# Improved PBFT Consensus Based on Reputation System in Vehicle Network

No Author Given

No Institute Given

**Abstract.** Blockchain technology is a decentralized distributed database technology, which greatly improves the security and credibility of the data exchange process through decentralization. A great deal of research has already been conducted on combining blockchain and vehicular ad-hoc networks(VANETs) applications to solve the problems of opaque user transactions, data tampering, and insufficient motivation of participating parties. However, in the large-scale VANETs, the commonly used blockchain consensus mechanism PBFT suffers from poor scalability, high communication volume, and insufficient control of node behaviour. Therefore, in this paper, an improved consensus mechanism N-PBFT is designed for the field of VANETs based on the construction of vehicle trust management system. The improved consensus mechanism N-PBFT solves the problems of node identity peering, poor scalability, and high communication volume. After experimental testing, the proposed reputation system is able to effectively manage the information of the VANETs. And the improved PBFT consensus algorithm has a significant performance improvement over the traditional PBFT, SG-PBFT and RIPPB in terms of both throughput and latency.

**Keywords:** VANETs · Blockchain · Consensus · PBFT · Reputation System.

## 1 Introduction

In the connected vehicular ad-hoc networks (VANETs), traditional data security sharing faces the problem of centralisation [1]. A large number of vehicles establish contact through a central node such as a base station, which brings two problems. First is that the central node is easy to be attacked, resulting in single point of failure problem and affecting the communication of the network. Second is that there are many nodes in the VANETs, which brings huge communication overhead to the central node. Fortunately, blockchain technology has the characteristics of decentralization. Therefore, the emergence of blockchain technology [2] provides a new possibility to solve the above problems. Blockchain technology has the characteristics of decentralisation, non-tampering and traceability. Blockchain can be applied to scenarios in which multiple parties do not trust each other. The emergence of blockchain technology provides technical means to solve the trust and security problems in the process of data sharing between vehicles. Currently, blockchain-based VANETs data sharing applications have received extensive attention from academia and industry [3–5]. Although blockchain technology is a good solution to the problem of difficult decentralisation in centralised networks. However, at this stage, blockchain-based VANETs data sharing still faces serious security and efficiency constraints.

On the one hand, entities in the VANETs are highly connected by sending and receiving a large number of messages, which may be subject to a variety of attacks, such as impersonation, denial of service (DoS), eavesdropping, false message injection, pseudo-symbolism, and malware attacks. Malicious entities in the VANETs may spread false messages which may lead to traffic accidents or dissemination of misleading information [6, 7]. In order to address the situation where malicious nodes and malicious messages are difficult to identify in the VANETs, reputation system is introduced to help users make informed decisions. The reputation system allows users to find trustworthy entities and access credible messages, and selectively interact with entities with high reputation scores [8, 9].

On the other hand, the application efficiency of blockchain mainly depends on the consensus mechanism. If the performance of the consensus mechanism used in the blockchain system is low, then blockchain-based message sharing for the internet of vehicles will be directly affected. VANETs has the characteristics of fast topology changes, high latency requirements, and a large number of nodes. These lead to the difficulty of applying traditional consensus mechanisms to blockchain-based VANETs applications. The arithmetic power of the VANETs is limited, and the traditional Proof-of-Work (PoW) [10] focuses more on high computational power, which will cause a large waste of arithmetic power. Proof-of-Stake (PoS) [11] bases the selection of miners on the equity of nodes and is suitable for stable networks. However, the rapid changes in vehicular networks may lead to the concentration of equity to a small number of nodes. Traditional Byzantine fault tolerance (PBFT) [12] works well in small scale networks. But the large number of nodes in a vehicular network can lead to large communication overheads. Therefore, there is an urgent need to combine the characteristics of the VANETs with traditional consensus methods to propose a consensus mechanism that is more applicable to VANETs.

In summary, this paper combines the specific characteristics of VANETs to design a reputation system for VANETs, which improves the vehicle's ability to discern the authenticity of messages in the network and excludes malicious nodes and malicious messages in time. Meanwhile, based on the reputation system, an improved PBFT consensus algorithm N-PBFT is proposed. The N-PBFT improves the efficiency of the blockchain system, so that the vehicle status and network messages can be updated in time.

Related works are discussed in Section 3. Then the system model is proposed in Section 4. In section 5 the proposed scheme is presented. In section 6 the experimental analysis is evaluated. The paper is concluded in Section 7.

## 2 Preliminaries

In this section, we will give a brief introduction to three traditional blockchain consensus algorithms. We will explain their rationale, advantages and disadvantages.

The consensus algorithm is the core of the blockchain, and the blockchain ensures the consistency of data through the consensus algorithm. Consensus mechanism is a mechanism for nodes to reach consensus on transactions recorded in the blockchain ledger. Consensus mechanism is a key component to ensure the security and effectiveness of the blockchain network. The following are several common blockchain consensus mechanisms:

- ***Proof of Work(POW):*** PoW is suitable for application in the blockchain public chain system. The proof of work process can be divided into two stages: solution and verification. The solution stage is to carry out a large number of complex mathematical calculations and find a mathematical solution. The verification stage is to perform a simple mathematical calculation to verify whether the obtained solution is correct, which can be completed in a very short time. PoW consensus algorithm needs to ensure the consistency and correctness of blockchain data through a large amount of computation. As a result, the algorithm needs to consume a lot of computing resources and energy. In order to ensure data security, PoW takes a long time to reach consensus, resulting in low efficiency of PoW consensus.
- ***Proof of Stake(POS):*** In POS algorithms, users participate in the consensus process by holding a specific cryptocurrency, such as Ethereum's ETH. The higher the number of tokens a user holds, the higher the probability of obtaining the accounting right. The system randomly selects a number of holders to become Validators, and these Validators are responsible for verifying transactions and generating new blocks. Validators who successfully generate blocks receive a reward, usually from transaction fees and newly issued tokens. PoS does not require a large amount of computing power consumption, so it has lower energy consumption and is more environmentally friendly. But the interest in the network may be concentrated to a small number of users. As a result, users holding a large number of tokens may control the network, creating centralization risk.
- ***Practical Byzantine Fault Tolerance (PBFT):*** The PBFT consensus algorithm is applicable in the alliance chain and is designed to solve the Byzantine general problem. The Byzantine General problem proves that when the number of Byzantine nodes is $f$ and the total number of nodes $N$ in the network is greater than $3f$, the distributed system can reach consensus, and the time complexity of its algorithm is $O(N^{f+1})$. The PBFT consensus algorithm network allows $f$ Byzantine nodes, bur requiring $f < (N-1)/3$. The time complexity of PBFT algorithm is $O(N^2)$, it can be applied to practical distributed systems.

  Consistency protocol is a core protocol that the PBFT algorithm can complete consensus. It is mainly divided into three stages: pre-prepare, prepare, and commit. The execution process is shown in Fig. 1.

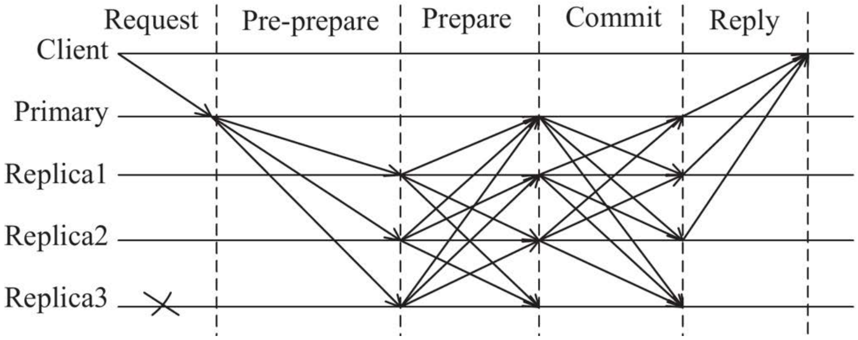

**Fig. 1.** The process of PBFT.

Pre-prepare: The primary node will generate the pre-prepare message from the received request sent by the client and broadcast the pre-prepare message to other nodes. The message format is $\{PRE - PREPARE, v, n, d, m\}$, where $v$ is the view number, $m$ is the message sent by the client, $d$ is the result of $m$ after hash operation, and $n$ is the message number.

Prepare: After receiving the pre-prepare message sent by the primary node, the consensus node generates the prepare message and broadcasts the prepare message to other nodes. The message format is $\{PREPARE, v, n, d, i\}$, where $i$ is the number of node. After each node receives the preparation message broadcast by other nodes, the node verifies the authenticity of the message, mainly comparing the fields $n$, $v$, and $m$. If more than $f+1$ prepare messages are correct, the node will enter the commit phase.

Commit: All nodes generate commit messages and broadcast them to other nodes. The message format is $\{Commit, v, n, d, i\}$. In this phase, the same validation work is done as in the prepare phase. When the verification passes, the confirmation result is sent to the client, and the consensus process of the request is completed.

In Table 1, we summarize the application fields, advantages and disadvantages of several commonly used blockchain consensus algorithms

**Table 1.** Comparison of consensus algorithms

| Consensus Name | Type of Blockchain | Advantages | Disadvantages |
|---|---|---|---|
| PoW | Public Blockchains | Totally decentralized | High energy consumption |
| PoS | Public Blockchains | Consumes less energy | May become centralized |
| DPoS | Public Blockchains | Short delays - Scalable | May become centralized |
| Raft | Private Blockchains | High resilience to attacks | Specific to some situations and cannot be generalized |
| PBFT | Private Blockchains | Byzantine Fault Tolerant | Not scalable - bandwidth consuming |

## 3 Related Work

In this section, we present some work that attempts to improve the consensus mechanism in VANETs.

We present some of the work on improving the PBFT protocol in terms of security, latency and communication traffic and its application in VANETs.

Li et al. [13] proposed a scalable PBFT-based consensus protocol. It reduces the time complexity of consensus and improves the scalability of consensus through a two-layer model that groups nodes hierarchically. However, the process of the method is too cumbersome and not applicable to the rapidly changing environment of VANETs.

Based on DPoS, Kang et al. [14] proposed an enhanced DPoS consensus scheme with a two-stage soft security solution for secure sharing of vehicle data. In the miner selection phase, a secure and efficient reputation management scheme is introduced using a multi-weighted subjective logic model. Miners are selected through reputation-based voting to reduce collusion between stakeholders with large shares and miner candidates. In the block validation phase, contract theory is used to incentivise highly reputable standby miners to participate in block validation to prevent internal collusion among active miners.

Zhu et al. [15] proposed a hierarchical trusted certificate service chain HCSC based on reputation value. A new hierarchical reputation consensus based on Delegated Proof of Stake (DPoS) and Proof of Workload (PoW) is given in the scheme. The reputation calculation method based on logistic regression model constructs a reputation value calculation model. Then it generates a consensus group based on the reputation value, and performs PoW within the group to reach a consensus result.

Zhang et al. [16] proposed a lightweight blockchain architecture based on Directed Acyclic Graph Lattice (DAG-lattice) structure in VANETs. The scheme adopts PBFT as the consensus algorithm. And the scheme proposed a parallel consensus mechanism based on the DAG structure in VANETs without the participation of RSU. It selects the nodes that are close to each other and relatively stable in moving to participate in the consensus. Then it solves the instability problem caused by the dynamic movement of vehicles.

Chen et al. [17] proposed a decentralised trust management system based on blockchain technology. It employs a decentralised consensus-based trust assessment model and uses a hierarchical structure to improve the efficiency of consensus and safeguard the dissemination of information.

Jiang et al. [18] proposed a blockchain consensus protocol for IoT multi-user collaboration scenarios. The Scheme combines Practical Byzantine Fault Tolerance (PBFT) and a game-based node selection mechanism. Then it divides the participating nodes into multiple collaborative groups and elects the primary node in a fair game within the group.

Ding et al. [19] proposed a robust and improved PBFT protocol RIPPB for blockchain by introducing reputation for nodes. The Scheme designs a new and improved logistic regression model that assigns reputation to nodes based on their behaviour in block generation. And it also add a pre-submission phase in PBFT, which reduces the number of inter-nodal communications and thus improves the performance of the protocol.

Xu et al. [20] proposed a secure and efficient distributed consensus algorithm SG-PBFT. This method adopts a score grouping mechanism to improve efficiency and optimize the consensus process. SG-PBFT effectively reduces the pressure on the central server and reduces the risk of single node attack, which is suitable for the VANETs.

In summary, past work has tended to focus on improving the efficiency of the blockchain system's consensus mechanism. However, in the environment of VANETs information sharing, there are some malicious nodes and false news. At the same time, we also need to think more about the behavior of nodes in the blockchain system and the reliability of messages. Therefore, our scheme evaluates the reliability of messages transmitted in the VANETs from the vehicle reputation system. Meanwhile, on this basis, the scheme will filter nodes to participate in consensus, and eventually reflect the behavior of each node to its reputation. Finally, we will make improvement process of consensus and improve the efficiency of block chain system.

## 4 System Model

In this section, we explain our proposed model for vehicle reputation management system. The annotation table is shown in Table 2.

**Table 2.** Notations and Definitions Used

| Notations | Definition |
|---|---|
| $Vehicle_i$ | The $i$th vehicle |
| $ID_i$ | The identifier of $vehicle_i$ |
| $CA$ | The certificate authority |
| $RSU_i$ | The $i$th RSU |
| $PK_i$ | The public key of $vehicle_i$ |
| $SK_i$ | The private key of the $vehicle_i$ |
| $M_k$ | The $k$th Message |
| $ticket_i$ | The ticket of $vehicle_i$ |
| $ReportingM$ | The reporting message |
| $VotingM$ | The voting message |
| $F_i$ | The interest of a $RSU_i$ |
| $tx_k$ | The message transaction of $M_k$ |
| $Location$ | Vehicle location information at message creation |
| $pos/nsg$ | The counts of positive/negative behaviors |
| $T_i$ | The current timestamp |

We present the participants in the vehicular reputation management system model shown in Fig. 2 and explain the roles of each part. The structure of the blockchain in the system is also explained.

1. **Participants:** the main participants of the proposed model are vehicles, roadside units (RSUs) and certificate authority(CA).

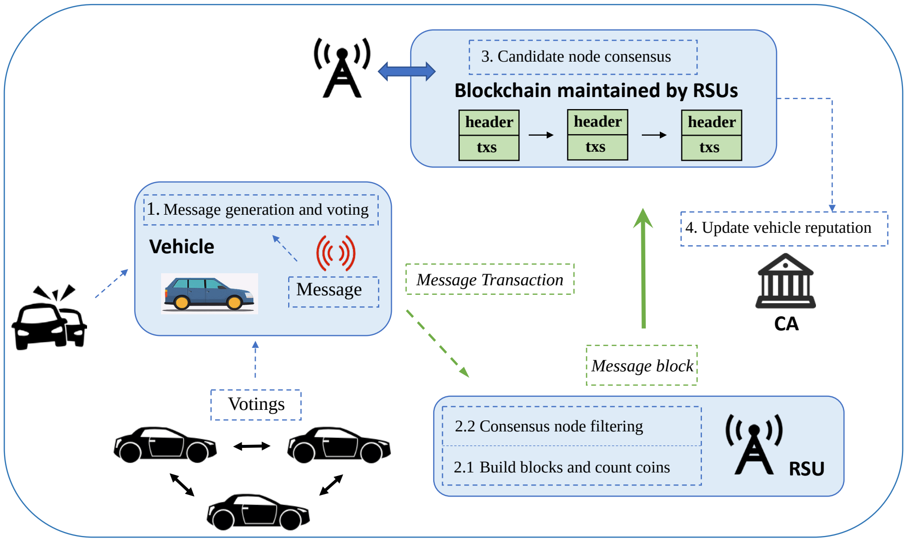
3. Candidate node consensus
Blockchain maintained by RSUs
header
txs
header
txs
header
txs
4. Update vehicle reputation
CA
1. Message generation and voting
Vehicle
Message
Message Transaction
Message block
Votings
2.2 Consensus node filtering
2.1 Build blocks and count coins
RSU

**Fig. 2.** System model.

- Vehicles: In our system, the role of vehicles can be categorised into message senders and message receivers. When there is some traffic situation around the vehicle, it can create a message and distributes it to nearby vehicles and RSU to share the information. The receiving vehicles of the message can vote on whether they agree that the message is credible or not. Each vehicle has a reputation score that reflects the trustworthiness of the vehicle's behaviour in the network. And vehicles can increase their reputation score by posting messages and participating in message voting.

- Roadside units (RSUs): RSUs are distributed on both sides of the road to collect messages and voting information sent by vehicles Blockchain is deployed on RSUs because of their fixed location and higher computational power with respect to vehicles. Therefore, the proposed scheme deploys the blockchain of the VANETs reputation system on the RSU. We assume that the RSU is a trusted party in our proposed system. Because the RSUs are regularly maintained by the operators, even if a failure occurs, it will be detected quickly and will not cause much impact on the overall operation of the system. RSUs collects and packages the messages between vehicles, participates in consensus and writes the messages to the blockchain after selecting by nodes. In addition, when a vehicle joins the network for the first time, it needs to be validated by the RSU first.The RSU will check the basic information of the vehicle and allow the vehicle to join after confirming that it is correct.

  - Certificate authority(CA): Drivers apply to CA when they register their vehicles for the first time, and CA will issue certificates to the vehicles after verifying the information of the vehicles. At the same time, CA updates the vehicle's reputation according to the vehicle's behaviour in the blockchain, and takes punitive measures, such as revocation, against vehicles with substandard reputation values.

2. **Blockchain:** The basic structure of the blockchain in the vehicle reputation system is shown in Fig. 3. Each block can be divided into two parts, block header and block body. The block header contains some basic information of the block, such as the block identifier, the denotation of the RSU that creates the block, and the block creation time. The block body contains two parts, the message transaction and the vehicle status. The message transaction consists of a broadcast message and a voting message, and the vehicle status message reflects the reputation status of the vehicle. As shown in Fig. 3, the basic structure of the Block is depicted. Each block header contains the block identifier $Block\,ID$, the RSU identifier $RSU\,ID$ and the creation time $T$ of the block. To link to the previous block, the block stores the hash value of the previous block($Prev\,Hash$) in the block header. Merkel Tree Root is a summary of the block data and stored in the block header to ensure data integrity. The nonce value and the target value $D_i$ are the values related to the consensus candidate node filtering calculation. The block body contains the transactions collected by the RSU, as well as a list of the vehicle's reputation.

## 5 Proposed Scheme

In this section, we will introduce the proposed scheme.

### 5.1 Vehicle Registration

Firstly, the $vehicle_i$ submits an application for registration to a trusted third party (CA). Then, after verifying the information of the vehicle, CA issues a certificate $\{ID_i, PK_i, SK_i\}$ to the vehicle. $ID_i$ is the identifier of the $vehicle_i$, $PK_i$ is the public key of the $vehicle_i$, and $SK_i$ is the private key of the $vehicle_i$.

When the vehicle $vehicle_i$ joins the vehicular network for the first time on the road, it sends a request for joining the network to the nearest RSU. The RSU checks the vehicle's information and grants it permission to join the network after it passes the validation. Then the RSU sends the blockchain ledger as well as a list of vehicles in the current regional network to the vehicle and assigns it the initially set reputation value (0.5) as well as the initial number of tickets.

### 5.2 Vehicle Broadcast Message

After a vehicle observes a traffic accident or a specific event, it creates a message $ReportingM$. Then, it broadcasts the message to vehicles within range of its

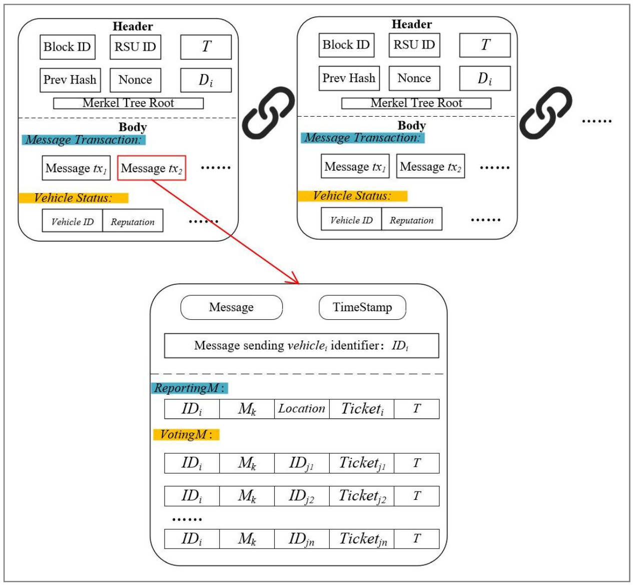

**Fig. 3.** Blockchain.

communication. The creation of the message consumes one ticket, and each ticket has an identifier in order to distinguish between different users. The message package $ReportingM$ is defined as follows:

$$ReportingM : \{ID_i, M_k, Location, ticket_i, T_1\} \tag{1}$$

where $M_k$ denotes the message, $Location$ is the position of the vehicle at which the message was sent, $ticket_i$ is the ticket used by the vehicle $vehicle_i$ to create the message, and $T_1$ denotes the time at which the message was created.

Once the message package is created, it is signed using the vehicle's private key and sent to other vehicles within the communication range.

### 5.3 Vehicle Voting

A vehicle that receives a message may decide whether to vote in favour of the message. It can decide base on an observation or by querying the reputation value of the message broadcaster, and voting on the message consumes tickets.

If $vehicle_j$ receives a certain message created by $vehicle_i$ and votes in favour of the message content, $vehicle_j$ should first create a voting message $VotingM$, which has the format:

$$VotingM : \{ID_i, ID_j, M_k, ticket_j, T\} \tag{2}$$

After creating the voting message, the $vehicle_j$ signs it and replies to the message sending vehicle $vehicle_i$. After receiving replies from more than half of the vehicles within its communication range, the $vehicle_i$ constructs the vote replies and the original message into a transaction $tx_k$. Then , the $vehicle_i$ sends $tx_k$ to the nearest RSU.

The specifics of the transaction $tx_k$ are shown in Fig. 3 and are constructed as follows:

$$tx_k : \{ReportingM, VotingM_1, ..., VotingM_n\} \tag{3}$$

### 5.4 Counting the total number of votes

The RSU collects the transactions generated in the vehicle network for a cycle, packages them, and waits for a consensus to write them to the blockchain.

Each RSU calculates the number of tickets it collects in a consensus cycle based on the specific messages and votes it collects in the transaction. If an RSU has more honest vehicles near it, the vehicles near it will create honest messages and receive more votes. So the total number of votes collected by an RSU in a cycle reflects the level of vehicle activity in the area near that RSU. Traffic messages in more active areas are more likely to significantly influence the overall vehicular network and have more information worth recording. Therefore, the number of tickets collected by an RSU in a cycle is taken as the interest of the RSU. And the consensus node filtering is based on RSU's interest. The number of tickets collected by $RSU_i$ in a cycle is denoted as $sum_i$. At the same time, if a node has too much interest, it will affect the randomness and fairness of the vehicular consensus. In order to ensure fairness, it is necessary to prevent a node from having too much interest. So the value of node's interest is defined as follows:

$$F_i = \min\{sum_i, F_{max}\} \tag{4}$$

$F_{max}$ is the upper limit of the node interest to prevent excessive node interest.

### 5.5 Consensus Node Filtering

According to the accumulated interest of the RSU in a cycle, the nodes in the network are filtered, and finally a set of nodes participating in the consensus is formed. Each RSU node performs a hash operation, and RSUs meets the calculation conditions become candidate nodes to participate in the consensus. The RSUs that collect a higher number of tickets in a cycle have higher interest. And RSUs with higher interset are more likely to satisfy the calculation conditions and become consensus nodes.

The following formula is calculated for each RSU:

$$Hash(ID_i, T_2, PreHash, Nonce) \leq D_i \tag{5}$$

$T_2$ is the time to perform the hash operation. Each RSU performs the above equation once according to its own situation. And the RSUs that satisfy the operation conditions become consensus candidate nodes. $D_i$ denotes the target value of the filtering calculation, which is related to the interest $F_i$ of different RSU. $D_i$ is calculated as follows:

$$D_i = 2^{(N_m - N_l)} - 1 \tag{6}$$

$$N_l = int(e^{-(\eta F_i + \mu)}) \tag{7}$$

Among them, $\eta$ and $\mu$ are system-specific parameters. If the node $RSU_i$ collects more votes in a cycle, the higher its $F_i$ value.The higher the $F_i$ value, the lower the $N_l$ value. And $N_m$ is a deterministic value in the case of deterministic hash function used by the vehicular network. Consequently, the target value of $D_i$ is higher. Then, $RSU_i$ has a higher probability to be a consensus candidate node in the filtering.

The node filtering process is shown in Fig. 4.

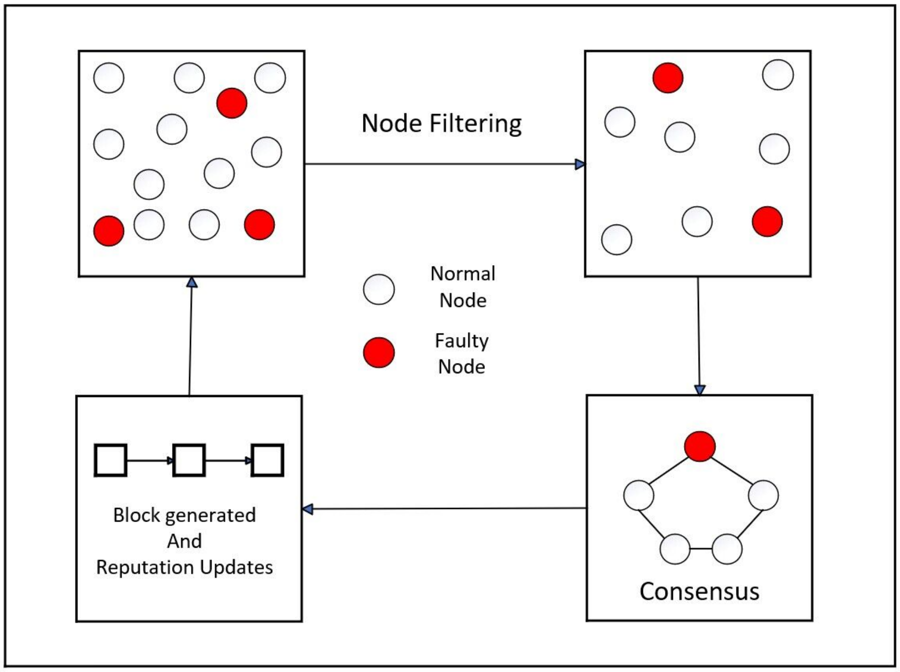

**Fig. 4.** Node filtering process.

After filtering, the nodes execute consensus to select a certain candidate block to be written into the blockchain ledger. Each RSU node needs only one hash calculation to complete the filtering, avoiding the waste of arithmetic power caused by large-scale mining. Meanwhile, the filtering reduces the number of candidate nodes participating in consensus, which makes the small-scale nodes show better communication efficiency and achieves a balance between arithmetic power and efficiency.

### 5.6 Reputation update

After filtering, consensus nodes are generated, and then consensus is executed in these nodes. The message transactions of the primary node are verified by the consensus of other nodes. And after verification, they will be written to the blockchain.

If the message or vote generated by the vehicle is successfully written to the block, it is rewarded with tickets. After a period of time, a portion of vehicles with higher reputation values are rewarded with tickets to encourage honest behaviour of vehicles.

Based on a vehicle's activity history in the blockchain ledger, the CA updates the vehicle's reputation using the *beta* reputation function and records the updated score in the blockchain. Each vehicle has a different probability $H$ of honestly using the ticket, using the $\Gamma$ function for $beta(H|pos, neg)$ defined as follows:

$$beta(H|pos, neg) = \frac{\Gamma(pos + neg)}{\Gamma(pos)\Gamma(neg)} \tag{8}$$

where $0 \leq H \leq 1$, $pos, neg > 0$, $pos$ and $neg$ are the counts of positive and negative behaviors respectively. The beta distribution function has a probability expected value of $pos/(pos + neg)$, which is used as a reputation score in the range of [0,1].

CA calculates the number of votes $st$ used by the vehicle and stored in the blockchain ledger, as well as the number of votes $st*$ used but not stored in the blockchain ledger. Then, RSU calculates the vehicle's reputation:

$$Reputation = \frac{pos}{(pos + neg)} \tag{9}$$

where the number of $pos$ is equal to $st + 1$ and the number of $neg$ is equal to $st^* + 1$.

After each consensus is completed, the reputation update is written to the blockchain simultaneously. Based on the latest vehicle activity record in the blockchain ledger, CA calculates the update of the vehicle's reputation value for inspection.

### 5.7 Consensus improvement

The process of the PBFT algorithm is mainly divided into the client sending the request, the three-stage consensus and replying to the client. In the reputation system of this paper, the message originator is changed from the client to the primary node in the network, and the process sent by the client can be merged. Therefore, the client part is deleted, and its functions are assumed by the primary node. This can reduce the propagation of information in the whole network and improve efficiency. At the same time, this paper simplifies the process of consensus by merging the pre-preparation and preparation phases in the traditional Byzantine Fault Tolerance (PBFT) on the basis of the node filtering in the previous section. Therefore, the process of the improved consensus protocol (N-PBFT) is a three-stage process.

As shown in the fig. 5, the improved PBFT consensus process consists of a request phase, a fast confirm phase and a commit phase. The specific processes are as follows.

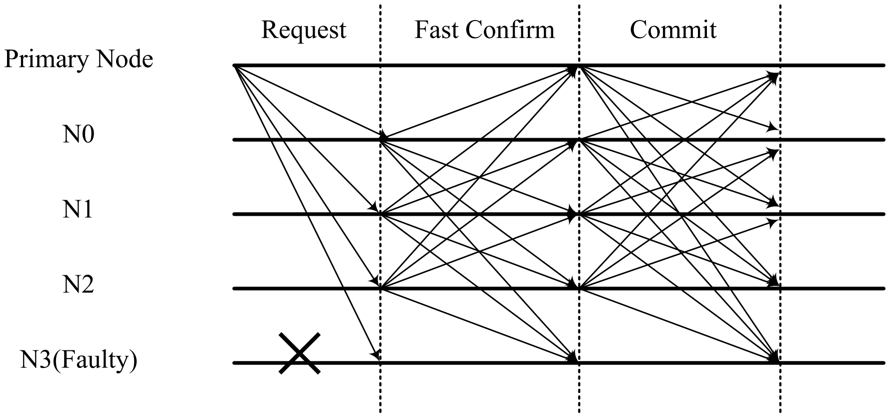

**Fig. 5.** N-PBFT Flowchart.

1. **Primary node generation:** after the end of a round of consensus node filtering, the filtered nodes' interest values are sorted. The nodes with the top 10% of the interest values are divided into a group, from which a node is randomly selected to become the primary node. The message transactions packaged by primary node in the previous cycle are validated first. Meanwhile, in order to prevent the same node from being elected consecutively, it is stipulated that the same node is elected as the primary node for at most three consecutive times. Otherwise, the node with the next highest interest is substituted.
2. **Request phase:** After the primary node is generated, the CA carries out the reputation value update calculation based on the data packed by the primary node in the previous cycle. After the primary node receives the vehicle reputation status update data sent by the CA, it packages it with the message transaction, signs the block data using the private key. Then the primary node sends it to other consensus nodes to verify it. The block contains the following information: $\{Message\,tx1, Message\,tx2, ...,$ $Vehicle\,Status\}$. The $Message\,tx$ represents the message transaction packed by the node and $Vehicle\,Status$ represents the vehicle reputation update.
3. **Fast confirmation Phase:** The other nodes participating in the consensus first authenticate the block sent by the master node after receiving it.Then, they should check the view number to ensure that they are under the same

view as the primary node. After passing the verification of the data, an answer is made via broadcast.

4. **Commit phase:** The master and normal nodes collect enough answer messages and verify them before timeout. When $2f + 1$ answer messages are received from different nodes, the validation passes and the response is confirmed by broadcasting, where $f$ denotes the number of Byzantine nodes. After the response phase, when the primary node receives $2f+1$ confirmation messages indicating that the current consensus is complete, it broadcasts the block and writes the block data to the blockchain.

## 6 Experimental Analysis

In this section, we first introduce the configuration to implement the experiment, and then we evaluate the performances of our scheme.

In this section, we perform performance tests on the proposed protocols and methods. Firstly, we test the filtering effect of the reputation management system on the nodes participating in the consensus, and test the effect under the condition of different number of nodes and different proportion of malicious nodes. Then the performance of the proposed N-PBFT[1] consensus mechanism is tested by simulation. We test the performance of N-PBFT consensus in terms of latency and throughput (TPS), and compare the performance with that of PBFT , RIPPB [19] and SG-PBFT [20]. The test program was written in GoLang, and the experimental configuration of the test was shown in Table 3.

**Table 3.** System Configuration.

| Environment | Configuration |
|---|---|
| CPU | Intel(R) Core(TM) i7-12700 CPU@ 2.3GHz |
| RAM | 32GB |
| OS | Ubuntu18.10 |
| Language | Go 1.21 |

Firstly we test for the filtering of nodes participating in consensus by the reputation management system in this paper. The experiment is set up with the number of nodes as 50 and 100, and the proportion of malicious nodes as 10%, 20% and 30% respectively. In order to ensure the generality of the data, the average of 10 tests is taken as the experimental result

As can be seen from Fig. 6 and Fig. 7. After the filtering of the reputation system, the size of nodes participating in the consensus has decreased to a certain extent, and smaller consensus participating nodes can improve the efficiency of PBFT consensus. At the same time, after filtering, the percentage of malicious nodes has slightly decreased, which guarantees the integrity of node consensus.

[1] https://github.com/Amos668/N-PBFT.git

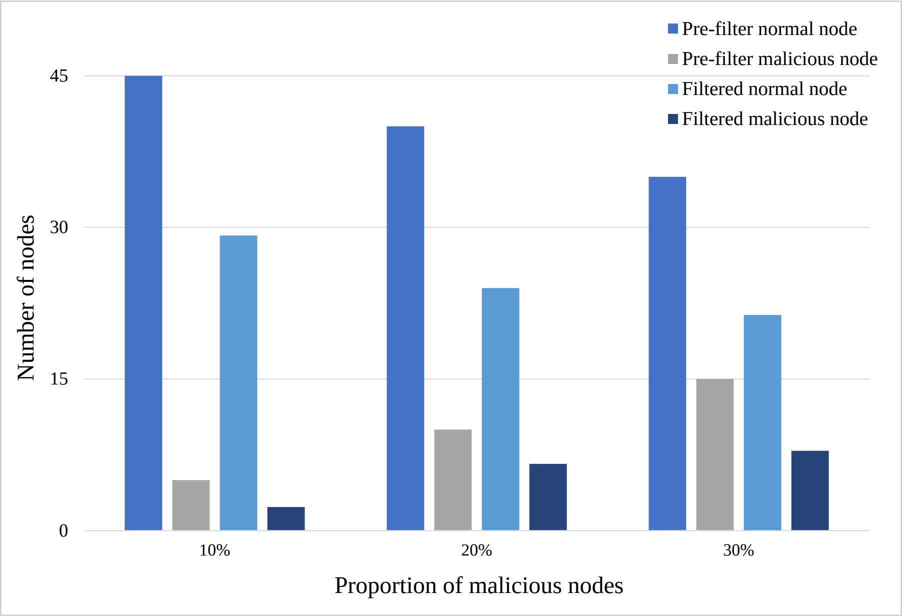

**Fig. 6.** Filtering with 50 nodes.

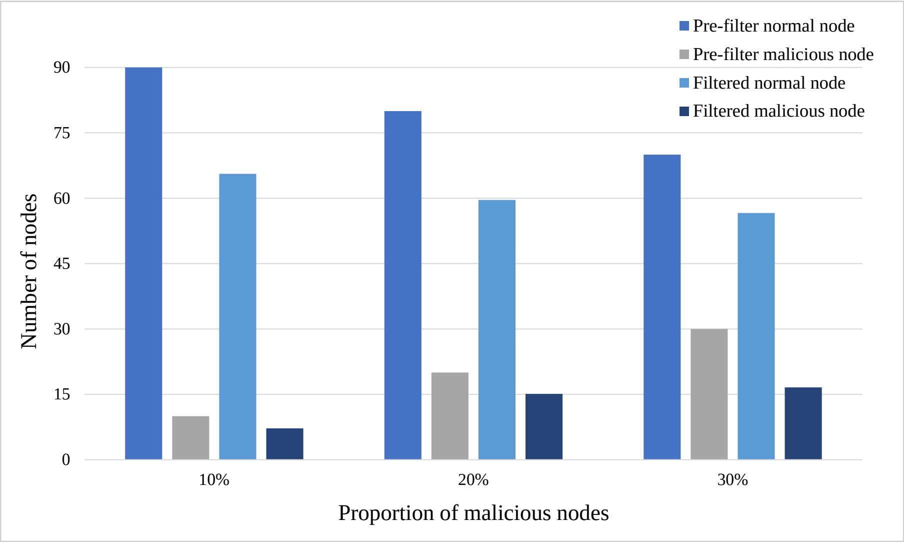

**Fig. 7.** Filtering with 100 nodes.

Therefore, it can be seen that the reputation management system in this paper can effectively reduce the size of nodes participating in consensus, while ensuring the integrity of nodes consensus, and will not make malicious nodes occupy the main body after filtering.

In order to evaluate the performance of N-PBFT, we tested the latency to reach consensus as well as the throughput when the number of nodes in the network was different. We ran the code for the protocols and saved the results at each number of nodes.

Fig. 8 and Fig. 9 show the results. For each test of N-PBFT, we defined three types of nodes in the network: fast and honest nodes, honest but slow nodes and faulty nodes. Slow nodes wait for some time (0.05 seconds in our experiments) before processing incoming messages. Faulty nodes send error messages or are unresponsive.

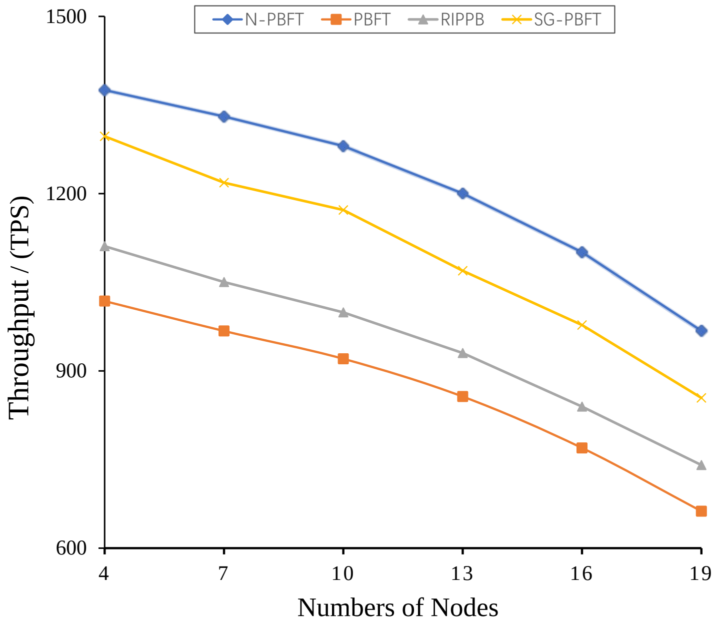

**Fig. 8.** Throughput.

In consensus mechanism, throughput is the amount of transactional data that can be processed over a time interval. $T$ is the time interval between when a transaction request is sent to be packaged into a block and uploaded to the chain, and Transactionsum is the total number of transactions contained in the block during this time interval. The TPS is defined as follows:

$$TPS = \frac{Transactionsum}{T} \tag{10}$$

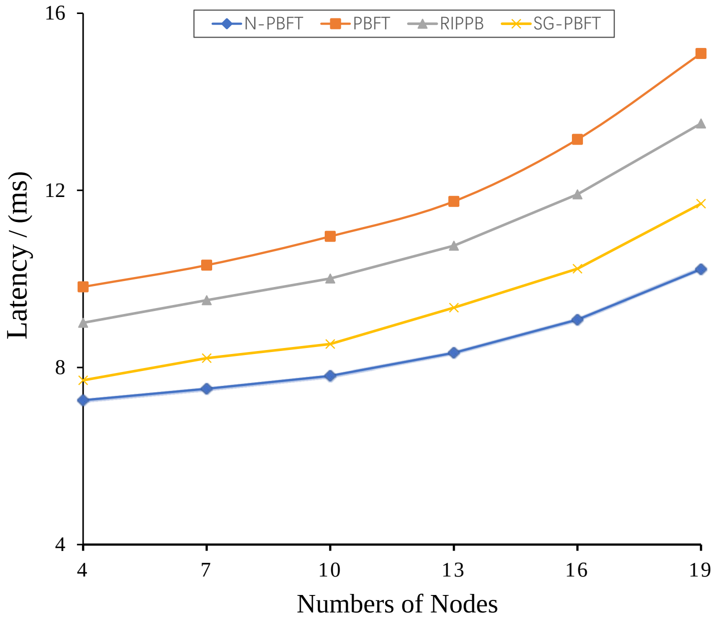

**Fig. 9.** Latency.

The participating nodes of both the consensus mechanisms are fixed as 4,7,10,13,16 and 19 and the throughput of 1000 transactions data volume is tested and compared. The results of throughput comparison between the four are shown in Fig. 8.

In the Fig. 8, as the number of nodes in the network increases, the throughput of all algorithms shows a decreasing trend. But in general, the throughput of the N-PBFT algorithm is significantly higher than that of the PBFT, RIPPB [19] and SG-PBFT [20]. This shows that N-PBFT can effectively improve the throughput of the system and has obvious advantages over the other three methods.

The transaction latency is the time interval between the node sending a transaction request to the primary node and the node confirming the completion of consensus. For the sake of generality, the transaction delay is averaged over 100 transactions and tested with different numbers of nodes. A comparison of the transaction latency of the four algorithms is shown in Fig. 9.

As shown in the Fig. 9, the latency of N-PBFT, RIPPB [19] and SG-PBFT [20] algorithm is significantly better than that of PBFT algorithm. Among them, N-PBFT shows the best improvement. And with the increase of the number of nodes, the transaction delay of PBFT algorithm grows faster, while the transaction delay of N-PBFT algorithm is more stable and grows slowly. Therefore, in the case of more nodes, the advantage of N-PBFT algorithm is more obvious.

## 7 Conclusion

VANETs provides users with safe, comfortable, intelligent and efficient driving experience and transportation services, while improving the efficiency of traffic operation and enhancing the intelligence of social transportation services. Blockchain technology is a decentralized distributed database technology, which improves the security and credibility of the data exchange process. In order to solve the problems of poor scalability of PBFT consensus algorithm and large communication volume when blockchain is combined with VANETs. In this paper, an improved consensus mechanism N-PBFT is designed for the field of VANETs, based on vehicle's reputation system. After experimental testing, the improved PBFT consensus algorithm has significantly improved performance compared with the traditional PBFT, SG-PBFT and RIPPB, which is of positive significance to the landing of vehicle applications based on blockchain.

## References

1. G. Luo, H. Zhou, N. Cheng, Q. Yuan, J. Li, F. Yang, and X. Shen, "Software-defined cooperative data sharing in edge computing assisted 5g-vanet," *IEEE Transactions on Mobile Computing*, vol. 20, no. 3, pp. 1212–1229, 2021.
2. S. Nakamoto, "Bitcoin: A peer-to-peer electronic cash system," 2008.
3. S. A. Khowaja, P. Khuwaja, K. Dev, I. H. Lee, W. U. Khan, W. Wang, N. M. F. Qureshi, and M. Magarini, "A secure data sharing scheme in community segmented vehicular social networks for 6g," *IEEE Transactions on Industrial Informatics*, vol. 19, no. 1, pp. 890–899, 2023.
4. M. Yuan, Y. Xu, C. Zhang, Y. Tan, Y. Wang, J. Ren, and Y. Zhang, "Trucon: Blockchain-based trusted data sharing with congestion control in internet of vehicles," *IEEE Transactions on Intelligent Transportation Systems*, vol. 24, no. 3, pp. 3489–3500, 2023.
5. Z. Su, Y. Wang, Q. Xu, and N. Zhang, "Lvbs: Lightweight vehicular blockchain for secure data sharing in disaster rescue," *IEEE Transactions on Dependable and Secure Computing*, vol. 19, no. 1, pp. 19–32, 2022.
6. Y. Xu, E. Yu, Y. Song, F. Tong, Q. Xiang, and L. He, "$\mathcal{R}$-tracing: Consortium blockchain-based vehicle reputation management for resistance to malicious attacks and selfish behaviors," *IEEE Transactions on Vehicular Technology*, vol. 72, no. 6, pp. 7095–7110, 2023.
7. Z. Zaccagni, R. Dantu, and K. Morozov, "Proof of review: Trust me, it's been reviewed," in *Proceedings of the 2023 5th Blockchain and Internet of Things Conference*, pp. 23–34, 2023.
8. K. Yan, P. Zeng, K. Wang, W. Ma, G. Zhao, and Y. Ma, "Reputation consensus-based scheme for information sharing in internet of vehicles," *IEEE Transactions on Vehicular Technology*, vol. 72, no. 10, pp. 13631–13636, 2023.
9. Y. Cheng, J. Ma, Z. Liu, Y. Wu, K. Wei, and C. Dong, "A lightweight privacy preservation scheme with efficient reputation management for mobile crowdsensing in vehicular networks," *IEEE Transactions on Dependable and Secure Computing*, 2022.
10. A. Gervais, G. O. Karame, K. Wüst, V. Glykantzis, H. Ritzdorf, and S. Capkun, "On the security and performance of proof of work blockchains," in *Proceedings*

*of the 2016 ACM SIGSAC conference on computer and communications security*, pp. 3–16, 2016.

11. F. Saleh, "Blockchain without waste: Proof-of-stake," *The Review of financial studies*, vol. 34, no. 3, pp. 1156–1190, 2021.
12. X. Fu, H. Wang, and P. Shi, "A survey of blockchain consensus algorithms: mechanism, design and applications," *Science China Information Sciences*, vol. 64, pp. 1–15, 2021.
13. W. Li, C. Feng, L. Zhang, H. Xu, B. Cao, and M. A. Imran, "A scalable multi-layer pbft consensus for blockchain," *IEEE Transactions on Parallel and Distributed Systems*, vol. 32, no. 5, pp. 1146–1160, 2020.
14. J. Kang, Z. Xiong, D. Niyato, D. Ye, D. I. Kim, and J. Zhao, "Toward secure blockchain-enabled internet of vehicles: Optimizing consensus management using reputation and contract theory," *IEEE Transactions on Vehicular Technology*, vol. 68, no. 3, pp. 2906–2920, 2019.
15. Q. Zhu, A. Jing, C. Gan, X. Guan, and Y. Qin, "Hcsc: A hierarchical certificate service chain based on reputation for vanets," *IEEE Transactions on Intelligent Transportation Systems*, 2023.
16. X. Zhang, R. Li, and H. Zhao, "A parallel consensus mechanism using pbft based on dag-lattice structure in the internet of vehicles," *IEEE Internet of Things Journal*, vol. 10, no. 6, pp. 5418–5433, 2022.
17. X. Chen, J. Ding, and Z. Lu, "A decentralized trust management system for intelligent transportation environments," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 1, pp. 558–571, 2020.
18. Y. Jiang, Y. Le, J. Wang, and X. You, "Gas-pbft: a game-based node selection consensus mechanism for internet of things," in *2022 14th International Conference on Wireless Communications and Signal Processing (WCSP)*, pp. 17–21, IEEE, 2022.
19. M. Ding, H. He, R. Qiao, and X. Zhou, "Rippb: A robust and improved pbft protocol for blockchain," in *2022 IEEE 17th Conference on Industrial Electronics and Applications (ICIEA)*, pp. 384–389, IEEE, 2022.
20. G. Xu, H. Bai, J. Xing, T. Luo, N. N. Xiong, X. Cheng, S. Liu, and X. Zheng, "Sg-pbft: A secure and highly efficient distributed blockchain pbft consensus algorithm for intelligent internet of vehicles," *Journal of Parallel and Distributed Computing*, vol. 164, pp. 1–11, 2022.