Received 12 September 2019; revised 8 January 2020; accepted 31 January 2020.
Date of publication 6 February 2020; date of current version 16 September 2021.

Digital Object Identifier 10.1109/TETC.2020.2971831

Blockchain-Based On-Demand Computing Resource Trading in IoV-Assisted Smart City

XI LIN, (Student Member, IEEE), JUN WU, (Member, IEEE), SHAHID MUMTAZ, (Senior Member, IEEE), SAHIL GARG, (Member, IEEE), JIANHUA LI, (Member, IEEE), AND MOHSEN GUIZANI, (Fellow, IEEE)

Xi Lin, Jun Wu, and Jianhua Li are with the School of Cyber Security, Shanghai Jiao Tong University, 12474Shanghai 200240, China, and also with Shanghai Key Laboratory of Integrated Administration Technologies for Information Security, Shanghai 200240, China
Shahid Mumtaz is with the Group of Mobile System, Instituto de Telecomunicaçaões (IT), 1049-001, Lisbon Portugal
Sahil Garg is with the Electrical Engineering Department, École de technologie supérieure, Université du Québec, 14845Montréal QCH3C 1K3, Canada
Mohsen Guizani is with the Computer Science and Engineering, Qatar University, 61780Doha 2713, Qatar
CORRESPONDING AUTHOR: J. WU (junwuhn@sjtu.edu.cn)

ABSTRACT In a smart city, Mobile Edge Computing (MEC) are generally deployed in static fashion in base stations (BSs). While moving vehicles with advanced on-board equipment can be regarded as dynamic computing resource transporters ignoring geographical limitations. Thus Internet of Vehicle (IoV) could assist the smart city to achieve flexible computing resource demand response (DR) via paid sharing the idle vehicle computing resources. Motivated by this, we propose a Peer-to-Peer (P2P) computing resource trading system to balance computing resource spatio-temporal dynamic demands in IoV-assisted smart city. On one hand, to guarantee transaction security and privacy-preserving in our system, we employ a consortium blockchain approach and demonstrate the process of secure computing resource trading without involving a centralized trusted third-party. On the other hand, to encourage individual smart vehicles to participate in our system, we construct a two-stage Stackelberg game jointly optimizing the utilities of buyers and sellers. And we also derive the optimal computing pricing and trading amount strategies in this proposed game. Finally, security analysis shows the security performance of our system and numerical simulations show that our strategies can encourage the collaboration between the buyer and smart vehicles.

INDEX TERMS Consortium blockchain, Internet of Vehicles, mobile edge computing, smart city

I. INTRODUCTION

Recently, Mobile User Equipment (MUE) is earning more and more popularity, and Mobile Applications (MAs), such as augment/virtual reality, online gaming, and image/video editing are emerged and caught wide attention [1], [2]. But limited computing capacity and constrained battery power at MUE could not fulfill the requirements of resource-hungry MAs [3]. In a smart city, Mobile Edge Computing (MEC) acts as a new paradigm to supply computing resources for local users. With the help of MEC systems, mobile users could break the limitations of MUE and improve users' experience.

However, traditional MEC systems integrated into Radio Access Network (RAN) are deployed statically in base stations (gNodeB or IoT gateway) by the network operators [4], [5]. Thus, the number of MEC nodes is limited due to additional deployment and maintenance costs. Besides, once deployed, MEC servers can hardly be redeployed with high inefficiency. Moreover, we claim that this traditional statical model is also contradicted to the spatial and temporal computing demands of mobile users. For instance, users would work at Central Business District (CBD) during the day (high demands), while returning home for resting at the residential district in the evening (high demands). Thus, at a specific time and place, the deployed statical MEC node will experience an overfull number of computing demands far beyond its service capacity, which directly reduces users' Quality of Service (QoS) for necessary service queueing time [6]. A straightforward solution is activating more servers for each MEC node to accommodate users' demands. While it will bring a huge financial burden to service providers. Besides, the computing demands are not constant over time for a certain region. This means the over-provisioned computing resources will be wasted during off-peak periods [7]. The other solution resorts to resource scheduling between multiple MECs [8], [9] or between cloud and edge [10]. But it would be harmful to mobile users' QoS with the adding service time by corresponding backhaul link and core network. Therefore, in MEC-enabled smart city, it is yet a significant issue to explore an effective computing resource provision scheme to meet users' QoS not in the pattern of pure economic expenses [6].

As the Gartner informed, a quarter billion vehicles with advanced on-board devices will appear by 2020 [11]. And the abundant on-board resources in vehicles are idle and wasted, such as cache, communication, and computing under ordinary circumstances [12]. Meanwhile, in 5G-enabled V2X (Vehicle-to-Everything), smart vehicles can communicate with MEC systems and mobile users with superior performance, such as delay, reliability, and mobility. Specifically, in 5G, the peak data rate for low and high mobility is about 10 Gb/s and 1 Gb/s, respectively. And the transmission latency for moving vehicles is shorter than around 1 ms [8]. Thus, those smart vehicles could be conceived as complement resources beyond the MEC service capacity without reducing users' QoS. In addition, the moving resources will not cause over-provisioning, for a highly positive correlation between vehicle movements and user social behaviors. That is, in many social hotspots, the parking or moving vehicles nearby are intuitively proportional to the number of mobile users. For example, on the weekend, a large number of mobile users will go to a modern shopping mall for shopping or leisure, causing congestion of MEC services nearby [7]. While it also brings lots of vehicles in the parking lot. Therefore, vehicles can trade their idle computing resources in a Peer-to-Peer (P2P) manner to meet dynamic computing demands in IoV-assisted smart city.

However, smart vehicles with surplus computing resources may be reluctant to participate in our trading process, due to their selfishness and distrust. They not only need the optimal trading strategies out of their benefits as incentives, but also need to ensure security and privacy in the trading process. Recently, blockchain, as a decentralized P2P trading system, causes great attention from all walks of life. Blockchain was first proposed by Satoshi in a famous P2P e-cash system *Bitcoin* [13], which has good properties, such as tamper-resistance and transparency. In general, blockchain technology could provide a secure and privacy-preserving trading platform for the P2P trading market, which could also encourage nodes to participate in the trading process. Therefore, we utilize the consortium blockchain technology to establish a secure P2P computing resource trading system in IoV-assisted smart city without involving a centralized trusted third-party. The main contributions of this paper are summarized as follows:

- First, we propose a novel P2P computing resource trading system to achieve dynamic computing resource demand response (DR) in IoV-assisted smart city.
- Second, we exploit a consortium blockchain approach to guarantee transaction security and privacy-preserving for the P2P computing resource trading system in a decentralized manner.
- Third, to stimulate selfish vehicles to participate in our system, we construct a two-stage Stackelberg game, jointly optimizing the profits of buyers and sellers. Besides, a gradient-based search algorithm is proposed to acquire the optimal strategies.
- Finally, we conduct security analysis and numerical simulation. The results show the security performance of our system and our strategies could stimulate vehicles to trade computing resources.

The rest of this paper is organized as follows. Section II gives an overview of the related work. Section III describes core elements and computing resource trading process in our system. Section IV introduces the Stackelberg game construction in our system. Section V shows the security analysis and numerical simulation, and Section VI presents the conclusion and future work.

II. RELATED WORKS

A. VEHICLE CLOUD/FOG COMPUTING IN SMART CITY

In [15], Olariu *et al.* proposed the concept of Vehicular Cloud Computing (VCC), which is to gather the under-utilized on-board computing resources. Zhang *et al.* employed the VCC as a cloud service provider to expand the available resources for task requests from smartphones in [16]. And the author in [17], [18] proposed a new architecture called Vehicular Fog Computing (VFC), employing vehicles' computing and communication resources as the infrastructures in the smart city. So, by employing the idle computing resources in vehicles nearby, MEC can extend their computing capacity with smaller costs. Motivated by this, in [7], The authors designed a Fog Vehicular Computing (FVC) concept to augment the computing and storage capacity of edge/fog computing. And an IoV-assisted computing framework named *vFog* was proposed in [19] for delay-sensitive tasks in the smart city. However, security issues and comprehensive pricing and computing resource allocation strategies have not been well resolved in the previous works.

B. MOBILE EDGE COMPUTING IN SMART CITY

Mobile (or Multi-access) Edge Computing was first proposed by European Telecommunications Standards Institute (ETSI) [20]. Many previous works concentrated on optimization algorithms for task offloading in MEC system, to optimize energy consumption or service latency [21], [22]. Recently, artificial intelligence has been introduced in MEC system to realize edge intelligence. In [23], a deep Q-network (DQN) based data offloading policy was proposed for mobile users. In order to balance the workload among MECs in smart city, one method called Virtual Machine (VM) migration is adapted to achieve effective computing resource demand response management (DRM). Gkatzikis and Koutsopoulos [24], and Mishra *et al.* [25] used VM migration to move VMs between MEC and improved the efficiency of MEC usage, avoiding the waste of ideal computing resources. In [26], the author proposed a task migration scheme in MEC system based on deep Q-network, which could learn from the previous experiences. However, Jia *et al.* [27] offered a different point on VM migration, that is migrations often need users to deliver tasks to MEC not nearby, so they might be detrimental

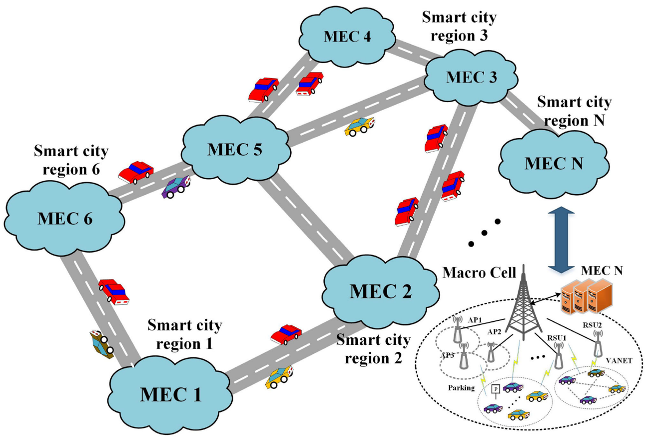


FIGURE 1. IoV-assisted computing resource demand response.

to users' QoS adding the time of migrations themselves and corresponding backhaul link.

C. BLOCKCHAIN TECHNOLOGY

Recently, blockchain technology is widely utilized in P2P systems, such as vehicular ad-hoc network (VANET) and Internet of Things (IoT). A blockchain-based privacy-preserving payment scheme for Vehicle-to-Grid (V2G) is designed in [28], [29] to realize secure energy trading. In [30], the authors proposed an efficient and privacy-preserving carpooling system using blockchain and vehicular fog computing. For data sharing aspects, data security sharing and storage in vehicular networks and IoT are also proposed in [31]. Besides, the authors also employed a consortium blockchain to establish a P2P market model for secure knowledge trading [32]. Combined blockchain and edge computing, some works focus on computing resource allocation for blockchain mining process. In [33], an economic approach for edge computing resource management for blockchain technology is proposed. Besides, the authors also designed a social welfare maximization auction mechanism in edge computing enabled blockchain technology in [34]. However, most of the previous works ignored the utilization of blockchain technology to implement secure P2P computing resource trading management in IoV-assisted smart city.

III. P2P COMPUTING RESOURCE TRADING SYSTEM

A. IOV-ASSISTED COMPUTING RESOURCE DEMAND RESPONSE

As shown in Figure 1, in 5G Heterogeneous Network (HetNet) enabled smart city, we assume each region has a Macro-Cell base station and several Small-Cell base stations, which are responsible for intra-cell communication resource orchestration. And MEC servers will be statically deployed in each Macro-Cell base station to provide cloud-like computing services in vicinity constrained. Thus, resource-constraint mobile users in each region could request reliable and low-latency computing services from the localized MEC node via 5G HetNet. However, during the peak hours in each region, the localized MEC node will be overwhelmed by the sharp increase of incoming computing demands, a group of smart vehicles moving or parking within the area of the local 5G mobile network could be employed to reduce the overload pressure of MEC node through computing resources trading. They could share their resources with mobile users in the local area for a period of time to perform task offloading and processing.

In order to compensate for the limited resources during rush hours, traditional MEC systems usually resort to the coordination of MECs or the synergy between cloud and MEC. However, it will require further offloading tasks to the cloud or other idle MEC, which essentially goes against the meaning of MEC performing local computing tasks. Thus, it will bring a large transmission delay, which inevitably reduces the users' QoS. In IoV-assisted smart city, the regions are connected by moving vehicles with idle computing resources, as shown in Figure 1. Meanwhile, as human-driven computing resources, the spatio-temporal distribution of vehicle computing resources is highly correlated with population distribution (i.e., computing demand distribution) in the smart city [17]. Therefore, the computing resource demand response dynamics of all the regions could be coupled with the mobility of the smart vehicles, which could achieve the flexible computing resource demand response. Motivated by the above statement, to balance computing demands for each MEC via smart vehicles, we propose a localized computing resource trading management system, which provides incentives to release the idle computing resources in vehicles out of their own benefits.

B. ELEMENTS OF OUR TRADING SYSTEM

In our localized P2P computing resource trading system, smart vehicles can share their idle computing resources with corresponding returns, named computing coins. It is considered as a new cryptographic currency similar to *BTC* and *Ether* in the famous *Bitcoin* system [13] and *Ethereum* system [14], respectively. As shown in Figure 2, our trading system consists of three major elements: trusted authority, smart vehicles(traders) and MEC nodes.

- *Trusted authority (TA):* Trusted authority is responsible for initializing our trading system. TA can generate digital certificates ($Cert_i$) and manage encryption keys (PK_i/SK_i) for all the trading system participants. Similar to the *Bitcoin* system, the pseudonym public key (PID_i) will also be used as the wallet address to store the digital asset of each node (i.e., computing coin). Once registered with the TA, TA will first assess whether participants' computing resources meet the basic standards. After that, the participants could be regarded as legitimate system members. Noted that, this role just implements the parameter initialization, which does not conflict with the decentralized characteristic of blockchain.
- *Smart vehicles:* Smart vehicles in the system could play different roles according to their computing resource states and mobility profiles: computing resource seller,

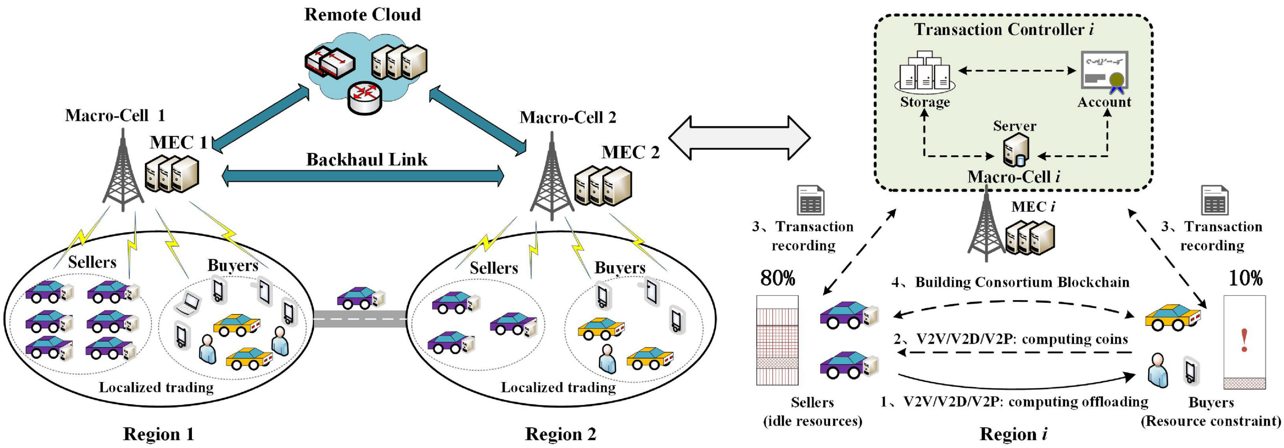


FIGURE 2. **P2P computing resource trading management system in smart city.**

computing resource buyer or idle node. Any local smart vehicle could paidly share their computing resources as a seller if it has adequate idle computing resources. It would accept and handle the offloading tasks from buyers utilizing the sold computing resources. Apart from resource-constrained mobile users/devices, computing-restricted vehicles can also become computing resource buyers. Therefore, as shown in Figure 2, the P2P computing resource trading mode could be Vehicle-to-Pedestrian (V2P), Vehicle-to-Device (V2D) or Vehicle-to-Vehicle (V2V). Idle nodes will neither sell nor buy computing resources in our trading system.

- *MEC nodes:* Apart from providing computing resources for edge computing services, MEC nodes are also considered as managers of the local P2P trading system. They could store and manage the mobility profiles and computing resource state of smart vehicles in own areas, the uploading computing resource transaction records from both computing resource sellers and buyers. Meanwhile, they are also pre-selected high power nodes of the consortium blockchain, which are responsible for maintaining the entire consortium blockchain. Specifically, MEC nodes act as blockchain miners and achieve the consensus process of transaction data in each region. Moreover, as the coordinator of the mobile network in each region, the MEC sometimes will also act as a relay point for task processing. That is, when the computing resource trading parties cannot establish direct short-distance communication (V2V, V2D or V2P) within the local area, the computing tasks or computing results can be sent to the MEC and then forwarded to the specified trading objects.

As shown in Figure 2, the detail trading process is described as follows: Computing resource traders (mobile devices and smart vehicles) first obtain the digital certificates and encryption keys from the TA. And then they choose trading roles based on their resource status and computing resource demands. Participants (mobile devices and smart vehicles) with idle resources become sellers in the market to balance local computing resource demands. Specifically, the sellers will provide their idle computing resources to implement computing tasks offloading from the buyers. The specific resource allocation and pricing strategies could refer to Section IV. Authorized traders can send resource status to the local MEC transaction server. The MEC server needs to verify their legal certificates to realize permissioned access to the trading market. The local MEC server broadcasts legitimate sellers' information to the buyers, matching the supply and demand side of the computing resource. And the buyer is required to pay the seller corresponding rewards from their wallets, i.e., computing coins. In specific, the buyer and the seller will initiate a new transaction and upload the transaction record to the local MEC server for audit. The MEC servers of each domain collect local transaction records in a period of time, packaging them into blocks, and build a cross-domain consortium blockchain to validate and audit transaction data. The detailed consortium blockchain consensus process is shown in Section III-C. In addition, both buyers and sellers can download the latest transaction data blocks from the local MEC server to verify the legality of the transaction.

C. CONSORTIUM BLOCKCHAIN FOR OUR TRADING SYSTEM

As dynamic computing resources travelling around the whole city, the same smart vehicles will perform computing resource trading when it is passing through different regions. Meanwhile, the same mobile devices, as the demand side of computing resources, will also roam throughout the whole city. That is, in our system, P2P computing resource trading will happen anywhere and anytime in the smart city. Therefore, it is significant to achieve consistent unification of the transaction data across geographic regions and time domains in a secure and privacy-preserving way.

Blockchain technologies could realize the P2P secure and trusted trading due to their tamper-proof and distributed consensus features, which are applicable for our trading system. Thus, we employ blockchain technologies for our system. In specific, blockchain is mainly divided into two types: public chain and permissioned chain (i.e., consortium chain). For the public chain, each node could freely join and exit the blockchain network, and has the right to read and write the on-chain data, such as *Bitcoin* and *Ethereum*. Besides, there is no centralized server node in the public chain network and it also requires achieving consensus process among all the participating nodes. While the permissioned chain (consortium chain) only requires licensed nodes to participate in the blockchain system. And consortium chain will form a consensus protocol among pre-selected high-capacity nodes, such as *EOS* and *Hyperledger*. In our system, we will employ the consortium blockchain technology for computing resource trading. The motivations behind it are summarized as follows:

- *Authorized access:* Consortium blockchain provides a controllable access mechanism. Only authorized participants with sufficient computing resources can participate in our computing resource market, avoiding some low-quality computing resource sellers.
- *Efficient and high throughput:* Only pre-selected nodes will implement the consensus process of the consortium blockchain, which makes the consensus process more efficient. It brings efficient and high throughput transactions for our trading system.
- *Resources limitation:* Mobile devices and vehicles (resources traders) are unwilling to waste additional computing resources for mining (i.e., Proof-of-Work) in public chain. Thus the consortium blockchain is more feasible in our system.
- *Semi-trusted MEC nodes:* In IoV-assisted smart city, MEC nodes are distributedly deployed and managed as trusted infrastructure, while they are still vulnerable to malfunctions, abnormalities and intrusions. Thus MEC nodes show the semi-trusted features in the smart city. Compared with mobile devices and vehicles, the MEC nodes have more superior resources (computing, communication and storage), which have the ability to implement the consortium blockchain.

1) STRUCTURE OF OUR BLOCKCHAIN

The structure of our blockchain consists of some blocks arranged in order, where each block contains the transaction records (TXs) and transaction data. The TXs are generated by participants in our system and are required to broadcast to all the other nodes to ensure consistent unification. Each block is connected to the prior one with a special hash digital digest of the prior block, which guarantees data integrity and tamper-resistance. In specific, each block involves a block header and a block body.

The block header contains three sets of metadata data: (1) Hash value of the prior block; (2) Target hash value, timestamp, nonce (solution of the hash puzzle); (3) The Merkle root hash value of all transaction records in the block body. The TXs are stored in the block body in the form of the Merkel tree structure. It is a hash binary tree structure to ensure efficient retrieval and TXs tamper-proof.

2) PROOF-OF-WORK (POW) CONSENSUS PROTOCOL

Multiple nodes will form a distributed network through asynchronous communication to ensure a consistent consensus in blockchain. Thus the consensus protocol plays the most important role in the blockchain system. Proof-of-Work (PoW) is a common consensus protocol employed in public chains, such as *Bitcoin* and *Ethereum* systems. This protocol requires participating nodes to solve the customized hash puzzle, which will consume participants' computing resources according to Eq. (1). And the winner has the right to publish the latest block. At the same time, the winning node (miner) will obtain the return (i.e., the cryptographic currency) contained in the block

$$\begin{aligned} &Hash(ID, timestamp, hash(pre\_block), nonce) \\ &\quad \leq target_{PoW} = 2^{N_{PoW}} - 1. \end{aligned} \tag{1}$$

All the miners have the same target, thus the miners could only pay more computing resources to find the answer to the hash puzzle. According to Eq. (2), the probability of solving the hash puzzle is proportional to the ratio of the consumed computing resources to the entire network computing power

$$\begin{aligned} p_{i\_PoW}(f_{i\_PoW}, F_{-i\_PoW}) &= \frac{f_{i\_PoW}}{\sum_j f_{j\_PoW}} > 0 \\ \sum_j p_{j\_PoW} &= 1. \end{aligned} \tag{2}$$

The PoW protocol has better security performance such as *Sybil-Resist*. However, it faces a lot of resource consumption (energy and computation) and low throughput in the P2P trading system.

3) PBFT CONSENSUS PROTOCOL

Practical Byzantine Fault Tolerance (PBFT) consensus protocol reduces the complexity of the Byzantine Fault Tolerance (BFT) protocol from exponential level to the polynomial level $o(N^2)$, which is widely used in the permissioned chain (or consortium chain), such as *EOS* and *Hyperledger*.

Compared with the PoW protocol, PBFT can achieve higher transaction throughput at the expense of some security performance. For instance, PBFT is hard to resist *Sybil attack* during the voting and election process, which will be solved well by the PoW protocol. The consensus process based on PBFT protocol is shown in Figure 3, which includes 5 steps. Besides, it is worth noting that the PBFT protocol requires less than 1/3 malicious nodes in the P2P trading system according to [35].

4) POW-PBFT CONSENSUS PROTOCOL IN OUR SYSTEM

A single consensus protocol is difficult to meet both security performance and system throughput. So in our trading system, we adopt a hybrid consensus protocol that combines the traditional PoW protocol with the PBFT protocol. Specifically, in our system, we determine the node identity based on the Proof-of-Work, which avoids the creation of multiple faulty identities (*Sybil attack*) by a malicious node. And the greater

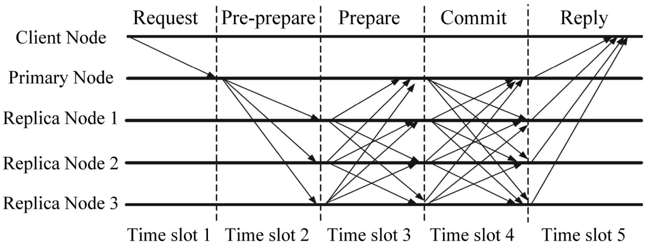


FIGURE 3. The process of PBFT consensus protocol.

the workload of the nodes, the more rewards will be obtained (similar to the *Bitcoin*). In general, we use the PoW protocol to ensure incentives and node elections. We then use the higher throughput consensus protocol PBFT for multi-party audit and verification. In PBFT protocol, we could also prevent blockchain forks, which will happen in traditional PoW protocol. Thus, PBFT makes the block generating speed only depends on the network transmission performance. In our system, network performance could be guaranteed via the high-speed high-bandwidth wired backhaul links between MEC nodes. Thus, the hybrid PoW-PBFT protocol could improve the trading performance on the premise of trading security. The PoW-PBFT based consensus algorithm in our computing resource trading management system is shown in Algorithm 1.

Algorithm 1. PoW-PBFT based consensus algorithm in our computing resource trading management system

Initialization: MEC^*: the leader of PBFT required to be verified; *Followers*: other MEC nodes; M: the number of PBFT consensus nodes; *VR*: the verification reply from other followers
Block: MECs collect and digitally sign the TXs in each area, and then put them into a block
PoW: MECs find hash puzzle solution (Eqs. (1) and (2)), the winner is selected as MEC^* of PBFT
MEC^*: Sends the TXs block and corresponding solution to other *followers*
Followers: Verifies the hash solution, TXs and digital certificate of MEC^* and reply the results (*VR*)
MEC^*: Collects all the *VR* from *followers*
If: more than $\frac{M-1}{3}+1=\frac{M+2}{3}$ *followers* have the consistent result
  then
    return the *VR*
  else
    return *null*
end if
MEC^*: Sends *VR* to *followers* for block storage and gain the block reward

IV. STACKELBERG GAME CONSTRUCTION FOR OPTIMAL TRADING STRATEGIES

Individual smart vehicles may be reluctant to participate in our trading system because of security threats and rational economic incentives. Therefore, apart from bringing the blockchain technology to guarantee trading security, we should also develop an optimal computing resource trading strategy considering the utilities of both parties. In the conventional MEC system, the computing resource buyer, such as a mobile device, could only access one or no more than two BSs. Unlike that, in IoV-assisted smart city, a mobile device could be surrounded by multiple smart vehicles (potential computing resource sellers). That is, one-buyer and multi-sellers resource trading and task allocation are more in line with our system. The trading process between the buyer and sellers could be well simulated by a hierarchical game model, such as Stackelberg game, which is widely employed in the smart city. In [36], the authors analyzed the performance of a game-based bidirectional electricity market in the smart home. Besides, a four-stage Stackelberg game with max-min fairness was also proposed for energy trading in electric vehicles enabled smart city [37]. However, compared with energy-related trading, the computing resource trading game model in the smart city is still vague. Therefore, motivated by previous works, we model a one-leader and multi-followers two-stage Stackelberg game for the interaction between the computing resource buyer and sellers, which is shown in Figure 4.

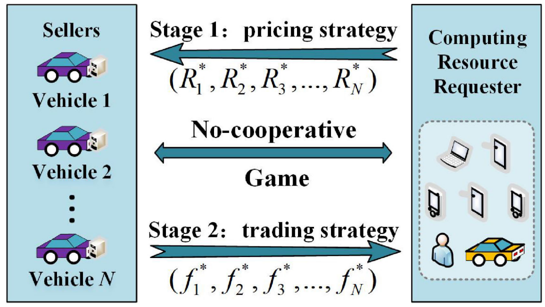


FIGURE 4. Two-stage Stackelberg game model in game step.

We suppose that some vehicles might be employed by a computing resource buyer, i.e., $\Omega = \{1, 2, 3, \ldots, N\}$. And we also define the State of Computing (SoC) of each computing resource seller (vehicle) v_i as $SoC_i = \{\bar{f}_i, \rho_i, r_i, c_i\}$ $(\forall i \in \Omega)$, which includes some parameters. a) $\bar{f}_i$ shows the maximal computing capacity of v_i; b) ρ_i shows the computing resource demand for v_i for own use (i.e., diverse in-car applications); c) r_i shows the unit reward of computing resource for v_i for their own use; d) c_i shows the unit cost of computing resource of v_i. It should be noticed that r_i is larger than v_i, which is more in accordance with real life. And at stage 1, computing resource requester (buyer) sets the discriminatory pricing $\boldsymbol{R} = (R_1, R_2, R_3, \ldots, R_N)$ for renting unit vehicles' computing resource to process computing tasks in unit time, and at stage 2 smart vehicles (sellers) decide the computing resource trading amount $\boldsymbol{f} = (f_1, f_2, f_3, \ldots, f_N)$ $(0 \leq f_i \leq \bar{f}_i)$ to the computing resource requester (buyer). The main symbols and explanations are shown in Table 1.

A. UTILITY OPTIMIZATION PROBLEM FORMULATION

The utility of computing resource requester (buyer) is defined as the computing cost without purchasing computing resource

IEEE TRANSACTIONS ON EMERGING TOPICS IN COMPUTING

TABLE 1. Main terms referred in our system.

Symbol	Explanations
$target_{PoW}$	The hash puzzle target in the PoW protocol for miners
p_{i_PoW}	The probability of finding a hash puzzle
M	The number of consensus nodes in the PBFT protocol
SOC_i	The state of computing for smart vehicles (sellers)
$\bar{f}_i$	The maximal computing capacity of smart vehicles
ρ_i	The computing resource demand for smart vehicles (sellers) for their own use
r_i	The unit reward of computing resource for smart vehicles (sellers) for their own use
R_i	The discriminatory pricing set by the computing resource requester (buyer)
f_i	The computing resource trading amount decided by smart vehicles (sellers)
U	The utility of the computing resource requester (buyer)
P_i	The benefit of each smart vehicle (seller)
U_i	The utility of each smart vehicle (seller)

minus the cost after purchasing with the purchasing pricing for each seller. And we employ $e^{\lambda f(x)}$ to present the buyer diminishing return of cost (i.e., marginal income). That is, the utility function of computing resource requester is expressed as

$$U = a_k e^{\lambda_k \sum \bar{f}_i} - \left(a_k e^{\lambda_k \sum (\bar{f}_i - f_i)} + \sum R_i f_i \right). \tag{3}$$

Where a_k and λ_k are the two fixed coefficient of operation utility that is distinct from each computing resource requester j. And $a_k, \lambda_k > 0$.

For each computing resource seller, we suppose that they would give priority to utilizing computing resources for their own use, which is more in line with the real world. That is, smart vehicles need to find a trade-off between the utilization of their own use and the benefit of external trading under the condition of external economic incentives. Thus, similar to [8], we define sellers satisfaction function $\varepsilon(.)$ with the logarithm utility function that is widely employed in network economics. Thus the satisfaction function of v_i is described as follow:

$$\varepsilon(\tilde{f}_i) = \varepsilon(\bar{f}_i - f_i) = \gamma_i \ln(\bar{f}_i - f_i - \rho_i + 1). \tag{4}$$

Where $\gamma_i > 0$ is the different satisfaction coefficient between smart vehicles. The higher γ_i of the smart vehicle means that it is more concerned about local resource utilization. Where $\tilde{f}_i = \bar{f}_i - f_i$ shows the remaining computing resources, when the remains is lower than ρ_i, satisfaction function $\varepsilon(\bar{f}_i - f_i) < 0$. While the remains is higher than ρ_i, $\varepsilon(\bar{f}_i - f_i) > 0$. While the remains is equal to ρ_i, $\varepsilon(\bar{f}_i - f_i) = 0$. Thus the benefit of smart vehicles after selling computing resources is denoted as follow:

$$P_i(f_i) = (r_i - c_i)(\tilde{f}_i + \varepsilon(\tilde{f}_i)) + (R_i - c_i)f_i. \tag{5}$$

The utility function of sellers should be the benefit after selling $P_i(f_i)$ minus the benefit before selling $P_i(f_0)$, so the utility of v_i is given as

$$U_i = (r_i - c_i)\left(\gamma_i \ln \frac{\tilde{f}_i - \rho_i + 1}{\bar{f}_i - \rho_i + 1} - f_i \right) + (R_i - c_i)f_i. \tag{6}$$

Our purpose is to achieve maximal utilities of both the buyer and sellers in our proposed game. By taking the negation, i.e., $U' = -U$, $U_i' = -U_i$ we can transform the max problems into the min problems that is more applicable in optimization theory. So, the utility optimization problems (P1 and P2) at stage 1 and stage 2, respectively, could be shown as follows:

$$\begin{aligned} &\min_R U'(R) = a_k e^{\lambda_k \sum (\tilde{f}_i)} + \sum R_i f_i - a_k e^{\lambda_k \sum \bar{f}_i} \\ &\quad s.t.\ R_i' \le R_i \le R_i'' \quad (R_i \in R) \end{aligned} \tag{7}$$

$$\begin{aligned} \min_{f_i} \quad & U'_i(f_i) = -(r_i - c_i)\left(\gamma_i \ln \frac{\tilde{f}_i - \rho_i + 1}{\bar{f}_i - \rho_i + 1} - f_i \right) \\ & \qquad - (R_i - c_i) f_i \\ & s.t. \quad 0 \le f_i \le \bar{f}_i \quad (f_i \in f) \end{aligned} \tag{8}$$

B. BACKWARD INDUCTION METHOD FOR THE GAME EQUILIBRIUM

We utilize the backward induction method to handle our two-stage Stackelberg game. So, at first, we need to analyze the non-cooperative subgame among smart vehicles at stage 2. And we define computing resource trading strategy $\boldsymbol{f} = (f_1, f_2, f_3, \ldots, f_N)$ $(0 \le f_i \le \bar{f}_i)$ a Nash Equilibrium (NE) if $U_i(f_i^*, F_{-i}^*(f_i^*)) \ge U_i(f_i, F_{-i}^*(f_i))$, that is $U_i'(f_i^*, F_{-i}^*(f_i^*)) \le U_i'(f_i, F_{-i}^*(f_i))$, where $F_{-i}^*(f_i)$ shows the optimal strategies of other sellers except for v_i when the trading strategy of v_i is f_i.

Theorem 1. *There exists a unique Nash Equilibrium strategy $\boldsymbol{f}^* = (f_1^*, f_2^*, \ldots, f_N^*)$ among the non-cooperative smart vehicles (sellers).*

Proof. In accordance with Eqs. (4), Eq. (6) and Eq. (8), $U_i'(f_i)$ is a continuous function. Thus, we derive the first-order and secondary-order derivatives of $U_i'(f_i)$ w.r.t. f_i that are described as

$$\frac{\partial U_i'}{\partial f_i} = (r_i - c_i)\left(1 + \frac{\gamma_i}{\bar{f}_i - f_i - \rho_i + 1} \right) - R_i + c_i \tag{9}$$

$$\frac{\partial^2 U_i'}{\partial f_i^2} = \frac{\gamma_i (r_i - c_i)}{(\bar{f}_i - f_i - \rho_i + 1)^2} > 0. \tag{10}$$

□

Therefore, according to Eqs. (9) and (10), we could prove the uniqueness of strategy $\boldsymbol{f} = (f_1, f_2, f_3, \ldots, f_N)$ in the subgame. In other words, a unique NE strategy among the smart vehicles (sellers) will be achieved. Let $\partial U_i' / \partial f_i = 0$, then we could obtain the optimal strategy via the Eq. (9)

$$f_i^* = \begin{cases} 0 & R_i < R_i' \\ \bar{f}_i - \rho_i + 1 + \frac{\gamma_i (r_i - c_i)}{r_i - R_i} & R_i' \le R_i \le R_i'' \\ \bar{f}_i & R_i'' < R_i \end{cases} \tag{11}$$

$$H_1 = \begin{pmatrix} \frac{2\gamma_i(r_1-c_1)(A-r_1)}{(R_1-r_1)(R_1-r_1)^2} & 0 & \cdots & 0 \\ 0 & \frac{2\gamma_i(r_2-c_2)(A-r_2)}{(R_2-r_2)(R_2-r_2)^2} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \frac{2\gamma_i(r_N-c_N)(A-r_N)}{(R_N-r_N)(R_N-r_N)^2} \end{pmatrix}$$

$$H_2 = \begin{pmatrix} \frac{\lambda_k\gamma_1\gamma_1 A(r_1-c_1)(r_1-c_1)}{(R_1-r_1)^2(R_1-r_1)^2} & \frac{\lambda_k\gamma_1\gamma_2 A(r_1-c_1)(r_2-c_2)}{(R_1-r_1)^2(R_2-r_2)^2} & \cdots & \frac{\lambda_k\gamma_1\gamma_N A(r_1-c_1)(r_N-c_N)}{(R_1-r_1)^2(R_N-r_N)^2} \\ \frac{\lambda_k\gamma_2\gamma_1 A(r_2-c_2)(r_1-c_1)}{(R_2-r_2)^2(R_1-r_1)^2} & \frac{\lambda_k\gamma_2\gamma_2 A(r_2-c_2)(r_2-c_2)}{(R_2-r_2)^2(R_2-r_2)^2} & \cdots & \frac{\lambda_k\gamma_2\gamma_N A(r_2-c_2)(r_N-c_N)}{(R_2-r_2)^2(R_N-r_N)^2} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\lambda_k\gamma_N\gamma_1 A(r_N-c_N)(r_1-c_1)}{(R_N-r_N)^2(R_1-r_1)^2} & \frac{\lambda_k\gamma_N\gamma_2 A(r_N-c_N)(r_2-c_2)}{(R_N-r_N)^2(R_2-r_2)^2} & \cdots & \frac{\lambda_k\gamma_N\gamma_N A(r_N-c_N)(r_N-c_N)}{(R_N-r_N)^2(R_N-r_N)^2} \end{pmatrix}.$$

$$\frac{\partial f_i^*(R_i)}{\partial R_i} = \frac{\gamma_i(r_i-c_i)}{(r_i-R_i)^2} > 0 \tag{12}$$

$$\begin{cases} R_i' = \frac{\gamma_i(r_i-c_i)}{f_i-\rho_i+1} + r_i & f_i^* = 0 \\ R_i'' = \frac{\gamma_i(r_i-c_i)}{-\rho_i+1} + r_i & f_i^* = \bar{f} \end{cases}. \tag{13}$$

According to Eq. (12), R_i increases with f_i^*. And f_i^* is limited to $[0,\bar{f}]$, so the corresponding response strategy is presented in a segmented form in Eq. (11). When $R_i \le R_i', f_i^* = 0$, this makes no sense in the subgame resulting from no computing resource trading. While $R_i'' \le R_i, f_i^* = \bar{f}_i$, the more pricing should be paid by the buyer without improving the computing resource amount. Thus, these two situations can not acquire maximal economic utility. To sum up, the best pricing strategy $R_i{}^*$ is situated between R_i' and R_i''.

Lemma 1. For the best pricing strategy $R_i{}^*$, there should exist the inequation: $R_i^* \ge R_i' \ge r_i \quad (\forall i \in \Omega)$

Proof. We could obtain the value of R_i' by Eq. (13). $\frac{\partial f_i^*}{\partial R_i} > 0$ according to Eq. (12), thus f_i^* varies monotonously with R_i. Let $f_i^* = 0$, $R_i' = \frac{\gamma_i(r_i-c_i)}{f_i-\rho_i+1} + r_i \ge r_i$. So we can prove the inequation in the Lemma 1. □

Theorem 2. There exists a unique Stackelberg Equilibrium among the buyer and sellers in our Stackelberg game.

Proof. Substituting Eq. (11) into Eq. (7), we could get $U'(\boldsymbol{R}, f^*(\boldsymbol{R}))$ is a function w.r.t. $\{R_1, R_2, R_3, \ldots, R_N\}$. We could get its Hessian matrix H via Eqs. (15) and (16). To simplify writing, we employ A to be on behalf of $\lambda_k a_k e^{\lambda_k \sum(\bar{f}_i - f_i^*)}$. Moreover, in accordance with Eqs. (11) and (12), we could divide H into H_1 and H_2: $H = H_1 + H_2$. The extension of H, H_1, and H_2 are shown as follows:

$$\frac{\partial U'}{\partial R_i} = f_i^* - \gamma_i(r_i - c_i)\frac{A - R_i}{(R_i - r_i)^2} \quad [R_i', R_i''] \tag{14}$$

$$\frac{\partial^2 U'}{\partial R_i \partial R_j} = \frac{\lambda_k \gamma_i \gamma_j A(r_i - c_i)(r_j - c_j)}{(R_i - r_i)^2 (R_j - r_j)^2} \tag{15}$$

$$\frac{\partial^2 U'}{\partial R_i{}^2} = \frac{\lambda_k \gamma_i^2 A(r_i - c_i)^2}{(R_i - r_i)^4} + \frac{2\gamma_i(r_i - c_i)(A - r_i)}{(R_i - r_i)(R_i - r_i)^2} \tag{16}$$

$$H = \begin{pmatrix} \frac{\partial^2 U'}{\partial R_1 \partial R_1} & \frac{\partial^2 U'}{\partial R_1 \partial R_2} & \cdots & \frac{\partial^2 U'}{\partial R_1 \partial R_N} \\ \frac{\partial^2 U'}{\partial R_2 \partial R_1} & \frac{\partial^2 U'}{\partial R_2 \partial R_2} & \cdots & \frac{\partial^2 U'}{\partial R_2 \partial R_N} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial^2 U'}{\partial R_N \partial R_1} & \frac{\partial^2 U'}{\partial R_N \partial R_2} & \cdots & \frac{\partial^2 U'}{\partial R_N \partial R_N} \end{pmatrix}.$$

□

Lemma 2. H_1 is a positive definite matrix in our game.

Proof. We just need to prove $A - r_i = \lambda_k a_k e^{\lambda_k \sum(\bar{f}_i - f_i^*)} - r_i \ge 0$, then we could prove H_1 is a positive definite matrix. In accordance with Eq. (14), let $\partial U'/\partial R_i = 0$, we could get optimal strategy R_i^* as follows:

$$A - R_i^* = \frac{f_i^*(R_i^* - r_i)^2}{\gamma_i(r_i - c_i)} \ge 0. \tag{17}$$

While $R_i^* \ge r_i$ in accordance with Lemma 1, thus we could prove that $A - r_i \ge A - R_i^* \ge 0$. Thus H_1 could be proved as a positive definite matrix. As for H_2, it is a real symmetric matrix, thus it should be a positive definite matrix, according to theorems in [38]. In addition, we could prove H is a strict positive definite matrix in accordance with Lemma 2, which shows the optimal strategy $\boldsymbol{R}^* = (R_1^*, R_2^*, \ldots, R_N^*)$ is unique. Moreover, it indicates that function $U'(\boldsymbol{R}, f^*(\boldsymbol{R}))$ is a convex function, and the problem (7) is a convex optimization problem. Meanwhile, Theorem 2 is also proved. □

C. GRADIENT-BASED SEARCHING ALGORITHM FOR EQUILIBRIUM

We could utilize the low complexity gradient based searching algorithm to find the unique Nash Equilibrium and Stackelberg Equilibrium in our proposed Stackelberg game for the unique $\boldsymbol{f}^* = (f_1^*, f_2^*, \ldots, f_N^*)$ and $\boldsymbol{R}^* = (R_1^*, R_2^*, \ldots, R_N^*)$. The gradient-based iterative searching algorithm is shown as the following Algorithm 2.

V. SECURITY ANALYSIS AND PERFORMANCE EVALUATION

In this section, we will analyse the security performance of our blockchain-based computing resource trading system and conduct numerical analysis to evaluate the incentive effects of our proposed price-based two-stage Stackelberg game.

Algorithm 2. Gradient-based iterative searching algorithm to find NE and SE

Input: a_k, λ_k, and $SOC_i = \{\bar{f}_i, \rho_i, r_i, c_i\}, \forall i \in \Omega$
Output: $\boldsymbol{f}^*$, $\boldsymbol{R}^*$, U, $\boldsymbol{U} = (U_1, U_2, \ldots, U_N)$
Initialization: Threshold ξ, step size θ, $n = 0$;
for Each computing resource seller v_i $(i \in \Omega)$ **do**
  Calculate R_i' and R_i'' based on $R_i' = \frac{\gamma_i(r_i - c_i)}{\bar{f}_i - \rho_i + 1} + r_i$, and $R_i'' = \frac{\gamma_i(r_i - c_i)}{-\rho_i + 1} + r_i$;
end for
Select initial input $\boldsymbol{R}^n = (R_1^n, R_2^n, \ldots, R_N^n)$;
Repeat
for Each computing resource seller v_i $(i \in \Omega)$ **do**
  If $R_i' \leq R_i^n \leq R_i''$ **then**
    $R_i^n = R_i^n$, calculate f_i^n according to Eq. (11): $f_i^n = \bar{f}_i - \rho_i + 1 + \frac{\gamma_i(r_i - c_i)}{r_i - R_i}$.
  else if $R_i^n > R_i''$ **then**
    $R_i^n = R_i'', f_i^n = \bar{f}_i$
  else if $R_i^n < R_i'$ **then**
    $R_i^n = R_i', f_i^n = 0$
end for
The buyer updates pricing strategy by gradient-based searching algorithm: $\boldsymbol{R}^{n+1} = \boldsymbol{R}^n - \theta \nabla U'(\boldsymbol{R}^n, \boldsymbol{f}^*(\boldsymbol{R}^n))$. Where $\nabla U'(\boldsymbol{R}^n, \boldsymbol{f}^*(\boldsymbol{R}^n))$ is the gradient based on Eqs. (11) and (14);
$n := n + 1$;
Until $\frac{\|f^{n+1} - f^n\|_1}{\|f^n\|_1} \leq \xi$; $\boldsymbol{R}^* = \boldsymbol{R}^{n+1} = \boldsymbol{R}^n$ is acquired;
Calculate $\boldsymbol{f}^*$, U, and $\boldsymbol{U} = (U_1, U_2, \ldots, U_N)$ according to Eqs. (3) and (6);

A. SECURITY ANALYSIS OF OUR TRADING SYSTEM

For traders in our system, security performance is a fundamental element for trading system availability. Besides, a secure and privacy-preserving trading system can also motivate individual users to participate in. The security performance of our trading system is achieved by the consortium blockchain technology. And the main security features are shown as follows.

1) ANONYMITY-BASED PRIVACY-PRESERVING

Similar to the conventional *Bitcoin* system, we also employ the anonymous public key as the wallet address in our trading system. On the one hand, it is difficult to find the true identity of smart vehicles or mobile users from the anonymous wallet addresses. On the other hand, each trader could also register multiple anonymous wallet addresses to further protect their identities and trading privacy.

2) TRADING PROCESS DECENTRALIZATION

Centralized trading systems are facing severe security threats, such as a single-point attack. As a decentralized trading system, our trading system utilizes consortium blockchain technology to avoid introducing a global trusted third-party. At the same time, all the trading nodes could conduct secure computing resource trading locally without sending the trading request and confirmation information to a centralized infrastructure, such as a public cloud platform, which greatly improves the trading efficiency of our system.

3) SYBIL-RESISTANCE CONSENSUS

In the pure PBFT protocol, the proportion of the number of nodes is very important, and the attacker can fake the identity to participate in the PBFT consensus process (election and voting), i.e., the *Sybil* attack. We use the PoW protocol commonly used in the public blockchain to select the leader node. The faster node that discovers the hash puzzle solution is selected as the leader node of the PBFT protocol. So the fake nodes with forged identities also need to deal with the hash puzzle to prove the adequate workload, thus resisting the *Sybil* attack. Meanwhile, in our system, the computing power of the MEC nodes themselves makes the implement of PoW protocol become possible.

4) TXS UNFORGEABLE AND TAMPER-PROOF

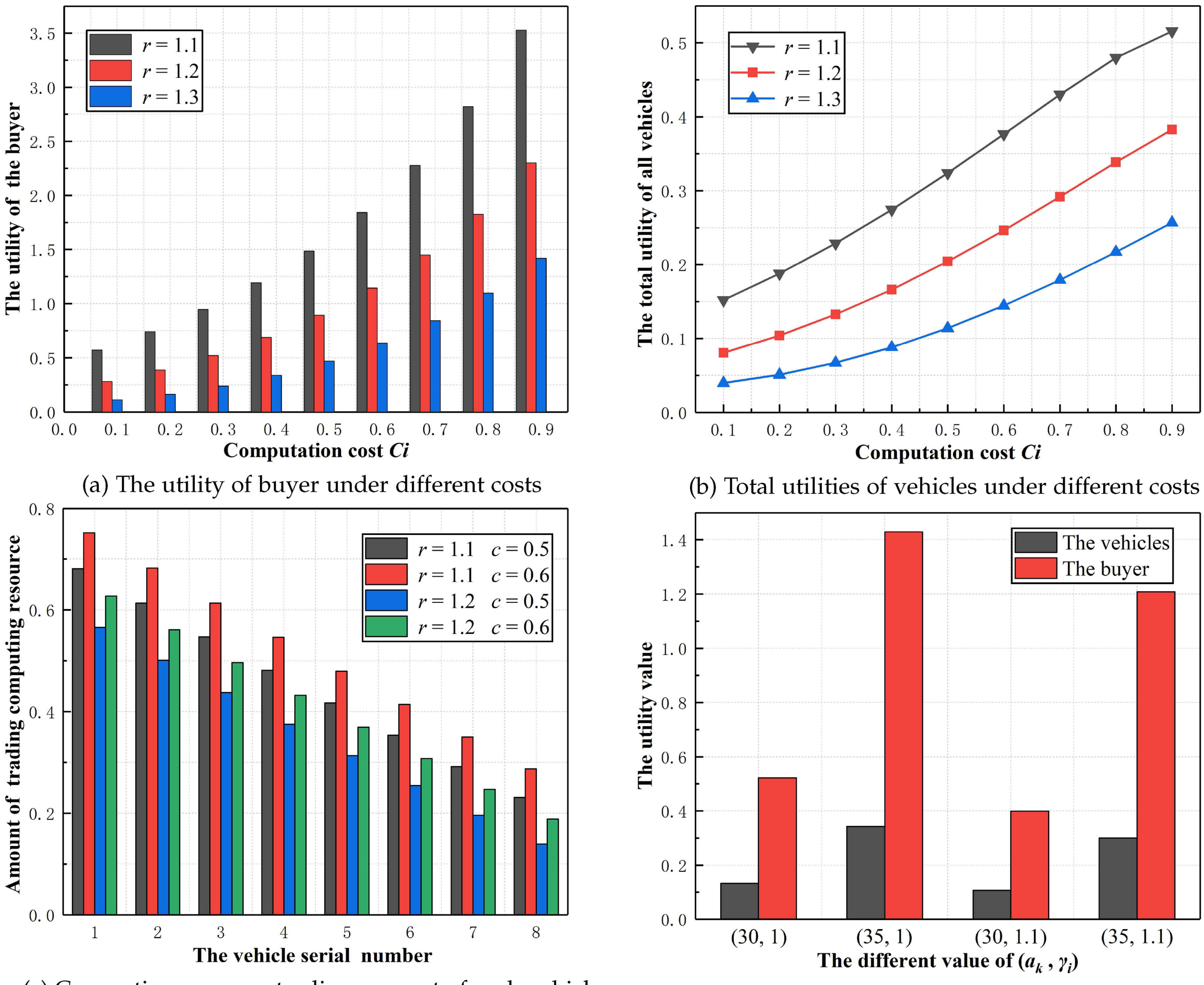
In our system, we employ the consortium blockchain technology, so the transaction data is maintained by all the MEC nodes in the smart city. According to the PBFT consensus protocol, as long as the total number of malicious nodes is less than $M/3$, the system will maintain the normal operation. We assume that there exists an attacker who would like to attack our trading system and tamper with the trading data. That is, this attacker is required to compromise at least 1/3 of the MEC nodes. The robustness of our trading system increases with the consensus MEC nodes. In the next 5G HetNet, the deployment of MEC-enabled base stations will become further densified. Therefore, we believe that the cost of compromising more than 1/3 of MEC nodes in the smart city is too high to afford. So our proposed system can realize TXs unforgeable and tamper-proof.

B. PERFORMANCE EVALUATION

In the performance evaluation of our proposed Stackelberg game, we suppose that there will exist 8 smart vehicles employed by a buyer in our system, which also could be extended. The maximal computing resource of each smart vehicle v_i is normalized to 1, while the computing resource demand is distinct from each other, $\rho = (0.1, 0.2, \ldots, 0.8)$. And a_k is the fixed coefficient of operation utility set to (30, 35) to show the difference between the buyers. The parameter λ_k is a fixed constant set to 0.05.

1) QUALITATIVE ANALYSIS OF OUR SCHEME

We give the qualitative analysis of our scheme in the Figure 5. Specifically, in Figure 5(a), we investigate the utility of the buyer w.r.t. different cost c_i and reward r_i. c_i varies from 0.1 to 0.9 to show the distinct computing costs. While the reward r_i should be larger than c_i, which is more realistic



(a) The utility of buyer under different costs

(b) Total utilities of vehicles under different costs

(c) Computing resource trading amount of each vehicle

(d) The utility under different system parameters

FIGURE 5. Qualitative analysis in our proposed Stackelberg game.

in real life. And r_i is set to (1.1, 1.2, 1.3). While Figure 5(b) studies the total utilities of all the 8 vehicles. Additionally, Figure 5(c) presents the computing resource trading amount of each vehicle in this case.

As shown in Figure 5(c), with a given the fixed computing cost c_i, if the reward r_i is larger, computing resource trading amount for each vehicle will decrease. It shows that vehicles (sellers) are unwilling to trade computing resources to the buyer in this case, for they could obtain greater benefits for their own use rather than trading to others, which is consistent with real life. And also the utility of the buyer is also decreasing for less computing resource trading amount as shown in Figure 5 (a). However, those vehicles will trade more computing resources to the buyer with the increasing cost c_i and fixed reward r_i, as illustrated in Figure 5(c). This means with the same reward r_i by own use, vehicles with larger cost c_i are prone to trading more computing resources to obtain more utilities from the buyer via the given pricing $\boldsymbol{R}^* = (R_1^*, R_2^*, \ldots, R_N^*)$. The total utilities of those vehicles will increase as presented in Figure 5 (b). Meanwhile, the utility of the buyer is also increasing as shown in Figure 5(a) resulting from more resources trading from those vehicles. In other words, all the participants would benefit from our computing resource trading process. Therefore, our proposed scheme could encourage the cooperation between the buyer and sellers to reach a win-win situation, which could be regarded as a strong incentive mechanism. In Figure 5(c), we could also show the influence of computing demands ρ_i. The computing resource demands for the first vehicle is 0.1, the second is 0.2, and so on. We find that smart vehicles with higher computing demands ρ_i for own use (such as, the eighth vehicle) trade less computing resources under the same r_i and c_i. That is, in order to meet their own computing demands, more computing resources would be remained by own use.

We also present the impact of different parameters (a_k, γ_i) in Figure 5(d). For the buyer, the larger a_k means higher operation cost on itself, it is necessary to reduce the operation cost by purchasing computing resources from others. Both the buyer and sellers will benefit from the win-win trading process. Thus, it will improve the utilities of both parties in the

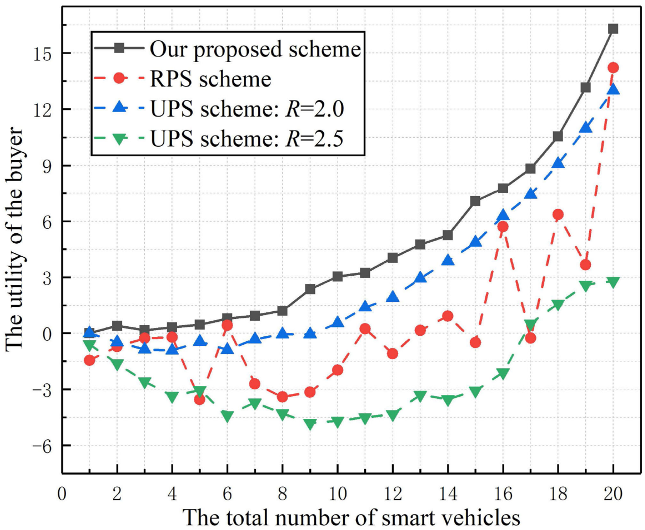


FIGURE 6. The utility of buyer in different schemes.

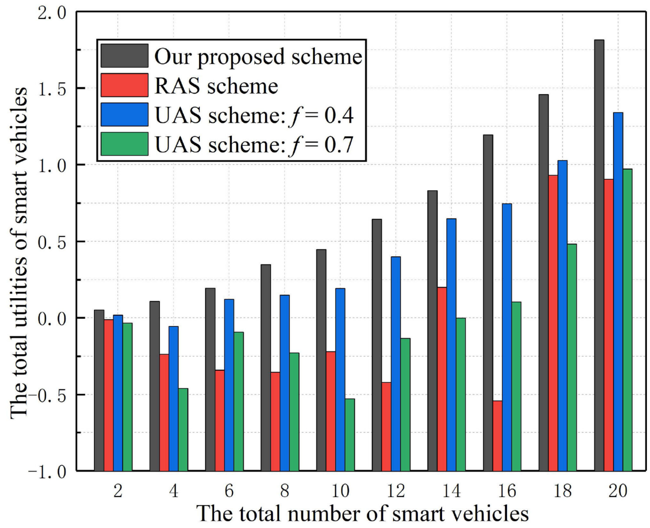


FIGURE 7. The utilities of vehicles in different schemes.

trading system as shown in Figure 5(d). For sellers, a higher γ_i means that they are more concerned about the satisfaction of their own resource use, which results in less computing resource trading amount. Both the buyer and sellers could not obtain more profits from the trading system. So, the utilities of both sides of the trading system will relatively reduce as shown in Figure 5(d).

2) COMPARATIVE ANALYSIS OF OUR SCHEME

Apart from the qualitative analysis for our scheme, in order to demonstrate the superiority of our optimized scheme, we also compare it with several other schemes, similar to [21]. In this case, $(a_k, \gamma_i) = (30, 1)$, while for each vehicle, c_i and ρ_i is randomly selected from (0, 1). And r_i is randomly selected from $[1, 2)$, which is higher than c_i.

To show the proposed optimized pricing strategy $\boldsymbol{R}^* = (R_1^*, R_2^*, \ldots, R_N^*)$, we compare it with random pricing scheme (RPS) and uniform pricing scheme (UPS), respectively. In RPS, we randomly select the discriminatory pricing strategy in $[R_i', R_i'']$, while we set uniform price $(R = 2.0, 2.5)$ in UPS. And we obtain $\boldsymbol{f}^*$ according to Eq. (11). As shown in Figure 6, the utility of the buyer in our scheme is higher than the RPS and UPS schemes. This is because our scheme is the optimal strategy between $[R_i', R_i'']$, so the performance is better than RPS. Compared to the RPS scheme, the UPS is closer to our scheme when $R = 2.0$. When R becomes larger $(R = 2.5)$, the performance of UPS scheme becomes worse. This is because a larger R will exceed the upper bound of some vehicles R_i'', which causes a saturated trading amount. In other words, larger pricing will be paid for computing resource trading by the buyer, while the trading amount is still limited to seller's largest computing resource $\bar{f}_i$. Thus, in this case, UPS scheme cannot motivate more computing resource trading amount but bring higher economic costs for the buyer.

And in order to indicate the optimized trading amount strategy $\boldsymbol{f}^* = (f_1^*, f_2^*, \ldots, f_N^*)$, we compare it with random amount scheme (RAS) and uniform amount scheme (UAS), respectively. We employ the optimized pricing strategy $\boldsymbol{R}^* = (R_1^*, R_2^*, \ldots, R_N^*)$ in RAS and UAS scheme. The trading amount scheme is chosen in $[0, 1]$ randomly in RAS scheme. And the UAS scheme sets the uniform trading amount as $(0.4, 0.7)$. As shown in Figure 7, neither RAS nor UAS can exceed the performance of our proposed scheme. That is, our computing resource trading amount scheme $\boldsymbol{f}^* = (f_1^*, f_2^*, \ldots, f_N^*)$ could optimize the trading utilities of smart vehicles compared with RAS and UAS scheme. Therefore, our trading amount scheme could also motivate smart vehicles to participate in the P2P trading system. Moreover, as shown in Figures 6 and 7, with the increasing number of vehicles, the utilities of the buyer and sellers also increase. This is because more vehicles will make a contribution to computing resource trading. It also presents that our proposed scheme has good scalability for more participating vehicles.

VI. CONCLUSION AND FUTURE WORK

In this paper, we have proposed an on-demand computing resource trading management system in IoV-assisted smart city. Besides, we employ consortium blockchain technology in our trading system, which can guarantee transaction security and privacy-preserving in a decentralized way. Then we also construct a two-stage Stackelberg game to stimulate the computing resource trading process among the buyer and sellers. In addition, we have proved the existence and uniqueness equilibrium of our game by backward induction method. Finally, we give security analysis and conduct numerical simulations to analyze the performance of our system. In future work, we will concentrate on the specific task allocation strategy in our system to achieve long-term stability in the spatio-temporal dynamic environment.

ACKNOWLEDGMENTS

This work was supported by the National Natural Science Foundation of China Under Grant 61972255.

REFERENCES

[1] T. Soyata, R. Muraleedharan, C. Funai, M. Kwon, and W. Heinzelman, “Cloud-vision: Real-time face recognition using a mobile-cloudlet-cloud acceleration architecture,” in *Proc. IEEE Symp. Comput. Commun.*, 2012, pp. 000059–000066.

[2] Y. Yang, Y. Xu, Y. Sun, Y. Dong, F. Wu, and Y. T. Zhuang, “Mining fraudsters and fraudulent strategies in large-scale mobile social networks,” *IEEE Trans. Knowl. Data Eng.*, to be published, doi: 10.1109/TKDE.2019.2924431

[3] P. Mach and Z. Becvar, “Mobile edge computing: A survey on architecture and computation offloading,” *IEEE Commun. Surveys Tuts.*, vol. 19, no. 3, pp. 1628–1656, Third Quarter 2017.

[4] P. Bellavista, S. Chessa, L. Foschini, L. Gioia, and M. Girolami, “Human-enabled edge computing: Exploiting the crowd as a dynamic extension of mobile edge computing,” *IEEE Commun. Mag.*, vol. 56, no. 1, pp. 145–155, Jan. 2018.

[5] N. Abbas, Y. Zhang, A. Taherkordi, and T. Skeie, “Mobile edge computing: A survey,” *IEEE Internet Things J.*, vol. 5, no. 1, pp. 450–465, Feb. 2018.

[6] T. G. Rodrigues, K. Suto, H. Nishiyama, N. Kato, and K. Temma, “Cloudlets activation scheme for scalable mobile edge computing with transmission power control and virtual machine migration,” *IEEE Trans. Comput.*, vol. 67, no. 9, pp. 1287–1300, Sep. 1, 2018.

[7] M. Sookhak *et al.*, “Fog vehicular computing: Augmentation of fog computing using vehicular cloud computing,” *IEEE Veh. Technol. Mag.*, vol. 12, no. 3, pp. 55–64, Sep. 2017.

[8] R. Yu, J. Ding, X. Huang, M. Zhou, S. Gjessing, and Y. Zhang, “Optimal resource sharing in 5G-enabled vehicular networks: A matrix game approach,” *IEEE Trans. Veh. Technol.*, vol. 65, no. 10, pp. 7844–7856, Oct. 2016.

[9] X. Lyu, C. Ren, W. Ni, H. Tian, and R. P. Liu, “Distributed optimization of collaborative regions in large-scale inhomogeneous fog computing,” *IEEE J. Sel. Areas Commun.*, vol. 36, no. 3, pp. 574–586, Mar. 2018.

[10] R. Chaudhary, N. Kumar, and S. Zeadally, “Network service chaining in fog and cloud computing for the 5G environment: Data management and security challenges,” *IEEE Commun. Mag.*, vol. 55, no. 11, pp. 114–122, Nov. 2017.

[11] X. Lin, J. Li, W. Yang, J. Wu, Z. Zong, and X. Wang, “Vehicle-to-Cloudlet: Game-based computation demand response for mobile edge computing through vehicles,” in *Proc. IEEE 89th Veh. Technol. Conf.*, 2019, pp. 1–6.

[12] S. Olariu, I. Khalil, and M. Abuelela, “Taking VANET to the clouds,” *Int. J. Pervasive Comput. Commun.*, vol. 7, no. 1, pp. 7–21, 2011.

[13] S. Nakamoto, “Bitcoin: A peer-to-peer electronic cash system,” 2008. [Online]. Available: https://bitcoin.org/bitcoin.pdf

[14] G. Wood, “Ethereum: A secure decentralised generalised transaction ledger,” *Ethereum Project Yellow Paper*, vol. 151, 2014.

[15] S. Olariu, M. Eltoweissy, and M. Younis, “Towards autonomous vehicular clouds,” *EAI Endorsed Trans. Mobile Commun. Appl.*, vol. 1, no. 1, 2011, Art. no. e2.

[16] H. Zhang, Q. Zhang, and X. Du, “Toward vehicle-assisted cloud computing for smartphones,” *IEEE Trans. Veh. Technol.*, vol. 64, no. 12, pp. 5610–5618, Dec. 2015.

[17] X. Hou, Y. Li, M. Chen, D. Wu, and D. Jin, “Vehicular fog computing: A viewpoint of vehicles as the infrastructures,” *IEEE Trans. Veh. Technol.*, vol. 65, no. 6, pp. 3860–3873, Jun. 2016.

[18] S. Liao, J. Li, J. Wu, W. Yang, and Z. Guan, “Fog-enabled vehicle as a service for computing geographical migration in smart cities,” *IEEE Access*, vol. 7, pp. 8726–8736, 2019.

[19] S. S. Shah, M. Ali, A. W. Malik, M. A. Khan, and S. D. Ravana, “vFog: A vehicle-assisted computing framework for delay-sensitive applications in smart cities,” *IEEE Access*, vol. 7, pp. 34900–34909, 2019.

[20] S. Kekki *et al.*, “MEC in 5G networks,” ETSI, Sophia Antipolis, France, White Paper, 2018.

[21] M. Chen and Y. Hao, “Task offloading for mobile edge computing in software defined ultra-dense network,” *IEEE J. Sel. Areas Commun.*, vol. 36, no. 3, pp. 587–597, Mar. 2018.

[22] Y. Dai, D. Xu, S. Maharjan, and Y. Zhang, “Joint computation offloading and user association in multi-task mobile edge computing,” *IEEE Trans. Veh. Technol.*, vol. 67, no. 12, pp. 12313–12325, Dec. 2018.

[23] C. Zhang, Z. Liu, B. Gu, K. Yamori, and Y. Tanaka, “A deep reinforcement learning based approach for cost- and energy-aware multi-flow mobile data offloading,” *IEICE Trans. Commun.*, vol. E101-B, no. 7, pp. 1625–1634, Jul. 2018.

[24] L. Gkatzikis and I. Koutsopoulos, “Migrate or not? Exploiting dynamic task migration in mobile cloud computing systems,” *IEEE Wireless Commun.*, vol. 20, no. 3, pp. 24–32, Jun. 2013.

[25] M. Mishra, A. Das, P. Kulkarni, and A. Sahoo, “Dynamic resource management using virtual machine migrations,” *IEEE Commun. Mag.*, vol. 50, no. 9, pp. 34–40, Sep. 2012.

[26] C. Zhang and Z. Zheng, “Task migration for mobile edge computing using deep reinforcement learning,” *Future Gener. Comput. Syst.*, vol. 96, pp. 111–118, Jul. 2019.

[27] M. Jia, W. Liang, Z. Xu, and M. Huang, “Cloudlet load balancing in wireless metropolitan networks,” in *Proc. IEEE 35th Int. Conf. Comput. Commun.*, Apr. 2016, pp. 1–9.

[28] F. Gao, L. Zhu, M. Shen, K. Sharif, Z. Wan, and K. Ren, “A blockchain-based privacy-preserving payment mechanism for vehicle-to-grid networks,” *IEEE Netw.*, vol. 32, no. 6, pp. 184–192, Nov./Dec. 2018.

[29] Z. Li, J. Kang, R. Yu, D. Ye, Q. Deng, and Y. Zhang, “Consortium blockchain for secure energy trading in industrial Internet of Things,” *IEEE Trans. Ind. Informat.*, vol. 14, no. 8, pp. 3690–3700, Aug. 2018.

[30] M. Li, L. Zhu, and X. Lin, “Efficient and privacy-preserving carpooling using blockchain-assisted vehicular fog computing,” *IEEE Internet Things J.*, vol. 6, no. 3, pp. 4573–4584, Jun. 2019.

[31] Z. Guan *et al.*, “ECOSECURITY: Tackling challenges related to data exchange and security: An edge-computing-enabled secure and efficient data exchange architecture for the energy Internet,” *IEEE Consum. Electron. Mag.*, vol. 8, no. 2, pp. 61–65, Mar. 2019.

[32] X. Lin, J. Li, J. Wu, H. Liang, and W. Yang, “Making knowledge tradable in edge-AI enabled IoT: A consortium blockchain-based efficient and incentive approach,” *IEEE Trans. Ind. Informat.*, vol. 15, no. 12, pp. 6367–6378, Dec. 2019, doi: 10.1109/TII.2019.2917307

[33] Z. Xiong, Y. Zhang, D. Niyato, P. Wang, and Z. Han, “When mobile blockchain meets edge computing,” *IEEE Commun. Mag.*, vol. 56, no. 8, pp. 33–39, Aug. 2018.

[34] Y. Jiao, P. Wang, D. Niyato, and Z. Xiong, “Social welfare maximization auction in edge computing resource allocation for mobile blockchain,” in *Proc. IEEE Int. Conf. Commun.*, 2018, pp. 1–6.

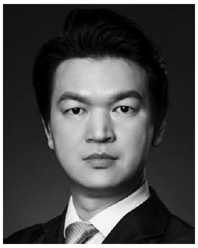
[35] M. Castro, B. Liskov *et al.*, “Practical byzantine fault tolerance,” in *Proc. 3rd Symp. Operating Syst. Des. Implementation*, 1999, vol. 99, pp. 173–186.

[36] Y. Liu, S. Hu, H. Huang, R. Ranjan, A. Y. Zomaya, and L. Wang, “Game-theoretic market-driven smart home scheduling considering energy balancing,” *IEEE Syst. J.*, vol. 11, no. 2, pp. 910–921, Jun. 2017.

[37] Y. Wang, Z. Su, Q. Xu, T. Yang, and N. Zhang, “A novel charging scheme for electric vehicles with smart communities in vehicular networks,” *IEEE Trans. Veh. Technol.*, vol. 68, no. 9, pp. 8487–8501, Sep. 2019.

[38] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge, U.K.: Cambridge Univ. Press, 2004.

XI LIN (Student Member, IEEE) received the BS degree from the School of Precision Instrument and Opto-Electronics Engineering, Tianjin University, Tianjin, China, in 2016. He is currently working toward the PhD degree in the School of Electronic Information and Electrical Engineering, Shanghai Jiao Tong University, Shanghai, China. His research interests include multiaccess edge computing, Internet of Things, Blockchain, and so on.



JUN WU (Member, IEEE) received the PhD degree in information and telecommunication studies from Waseda University, Japan, in 2011. He is currently an associate professor of School of Electronic Information and Electrical Engineering, Shanghai Jiao Tong University, China. He is the chair of IEEE P21451-1-5 Standard Working Group. His research interests include the advanced computing, communications and security techniques of SDN, ICN networks smart grids, Internet of Things, 5G, etc. He has been the TPC member of more than ten international conferences including ICC, GLOBECOM, etc. He has been a guest editor of the *IEEE Sensors Journal*, the *Sensors*, the *ICT Express*. He is an associate editor of the *IEEE Access*.

SHAHID MUMTAZ (Senior Member, IEEE) received the master's and PhD degrees in electrical and electronic engineering from the Blekinge Institute of Technology, Sweden, and the University of Aveiro, Portugal, in 2006 and 2011, respectively. He is an ACM distinguished speaker, EiC of IET Journal of Quantum Communication; vice chair: Europe/Africa Region-IEEE ComSoc: Green Communications & Computing society and vice-chair for IEEE standard on P1932.1: Standard for Licensed/Unlicensed Spectrum Interoperability in Wireless Mobile Networks. He has been with the Instituto de Telecomunicaçaões since 2011, where he currently holds the position of auxiliary researcher and adjunct positions with several universities across the Europe-Asian Region. He is also a visiting researcher at Nokia Bell Labs. He is the author of four technical books, 12 book chapters, and more than 150 technical papers in the area of mobile communications.

SAHIL GARG (Member, IEEE) received the PhD degree from the Thapar Institute of Engineering and Technology, Patiala, India, in 2018. He is currently working as a postdoctoral research fellow with the Department of Electrical Engineering, École de technologie supérieure, Université du Québec, Montréal, Canada. He has many research contributions in the area of machine learning, knowledge discovery, cloud computing, Internet of Things, and vehicular ad-hoc networks. Some of his research findings are published in top-cited journals such as the *IEEE Transactions on Industrial Informatics*, the *IEEE Transactions on Multimedia*, the *IEEE Transactions on Vehicular Technology*, the *IEEE IoT Journal*, the *IEEE Communications Magazine*, the *IEEE Network*, the *IEEE Wireless Communications*, the *Future Generation Computing Systems*, the *Information Sciences*, and the *Computers & Electrical Engineering* including various International conferences of repute such as-IEEE Globecom, IEEE ICC, IEEE Infocom Workshops, ACM Mobicom Workshops, etc. He also received the IEEE ICC Best Paper Award, in 2018. He is a member of IEEE ComSoc, IEEE Computer, IEEE IES, IEEE Smart Grid Community, ACM, and IAENG.

JIANHUA LI (Member, IEEE) received the BS, MS, and PhD degrees from Shanghai Jiao Tong University, Shanghai, China, in 1986, 1991, and 1998, respectively. He is a professor/PhD supervisor of School of Cyber Security, Shanghai Jiao Tong University, China. He was the leader of more than 30 state/province projects of China, and published more than 300 papers. He published six books and has about 20 patents. He made three standards and has five software copyrights. His research interests include information security, computer network communication, etc.

MOHSEN GUIZANI (Fellow, IEEE) received the BS and MS degrees in electrical engineering, the MS and PhD degrees in computer engineering from Syracuse University, Syracuse, New York, in 1984, 1986, 1987, and 1990, respectively. He is currently a professor with the CSE Department, Qatar University, Qatar. His research interests include wireless communications and mobile computing, computer networks, security, and smart grid. He is currently the editor-in-chief of the *IEEE Network*, serves on the editorial boards of several international technical journals and the founder and editor-in-chief of the *Wireless Communications and Mobile Computing Journal* (Wiley). He was the chair of the IEEE Communications Society Wireless Technical Committee and the chair of the TAOS technical committee. He is a senior member of ACM.