A Proof-of-Quality-Factor (PoQF) based blockchain and edge computing for vehicular message dissemination

Article (Accepted Version)

A Proof-of-Quality-Factor (PoQF) based Blockchain and Edge Computing for Vehicular Message Dissemination

Ferheen Ayaz, *Student Member, IEEE,* Zhengguo Sheng, *Senior Member, IEEE,* Daxin Tian, *Senior Member, IEEE* and Yong Liang Guan, *Senior Member, IEEE*

***Abstract*—Blockchain applications in vehicular networks can offer many advantages including decentralization and improved security. However, most of consensus algorithms in blockchain are difficult to be implemented in a Vehicular Ad-Hoc Networks (VANET) without the help of edge computing services. For example, the connectivity in VANET only remains for a short period of time, which is not sufficient for highly time consuming consensus algorithms, e.g., Proof-of-Work, running on mobile edge nodes (vehicles). Other consensus algorithms also have some drawbacks, e.g. Proof-of-Stake (PoS) is biased towards nodes with higher amount of stakes and Proof-of-Elapsed-Time (PoET) is not highly secure against malicious nodes. For these reasons, we propose a voting blockchain based on Proof-of-Quality-Factor (PoQF) consensus algorithm, where threshold number of votes is controlled by edge computing servers. Specifically, PoQF includes voting for message validation and a competitive relay selection process based on probabilistic prediction of channel quality between transmitter and receiver. The performance bounds of failure and latency in message validation are obtained. The paper also analyzes the throughput of block generation, as well as the asymptotic latency, security and communication complexity of PoQF. An incentive distribution mechanism to reward honest nodes and punish malicious nodes is further presented and its effectiveness against collusion of nodes is proved using game theory. Simulation results show that PoQF reduces failure in validation by 11% and 15% as compared to PoS and PoET, respectively, and is 68 ms faster than PoET.**

***Index Terms*—blockchain, PoS, PoET, PBFT, edge computing.**

I. Introduction

VEHICLES equipped with on-board computers offer limited computing and storage capabilities. However, in a vehicular edge computing (VEC) network, the mobile edge nodes (vehicles) with limited resources are able to offload heavy computational tasks to nearby Road Side Units (RSUs). One of the main objectives of VEC is to support infotainment applications and ensure road safety. However, due to high mobility of nodes and changing transmission rates, there are a large number of communications failures and delays between mobile nodes and RSUs [1]. For delay-sensitive applications, such as, emergency message dissemination, VEC allows nodes to exchange messages among themselves in a decentralized manner, forming a Vehicular Ad-Hoc Network (VANET). On the other hand, a blockchain is a distributed ledger which can record transactions in a trusted and credible environment without the requirement of a central authority. The features of blockchain, such as distributed nature, independence from third party and consensus to validate transactions, are some of the essential requirements of message dissemination in VANET. Therefore, the combination of blockchain and VANET can potentially result in secure and reliable vehicle-to-vehicle (V2V) communications [2]. Table I summarizes challenges associated with message dissemination in VANETs and corresponding solutions provided by blockchain. However, the dynamic network nature of VANET limits the connectivity between two nodes to a short period of time. Moreover, technical challenges, such as broadcast storm, packet collision and computing complexity need to be addressed in VANET environment while implementing blockchain [3]. Therefore, new blockchain solutions need to be developed using VEC networks for fully utilizing blockchain framework.

TABLE I: VANET issues & opportunities using blockchain.

Issues in VANET	Blockchain-based Solutions
False message generation	Consensus for validation
Privacy requirement	Cryptographic keys
Broadcast storm / relay selection	Leader election
Need of economic model	Incentives for block generation
Trust without third party required	Decentralization

The consensus algorithm in a blockchain is used for trusting a transaction. Nodes undergo a validation process, termed as consensus, before recording a transaction in a block. The nodes participating in a consensus are mining nodes and the node which successfully generates a block is known as leader [4]. One of the most popular consensus algorithms is Proof-of-Work (PoW), in which all nodes attempt to find a solution to a hash puzzle. The node which finds the solution first is elected as a leader, it will add next block to the blockchain and earn mining incentive. The computation cost to find a solution of a hash puzzle takes around ten minutes [5]. Distribution

This research was sponsored by H2020-MSCA-RISE (101006411 — SEEDS), National Natural Science Foundation of China under Grant No. 61822101, Beijing Municipal Natural Science Foundation No. L191001 and 4181002 and A*STAR under its RIE2020 Advanced Manufacturing and Engineering (AME) Industry Alignment Fund–Pre Positioning (IAF-PP) (Grant No. A19D6a0053). *(Corresponding author: Daxin Tian.)*

F. Ayaz and Z. Sheng are with the Department of Engineering and Design, University of Sussex, Brighton, BN1 9RH, U.K. e-mail: (f.ayaz@sussex.ac.uk, z.sheng@sussex.ac.uk).

D. Tian is with the School of Transportation Science and Engineering, Beihang University, Beijing, 100191, China. e-mail: (dtian@buaa.edu.cn)

Y. L. Guan is with the School of Electrical and Electronic Engineering, Nanyang Technological University, 639798, Singapore. e-mail: (eylguan@ntu.edu.sg)

of huge computation load of PoW over the edge system is recommended as a solution but evaluation of cost and contribution of individual edge device in a heterogeneous network is still unexplored [6]. A number of time-saving alternatives to PoW have also been proposed. One of the most commonly used consensus algorithms with connected vehicles is Proof-of-Stake (PoS), where the reputation of a node is considered as stake [7], [8]. PoS reduces the latency of a consensus but does not provide a fair competition to elect a leader. It is biased towards nodes with higher amount of stakes. The fairness with less computation workload is achieved by another consensus algorithm, known as Proof-of-Elapsed-Time (PoET), in which each node generates a random number to determine the waiting time after which it can generate a block. However, existing literature proves its weakness in security and vulnerability in the presence of malicious nodes [9]. The Practical Byzantine Fault Tolerant (PBFT) consensus algorithm which requires at least $2f + 1$ votes to validate a transaction, where f is the number of faulty nodes [10], is suggested to be suitable for vehicular applications because of its high throughput and ability to negotiate message validity [11], [12]. It is analogous to threshold based message validation in which a message is considered valid only if it is confirmed by a threshold number of nodes located in a close proximity of a sender [13]. The threshold value is crucial in such validation. A low threshold value may lead to false validation, whereas a high threshold value can result in increased latency. However, a threshold based message validation can be made efficient if the threshold value is adaptable to network conditions and requirements and can be varied using edge computing resources. It can be summarized that the measurements required to evaluate the performance of a consensus algorithm are

- *Security*: number of malicious nodes it can control without altering the original validity status of a transaction and its ability to resolve forks and prevent cheating.
- *Validation latency*: time required to validate a transaction.
- *Throughput*: number of blocks generated per second.

This paper proposes a Proof-of-Quality-Factor (PoQF) consensus algorithm for vehicular networks, where the message validation and Quality Factor in determining multi-hop relaying can be run efficiently on mobile edge nodes in a decentralized manner. For a successful packet transmission, Signal to Interference Noise Ratio (SINR) plays a crucial role [14], [15]. Therefore, SINR is considered as a metric in relay node selection. As SINR depends on the distances among nodes which vary with time in vehicular networks, the probability that SINR exceeds a certain threshold is predicted using mobility models in which positions or distances are regarded as random variables following some probability distribution [16]. The main contributions of the paper are

- We propose a PoQF consensus, where mobile edge nodes serve as mining nodes. Instead of solving a hash puzzle, they select relays along with validating a message.
- We derive the bounds of failure and latency in validating a message as well as the throughput of block generation. The asymptotic latency, security and communication complexity of PoQF are also discussed.
- We propose an incentive distribution mechanism to reward honest nodes and punish malicious mining nodes and analyze its performance using game theory.

TABLE II: Consensus algorithms used in VANETs.

Purpose	Consensus
Consensus run by RSUs	PoW [8], [31], [38]
Use of edge computing	PoW [37]
Trust / reputation management	PoS / DPoS [7], [8], [40]
Message Validation	PBFT [12], [29], [30], [35], [41], [42]
Achieving high throughput	PBFT [11], [31], PoS [36]

The rest of the paper is organized as follows. Section II describes the related work. Section III explains the proposed blockchain design. The theoretical performance of our work is analyzed in Section IV. Simulation results are discussed in Section V and Section VI concludes the paper.

II. Related Work

A. Vehicular Edge Computing (VEC)

In [1], the challenges in VEC such as transmission failures and delays during offloading are addressed and a context-aware opportunistic offloading scheme utilizing fog computing is proposed. VEC is recommended as an efficient support to emerging applications such as Artificial Intelligence (AI), Software Define Network (SDN) and blockchain in [17]. The advantages of combining mobile edge computing, Internet of Vehicles (IoV) and AI are highlighted in [18] and [19]. Both of them suggest Deep Reinforcement Learning (DRL) as the key technique to bring intelligence in VEC networks. Collocating edge computing servers with radio access networks for satisfying latency requirements of message dissemination in IoV is proposed in [20]. In [21], the problem of inappropriate utilization of resources is resolved by blockchain.

B. Collective Mining

Existing literature aims to achieve a better performance of blockchain consensus, at the same time retaining its feature of decentralization. One of the solutions to improve validation latency and throughput in block generation is to introduce collective mining. In this scheme, multiple mining nodes collectively decide whether a transaction is valid and should be added to a blockchain [22]. Byzcoin is an example of collective mining [23]. It leads to parallel blockchain extension and the mining incentive is shared among all mining nodes. Bitcoin-NG [24] divides time into multiple slots. In each slot, a leader can append transactions until a new leader is elected. There are two types of blocks in Bitcoin-NG: keyblock and microblock. The leader is elected by solving a cryptographic puzzle. The keyblock stores the solution of hash puzzle and the microblock contains ledger entries. Another approach of collective mining is called sharding in which mining nodes are grouped into committees and work in parallel. Each committee runs PBFT consensus on different set of transactions (shards) at the same time for achieving a high throughput [25].

C. Blockchain based Incentive Distribution

Blockchain based economic model for incentive distribution in federated learning utilizing edge computing framework is recommended in [6]. Secure blockchain based incentive mechanisms are also proposed in literature to encourage cooperative message delivery and data sharing in distributed peer-to-peer (P2P) applications. In [26], a pricing strategy to ensure successful message delivery using blockchain is presented and proved to be secure against collusion of intermediate nodes and receiver using game theory. It is proposed to verify transactions of incentives distributed among relay nodes by mining nodes. Similarly, in [27], P2P data sharing using public blockchain is proposed. Its incentive mechanism is analyzed by evolutionary game model and the cooperative behavior of nodes is analyzed by repeated game model. In both [26] and [27], the incentive mechanism is proved to encourage cooperation among nodes by including a charge mandatory to be paid by transmitting nodes. Incentive based message relaying in distributed P2P applications using blockchain is also proposed in [28] and proved to be secure against selfish behavior. In [26] - [28], the incentive distribution among relay nodes is proposed, but the incentive for mining nodes to promote participation and the type of consensus algorithm to be processed are not discussed. In [29] and [30], the incentive based message delivery in wireless ad-hoc networks for smart cities and Intelligent Transportation Systems (ITS) is presented, where message is validated using PBFT, and the incentives and privacy are controlled using blockchain. In [31], a blockchain based data sharing in VANETs is proposed. PoW is used by Road Side Units (RSUs) to add a data block, whereas PBFT is used by vehicle nodes for block announcement.

D. Blockchain based Vehicular Communications

In [32], blockchain is proposed for decentralization, data security and privacy in IoV and technical difficulties to implement blokchchain in IoV, such as, high speed of moving vehicles and error-prone wireless transmission links, are discussed. In [33], these technical difficulties are suggested to be solved using DRL for altering block size and interval. Blockchain is also recommended for privacy preserving and efficient database management in railway vehicles [34].

Selection of blockchain consensus suited to IoV is widely discussed in literature. PBFT is recommended as a suitable consensus for message validation among connected vehicles in [12], [33] and [35]. Meanwhile, PoS is also compared with PoW and suggested as a promising consensus for IoV because of its low energy consumption and reduced time delay in [36]. A blockchain based message dissemination in VANETs utilizing edge computing is proposed in [37]. It uses PoW and achieves latency reduction in block generation by offloading complex computations to capable edge devices. Its blockchain is used to store trust values of nodes, which is updated according to the validity of message initiated by the individual node. Similarly, [38] also proposes a blockchain to store trust values and message ratings, where hash computations are performed by RSUs. On the contrary, [39] shows that a completely distributed P2P blockchain in VANETs with least possible reliance on RSU and infrastructure is not possible to be implemented with PoW, but an RSU-dependent network will be a costly solution. A joint PoW and PoS consensus managed by RSUs is proposed in [8] to store trust values of nodes and evaluate the credibility of message based on trust value of senders. Delegated Proof-of-Stake (DPoS) is proposed in [7], where only selected nodes take part in consensus. The mining nodes are selected on the basis of reputation. This approach is based on the assumption that RSUs with edge computing infrastructures have sufficient computation and storage resources to process and store reputation of all nodes. DPoS is also used for blockchain-enabled data sharing during rescue missions in disaster-affected areas, where IoV is assisted by Unmanned Aerial Vehicles (UAVs) [40].

The prior work related to blockchain in VANETs mostly focus on credibility based message validation. In [2], the impact of high mobility on blockchain based VANETs is evaluated, but a cost-effective solution to overcome this challenge is still needed. A consolidated solution integrating message validation and dissemination using PBFT based consensus, blockchain based incentives and reputation management is presented in [41] and [42], but a detailed performance analysis is required so as to evaluate its practical feasibility. This paper analyzes both theoretical and simulation based performance of the voting blockchain incorporating relay node selection and incentive mechanism supported by edge computing server. In addition to the mobility constraint in VANETs, the performance analysis also examines the practical feasibility of the proposed solution with varying number of mining and malicious nodes. Based on existing literature, Table II summarizes different consensus algorithms used for various purposes in VANETs and indicates multiple advantages of PBFT including message validation by voting, high throughput and ability to finalize transactions independently without relying RSU.

III. System Modeling and the Proposed Blockchain Design

This section describes PoQF consensus including relay node selection, QF_i calculations, adversary model and incentive distribution mechanism. Key notations used in this paper are listed in Table III. As categorized in [43], we define edge devices present in the network into two types: the mobile edge “nodes”, i.e., vehicles and “edge computing servers”, i.e., RSUs. Before joining the blockchain network, a node needs to register itself and acquire a wallet address and a pair of public and private keys for privacy-preserving communications. This can be accomplished by Vehicle-to-Infrastructure (V2I) communications with regulation authorities via a nearby edge computing server. Each node updates its copy of blockchain through edge caching, as described in [44]. The regulation authorities control the expiration of idle keys, thereby preventing long-range attacks in which attackers use old accounts [45].

A. The Proposed PoQF Consensus

The proposed PoQF consists of four stages, as illustrated in Fig. 1. At the first stage, an incident occurs and a message is initiated by *originator* involved in the incident. An originator

TABLE III: Key Notations.

Notation	Description
$d_{i,j}$	Distance between node i and j
d_{neigh}^{min}	Minimum distance between neighbors nodes
d_{hop}^{min}	Minimum hop distance
n_{hop}	Hop number
n_{th}	Threshold number of votes
n_{tr}	Number of simultaneous transmissions
n_{itf}, n_{neigh}	Number of interference, neighbor nodes
n_{mn}	Number of mining nodes
n_m, n_h	Number of malicious, honest nodes
μ_m, μ_h	Mean number of malicious, honest nodes
p_m	Probability of malicious nodes
p_t	Average transmission probability
p_{suc}, p_{col}	Probability of success, collided transmission
p_{idle}	Probability of node encountering idle slot
τ_i	Validation time of node i
a_{τ_i}, b_{τ_i}	Lower, upper limit of τ_i
t_{icd}	Time at which incident message is received
T_{delay}	Time delay to finalize consensus
T_{slot}	Time slot in MAC layer
T_{suc}, T_{col}	Time for success, collided transmission
T_{DIFS}, T_{SIFS}	Time intervals for DIFS, SIFS
T_{CTS}, T_{ACK}, T_{RTS}	Time intervals for DCF related operations
T_{avg}	Average length of a time slot in DCF
T_{MB}	Time to transmit a microblock
T_{eyp}	Time to encrypt a keyblock
L	Length of a packet
W	Window size
C	Transmission rate
λ_{MAC}, λ_{KB}	MAC, Keyblock throughput
α	Path loss exponent
β	Threshold of SINR
γ	Number of nodes per square meter
R	Transmission range
κ	Consensus parameter
v_i	Velocity of node i
σ_i^2	Variance of v_i
QF_i, DF_i	Quality, Distance Factor of node i
$SINR_{i,j}$	SINR between node i and j
P_{noise}	Noise Power
$Q(SINR_i)$	Quality of node i's SINR
B_i	Behavior of mining node i
CC_{mn}, CC_r	Call Compensation for mining, relay nodes
U_i	Utility of node i
TC	Transaction Charge
FV	Failure in Validation

is the sender s of the message at first hop, i.e., when $n_{hop} = 1$, where n_{hop} is the hop number. The message is analogous to a *transaction proposal* in a consensus that requires validation.

At the second stage, a node which receives and responds to the message performs the role of *mining node*. Each mining node i generates a microblock, in which it records its vote towards validity of the message and its Quality Factor, QF_i, to become a potential $relay\ node$. A node i waits for time τ_i before it announces a microblock. τ_i is a randomly generated number following uniform distribution, i.e., $\tau_i \sim \mathcal{U}(a_{\tau_i}, b_{\tau_i})$, where a_{τ_i} and b_{τ_i} are lower and upper limits of τ_i, respectively, which are dependent on QF_i. The motivation behind using τ_i is three fold: one is to prevent all nodes from transmitting at the same time and causing packet collision, second is to introduce fairness by giving less waiting time to nodes with higher QF_i and the third is to ensure randomization if node i and node j have $QF_i = QF_j$. Using uniform distribution to randomize scheduling of messages so as to avoid packet collision has been previously used in literature [46].

At the third stage, node i is selected as a relay node if it fulfills two conditions. First, it has received at least n_{th} microblocks with the same votes as its own. Second, its QF_i is the highest among all microblocks with the same votes as its own. The motivation behind these two conditions instead of QF_i only is to enhance security of PoQF. For example, if a malicious node i with the highest QF_i among all mining nodes votes false for an originally true message and receives n_{th} microblocks with true votes, it cannot become a relay node and earn incentive. Similarly, if an honest node i with the highest QF_i votes false for an originally false message, but receives n_{th} microblocks with true votes, it cannot become a relay node to forward a false message. A relay node will forward the message only if it is validated as true but always generate a keyblock to record message validity after PoQF and transactions, which are related to incentive distribution. As shown in Fig. 2, if no node receives at least n_{th} microblocks with the same votes as its own until 1s, i.e. the maximum allowable latency for emergency message dissemination [47], the message is considered as false and a keyblock to record such transaction will be generated by the mining node i with highest QF_i, which voted false. If two relay nodes with opposing votes are selected (one with true vote and another with false vote), the message is considered true so that the cooperation may not be stopped in case of a true incident. The value of n_{th} corresponding to real traffic conditions is communicated to nodes by an edge computing server.

The fourth stage is continuation of message dissemination. If the message is validated as true, it is disseminated after a new relay node selection by PoQF at each hop until $n_{hop} \leq n_{hop}^{max}$, where n_{hop}^{max} is the maximum number of hops up to which a message is required to be forwarded and is updated by an edge computing server. It is noted that votes to validate a message are not required for $n_{hop} > 1$. It is simply because the validation of message has been done by adjacent witness nodes (mining nodes at $n_{hop} = 1$) through a camera or location/speed verification [48]. All other nodes beyond the first hop may not have access to validate the originator.

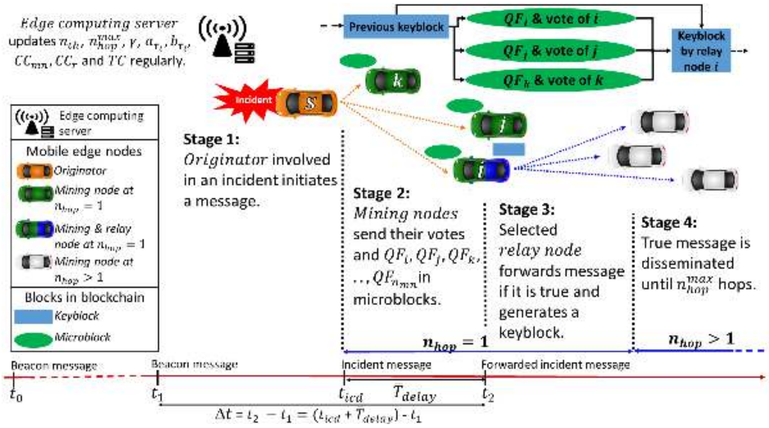

Edge computing server updates n_{th}, n_{hop}^{max}, γ, a_{τ_i}, b_{τ_i}, CC_{mn}, CC_r and TC regularly.
Previous keyblock
QF_i & vote of i
QF_j & vote of j
QF_k & vote of k
Keyblock by relay node i
Incident
S
k
j
i
Edge computing server
Mobile edge nodes
Originator
Mining node at $n_{hop} = 1$
Mining & relay node at $n_{hop} = 1$
Mining node at $n_{hop} > 1$
Blocks in blockchain
Keyblock
Microblock
Stage 1: Originator involved in an incident initiates a message.
Stage 2: Mining nodes send their votes and $QF_i, QF_j, QF_k, .., QF_{n_{mn}}$ in microblocks.
Stage 3: Selected relay node forwards message if it is true and generates a keyblock.
Stage 4: True message is disseminated until n_{hop}^{max} hops.
$n_{hop} = 1$
$n_{hop} > 1$
Beacon message
Beacon message
Incident message
Forwarded incident message
t_0
t_1
t_{icd}
T_{delay}
t_2
$\Delta t = t_2 - t_1 = (t_{icd} + T_{delay}) - t_1$

Fig. 1: The proposed PoQF consensus.

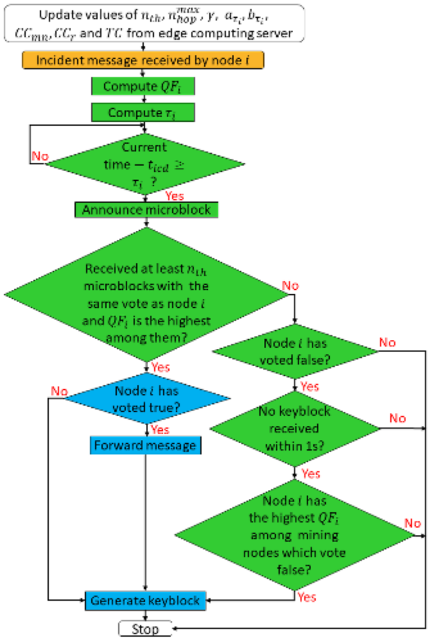

Update values of n_{th}, n_{hop}^{max}, γ, a_{τ_i}, b_{τ_i}, CC_{mn}, CC_r and TC from edge computing server
Incident message received by node i
Compute QF_i
Compute τ_i
Current time − $t_{icd} \geq \tau_i$?
No
Yes
Announce microblock
Received at least n_{th} microblocks with the same vote as node i and QF_i is the highest among them?
No
Yes
Node i has voted false?
No
Yes
Node i has voted true?
No
Yes
Forward message
No keyblock received within 1s?
No
Yes
Node i has the highest QF_i among mining nodes which vote false?
No
Yes
Generate keyblock
Stop

Fig. 2: Flowchart of actions by mining node at $n_{hop} = 1$.

B. QF_i Calculations

QF_i determines the quality of mining node i at the time when it forwards the message as a relay node. Each node regularly shares its position and velocity via beacon message. As shown in Fig. 1, two consecutive beacons messages are exchanged at t_0 and t_1 before the occurrence of incident. To compute QF_i, node i makes probability based predictions of distances with its neighbor nodes at time $t_2 = t_{icd} + \overline{T}_{delay}$, where t_{icd} is the time at which the incident message is received from the sender s (originator or previous relay node) and $\overline{T}_{delay}$ is the mean time delay to finalize consensus and is described in details in Section IV. As QF_i decides the relay node, it is governed by two factors [49]: the probability of success that a node's transmission can reach to all of its neighbor nodes, i.e., Quality of SINR at t_2, $Q(SINR_i^{t_2})$, and the probability that its distance to the sender s is larger than a threshold for ensuring successful transmission over longer distances, i.e., Distance Factor at t_2, $DF_i^{t_2}$. Hence, $QF_i = Q(SINR_i^{t_2}) \cdot DF_i^{t_2}$.

1) $Q(SINR_i^{t_2})$: If node i becomes a relay node, the SINR of a signal received at node j from node i at t_2 is

$$SINR_{i,j} = \frac{(d_{i,j})^{-\alpha}}{P_{noise} + \Sigma_{k=1,k\neq i}^{n_{itf}} (d_{j,k})^{-\alpha}}, \tag{1}$$

where α is the path loss exponent and its value depends on fading environment [16], $d_{i,j}$ is the distance between node i and node j, $d_{j,k}$ is the distance between node j and interfering node k, n_{itf} is the number of interference nodes and P_{noise} is the noise power. For a successful message transmission, it is required that the SINR exceeds a certain threshold β, i.e., $SINR_{i,j} \geq \beta$. The probability that $SINR_{i,j} \geq \beta$ at t_2, i.e., $Pr(SINR_{i,j}^{t_2} \geq \beta)$ is given as

$$\begin{aligned} & Pr(SINR_{i,j}^{t_2} \geq \beta) \\ & = Pr\Big(\frac{(d_{i,j}^{t_2})^{-\alpha}}{P_{noise} + \Sigma_{k=1,k\neq i}^{n_{itf}} (d_{j,k}^{t_2})^{-\alpha}} \geq \beta\Big) \\ & = Pr\Big(d_{i,j}^{t_2} \leq \big(\beta(P_{noise} + \Sigma_{k=1,k\neq i}^{n_{itf}} (d_{j,k}^{t_2})^{-\alpha})\big)^{-\frac{1}{\alpha}}\Big), \end{aligned} \tag{2}$$

where $d_{i,j}^{t_2} = d_{i,j}^{t_1} + \Delta d_{i,j}^{\Delta t}$ is the distance between node i and node j at t_2 and $\Delta d_{i,j}^{\Delta t}$ is the relative distance change between node i and node j during $\Delta t = t_2 - t_1$. $d_{i,j}^{t_1}$ can be obtained from the beacon message received at t_1 and the expected value of $\Delta d_{i,j}^{\Delta t}$ can be found using Probability Density Function (PDF) of standard Gaussian distribution. Referring to the results in [16], [50] and [51], the velocity of a node i follows a standard Gaussian distribution, i.e., $v_i \sim \mathcal{N}(0, \sigma_i^2 t)$, where $\sigma_i^2 = \frac{(v_i^{t_1} - v_i^{t_0})^2}{t_1 - t_0}$ is variance of v_i and $v_i^{t_0}$, $v_i^{t_1}$ denote v_i at t_0 and t_1, respectively, which are shared by node i via beacon messages. $\Delta d_{i,j}^{\Delta t}$ is defined as

$$\Delta d_{i,j}^{\Delta t} = (v_i^{t_1} - v_j^{t_1} + \Delta v_i^{\Delta t} - \Delta v_j^{\Delta t})\Delta t, \tag{3}$$

where $\Delta v^{\Delta t}$ is the change in velocity during Δt. By the principle of linear combination of Gaussian variables, $\Delta v_i^{\Delta t} \sim \mathcal{N}(0, \sigma_i^2 \Delta t)$, $\Delta v_i^{\Delta t} - \Delta v_j^{\Delta t} \sim \mathcal{N}(0, (\sigma_i^2 + \sigma_j^2)\Delta t)$ and hence, $\Delta d_{i,j}^{\Delta t} \sim \mathcal{N}(0, (\sigma_i^2 + \sigma_j^2)\Delta t^3)$. If $v_i^{t_2}$ is not known, (2) can be calculated by assuming $\Delta d_{i,j}^{\Delta t}$ as a standard Gaussian variable.

Each node i calculates (2) with respect to its neighbor node j. As n_{itf} is the number of neighbors of node j except node i, n_{itf} and $d_{j,k}^{t_2}$ are unknown to node i. It can estimate the expected values to find $Pr(SINR_{i,j}^{t_2} \geq \beta)$. Hence $\big(\beta(P_{noise} + \Sigma_{k=1,k\neq i}^{n_{itf}} (d_{j,k}^{t_2})^{-\alpha})\big)^{-\frac{1}{\alpha}}$ in (2) can be rewritten as $\Big(\beta\Big(P_{noise} + E(n_{itf}) E(d_{j,k}^{t_2})^{-\alpha}\Big)\Big)^{-\frac{1}{\alpha}}$, where $E(.)$ denotes expected value. The location of nodes on road is assumed to follow an independent homogeneous spatial Poisson distribution with density parameter γ nodes/m^2 on a two dimensional road segment with no separation of lanes in order to make it general and allow dynamic movement of nodes [2]. Therefore, $E(n_{itf})$ can be estimated as the number of vehicles within the transmission range of node j. Assuming that transmission range of each node is a uniform circular area with radius R, $E(n_{itf})$ can be calculated as the number of nodes inside the area excluding node i, i.e., $E(n_{itf}) = \sum_{k=1}^{\pi R^2\gamma}\left(\frac{(\pi R^2\gamma)^k}{k!}e^{-\pi R^2\gamma}\right) - 1$, where γ is pre-defined and known to each node. It is noted that an adaptive γ corresponding to real traffic conditions is out of the scope of this paper, but can be locally estimated by calculating the number of received beacons [52] or with the use of edge computing servers [53].

Lemma 1: $E(d_{j,k}^{t_2}) = \frac{2}{3R^2}(R^3 - {d_{neigh}^{min}}^3)$, where d_{neigh}^{min} is the minimum allowed distance between two neighbor nodes.

Proof: See Appendix A.1. □

Theorem 1: $Pr(SINR_{i,j}^{t_2} \geq \beta)$

$$= \begin{cases} \frac{1}{2}\left(erf\left(\frac{\Delta d_{i,j}^{\Delta t}}{\sqrt{2(\sigma_i^2+\sigma_j^2)\Delta t^3}}\right) - erf\left(\frac{-\Delta d_{i,j}^{\Delta t}}{\sqrt{2(\sigma_i^2+\sigma_j^2)\Delta t^3}}\right)\right), \\ \hfill \textit{if } d_{i,j}^{t_1} \leq d_x, \\ 1 - \frac{1}{2}\left(erf\left(\frac{\Delta d_{i,j}^{\Delta t}}{\sqrt{2(\sigma_i^2+\sigma_j^2)\Delta t^3}}\right) - erf\left(\frac{-\Delta d_{i,j}^{\Delta t}}{\sqrt{2(\sigma_i^2+\sigma_j^2)\Delta t^3}}\right)\right), \\ \hfill \textit{otherwise,} \end{cases}$$

where $d_x = \left(\beta\left(P_{noise} + E(n_{itf})E(d_{j,k}^{t_2})^{-\alpha}\right)\right)^{-\frac{1}{\alpha}}$.

Proof: See Appendix A.2. □

$Q(SINR_i^{t_2}) = \sum_{j=1}^{j=n_{neigh}} Pr(SINR_{i,j}^{t_2} \geq \beta)$, where n_{neigh} is the number of neighbors of node i whose position and velocities are exchanged through beacon messages.

2) $DF_i^{t_2}$: It is the probability that one hop distance between node i and the sender s is larger than a minimum threshold, d_{hop}^{min}, and is defined as

$$DF_i^{t_2} = Pr(d_{i,s}^{t_2} > d_{hop}^{min}) = 1 - Pr(d_{i,s}^{t_2} \leq d_{hop}^{min}), \quad (4)$$

where $Pr(d_{i,s}^{t_2} \leq d_{hop}^{min})$ can be found by using the same calculation as described in Appendix A.2.

Proposition 1: The range of QF_i is $0 \leq QF_i \leq n_{neigh}$.

Proof: $Q(SINR_i^{t_2})$ is a sum of $Pr(SINR_{i,j}^{t_2})$, for all neighbor nodes of i. Therefore, the possible range of $Q(SINR_i^{t_2})$ is $0 \leq Q(SINR_i^{t_2}) \leq n_{neigh}$. According to (4), the possible range of $DF_i^{t_2}$ is $0 \leq DF_i^{t_2} \leq 1$. As $QF_i = Q(SINR_i^{t_2}) \cdot DF_i^{t_2}$, it can be concluded that $0 \leq QF_i \leq n_{neigh}$. □

C. Adversary Model

It is assumed that all edge computing servers in VEC network are honest. Let n_{mn} be the number of mobile edge nodes taking part as mining nodes in a PoQF consensus out of which n_m nodes are malicious and n_h nodes are honest when $n_{hop} = 1$. A malicious node in the proposed blockchain design is defined as the node voting against the original validity of a message, that is, if a message is true, the malicious node will vote false and vice-versa. Let B_i be the behavior of mining node i. $B_i = 1$ when it is malicious and $B_i = 0$ when it is honest and $n_m = \sum_{i=1}^{n_{mn}} B_i$. B_i follows Binomial distribution, i.e., $B_i \sim \mathcal{B}(n_{mn}, p_m)$, where $p_m \,\epsilon\, [0, 1]$ is the probability that $B_i = 1$. The reason for considering Binomial distribution is because it has only two possible outcomes for a discrete random number [54]. So, we can define one of the outcomes as malicious and another as honest. $\mu_m = p_m n_{mn}$ and $\mu_h = (1 - p_m)n_{mn}$ represent the mean number of malicious and honest nodes, respectively.

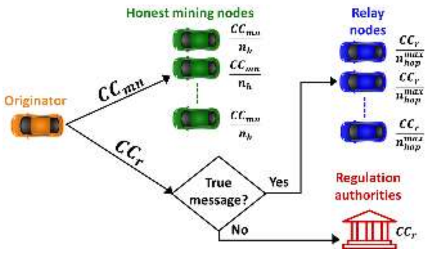


Fig. 3: Distribution of Call Compensation.

D. Incentive Distribution Mechanism

As a compensation of causing an incident, the originator pays a credit known as *Call Compensation*. Assuming that a message is successfully validated, as shown in Fig. 3, Call Compensation, at each direction, consists of CC_{mn}, which is equally distributed among honest mining nodes at $n_{hop} = 1$, and CC_r, which is equally distributed among relay nodes at $n_{hop} = \{1, 2, .., n_{hop}^{max}\}$, in case a message is validated as true. Otherwise, CC_r is transferred to regulation authorities as a penalty charge. If the message is successfully validated, the utility of a mining node i, U_i^{mn}, after taking part in a PoQF consensus at $n_{hop} = 1$ is given as

$$U_i^{mn} = \begin{cases} \frac{CC_{mn}}{n_h}, & \text{if } B_i = 0 \text{ and message is true,} \\ \frac{CC_{mn}}{n_h} - TC, & \text{if } B_i = 0 \text{ and message is false,} \\ -TC, & \text{if } B_i = 1 \text{ and message is true,} \\ 0, & \text{if } B_i = 1 \text{ and message is false,} \end{cases} \quad (5)$$

where $TC > 0$ is the *Transaction Charge* paid by mining node i only when it votes that a message is false. It is later paid to the relay node which generates the last keyblock related to a particular incident. The motivation of introducing TC is to discourage malicious false votes and promote fast dissemination of true message in case of emergency. The values of CC_{mn}, CC_r and TC are updated by edge computing servers. The utility of a relay node, U_i^r, is given as

$$U_i^r = \begin{cases} \frac{CC_r}{n_{hop}^{max}}, & \text{if } n_{hop} \leq n_{hop}^{max} \text{ and message is true,} \\ n_m TC, & \text{if } n_{hop} > n_{hop}^{max} \text{ and message is true,} \\ n_h TC, & \text{if } n_{hop} = 1 \text{ and message is false,} \end{cases} \quad (6)$$

It is worth noting that a mining node i at $n_{hop} = 1$ selected as relay will earn a cumulative utility of $U_i^{mn} + U_i^r$. A relay node records transactions related to U_i^{mn} in the keyblock at $n_{hop} = 1$. For $n_{hop} > 1$, the corresponding relay node records transaction related to U_i^r of previous hop. The message is disseminated until $n_{hop} \leq n_{hop}^{max}$ and PoQF is repeated until $n_{hop} \leq n_{hop}^{max} + 1$, because the last relay node at $n_{hop}^{max} + 1$ records U_i^r of relay node at n_{hop}^{max}. As an incentive, it gains the reward of $n_m TC$ and records this transaction itself. The summary of incentive distribution among mining and relay nodes is shown in Table IV.

IV. Theoretical Performance Analysis

A. Security

1) Failure in Validation: We define the term *Failure in Validation*, FV, as the probability that the original validity

TABLE IV: Incentives distribution among nodes when message is successfully validated.

Incentive	B_i	n_{hop}	True Message	False Message
U_i^{mn}	0	1	$\frac{CC_{mn}}{n_h}$	$\frac{CC_{mn}}{n_h} - TC$
	1	1	$-TC$	0
U_i^r	0	1	$\frac{CC_r}{n_{hop}^{max}}$	n_hTC
		$\leq n_{hop}^{max}$	$\frac{CC_r}{n_{hop}^{max}}$	0
		$> n_{hop}^{max}$	n_mTC	0

of a message is inverted after PoQF consensus at $n_{hop} = 1$. Without loss of generality, we assume the probability that an originator generates a false message, i.e., the originator is malicious, is p_m and the probability of true message generation is $1 - p_m$. Therefore, FV can be expressed as

$$FV = p_mFV_{false} + (1 - p_m)FV_{true}, \tag{7}$$

where FV_{false} and FV_{true} denote Failure in Validation of false and true message, respectively. FV_{false} occurs when a malicious mining node receives at least n_{th} microblocks with malicious votes to mark an originally false message as true. Therefore, FV_{false} can be given as

$$FV_{false} = p_mPr(n_m \geq n_{th}). \tag{8}$$

FV_{true} occurs when an honest mining node does not receive n_{th} microblocks with honest votes to validate an originally true message and can be expressed as

$$FV_{true} = 1 - (1 - p_m)Pr(n_h \geq n_{th}). \tag{9}$$

Bringing (8) and (9) into (7) gives

$$FV = 1 - p_m + p_m^2Pr(n_m \geq n_{th}) - (1 - p_m)^2Pr(n_h \geq n_{th}). \tag{10}$$

Using tail inequalities for Binomial distribution [55], we have the following propositions.

Proposition 2: The upper bound of $Pr(n_x \geq n_{th})$, where $x = m$ or h can be given as

$$Pr(n_x \geq n_{th})^{UB} = \begin{cases} e^{-\frac{(n_{th}-\mu_x)^2}{\mu_x+n_{th}}}, & \textit{if } n_{th} \geq \mu_x, \\ 1, & \textit{otherwise,} \end{cases}$$

Proof: See Appendix B.1. □

Proposition 3: The lower bound of $Pr(n_x \geq n_{th})$, where $x = m$ or h can be given as

$$Pr(n_x \geq n_{th})^{LB} = \begin{cases} 1 - e^{-\frac{(\mu_x-n_{th})^2}{2\mu_x}}, & \textit{if } 0 \leq n_{th} \leq \mu_x, \\ 0, & \textit{otherwise,} \end{cases}$$

Proof: See Appendix B.2. □

By applying Proposition 2 and 3 in (10), the upper and lower bounds of FV, FV^{UB} and FV^{LB} can be derived as

$$\begin{aligned} FV^{UB} &= 1 - p_m + p_m^2Pr(n_m \geq n_{th})^{UB} \\ &- (1 - p_m)^2Pr(n_h \geq n_{th})^{LB}, \end{aligned} \tag{11}$$

$$\begin{aligned} FV^{LB} &= 1 - p_m + p_m^2Pr(n_m \geq n_{th})^{LB} \\ &- (1 - p_m)^2Pr(n_h \geq n_{th})^{UB}. \end{aligned} \tag{12}$$

The expanded forms of (11) and (12) under a varying range of n_{th} can be seen in Appendix B.3.The role of n_{th} in decreasing FV is described in Appendix B.4. Edge computing servers optimize the value of n_{th} for minimizing FV.

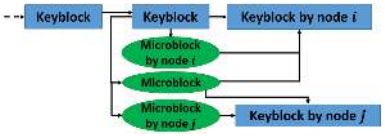


Fig. 4: Potential fork situation.

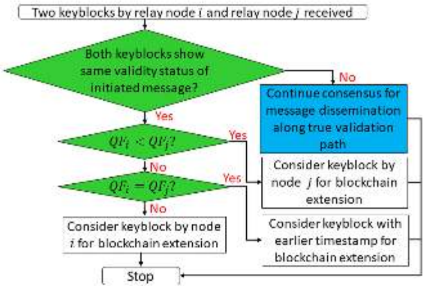

Two keyblocks by relay node i and relay node j received
Both keyblocks show same validity status of initiated message?
No
Continue consensus for message dissemination along true validation path
Yes
QFi < QFj?
Yes
No
QFi = QFj?
Yes
No
Consider keyblock by node j for blockchain extension
Consider keyblock by node i for blockchain extension
Consider keyblock with earlier timestamp for blockchain extension
Stop

Fig. 5: Flowchart of actions to resolve fork.

2) Resolving forks: In the proposed blockchchain, a fork may be created as shown in Fig. 4 when two keyblocks are generated by different relay nodes at the same hop. Forks occur when two or more nodes fulfill both conditions of becoming a relay node, which are defined in Section III. The flowchart of actions by a node in case of fork occurrence is shown in Fig. 5. If the keyblock by relay node i marks the message as false and the keyblock by relay node j marks the message as true, then the message dissemination is continued and new blocks are linked with the keyblock generated by relay node j. If both nodes show the same validity and $QF_i = QF_j$, the timestamps of both keyblocks are checked and the keyblock with the earlier timestamp is considered valid. However, if $QF_i > QF_j$, then new blocks are added in continuation with the keyblock generated by relay node i, regardless of the timestamp of relay node j. The motivation behind selecting the keyblock on the basis of QF_i instead of timestamp for blockchain extension is to discourage a possible cheating attempt by mining node j to become a relay node despite having $QF_j < QF_i$. Cheating by manipulating QF_i is difficult, as it is based on position and velocity of nodes which are shared through regular beacon message exchange and a cheating attempt can be easily detected and reported to concerned authority. In presence of forks, edge computing servers store the longest chain only.

3) Game Theory Analysis of Incentive Distribution Mechanism: We apply the game theory to analyze the impact of the proposed incentive distribution mechanism on actions of mining nodes at $n_{hop} = 1$ and evaluate the security of PoQF against nothing-at-stake and colluding attack by mining nodes.

a) Players: This game has n_{mn} players out of which n_h are honest and n_m are malicious. All players follow PoQF consensus as mining nodes and are located at $n_{hop} = 1$.

b) Action: Every player has two possible actions, honest, H, or malicious, M.

c) Utilities: The payoff matrix in Table V shows (U_i, U_y), if FV does not occur after PoQF at $n_{hop} = 1$.

TABLE V: Payoff Matrix (U_i, U_y), where U_i=Utility of mining node i with the highest QF_i and U_y=Utility of any other mining node at $n_{hop} = 1$.

(a) True Message

		Any other mining node	
		H	M
Mining node i with the highest QF_i	H	$(\frac{CC_{mn}}{n_h} + \frac{CC_r}{n_{hop}^{max}}, \frac{CC_{mn}}{n_h})$	$(\frac{CC_{mn}}{n_h} + \frac{CC_r}{n_{hop}^{max}}, -TC)$
	M	$(-TC, \frac{CC_{mn}}{n_h})$	$(-TC, -TC)$

(b) False Message

		Any other mining node	
		H	M
Mining node i with the highest QF_i	H	$(\frac{CC_{mn}}{n_h} + (n_h - 1)TC, \frac{CC_{mn}}{n_h} - TC)$	$((n_h - 1)TC, 0)$
	M	$(0, \frac{CC_{mn}}{n_h} - TC)$	$(0, 0)$

We present the following analysis of our incentive distribution mechanism.

Lemma 2: Playing honest is the best response action of a mining node, if $CC_{mn} \geq n_h TC$.

Proof: As shown in Table V, if $TC \geq \frac{CC_{mn}}{n_h}$ and the message is false, playing honest will result in $U_y \leq 0$ which will be motivated to play maliciously. On the contrary, if $TC \leq \frac{CC_{mn}}{n_h}$, it makes $U_y \geq 0$ which will motivate the mining nodes to play honestly. Therefore, to make honest as the best response action of mining nodes, it is required that $TC \leq \frac{CC_{mn}}{n_h}$ or $CC_{mn} \geq n_h TC$. □

Proposition 4: The action set (H, H) is both Pareto-optimal and Nash equilibrium and prevents nothing-at-stake attack.

Proof: From the payoff matrix in Table V, we can see that no player can get the maximum utility by deviating from the action set (H, H), provided that Lemma 2 is fulfilled. In both true and false message cases, all mining nodes can get the highest payoff by playing honestly only. Therefore, the action set (H, H) is both Pareto-optimal and Nash equilibrium of this game. The utilities of players will be at risk by playing maliciously, and therefore they will not be motivated to generate a keyblock without message validation which happens in nothing-at-stake attack [45]. □

Theorem 2: A mining node cannot increase its expected utility sum by colluding with its malicious neighbors if $n_h TC \leq CC_{mn} \leq \frac{n_m CC_r}{n_{hop}^{max}(n_h - n_m)}$ and $p_m \leq 0.5$.

Proof: See Appendix C. □

Thus the incentive distribution mechanism is collusion resistant if edge computing servers adjust the values of CC_{mn}, CC_r and n_{hop}^{max} such that Theorem 2 is fulfilled.

B. Validation Latency and Throughput

The MAC throughput in bit/second is defined in [56] as $\lambda_{MAC} = p_t \cdot p_{suc} \cdot \frac{L}{T_{avg}}$, where $p_t = \frac{2}{W+2}$ is the average transmission probability of a node, W is the contention window size, p_{suc} is the probability of success transmission, L is the average length of a packet and T_{avg} is the average length of a time slot in Distributed Coordination Function (DCF). $p_{suc} = n_{tr} \cdot p_t \cdot (1 - p_t)^{n_{tr}-1}$, where n_{tr} is the number of nodes contending the channel for transmission. According to IEEE 802.11 standard [56], T_{avg} is

$$T_{avg} = p_{idle} \cdot T_{slot} + p_{suc} \cdot T_{suc} + p_{col} \cdot T_{col}, \tag{13}$$

where T_{slot} is the unit time slot in DCF scheme, $p_{idle} = (1 - p_t)^{n_{tr}}$ and $p_{col} = 1 - p_{idle} - p_{suc}$ are the probabilities of a node encountering an idle slot and collided transmission respectively, T_{suc} and T_{col} are average time for success and collided transmission respectively and are given as

$$T_{col} = T_{RTS} + T_{DIFS} + T_{slot}, \tag{14}$$

$$\begin{aligned} T_{suc} = T_{RTS} + T_{DIFS} + T_{CTS} + T_{ACK} \\ +3T_{SIFS} + 4Tslot + \frac{L}{C}, \end{aligned} \tag{15}$$

where T_{DIFS} and T_{SIFS} are time intervals for DCF Interframe space (DIFS) and Short Interframe Space (SIFS) respectively, T_{RTS}, T_{CTS} and T_{ACK} are pre-specified time intervals reserved for DCF related operations and C is the average transmission rate among nodes. As λ_{MAC} is defined in bit/second, the average time consumption, T_{MB}, to successfully transmit a vote in a microblock of length L bits is

$$T_{MB} = \frac{L(bits)}{\lambda_{MAC}(bit/second)} = \frac{T_{avg}}{p_t \cdot p_{suc}}. \tag{16}$$

T_{MB} varies with n_{tr} only if W, L, T_{slot}, T_{col} and T_{suc} are considered as fixed value for all transmitting mining nodes. As $1 \leq n_{tr} \leq n_{mn}$, we consider two boundaries for T_{MB}, i.e., T_{MB}^1 with $n_{tr} = 1$ and $T_{MB}^{n_{mn}}$ with $n_{tr} = n_{mn}$. Therefore, $T_{MB}^{min} = min(T_{MB}^1, T_{MB}^{n_{mn}})$ and $T_{MB}^{max} = max(T_{MB}^1, T_{MB}^{n_{mn}})$. Considering a fixed transmission range and a homogeneous distribution for all nodes, we can assume that n_{neigh} is statistically the same for every node. According to Proposition 1, $n_{neigh} - QF_i$ can be considered as the ranking of mining node i to announce its microblock. In this way, node i with a large QF_i can have less validation time before announcing a microblock. An edge computing server provides τ_i bounds to be followed by mining nodes by considering T_{MB} as the time required by a mining node to successfully transmit a microblock. The lower bound of τ_i is given as $a_{\tau_i} = T_{MB}^{min}(n_{neigh} - QF_i)$ and the upper bound of τ_i is given as $b_{\tau_i} = T_{MB}^{max}(n_{neigh} - QF_i)$.

For n_{mn} microblocks, the total time consumption (or validation latency), T_{delay}, can be in the range $T_{MB}^{min} \cdot n_{mn} \leq T_{delay} \leq T_{MB}^{max} \cdot n_{mn}$. A mining node i with the highest QF_i becomes a relay node as soon as it receives at least n_{th} microblocks and does not need to wait for receiving all n_{mn} microblocks. Therefore, T_{delay} is reduced for small n_{th} and we can find lower and upper bounds of T_{delay} when a message is successfully validated by an honest node at $n_{hop} = 1$.

Proposition 5: $T_{delay}^{LB} = T_{MB}^{min} \cdot n_{th}$.

Proof: T_{delay} is the minimum when a relay node receives first n_{th} consecutive microblocks with same votes immediately following the incident message. □

Proposition 6: $T_{delay}^{UB} = T_{MB}^{max} \cdot (n_{th} + p_m n_{mn})$.

Proof: The maximum number of microblocks with malicious votes is $p_m n_{mn}$. T_{delay} will be the maximum if an honest relay node receives all $p_m n_{mn}$ microblocks before receiving n_{th} honest microblocks. □

A keyblock is generated by a relay node after the message validation. Therefore, the throughput in terms of number of keyblocks generated per second can be estimated as

$$\lambda_{KB} = \frac{1}{T_{delay} + T_{eyp}}, \tag{17}$$

where T_{eyp} is the time required to encrypt a keyblock.

C. Asymptotic Complexities

In this subsection, we compare the scalability of various consensus algorithms by analyzing latency complexity, i.e., the time consumption required to confirm a transaction, security complexity, i.e., the minimum number of malicious nodes to control consensus, and communication complexity, i.e., the number of exchange messages required to validate a transaction. Without loss of generality, we derive the asymptotic latency, security and communication complexity of various consensus algorithms in Table VI in terms of number of nodes participating in mining competition, n_{mn} and consensus parameter, κ, which is unique to each algorithm. κ refers to the difficulty level of hash puzzle in PoW, synchronization level in PoS, waiting time in PoET and number of minimum votes required in voting based algorithms (PBFT and PoQF). Standard mathematical notations are used in Table VI, i.e., $\Omega(.)$, $O(.)$ and $\Theta(.)$ denote the order of *at least*, *at most* and *exactly* respectively. Table VI shows that κ affects the latency in PoW, PoS and PoET. Despite the fast consensus of PoS, a strong synchronization among edge computing resources is needed for efficient running [58]. Latency of PoET depends on the length of waiting time which follows a fixed probability distribution. PoQF has to wait for a threshold number of votes, which has an impact on latency but its scalability does not rely on large computation power or storage capacity of mobile edge nodes. Similar to PoS, synchronization among edge computing servers and mobile edge nodes is needed in PoQF, but the requirement is independent of n_{mn}. PoET offers the least security and can be controlled by only a small fraction of malicious nodes [9]. According to Theorem 3, PoQF is secure against the collusion attack when $p_m \leq 0.5$. It provides the same security as PoW which is better than PBFT but worse than PoS [59]. In communication complexity, PoW, PoS and PoET are more efficient than PoQF, since they do not require multiple message exchanges. Despite the voting nature of PoQF, it has lower communication complexity than PBFT. Moreover, in VANETs, n_{mn} cannot be increased beyond a certain threshold due to limited number of nodes within a transmission range R and d_{min}^{neigh}, which makes PoQF scalable and applicable in V2V communications.

TABLE VI: Comparison of Asymptotic Complexities.

Consensus	Latency	Security	Communication
PoW	$\Theta(\kappa)$[58]	$\Omega\left(\frac{n_{mn}}{2}\right)$[60]	$\Theta(1)$[58]
PoS	$\Omega(\kappa)$[58]	$\Omega\left(\frac{2n_{mn}}{3}\right)$[59]	$\Theta(1)$[58]
PoET	$\Omega(\kappa)$[9]	$\Omega\left(\frac{loglogn_{mn}}{logn_{mn}}\right)$[9]	$\Theta(1)$[9]
PBFT	$n_{mn}O(1)$[58]	$\Omega\left(\frac{n_{mn}-1}{2}\right)$[35]	$O(n_{mn}^2)$[10]
PoQF	$\kappa O(1)$	$\Omega\left(\frac{n_{mn}}{2}\right)$	$O(n_{mn})$

TABLE VII: Simulation Parameters.

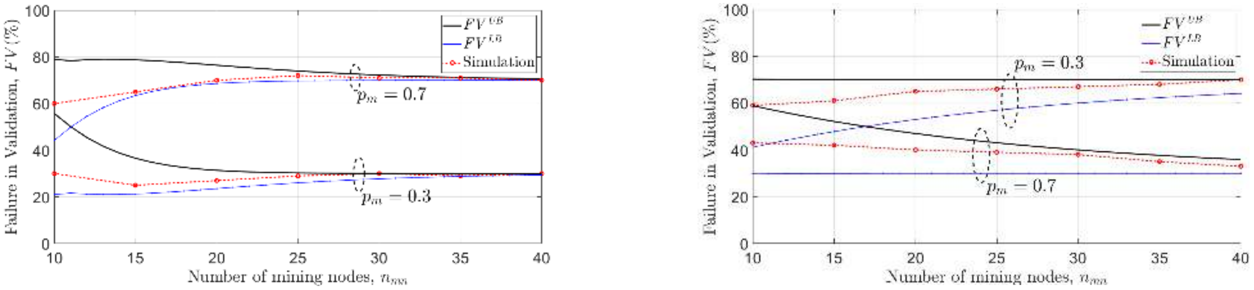
Parameters	Values	Parameters	Values
Simulation Time	200 s	Protocol	IEEE 802.11p
Size of area	10 km × 10 km	Encryption	SHA-256
Beacon frequency	0.1 s	P_{noise}	-99 dBm
γ	50, 75, 100, 125	R	250 m
	150, 175, 200	α	3
	nodes/km^2	β	8 dB
Mobility model	Freeway	d_{neigh}^{min}	12 m
Average velocity	40 km/hr	d_{hop}^{min}	100 m
L	756 bytes	W	32
T_{RTS}	53 μs	T_{DIFS}	58 μs
T_{CTS}	37 μs	T_{SIFS}	32 μs
T_{ACK}	37 μs	T_{slot}	13 μs
T_{eyp}	3332.11 μs	C	6 Mbps

V. Simulation Results

In this section, we analyze the performance of the proposed blockchain and PoQF consensus using OMNeT++ integrated with SUMO (Simulation of Urban Mobility) [1]. The simulation parameters listed in Table VII align with other VANET applications [15], [56], [57], [61]. Since n_{mn} are neighbor nodes of a sender s, $n_{mn} \leq 40$ will be considered when nodes are homogeneously distributed with a maximum of 200 nodes/km^2 and it is a reasonable assumption of maximum number of vehicles within a transmission range when the safe distance between nodes are maintained on road [62]. Evaluation results are averaged over 100 simulation runs.

Fig. 6 shows FV with respect to n_{mn} at different p_m and n_{th}. Two different values of p_m are chosen to show the results at both low ($p_m < 0.5$) and high ($p_m > 0.5$) densities of malicious mining nodes presented in the network. As shown in Fig. 6 (a), FV with $p_m = 0.3$ is lower than FV with $p_m = 0.7$ when $n_{th} = 3$, i.e., $n_{th} \leq \mu_m$. It shows that a low n_{th} is suitable only for low p_m when honest nodes are in majority. On the contrary, as shown in Fig. 6 (b), FV with $p_m = 0.3$ is higher than FV with $p_m = 0.7$ when $n_{th} = n_{mn}$, i.e., $n_{th} > \mu_m$. This is because when $n_{th} = n_{mn}$ both malicious and honest mining nodes are unable to finalize consensus within the maximum allowable latency of 1s and the message is marked as false. With $p_m = 0.3$, the probability of false message occurrence is lower than that of true message occurrence and it is hard to collect $n_{th} = n_{mn}$ honest votes to validate a true message. In this case, $FV^{UB} \approx 1 - p_m$ depicts the worst case scenario of maximum probability of true message generation which will be marked as false. With $p_m = 0.7$, the probability of true message occurrence is lower than that of false message occurrence. FV does not occur when both honest and malicious nodes are unable to collect

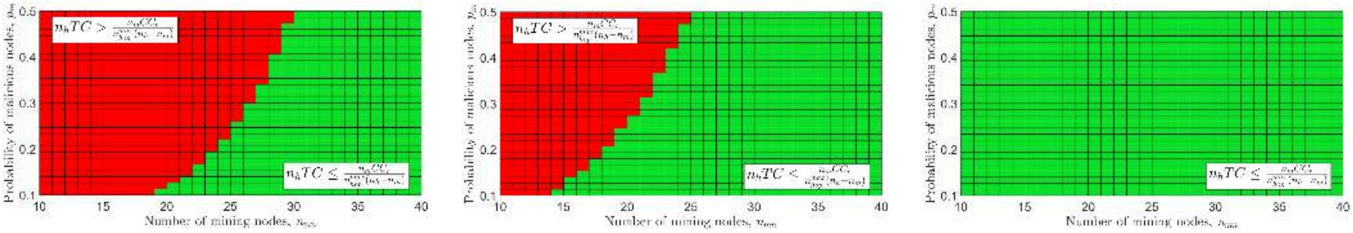
[1] Source code is available at https://github.com/ferheenayaz/PoQF.



(a) $n_{th} = 3$

(b) $n_{th} = n_{mn}$

Fig. 6: FV with respect to n_{mn}.

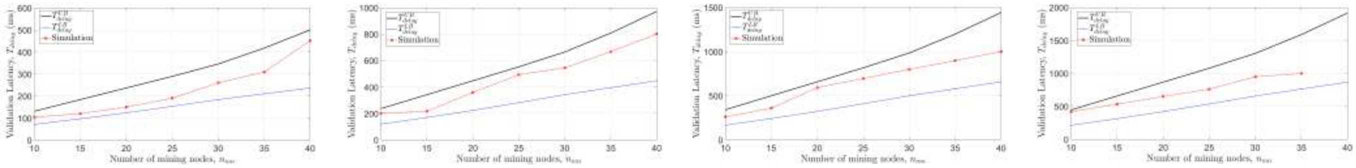


(a) $CC_r = 100$, $TC = 0.5$, $n_{hop}^{max} = 10$

(b) $CC_r = 200$, $TC = 0.5$, $n_{hop}^{max} = 10$

(c) $CC_r = 200$, $TC = 0.1$, $n_{hop}^{max} = 6$

Fig. 7: Setting up CC_r, TC and n_{hop}^{max} for collusion resistant incentive distribution mechanism.



(a) $p_m = 0.2$

(b) $p_m = 0.4$

(c) $p_m = 0.6$

(d) $p_m = 0.8$

Fig. 8: T_{delay} with respect to n_{mn}.

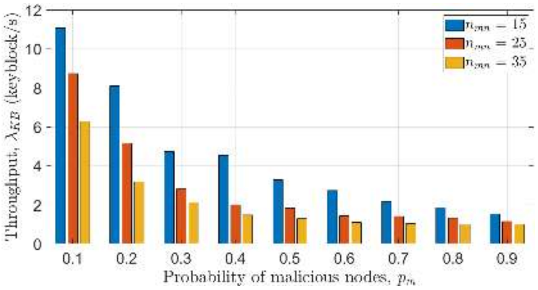
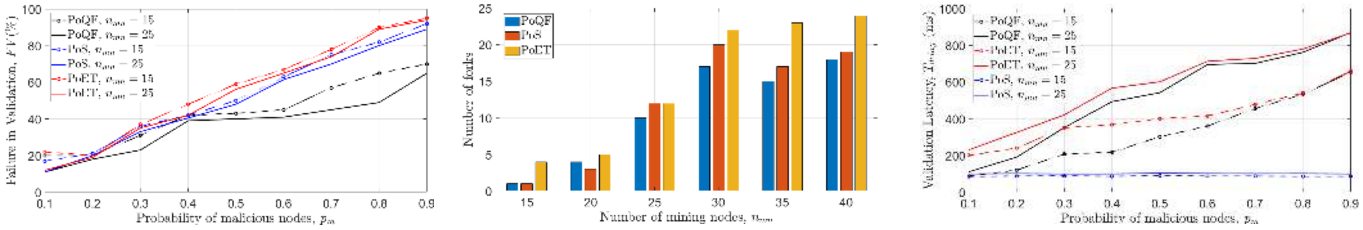


Fig. 9: λ_{KB} with respect to p_m.

votes for a false message. It only occurs when a true message is not validated. As shown in Fig. 6 (b), $FV^{LB} \approx 1 - p_m$ with $p_m = 0.7$, depicts the percentage of true messages which are not validated by PoQF. The dependence of n_{th} on p_m is further discussed in Appendix B.4. If p_m in the network is known, VEC technique can achieve a low FV even with high values of p_m by adjusting n_{th} accordingly.

Fig. 7 shows the impact of parameters: CC_r, TC and n_{hop}^{max}, on the collusion resistance feature of incentive distribution mechanism. According to Theorem 3, the incentive distribution mechanism is collusion resistant if $n_hTC \leq CC_{mn} \leq \frac{n_mCC_r}{n_{hop}^{max}(n_h-n_m)}$ and $p_m \leq 0.5$. As B_i follows Binomial distribution, it can be assumed that $n_m \approx \mu_m$ and $n_h \approx \mu_h$. Based on this assumption, Fig. 7 (a) and (b) show that the incentive distribution mechanism cannot be collusion resistant for every n_{mn}, p_m under the fixed CC_r, TC and n_{hop}^{max}. However, in Fig. 7 (c), when $CC_r = 200$, $TC = 0.1$ and $n_{hop}^{max} = 6$, $n_hTC \leq \frac{n_mCC_r}{n_{hop}^{max}(n_h-n_m)}$ is satisfied for every $n_{mn}\epsilon(10, 40)$ and $p_m\epsilon(0, 0.5)$. Therefore, for a collusion resistant incentive distribution mechanism, it is required that the edge computing servers should adjust the combination of these parameters with varying n_{mn} and p_m, such that it is possible to choose CC_{mn} within the boundaries defined by Theorem 3. Apart from the security reason, a low n_{hop}^{max} is also favorable for successful message delivery, as the failure of multi-hop connectivity in VANETs increases with the number of hops [63].

Fig. 8 shows T_{delay} of successful message validation with respect to n_{mn} at different values of p_m with $n_{th} = \mu_m + 1$. It can be seen that T_{delay} increases with p_m because of more frequent generation of microblocks by malicious nodes. Fig. 8 (c) and (d) show that T_{delay}^{UB} exceeds the maximum allowable latency requirement of 1s [47] when $p_m \geq 0.6$ and $n_{mn} \geq 30$. At $p_m = 0.8$ and $n_{mn} > 35$, the mining nodes are unable to finalize consensus within 1s. Fig. 9 shows λ_{KB} of PoQF consensus at various n_{mn} and p_m. The highest



(a) FV

(b) Number of forks

(c) T_{delay}

Fig. 10: Comparison of PoQF with PoS and PoET with $n_{th} = \mu_m + 1$.

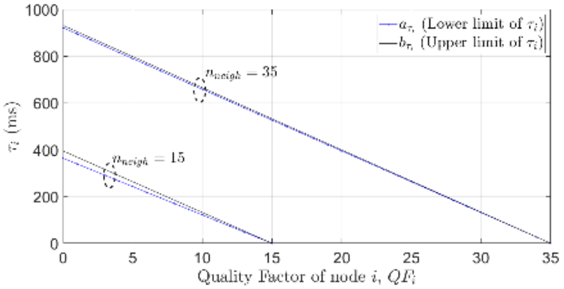


Fig. 11: Values of a_{τ_i} and b_{τ_i}.

λ_{KB} achieved is 11 keyblock/s at $p_m = 0.1$ and $n_{mn} = 15$ and the lowest is 0.9 keyblock/s at $p_m \geq 0.8$ and $n_{mn} = 35$, which means that at higher n_{mn} and p_m, PoQF with $\lambda_{KB} < 1$ keyblock/s may not be able to generate block within the limit of maximum allowable latency of 1s. This shows that our proposed blockchain exhibits better performance specifically at lower values of p_m and n_{mn}. λ_{KB} can be improved by offloading computations required for encrypting a keyblock to a nearby edge computing server which have high computation power, thereby reducing T_{eyp}, as suggested in [64].

Fig. 10 compares the performance of PoQF with PoET and PoS. In Fig. 10 (a), FV of PoQF is compared with PoET and PoS at different values of p_m and n_{mn}, while n_{th} is fixed at $\mu_m + 1$, as it results in low FV for all values of p_m. We implement PoET such that its waiting time is uncontrolled by VEC. Each node generates a random number between 0 to 1 s to determine its waiting time for collecting microblocks. It shows that FV of PoQF and PoET are closing to each other at low p_m. For high values of p_m, an honest node i with the highest QF_i is unable to collect sufficient microblocks from honest mining nodes within a random waiting time of PoET and therefore its FV rises with p_m at a higher rate than PoQF. In the reputation based PoS, a node is considered honest if its reputation exceeds a certain threshold. We randomly assign a reputation value to nodes on a scale of 0 to 100, thus the probability of reputation falling below a threshold of 50 is defined as p_m. We implement PoS such that a malicious relay node only forwards the message from a malicious sender and an honest relay node only forwards the message from an honest sender. On an average, PoQF reduces FV by 11% and 15%, as compared to PoS and PoET, respectively. Fig. 10 (b) compares the number of forks created by PoQF, PoS and PoET consensus. Although solutions to resolve forks are discussed in Section IV, a blockchain consensus should be able to avoid creation of forks in order to control discrepancies. In PoQF, node i with the highest QF_i is most likely to announce its microblock prior to other mining nodes. In this way, node j with $QF_j < QF_i$ cannot become a relay node, if votes of both nodes are the same. This is how creation of forks is reduced in the proposed consensus. We implement PoS consensus by selecting a relay node on the basis of the highest reputation which is randomly generated from 0 to 100 in the simulation. A fork appears when two nodes with same reputation simultaneously become relay node. In PoET, the time to announce microblock is not controlled by VEC. Therefore, node j with lower QF_j becomes a relay node before receiving a microblock from node i even though $QF_i > QF_j$. In that case, a fork appears if both node i and node j generate keyblocks. It is noted that the number of forks in PoS is equal or lower than PoQF when $n_{mn} \leq 20$. Due to unreliable nature of vehicle connectivity, there remains a possibility of fork occurrence when an announced microblock by node i is not received by node j. It usually happens when mining nodes are in distance and beyond the transmission range of each other. This is why a low node density, ultimately leading to low n_{mn}, may result in a higher or equal number of forks created by PoQF as compared to PoS. Fig. 10 (c) compares T_{delay} of successful message validation consumed by PoQF, PoET and PoS. By using PoET, the mining node i is allowed to announce its microblock at a random time irrespective of its QF_i. On an average, T_{delay} of PoET is 68 ms higher than PoQF. However, the difference is larger at lower p_m. Since τ_i is independent of QF_i in PoET, node j with lower QF_j may announce its microblock earlier than node i with $QF_i > QF_j$ and node i might have to wait longer. This waiting time is reduced in PoQF by utilization of VEC. However, with large p_m, an honest mining node i with the highest QF_i has to wait longer in PoQF for receiving n_{th} honest microblocks. It is because the frequency of malicious microblocks generation is increased with a large p_m. Hence the T_{delay} difference between PoQF and PoET becomes smaller. T_{delay} of PoS is independent of p_m and increases with n_{mn}. It is the smallest because it only consumes time in relay node selection, while the voting time is eliminated by the reputation based message validation.

Fig. 11 displays a_{τ_i} and b_{τ_i} which are governed by edge computing servers to regulate τ_i, generated by a mining node i. It shows that a_{τ_i} and b_{τ_i} reduce with an increasing QF_i and therefore τ_i leads to less waiting time for potential relay nodes. Due to homogeneous distribution of nodes, n_{neigh} is

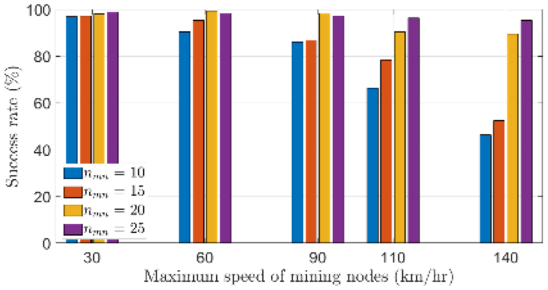


Fig. 12: Average success rate with respect to speed.

the same for every node in a network. Therefore, deliberately reducing τ_i by node i is bounded by the limit of a_{τ_i}, which is known to every node in a network. Such attempt can be easily detected and reported to concerned authority.

Fig. 12 shows the success rate of transmitting a true message under different maximum speeds of a mining node. For $n_{hop} > 1$, PoQF consensus is only used for relay node selection since the message is already validated at $n_{hop} = 1$, and therefore, the transmission success rate is independent of p_m. It shows that the success rate is falling with increasing speed, specifically for small n_{mn}. This is because a small n_{mn} depicts a low traffic density, so the nodes are likely to attain their maximum speeds and may lose connectivity before finalizing a consensus to select a relay node. In order to speed up consensus, a possible solution is that the edge computing servers reduce n_{th} when $n_{hop} > 1$. At $n_{hop} > 1$, the consensus only depends on the highest QF_i and does not require message validation. Since the mining node i sends its microblock earlier than the mining node j with $QF_j < QF_i$, it is not necessary for the mining node i to wait until QF_j is received. It is noted that in case of an incident or traffic jam, a high speed is not likely to be attained in the affected area and therefore, it is not recommended to reduce n_{th} at $n_{hop} = 1$, as it may result in large FV.

VI. Conclusion

In this paper, we have proposed a blockchain based on PoQF consensus algorithm for message dissemination in VEC networks. The theoretical performance of proposed consensus is evaluated by deriving bounds on failure and latency in message validation, throughput of block generation and asymptotic latency, security and communication complexity. Moreover, an incentive distribution mechanism to promote positive cooperation and discourage malicious behavior of nodes has been presented and analyzed using game theory.

From the simulation analysis, the proposed blockchain shows 11% reduction in FV by PoQF as compared to reputation based PoS. As a trade-off, it results in increased validation latency. Specifically due to VEC, PoQF is 15% more secure and 68 ms faster in validating a message as compared to PoET. Furthermore, PoQF results in less number of forks than PoET and PoS. Similar to PoW, PoQF is vulnerable to malicious nodes if they compose more than 50% of mining group but its performance is not dependent on the presence of at least $2f + 1$ mining nodes as in PBFT.

In the future work, we aim to reduce latency by proposing an alternative to voting solution for message validation. An adaptive and intelligent blockchain can be designed to achieve higher throughput with varying number of mining nodes.

Appendix A

1) Appendix A.1: The PDF of interference nodes at location (X, Y) within the area πR^2 is defined in [14] as $\frac{1}{\pi R^2}$. Therefore, $E(d_{j,k}^{t_2})$ can be calculated as

$$E(d_{j,k}^{t_2}) = \int (X^2 + Y^2) f_{(X,Y)} dX dY. \tag{18}$$

Bringing $X = z cos\phi$ and $Y = z sin\phi$ into (18) leads to

$$E(d_{j,k}^{t_2}) = \int_{d_{neigh}^{min}}^{R} \int_{0}^{2\pi} \frac{z^2}{\pi R^2} d\phi dz = \frac{2}{3R^2}(R^3 - {d_{neigh}^{min}}^3). \tag{19}$$

2) Appendix A.2: Since $d_{i,j}^{t_2} = d_{i,j}^{t_1} + \Delta d_{i,j}^{\Delta t}$, we find the probability of $\Delta d_{i,j}^{\Delta t} \leq d_x - d_{i,j}^{t_1}$. If $d_{i,j}^{t_1} \leq d_x$, the actual required communication distance, $\Delta d_{i,j}^{\Delta t}$ can be calculated as in [50]

$$\Delta d_{i,j}^{\Delta t} = \begin{cases} d_x + d_{i,j}^{t_1}, & \textit{Case 1} \\ d_x - d_{i,j}^{t_1}, & \textit{Case 2} \end{cases} \tag{20}$$

where *Case 1* is either of the following

- node i and node j are moving towards each other
- node i is in front of node j, both moving in same direction, and $v_i < v_j$
- node j is in front of node i, both moving in same direction, and $v_i > v_j$

and *Case 2* is either of the following

- node i and node j are moving away from each other
- node i is in front of node j, both moving in same direction, and $v_i > v_j$
- node j is in front of node i, both moving in same direction, and $v_i < v_j$

Therefore, PDF of $\Delta d_{i,j}^{\Delta t}$ can be defined as

$$f(\Delta d_{i,j}^{\Delta t}) = \frac{1}{\sqrt{2\pi(\sigma_i^2 + \sigma_j^2)\Delta t^3}} e^{-\frac{(\Delta d_{i,j}^{\Delta t})^2}{2(\sigma_i^2 + \sigma_j^2)\Delta t^3}}. \tag{21}$$

Consider both acceleration and deceleration, Cumulative Density Function (CDF) can be calculated as

$$F(\Delta d_{i,j}^{\Delta t}) = \int_{-\Delta d_{i,j}^{\Delta t}}^{\Delta d_{i,j}^{\Delta t}} f(\Delta d_{i,j}^{\Delta t}) d(\Delta d_{i,j}^{\Delta t}), \tag{22}$$

As $F(\Delta d_{i,j}^{\Delta t}) = Pr(d_{i,j}^{t_2} \leq d_x) = Pr(SINR_{i,j}^{t_2} \geq \beta)$,

$$\begin{aligned} Pr(SINR_{i,j}^{t_2} \geq \beta) = \frac{1}{2}\Bigg[& erf\left(\frac{\Delta d_{i,j}^{\Delta t}}{\sqrt{2(\sigma_i^2 + \sigma_j^2)\Delta t^3}}\right) \\ & - erf\left(\frac{-\Delta d_{i,j}^{\Delta t}}{\sqrt{2(\sigma_i^2 + \sigma_j^2)\Delta t^3}}\right)\Bigg]. \end{aligned} \tag{23}$$

Otherwise, if $d_{i,j}^{t_1} > d_x$, the actual required communication distance $\Delta d_{i,j}^{\Delta t}$ can be calculated as

$$\Delta d_{i,j}^{\Delta t} = \begin{cases} d_{i,j}^{t_1} - d_x, & \textit{Case 1} \\ d_{i,j}^{t_1} + d_x, & \textit{Case 2} \end{cases} \tag{24}$$

where *Case 1* and *Case 2* are the same as defined in (20). As $d_{i,j}^{t1} > d_x$, for $d_{i,j}^{t_2} \leq d_x$, we need $\Delta d_{i,j}^{\Delta t} < 0$. Therefore, we calculate $1 - f(\Delta d_{i,j}^{\Delta t})$ and ultimately $Pr(d_{i,j}^{t_2} \leq d_x) = Pr(SINR_{i,j}^{t_2} \geq \beta)$ is expressed as

$$Pr(SINR_{i,j}^{t_2} \geq \beta) = 1 - \frac{1}{2}\left[erf\left(\frac{\Delta d_{i,j}^{\Delta t}}{\sqrt{2(\sigma_i^2 + \sigma_j^2)\Delta t^3}}\right) - erf\left(\frac{-\Delta d_{i,j}^{\Delta t}}{\sqrt{2(\sigma_i^2 + \sigma_j^2)\Delta t^3}}\right)\right]. \tag{25}$$

Appendix B

1) Appendix B.1: According to the multiplicative form of Chernoff bound [55], $Pr(X \geq (1+\delta)\mu) \leq e^{-\frac{\delta^2 \mu}{2+\delta}}$, where X is a sum of independent Binomial variables with mean μ and $\delta > 0$. Bringing $\mu = \mu_x$ and $(1+\delta)\mu = n_{th}$, gives

$$Pr(n_x \geq n_{th}) \leq \begin{cases} e^{-\frac{(n_{th}-\mu_x)^2}{\mu_x + n_{th}}}, & \textit{if } n_{th} \geq \mu_x, \\ 1, & \textit{otherwise.} \end{cases} \tag{26}$$

2) Appendix B.2: For $0 \leq \delta \leq 1$, Chernoff bound [55] states that $Pr(X \leq (1-\delta)\mu) \leq e^{-\frac{\delta^2 \mu}{2}}$, which can be rewritten as $Pr(X \geq (1-\delta)\mu) \geq 1 - e^{-\frac{\delta^2 \mu}{2}}$. Therefore,

$$Pr(n_x \geq n_{th}) \geq \begin{cases} 1 - e^{-\frac{(\mu_x - n_{th})^2}{2\mu_x}}, & \textit{if } 0 \leq n_{th} \leq \mu_x, \\ 0, & \textit{otherwise.} \end{cases} \tag{27}$$

3) Appendix B.3: Using (10), (26) and (27), we get $FV \leq$

$$\begin{cases} 1 - p_m + p_m^2 - (1-p_m)^2(1 - e^{-\frac{(\mu_h - n_{th})^2}{2\mu_h}}), \\ \qquad\qquad \textit{if } n_{th} < min(\mu_m, \mu_h), \\ 1 - p_m + p_m^2 e^{-\frac{(n_{th}-\mu_m)^2}{\mu_m + n_{th}}} - (1-p_m)^2(1 - e^{-\frac{(\mu_h - n_{th})^2}{2\mu_h}}), \\ \qquad\qquad \textit{if } \ \mu_m \leq n_{th} \leq \mu_h, \\ 1 - p_m + p_m^2, \qquad \textit{if } \ \mu_h < n_{th} < \mu_m, \\ 1 - p_m + p_m^2 e^{-\frac{(n_{th}-\mu_m)^2}{\mu_m + n_{th}}}, \qquad \textit{if } n_{th} > max(\mu_m, \mu_h). \end{cases} \tag{28}$$

and $FV \geq$

$$\begin{cases} 1 - p_m + p_m^2(1 - e^{-\frac{(\mu_m - n_{th})^2}{2\mu_m}}) - (1-p_m)^2, \\ \qquad\qquad \textit{if } n_{th} < min(\mu_m, \mu_h), \\ p_m - p_m^2, \qquad \textit{if } \ \mu_m \leq n_{th} \leq \mu_h, \\ 1 - p_m + p_m^2(1 - e^{-\frac{(\mu_m - n_{th})^2}{2\mu_m}}) - (1-p_m)^2 e^{-\frac{(n_{th}-\mu_h)^2}{\mu_h + n_{th}}}, \\ \qquad\qquad \textit{if } \ \mu_h < n_{th} < \mu_m, \\ 1 - p_m - (1-p_m)^2 e^{-\frac{(n_{th}-\mu_h)^2}{\mu_h + n_{th}}}, \textit{if } n_{th} > max(\mu_m, \mu_h). \end{cases} \tag{29}$$

4) Appendix B.4: As we know that, $0 < e^{-x} \leq 1$ for any real valued x and $p_m \epsilon [0,1]$, it can be deduced from (29) that FV^{LB} is the minimum when $\mu_m \leq n_{th} \leq \mu_h$, which is only possible for $p_m \leq 0.5$. For $p_m > 0.5$, the minimum FV^{LB} can be obtained when $n_{th} > \mu_m$. To find the minimum FV^{UB}, we compare its value at two conditions of (28), i.e., $n_{th} > max(\mu_m, \mu_h)$ and $n_{th} < min(\mu_m, \mu_h)$.

$$1 - p_m + p_m^2 e^{-\frac{(n_{th}-\mu_m)^2}{\mu_m + n_{th}}} < 1 - p_m + p_m^2 - (1-p_m)^2(1 - e^{-\frac{(\mu_h - n_{th} - 1)^2}{2\mu_h}}), \tag{30}$$

Assuming that $p_m^2 e^{-\frac{(n_{th}-\mu_m)^2}{\mu_m + n_{th}}} \approx (1-p_m)^2(1 - e^{-\frac{(\mu_h - n_{th} - 1)^2}{2\mu_h}}) \approx 0$, (30) leads to $p_m > \frac{1}{2}$. It proves that $n_{th} > \mu_m$ results in the minimum FV^{UB} for $p_m > \frac{1}{2}$.

Appendix C

Let n_{cp} colluding players form a group to mark a true message as false or a false message as true with a probability p_{cp}. The expected utility sum of colluding players as mining nodes, $E(U_{cp}^{mn})$, if they mark a true message as false is

$$E(U_{cp}^{mn}) = p_{cp}\left(\frac{CC_{mn}}{n_m} n_{cp} - n_{cp} TC\right) + (1 - p_{cp})\left(\frac{CC_{mn}}{n_h} n_{cp}\right). \tag{31}$$

The probability that one of the colluding players is selected as a relay node if the colluding attack is successful is n_{cp}/n_m and if colluding players play honestly is n_{cp}/n_h. Therefore, the total expected utility sum $E(U_{cp})$ is given as

$$E(U_{cp}) = p_{cp}\left(\frac{CC_{mn}}{n_m} n_{cp} - n_{cp} TC + \left(\frac{n_{cp}}{n_m}\right) n_m TC\right) + (1 - p_{cp})\left(\frac{CC_{mn}}{n_h} n_{cp} + \left(\frac{n_{cp}}{n_h}\right)\frac{CC_r}{n_{hop}^{max}}\right), \tag{32}$$

To prevent collusion, we want $E(U_{cp}) \leq E(U'_{cp})$, where U'_{cp} represents the utility of colluding players playing honestly, i.e.,

$$p_{cp}\left(\frac{CC_{mn}}{n_m} n_{cp}\right) + (1 - p_{cp})\left(\frac{CC_{mn}}{n_h} n_{cp} + \left(\frac{n_{cp}}{n_h}\right)\frac{CC_r}{n_{hop}^{max}}\right) \leq \frac{CC_{mn}}{n_h} n_{cp} + \left(\frac{n_{cp}}{n_h}\right)\frac{CC_r}{n_{hop}^{max}}, \tag{33}$$

which leads towards the condition, $CC_{mn} \leq \frac{n_m CC_r}{n_{hop}^{max}(n_h - n_m)}$. If $CC_{mn} \geq 0$, this condition can only be fulfilled when $n_h \geq n_m$, i.e., when $p_m \leq 0.5$. Similarly, if colluding players attempt to mark a false message as true, then $E(U_{cp})$ is given as,

$$E(U_{cp}) = p_{cp}\left(\frac{CC_{mn}}{n_m} n_{cp} + \left(\frac{n_{cp}}{n_m}\right)\frac{CC_r}{n_{hop}^{max}}\right) + (1 - p_{cp})\left(\frac{CC_{mn}}{n_h} n_{cp} - n_{cp} TC + \left(\frac{n_{cp}}{n_h}\right) n_h TC\right). \tag{34}$$

To prevent collusion, we require $E(U_{cp}) \leq E(U'_{cp})$, which leads towards the condition $CC_{mn} \geq \frac{n_h CC_r}{n_{hop}^{max}(n_m - n_h)}$. Combining the condition of Lemma 2, we want, $CC_{mn} \geq max\big(n_h TC, \frac{n_h CC_r}{n_{hop}^{max}(n_m - n_h)}\big)$. For $p_m \leq 0.5$, $CC_r > 0$ and $n_{hop}^{max} > 0$, we always get $\frac{n_h CC_r}{n_{hop}^{max}(n_m - n_h)} \leq 0$ and $n_h TC \geq \frac{n_h CC_r}{n_{hop}^{max}(n_m - n_h)}$. Therefore, we can prove that the incentive distribution mechanism is collusion resistant if $n_h TC \leq CC_{mn} \leq \frac{n_m CC_r}{n_{hop}^{max}(n_h - n_m)}$ and $p \leq 0.5$.

Acknowledgment

The authors would like to thank Saqib Ayaz, Technology Department, Workflow Management and Optimization Inc., Somerset, NJ, US for his excellent comments and suggestions.

References

[1] A. U. Rahman, A. W. Malik, V. Sati, A. Chopra, and S. D. Ravana, "Context-aware opportunistic computing in vehicle-to-vehicle networks," *Vehicular Communications*, vol. 24, p. 100236, Jan. 2020.

[2] S. Kim, "Impacts of Mobility on Performance of Blockchain in VANET," *IEEE Access*, vol. 7, pp. 68646-68655, Jun. 2019.

[3] M. Saravanan and P. Ganeshkumar, "Routing using reinforcement learning in vehicular ad hoc networks," *Computational Intelligence*, Jan. 2020.

[4] W. Wang, D. T. Hoang, P. Hu, Z. Xiong, D. Niyato, P. Wang, Y. Wen, and D. I. Kim, "A Survey on Consensus Mechanisms and Mining Strategy Management in Blockchain Networks," *IEEE Access,* vol. 7, pp. 22328–22370, Mar. 2019.

[5] S. Bano, A. Sonnino, M. Al-Bassam, S. Azouvi, P. Mccorry, S. Meiklejohn, and G. Danezis, "SoK: Consensus in the Age of Blockchains," *Proc. of the 1st ACM Conference on Advances in Financial Technologies*, Zurich, Switzerland, Oct. 2019.

[6] X. Wang, Y. Han, C. Wang, Q. Zhao, X. Chen, and M. Chen, "In-Edge AI: Intelligentizing Mobile Edge Computing, Caching and Communication by Federated Learning," *IEEE Network,* vol. 33, no. 5, pp. 156–165, 2019.

[7] J. Kang, Z. Xiong, D. Niyato, D. Ye, D. I. Kim, and J. Zhao, "Toward Secure Blockchain-Enabled Internet of Vehicles: Optimizing Consensus Management Using Reputation and Contract Theory," *IEEE Transactions on Vehicular Technology*, vol. 68, no. 3, pp. 2906-2920, Mar. 2019.

[8] Z. Yang, K. Yang, L. Lei, K. Zheng, and V. C. M. Leung, "Blockchain-based Decentralized Trust Management in Vehicular Networks," *IEEE Internet of Things Journal*, vol. 6, no. 2, pp. 1495-1505, Apr. 2019.

[9] L. Chen, L. Xu, N. Shah, Z. Gao, Y. Lu, and W. Shi, "On Security Analysis of Proof-of-Elapsed-Time (PoET)," *Lecture Notes in Computer Science Stabilization, Safety, and Security of Distributed Systems,* pp. 282-297, Oct. 2017.

[10] B. Choi, J.-Y. Sohn, D.-J. Han, and J. Moon, "Scalable Network-Coded PBFT Consensus Algorithm," *Proc. of IEEE International Symposium on Information Theory,* Paris, France, Jul. 2019.

[11] M. Cebe, E. Erdin, K. Akkaya, H. Aksu, and S. Uluagac, "Block4Forensic: An Integrated Lightweight Blockchain Framework for Forensics Applications of Connected Vehicles," *IEEE Communications Magazine,* vol. 56, no. 10, pp. 50-57, Oct. 2018.

[12] L. Zhang, M. Luo, J. Li, M. H. Au, K.-K. R. Choo, T. Chen, and S. Tian, "Blockchain based secure data sharing system for Internet of vehicles: A position paper," *Vehicular Communications,* vol. 16, pp. 85-93, Mar. 2019.

[13] W. Gao, M. Wang, L. Zhu, and X. Zhang, "Threshold-Based Secure and Privacy-Preserving Message Verification in VANETs," *Proc. of 13th IEEE International Conference on Trust, Security and Privacy in Computing and Communications*, Beijing, China, Sep. 2014.

[14] Y. Sun, L. Zhang, G. Feng, B. Yang, B. Cao, and M. A. Imran, "Blockchain-Enabled Wireless Internet of Things: Performance Analysis and Optimal Communication Node Deployment," *IEEE Internet of Things Journal,* vol. 6, no. 3, pp. 5791-5802, Jun. 2019.

[15] M. Ni, M. Hu, Z. Wang, and Z. Zhong, "Packet reception probability of VANETs in urban intersecton scenario," *Proc. of International Conference on Connected Vehicles and Expo,* Shenzhen, China, Oct. 2015.

[16] N. Li, J.-F. Martinez-Ortega, V. H. Diaz, and J. A. S. Fernandez, "Probability Prediction-Based Reliable and Efficient Opportunistic Routing Algorithm for VANETs," *IEEE/ACM Transactions on Networking,* vol. 26, no. 4, pp. 1933-1947, Aug. 2018.

[17] L. Liu, C. Chen, Q. Pei, S. Maharjan, and Y. Zhang, "Vehicular Edge Computing and Networking: A Survey," *Mobile Networks and Applications*, Jul. 2020.

[18] H. Ji, O. Alfarraj, and A. Tolba, "Artificial Intelligence-Empowered Edge of Vehicles: Architecture, Enabling Technologies, and Applications," *IEEE Access*, vol. 8, pp. 61020–61034, Apr. 2020.

[19] M. Li, J. Gao, L. Zhao, and X. Shen, "Deep Reinforcement Learning for Collaborative Edge Computing in Vehicular Networks," *IEEE Transactions on Cognitive Communications and Networking,* Jun. 2020.

[20] L. Nkenyereye, L. Nkenyereye, S. M. R. Islam, C. A. Kerrache, M. Abdullah-Al-Wadud, and A. Alamri, "Software Defined Network-Based Multi-Access Edge Framework for Vehicular Networks," *IEEE Access,* vol. 8, pp. 4220–4234, Jan. 2020.

[21] G. S. Aujla, A. Singh, M. Singh, S. Sharma, N. Kumar, and K.-K. R. Choo, "BloCkEd: Blockchain-Based Secure Data Processing Framework in Edge Envisioned V2X Environment," *IEEE Transactions on Vehicular Technology,* vol. 69, no. 6, pp. 5850–5863, Jun. 2020.

[22] S. Bano, M. Al-Bassam, and G. Danezis, "The road to scalable blockchain designs," *USENIX; login: magazine,* Dec. 2017.

[23] E.K. Kogias, P. Jovanovic, N. Gailly, I. Khoffi, L. Gasser, and B. Ford, "Enhancing bitcoin security and performance with strong consistency via collective signing," *Proc. of 25th USENIX Security Symposium,* Vancouver, BC, Canada, pp. 279-296, Aug. 2016.

[24] I. Eyal, A. E. Gencer, E. G. Sirer, and R. V. Renesse, "Bitcoin-NG: A scalable blockchain protocol," *Proc. of 13th USENIX Symposium on Networked Systems Design and Implementation,* Santa Clara, CA, USA, pp. 45-59, Feb. 2016.

[25] L. Luu, V. Narayanan, C. Zheng, K. Baweja, S. Gilbert, and P. Saxena, "A Secure Sharding Protocol For Open Blockchains," *Proc. of ACM Special Interest Group on Security, Audit and Control Conference on Computer and Communications Security,* Vienna, Austria, pp. 17-30, Oct. 2016.

[26] Y. He, H. Li, X. Cheng, Y. Liu, C. Yang, and L. Sun, "A Blockchain Based Truthful Incentive Mechanism for Distributed P2P Applications," *IEEE Access,* vol. 6, pp. 27324-27335, Jun. 2018.

[27] Q. Zhang, Y. Leng, and L. Fan, "Blockchain-based P2P File Sharing Incentive," *IACR Cryptology ePrint Archive,* p. 1152, 2018.

[28] H. Ichikawa and A. Kobayashi, "Messaging Protocol for Relaying Messages between Participants with Autonomous Distributed Blockchain Propagation," *Proc. of 5th International Symposium on Computing and Networking,* Aomori, Japan, Nov. 2017.

[29] L. Li, J. Liu, L. Cheng, S. Qiu, W. Wang, X. Zhang, and Z. Zhang, "CreditCoin: A Privacy-Preserving Blockchain-Based Incentive Announcement Network for Communications of Smart Vehicles," *IEEE Transactions on Intelligent Transportation Systems*, vol. 19, no. 7, pp. 2204-2220, Jan 2018.

[30] S. Zou, J. Xi, S. Wang, Y. Lu, and G. Xu, "Reportcoin: A Novel Blockchain-Based Incentive Anonymous Reporting System," *IEEE Access,* vol. 7, pp. 65544-65559, Jun. 2019.

[31] X. Zhang and X. Chen, "Data Security Sharing and Storage Based on a Consortium Blockchain in a Vehicular Ad-hoc Network," *IEEE Access,* vol. 7, pp. 58241-58254, May 2019.

[32] T. Jiang, H. Fang, and H. Wang, "Blockchain-Based Internet of Vehicles: Distributed Network Architecture and Performance Analysis," *IEEE Internet of Things Journal*, vol. 6, no. 3, pp. 4640-4649, Jun. 2019.

[33] M. Liu, Y. Teng, F. R. Yu, V. C. M. Leung, and M. Song, "Deep Reinforcement Learning Based Performance Optimization in Blockchain-Enabled Internet of Vehicle," *Proc. of IEEE International Conference on Communications,* Shanghai, China, May 2019.

[34] P. Mcmahon, T. Zhang, and R. Dwight, "Requirements for Big Data Adoption for Railway Asset Management," *IEEE Access*, vol. 8, pp. 15543–15564, Jan. 2020.

[35] W. Hu, Y. Hu, W. Yao, and H. Li, "A Blockchain-Based Byzantine Consensus Algorithm for Information Authentication of the Internet of Vehicles," *IEEE Access,* vol. 7, pp. 139703-139711, Oct. 2019.4

[36] C. T. Nguyen, D. T. Hoang, D. N. Nguyen, D. Niyato, H. T. Nguyen, and E. Dutkiewicz, "Proof-of-Stake Consensus Mechanisms for Future Blockchain Networks: Fundamentals, Applications and Opportunities," *IEEE Access,* vol. 7, pp. 85727-85745, Jul. 2019.

[37] R. Shrestha, R. Bajracharya, A. P. Shrestha, and S. Y. Nam, "A new type of blockchain for secure message exchange in VANET," *Digital Communications and Networks,* pp. 1-10, Apr. 2019.

[38] A. S. Khan, K. Balan, Y. Javed, S. Tarmizi, and J. Abdullah, "Secure Trust-Based Blockchain Architecture to Prevent Attacks in VANET," *Sensors,* vol. 19, no. 22, pp. 4954-4981, Nov. 2019.

[39] M. Wagner and B. Mcmillin, "Cyber-Physical Transactions: A Method for Securing VANETs with Blockchains," *Proc. of 23rd IEEE Pacific Rim*

International Symposium on Dependable Computing, Taipei, Taiwan, Dec. 2018.

[40] Z. Su, Y. Wang, Q. Xu, and N. Zhang, "LVBS: Lightweight Vehicular Blockchain for Secure Data Sharing in Disaster Rescue," *IEEE Transactions on Dependable and Secure Computing*, Mar. 2020.

[41] F. Ayaz, Z. Sheng, D. Tian, G. Y. Liang, and V. Leung, "A Voting Blockchain based Message Dissemination in Vehicular Ad-Hoc Networks (VANETs)," *Proc. of IEEE International Conference on Communications,* Dublin, Ireland, Jun. 2020.

[42] F. Ayaz, Z. Sheng, D. Tian,and V. Leung, "Blockchain-enabled Security and Privacy for Internet of Vehicles," *Internet of Vehicles and its Applications in Autonomous Driving,* Springer, 2020.

[43] X. Wang, Y. Han, V. C.M. Leung, D. Niyato, X. Yan, and X. Chen, "Convergence of edge computing and deep learning: A comprehensive survey," *IEEE Communications Surveys & Tutorials,* vol. 22, no. 2, pp.869-904, Jan. 2020.

[44] X. Wang, C. Wang, X. Li, V. C. M. Leung, and T. Taleb, "Federated Deep Reinforcement Learning for Internet of Things with Decentralized Cooperative Edge Caching," *IEEE Internet of Things Journal,* Apr. 2020.

[45] W. Li, S. Andreina, J.-M. Bohli, and G. Karame, "Securing Proof-of-Stake Blockchain Protocols," *Lecture Notes in Computer Science Data Privacy Management, Cryptocurrencies and Blockchain Technology,* pp. 297–315, Sep. 2017.

[46] F. Dressler, P. Handle, and C. Sommer, "Towards a vehicular cloud - using parked vehicles as a temporary network and storage infrastructure," *Proc. of ACM international workshop on Wireless and mobile technologies for smart cities*, Philadelphia, USA, Aug. 2014.

[47] Y. Qian and N. Moayeri, "Design of Secure and Application-Oriented VANETs," *Proc. of IEEE Vehicular Technology Conference,* Singapore, May 2008.

[48] V. Ortega, F. Bouchmal, and J. F. Monserrat, "Trusted 5G Vehicular Networks: Blockchains and Content-Centric Networking," *IEEE Vehicular Technology Magazine*, vol. 13, no. 2, pp. 121–127, April 2018.

[49] M. Haenggi, "Twelve reasons not to route over many short hops," *Proc. of* 60^{th} *IEEE Vehicular Technology Conference,* Los Angeles, CA, USA, Sep. 2004.

[50] Y. Yokoya, Y. Asano, and N. Uchida, "Qualitative change of traffic flow induced by driver response," *Proc. of IEEE International Conference on Systems, Man and Cybernetics*, Singapore, pp. 2315-2320, Oct. 2008.

[51] X. M. Zhang, X. Cao, L. Yan, and D. K. Sung, "A street-centric opportunistic routing protocol based on link correlation for urban VANETs," *IEEE Transactions on Mobile Computing*, vol. 15, no. 7, pp. 1586-1599, Jul. 2016.

[52] R. Stanica, E. Chaput, and A.-L. Beylot, "Local Density Estimation for Contention Window Adaptation in Vehicular Networks," *Proc. of* 22^{nd} *IEEE International Symposium on Personal, Indoor and Mobile Radio Communications,* Toronto, Canada, Sep. 2011.

[53] C. Yeshwanth, P. S. A. Sooraj, V. Sudhakaran, and V. Raveendran, "Estimation of intersection traffic density on decentralized architectures with deep networks," *Proc. of International Smart Cities Conference*, Wuxi, China, Sep. 2017.

[54] J. Wroughton and T. Cole, "Distinguishing Between Binomial, Hypergeometric and Negative Binomial Distributions," *Journal of Statistics Education,* vol. 21, no. 1, Mar. 2013.

[55] M. Mitzenmacher and E. Upfal, "Probability and computing: randomized algorithms and probabilistic analysis," *Cambridge University Press,* Jul. 2017.

[56] D. Tian, J. Zhou, M. Chen, Z. Sheng, Q. Ni, and V. C. Leung, "Cooperative Content Transmission for Vehicular Ad Hoc Networks using Robust Optimization," *Proc. of IEEE Conference on Computer Communications*, Honolulu, HI, USA, Oct. 2018.

[57] M. Killat and H. Hartenstein, "An Empirical Model for Probability of Packet Reception in Vehicular Ad Hoc Networks," *EURASIP Journal on Wireless Communications and Networking*, vol. 2009, no. 1, Dec. 2009.

[58] A. Durand, E. Ben-Hamida, D. Leporini, G. Memmi. "Asymptotic Performance Analysis of Blockchain Protocols," *arXiv preprint arXiv:1902.04363,* Feb. 2019.

[59] V. Buterin, D. Reijsbergen, S. Leonardos, and G. Piliouras, "Incentives in Ethereum's Hybrid Casper Protocol," *Proc. of IEEE International Conference on Blockchain and Cryptocurrency*, Seoul, South Korea, May 2019.

[60] H. Watanabe, S. Fujimura, A. Nakadaira, Y. Miyazaki, A. Akutsu, and J. Kishigami, "Blockchain contract: Securing a blockchain applied to smart contracts," *Proc. of IEEE International Conference on Consumer Electronics,* Las Vegas, NV, USA, Jan. 2016.

[61] J. Petit, "Analysis of ECDSA Authentication Processing in VANETs," *Proc. of* 3^{rd} *International Conference on New Technologies, Mobility and Security,* Cairo, Egypt, Dec. 2009.

[62] A.S. A. Al-Sobky and R. M. Mousa, "Traffic density determination and its applications using smartphone," *Alexandria Engineering Journal,* vol. 55, no. 1, pp. 513–523, Mar. 2016.

[63] S. M. A. El-Atty and G. K. Stamatiou, "Performance analysis of Multihop connectivity in VANET," *Proc. of* 7^{th} *International Symposium on Wireless Communication Systems*, York, UK, Sep. 2010.

[64] S. Shen, Y. Han, X. Wang, and Y. Wang, "Computation offloading with multiple agents in edge-computing–supported IoT," *ACM Transactions on Sensor Networks*, vol. 16, no. 1, pp. 1-27, Dec. 2019.

Ferheen Ayaz (Student Member, IEEE) received her B.E. and M.E. degree from NED University of Engineering and Technology, Pakistan, in 2010 and 2014, respectively. She is currently pursuing Ph.D. degree with University of Sussex, UK. Her current research interests include blockchain applications in vehicular communications.

Zhengguo Sheng (Senior Member, IEEE) received the B.Sc. degree from University of Electronic Science and Technology of China, in 2006 and M.S. and Ph.D. degrees from Imperial College London, UK, in 2007 and 2011, respectively. He is currently a Senior Lecturer with University of Sussex, UK. Previously, he was with UBC as a Research Associate and with Orange Labs as a Senior Researcher. He has more than 100 publications. His research interests cover IoT, vehicular communications, cloud/edge computing.

Daxin Tian (Senior Member, IEEE) is a professor in the School of Transportation Science and Engineering, Beihang University, Beijing, China. His current research interests include mobile computing, intelligent transportation systems, vehicular ad hoc networks, and swarm intelligence.

Yong Liang Guan (Senior Member, IEEE) obtained his PhD from the Imperial College London, UK, and Bachelor of Engineering with first class honors from the National University of Singapore. He is a Professor of Communication Engineering at the School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore, where he now leads two industry collaboration labs (Continental-NTU Corporate Research Lab, and Schaeffler Hub for Advanced Research at NTU) and led the successful deployment of the campus-wide NTU-NXP V2X Test Bed. His research interests broadly include coding and signal processing for communication systems and data storage systems. He is an Editor for the IEEE Transactions on Vehicular Technology.