Towards Secure Blockchain-enabled Internet of Vehicles: Optimizing Consensus Management Using Reputation and Contract Theory

Jiawen Kang, Zehui Xiong, Dusit Niyato, Fellow, IEEE, Dongdong Ye, Dong In Kim, Fellow, IEEE, Jun Zhao, Member, IEEE

Abstract—In the Internet of Vehicles (IoV), data sharing among vehicles is critical to improving driving safety and enhancing vehicular services. To ensure security and traceability of data sharing, existing studies utilize efficient Delegated Proof-of-Stake consensus scheme as hard security solutions to establish blockchain-enabled IoV (BIoV). However, as the miners are selected from the miner candidates by stake-based voting, defending against voting collusion between the candidates and compromised high-stake vehicles becomes challenging. To address the challenge, in this paper, we propose a two-stage soft security enhancement solution: (i) miner selection and (ii) block verification. In the first stage, we design a reputation-based voting scheme to ensure secure miner selection. This scheme evaluates candidates’ reputation using both past interactions and recommended opinions from other vehicles. The candidates with high reputation are selected to be active miners and standby miners. In the second stage, to prevent internal collusion among active miners, a newly generated block is further verified and audited by standby miners. To incentivize the participation of the standby miners in block verification, we adopt the contract theory to model the interactions between active miners and standby miners, where block verification security and delay are taken into consideration. Numerical results based on a real-world dataset confirm the security and efficiency of our schemes for data sharing in BiOv.

Index Terms—Internet of Vehicles, blockchain, reputation management, delegated proof-of-stake, contract theory, security

I. INTRODUCTION

A. Background and Motivations

Copyright (c) 2019 IEEE. Personal use of this material is permitted. However, permission to use this material for any other purposes must be obtained from the IEEE by sending a request to pubs-permissions@ieee.org.

This work was supported in part by WASP/NTU M4082187 (4080), Singapore MOE Tier 1 under Grant 2017-T1-002-007 RG122/17, MOE Tier 2 under Grant MOE2014-T2-2-015 ARC4/15, NRF2015-NRF-IFSP001-2277, and EMA Energy Resilience under Grant NRF2017EW4T-EP03-041, and the National Research Foundation of Korea (NRF) Grant funded by the Korean Government under Grant 2017R1A2B2003953 and 2014R1A1A1011478.

Jiawen Kang, Zehui Xiong, Dusit Niyato, and Jun Zhao are with School of Computer Science and Engineering, Nanyang Technological University, Singapore. (Emails: kvinkang@ntu.edu.sg, zxiong002@e.ntu.edu.sg, d-niyato@ntu.edu.sg, junzha@ntu.edu.sg). Dongdong Ye is with School of Automation, Guangdong University of Technology, China. (Email: d-donngye@163.com). Dong In Kim is with School of Information & Communication Engineering, Sungkyunkwan University, Korea. (Email: dikim@skku.ac.kr). Corresponding author: Dong In Kim.

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org

Digital Object Identifier xx.xxxx/TVT.xxxx.xxxxxx

WITH the rapid development of the automobile industry and Internet of Things (IoT), vehicles generate a huge amount and diverse types of data through advanced on-board devices. Vehicles collect and share data to improve driving safety and achieve better intelligent transportation system service quality [1]. However, there exist significant security and privacy challenges for data sharing in IoV. On the one hand, vehicles may not be willing to upload data to infrastructures, e.g., through roadside units, with a centralized management architecture because of the concern on a single point of failure and personal data manipulation. On the other hand, although Peer-to-Peer (P2P) data sharing among the vehicles can solve the issues of the centralized management architecture, it is still facing with the problems of data access without authorization and security protection in a decentralized architecture. These challenges adversely affect the circulation of vehicle data, even forming data ‘island’, and thus hinder the future development of IoV [2].

Recently, integrating blockchain technology with IoV has attracted increasing attention of researchers and developers because of decentralization, anonymity, and trust characteristics of blockchain. A secure, trusted, and decentralized intelligent transport ecosystem is established by blockchain to solve vehicle data sharing problems [2], [3]. Yang et al. [1] proposed a decentralized trust management system for vehicle data credibility assessment using blockchain with joint Proof-of-Work (PoW) and Proof-of-Stake (PoS) consensus schemes. Car manufacturers Volkswagen1 and Ford2 have applied for patents that enable secure inter-vehicle communication through blockchain technologies. An intelligent vehicle-trust point mechanism using proof-of-driving-based blockchain is presented to support secure communications and data sharing among vehicles [4]. Li et al. [5] proposed a privacy-preserving incentive announcement network based on public blockchain. The Byzantine fault tolerance algorithm is adopted to incentivize vehicles to share traffic information. Nevertheless, the cost is exorbitant to establish a blockchain in resource-limited vehicles using computation-intensive PoW or unfair stake-based PoS [6]. Existing research attempts cannot neatly address the P2P data sharing problem among vehicles in IoV.

In this paper, we utilize high-efficiency Delegated Proof-of-

1https://www.ccn.com/volkswagen-seeks-patent-for-inter-vehicular-blockchain-communications-system/
2https://news.bitcoin.com/ford-cryptocurrency-inter-vehicle-communication-system/
active miners can be further verified and audited by standby tamper-proof reputation records in transparent blockchain for verification. A reputation management scheme established on blockchain technologies is proposed for the miner selection.

Solution for Enhancing DPoS Schemes through a Two-stage Stake-based Voting Scheme

BIoV suffers from the following collusion attacks in BIoV:

1. **Miner Voting Collusion**: Malicious RSUs collude with compromised high-stake stakeholders to be voted as miners. These malicious miners may falsify or discard transaction data during its mining process. Although the malicious miners can be voted out of the BIoV by the majority of well-behaved stakeholders in the next voting round, the stakeholders may not participate in all the voting rounds. Thus, some malicious miners cannot be removed in a timely fashion, which enables the malicious miners to launch attacks to damage the system continuously.

2. **Block Verification Collusion**: Malicious miners may internally collude with other miners to generate false results in the block verification stage, even to launch double-spending attack, which is also challenging.

Therefore, it is necessary to design an enhanced DPoS consensus scheme with secure miner selection and block verification to defend against the collusion attacks in BIoV.

### B. Solutions and Contributions

Reputation is defined as the rating of an entity’s trustworthiness by others based on its past behaviors. We utilize reputation as a fair metric to propose a soft security solution for enhancing DPoS schemes through a two-stage mechanism: (i) secure miner selection, and (ii) reliable block verification. A reputation management scheme established on blockchain technologies is proposed for the miner selection.

Miner candidates with high reputation are selected to form a miner group including active miners and standby miners, e.g., 21 active miners and 150 standby miners in Enterprise Operation System (EOS) [15]. Each vehicle has its reputation opinion on an interacting miner candidate through a subjective logic model that combines recommended opinions from other vehicles and its own opinions based on past interactions into an accurate reputation opinion [16]. All the reputation opinions of vehicles on the candidates are recorded as reliable and tamper-proof reputation records in transparent blockchain for reputation calculation.

Moreover, for secure block verification, blocks generated by active miners can be further verified and audited by standby miners to prevent internal collusion among active miners [17]. Here, the active miners take turn to act as the block manager to generate and distribute unverified blocks. To incentivize the standby miners to participate in the block verification, we utilize contract theory to model interactions among the block manager and miners to prevent collusion attacks. The block manager works as the contract designer. Meanwhile, the miners including active miners and standby miners are the followers to finish block verification for obtaining a part of transaction fee according to verification contribution [17].

The main contributions of this paper are summarized as follows.

- We propose an enhanced DPoS consensus scheme with two-stage soft security solution for secure vehicle data sharing in BIoV.
- In the miner selection stage, we introduce a secure and efficient reputation management scheme by using a multi-weight subjective logic model. Miners are selected by reputation-based voting for decreasing collusion between stakeholders with a lot of stake and miner candidates.
- In the block verification stage, high-reputation standby miners are incentivized to participate in block verification using contract theory for preventing internal collusion among active miners.

The rest of this paper is organized as follows. We present the system model and the enhanced DPoS consensus scheme with detailed steps for secure P2P vehicle data sharing in Section II. We illustrate the secure reputation management scheme by using the multi-weight subjective logic model in Section III. The incentive mechanism for secure block verification using contract theory is proposed in Section IV, followed by optimal contract designing in Section V. We illustrate numerical results in Section VI. Section VII concludes the paper.

### II. Related Work

#### A. Consensus Mechanisms for Blockchain-enabled Vehicular Networks

Recent studies have utilized different consensus mechanisms to establish blockchain-enabled vehicular networks for secure data sharing. Kang et al. [18] proposed a secure vehicle data sharing system in vehicular edge computing and networks, in which PoW consensus scheme is used for data sharing record auditing and verification. An incentive scheme named proof-of-stage is also presented to encourage edge nodes to contribute storage resource. Kchaou et al. [19] proposed a distributed trust management scheme developed based on blockchain technologies to estimate the credibility of exchanged messages in vehicular networks. RSUs working as miners perform PoW consensus schemes in this system. Similarly, Yang et al. [1] also proposed a decentralized trust management system for credibility assessment of received vehicle data by using blockchain with joint PoW and PoS consensus schemes. In [20], the authors proposed a blockchain-enabled anonymous reputation system using PoW for trust management in vehicular networks. This system can prevent the distribution of forged broadcasting messages and protect identity privacy of vehicles. However, all these schemes are
computation-intensive because each miner needs to deliver a hash querying rate as high as possible to win the puzzle solving race. There exists an exorbitant cost to establish blockchain-enabled vehicular networks by using computation-intensive consensus schemes for resource-limited vehicles [6].

To decrease resource cost of consensus schemes, a few studies have replaced the computation-intensive consensus schemes by “virtual mining” [6]. Singh et al. [4] presented an intelligent vehicle-trust point mechanism using proof-of-driving-based blockchain to support secure communications and data sharing among vehicles. Li et al. [5] designed a blockchain-based privacy-preserving incentive announcement network by using the Byzantine fault tolerance consensus algorithm. This incentive announcement network encourages vehicles to forward and receive reliable information by using reputation points. Nevertheless, these consensus schemes suffer from limited scalability and efficiency for vehicular networks. Yuan et al. [3] indicated that DPoS is particularly suitable for vehicles to establish blockchain-based vehicular networks in ITS ecosystems. However, there exist security challenges, such as stakeholder voting collusion, in traditional DPoS consensus schemes. Therefore, in this paper, we present an enhanced DPoS consensus scheme to defend against the voting collusion for secure blockchain-enabled vehicular networks.

B. Incentive Schemes for Secure Block Verification

In blockchain, miners may launch miner collusion attacks, i.e., majority attacks, for maximizing their profit. The attacks may result in reversing transactions, preventing new transactions from confirmation, and even double-spending cryptocurrency. To defend against the attacks, Kim et al. [21] presented a blockchain governance game to keep the blockchain network decentralized for private blockchains. In this model, the miners intending to fork the blockchain are the attackers, while the miners honestly performing mining tasks act as defenders. The interaction between the attackers and the defenders is modeled as a stochastic game. The majority attack can also happen in the consortium blockchain using PoS consensus scheme [17]. The blockchain users generate transactions and upload to miners for verification. Due to the limited number of the miners, malicious miners can launch the majority attack to destroy the consortium blockchain system. To prevent this majority attack, the miners are encouraged to propagate unverified blocks to more verifiers to avoid centralized block verification. The more verifiers participating in block verification leads to a more secure consortium blockchain, but also results in larger verification delay and higher cost (i.e., transaction fee). A Stackelberg game is formulated to trade off security requirements, verification delay and cost, which jointly maximizes the utility of the blockchain user and the miners in the scenario with secure block verification [17]. However, the game model is under the assumption of complete information of the all miners’ strategies, which may not be practical for all the blockchain networks. For DPoS-based blockchain, incentive schemes for secure block verification are ignored and are studied in this paper. Furthermore, to the best of our knowledge, our work is the first work to combine contract theory into block verification for blockchain networks, leading desirable economic and resource allocation properties.

III. SYSTEM MODEL AND THE ENHANCED DPoS CONSENSUS SCHEME

A. System Model

As shown in Fig. 1, vehicles equipped with on-board units and advanced communication devices can access vehicular services by communicating with nearby RSUs in BIoV. The on-board units can perform simple computation, collect local data from sensing devices, and upload the data to the RSUs [22]. Vehicles act as data providers and share their own data with data requesters through wireless communication. Next, the vehicles upload their data sharing records as “transactions” to nearby RSUs. RSUs are deployed along roads to ensure that the vehicles are able to communicate with other vehicles and miners in a timely fashion [1], [8], [9]. Unlike traditional PoS schemes that miners are selected by stake-based voting, RSUs with high reputation are selected as miners, whose reputation values are calculated by a multi-weight subjective logic model. More details about the model are given in Section III. The data providers share data with each other and obtain a reward from data requesters. Next, the data providers upload data sharing records to active miners, and the miners execute the consensus process of our enhanced DPoS consensus scheme. Finally, the vehicle’s data sharing records are stored as block data and added into a blockchain, named vehicular blockchain, for achieving efficient proof-of-presence of the data sharing.

The vehicular blockchain is also a public ledger that records vehicles’ reputation opinions for RSUs and miners into the block data. These reputation opinions are persistent and transparent evidence when disputes and destruction occur [20].
Fig. 2: The enhanced DPoS consensus scheme for blockchain-enabled IoV. The steps of reputation calculation include: B-① Generate local reputation opinion on RSU i. B-② Download reputation opinions of other vehicles on RSU i from the vehicular blockchain. B-③ Combine local opinions with recommended opinions to obtain the final reputation opinion on RSU i. B-④ Calculate the average value of final reputation opinions on RSU i and compare with a threshold of trust. B-⑤ High-reputation RSU i becomes a miner candidate.

Vehicles assess both RSUs during vehicular services and active miners in the consensus process. The vehicles also download the existing reputation opinions about these entities in vehicular blockchain as recommended opinions. Then, the vehicles generate their reputation opinions through combining their own assessments with the recommended opinions, and upload these new opinions with digital signatures to new active miners through nearby RSUs [1]. The miners perform the consensus process similar to that in data sharing. All the vehicles can obtain the latest RSUs’ reputation after the reputation opinions being added into the vehicular blockchain. The system can calculate the average reputation of RSUs according to the reputation opinions in the vehicular blockchain, which is an important metric for the miner selection in the next round of the consensus process [20].

B. Adversary Model for DPoS Consensus Process

In traditional DPoS consensus schemes, miners are selected from miner candidates according to stake-based voting among stakeholders, i.e., vehicles with stake. In BIoV, RSUs acting as miner candidates may be distributed along the road without sufficient security protection, they are semi-trusted and may be vulnerable to be directly compromised by attackers [1], [23]. Both stakeholders and miner candidates are vulnerable to arbitrary manipulation by plutocrats [12], and become compromised stakeholders and malicious miner candidates. The plutocrats, i.e., attackers, can launch voting collusion that compromises some high-stake stakeholders with greater voting power, and ask the compromised stakeholders to vote for certain miner candidates. Moreover, compromised vehicles in BIoV can generate and upload fake reputation opinions to an RSU to increase or decrease the reputation of the target RSU [1]. Due to the overwhelming cost, we consider that the attackers cannot compromise the majority of vehicles [20]. Only a small subset of vehicles can be compromised during a short period of time in BIoV [1], i.e., due to the high mobility of vehicles.

C. The Enhanced DPoS Scheme for Blockchain-enabled IoV

As depicted in Fig. 2, there are mainly three parts in the enhanced DPoS consensus scheme for secure P2P vehicle data sharing: (i) updating block data (data sharing records and reputation opinions from vehicles) and miner candidates joining, (ii) reputation-based voting for miner selection and (iii) secure block verification using contract theory. More details about the steps of the proposed parts are given in the subsequent discussions.

**Step 1: System Initialization:** In vehicular blockchain, elliptic curve digital signature algorithm and asymmetric cryptography are adopted for system initialization. Every entity becomes legitimate after passing identity authentication by a global Trust Authority (TA), e.g., a government department.
of transportation. Each legitimate entity obtains its public & private keys and the corresponding certificates for information encryption and decryption [7]. An RSU that wants to be a miner candidate first submits its identity-related information to the TA. As shown in Fig. 2, the TA verifies the validity of the RSU by calculating its average reputation according to stored reputation opinions from vehicles in the vehicular blockchain. Only if the average reputation of this RSU is higher than a threshold of trust, the RSU can become a miner candidate. The threshold can be set according to different security-level requirements [14], which is explained in Section VI-B.

**Step 2:** Miner candidate joining: Each miner candidate submits a deposit of stake to an account under public supervision after being a miner candidate. This deposit will be confiscated by the vehicular blockchain system if the candidate behaves maliciously and causes damage during the consensus process, e.g., failing to produce a block in its time slot [15], [24].

**Step 3:** Reputation calculation: As shown in Fig. 2, stakeholders can calculate all miner candidates’ reputation by using a subjective logic model, which is based on past interactions with the miner candidates and recommended opinions from other vehicles. The subjective logic model takes three weights about the past interactions into consideration to form the local opinion on each miner candidate. The latest recommended opinions can be downloaded from the vehicular blockchain. Thus each stakeholder combines its local opinion with the recommended opinions to obtain a final reputation opinion on every miner candidate. More details about the reputation calculation are presented in Section III.

**Step 4:** Miner selection: According to the final reputation opinions calculated by Step 3, as shown in Fig. 2, each stakeholder votes for y candidates as the miners according to its ranking of the final reputation opinions for the candidates. Unlike traditional DPOS schemes, all the stakeholders have the same weight in miner voting (same voting power) even though some stakeholders owning larger stake. The top k miner candidates with the highest reputation are selected to be active miners and (y−k) miner candidates can be standby miners. The active miners and standby miners form a miner group in vehicular blockchain. Here y > k, and k is an odd integer, such as 21 in EoS and 101 in Bitshares [15].

**Step 5:** Block manager generation: In line with traditional DPOS schemes, each of the k active miners takes turn to act as the block manager during k time slots of the consensus process. Every active miner plays the role of the block manager in its time slot. The block manager is responsible for two kinds of tasks: (i) working as a miner manager to perform block generation, broadcasting, verification and block management during the consensus process, and (ii) acting as a contract designer to broadcast contract items to miners in the incentive mechanism.

**Step 6:** Consensus process: As shown in Fig. 2, in a time slot, the block manager first generates an unverified block, and broadcasts this block to other active miners in a centralized manner for block verification. However, due to the limited number of active miners, malicious active miners may launch the block verification collusion attack to generate false block verification results. In the block verification stage, the more verifiers result in a more secure blockchain network [17]. Therefore, to defend this attack and further enhance security performance of the proposed DPOS consensus scheme, more verifiers are motivated and incentivized to participate in the block verification instead of only active miners finishing the verification. In other words, the miners including active miners and standby miners can act as verifiers and join the block verification process, especially the high-reputation miners, which can effectively prevent the block verification collusion among the active miners. As such, we design an incentive mechanism by using contract theory to encourage high-reputation miners to participate in the block verification. In the incentive mechanism, the active miner acts as the block manager and the contract designer to broadcast contract items to miners. Meanwhile, the miners choose and sign their best contract items. More details about the block verification using contract theory are described in Section IV.

In Fig. 2, for mutual supervision and verification, high-reputation miners locally audit the data block and broadcast their audit results with their signatures to each other in a distributed manner. After receiving the audit results, each miner compares its result with those of other miners and sends a reply as a feedback to the block manager in each time slot. This reply consists of the miner’s audit result, comparison result, signatures, and records of received audit results. The block manager analyzes the received replies from the miners. If more than two thirds of the miners agree on the data block, the block manager will send the records including the current audited data block and the corresponding signature to all miners for storage after checking. Next, this block is formally stored in the vehicular blockchain. The block manager is rewarded with cryptocurrency, and the other miners participating in block verification will receive a part of the transaction fee. After k time slots, the group of miners and their categories, i.e., active or standby miners, will be updated and shuffled through a new miner selection round.

**Step 7:** Reputation updating: After each round of the consensus process, vehicles download and check a new data block related to their data sharing records or reputation opinions in the vehicular blockchain. If the data is correct, the vehicles will update their reputation opinions for these miners and forward their opinions to new miners of the next round of consensus process. The miners perform consensus process in Step 6 to add valid reputation values into the vehicular blockchain.

The proposed scheme can address the issues of the compromised block manager through the following security analysis:

(i) Reputation is utilized to select active miners that is a time-accumulated metric to indicate trustworthiness of entities according to their past behaviors. The block manager is chosen from active miners with high-reputation. The reputation value is accurately and reliably calculated by the multi-weight subjective logic model based on direct and recommended reputation opinions from a large number of vehicles. Therefore, the block managers are basically trustworthy that have a small
IV. EFFICIENT REPUTATION CALCULATION USING SUBJECTIVE LOGIC MODEL

When a positive interaction between vehicles and RSUs/miners happens, the vehicles will generate a positive rating for the RSUs/miners. Consequently, the vehicle’s local reputation opinion on the RSUs/miners is increased. The positive interaction means that the vehicles believe that the services provided by RSUs are relevant and useful or the new data block generated by a miner is true. Note that the miner candidates with high reputation acting as miners can ensure a secure and reliable consensus process. On the contrary, some compromised vehicles may generate fake rating because of collusion with malicious RSUs. More false ratings cause more negative effects on the miner selection in the proposed DPoS scheme, thus resulting in unreliable and insecure BloV. Therefore, it is necessary to design a secure and efficient reputation management scheme of RSUs, and also to defend against the collusion between RSUs and vehicles. Vehicles choose their own best miner candidates as the miners according to reputation calculation [25]. A multi-weight subjective logic model for reputation calculation is proposed in this section.

Subjective logic is utilized to formulate individual evaluation of reputation based on past interactions and recommended opinions. It is a framework for probabilistic information fusion operated on subjective beliefs about the world. The subjective logic utilizes the term “opinion” to denote the representation of a subjective belief, and models positive, negative statements and uncertainty. It also offers a wide range of logical operators to combine and relate different opinions [14]. In this paper, each vehicle (stakeholder) calculates reputation opinion taking all the recommended opinions into consideration. Due to the limited number of compromised vehicles, the false recommended opinions from the compromised vehicles will have only marginal effect on reputation calculation using subjective logic model since most vehicles are well-behaved and reliable. The key notations used in this paper are listed in Table I.

### A. Local Opinions for Subjective Logic

Considering a vehicle $V_i$ and an RSU $RU_j$, the vehicle may interact with the RSU during driving, e.g., crowdsensing or vehicle data sharing. The trustworthiness (i.e., local opinion) of $V_i$ to $RU_j$ in the subjective logic can be formally described as a local opinion vector $\omega_{i\rightarrow j} := \{b_{i\rightarrow j}, d_{i\rightarrow j}, u_{i\rightarrow j}\}$, where $b_{i\rightarrow j}, d_{i\rightarrow j}$, and $u_{i\rightarrow j}$ represent the belief, distrust, and uncertainty, respectively. We consider that all vehicles have the same evaluation criteria to generate local opinions. Here, $b_{i\rightarrow j} + d_{i\rightarrow j} + u_{i\rightarrow j} = 1$. According to the subjective logic model [14], [16], we have

$$
\begin{align*}
    b_{i\rightarrow j} &= (1 - u_{i\rightarrow j}) \frac{\alpha_i}{\alpha_i + \beta_i}, \\
    d_{i\rightarrow j} &= (1 - u_{i\rightarrow j}) \frac{\beta_i}{\alpha_i + \beta_i}, \\
    u_{i\rightarrow j} &= 1 - b_{i\rightarrow j} - d_{i\rightarrow j}.
\end{align*}
$$

(1)

$\alpha_i$ is the number of positive interactions and $\beta_i$ is the number of negative interactions. The communication quality $s_{i\rightarrow j}$ of a link between the vehicle $i$ and the RSU $j$, i.e., the successful transmission probability of data packets, determines the uncertainty of local opinion vector $u_{i\rightarrow j}$ [14]. According to $\omega_{i\rightarrow j}$, the reputation value $T_{i\rightarrow j}$ represents the expected belief of vehicle $V_i$ that RSU $RU_j$ is trusted and behaves normally during a consensus process, which is denoted by

$$
T_{i\rightarrow j} = b_{i\rightarrow j} + \gamma u_{i\rightarrow j}.
$$

(2)

Here, $0 \leq \gamma \leq 1$ is the given constant indicating the effect level of the uncertainty for reputation [16].

### B. Multi-weight Local Opinions for Subjective Logic

Local opinions using the subjective logic model are affected by different factors. Traditional subjective logic is evolved towards multi-weight subjective logic when considering weighting operations. Similar to [14], we consider the following weights to formulate local opinions.

<table>
<thead>
<tr>
<th>Notation</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>${b_{i\rightarrow j}, d_{i\rightarrow j}, u_{i\rightarrow j}}$</td>
<td>A local opinion vector $\omega_{i\rightarrow j}$ of a vehicle $V_i$ to an RSU $RU_j$</td>
</tr>
<tr>
<td>$T_{i\rightarrow j}$</td>
<td>The expected belief of $V_i$ to $RU_j$</td>
</tr>
<tr>
<td>$\alpha_i, \beta_i$</td>
<td>The number of positive / negative interactions</td>
</tr>
<tr>
<td>$\gamma, \tau, \zeta, \sigma$</td>
<td>The weight of positive / negative interactions, the weight of recent / past interactions</td>
</tr>
<tr>
<td>$q, q \in {1,\ldots,Q}$</td>
<td>The types of verifiers</td>
</tr>
<tr>
<td>$R_q$</td>
<td>The incentive of type-$q$ verifier</td>
</tr>
<tr>
<td>$\eta_q$</td>
<td>The latency of type-$q$ verifier</td>
</tr>
<tr>
<td>$\pi_q$</td>
<td>The benefit function of the block manager</td>
</tr>
<tr>
<td>$\eta_q$</td>
<td>The valuation function of verifiers</td>
</tr>
<tr>
<td>$l$</td>
<td>The weight parameter about the type-$q$ verifier’s incentive</td>
</tr>
<tr>
<td>$P$</td>
<td>The unit resource cost of block verification</td>
</tr>
<tr>
<td>$p_{\eta}$</td>
<td>The probability of a verifier belonging to type-$q$</td>
</tr>
<tr>
<td>$</td>
<td>M</td>
</tr>
<tr>
<td>$q_1$</td>
<td>The unit profit gain for the block manager</td>
</tr>
<tr>
<td>$\epsilon_1, \epsilon_2$</td>
<td>Coefficients about the network scale / verification latency</td>
</tr>
</tbody>
</table>

TABLE I: Key notations
• **Interaction Frequency:** It is known that the higher interaction frequency means that vehicle $V_i$ has more prior knowledge about RSU $RU_j$. The interaction frequency between $V_i$ and $RU_j$ is the ratio of the number of times that $V_i$ interacts with $RU_j$ to the average number of times that $V_i$ interacts with other RSUs during a time window $T$, i.e.,

$$IF_{i \rightarrow j} = \frac{N_{i \rightarrow j}}{N_i}$$

where $N_{i \rightarrow j} = (\alpha_i + \beta_i)$, and $N_i = \frac{1}{|S|} \sum_{s \in S} N_{i \rightarrow s}$. $S$ is the set of all RSUs (denoted as $RU_s$) interacting with vehicle $V_i$ during the time window. The higher interaction frequency leads to a higher reputation.

• **Interaction Timeliness:** In BloV, an RSU is not always trusted and reliable because the widely distributed RSUs may lack sufficient security protection and are vulnerable to be compromised. Both the trustfulness and reputation of $V_i$ to $RU_j$ are changing over time. The recent interactions have a higher impact on the local opinion of $V_i$ to $RU_j$. The time scale of recent interactions and past interactions is defined by $t_{recent}$, e.g., three days. The recent interactions and past interactions have different weights on the local opinions of vehicles. The parameter $\zeta$ represents the weight of recent interactions, and $\sigma$ represents the weight of past interactions. $\zeta + \sigma = 1, \zeta > \sigma$.

• **Interaction Effects:** Note that positive interactions increase RSUs’ reputation and negative interactions decrease the reputation of RSUs. Therefore, the negative interactions have a higher weight on the local opinions of vehicles than that of the positive interactions. Here, the weight of positive interactions is $\zeta$, and the weight of negative interactions is $\tau$, where $\chi + \tau = 1, \chi < \tau$. The weights of interaction timeliness and interaction effects are combined together to form a new interaction frequency as follows:

$$\begin{cases}
\alpha_i = \zeta \alpha_1^i + \sigma \chi \alpha_2^i, \\
\beta_i = \zeta \beta_1^i + \sigma \tau \beta_2^i.
\end{cases}$$

(4)

The positive and negative recent interactions are $\alpha_1^i$ and $\beta_1^i$ when the current time $t$ satisfies $t \leq t_{recent}$, respectively. When $t > t_{recent}$, the positive and negative past interactions are $\alpha_2^i$ and $\beta_2^i$, respectively. Therefore, the interaction frequency between $V_i$ to $RU_j$ is updated as follows:

$$IF_{i \rightarrow j} = \frac{N_{i \rightarrow j}}{N_i} = \frac{\chi (\alpha_1^i + \sigma \chi \alpha_2^i) + \tau (\beta_1^i + \sigma \tau \beta_2^i)}{\frac{1}{|S|} \sum_{s \in S} N_{i \rightarrow s}}.$$  

(5)

Therefore, the overall weight of reputation for local opinions is $\delta_{i \rightarrow j} = \rho_i \ast IF_{i \rightarrow j}$, where $0 \leq \rho_i \leq 1$ is a pre-defined weight parameter for reputation calculation.

### C. Recommended Opinions for Subjective Logic

After being weighted, the recommended opinions are combined into a common opinion in the form of $\omega_{x \rightarrow j}^{rec} := \{b_{x \rightarrow j}^{rec}, d_{x \rightarrow j}^{rec}, u_{x \rightarrow j}^{rec}\}$. Here,

$$\begin{align*}
b_{x \rightarrow j}^{rec} &= \frac{1}{|X|} \sum_{x \in X} \delta_{x \rightarrow j} b_{x \rightarrow j}, \\
d_{x \rightarrow j}^{rec} &= \frac{1}{|X|} \sum_{x \in X} \delta_{x \rightarrow j} d_{x \rightarrow j}, \\
u_{x \rightarrow j}^{rec} &= \frac{1}{|X|} \sum_{x \in X} \delta_{x \rightarrow j} u_{x \rightarrow j},
\end{align*}$$

(6)

where $x \in X$ is a set of recommenders, i.e., the other vehicles that have interacted with $RU_j$. Thus, the subjective opinions from different recommenders are combined into one single opinion, which is called the recommended opinion according to each opinion’s weight [11].

### D. Combining Local Opinions with Recommended Opinions

After obtaining ratings of $RU_j$ from the other vehicles, a particular vehicle has a subjective opinion (i.e., local opinion) on each RSU based on its interaction history. This local opinion should still be considered while forming the final reputation opinion to avoid cheating [11]. The final reputation opinion of $V_i$ to $RU_j$ is formed as $\omega_{x \rightarrow j}^{final} := \{b_{x \rightarrow j}^{final}, d_{x \rightarrow j}^{final}, u_{x \rightarrow j}^{final}\}$, where $b_{x \rightarrow j}^{final}$, $d_{x \rightarrow j}^{final}$, and $u_{x \rightarrow j}^{final}$ are respectively calculated as follows [14]:

$$\begin{align*}
b_{final}^{x \rightarrow j} &= b_{x \rightarrow j}^{rec} + \frac{\delta_{x \rightarrow j} \rho_{i} \ast IF_{i \rightarrow j}}{\tau_{x}^{\delta_{x \rightarrow j} \rho_{i} \ast IF_{i \rightarrow j}} + \tau_{x}^{\delta_{x \rightarrow j} \rho_{i} \ast IF_{i \rightarrow j}}}, \\
d_{final}^{x \rightarrow j} &= d_{x \rightarrow j}^{rec} + \frac{\delta_{x \rightarrow j} \rho_{i} \ast IF_{i \rightarrow j}}{\tau_{x}^{\delta_{x \rightarrow j} \rho_{i} \ast IF_{i \rightarrow j}} + \tau_{x}^{\delta_{x \rightarrow j} \rho_{i} \ast IF_{i \rightarrow j}}}, \\
u_{final}^{x \rightarrow j} &= u_{x \rightarrow j}^{rec} + \frac{\delta_{x \rightarrow j} \rho_{i} \ast IF_{i \rightarrow j}}{\tau_{x}^{\delta_{x \rightarrow j} \rho_{i} \ast IF_{i \rightarrow j}} + \tau_{x}^{\delta_{x \rightarrow j} \rho_{i} \ast IF_{i \rightarrow j}}}.
\end{align*}$$

(7)

Similar to Eqn. (2), the final reputation opinion of $V_i$ to $RU_j$ is

$$T_{x \rightarrow j}^{final} = \delta_{x \rightarrow j}^{final} + \gamma u_{final}^{x \rightarrow j}.$$  

(8)

The final reputation opinions can be used in different steps of the proposed DPoS scheme. For Step 2 and Step 7 in Section II-C, after obtaining the final reputation opinion on an RSU, vehicles will upload and store their final reputation opinions as recommended opinions for other vehicles (stakeholders) in the vehicular blockchain. For Step 3 and Step 4 in Section II-C, stakeholders vote high-reputation miner candidates according to the reputation opinions.

### V. INCENTIVE MECHANISM FOR SECURE BLOCK VERIFICATION USING CONTRACT THEORY

After selecting high-reputation miner candidates as active miners by using the multi-weight subjective logic model, there still exists a potential block verification collusion attack in the vehicular blockchain. In this section, for secure block verification, we aim to design an incentive mechanism to motivate more miners (both active miners and standby miners) to participate in the block verification. Every block manager will offer a part of the transaction fee as a reward to verifiers that participate in block verification and accomplish the tasks in time. Nevertheless, to do so, there are issues for the block manager in every consensus process. Firstly, the block manager does not have prior knowledge about which miners would like to participate in verification. Secondly, it does not have an accurate reputation value of a verifier. Thirdly, it does not
know the amount of resource that each verifier can contribute. The information asymmetry between the block manager and verifiers may incur too much cost for the block manager to give an incentive to the verifiers. Thus, the best strategy for the block manager is to design an incentive mechanism that can reduce the impact of information asymmetry. Moreover, the verifiers that contribute more should be rewarded more. Thus, we adopt contract theory [26] in designing the incentive mechanism.

In the $k$th block verification, consider a monopoly market consisting of a block manager acting as the task publisher and a set of verifiers $\mathbb{M} = \{M_1, \ldots, M_m\}$ including active miners and standby miners. The verifiers are willing to contribute different computation resources $C = \{c_1, \ldots, c_m\}$, i.e., CPU cycles per unit time to execute the block verification. $I_k$ and $O_k$ are the sizes of the transmitted block before verification and the verified results, respectively [26]. For simplicity, for all verifiers, the values of $I_k$ and $O_k$ are the same in the $k$th block verification. For a verifier $m$, the occupied CPU resource of block verification task is $\text{Task}_k^m$. Here, we consider that $\text{Task}_k^1 = \text{Task}_k^2 = \cdots = \text{Task}_k^m$. Therefore, the block verification task is denoted as a three-tuple $(\text{Task}_k^m, I_k, O_k)$. To attract more high-reputation verifiers, we define reputation as the type of a verifier. There are $Q$ types, and the verifiers are sorted in an ascending order of reputation: $\theta_1 < \cdots < \theta_q < \cdots < \theta_Q$, $q \in \{1, \ldots, Q\}$. The larger $\theta_q$ implies a higher reputation verifier for secure block verification among miners [6], [17].

With information asymmetry, the block manager should design specific contracts to overcome its economic problem. For different types of verifiers with different reputations, the block manager offers the verifiers a contract $(R_q, L_q^{-1})$, which includes a series of latency-reward bundles. Here, $L_q$ is the latency of block verification for type-$q$ verifiers and $L_q^{-1}$ is the reciprocal of $L_q$. $R_q(L_q^{-1})$ is the corresponding incentive. Note that if verifiers finish block verification faster, i.e., with smaller latency, can receive higher reward [26].

### A. Latency in Block Verification

As mentioned in Step 6 of Section II-C, there are four steps in the block verification process for a verifier: (i) unverified block transmission from the block manager to verifiers, (ii) local block verification, (iii) verification result broadcasting and verification result comparison among verifiers, and (iv) verification feedback transmission from the verifiers to the block manager. For a verifier $m$, the latency consisting of the corresponding delays of the aforementioned steps is defined as follows [26]:

$$L_q^{r_m} = I_k + \frac{\text{Task}_k^m}{c_m} + \psi I_k \lvert \mathbb{M} \rvert + \frac{O_k}{r_m}$$

Here, $L_q^{r_m}$ is the uplink transmission time from the block manager to verifier $m$, and $r_m$ is the downlink transmission time from the block manager to the verifiers. The transmission time of an unverified block from the block manager to the verifier is $\frac{I_k}{r_m}$. The local verification time of this block is $\frac{\text{Task}_k^m}{c_m}$. Similar to that in [17], [27], the time of verification result broadcasting and comparison among verifiers is a function of the block size $I_k$, network scale (i.e., the number of verifiers $\lvert \mathbb{M} \rvert$) and average verification speed of each verifier, which is denoted as $\psi I_k \lvert \mathbb{M} \rvert$. Here, $\psi$ is a pre-defined parameter of verification result broadcasting and comparison, which can be obtained from statistics of previous block verification processes. The time of verification feedback is $\frac{O_k}{r_m}$.

### B. Profit of the Block Manager

According to the signed contract $(R_q, L_q^{-1})$ between the block manager and type-$q$ verifier, the profit of the block manager obtained from type-$q$ verifier is denoted as

$$U_m(q) = \pi[\phi_q(L_q)] - lR_q,$$

where $l$ is a pre-defined weight parameter about the type-$q$ verifier’s incentive $R_q$. $\pi[\phi_q(L_q)]$ is the benefit of the block manager regarding a security-latency metric $\phi_q$ for type-$q$ verifier. Intuitively, the block manager obtains a higher profit when the $\phi_q$ is bigger. Moreover, both more high-reputation verifiers and less latency can lead to bigger $\phi_q$, i.e., $\frac{\phi_q(L_q)}{\phi_q} > 0$, $\frac{\phi_q(L_q)}{L_q} > 0$ and $\frac{L_q}{\phi_q} < 0$. The more verifiers participating in block verification leads to more secure block verification stage. However, this causes larger latency since the verifiers may need to communicate with verifiers through multi-hop relays [17]. Similar to that in [17], [28], we define a more general security-latency metric to balance the network scale and the block verification time for type-$q$ verifier, which is expressed by

$$\phi_q = \begin{cases} e_1(\theta_q \lvert \mathbb{M} \rvert)p_q e_1^2 \left(\frac{L_q}{\sum p_q}\right)^2 & \text{if } 0 < L_q < A, \\ 0 & \text{otherwise}. \end{cases}$$

Here $A = T_{\text{max}}e_1z_1^2 \left(\frac{\psi I_k \lvert \mathbb{M} \rvert}{e_2}z_2^2\right)$, $e_1 > 0$ and $e_2 > 0$ are pre-defined coefficients about the network scale and verification latency, respectively. $p_q$ is the prior probability of type-$q$, and $\sum p_q = 1$. We consider that the block manager can obtain the distribution of verifier types from observations and statistics of past behaviours of the verifiers [26]. $T_{\text{max}}$ denotes the maximum tolerable block verification latency to blockchain users. $z_1 \geq 1$ and $z_2 \geq 1$ are given factors indicating the effects of network scale and verification latency on block.
verification, respectively. The goal of the block manager is to maximize its profit through block verification as follows:

$$\max_{(R_q, L_q)} U_{bm}(q) = \sum_{q=1}^{Q} [M[p_q] (\pi[\phi_q(L_q)] - IR_q)].$$  \hspace{1cm} (13)

C. Utility of Block Verifiers

For type-\(q\) verifier, the utility function of block verification based on a signed contract is defined as

$$U_q = \theta_q \eta(q) - l' L_q^{-1},$$ \hspace{1cm} (14)

where \(\eta(q)\) is a monotonically increasing valuation function of type-\(q\) verifier in terms of the incentive \(R_q\). \(l'\) is the unit resource cost of block verification including the computation resource overhead and network resource overhead. Moreover, the valuation is zero when there is no incentive, i.e., \(\eta(0) = 0\). The higher type-\(q\) verifier should have higher utility because of higher reputation in block verification. However, the verifier wants to maximize its utility through minimizing resource consumption in block verification. Specifically, the objective of type-\(q\) verifier is to maximize utility obtained by joining block verification, expressed by

$$\max_{(R_q, L_q)} U_q = \theta_q \eta(q) - l' L_q^{-1}, \forall q \in \{1, \ldots, Q\}.$$ \hspace{1cm} (15)

VI. OPTIMAL CONTRACT DESIGNING

According to [29], to make contracts feasible, each contract item for verifiers must satisfy the following principles: (i) Individual Rationality (IR) and (ii) Incentive Compatibility (IC). IR means that each verifier will join the block verification when it receives a non-negative utility, i.e.,

$$\theta_q \eta(q) - l' L_q^{-1} \geq 0, \forall q \in \{1, \ldots, Q\}.$$ \hspace{1cm} (16)

IC refers to that type-\(q\) verifier can only receive the maximum utility when choosing the contract designed for itself instead of all other contracts \((R_{q'}, L_{q'}^{-1})\), i.e.,

$$\theta_q \eta(q) - l' L_{q'}^{-1} \geq \theta_{q'} \eta(q') - l' L_{q'}^{-1}, \forall q', q' \in \{1, \ldots, Q\}, q \neq q'.$$ \hspace{1cm} (17)

In what follows, we consider \(\pi[\phi_q(L_q)] = g_1 [c_1(\theta_q)M[p_q]^{\gamma_1} - c_2(L_q^{-1})^{\gamma_2}]\) for ease of presentation, where \(g_1\) is unit profit gain for the block manager. Therefore, the optimization problems in (13) and (15) can be defined as follows:

$$\max_{(R_q, L_q)} U_{bm} = \sum_{q=1}^{Q} [M[p_q]g_1c_1(\theta_q)M[p_q]^{\gamma_1} - g_1c_2(L_q^{-1})^{\gamma_2} - IR_q],$$ \hspace{1cm} \text{s.t.}

$$\theta_q \eta(q) - l' L_q^{-1} \geq 0, \forall q \in \{1, \ldots, Q\},$$

$$\theta_{q'} \eta(q') - l' L_{q'}^{-1} \geq \theta_{q'} \eta(q') - l' L_{q'}^{-1}, \forall q', q' \in \{1, \ldots, Q\}, q \neq q'.$$

$$\max\{L_q\} \leq T_{max}, \forall q \in \{1, \ldots, Q\},$$

$$\sum_{q=1}^{Q} |M[p_q]R_q | \leq R_{max}, \forall q \in \{1, \ldots, Q\}.$$ \hspace{1cm} (18)

This problem is not a convex optimization problem. However, we can find its solution by performing the following transformation.

**Lemma 1 (Monotonicity).** For contract \((R_i, L_i^{-1})\) and \((R_j, L_j^{-1})\), we have \(R_i \geq R_j\) and \(L_i^{-1} \geq L_j^{-1}\), if and only if \(\theta_i \geq \theta_j, i \neq j, i, j \in \{1, \ldots, Q\}\).

**Proof:** According to the IC constraints of type-\(i\) verifier and type-\(j\) verifier, we have

$$\theta_i \eta(R_i) - l' L_i^{-1} \geq \theta_j \eta(R_j) - l' L_j^{-1},$$ \hspace{1cm} (19)

$$\theta_i \eta(R_i) - l' L_i^{-1} \geq \theta_j \eta(R_j) - l' L_j^{-1}.$$ \hspace{1cm} (20)

By adding together (19) and (20), we can obtain \((\theta_i - \theta_j)[\eta(R_i) - \eta(R_j)] \geq 0\). \(\eta(R_q) \geq 0\) is a monotonically increasing valuation function of \(R_q\). When \(\theta_i \geq \theta_j\), we can deduce that \(\eta(R_i) - \eta(R_j) \geq 0\), i.e., \(R_i \geq R_j\). When \(R_i \geq R_j\), we have \(\eta(R_i) - \eta(R_j) \geq 0\). Thus, we can deduce that \(\theta_i \geq \theta_j\) must be satisfied [30].

**Proposition 1:** \(R_i \geq R_j\), if and only if \(L_i^{-1} \geq L_j^{-1}\).

**Proof:** According to the IC constraint in (19), we can obtain

$$\theta_i \eta(R_i) - \theta_j \eta(R_j) \geq l' (L_i^{-1} - L_j^{-1}),$$ \hspace{1cm} (21)

$$\theta_i \eta(R_i) - \theta_j \eta(R_j) \leq l' (L_i^{-1} - L_j^{-1}).$$ \hspace{1cm} (22)

As \(L_i^{-1} \geq L_j^{-1}\), we have \(\eta(R_i) \geq \eta(R_j)\) according to (21), and thus \(R_i \geq R_j\). In addition, when \(R_i \geq R_j\), we can obtain \(L_i^{-1} \geq L_j^{-1}\) from (22). **Proposition 1** indicates that an incentive compatibility contract requires a higher payment, if verifiers have less latency in block verification.

**Lemma 2.** If the IC condition of type-\(1\) verifier is satisfied, the IR constraints of other types will hold.

**Proof:** According to the IC constraints, \(\forall i \in \{2, \ldots, Q\}\), we have

$$\theta_i \eta(R_i) - l' L_i^{-1} \geq \theta_i \eta(R_i) - l' L_i^{-1}. \hspace{1cm} (23)$$

Given that \(\theta_1 < \cdots < \theta_i < \cdots < \theta_Q\), we also have

$$\theta_i \eta(R_i) - l' L_i^{-1} \geq \theta_i \eta(R_i) - l' L_i^{-1}. \hspace{1cm} (24)$$

According to (23) and (24), we have

$$\theta_i \eta(R_i) - l' L_i^{-1} \geq \theta_i \eta(R_i) - l' L_i^{-1} \geq 0. \hspace{1cm} (25)$$

The (25) indicates that with the IC condition, when the IR constraint of type-\(1\) verifier is satisfied, the other IR constraints will also hold. So the other IR constraints can be bounded into the IR condition of type-\(1\) verifier [30].

**Lemma 3.** By utilizing the monotonicity in **Lemma 1**, the IC condition can be transformed into the Local Downward Incentive Compatibility (LDIC), which is given as follows:

$$\theta_i \eta(R_i) - l' L_i^{-1} \geq \theta_j \eta(R_{i-1}) - l' L_{i-1}^{-1}, \forall i \in \{2, \ldots, Q\}.$$ \hspace{1cm} (26)

**Proof:** The IC constraints between type-\(i\) and type-\(j\), \(i \in \{1, \ldots, i-1\}\) are defined as Downward Incentive Compatibility (DIC), given by \(\theta_i \eta(R_i) - l' L_i^{-1} \geq \theta_j \eta(R_j) - l' L_j^{-1}\). The IC constraints between type-\(i\) and type-\(j\), \(i \in \{i+1, \ldots, Q\}\) are defined as Upward Incentive Compatibility (UIC), given by \(\theta_i \eta(R_i) - l' L_i^{-1} \geq \theta_j \eta(R_j) - l' L_j^{-1}\). We first prove that DIC can be reduced as two adjacent types in DIC, called LDIC. Consider three continuous types
where
\[ f_q = \begin{cases} \frac{v_p}{q} + \left[ \frac{v_p}{q} - \frac{v_{p,q+1}}{q} \right] \sum_{i=q+1}^{Q} p_i, & \text{if } q < Q, \vspace{0.5em} \\
\frac{v_{p,q}}{q}, & \text{if } q = Q. \end{cases} \] (37)

We substitute the expression in (36) into the problem in (34) and remove all \( R_q \), \( \forall q \in \{1, \ldots, Q\} \) from the problem in (34). The problem in (34) is rewritten as follows:
\[
\max_{(R_q, L_q^{-1})} U_{bn} = \sum_{q=1}^{Q} [M|p_q]|g_1\varepsilon_1(\theta_q[M|p_q]|) - g_1\varepsilon_2\left(\frac{T_{max}}{\lambda}\right)^{2z} - |M|\sum_{q=1}^{Q} f_q L_q^{-1},
\]
subject to
\[
L_q^{-1} \geq \frac{1}{\lambda_{max}}, \forall q \in \{1, \ldots, Q\}; \vspace{0.5em} \\
|M|\sum_{q=1}^{Q} f_q L_q^{-1} \leq R_{max}, \forall q \in \{1, \ldots, Q\}. \vspace{0.5em} \]
(38)

By differentiating \( U_{bn} \) with respect to \( L_q^{-1} \), we have
\[
\frac{\partial U_{bn}}{\partial L_{q}} = -\frac{|M|g_1\varepsilon_2 z^2 [L_q^{-1}] - (z^2 + 1) - |M|f_q}{T_{max}/\lambda - \lambda}; \vspace{0.5em} \\
\frac{\partial^2 U_{bn}}{\partial L_{q}^2} = -\frac{|M|g_1\varepsilon_2 z [L_q^{-1}] + z^2}{(z^2 + 1)^2}, \vspace{0.5em} \\
\frac{\partial^2 U_{bn}}{\partial L_{q}^{2}} < 0. \vspace{0.5em} \]
Thus, the function \( U_{bn} \) is concave. The problem defined in (38) is a convex optimization problem because the summation of concave functions \( U_{bn} \) is still a concave function, and the constraints are affine. We can obtain the optimal latency requirement \( L_{q}^{-1}^* \) and the corresponding incentive \( R_q^* \) by using convex optimization tools. Moreover, if the types of verifiers follow a uniform distribution, the monotonicity can be automatically met [29], [30]. If not, we can use infeasible sub-sequence replacing algorithm to satisfy the final optimal latency requirement [31].

Note that the proposed incentive mechanism based on contract theory can encourage efficiently high-reputation miners to join the block verification for further improving the security of the vehicular blockchain.

VII. NUMERICAL RESULTS

In this section, we first evaluate the performance of the proposed Multi-Weight Subjective Logic (MWSL) scheme based on a real-world dataset of San Francisco Yellow Cab [32]. Next, we evaluate and compare the performance of the proposed incentive mechanism based on contract theory. The mobility traces of 536 taxis driving during a month are recorded in this dataset. We observe 200 taxis running in an urban area, whose latitude and longitude are from 37.7 to 37.81 and from -122.52 to -122.38, respectively. Fig. 3 shows trace points of the 200 taxis during a month. The average time gap between two trace records is 43.34 seconds. There are 400 RSUs (miner candidates) deployed uniformly in the observation area. The update period of RSUs’ reputation is 1 minute. These miner candidates are initially classified into 10 types according to their reputation values, wherein the probability for a candidate belonging to a certain type is 0.1. Major parameters used in the simulation are given in Table II, most of which are adopted from [14], [26], [30].

of verifiers, i.e., \( \theta_i < \theta_{i+1}, i \in \{2, \ldots, Q - 1\} \), we have
\[
\theta_{i+1}\eta(R_{i+1}) - l' L_{i+1}^{-1} \geq \theta_i\eta(R_i) - l' L_i^{-1}; \vspace{0.5em} \\
\theta_i\eta(R_i) - l' L_i^{-1} \geq \theta_{i+1}\eta(R_{i+1}) - l' L_{i+1}^{-1}. \vspace{0.5em} \]
(27)
Combining (28) and (30), we have
\[
\begin{align*}
\theta_{i+1}\eta(R_{i+1}) - \theta_i\eta(R_i) & \geq (\theta_{i+1} - \theta_i)[\eta(R_{i+1}) - \eta(R_i)]; \\
\theta_i\eta(R_i) - \eta(R_{i+1}) & \geq (\theta_i - \theta_{i+1})[\eta(R_{i+1}) - \eta(R_i)].
\end{align*} \vspace{0.5em} \\
(29)
\end{align}
Combining (28) and (30), we have
\[
\begin{align*}
\theta_{i+1}\eta(R_{i+1}) - \eta(R_{i+1}) & \geq \theta_i\eta(R_i) - \eta(R_{i+1}); \\
\theta_i\eta(R_i) - \eta(R_{i+1}) & \geq \theta_{i+1}\eta(R_{i+1}) - l' L_{i+1}^{-1}.
\end{align*} \vspace{0.5em} \\
(30)
\end{align}
Combining (27) and (31), we have
\[
\theta_{i+1}\eta(R_{i+1}) - l' L_{i+1}^{-1} \geq \theta_{i+1}\eta(R_{i+1}) - l' L_{i+1}^{-1}. \vspace{0.5em} \]
(31)
Combining (27) and (31), we have
\[
\theta_{i+1}\eta(R_{i+1}) - l' L_{i+1}^{-1} \geq \theta_{i+1}\eta(R_{i+1}) - l' L_{i+1}^{-1}. \vspace{0.5em} \]
(32)

Furthermore, to simplify the analysis without loss of generality, we define the concave function \( \eta(R_q) = R_q^* \). The optimization problem in (34) is solved sequentially. Firstly, we solve the relaxed problem in (34) without monotonicity to obtain a solution. Secondly, we verify whether the solution satisfies the condition of the monotonicity. We use the method of iterating the IC and IR constraints to obtain \( R_q \) which can be expressed as follows:
\[
R_q = \frac{l' L_{i}^{-1}}{\theta_i} + \sum_{k=2}^{Q} \Delta_k, \vspace{0.5em} \]
(35)
where \( \Delta_k = \frac{l' L_{i}^{-1}}{\theta_k} - \frac{l' L_{i}^{-1}}{\theta_k} \) and \( \Delta_1 = 0 \). By substituting \( R_q \) into \( \sum_{q=1}^{Q} [M|p_q|R_q] \), we have
\[
\sum_{q=1}^{Q} [M|p_q|R_q] = [M|q] \sum_{q=1}^{Q} f_q L_q^{-1}, \vspace{0.5em} \]
(36)
TABLE II: Parameter Setting in the Simulation

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Setting</th>
</tr>
</thead>
<tbody>
<tr>
<td>Interaction frequency between vehicles and RSUs</td>
<td>[50, 200] times/week</td>
</tr>
<tr>
<td>Coverage range of RSUs</td>
<td>[300, 500] m</td>
</tr>
<tr>
<td>Speed of vehicles</td>
<td>[50, 150] km/h</td>
</tr>
<tr>
<td>Weight parameters</td>
<td>(\chi = 0.4, \tau = 0.6, \zeta = 0.6, \sigma = 0.4, \rho = 1)</td>
</tr>
<tr>
<td>Time scale of recent and past events (t_{\text{recent}})</td>
<td>three days</td>
</tr>
<tr>
<td>Rate of compromised vehicles</td>
<td>[10%, 90%]</td>
</tr>
<tr>
<td>Successful transmission probability of data packets</td>
<td>[0.5, 1]</td>
</tr>
<tr>
<td>Vehicle to RSU bandwidth</td>
<td>20 MHz</td>
</tr>
<tr>
<td>Noise spectrum density</td>
<td>-174 dBm/Hz</td>
</tr>
<tr>
<td>Transmission power</td>
<td>[10, 23] dBm</td>
</tr>
<tr>
<td>Receiver power</td>
<td>14 dBm</td>
</tr>
<tr>
<td>Computation resource</td>
<td>(10^5, 10^7) CPU cycles/unit time</td>
</tr>
<tr>
<td>Input/output block data size</td>
<td>[50, 500] KB</td>
</tr>
<tr>
<td>Pre-defined parameters</td>
<td>(g_1 = 1.2, e_1 = 15, e_2 = 10, z_1 = 2, z_2 = 1, l = 5, l' = 1, T_{\text{max}} = 300 \text{s}, R_{\text{max}} = 1000, \psi = 0.5)</td>
</tr>
</tbody>
</table>

![Fig. 3: Spatial distribution of vehicle trace points.](image)

**A. Performance of the proposed reputation scheme**

In the proposed MWSL scheme, vehicles calculate reputation value of miner candidates according to local opinions and recommended opinions from other vehicles. We compare our MWSL scheme with a Traditional Subjective Logic (TSL) scheme which is a typical model using a linear function to calculate reputation [14]. More specifically, \(T_i^{t_{\text{recent}}} = (1 - \kappa)T_{\text{ave}} + \kappa T_{\text{lais}}\), where \(T_{\text{ave}} = b_{i_{\text{ave}}}^{j_{\text{ave}}} + 0.5a_{i_{\text{ave}}}^{j_{\text{ave}}}\) and \(T_{\text{lais}} = b_{i_{\text{lais}}}^{j_{\text{lais}}} + 0.5a_{i_{\text{lais}}}^{j_{\text{lais}}}\). Here \(\kappa\) is the weight and is set to be 0.5. \(b_{i_{\text{ave}}}^{j_{\text{ave}}}\) and \(a_{i_{\text{ave}}}^{j_{\text{ave}}}\) are average values of other vehicles’ \(b_{i_{\text{ave}}}^{j_{\text{ave}}}\) and \(a_{i_{\text{ave}}}^{j_{\text{ave}}}\), respectively. \(b_{i_{\text{lais}}}^{j_{\text{lais}}}\) and \(a_{i_{\text{lais}}}^{j_{\text{lais}}}\) are the latest \(b_{i_{\text{lais}}}^{j_{\text{lais}}}\) and \(a_{i_{\text{lais}}}^{j_{\text{lais}}}\) in the local opinion of vehicle \(i\) for RSU \(j\). We consider a malicious miner candidate will initially pretend to behave well to obtain positive reputation values from vehicles in the former 5 minutes. Then, this candidate colludes with 10 compromised vehicles and begins to misbehave to 50 well-behaved vehicles randomly. These well-behaved vehicles will generate negative reputation opinions for the candidate, while the colluded vehicles still generate positive reputation opinions for the candidate and vote it as a miner in the voting stage.

![Fig. 4: The reputation values of a malicious miner.](image)

![Fig. 5: Detection rate of malicious miners under different threshold values of trusted miners.](image)
We observe the detection rate of 10 malicious miner candidates using the TSL and MWSL schemes during 60 minutes. Fig. 5 shows that the MWSL scheme has a much higher successful detection rate of malicious miners than that in the TSL scheme. We define a metric as the reputation threshold of successful detection, in which only the reputation of malicious miners below the threshold can be detected successfully. When the reputation threshold of successful detection is 0.5, the detection rate of the MWSL scheme is 100%, which is 100% higher than that of the TSL scheme. Due to higher detection rate in the MWSL scheme, potential security threats can be detected and prevented more effectively, which leads to a more secure BIoV.

From Fig. 5, we can observe that successful detection probability is not high enough when the reputation threshold of successful detection is very low, e.g., 0.2. In the case with a very low threshold, the active miners generated by reputation voting may launch the verification collusion attack, that more than 1/3 active miners collude to generate false verification result for a data block [13], [33]. To defend this intractable attack, standby miners should participate in block verification to improve the correct probability of verified block. The correct probability of verified block indicates that the data block is correctly verified without the effects of the verification collusion attack. Fig. 6 shows the correct probability of data block after verification with respect to different reputation thresholds of successful detection. When the reputation threshold of successful detection is 0.2, the correct probability in our MWSL scheme with standby miners is 13% higher than that of MWSL scheme without standby miners, while the TSL scheme without standby miners cannot defend against this collusion attack. This indicates that the proposed MWSL can ensure a secure block verification, even when attackers launch internal active miner collusion attack.

B. Performance of the incentive mechanism based on contract theory scheme

A block manager acting as the contract publisher announces the designed contract items to other active miners and standby miners. Each miner chooses a contract item \((R_q, L_q^{-1})\) to
Fig. 10: Utility comparison under different resource overhe ads.

Fig. 11: Utility comparison under different vehicle speeds.

Fig. 12: Running time of the proposed incentive mechanism.

The social welfare of the contract theory is larger than those game models. The reason is that in social welfare achieved by the contract theory and Stackelberg game models. Similarly, Fig. 9 shows that the values of the more secure block verification. The proposed contract choices for high-type (high-reputation) verifiers, leading to verifier types bring both more verifiers and contract item increases with the total number of verifier types. The more from [30]. Fig. 8 shows that the profit of a block manager which validates the IR constraint [26].

We compare the profit of a block manager obtained from the proposed contract model, and Stackelberg game models from [30]. Fig. 8 shows that the profit of a block manager increases with the total number of verifier types. The more verifier types bring both more verifiers and contract item choices for high-type (high-reputation) verifiers, leading to the more secure block verification. The proposed contract model has better performance than those of the Stackelberg game models. Similarly, Fig. 9 shows that the values of social welfare achieved by the contract theory and Stackelberg game models increase with the total number of verifier types. The social welfare of the contract theory is larger than those of the two Stackelberg game models. The reason is that in the monopoly market, the block manager working as the monopolist provides limited contract items to extract more benefits from the verifiers. The proposed contract theory favors the utility of the block manager. However, in the Stackelberg game models, rational verifiers can optimize their individual utilities thus leading to a less profit for the block manager. Although the block manager needs to consider the IR and IC constraints during designing the contract items, these constraints are only basic constraints for verifiers’ benefit. The constraints are weak in terms of maximizing the benefits of the verifiers when comparing with the Stackelberg game models [34]. Therefore the block manager can obtain a higher profit than those in the Stackelberg game models. Moreover, the Stackelberg game model with symmetric information has better performance than that of Stackelberg game model with asymmetric information. The reason is that the game leader (the block manager) in the Stackelberg game with symmetric information can effectively optimize its profit because of knowing the actions of followers (verifiers), i.e., the symmetric information, and set the utilities of the followers to zero [30].

In the proposed schemes, there exist resource overheads including computation and network resource cost in block verification for verifiers. Fig. 10 shows that the utility of the block manager and the total utility of all verifiers both decrease with the increase of resource overheads. Similarly, the social welfare of the proposed scheme also decreases when the overhead increases. To exhibit the impact of vehicle speeds, we perform utility comparison of both block managers and verifiers under different vehicle speeds. Fig. 11 shows that higher vehicle speed results in larger utility of the block managers but smaller average utility of the verifiers. The reason is that the uncertainty of a local opinion vector is determined by the communication quality between a vehicle and the RSU according to the subjective logic model. Higher vehicle speed results in more serious negative effects, e.g., bigger delivery delay and more frame loss, on the communication quality, which leads to greater uncertainty [35], [36]. The greater uncertainty decreases the accuracy degree of reputation calculation and thus leads to more miner candidates meeting the condition on a certain threshold of trusted miner (e.g., 3.5) participating in block verification. As a result, the utility of the block
manager increases with the increasing vehicle speed when the total number of verifier types is 10. However, in such an environment, the average utility of the verifiers decreases because of the competition among the increasing number of verifiers.

We also perform complexity analysis for the proposed schemes. For the reputation calculation scheme, the reputation values of miner candidates are calculated by integrating vehicles’ local opinions with other vehicles’ recommended opinions. These recommended opinions from vehicles are combined into a common opinion. The complexity of the reputation calculation scheme is \( O(n^2) \). For the incentive mechanism, we solve the optimization problem of the incentive mechanism (i.e., Eqn. (38)) by using the CVX tool, which is a MATLAB-based modeling system for convex optimization [29]. We analyze the running time of the incentive mechanism with respect to different network scales in Fig. 12. The results in the figure are averaged over 50 trials. Fig. 12 shows the relation between the running time and the network scales, i.e., the number of verifiers, is close to being linear. This indicates that the incentive mechanism can be easily applicable to large networks. In summary, both the reputation calculation scheme using the multi-weight subjective logic model and the incentive mechanism using contract theory are the effective solutions for large vehicular networks with acceptable complexity.

VIII. CONCLUSION

In this paper, we have introduced blockchain-enabled Internet of vehicles for secure P2P vehicle data sharing by using a hard security solution, i.e., the enhanced Delegated Proof-of-Stake consensus scheme. This DPoS consensus scheme has been improved by a two-stage soft security enhancement solution. The first stage is to select miners by reputation-based voting. A multi-weight subjective logic scheme has been utilized to calculate securely and accurately the reputation of miner candidates. The second stage is to incentivize stand-by miners to participate in block verification using contract theory, which can further prevent internal collusion of active miners. Numerical results have indicated that our multi-weight subjective logic scheme has great advantages over traditional reputation schemes in improving detection rate of malicious miner candidates. Likewise, the proposed contract-based block verification scheme can further decrease active miners collusion and optimize the utilities of both the block manager and verifiers to further improve the security of vehicle data sharing. In the future work, we can further improve the accuracy of the miner candidates’ reputation calculation by taking more weights into consideration.

REFERENCES


Jiawen Kang received the M.S. degree from the Guangdong University of Technology, China, in 2015, and the Ph.D. degree at the same school in 2018. He is currently a postdoc at Nanyang Technological University, Singapore. His research interests mainly focus on blockchain, security and privacy protection in wireless communications and networking.

Zehui Xiong (S’17) received his B.Eng degree with honors in Telecommunication Engineering from Huazhong University of Science and Technology, Wuhan, China, in 2016. He is currently working towards the Ph.D. degree in the School of Computer Science and Engineering, Nanyang Technological University, Singapore. His research interests include network economics, game theory for resource management, market models and pricing.

Dusit Niyato (M’09–SM’15–F’17) is currently a professor in the School of Computer Science and Engineering, at Nanyang Technological University, Singapore. He received B.Eng. from King Mongkuts Institute of Technology Ladkrabang (KMITL), Thailand in 1999 and Ph.D. in Electrical and Computer Engineering from the University of Manitoba, Canada in 2008. His research interests are in the area of energy harvesting for wireless communication, Internet of Things (IoT) and sensor networks.

Dongdong Ye received the M.S. degree from the Guangdong University of Technology, China, in 2018. He is currently working towards the Ph.D. degree in Guangdong University of Technology, China. His research interests mainly focus on game theory, resource management in wireless communications and networking, data sharing and trading.

Jiawen Kang

Zehui Xiong (S’17)

Dusit Niyato (M’09–SM’15–F’17)

Dongdong Ye

Dong In Kim (S’89–M’91–SM’02–F’19) received the Ph.D. degree in electrical engineering from the University of Southern California, Los Angeles, CA, USA, in 1990. He was a Tenured Professor with the School of Engineering Science, Simon Fraser University, Burnaby, BC, Canada. Since 2007, he has been with Sungkyunkwan University, Suwon, South Korea, where he is currently a Professor with the College of Information and Communication Engineering. He was a first recipient of the NRF of Korea Engineering Research Center in Wireless Communications for RF Energy Harvesting from 2014 to 2021. From 2002 to 2011, he served as an Editor and a Founding Area Editor of Cross-Layer Design and Optimization for the IEEE Transactions on Wireless Communications. From 2008 to 2011, he served as the Co-Editor-in-Chief of the IEEE/KICS Journal of Communications and Networks. He served as the Founding Editor-in-Chief of the IEEE Wireless Communications Letters from 2012 to 2015. From 2001 to 2014, he served as an Editor of Spread Spectrum Transmission and Access for the IEEE Transactions on Communications. Since 2015, he is serving as an Editor-at-Large of Wireless Communication I for the IEEE Transactions on Communications. He has been elevated to the grade of Fellow of the IEEE for contributions to cross-layer design of wireless communication systems, Fellow of the Korean Academy of Science and Technology (KAST) and Fellow of the National Academy of Engineering of Korea (NAEK).

Dong In Kim

Jun Zhao (S’10–M’15) is currently an Assistant Professor in the School of Computer Science and Engineering at Nanyang Technological University (NTU) in Singapore. He received a PhD degree in Electrical and Computer Engineering from Carnegie Mellon University (CMU) in the USA (advisors: Virgil Gligor, Osman Yagan; collaborator: Adrian Perrig) and a bachelor’s degree from Shanghai Jiao Tong University in China. Before joining NTU first as a postdoc with Xiaokui Xiao and then as a faculty member, he was a postdoc at Arizona State University as an Arizona Computing PostDoc Best Practices Fellow (advisors: Junshan Zhang, Vincent Poor). His research interests include blockchains, security, and privacy with applications to the Internet of Things and deep learning. One of his first-authored papers was shortlisted for the best student paper award in IEEE International Symposium on Information Theory (ISIT) 2014, a prestigious conference in information theory.

Jun Zhao