Towards Secure and Practical Consensus for Blockchain based VANET

Article *in* Information Sciences · August 2020

DOI: 10.1016/j.ins.2020.07.060

5 authors, including:

Sowmya Kudva
University of Nevada, Reno
3 PUBLICATIONS **94** CITATIONS
SEE PROFILE

Shahriar Badsha
University of Nevada, Reno
60 PUBLICATIONS **841** CITATIONS
SEE PROFILE

Shamik Sengupta
University of Nevada, Reno
169 PUBLICATIONS **2,295** CITATIONS
SEE PROFILE

Ibrahim Khalil
RMIT University
273 PUBLICATIONS **5,733** CITATIONS
SEE PROFILE

Towards Secure and Practical Consensus for Blockchain based VANET

Sowmya Kudva[a], Shahriar Badsha[a], Shamik Sengupta[a], Ibrahim Khalil[b], Albert Zomaya[c]

[a] *University of Nevada, Reno, USA*
[b] *RMIT University, Melbourne, Australia*
[c] *University of Sydney, Sydney, Australia*

Abstract

The massive adoption of the blockchain-based distributed framework has made it possible to store and transmit Vehicular Ad Hoc Network (VANET) application data transparently, securely, and without a central control point of trust. Introducing an efficient and scalable consensus mechanism, which is one of the most crucial components in the blockchain-based VANET application, is still an open research challenge, given the features related to high mobility vehicular network and resource constraint devices in vehicles. Considering the efficiency, fairness and scalability issues of state-of-the-art consensus protocols like Proof of Work (PoW), Proof-of-Stake (PoS) and Practical Byzantine Fault Tolerant (PBFT), in this paper we propose a new technique called Proof of Driving (PoD), to randomize the selection of honest miners for generating the blocks efficiently for blockchain-based VANET applications. Additionally, we introduce a filtering technique based on Service Standard Score (S_c) of the vehicular miner nodes to detect and eliminate the malicious nodes. Our proposed technique achieves an efficient and fair selection of miners in a blockchain-based VANET application (for example, ride-sharing) and it makes PBFT consensus adaptable in a vast public vehicular network. The proposed method also addresses the efficiency and fairness

⋆Submitted to Information-Sciences Special Issue

Email addresses: skudva@nevada.unr.edu (Sowmya Kudva), sbadsha@unr.edu (Shahriar Badsha), ssengupta@unr.edu (Shamik Sengupta), ibrahim.khalil@rmit.edu.au (Ibrahim Khalil), albert.zomaya@sydney.edu.au (Albert Zomaya)

issues caused by PoW and PoS, respectively. Our extensive experimental results reflect that the proposed method is efficient as well as scalable and, more importantly, achieves smaller consensus sets with higher quality to eliminate the malicious vehicle nodes from participating in consensus. Finally, the security analysis shows that the proposed method is secure and fault-tolerant against various attacks.

Keywords: , Distributed Vehicular Network, Blockchain, Consensus, Proof of Driving (PoD), Service Standard Score.

---

1. Introduction

Vehicular networks are a novel class of distributed wireless networks that have emerged due to the technological advances in wireless sensing and the automotive industry [1]. Vehicle to Everything (V2X) includes possible modes of communication in VANET such as communication pathways between moving vehicles (V2V) equipped with onboard units (OBU) and controller area network (CAN) [2], between vehicles and nearby fixed equipment (V2I), and between moving vehicles and pedestrians (V2P). All of these modes improve road safety, traffic efficiency, and the availability of infotainment services [3].

Distributed road safety-oriented VANET applications help to disseminate safety messages through V2V communication [4] and keep drivers informed about forward and opposing vehicle speed, optimal acceleration, and deceleration settings. The distributed applications related to environmental aspects can reduce fuel consumption and increase safety for passenger cars and trucks [5]. Non-safety applications in vehicular networks provide secondary benefits for vehicles such as travel time savings and in-car entertainment. These applications provide drivers with ways to make more informed route choice decisions. It can also supply drivers with important information on parking congestion and rates. Certain VANET applications can improve the safety of pedestrians crossing at intersections and facilitate carpooling and ridesharing via V2P channel, by providing real-time information about rides [6]. Vehicle platooning systems are another advanced application that enables inter-vehicle (V2V) sensor data sharing via VANET [7]. In general, applications provided by this network tremendously enhance the driving and traffic efficiency by sharing sensor data using reliable communication capabilities of the deployed sensor networks [8, 9].

However, there are significant issues in establishing an effective data sharing network for any VANET application [10]. The vehicle data needs to be shared securely and in real-time with one's community of interest. Because of the openness of communication networks, VANET security can be compromised. Therefore, before deploying the attractive VANET application into practice, security, and privacy issues in VANETs must be resolved [11, 12]. Additionally, these applications require efficient protocols to achieve confidentiality [13], data integrity and authentication [14]. Otherwise, false and untrustworthy information can be sent by malicious vehicles, or some important messages cannot be disseminated accurately in real-time. Distributed vehicle to pedestrian (V2P) communication in VANET is investigated in [6], which relies on using sensor data collected by the nodes in the VANET. If the sensor data is tampered, it leads to collateral damage to neighboring vehicles, drivers, and passengers. These challenges tremendously affect the data sharing services in VANETs.

In this context, various initiatives have been recently launched to investigate the capability of the blockchain technology in securing vehicular data over a tamper-proof decentralized ledger [15, 16]. Blockchain is an emerging decentralized and distributed computing paradigm that underpins the bitcoin cryptocurrency [17], which can provide security and accountability [18] in Peer-to-Peer (P2P) networks. In some cases, it may support privacy [19] in conjunction with cryptographic enhancements. Since its introduction through bitcoin, this technology has evolved beyond cryptocurrency to support the deployment of more general-purpose distributed applications in diverse fields [20, 21, 22, 23, 24, 25] and hence there have been many efforts to implement blockchain-based VANET application to resolve critical issues of information dissemination [26, 15, 27]. For instance, ride-sharing or car-pooling application is most widely accepted and used by the public and the government [28, 6]. Also, the car-sharing service based on the Dedicated Short Range Communications (DSRC) in VANET [29] is another attractive choice for the blockchain implementation.

1.1. Motivation

In this paper, we highlight the ride-sharing as one of the VANET applications to better motivate the proposed method. Existing centralized infrastructure in ride-sharing introduces waiting time for the clients and requires resources to manage and monitor the rides and maintain the liveliness of the system. Specifically, the centralized system manages ride requests made by

passengers in a queue and lets drivers view the queue to accept requests. The passenger receives the accepted driver's service details after the centralized system sets a match. Upon successful transportation service, the payment is processed from the passenger's account to the driver's account via the central managing system. The complexity of this system necessitates the need for a simple, faster, and a decentralized system involving direct communication between vehicles and passengers (clients) using the Adhoc network. Such decentralization would make the systems less vulnerable to the single point of failure and eliminates the need of the third party to manage rides, low-cost management, and robustness. However, while achieving decentralization in this VANET application, the vehicle or the passengers' data needs to be shared securely and stored in a decentralized manner. Additionally, the records of ride-sharing should not be disclosed, altered, or even deleted. In this context, blockchain makes it possible to store and transmit information transparently and securely by safeguarding integrity of data without a central control point of trust[1].

As the core of any blockchain-based system, a consensus algorithm directly affects the performance of the blockchain system [16] in terms of transaction confirmation delays. In specific VANET applications, RSUs are utilized as pre-authorized validators to perform consensus [31]. However, for the ride-sharing application, the validation is minimal, and thus usage of RSU keeps them underutilized. The classic Proof of Work (PoW) [32] might not be the best way to contribute to the vehicular network due to its focus on high computation power. On the other hand, the Proof of Stake (PoS) may only incorporate stake-based selection, which may raise unfair conditions. This necessitates us to find a proof variant that is adaptable to the nature of the vehicular nodes (mobility and resource constraint devices), more randomized, and finally impartial and dynamic such that only a smaller number of randomly selected vehicles get to perform the mining. Hence in this paper, we explore solutions to the problem of *(1)* how can we identify a set of criteria to optimize the number of vehicular nodes performing mining in VANET application *(2)* how randomness can be introduced in the model for miner node selection yet without compromising the quality of miners.

[1]potential privacy concern where the data can be publicly accessible, can be carefully addressed using zero knowledge proof [30].

1.2. Contribution

In this paper, we propose an efficient and scalable proof-variant that can be implemented in various VANET applications based on blockchain, specifically for blockchain-based ride-sharing [33], in combination with Practical Byzantine Fault Tolerance (PBFT) consensus [34]. We introduce a new proof-variant named Proof of Driving (PoD) that brings randomness in a blockchain-based public ride-sharing VANET application, in which the actively driving vehicle nodes earn certain driving coins ω in their registered account wallet based on the distance traveled. The idea of mining is similar to that of PoW, but the proposed PoD is designed to consume fewer resources than PoW. Other features that are considered while designing PoD are, it should be more (i)*Accessible* to all, (ii) *Fair and Square* from the selection perspective (iii) *Economical* from the computation perspective, (iv) *Resilient and Robust* against attacks.

After developing the PoD, we further narrow down the pool of miner nodes by considering Service Standard Score S_c of each node, which is an indicator of various performance factors such as error rate, success rate, and reputation. This is an essential component in our design, as we group the nodes to achieve more than 50% of total S_c of the network while minimizing the actual number of nodes contributing to that S_c. Finally, a group with higher S_c value can perform the consensus on the order of the ride-sharing transactions using classic deterministic Byzantine agreement protocols, such aså PBFT (note that in this paper we focus on developing efficient and fair miner node selection process and hence applying PBFT is beyond the scope of this paper). As a result, a block containing ride-sharing records is created, which finally gets added to the blockchain.

The main contributions of this paper are summarized as below.

- We introduce a new proof variant named Proof of Driving (PoD) into the design of blockchain-based ride-sharing service in VANET, which consumes fewer resources than PoW, maintain fair selection as well as the randomness of consensus nodes in a public distributed network of vehicles.

- We design a real-time service standard score protocol S_c to efficiently optimize the number of miner nodes considering their performance while eliminating the poor quality or malicious nodes from being part of the consensus.

- We present extensive experimental and security analyses on proposed PoD and S_c protocols to show the effectiveness, security, and feasibility in terms of miner node selection.

The rest of this paper is organized as follows. Section 2 discusses about the prelimanaries of proposed study, section 3 and 4 present the related work and framework details along with overall system architecture respectively. Section 5 and 6 present the proposed PoD and S_c protocol respectively. Section 7 and 8 discuss about the defense analysis against possible attacks and the experimental validations respectively. Finally Section 9 concludes the paper.

2. Preliminaries

In this section, we introduce the blockchain technology and popular consensus mechanisms that are commonly used to construct a blockchain-based system.

2.1. Blockchain

The original premise of blockchain [35] is to establish a peer to peer (P2P) network with the main idea of maintaining a distributed ledger for monetary value transfer such as Bitcoin where no bank or any other financial institution is required to make a value transfer transaction with anyone else on Bitcoin's blockchain network. Peers in the network use their processing power to solve the mathematical puzzle to validate transactions without any centralized administration. Once the block is validated, it is added to the blockchain, and the transaction gets visible to the entire network and thus become immutable.

2.2. Anatomy of Current Consensus Mechanisms

Consensus in a blockchain is a process where all the peers of the network reach a common agreement about the present state of the distributed ledger. At present, the most common consensus algorithms are PoW, PoS, and PBFT. From the emergence of Bitcoin to today, there are more than 30 consensus algorithms [36], most of which are based on the above three consensus algorithms.

Proof of Work (PoW): PoW consensus protocol, which was introduced via the Bitcoin, requires each validating user to prove that he/she has performed

a computational action by solving complex cryptographic problems using their computational resources. The one who eventually finds the solution can confirm the transactions and write the block onto the chain. Hence PoW is computation-intensive as each miner needs to deliver a hash querying rate as high as possible to win the puzzle-solving race [32]. Hence it is unsuitable for the vehicular network to establish a blockchain-based system by using computation-intensive consensus schemes for resource-limited vehicles.

Proof of Stake (PoS): On the other hand, PoS emerged in the form of the Peer-coin [37] project back in 2012 was an attempt to overcome the PoW mining algorithms in terms of the high resource consumption. PoS completely replaces the mining operation with an alternative approach called validating a new block involving a user's stake or ownership of virtual currency in the blockchain system. The cryptographic calculations in PoS are much more straightforward for computers to solve. Nevertheless, a rich validator may keep on winning the bid for the next block to be validated, and accumulate the block reward. Hence selection by account balance would result in undesirable centralization around those large stakeholders and raise significant trust concerns.

Practical Byzantine Fault Tolerance (PBFT): PBFT [34], as opposed to the other discussed consensus mechanisms, was not explicitly introduced for blockchain, but several years before the seminal bitcoin paper by Nakamoto and tries to answer the open questions posed by other consensus mechanisms. This consensus model gets the idea from byzantine generals problem in which votes of all generals need to be transmitted to all others through broadcasting. It works efficiently even when a portion of the network is faulty. However, PBFT only scales to few tens of nodes since it needs to exchange messages to reach consensus on a single operation among n servers resulting in $O(n^2)$ complexity. Hence for it to be efficient in ride-sharing VANET application set up, the number of nodes performing consensus in the network should be significantly reduced.

3. Related Work

Various consensus mechanisms have been studied and utilized recently to establish blockchain-enabled vehicular networks for secure data sharing. Authors in [38] have presented a reward-based intelligent vehicle communication

exploring the features of open ledger technology based on the PoW consensus mechanism. Authors in [39] also proposed a blockchain-based reputation system in vehicular networks to be able to judge the received messages as either true or false based on ratings generated by a temporary center node chosen by a PoW consensus mechanism. Researchers in [26] employed the joint PoW and PoS consensus mechanism, to propose a decentralized trust management system for the credibility of received messages from neighboring vehicles using Bayesian Inference Model. In all of these schemes, PoW is incorporated, which is computation-intensive and time-consuming. However, it is not feasible for a dynamic network consisting of resource limited vehicles in a real-world scenario.

In [31], authors designed their own consensus phases based on the Byzantine Fault Tolerance algorithm while proposing a privacy-preserving incentive announcement network based on the blockchain via an efficient anonymous vehicular announcement aggregation protocol. Authors in [40] indicated that Delegated PoS (DPoS) is particularly suitable for vehicles to establish blockchain-based vehicular networks in ITS ecosystems. In [41], a blockchain-based distributed framework for the automotive industry in the smart city is proposed that includes a novel miner node selection strategy based on a fruit fly algorithm with underlying PoW consensus mechanism. Later, authors in [42] proposed an enhanced DPoS consensus scheme with two stage soft security solution for secure vehicle data sharing and ensured secure miner selection based on reputation to establish blockchain-enabled Internet of Vehicles (IoV).

Several other works have chosen many different approaches. In [43], authors proposed a random miner selection consensus protocol to elect a miner. Here the general intent is similar to that of bitcoin. However, they employ a substantially different approach called Proof of Interoperability (PoI), an alternative method for network consensus that avoids some of the disadvantages of PoW eliminating wastage of computational overhead such as power. However, inefficient miners have a chance to be selected in random miner selection, which might increase the latency in the blockchain. In [44], proof of reputation is proposed in which reputation based weighted voting is performed as an alternative way to provide a strong deterministic consensus in a permissionless distributed blockchain system. However, our work incorporates not just reputation but also considers the contribution in terms of hosting shared drives by spending vehicle resources such as fuel.

In [45], authors have proposed a permissioned vehicular blockchain called

Parkingchain, where the parked vehicles (PVs) can share their idle computational resources with service requesters (SRs). They have utilized the blockchain technology to design an efficient resource sharing. A subjective logic-based Delegated BFT consensus mechanism has been presented to enhance the consensus process in Parkingchain. However, it applies only to the permissioned network.

Authors in [46] presented a new technique called Algorand that uses a new Byzantine Agreement (BA) protocol to reach consensus among users on the next set of transactions. BA chooses committee members randomly among all users based on the users' weights. In contrast, in our work, we make use of vehicle dynamics such as distance covered to generate proof on the fly while vehicles are moving efficiently. More specifically, we employ an approach consisting of proof variant PoD and filtering technique S_c to efficiently choose nodes for adopting PBFT in an attempt to make it scale in a larger network of vehicles. With these proposed mechanisms, we aim to overcome the drawbacks of existing approaches and provide fair participation and randomized selection while also considering the quality of miners in terms of service score.

4. Framework details

In this section, we first give an overview of the ride-sharing application based on blockchain. Then we focus more on presenting the proposed proof variant, the threat model for the system, and system methodology.

4.1. Ride-Sharing VANET Application based on Blockchain

In order to reduce fuel costs, efficient ride-sharing systems are necessary. So the car-sharing service based on the Dedicated Short Range Communications (DSRC) frequency in VANET is more attractive. Vehicles are equipped with advanced communication devices and passengers who need to find rides use their handheld devices with wireless transmission modules to transmit messages via VANETs V2P mode.

When a ride requesting passenger sends a request message, if a nearby vacant car driver receives the request, she sends a response message back to the passenger to accept the ride request. Since the connectivity of VANET is guaranteed, the cars whose response message first received by the passenger have the shortest driving distance to the passenger. The passenger then sends an ACK back to indicate his confirmation to ride.

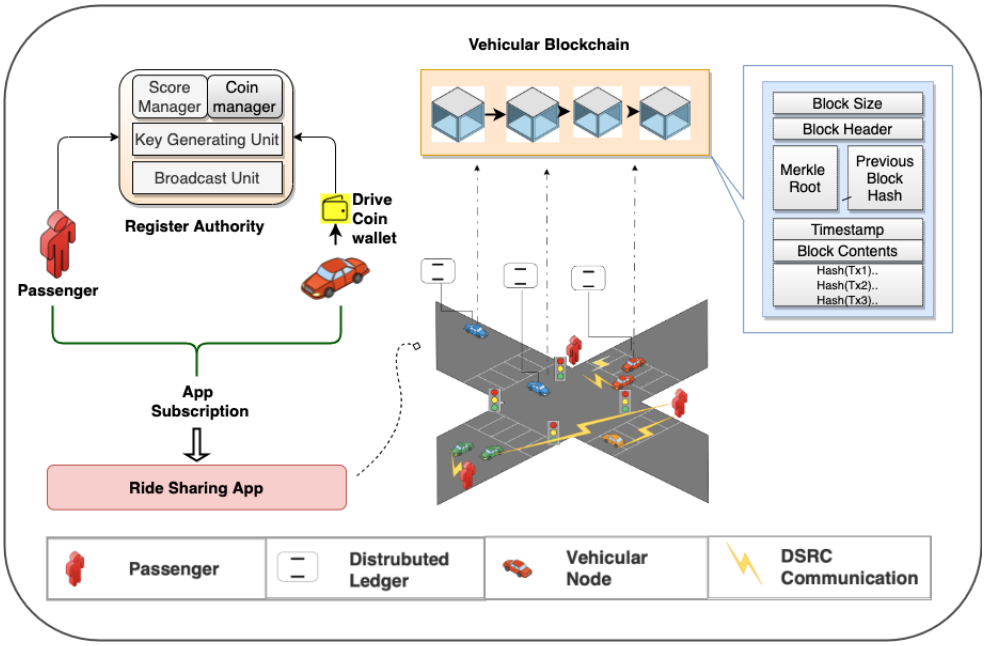

Figure 1: Overview of Blockchain based Vehicular Network

Our earlier work PEBERS [33] is aimed at implementing a ride-sharing service in a decentralized manner by using the blockchain technology and smart contracts to harvest the benefits of decentralized collaborations and the blockchain perspectives. Each vehicular and passenger node will have an associated account unique to each user on the blockchain platform. All the records of rideshare are considered as transactions that are validated by the mining nodes and finally added into immutable blocks of the distributed ledger by a consensus process.

4.2. Overview of Our Proposed System Model

We have seen in our earlier discussions that PBFT consensus in a public blockchain system scales very poorly. Hence design goal of our work is to make PBFT usable in a public blockchain network by introducing proof of driving in a VANET application. A decentralized network in our model mainly includes several vehicular nodes, passenger handheld devices that communicate with each other by sharing information via DSRC radio and Register Authority (RA), as shown in Fig 1. It illustrates how different components of the system are connected. Detailed descriptions of these compo-

nents are given in the following.

Register Authority (RA): RA is a legitimate and authoritative component of a blockchain system. To maintain secure communication between every vehicle and passenger nodes, they are made to register with the Register Authority (RA). It is the aptest method in current research work to have permission granting authority like RA. This is similar to [47], where System Administrator is considered into the design for registration, in our work we consider the existence of a RA that administers registration to the blockchain network. We assume that RA is configured on Road Side Units (RSU) in a decentralized manner that serves to shield the system from traffic and DDoS attacks. RA has four important units: *Key generating unit*, *Coin Manager*, *Score manager* and *Broadcast unit*.

The *Key generating unit* is responsible for issuing public parameters and cryptographic keys to the vehicles and the passengers. It keeps the database containing the linkability between the users' public keys and the account identity with a high level of security. The *Coin Manager unit* is responsible for generating new coins in the system and evaluating the coin earnings of the network and, *Score manager unit* is responsible for evaluating the S_c of each vehicle based on various parameters recorded in the network about the vehicle nodes. Finally, *Broadcast unit* is responsible for performing the network level broadcast in specified intervals.

Vehicular Nodes : In our blockchain environment, each vehicular node is represented as V_i where $i \in \{1, 2, 3, ..., N\}$ and N is the total number of vehicular nodes in the network. Vehicles are installed with advanced communication devices, wireless transmission modules like IEEE802.11, GPS receivers, and can perform simple computations such as calculating location coordinates, determine their location by matching the electronic maps and verify transaction signature.

Passenger Devices: Passengers also need the necessary hardware equipment to support communication. Any passenger device is denoted by P_j where $j \in \{1, 2, 3, 4...m\}$ and m is the total number of passenger nodes in the network. They are equipped with wireless pocket devices to call vehicles for the ride, which can be regarded as a cheap device with an electronic map and simple input/output. These devices are termed as light nodes as they only participate in obtaining services and leaving service ratings. They are not the part of the consensus process.

Node Categories: The total number of vehicular nodes in the network is represented by N and N_{PoD} represents the number of intermediate nodes

that are filtered through PoD. From the filtered nodes N_{PoD}, we arrive at a group of consensus nodes representing N_{Sc} after further filtering based on service score S_c. Finally, N_{Sc} are the entities responsible for carrying out consensus and approve transactions recorded to be added into an immutable ledger. We also represent unfiltered nodes from the second stage as set N'_{Sc}. Table 1 summarizes all the used notations.

Permissionless Blockchain and Transaction: We consider a permissionless blockchain [33] that handles all the ride-sharing records. Any vehicle V_i or passenger P_i can use the platform to obtain services after correct registration. Transactions are the essential communication primitive in blockchain for information exchange among entities in the system model. In our model, the transaction is the record of every event such as passenger requesting for the ride, driving vehicle accepting the request with a response, passenger acknowledgment at a particular date, and time along with the amount of fee paid.

4.3. Threat Model

We consider the proposed model to be vulnerable to probable internal and external attackers, which can severely interfere with the operation of the ride-sharing application. The specific adversarial attackers considered in this paper are internal malicious vehicular node and external adversaries compromising certain network nodes. Both of these attacks could try to compromise the security of the ride sharing system in our model. However, we assume the property of honest majority, which means a good majority of vehicular nodes in the network are benign and honest out of N total nodes. Moreover, due to the periodical reputation check in the model, the compromised vehicular node cannot be controlled by attackers for a long time. Based on this fact, it is assumed that attackers can only compromise a small portion of vehicles during a short period of time.

4.3.1. Internal Attacks

We consider that it is possible for an arbitrary number of vehicular nodes who might plan to collude to guess the target hash in the proposed PoD mechanism and infiltrate the consensus group N_{sc}. These colluding nodes might exchange driving distance information to gain control over the central mechanism of PoD and be part of N_{pod}. However, in a large and dynamically changing vehicular network containing N nodes, number of colluding vehic-

Table 1: Important Notations

Notation	*Description*
N	# Total Nodes
N_{PoD}	# PoD selected Nodes
N_{Sc}	# S_c filtered nodes
N'_{Sc}	# S_c unfiltered nodes
V_i	Vehicular Node
P_i	Passenger
id_{pi}	Pseudo identity of passenger
id_{vi}	Pseudo identity of vehicle
PK_{vi},PK_{pi}	Public Key of the vehicle and passenger.
SK_{vi},SK_{pi}	Secret Key of the vehicle and passenger.
WID_{vi}	Wallet id of the vehicle
ω_{total}	Total coins generated.
ϕ_{target}	Broadcast Target Hash
S_c	Service Standard Score

ular nodes 'x' are assumed to be minority i.e x becomes insignificant when N is a big number, thus eliminating the successful attempts of such attacks.

4.3.2. External Attacks

Although we have assumed the property of honest majority, this does not completely rule out the possibility of having malicious miners. Potential miner vehicular nodes out of N total nodes are honest but curious and may also be vulnerable to be directly compromised by attackers and become malicious miner candidates. Such adversary can control many nodes in the vehicular network and can infiltrate the final consensus group N_{sc} in our model. Compromised miner nodes can modify or delete transactions to be added into the block and can deteriorate the quality of service of the proposed model. Let $f \subset N$ be the set of the malicious nodes. However, we discussed above that such an adversary can control at most $f < \lfloor \frac{N}{3} \rfloor$ nodes for a short

duration of time. This threat for the proposed PoD scheme can be reviewed as the probability $P(f)$ of the malicious node given its coin balance, getting filtered as the intermediate consensus node N_{pod} from f malicious nodes.

4.3.3. Observe-Act Attack

Another type of attack could be Observe-Act Attack. Here the attacker observes the reputation score distribution of the normal nodes. Let a_i, b_j, c_k be the reputation score distribution for few random honest nodes i, j, k. Then the attacker can hijack the reputation score calculation to update one of its controlled malicious nodes l to have the reputation score d_l to be in the range of normal node's reputation. It indicates that the malicious nodes l would now have equal chances to be grouped into the final group of consensus nodes N_{sc}.

The coverage of the explained threat model, i.e., how to constrain the adversary's ability of infiltrating the consensus group N_{sc}, to meet the security and liveliness requirement for the proposed system design is discussed in Section 7.

4.4. System Methodology

In this section we outline the overview of our overall system methodology.

- *System initialization*: The vehicle and passenger nodes joining the blockchain network for the first time submits thier identification details such as name, address, Electronic License Plate (ELP) number for vehicles, Personal Identification Number for passengers and other required identification details to the RA. The key generating unit of RA, in turn, assigns a pseudo identity id_{vi} for vehicular nodes V_i and id_{pi} for passenger P_i along with generating a public-private key pair by using Elliptic-Curve Diffie–Hellman(ECDH) key agreement protocol.

  A vehicle node V_i also obtains a driving-coin wallet address WID_{vi} from the authority. The authority generates a mapping list $\{id_{vi}, PK_{vi}, SK_{vi}, WID_{vi}\}$ for each V_i and $\{id_{pi}, PK_{pi}, SK_{pi}\}$ for each passenger. This identification vector of the V_i and P_i along with its details, digitally signed by the RA are stored as a single transaction in the identification ledger.

- *Genesis Block creation*: The blockchain begins with a genesis block on top of which are stacked the successor blocks. Genesis block in our

model contains empty transaction lists, placeholder for S_c values of mining nodes. When a vehicular node joins the network for the first time, it would receive a default of 0.5, 0.0 and 0.0 for reputation r_i, error rate e_i and success rate s_i respectively. More details about the need of these parameters are provided in the section 6.

- *Ride sharing Record*: Records of passenger REQ messages containing passenger public key, encrypted locations and ACK messages are logged as transaction in the distributed ledger. Records of RESP message from vehicular nodes responding to REQ message are also logged respectively by each responding vehicular nodes. These records might contain vehicular public key, location information, REQ id to which the response is being sent, total distance between the points of passenger and driver. These digitally signed transactions get accumulated periodically in the unconfirmed transaction pool of distributed ledger maintained by all the members of the network.

- *Mining by Vehicular Nodes*: To avoid the mining process carried out by a huge size of mining pool and to improve block creation time using PBFT consensus in vehicular network, we make use of new proof variant called PoD and a filtering technique based on service standard score value S_c for selecting miner nodes which are discussed in detail in section 5 and 6. We select optimized number of best possible miners to compute and approve blocks in our framework during block generation.

- *Achieving consensus by adapting the existing PBFT protocol*: We use PBFT as the underlying consensus protocol. Developed by Castro and Liskov [48] in 1999, PBFT is the first BFT consensus protocol that has gained wide recognition for practicality. In our system, after efficiently identifying the miner nodes and forming a group of best possible nodes constituting higher S_c value sum, we use existing PBFT protocols to generate and broadcast blocks. Nodes in this stage are sequentially ordered with one node being the primary(or the leader node with highest S_c value) and others referred to as secondary(or the backup nodes). Every node in N_{sc} creates blocks taking turns randomly. For the very first block generation only, since all filtered nodes have the same S_c value, a leader node is chosen in this stage based on time of joining the network.

Once a leader node successfully generates a block, it is validated by the secondary nodes and all honest nodes help reach a consensus regarding the state of the system using the majority rule. A PBFT enabled distributed system provides a practical byzantine state machine replication that can work even when malicious nodes operate in the system with an assumption that honest nodes are more than $2f + 1$ where f is the number of faulty nodes. Normal execution of the protocol can be summarized as:

1. The leader of the selected group of miners successfully generates a block and proposes it to secondary nodes through broadcast.
2. Upon receipt of a block, secondary nodes verify the block against its replica and check if the block is valid and is signed by a leader node.
3. If the block is valid, each secondary node approves the proposed block and acknowledges the leader node.
4. Upon receipt of 2/3 signatures from secondary nodes leader node commits the block and broadcasts the block to the whole network and hence adds a block permanently to the blockchain. The block contains the merkle root of all the transaction hashes, previous block hash, and timestamp and current block hash. In our proposed framework, instead of voting by the whole network, only the nodes in final consensus group N_{Sc} will send votes to the current leader. Hence reducing the the communication cost of broadcasting the votes.

Incentives are released to the block creating node from the transactions validated and faulty node if found is penalised for subverting the network by downgrading its S_c value to negative and revoking its registered keys from RA of VANET application thereby eliminating it from taking part in the network.

- *Rating the Leader Node:* Every peer node whose transaction is included in the current block is notified when the new block is created. These nodes rate the leader node based on its efficiency in confirming transaction. Additionally, vehicular nodes serving the ride share platform is also rated by the passengers for their quality of service. Both of these factors accounts for r_{vi} of each vehicular node. Consensus nodes are rated based on the outcome of block creation. Alongside aggregated r_{vi}

fetched from several nodes, success rate s_{vi} and error rate e_{vi} for each consensus node is also managed by score manager. This mechanism is discussed in more detail in section 6.

5. Proof-of-Driving (PoD) Protocol

In this section, we discuss the protocol assumptions, and then we present the design methodology. As mentioned in the above section 1.2 and 4, we develop a proof of driving and service quality based selection protocol, which is multistage filtering of miner nodes, as shown in Fig 2. This method can be applied in blockchain-based ride-sharing [33, 30] to achieve efficient and scalable consensus.

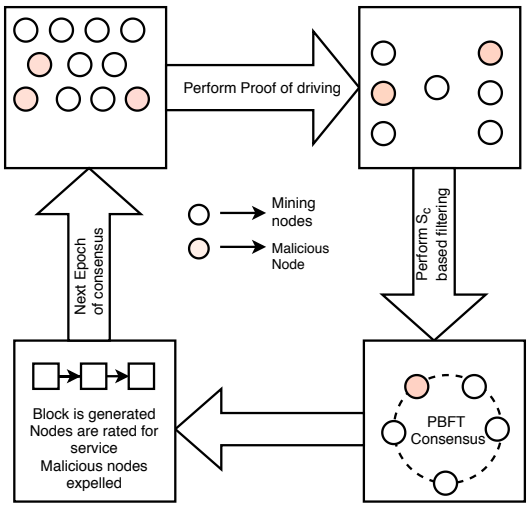

Figure 2: Overview of Blockchain Based ride sharing application Proposed Method

5.1. Assumptions

PoD can run securely in an application scenario with the following assumptions.

- As it is crucial for the vehicles to be connected to the blockchain vehicular network, they are assumed to have configurations such as sensors, data storage, computing resources, and capabilities as well.

- As the honest nodes drive, they earn driving tokens in the associated coin wallet. The driving coin's wallet is implemented with enough security measures, and it is assumed to be impossible to tamper the coin balance.

- It is assumed that the PoD algorithm always returns some number of nodes from a very large pool of potential mining nodes. There should never be a case that no nodes are getting selected.

- All the communication between different entities is assumed to take place via a secure channel, and the design of the secure protocol is not in the scope of this work.

- As this proof only helps the node to be able to get selected intermediately as the potential consensus node, there will not be any reason to drive in order to get only driving tokens. These tokens are non-exchangeable/non-tradable in return for any services within the vehicular network.

- We assume dishonest or greedy vehicular nodes exist who always want to be part of consensus to earn more incentives. They might tamper with distance traveled to obtain a higher number of coins. However, higher coins do not guarantee the selection of nodes.

- We assume the existence of malicious nodes that tamper the records or insert false records in the ledger. In our research, malicious nodes remain below $1/3^{rd}$ of the total network. While we assume it is impossible for the attackers to control all the nodes within the network, causing eclipse attack, designing countermeasures of eclipse attack are beyond our research scope.

5.2. PoD Design

PoD proof scheme encourages vehicular nodes to compete in a nomination process where every vehicle has a fair chance. We define the methodology based on proof of driving in the step by step process as follows and provide the pseudo-code in *Algorithm 1*

- *Step 1:* Actively ride hosting vehicles V_i carrying passengers on board earn driving coins ω in their driving coin wallet WID_{vi} for every certain distance as rewards. For instance, each node earns a driving coin ω for

every two miles of driving as per our design. New driving coins are created and tracked at coin generating units. This competitive process takes place just as it happens in bitcoin network [17]. It also encourages vehicles to provide more services to passengers.

Algorithm 1 Algorithm for Proof-of-Driving

Input: $\omega_{total} \rightarrow$ Total coins generated
Output: $N_{PoD} \rightarrow$ List of intermediately filtered nodes
Initialisation : Total Number of active driving nodes N
Total coins generated ω_{total}
1: *Calculate the* $avg_{current} = \frac{\omega_{total}}{N}$
2: *Broadcast Target hash*
$\phi_{target} \rightarrow HASH(avg_{current})$
3: **for** $i = 1$ to N **do**
4: N_i *calculates* $\rightarrow HASH(\omega_i)$
5: *Submit the hash to the Broadcast Unit*
6: *Broadcast Unit checks*
7: **if** HASH(ω_i) $<\phi_{target}$) **then**
8: *Add* N_i *to* N_{PoD} *list*
9: **else**
10: *Do nothing*
11: **end if**
12: **end for**
13: *Broadcast Unit notifies list of nodes to be in* N_{PoD} *group*
14: **return** N_{PoD}

- *Step 2:* At certain point of time when Broadcast unit of RA invites nominations for miner selection process, it queries coin generating unit for average earnings of the network. Upon this request, coin generating unit of the system calculates the average number of coins $avg_{current}$ that are generated so far. We consider hash of average coins as the target hash for that particular round of miner selection. Hence Target hash value is based on average coin earnings value of the entire network. No member of the network can predict the target hash at a particular time, because, it would be purely based on active vehicles at that instance and distance travelled by them.

$$\phi_{target} \rightarrow HASH(avg_{current})$$

- *Step 3:* The Broadcast unit of RA announces the target hash ϕ_{target} to every node in the entire network. After receiving the broadcast message, node V_i runs a simple computation to check if its earned coins ω_i in the wallet passes the criteria or not. V_i calculates the value of HASH (ω_i) and checks the condition programmatically with the inbuilt application code.

  $$HASH(\omega_i) < \phi_{target}$$

  If the value is false, no further steps are taken and the node continues to participate in the coming nomination processes. Also coins earned are reset to 0. Else if it's true, then V_i notifies {HASH (ω_i)} to the broadcast unit. Here, we are checking if each node passes the set criteria. Hence the complexity of the PoD scheme is O(n). A vehicular node might also choose to exempt from taking part. In all those cases as well, wallet balance will be reset to 0 at the end of the current round.

- *Step 4:* After receiving verified notification messages from V_i nodes, the system chooses these nodes and then assigns them into the set of intermediate consensus node list represented by N_{PoD}.

Theorem 1. *Although in PoD, the reference of driving coins is taken as one of the selection criteria, the selection process is more random and the final consensus group can never consist of majority of faulty nodes, provided that the total number of faulty nodes in the network do not exceed f.*

Proof. Let us consider, out of the N total mining nodes f are adversaries (bad nodes) such that $f < \frac{N}{3}$. Let us assume we have A which is the set of honest nodes out of N such that $A = N - f$. Let c denote the group of nodes that are filtered from PoD.

If we have to randomly select c out of N nodes based on the proof of driving, then the probability to pick f malicious nodes is given by $P_f = \frac{f}{n}$ but $f < \frac{N}{3}$. Hence the probability of choosing bad nodes from this stage is less than $\frac{1}{3}$. Additionally probability of choosing honest nodes is $P_A = \frac{N-f}{N}$ is higher and thus we can say that our protocol is efficient enough not to filter the malicious nodes into the next stage. □

Theorem 2. *With PoD, the mining in blockchain based vehicular network is more randomized, impartial and dynamic than existing PoW and PoS systems in terms of resource consumption and fairness.*

Proof. In PoW, target hash is set based on block difficulty ϕ for the current round, and the miners consume their CPU power to continuously hash the block to find a particular nonce below the target. However in PoD, coins earned ω by hosting ride share is a proof of driving for each vehicle. In PoS, the mining is based on user's stake or ownership of virtual currency in the blockchain system causing partial centralization. In contrast to that in PoD, selection of miners is randomized based driving coin balance, therefore, the highest coin earner is not always the block validator. Let us consider, N as the total active driving nodes at a given window time t. Note that N is a stochastic, dynamic and discrete component which is changing with time when VANET is quiet large. Since the target hash is set as the hash of average number of coins $avg_{current}$ earned by the entire network, the coins earned at a particular time t is beyond any one's guess as one vehicle does not know about other vehicles' driving patterns and how many active vehicles are taking part in the process. PoD enables vehicles to provide proof with one time hashing unlike continuous hashing in PoW. Additionally, the system uses SHA256 to generate the hash of average coins which guarantees that attacker cannot guess the final average making it difficult for a malicious user to hijack the proof in order to increase the probability of being selected as miner node. We want to emphasis here that, PoD mechanism is designed by picking the best features of existing mechanism yet preserving the security features as discussed in the future sections. □

6. Service Standard Score S_c Protocol

Once the PoD process is completed, in the second stage miner node selection takes place in the vehicular network where we make use of S_c to choose a group of final consensus nodes N_{Sc} which is a subset of N_{PoD} capable of controlling the consensus protocol.

Algorithm 2 Algorithm for S_c calculation

Input: r_{vi}, e_{vi}, s_{vi}, N
Output: $Sc_i \in (0,2)$ service score of corresponding V_i node
1: **for** $i = 1$ to N **do**
2: *Calculate the* $Sc_i = r_{vi}$-e_{vi}+s_{vi}
3: *Broadcast* Sc_i
4: **end for**

Algorithm 3 Algorithm for S_c based grouping

Input: Sorted Array of S_c value for N_{PoD} nodes where N_{PoD} is the number of filtered nodes from PoD
Target S_c Sum

Output: List of selected node N_{Sc} from the N_{PoD} number of nodes

Initialisation : Empty List N_{Sc} of miner nodes
Target S_c = 1+(0.50 Total S_c)*
CurrentSum = S_c[n-1]
Loop through remaining S_c values
for $i = n - 2$ to 0 **do**
  if *CurrentSum* $\leq$ *Target S_c* **then**
    set CurrentSum = CurrentSum+S_c[i]
    Add node i to the List N_{Sc}
  else
    Skip node
  end if
end for
return *List of node indexes selected as N_{Sc}*

The group size is formed by the minimum number of miners contributing to more than half of the total S_c value of the filtered mining nodes N_{PoD} from the PoD process. We try to maximize S_c sum with the constraint on the number of nodes. From this stage, selected miners apply PBFT to decide which transactions should be involved in the proposed block and broadcast the new block to the network. The rest of the mining nodes synchronize with the newly appended block.

6.1. S_c Aggregation

A node's trustworthiness is defined by S_c it holds in return to generating a successful block in the past. Such a score reflects the degree of trust that other peers in the community have on the given peer based on their past experiences, and it determines the node's ability to obtain the leadership authority. We identify three important factors for S_c evaluation as *(1)* Reputation r_{vi} *(2)* Error rate e_{vi} and *(3)* Success rate s_{vi}. We illustrate the importance of these parameters and explain how they can be calculated in our design.

- *Reputation* r_{vi}: When a consensus process is completed, and the block is published in the blockchain successfully, every consensus node is rated by the peer nodes either positively or negatively for each transaction validated based on the honest behavior and efficiency that the node has demonstrated in creating a valid block. However, it is also possible that malicious peers can attack the rating process by deliberately giving high or low scores for the consensus nodes. Consequently, the consensus nodes' reputation as miners is either increased or decreased. Hence these ratings are aggregated by the *score manager unit* of RA to obtain the weighted average rating over multiple transactions, which is considered as ground truth for a particular node. Lastly, due to the limited number of malicious nodes, as discussed in section 4.3, these unfair ratings can hardly disrupt the system. The calculation of aggregation can be represented, as shown below:

  $$r_{vi} = \frac{1}{t_x} \sum_{i=1}^{t_x} m \text{ - } \frac{1}{t_x} \sum_{i=1}^{t_x} n \in [0,1]$$

  where r_{vi} is the reputation value of vehicle V_i based on n^{th} consensus round and m and n are the number of positive and negative ratings, t_x is the number of transactions validated by the consensus nodes. r_{vi} is normalized to a value between [0,1]. When a node first joins the network, it gets a default reputation of 0.5.

- *Error rate* e_{vi} : We define the error rate factor e_i for node V_i as the ratio of the number of times a node has failed to generate a valid block e to the total number of attempts made by this node A_{total} to generate blocks. We define the error rate of the nodes which has generated no blocks as 0. This parameter has a negative impact on the S_c of a node

  $$e_{vi} = \frac{e}{A_{total}} \in [0, 1]$$

- *Success rate* s_i : We define the success rate factor s_i for node V_i as the ratio of the number of times a node has created block successfully S to the total number of blocks B_{total}. We define the success rate of the nodes which has generated no blocks as 0.

  $$s_{vi} = \frac{S}{B_{total}} \in [0, 1]$$

Given the blockchain, S_c score of any miner can be calculated at any point of time. Accordingly, each miner maintains its own copy of the S_c score of all the miners. Algorithm 2 shows how the S_c is calculated for a miner.

6.2. Sc Score based Grouping

Now that we have S_c calculated for each of the miner nodes, our goal is to select group of nodes N_{Sc} from the set of filtered nodes N_{PoD} so as to maximize S_c value of the group, subject to the restriction on the number of nodes. In other words, given a list of S_c values for each node V_i, we need to find the maximum possible sum of S_c such that the sum of S_c of selected nodes N_{Sc} will be greater than combining S_c values of not-selected nodes N'_{Sc} from this stage. We provide the pseudo-code for implementing the above-described methodology in algorithm 3.

In the first step, the target S_c value for the group to be selected is set by considering above 50% of the total S_c value. The group of nodes to be selected N_{Sc} would have higher S_c sum compared to the combined S_c sum value of unfiltered nodes N'_{Sc}.

In the second step, we are sorting the nodes based on S_c score first and then score manager unit identifies the set of nodes programmatically looping through the S_c values of the nodes and by checking the cumulative S_c sum value until target S_c sum set before is reached. However, for the very first round, as the S_c is the same for all the nodes, second filter criteria based on time of joining the network is considered. We analyse that complexity of the S_c filtering scheme is *O(nlogn)*. Finally, at the end of this stage, N_{Sc} a group of nodes performing PBFT is obtained and notified to broadcast unit, which disseminates the information to the rest of the network.

7. Defense against attacks on PoD and Sc mechanism

We recall here that how various attacks can be launched on the proposed model, as discussed in section 4.3. In this section, we focus on explaining how the proposed methods can defend itself from those known attacks. Much like PoW consensus, the PoD consensus mechanism as well implements a target hash-based mining mechanism. However, unlike in PoW, which gives the ability for the attacker to break the system by owning higher computation power, PoD does not give way for it. An attacker rather has to contribute to the network by hosting ride shares and gain driving balance in their wallet. We also would like to recall here that the Service score S_c of vehicular miner

nodes with honest behavior in the network, builds essentially on its continued participation and regular contribution to block creation in the entire blockchain system. Below we describe the details of the effect and defense of various attacks.

7.1. Security against Internal Attacks

Effect: Internal node is an authenticated member of the network, which, if turned malicious, severely impacts the safety in the proposed system. There could be an individual or multiple colluding malicious nodes working together to break into the system. In PoW based systems, if an internal malicious node owns the temporary majority of the computing power of the entire network, then it breaks the PoW based distributed systems.

Defense: In the proposed system model, we not only rely on computing power but also have incorporated randomization through proof of driving protocol. Consequently, for an internal adversary, there is no time guarantee when it gets selected as an intermediate miner, as discussed in theorem 1. Due to randomization, it introduces a dilemma for an adversary whether to earn more coins or to earn more service scores. Either of these would be unpredictable because the first stage of filtering fully depends on the average coin earnings of the network, which one cannot guess accurately at a given point of time. This also depends on the network as to how many nodes are active. Therefore, even if an insider adversary exists, his/her chance of becoming the mining node is no larger than other users.

7.2. Security against External Attack

Effect: Here, we consider a specific case of an external attack in one of the existing systems based on the PoS mechanism. If the number of coins owned by a single miner is more than 50% of the total blockchain network, an external adversary can directly try to compromise such a node. As subsequent rounds of consensus are solely dependant on the highest staked node in the network, controlling such nodes can cause arbitrarily manipulating and modifying the blockchain information.

Defense: In the proposed model, it is not the case due to randomization in the filtering process in the first stage. We show that the selection of nodes is diverse across various coin balances in Fig 3(b). The highest coins earned node is not always getting filtered, eliminating partial centralization as well

as the described attack scenario. Additionally, it is impractical for one vehicle node to earn more than 50% driving coins than the rest at a given window.

7.3. Security against Attacking the Leader Node

Effect: The proposed model adopts PBFT for consensus in which block creation is done by leader node once the miner group N_{sc} is finalized. It is highly possible that an external observer attacks the potential leader node in order to control the block creation process. This attack is similar to the attack on the highest staked node.

Defense: This attack is ineffective for the proposed model due to the active change of the leader nodes. After fixed rounds of consensus, the leader node will be changed. The leader for the subsequent consensus rounds is randomly selected based on produced proof of driving and service score protocols. Since the selection of the new leader depends only on this, no node knows the next leader in advance. Thus, the probability of a successful attack is greatly reduced.

7.4. PoD and Sc Attack Resilience

Here, we consider a passive approach where a miner node that intends to break the system in the future may behave honestly by contributing to the blockchain system creating good blocks for some period of time, which allows it to have a higher reputation. In the worst-case scenario, one of such passive attacker node might successfully get filtered through PoD mechanism and gain its spot in the final consensus group N_{sc} to create blocks in the system. If N_{PoD} is the number of nodes chosen from the first stage of filtering and Sc_i is their corresponding service score, our model ensures safety against such attacker node being a leader if (1) $\frac{2}{3}$rd nodes from the consensus group N_{Sc} are honest (2) the sum of service standard score of few attacker nodes Sc_a is less than the sum of service standard score of honest nodes. i.e., $Sc_a < \frac{1}{3}\sum_{i=1}^{N_{PoD}} Sc_i$. This clause holds good considering the fact that the honest mining node's service score would be much higher due to the complimenting weights for success s_{vi} and error rates e_{vi} in service score S_c calculation as per algorithm 2. In summary, although the attacker breaks into the system, the system can be lively and resilient if both the conditions mentioned above are true. Otherwise, block creation will be hampered.

8. Experimental Analysis

To conduct the experiments, we use Java SE 8 on a hardware platform with Windows 7 OS having the following specification: Intel(R) Core(TM) i7-9700 CPU@3.00 GHz, 8.00GB RAM. To study the effectiveness of our proposed protocol, we mainly focus on

1. How effectively PoD and S_c strategies can reduce the number of vehicle mining nodes from the pool of miners compared to traditional PBFT? Are the highest coins earner always get filtered?
2. What is the S_c value of group of selected nodes N_{Sc} versus not-selected nodes N'_{Sc}. Is there a considerable number of miner optimization?
3. Is the proposed mechanism PoD secure against excessively driving nodes?
4. Is the PoD and S_c mechanism secure against the infiltration of malicious nodes?
5. How scalable the proposed system is in terms of PoD and S_c protocols?

The rest of the experimental section is organized by answering above question with details analysis and discussion. To conduct our experiments, we collected data[2] from one of the traffic surveys that has recorded the distribution of vehicles across various time periods of the day in the interval of one hour with varying speeds across four different regions. Using this data reference, we simulate the vehicular mining node distribution for all four regions with the same speed pattern from 10:00 to 17:00 during a particular day of the month as captured by the survey. However, the time interval for the mining process is a configurable parameter tailored to the application's needs.

We recall here that in our model for every two miles of distance traveled by mining node, it earns one coin. Hence this data set is very suitable for our experiments as we can accurately derive the number of coins earned based on the speed per hour input.

8.1. Analyzing PoD based Filtering Process

To validate the feasibility of the proposed PoD selection strategy, we perform this experiment in several rounds during the different periods of the

[2]https://data.world/cityofaustin/et93-wr2y

day. We observe that from Fig 3(a), the number of nodes competing for mining is varying each time, reaching the peak at 13:00 with over 680 nodes as captured by the traffic survey data.

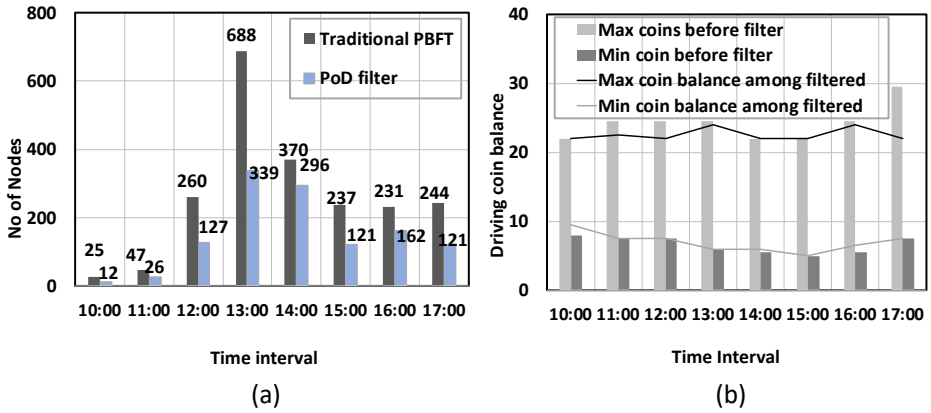

Figure 3: (a) Number of miner selected by Proof of Driving algorithm during different time intervals (b) Distribution of Coin Balance Before and After the Filter Process

We first simulate speed data referring to the data set for larger values of N, where N is the number of potential miner nodes from across different regions during the different time intervals of the day. Then this data is fed into the PoD to compute coin earnings of each vehicle node. Here we recall that coins are earned based on distance traveled. Finally, once the computation is complete, we display the number of filtered nodes.

At this first stage of filtering through PoD, we observe how our proposed algorithm only selects a subset of mining nodes based on the hash of coins earned for the next stage of filtering, and we represent the results in Fig 3(a).

It can be inferred that there is no significant difference in the filtering process when the mining pool is smaller during the early hours of the day. However, as the mining pool grows at around 13:00, which is as in the real world scenario, we see that the significant number of miner nodes are filtered and almost halved by implementing this strategy, which is an excellent pre-requisite to run PBFT with a fewer number of nodes.

We also calculate the highest and lowest coin earnings at each interval before and after filtering. Fig 3(b) represents the minimum and maximum driving coin balance of different groups of nodes before and after the filtering process. From this visualization, we can infer that, unlike PoS, which

only selects the highest staked nodes(here stakes are referred to driving coin balance), in our selection process, there is randomness. The PoD does not always filter the highest coins earner but filters a range of nodes with different coin balances.

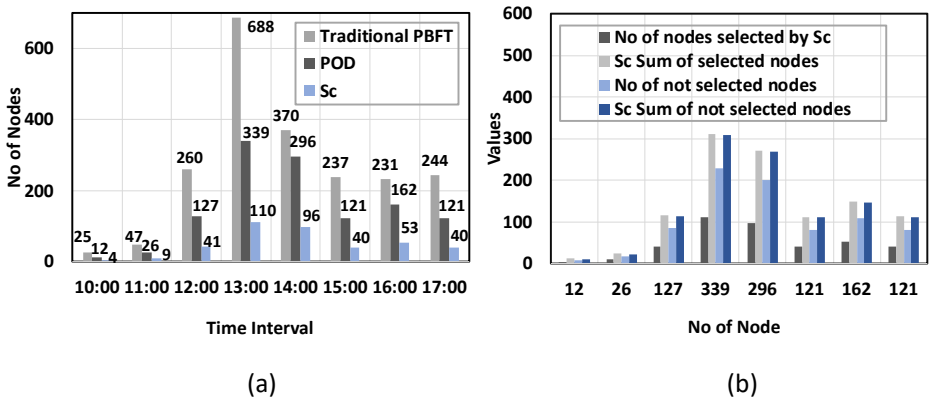

Figure 4: (a) Comparison of the Miner nodes selected by using PBFT alone versus using Proof of Driving and S_c filter (b) Visualization of number of selected nodes N_{Sc} vs not-selected nodes N'_{Sc}, after running S_c protocol

8.2. Analyzing S_c based Filtering

In this experiment, to validate the effectiveness of the S_c algorithm, we take the list of filtered nodes from PoD with simulated S_c values ranging from 0 to 2. We trace how these nodes are getting further filtered based on our proposed algorithm 3. This experiment is also performed in several rounds during different periods of the day. The results are represented graphically in Fig 4(a) by comparing the number of miners considered by traditional PBFT and our proposed filtering strategies. From Fig 4(a), we can infer that S_c based filtering strategy is providing a better reduction in the number of miners, based on group S_c value, which helps PBFT consensus to execute in a short period of time.

In order to visualize the optimum number nodes and their corresponding sum of S_c, we plot a bar graph, as shown in Fig 4(b). Here we can see that the protocol is selecting such groups containing an optimized number of nodes N_{Sc} that makes up the total S_c value more than the group of not-selected nodes N'_{Sc}. Also, it is worth noticing that the number of selected nodes N_{Sc} is always much lesser than the number of not-selected nodes N'_{Sc}

as we maximize the total S_c values with a minimum number of nodes to be selected.

8.3. Analyzing the Security Against Malicious Activities

In this subsection, we present an analysis of the security of our proposed method. Since the main criteria to be selected as miner node is to drive the vehicle honestly, to test its security, first, we experiment to see if any driver can win the selection process by intentionally driving more than the system average. Then we conduct another analysis against infiltration attacks of malicious nodes.

8.3.1. Security Against Intentionally Excessive Driving

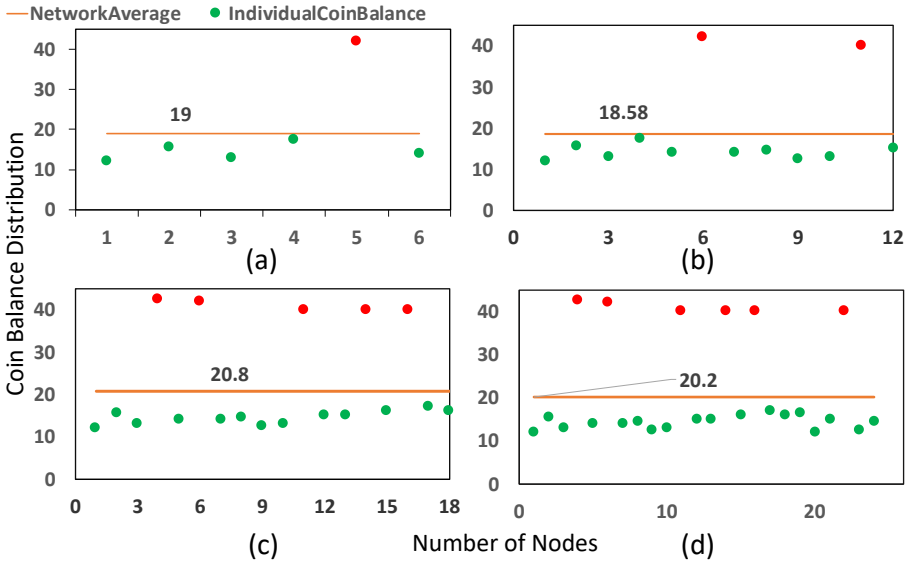

Figure 5: Analysing the effect of driving too much on PoD filtering mechanism. (a) Distribution of coin balance and network average with few abnormally driving nodes with 6 network nodes (b) with 12 nodes (c) with 18 nodes (d) with 24 nodes

In this experiment, we analyze the impact of malicious nodes that intentionally drive much distance in order to earn more coins in their driving coin wallet to infiltrate the final consensus group. The effects of this attack are also similar to the described attack in subsection 8.4, i.e., to break liveliness of the system. However, it differs from the way of nodes operation. Here we simulated few nodes that are driving abnormally higher distance, and we

increased the number of such nodes exponentially in every round, as depicted in Fig 5 (a)(b)(c)(d) by red markers. We infer from the experiments that such an attack from the nodes does not lead to successful filtration through the PoD mechanism. We recall here that the nodes are selected based on the overall network average hash. Thus the regularly driving nodes always suppress such attacks. E.g., the coin balance of the abnormal node in Fig 5 (d) is above 40, which is much far off from the actual average 20.2 of the entire network when there are a total of 24 nodes with 6 abnormal nodes. This makes the PoD mechanism secure against the nodes that intentionally drive a lot or corrupt their speedometer to show higher driving distance, thus filtering them out from infiltrating the final consensus group.

8.3.2. Analysis of PoD and Sc Against Infiltration of Malicious Nodes

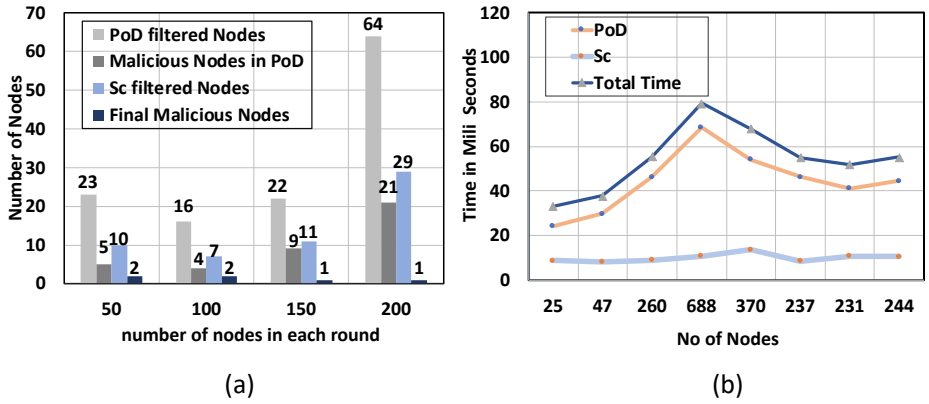

Figure 6: (a) Analysis of PoD and Sc mechanism's efficiency in filtering out malicious nodes (b) Latency for one round of PoD and S_c with a varying number of vehicular miners during different time intervals

In section 7.4, we have discussed that PoD combined with Sc is safe and live as long as the combined service score of attackers is below the service score of honest nodes; In this subsection, we experimentally prove the analysis by visualizing the outcomes in different consensus rounds. We have executed this experiment in four rounds with a varying number of total miner nodes that are taking part in the PoD mechanism. We have also simulated the presence of $\frac{1}{3}$ of malicious nodes in every round that are passive in nature and are actively taking part in ride-sharing service to gain a good reputation with the sole intentions of infiltrating the final consensus group. Figure 6(a)

represents the results, which shows that, although there are few of the malicious nodes that are getting filtered by PoD, the application of Sc filtering mechanism in the second stage is ensuring that the final consensus group N_{sc} is majorly containing nodes of higher service scores. Results also show a minute number of malicious nodes in the final consensus group represented by the dark blue bar in Fig 6(a). However, it is worth noting that, at this stage application of PBFT would ensure the system is fault-tolerant and can perform efficiently.

8.4. Analyzing the Time Consumption to Run PoD and S_c Filtering Process

We also measure the latency of our proposed schemes to select a certain number of nodes from a pool of nodes ranging at various periods of the day. In Fig 6(b), the curve in the middle represents the trend w.r.t time consumed by PoD, and the bottom-most curve represents S_c scheme to select nodes from a large pool of miner nodes. The topmost curve is the total time spent to run both the algorithms during one round of miner selection. The results show the relationship between the running time and the network scales, i.e., the number of miners, which is close to being linear as the network grows. It is evident that the time consumption is minimal; therefore, the protocol can be scaled to a larger number of nodes.

Although there is overhead time for using PoD and S_c filtering before PBFT consensus, our protocol reduced the mining group size by approximately 84 percent (without compromising on the quality of filtered nodes as shown in fig 4(b)) during peak time as shown in fig 4(a). We have set the time interval for the miner selection process to take place every one hour. During this time, a group of selected vehicular nodes will be taking turns in creating subsequent blocks. Therefore, the overhead time associated with the PoD and S_c filtering technique is applicable only once at the beginning of every interval. This total overhead time shown in Fig 6(b) is minimal (approx. 33ms & 79ms for 25 and 700 nodes respectively) compared to the total consensus time required in traditional PBFT without any filtering technique.

8.5. Remarks

In this paper, an application-specific consensus mechanism and a filtering strategy is developed. As this paper is addressed primarily towards measuring the efficiency of new methods and strategies, our experiments are mainly focused on finding out whether the proposed methods deliver the required features such as fairness, randomness, and quality of mining nodes in a public

network of vehicles. Based on the results, we infer that the methods are efficient to be applied alongside PBFT consensus. Hence the implementation on a specific blockchain platform such as Ethereum or Hyperledger fabric are not included in the scope of our experiments. Another critical area of discussion is storage in the vehicular node, which is limited. Due to resource constraints, vehicles cannot support the storage of a massive number of blocks. In order to support massive data storage and sharing close to the vehicles Vehicular networks integrated with mobile edge computing are evolving toward vehicular edge computing and networks(VECONs) which can be utilized to store historical data [49]

9. Conclusion

We proposed an efficient and effective miner node selection strategy in the application of vehicular blockchain. More specifically, we proposed Proof of Driving protocol introducing the driving coins associating it with one of the vehicle features such as distance traveled to achieve more randomness in selecting the miner nodes. By this strategy, we proved that not only the high staked nodes are selected as miners, but a range of nodes based on coin earnings are chosen as potential consensus nodes. Additionally, we also proposed the service score based protocol to ensure that the selected nodes are of high reputation, low error rate and high success rate w.r.t successful block mining and ensured that quality of mining node is not compromised while aiming to achieve randomness. We tested the proposed method extensively using realtime captured vehicle data. The experimental results show that the proposed method is effective and efficient in selecting the miner nodes with good reputation score, and more importantly, secure against various infiltration attacks.

References

[1] D. Dorrell, A. Vinel, D. Cao, Connected vehicles-advancements in vehicular technologies and informatics, IEEE Transactions on Industrial Electronics 62 (12) (2015) 7824–7826.

[2] D. Zhang, Q.-L. Han, X.-M. Zhang, Network-based modeling and proportional–integral control for direct-drive-wheel systems in wireless network environments, IEEE Transactions on Cybernetics 50 (6) (2020) 2462–2474.

[3] S. Chen, J. Hu, Y. Shi, Y. Peng, J. Fang, R. Zhao, L. Zhao, Vehicle-to-everything (v2x) services supported by lte-based systems and 5g, IEEE Communications Standards Magazine 1 (2) (2017) 70–76.

[4] H. Boeglen, B. Hilt, P. Lorenz, J. Ledy, A.-M. Poussard, R. Vauzelle, A survey of v2v channel modeling for vanet simulations, in: Proc. of the 8th IEEE International Conference on Wireless On-Demand Network Systems and Services, Bardonecchia, Italy, 2011, pp. 117–123.

[5] R. Doolan, G.-M. Muntean, Vanet-enabled eco-friendly road characteristics-aware routing for vehicular traffic, in: Proc. of the 77th IEEE Vehicular Technology Conference (VTC Spring), Dresden, Germany, 2013, pp. 1–5.

[6] N. Liu, M. Liu, J. Cao, G. Chen, W. Lou, When transportation meets communication: V2p over vanets, in: Proc. of the 30th IEEE International Conference on Distributed Computing Systems, Genova, Italy, 2010, pp. 567–576.

[7] E. Mousavinejad, F. Yang, Q.-L. Han, X. Ge, L. Vlacic, Distributed cyber attacks detection and recovery mechanism for vehicle platooning, IEEE Transactions on Intelligent Transportation Systems (2019). doi:10.1109/TITS.2019.2934481.

[8] X. Ge, Q.-L. Han, X.-M. Zhang, L. Ding, F. Yang, Distributed event-triggered estimation over sensor networks: A survey, IEEE Transactions on Cybernetics 50 (3) (2020) 1306–1320.

[9] X. Ge, Q.-L. Han, M. Zhong, X.-M. Zhang, Distributed krein space-based attack detection over sensor networks under deception attacks, Automatica 109 (2019) 108557.

[10] J. Ni, A. Zhang, X. Lin, X. S. Shen, Security, privacy, and fairness in fog-based vehicular crowdsensing, IEEE Communications Magazine 55 (6) (2017) 146–152.

[11] Y. Kim, I. Kim, Security issues in vehicular networks, in: Proc. of the IEEE International Conference on Information Networking, Bangkok, Thailand, 2013, pp. 468–472.

[12] R. Lu, X. Lin, X. Shen, Spring: A social-based privacy-preserving packet forwarding protocol for vehicular delay tolerant networks, in: Proc. of the IEEE INFOCOM, San Diego, CA, USA, 2010, pp. 1–9.

[13] T. W. Chim, S.-M. Yiu, L. C. Hui, V. O. Li, Specs: Secure and privacy enhancing communications schemes for vanets, Ad Hoc Networks 9 (2) (2011) 189–203.

[14] Y. Sun, R. Lu, X. Lin, X. Shen, J. Su, An efficient pseudonymous authentication scheme with strong privacy preservation for vehicular communications, IEEE Transactions on Vehicular Technology 59 (7) (2010) 3589–3603.

[15] P. K. Sharma, S. Y. Moon, J. H. Park, Block-vn: A distributed blockchain based vehicular network architecture in smart city., JIPS 13 (1) (2017) 184–195.

[16] R. Shrestha, R. Bajracharya, A. P. Shrestha, S. Y. Nam, A new-type of blockchain for secure message exchange in vanet, Digital Communications and Networks (2019). doi:10.1016/j.dcan.2019.04.003.

[17] S. Nakamoto, Bitcoin: A peer-to-peer electronic cash system," http://bitcoin.org/bitcoin.pdf (2008).

[18] X. Li, P. Jiang, T. Chen, X. Luo, Q. Wen, A survey on the security of blockchain systems, Future Generation Computer Systems 107 (2020) 841 – 853.

[19] G. Zyskind, O. Nathan, et al., Decentralizing privacy: Using blockchain to protect personal data, in: Proc. of the IEEE Security and Privacy Workshops, San Jose, CA, USA, 2015, pp. 180–184.

[20] S. A. Abeyratne, R. P. Monfared, Blockchain ready manufacturing supply chain using distributed ledger, International Journal of Research in Engineering and Technology 5 (9) (2016) 1–10.

[21] I. Vakilinia, S. Badsha, S. Sengupta, Crowdfunding the insurance of a cyber-product using blockchain, in: Proc. of the 9th IEEE Annual Ubiquitous Computing, Electronics & Mobile Communication Conference, New York City, NY, USA, 2018, pp. 964–970.

[22] I. Vakilinia, S. Badsha, E. Arslan, S. Sengupta, Pooling approach for task allocation in the blockchain based decentralized storage network, in: Proc. of the 15th IEEE International Conference on Network and Service Management, Halifax, NS, Canada, 2019, pp. 1–6.

[23] S. Badsha, I. Vakilinia, S. Sengupta, Blocynfo-share: Blockchain based cybersecurity information sharing with fine grained access control, in: Proc. of the 10th IEEE Annual Computing and Communication Workshop and Conference, Las Vegas, NV, USA, 2020, pp. 0317–0323.

[24] A. Dorri, S. S. Kanhere, R. Jurdak, P. Gauravaram, Blockchain for iot security and privacy: The case study of a smart home, in: Proc. of the IEEE international conference on pervasive computing and communications workshops, Kona, HI, USA, 2017, pp. 618–623.

[25] N. J. Witchey, Healthcare transaction validation via blockchain, systems and methods, uS Patent 10,340,038 (Jul. 2 2019).

[26] Z. Yang, K. Yang, L. Lei, K. Zheng, V. C. Leung, Blockchain-based decentralized trust management in vehicular networks, IEEE Internet of Things Journal 6 (2) (2018) 1495–1505.

[27] Y. Yao, X. Chang, J. Mišić, V. B. Mišić, L. Li, Bla: Blockchain-assisted lightweight anonymous authentication for distributed vehicular fog services, IEEE Internet of Things Journal 6 (2) (2019) 3775–3784.

[28] P.-Y. Chen, J.-W. Liu, W.-T. Chen, A fuel-saving and pollution-reducing dynamic taxi-sharing protocol in vanets, in: Proc. of the 72nd IEEE Vehicular Technology Conference-Fall, Ottawa, ON, Canada, 2010, pp. 1–5.

[29] W. Zhao, Y. Qin, D. Yang, L. Zhang, W. Zhu, Social group architecture based distributed ride-sharing service in vanet, International Journal of Distributed Sensor Networks 10 (3) (2014) 650923.

[30] M. Baza, N. Lasla, M. Mahmoud, G. Srivastava, M. Abdallah, B-ride: Ride sharing with privacy-preservation, trust and fair payment atop public blockchain, IEEE Transactions on Network Science and Engineering (2019). doi:10.1109/TNSE.2019.2959230.

[31] L. Li, J. Liu, L. Cheng, S. Qiu, W. Wang, X. Zhang, Z. Zhang, Creditcoin: A privacy-preserving blockchain-based incentive announcement network for communications of smart vehicles, IEEE Transactions on Intelligent Transportation Systems 19 (7) (2018) 2204–2220.

[32] K. J. O'Dwyer, D. Malone, Bitcoin mining and its energy footprint, in: Proc. of the 25th IET Irish Signals Systems Conference 2014 and 2014 China-Ireland International Conference on Information and Communications Technologies (ISSC 2014/CIICT 2014), Limerick, Ireland, 2014, pp. 280–285.

[33] S. Kudva, R. Norderhaug, S. Badsha, S. Sengupta, A. Kayes, Pebers: Practical ethereum blockchain based efficient ride hailing service, in: Proc. of the IEEE International Conference on Informatics, IoT and Enabling Technologies, Doha, Qatar, 2020, pp. 422–428.

[34] M. Castro, B. Liskov, Practical byzantine fault tolerance, in: Proc. of the Third Symposium on Operating Systems Design and Implementation, USENIX Association, New Orleans, Louisiana, USA, 1999, p. 173–186.

[35] S. Underwood, Blockchain beyond bitcoin, Communications of the ACM 59 (11) (2016) 15–17.

[36] Z. Zheng, S. Xie, H. Dai, X. Chen, H. Wang, An overview of blockchain technology: Architecture, consensus, and future trends, in: Proc. of the IEEE International Congress on Big Data, Honolulu, HI, USA, 2017, pp. 557–564.

[37] S. King, S. Nadal, Ppcoin: Peer-to-peer crypto-currency with proof-of-stake, self-published paper, August 19 (2012).

[38] M. Singh, S. Kim, Blockchain based intelligent vehicle data sharing framework (2017). arXiv:1708.09721.

[39] Z. Yang, K. Zheng, K. Yang, V. C. Leung, A blockchain-based reputation system for data credibility assessment in vehicular networks, in: Proc. of the 28th IEEE annual international symposium on personal, indoor, and mobile radio communications, Montreal, QC, Canada, 2017, pp. 1–5.

[40] Y. Yuan, F.-Y. Wang, Towards blockchain-based intelligent transportation systems, in: Proc. of the IEEE 19th International Conference on Intelligent Transportation Systems, Rio de Janeiro, Brazil, 2016, pp. 2663–2668.

[41] P. K. Sharma, N. Kumar, J. H. Park, Blockchain-based distributed framework for automotive industry in a smart city, IEEE Transactions on Industrial Informatics 15 (7) (2019) 4197–4205.

[42] J. Kang, Z. Xiong, D. Niyato, D. Ye, D. I. Kim, J. Zhao, Toward secure blockchain-enabled internet of vehicles: Optimizing consensus management using reputation and contract theory, IEEE Transactions on Vehicular Technology 68 (3) (2019) 2906–2920.

[43] K. Peterson, R. Deeduvanu, P. Kanjamala, K. Boles, A blockchain-based approach to health information exchange networks, in: Proc. of the NIST Workshop Blockchain Healthcare, Vol. 1, 2016, pp. 1–10.

[44] J. Yu, D. Kozhaya, J. Decouchant, P. Esteves-Verissimo, Repucoin: Your reputation is your power, IEEE Transactions on Computers 68 (8) (2019) 1225–1237.

[45] S. Wang, X. Huang, R. Yu, Y. Zhang, E. Hossain, Permissioned blockchain for efficient and secure resource sharing in vehicular edge computing (2019). arXiv:1906.06319.

[46] Y. Gilad, R. Hemo, S. Micali, G. Vlachos, N. Zeldovich, Algorand: Scaling byzantine agreements for cryptocurrencies, in: Proc. of the 26th ACM Symposium on Operating Systems Principles, Shanghai, China, 2017, pp. 51–68.

[47] L. Zhu, Y. Wu, K. Gai, K.-K. R. Choo, Controllable and trustworthy blockchain-based cloud data management, Future Generation Computer Systems 91 (2019) 527–535.

[48] M. Castro, B. Liskov, Practical byzantine fault tolerance and proactive recovery, ACM Transactions on Computer Systems (TOCS) 20 (4) (2002) 398–461.

[49] X. Huang, R. Yu, J. Kang, Y. Zhang, Distributed reputation management for secure and efficient vehicular edge computing and networks, IEEE Access 5 (2017) 25408–25420.