 2, pp. 1–42, 2009.

[12] S. Kudva, S. Badsha, S. Sengupta, I. Khalil, and A. Zomaya, “Towards secure and practical consensus for blockchain based VANET,” *Information Sciences*, vol. 545, pp. 170–187, 2021.

[13] D. Suo, J. Zhao, and S. E. Sarma, “Proof of travel for trust-based data validation in V2I communication part i: methodology,” 2021, https://arxiv.org/abs/2104.05070.

[14] L. Sun and J. Wu, “A scalable and transferable federated learning system for classifying healthcare sensor data,” *IEEE Journal of Biomedical and Health Informatics*, vol. PP, pp. 1–1, 2022.

[15] Q. Yu and L. Sun, “LPClass: lightweight personalized sensor data classification in computational social systems,” *IEEE Transactions on Computational Social Systems*, pp. 1–11, 2022.

[16] Y. Wang, L. Sun, and S. Subramani, “CAB: classifying arrhythmias based on imbalanced sensor data,” *KSII Transactions on Internet and Information Systems (TIIS)*, vol. 15, no. 7, pp. 2304–2320, 2021.

[17] L. Sun, R. Zhou, D. Peng, A. Bouguettaya, and Y. Zhang, “Automatically building service-based systems with function relaxation,” *IEEE Transactions on Cybernetics*, pp. 1–14, 2022.

[18] L. Sun, Q. Yu, D. Peng, S. Subramani, and X. Wang, “Fogmed: a fog-based framework for disease prognosis based medical sensor data streams,” *CMC-COMPUTERS MATERIALS & CONTINUA*, vol. 66, no. 1, pp. 603–619, 2020.

[19] M. Li and L. Wang, “Privacy preservation of location information based on MinHash algorithm in online ride-hailing services,” in *2018 Sixth International Conference on Advanced Cloud and Big Data (CBD)*, pp. 257–262, Lanzhou, China, 2018.

[20] Y. Wang, Z. Su, K. Zhang, and A. Benslimane, “Challenges and solutions in autonomous driving: a blockchain approach,” *IEEE Network*, vol. 34, no. 4, pp. 218–226, 2020.

[21] Z. Yang, K. Zheng, K. Yang, and V. C. Leung, “A blockchain-based reputation system for data credibility assessment in vehicular networks,” in *2017 IEEE 28th annual international symposium on personal, indoor, and mobile radio communications (PIMRC)*, pp. 1–5, Montreal, QC, Canada, 2017.

[22] Y. Yuan and F.-Y. Wang, “Towards blockchain-based intelligent transportation systems,” in *2016 IEEE 19th international conference on intelligent transportation systems (ITSC)*, pp. 2663–2668, Rio de Janeiro, Brazil, 2016.

[23] L. A. Hîrţan, C. Dobre, and H. González-Vélez, “Blockchain-based reputation for intelligent transportation systems,” *Sensors*, vol. 20, no. 3, p. 791, 2020.

[24] D. Wang and X. Zhang, “Secure ride-sharing services based on a consortium blockchain,” *IEEE Internet of Things Journal*, vol. 8, no. 4, pp. 2976–2991, 2020.

[25] S. A. Renu and B. G. Banik, “Implementation of a secure ride-sharing DApp using smart contracts on Ethereum blockchain,” *International Journal of Safety and Security Engineering*, vol. 11, no. 2, pp. 167–173, 2021.

[26] M. Li, L. Zhu, and X. Lin, “Efficient and privacypreserving carpooling using blockchain-assisted vehicular fog computing,” *IEEE Internet of Things Journal*, vol. 6, no. 3, pp. 4573–4584, 2018.

[27] H. Zhang, E. Deng, H. Zhu, and Z. Cao, “Smart contract for secure billing in ride-hailing service via blockchain,” *Peer-to-Peer Networking and Applications*, vol. 12, no. 5, pp. 1346–1357, 2019.

[28] Y. Liu, K. Li, Y. Jin, Y. Zhang, and W. Qu, “A novel reputation computation model based on subjective logic for mobile ad hoc networks,” *Future Generation Computer Systems*, vol. 27, no. 5, pp. 547–554, 2011.