This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TVT.2020.3041587, IEEE Transactions on Vehicular Technology

A Game Theoretic Scheme for Collaborative Vehicular Task Offloading in 5G HetNets

Yilong Hui, *Member, IEEE,* Zhou Su, *Senior Member, IEEE,* Tom H. Luan, *Senior Member, IEEE,* Changle Li, *Senior Member, IEEE,* Guoqiang Mao, *Fellow, IEEE* and Weigang Wu, *Member, IEEE*

***Abstract*—The 5G heterogeneous networks (HetNets) are capable of providing real-time computing services for autonomous vehicles (AVs) by deploying edge computing devices (ECDs) at macro cell base stations (MCBSs) and small cell base stations (SCBSs). With the imbalanced distribution and fast moving AVs contending intensely for computing services, how to efficiently exploit cooperations among participants in 5G HetNets to improve the service performance is therefore challenging. In this paper, we develop a game theoretic scheme for collaborative vehicular task offloading to facilitate the computing services in 5G HetNets. Specifically, we propose a two-stage vehicular task offloading mechanism to promote the cooperation among participants with the target of improving the task completion rate and the utilities of the participants, where the mechanism jointly considers the network architecture of the HetNets, the imbalanced distribution of AVs and the reuse of task results. In the first stage, an auction model is designed to help the MCBS select the optimal SCBS to execute the offloaded task based on the requirement of the task and the available computing resources of SCBSs. According to the task execution cost declared by the selected SCBS, the MCBS then bargains with the AV for the agreement of the task offloading service to maximize their utilities in the second stage. Using simulations, we show that the proposed collaborative task offloading scheme can achieve a higher task completion rate for the task offloading service and bring higher utilities to all participants than conventional schemes.**

***Index Terms*—5G heterogeneous networks, vehicular task offloading, edge computing, game theory.**

I. Introduction

RECENT years have witnessed the breakthrough of autonomous driving worldwide which could potentially change the way of our transportation and daily life [1]–[3].

This work was supported in part by the National Key R&D Program of China under Grant 2019YFB1600100, in part by the NSFC under Grant 61901341, Grant U1808207, Grant U1801266 and Grant 91746114, in part by the Fundamental Research Funds for the Central Universities of Ministry of Education of China under Grant XJS200109, in part by the National Natural Science Foundation of Shaanxi Province under Grant 2020JQ-301, in part by the 111 Project under Grant D18003, in part by the Project of Shanghai Municipal Science and Technology Commission under Grant 18510761000, in part by the Key R&D Program of Shaanxi under Grant 2018ZDXM-GY-038 and 2018ZDCXL-GY-04-02, in part by the Youth Innovation Team of Shaanxi Universities, and in part by the Science and Technology Projects of Xi'an, China under Grant 201809170CX11JC12. (Corresponding author: Changle Li.)

Y. Hui, C. Li and G. Mao are with the State Key Laboratory of Integrated Services Networks, Xidian University, Xi'an 710071, China (e-mail: ylhui@xidian.edu.cn; clli@mail.xidian.edu.cn; gqmao@xidian.edu.cn).

Z. Su is with the School of Mechatronic Engineering and Automation, Shanghai University, Shanghai 200444, China (e-mail: zhousu@ieee.org).

T. H. Luan is with the School of Cyber Engineering, Xidian University, Xi'an 710071, China (e-mail: tom.luan@xidian.edu.cn).

W. Wu is with the School of Data and Computer Science, Sun Yat-sen University, Guangzhou 510006, China (e-mail: wuweig@mail.sysu.edu.cn).

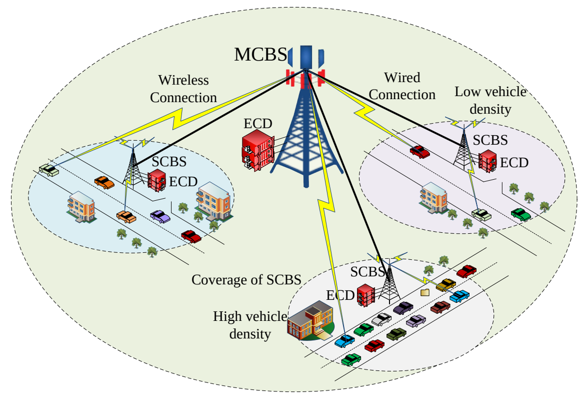


Fig. 1. Illustration of the collaborative vehicular task offloading in HetNets.

For example, with vehicles becoming self-driving, people can be released to work or play video games on the trip and let the vehicle drive and park by itself. Never feeling tired and distracted, the autonomous driving is much safe and can reduce energy consumption with better planned routes and driving strategies. For AVs driving on the roads, a huge body of computing tasks need to be executed to facilitate computationally intensive applications. The tasks, such as camera-based traffic monitoring, sensed data processing and artificial intelligence, typically have high computational demands which pose challenges to the autonomous driving [4].

To meet the ever increasing demands of AVs, the integration of edge computing and 5G heterogeneous networks (HetNets) is advocated to provide task computing services to facilitate autonomous driving [5]–[7]. In 5G HetNets, as shown in Fig. 1, several small cell base stations (SCBSs) are covered by the communication coverage of a macro cell base station (MCBS) [8]–[10]. By deploying edge computing devices (ECDs) at the MCBS and SCBSs, an AV can offload its computing task to the MCBS or its connected SCBS to improve the driving experience. Compared to the cloud server, the MCBS and SCBSs are in proximity of AVs so that they can provide services to AVs at the edge of the networks. With the rich computing resources provided by the ECDs deployed at MCBS and SCBSs, the long latency involved in remote transmissions can be reduced [11]–[13].

Motivated by the aforementioned observations, edge computing and computation offloading schemes have been extensively studied to facilitate different vehicular applications [14]–[21]. However, few of them concurrently consider the imbalanced distribution of AV's task computing requests, the

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TVT.2020.3041587, IEEE Transactions on Vehicular Technology

reuse of task results as well as the dynamic cooperation among participants including AVs, SCBSs and MCBS. In general, an AV with a high driving speed may leave the coverage of its connected SCBS before receiving the result of the task. In addition, AVs are not evenly distributed in the coverage of the MCBS [22]. Consequently, in the coverage of the SCBSs with a high vehicle density, the available computing resources of the SCBSs may not be able to serve all the tasks. On the contrary, in the coverage of the SCBSs with a low vehicle density, the computing resources may not be fully utilized. More importantly, the reuse of task results is rarely mentioned in existing works. In practice, the results or partial results of some computing tasks, such as the amount of traffic on a road, can often be reused for a period of time. As such, the utilization of computing resources can be significantly improved. Besides, the tasks requested by AVs typically have stringent time constraints (represented by a time to live (TTL) metric) and the base stations with different computing abilities may declare different costs for outsourcing computing services. For the AVs, they intend to execute tasks within the TTL and minimize the costs. In contrast, the base stations intend to maximize their profits. The negotiation of the transaction for the task computing outsourcing therefore needs to be studied with the target of improving their utilities. To summarize, an integrated task offloading architecture is needed for 5G HetNets to help the participants work cooperatively to compute tasks and enhance their utilities.

To this end, we propose a game theoretic scheme for collaborative vehicular task offloading in 5G HetNets. Specifically, by considering the network architecture of the HetNets, the imbalanced distribution of task requests and the reuse of task results, a two-stage task offloading mechanism is designed to promote the cooperation among the participants. With such a mechanism, the participants can cooperate with each other to compute the task, where the efficient scheduling can adjust the distribution of task computing requests and make full use of the resources in the networks. In the first stage, we formulate the second price sealed auction (SPSA) model to help the MCBS make the task allocation strategy, where the strategy is used to select the optimal SCBS to execute the offloaded task based on the requirement of the task and the available computing resources. By doing this, the task can be completed within the TTL and the cost for completing the task can be minimized. According to the cost of completing the task bade by the optimal SCBS, the MCBS and the AV then bargain with each other to reach an agreement for the task offloading service in the second stage to maximize their utilities. Our main contributions are three-fold.

- **Mechanism design:** We propose a two-stage task offloading mechanism to provide computing services for AVs in 5G HetNets. With the designed task offloading mechanism, the distribution of AVs' task computing requests can be balanced and the task completion rate can be improved.
- **Problem formulation:** Based on the designed mechanism, the utility optimization problems for the SCBSs, the MCBS and the AV are formulated by jointly considering the requirement of the task, the reuse of the task result and the available computing resources in the networks.
- **Game analysis:** To solve the formulated problems and promote the cooperation among the participants, the interactions among the SCBSs and the interactions between the MCBS and the AV are modeled as the SPSA and a two-round bargaining game, where the optimal strategies of them are respectively obtained by analyzing the two game models to maximize their utilities.

The remainder of this paper is organized as follows. Section II reviews the related works. The system model is presented in Section III. Section IV presents the proposed game theoretic task offloading scheme in detail. Section V evaluates the proposed scheme by simulations, and Section VI closes the paper with conclusions.

II. Related Work

A. Task Offloading Schemes in Vehicular Networks

In vehicular networks, the task offloading schemes have been extensively studied. To make full use of the computing resources, Guo *et al*. [23] developed a collaborative task offloading scheme in vehicular networks. With the designed scheme, the overall processing delay can be minimized. By considering the realistic environment, Li *et al*. [24] developed a coding-based data offloading scheme in vehicular networks, where the offloading problem is formulated as a utility maximization problem and solved by designing an algorithm to decide the optimal coding policy. Sun *et al*. [25] proposed a cooperative task scheduling scheme for computation offloading in vehicular networks, where the task is divided into subtasks to minimize the execution time. Zhou *et al*. [26] presented a reliable task offloading scheme in vehicular networks by analyzing the information asymmetry and information uncertainty. In this scheme, a stable task offloading algorithm is designed to minimize the network delay. By combining the cloud and edge computing, Zhao *et al*. [27] studied a collaborative offloading problem to support the computation offloading services in vehicular networks, where the problem is solved by designing a distributed optimization algorithm. Sun *et al*. [28] proposed a learning-based task offloading scheme which considers the dynamic and uncertain vehicular environment. The simulation results demonstrate that the proposed scheme can reduce the offloading delay compared with the existing algorithm.

Different from these existing works, the vehicular task offloading scheme proposed in our paper focuses on the cooperation among participants in the task offloading process to improve the task completion rate and enhance their utilities. In the proposed scheme, the utilities of the participants are analyzed by jointly considering the imbalanced distribution of the task requests, the reuse of the task results and the available resources of the MCBS and the SCBSs.

B. Game-based Task Offloading Schemes

There have been a lot of works focusing on the game-based task offloading schemes to complete computing services. Cao *et al*. [29] formulated the offloading decision making problem as a non-cooperative game, where the Nash equilibrium of

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TVT.2020.3041587, IEEE Transactions on Vehicular Technology

TABLE I
SUMMARY OF NOTATIONS

Notations	Description
$\mathbb{K}$, $\mathbb{I}$, $\mathbb{Q}$	Sets of SCBSs, AVs and tasks, respectively.
$\widetilde{\mathbb{K}}$	Set of SCBSs which can complete the task within the TTL.
$\overline{s_q}$, s_q	Input size and output size of task q.
T_q, $\overline{T_{i,q}}$	TTL of task q and TTL requested by AV i.
p_q	Popularity of task q's result.
D_q	Number of resources that task q needs to be used.
$s_{\min}, s_{\max}$	Minimum and maximum values of $\overline{s_q}$ and s_q.
r_i	Transmission rate between the MCBS and AV i.
$r_{\min}, r_{\max}$	Minimum and maximum values of r_i.
$D_{\min}, D_{\max}$	Minimum and maximum values of D_q.
$T_{\min}, T_{\max}$	Minimum and maximum values of T_q.
ϕ	Parameter of the Zipf-like distribution.
ς	Cost of the MCBS for delivering task per unit time.
$\mathbf{P}_{i,q}$	Transaction price that AV i pays for the service.
$C_{MC,q}$	Cost spent on completing the task.
$d_{k,q}$	Idle computing resources owned by SCBS k.
$d_{\min}, d_{\max}$	Minimum and maximum values of $d_{k,q}$.
$C_{i,q}$	Reserve price of AV i for computing the task.
$\lambda_{k,q}$	Cost of SCBS k for computing the task.
$C_{k,q}(\lambda_{k,q})$	Bid of SCBS k for computing the task in the SPSA game.
$B_{k^*,q}$	Price that the MCBS pays the selected SCBS k^*.
ψ_{MC}, ψ_i	The discounting rates of the MCBS and AV i.
$\Gamma_{k,q}\|_g$	The proposed profit distribution strategy of the MCBS for the task offloading service in round g.
$\mathrm{P}_{i,q}\|_g$	Reserve price of AV i for the task offloading service in round g estimated by the MCBS.
$p_a\|_g$, $p_r\|_g$	Probabilities that AV i accepts and rejects $\Gamma_{k,q}\|_g$.
$X_{MC,q}$	Utilities of the MCBS.
$X_{i,q}$	Utilities of AV i.
$X_{k^*,q}$	Utilities of the selected SCBS k^*.

the game is obtained by designing a machine learning-based computation offloading algorithm. Liwang *et al*. [30] developed a truthful reverse auction mechanism for computation offloading in vehicular networks. The scheme considers the opportunistic contacts between vehicles while satisfying the properties of truthfulness and individual rationality. Cheng *et al*. [31] proposed an auction game-based offloading scheme to offload the cellular traffic through carrier-WiFi networks. Through simulations, the authors show that the scheme can achieve lower service delay than the existing mechanism. Wang *et al*. [32] proposed a market framework to price the offloading service, where the interactions between the service providers and the service consumers are formulated as a multi-leader multi-follower Stackelberg game. By taking the selfishness into account, Zheng *et al*. [33] modeled the offloading decision process of mobile users as a stochastic game, where the Nash equilibrium of the game is reached by using a multi-agent stochastic learning algorithm. To minimize the task's computation time and energy consumption, Hong *et al*. [34] formulated the multi-hop computation offloading problem as a potential game, where the QoS-aware distributed algorithm is designed to achieve the game's Nash equilibrium. Peng *et al*. [35] proposed a multiattribute-based double auction mechanism to facilitate the task offloading in vehicular networks. With the designed mechanism, ECDs can serve the vehicles by providing the requested tasks and attributes.

Although the game-based task offloading schemes have been widely studied, few of them consider the cooperation among the participants (i.e., MCBS, SCBSs and AVs) in the 5G HetNets to promote the task offloading performance. In contrast to the existing works, the imbalanced distribution of computing requests, the reuse of task results and the cooperation among the participants are jointly considered in our paper to improve the task completion rate and enhance their utilities for the task offloading services.

III. System Model

In this section, we model the system of the proposed collaborative vehicular task offloading scheme in 5G HetNets. The notations used in this paper are summarized in Table I.

A. Network Model

In the 5G HetNets, as shown in Fig. 1, there are several SCBSs covered by a MCBS. Let $\mathbb{K} = \{1, .., k, ..., K\}$ and $\mathbb{I} = \{1, ..., i, ..., I\}$ be the sets of SCBSs and AVs in the communication coverage of the MCBS. The ECDs are deployed at base stations to provide AVs with task computing services, where the computing ability of SCBS k is determined by its idle computing resources. The SCBSs distributed in the networks are connected to the MCBS with high speed wired links. As the communication coverage of the SCBSs is limited while the driving speed of AVs is high, an AV which offloads the task to its connected SCBS may leave the SCBS's coverage before the task is completed. In addition, the distribution of task computing requests generated by AVs are usually imbalanced in the coverage of the MCBS. Therefore, in this paper, we focus on the scenario that the task of each AV is offloaded to the MCBS. Specifically, the MCBS is responsible for scheduling the tasks requested by AVs with the target of adjusting the distribution of computing requests and making full use of the resources in the networks. For AVs, they are equipped with on-board unit (OBU) and thus can communicate with the MCBS for requesting task computing services [36]. If an AV has a task needs to be computed, the AV can offload the task to the MCBS using wireless communication. Apart from the computing and transmission, the MCBS and the SCBSs also have the caching ability to store the computing results that are frequently used. On the other hand, some of the requested tasks have been partially computed, where the results and the associated tasks are also cached in the networks to serve AVs. For example, the task of traffic flow can be divided into several subtasks, where each subtask is in charge of one road. If the result of a subtask is obtained in advance, it can be used to support the task directly. If one of these tasks is required by an AV, the rest of the requested task will be computed. In this way, the task results can be reused and the utilization of the computing resources can be improved.

B. Task Model

To achieve different computing goals for the autonomous driving, computing tasks requested by AVs may have different requirements and values. Let $\mathbb{Q} = \{1, ..., q, ..., Q\}$ be the set of tasks that can be requested by AVs in the networks. For task $q (q \in \mathbb{Q})$, it can be characterized by a tuple

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TVT.2020.3041587, IEEE Transactions on Vehicular Technology

$(\overline{s_q}, \underline{s_q}, D_q, T_q, p_q)$, where $\overline{s_q}$ and $\underline{s_q}$ are the input size and the output size of the task [37]. Both $\overline{s_q}$ and $\underline{s_q}$ range from $s_{\min}$ to $s_{\max}$. D_q is the computing resources that task q needs to use for completing the task, which follows a uniform distribution within $[D_{\min}, D_{\max}]$. T_q is the TTL associated with task q. The result of the task will become useless if the time exceeds the task's TTL. p_q is the popularity of task q's result. For ease of analysis, p_q is determined by the Zipf-like distribution in this paper [38]–[40]. We have

$$p_q = 1/\left(\widetilde{q}^{\phi}\sum_{\varrho=1}^{Q}\frac{1}{\varrho^{\phi}}\right). \tag{1}$$

Here, $\widetilde{q}$ is the rank of task q. ϕ is the parameter to charge the distribution and ranges from 0 to 1. For example, the distribution follows the strict Zipf distribution if $\phi = 1$ while it follows the uniform distribution if $\phi = 0$.

IV. Problem Formulation and Game Analysis

In this section, we first introduce the two-stage task offloading mechanism, followed by the problem formulation and game analysis for collaborative vehicular task offloading.

A. Two-stage Task Offloading Mechanism

Consider that AV i is in the coverage of the MCBS and intends to offload task q, the AV then sends a task computing request to the MCBS using the wireless connection. The request includes the AV's identification number, $\overline{s_q}$, D_q and $T_{i,q}$, where $T_{i,q}$ is the TTL of the task requested by AV i, we have $T_{i,q} \le T_q$. After receiving the request, the MCBS then makes a task allocation strategy based on the following cases.

Case 1 : The task result is cached in the MCBS.

In this case, if the task result is available in the MCBS's cache and the result can be transmitted to the AV within the requested TTL, the mechanism moves to the second stage directly. In the second stage, the MCBS negotiates with AV i for the transaction price. If an agreement between the MCBS and AV i is achieved, the MCBS delivers the task result to the AV. Otherwise, the task computing service will be canceled.

Case 2 : The task result is not cached in the MCBS.

In this case, the task result is not cached in the MCBS. Therefore, in the first stage, the MCBS needs to make the task allocation strategy to minimize the task completion cost based on the task transmission time and the computing resources that can be provided to execute the task. The task allocation strategy indicates the optimal SCBS selected by the MCBS to execute the task. If the task cannot be completely executed within the TTL requested by the AV, the MCBS will deliver a message to AV i to cancel the service. Otherwise, the mechanism moves to the second stage, where the MCBS negotiates with AV i for the transaction price of the service based on the strategy and the minimum task computing cost obtained in the first stage. If an agreement between AV i and the MCBS is achieved, the AV will send the input of the task to the MCBS. Then, the MCBS allocates the task to the SCBS according to the optimal task allocation strategy. After the selected SCBS finishes the task computing service, the MCBS collects the result and delivers it to AV i.

Based on the task offloading mechanism, the MCBS intends to select the optimal SCBS to execute the task. In this way, the SCBSs covered by the MCBS need to compete with each other to win the chance for executing the task. After the MCBS selects the optimal SCBS, the MCBS and the AV need to negotiate with each other for the transaction price of the task offloading service to maximize their utilities. We thus resort to game theory to model the interactions among the participants in the task offloading process. Specifically, for the interactions among SCBSs in the first stage, we model this process as the SPSA game to determine their optimal bidding prices. This is because the SPSA enables the SCBSs to bid with their real valuation of the task computing service. In the SPSA, the players are the SCBSs, where the strategy of each SCBS is the bid based on its cost for completing the task. With the SPSA, each SCBS sends its bid for the task computing service to the MCBS and the MCBS selects the one with the lowest price as the winer of the game. For the winner, it will obtain the second lowest price bade by other SCBS. To promote the cooperation between the AV and the MCBS in the second stage of the mechanism, the interactions between them is formulated as a two-round bargaining game, where the MCBS is the leader of the game. In the first round of the bargaining game, the MCBS proposes a transaction price for the task computing service. If the AV accepts the proposal, an agreement between the AV and the MCBS is reached and the game is over. Otherwise, the game moves to the next round, where the MCBS proposes a new transaction price. In this round, the game is over no matter whether the agreement is achieved or not. Specifically, if the AV accepts the proposal, an agreement between the AV and the MCBS is reached. Otherwise, the task offloading service will be canceled.

According to the game models, we then analyze the utilities of the participants in the task offloading process. We first define the cost for completing the service. It consists of three parts which are the cost for transmitting the input of the task, the cost for computing the task and the cost for transmitting the output of the task, respectively. Through wired connections, the SCBSs and the MCBS can transmit data with high speed, we therefore ignore the time and cost incurred for transmitting data between the MCBS and the SCBSs. The cost for computing the task thus can be given by

$$C_{MC,q} = \begin{cases} \frac{\overline{s_q}+\underline{s_q}}{r_i}\varsigma, \text{ if } \ell_{i,q}=1, \delta_{MC,q}=1, \\ B_{k^*,q} + \frac{\overline{s_q}+\underline{s_q}}{r_i}\varsigma, \text{ if } \ell_{i,q}=1, \delta_{MC,q}=0, \end{cases} \tag{2}$$

where $\delta_{MC,q} = 1$ indicates that the task result is available in the MCBS's cache and $\delta_{MC,q} = 0$ otherwise. $\ell_{i,q} = 1$ means that the result of the requested task can be received by the AV within the requested TTL (i.e., $T_{i,q}$). r_i denotes the communication rate between the MCBS and AV i and ranges from $r_{\min}$ to $r_{\max}$. $B_{k^*,q}$ is the final price that the MCBS pays the selected SCBS k^* for executing the task computing service. ς is the cost for delivering data per unit time. $\frac{\overline{s_q}}{r_i}\varsigma$ and $\frac{\underline{s_q}}{r_i}\varsigma$ are the costs of the MCBS for receiving the input and delivering the output of the task, respectively.

According to the cost for completing the task computing service, we then analyze the utilities of the MCBS, the selected

SCBS and the AV, respectively. The utility of the MCBS is related to the cost spent on completing the task and the transaction price paid by the AV. It can be defined as

$$X_{MC,q} = \mathbf{P}_{i,q} - C_{MC,q} \tag{3}$$
$$= \begin{cases} \mathbf{P}_{i,q} - \frac{\overline{s_q}+\underline{s_q}}{r_i}\varsigma, \text{ if } \ell_{i,q}=1, \delta_{MC,q}=1, \\ \mathbf{P}_{i,q} - \left(B_{k^*,q} + \frac{\overline{s_q}+\underline{s_q}}{r_i}\varsigma\right), \text{ if } \ell_{i,q}=1, \delta_{MC,q}=0, \\ 0, \text{otherwise}, \end{cases}$$

where $\mathbf{P}_{i,q}$ is the transaction price that AV i pays the MCBS for the task offloading service. In order to increase profits, the transaction price of the task offloading service should be larger than the cost for completing the task, we have $C_{MC,q} < \mathbf{P}_{i,q}$. As such, $C_{MC,q}$ can be regarded as the reserve price of the MCBS. Namely, if $C_{MC,q} \geq \mathbf{P}_{i,q}$, the MCBS will cancel the task offloading service.

Similar to the MCBS, the utility of the SCBSs is decided by whether the task result is cached in the MCBS. If the task result is available in the MCBS's cache, the utility of the SCBSs will be zero. Otherwise, the task computing service will be executed by the selected SCBS. If SCBS k^* wins the game, its utility can be expressed as

$$X_{k^*,q} = B_{k^*,q} - C_{k^*,q}(\lambda_{k^*,q}), \ell_{i,q}=1, \delta_{MC,q}=0, \tag{4}$$

where $C_{k^*,q}(\lambda_{k^*,q}) = \min\{C_{k,q}(\lambda_{k,q}), \forall k \in \mathbb{K}\}$ and $B_{k^*,q} > C_{k^*,q}(\lambda_{k^*,q})$. $C_{k,q}(\lambda_{k,q})$ is the bid of SCBS k in the SPSA game.

Then, we define the utility of AV i. Intuitively, if the task computing service is completed within the TTL requested by the AV and with a low price, the AV will obtain a high utility. We define the utility of AV i for task q as

$$X_{i,q} = \begin{cases} C_{i,q} - \mathbf{P}_{i,q}, \text{ if } \ell_{i,q}=1, \\ 0, \text{otherwise}, \end{cases} \tag{5}$$

where $C_{i,q}$ is the reserve price charged by AV i for the task offloading service. Namely, if $C_{i,q} < \mathbf{P}_{i,q}$, AV i will cancel the task offloading service.

B. Problem Formulation and Game Analysis

From the above analysis, we can know that the utilities of the MCBS, the selected SCBS and the AV are related to the reserve price of AV i (i.e., $C_{i,q}$), the reserve price of the MCBS (i.e., $C_{MC,q}$) and the transaction price paid for the task (i.e., $\mathbf{P}_{i,q}$). In this subsection, we first analyze $C_{MC,q}$ and $C_{i,q}$ by modeling the interactions among SCBSs in the first stage as the SPSA game. Then, we formulate the interactions between the MCBS and AV i as a two-round bargaining game to determine $\mathbf{P}_{i,q}$ in the second stage. The proposed two-stage task offloading algorithm is detailed in Algorithm 1.

Stage 1: SPSA-based SCBS selection

In this stage, if the task result is cached in the MCBS, we have $B_{k^*,q}=0$ and $C_{MC,q} = \frac{\overline{s_q}+\underline{s_q}}{r_i}\varsigma$. Otherwise, the MCBS needs to determine $C_{MC,q}$ to maximize its utility. To make sure that the task can be executed within the TTL, the MCBS needs to select the SCBSs which can complete the task on time from set $\mathbb{K}$. Namely, the SCBSs that can be selected by the MCBS to execute the task should satisfy

$$T_{i,q} \geq \frac{\overline{s_q}+\underline{s_q}}{r_i} + \frac{(1-\epsilon_{k,q})D_q}{d_{k,q}}, k \in \mathbb{K}, \tag{6}$$

where $\epsilon_{k,q}$ is the proportion of the task that the SCBS has executed. Specially, $\epsilon_{k,q}=1$ means that the task result is cached in this SCBS. $d_{k,q}$ is the idle computing resources owned by SCBS k. It follows a uniform distribution within $[d_{\min}, d_{\max}]$.

After this, the MCBS needs to select the optimal SCBS to maximize $X_{MC,q}$. For the MCBS, the problem can be formulated as

$\mathbf{P1}$:

$$\max_k X_{MC,q} = \max_k \{\mathbf{P}_{i,q} - C_{MC,q}\} \tag{7}$$
$$= \max_k \left\{ \mathbf{P}_{i,q} - \delta_{MC,q}\frac{\overline{s_q}+\underline{s_q}}{r_i}\varsigma - (1-\delta_{MC,q})\left(B_{k^*,q} + \frac{\overline{s_q}+\underline{s_q}}{r_i}\varsigma\right)\right\}.$$

$$\begin{aligned} s.t. \quad & C1: D_{\min} \leq D_q \leq D_{\max}, \forall q \in \mathbb{Q}, \\ & C2: s_{\min} \leq \overline{s_q}, \underline{s_q} \leq s_{\max}, \forall q \in \mathbb{Q}, \\ & C3: d_{\min} \leq d_{k,q} \leq d_{\max}, \forall k \in \mathbb{K}, \\ & C4: \ell_{i,q} = 1, \\ & C5: r_{\min} \leq r_i \leq r_{\max}, \forall k \in \mathbb{K}, \\ & C6: C_{MC,q} < \mathbf{P}_{i,q}, \\ & C7: T_q \geq T_{i,q} \geq \delta_{MC,q}\frac{\overline{s_q}+\underline{s_q}}{r_i} \\ & \quad + (1-\delta_{MC,q})\left(\frac{\overline{s_q}+\underline{s_q}}{r_i} + \frac{(1-\epsilon_{k,q})D_q}{d_{k,q}}\right). \end{aligned} \tag{8}$$

Given the value of $\mathbf{P}_{i,q}$, which is analyzed in the next stage, the maximization problem of the MCBS can be rewritten as

$\mathbf{P2}$:

$$\max_k X_{MC,q} = \max_k \{\mathbf{P}_{i,q} - C_{MC,q}\} \tag{9}$$
$$= \max_k \left\{ \mathbf{P}_{i,q} - \delta_{MC,q}\frac{\overline{s_q}+\underline{s_q}}{r_i}\varsigma - (1-\delta_{MC,q})\left(B_{k^*,q} + \frac{\overline{s_q}+\underline{s_q}}{r_i}\varsigma\right)\right\}$$
$$= \mathbf{P}_{i,q} - \delta_{MC,q}\frac{\overline{s_q}+\underline{s_q}}{r_i}\varsigma - (1-\delta_{MC,q})\frac{\overline{s_q}+\underline{s_q}}{r_i}\varsigma - \min_k \{(1-\delta_{MC,q})B_{k^*,q}\}.$$

$$s.t. C1 - C7. \tag{10}$$

To solve P2, the MCBS resorts to the SPSA game to model the interactions among the SCBSs. In the SPSA, each SCBS bids for the task computing service and the MCBS selects the one which bids with the lowest price, where the final transaction price (i.e., $B_{k^*,q}$) is the second lowest price in the SPSA. Let $\widetilde{\mathbb{K}} = \{1, 2, ..., \widetilde{K}\}$ denote the set of SCBSs that satisfy (6). If SCBS k^* is selected by the MCBS, its utility is calculated by (4). Otherwise, its utility is zero. Therefore, the problem of SCBS k^* can be given by

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TVT.2020.3041587, IEEE Transactions on Vehicular Technology

P3 :

$$\max_{C_{k^*,q}(\lambda_{k^*,q})} X_{k^*,q} = \max_{C_{k^*,q}(\lambda_{k^*,q})} \{B_{k^*,q} - C_{k^*,q}(\lambda_{k^*,q})\}. \tag{11}$$

$$\begin{aligned} s.t.\ & C3,\ C4,\ C7, \\ & C8:\ \delta_{MC,q} = 0, \\ & C9:\ C_{k^*,q}(\lambda_{k^*,q}) = \min\{C_{k,q}(\lambda_{k,q})\}, k \in \widetilde{\mathbb{K}}, \\ & C10:\ C_{k^*,q}(\lambda_{k^*,q}) > \lambda_{k^*,q}, \end{aligned} \tag{12}$$

where $\lambda_{k,q}(k \in \widetilde{\mathbb{K}})$ is the cost of SCBS k for computing the task. It depends on the proportion of the task that the SCBS has executed. We have

$$\lambda_{k,q} = (1 - \epsilon_{k,q}) D_q c_{k,q}, \tag{13}$$

where $c_{k,q}$ is the cost per unit resource for computing. When the SCBS has more idle resources, a higher execution rate can be provided to compute the task with the result that the SCBS declares a higher price. As such, $c_{k,q}$ can be defined by

$$c_{k,q} = \frac{d_{k,q} - d_{\min}}{d_{\max} - d_{\min}}. \tag{14}$$

From (12), we can know that the maximization problem of SCBS k is related to the bidding strategies of all the SCBSs in the game, where the bidding strategy of SCBS $k(k \in \widetilde{\mathbb{K}})$ is a function of $\lambda_{k,q}$. Based on the value of $\lambda_{k,q}$, SCBS k needs to decide the optimal bidding strategy to win the game, where the optimal bidding strategy of SCBS k is given by the following theorem.

***Theorem** 1:* In the SPSA, the optimal bidding strategy of SCBS $k(k \in \widetilde{\mathbb{K}})$ for the task computing service is $C_{k,q}(\lambda_{k,q}) = \lambda_{k,q}$.

The proof of this theorem is given in appendix A.

Based on theorem 1, we can know that the optimal strategy of each SCBS is $C_{k,q}(\lambda_{k,q}) = \lambda_{k,q}$. It indicates that each SCBS cannot obtain a higher utility by changing the strategy. In other words, all the SCBSs will adopt this strategy to compete with each other so that the game achieves the Nash equilibrium. From the above analysis, we have

$$B_{k^*,q} = C_{\widetilde{k},q}(\lambda_{\widetilde{k},q}) = \lambda_{\widetilde{k},q}, \tag{15}$$

where $\lambda_{\widetilde{k},q}$ is the second lowest bid declared by SCBS $\widetilde{k}$.

Then, we define the reserve price of AV i. If the task needs to use more computing resources, the AV will have a higher reserve price. Therefore, $C_{i,q}$ increases with D_q. It can be defined as

$$C_{i,q} = \frac{\overline{s_q} + \underline{\underline{s_q}}}{r_i}\varsigma + \Delta D_q, \tag{16}$$

where Δ is the cost per unit resource expected by AV i for computing the task. We have

$$\Delta = \max\left\{\gamma_{i,q}, \frac{D_q - D_{\min}}{D_{\max} - D_{\min}}\right\}, \tag{17}$$

where $\gamma_{i,q}(0 \le \gamma_{i,q} \le 1)$ is used to reflect the personal preference for the task.

In the above analysis, we assumed that the number of SCBSs that can complete the task on time in the networks is larger than 1 (i.e., $\widetilde{K} > 1$). Here, we analyze the situation where $\widetilde{K} = 1$. In this situation, the MCBS only has one SCBS that can be selected to execute the task. It means that the SCBS can win the SPSA game without competing with other SCBSs. If this situation occurs, the cost for completing the task is $C_{MC,q} = \lambda_{k^*,q} + \frac{\overline{s_q} + \underline{\underline{s_q}}}{r_i}\varsigma$. Accordingly, the profits for completing the task computing service will be equally shared between the MCBS and the SCBS. We have

$$\begin{aligned} B_{k^*,q} &= \lambda_{k^*,q} + \frac{\mathbf{P}_{i,q} - C_{MC,q}}{2} \\ &= \lambda_{k^*,q} + \frac{\mathbf{P}_{i,q} - \lambda_{k^*,q} - \frac{\overline{s_q} + \underline{\underline{s_q}}}{r_i}\varsigma}{2}. \end{aligned} \tag{18}$$

In this way, the utilities of the MCBS and the selected SCBS can be given by

$$X_{MC,q} = X_{k^*,q} = \frac{\mathbf{P}_{i,q} - \lambda_{k^*,q} - \frac{\overline{s_q} + \underline{\underline{s_q}}}{r_i}\varsigma}{2}. \tag{19}$$

Stage 2: Bargaining-based task offloading

After the MCBS makes the optimal task allocation strategy and obtains the cost for completing the task (i.e., $C_{MC,q}$), the MCBS then negotiates with the AV to achieve an agreement. Obviously, the AV intends to minimize the transaction price for the task offloading service while the MCBS intends to maximize the profits. Thus, we model the interactions between the AV and the MCBS as a bargaining game with two rounds to determine the transaction price (i.e., $\mathbf{P}_{i,q}$) and distribute the profits (i.e., $C_{i,q} - C_{MC,q}$).

For the MCBS, the problem can be formulated as

P4 :

$$\begin{aligned} \max X_{MC,q} &= \max\{\mathbf{P}_{i,q} - C_{MC,q}\} \\ &= \max\{\mathbf{P}_{i,q}\} - \delta_{MC,q}\frac{\overline{s_q} + \underline{\underline{s_q}}}{r_i}\varsigma \\ &\quad - (1 - \delta_{MC,q})\left(B_{k^*,q} + \frac{\overline{s_q} + \underline{\underline{s_q}}}{r_i}\varsigma\right), \end{aligned} \tag{20}$$

$$s.t. \quad \mathbf{P}_{i,q} > C_{MC,q}, \tag{21}$$

where the value of $B_{k^*,q} + \frac{\overline{s_q} + \underline{\underline{s_q}}}{r_i}\varsigma$ is obtained in the first stage.

Similarly, the problem of the AV can be formulated as

P5 :

$$\max\{C_{i,q} - \mathbf{P}_{i,q}\} = C_{i,q} - \min\{\mathbf{P}_{i,q}\}, \tag{22}$$

$$s.t. \quad \mathbf{P}_{i,q} < C_{i,q}. \tag{23}$$

To solve P4 and P5, we then analyze the two-round bargaining game in detail. In round $g(g \in \{1, 2\})$, the MCBS proposes a transaction price $\Gamma_{MC,q}|_g$ for the task offloading service. Then, the AV makes a decision based on $\Gamma_{MC,q}|_g$. Specifically, if the AV accepts $\Gamma_{MC,q}|_1$, an agreement between the AV and the MCBS is achieved and the bargaining is finished. Otherwise, the game moves to the next round, where the MCBS proposes $\Gamma_{MC,q}|_2$. In round two, the bargaining game is over no matter whether the AV accepts or rejects $\Gamma_{MC,q}|_2$. The utilities of the MCBS and the AV will have discounts if the game moves to the next round to reflect their patience for the service. The discounting rates of the MCBS and AV i are denoted as ψ_{MC} and ψ_i, respectively.

In the bargaining game, the reserve prices of the MCBS and AV i are $C_{MC,q}$ and $C_{i,q}$. As the MCBS does not know the reserve price of the AV, it estimates that the profits (i.e., $C_{i,q}-C_{MC,q}$) for the task offloading service follows a uniform distribution within $[0, \mathrm{P}_{i,q}|_1 - C_{MC,q}]$ in the first round, where $\mathrm{P}_{i,q}|_1$ can be defined as

$$\mathrm{P}_{i,q}|_1 = \frac{\overline{s_q}+\underline{s_q}}{r_i}\varsigma + D_q. \tag{24}$$

According to these conditions, the MCBS needs to decide the optimal strategy in each round to maximize its utility. In order to promote the cooperation between the MCBS and the AV, the MCBS sends the information (i.e., reserve price, patience for the task and $\Gamma_{MC,q}|_1$) to the AV in the first round. For the AV, it needs to make a decision for accepting or rejecting the proposal. The optimal strategies of the MCBS and the AV in the first round are given by the following theorem 2 and theorem 3.

Theorem 2: In the first round of the bargaining game, the optimal strategy of the MCBS is

$$\Gamma_{MC,q}|_1 = \frac{\mathrm{P}_{i,q}|_1 - C_{MC,q}}{\frac{4}{2-\psi_i} - \psi_{MC}\frac{2}{(2-\psi_i)^2}}. \tag{25}$$

The proof of this theorem is given in appendix B.

Theorem 3: In the first round of the bargaining game, the optimal strategy of the AV is that if

$$C_{i,q} > \frac{\mathrm{P}_{i,q}|_1 - C_{MC,q}}{2-\psi_{MC}\frac{1}{(2-\psi_i)}} + C_{MC,q}, \tag{26}$$

the AV will accept the proposal and the agreement is reached. Otherwise, the proposal will be rejected.

The proof of this theorem is given in appendix C.

If the proposal is accepted by the AV, the transaction price can be given by $\mathbf{P}_{i,q} = \frac{\mathrm{P}_{i,q}|_1 - C_{MC,q}}{\frac{4}{2-\psi_i} - \psi_{MC}\frac{2}{(2-\psi_i)^2}} + C_{MC,q}$. If an agreement between the MCBS and the AV is not reached in the first round, the bargaining game will move to the next round, where the optimal strategies of the MCBS and the AV are given by the following two theorems.

Theorem 4: In the second round of the bargaining game, the optimal strategy of the MCBS is

$$\Gamma_{MC,q}|_2 = \frac{\mathrm{P}_{i,q}|_1 - C_{MC,q}}{4-\psi_{MC}\frac{2}{(2-\psi_i)}}. \tag{27}$$

The proof of this theorem is given in appendix D.

Theorem 5: In the second round of the bargaining game, the optimal strategy of the AV is that if

$$C_{i,q} > \frac{\mathrm{P}_{i,q}|_1 - C_{MC,q}}{4-\psi_{MC}\frac{2}{(2-\psi_i)}} + C_{MC,q}, \tag{28}$$

an agreement between the AV and the MCBS will be achieved.

The proof of this theorem is given in appendix E.

If an agreement is achieved in this round, we have $\mathbf{P}_{i,q} = \frac{\mathrm{P}_{i,q}|_1 - C_{MC,q}}{4-\psi_{MC}\frac{2}{(2-\psi_i)}} + C_{MC,q}$.

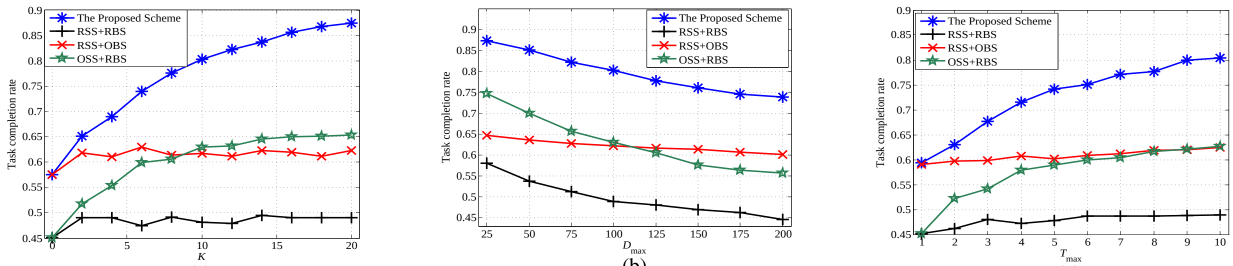
Algorithm 1 Two-Stage Task Offloading Algorithm

Initial the parameters of the task offloading service.
$\ell_{i,q} = 0$.
AV i sends a task computing request to the MCBS.
//**Stage1** :
if $\delta_{MC,q} = 1$ **then**
  if $T_{i,q} \geq \frac{s_q}{r_i}$ **then**
    $\ell_{i,q} = 1$.
    Calculate $C_{MC,q}$ using (2).
  else
    Cancel the task offloading service.
  end if
else
  for $k=1; k \leq K$ **do**
    if $T_{i,q} \geq \frac{\overline{s_q}+\underline{s_q}}{r_i} + \frac{(1-\epsilon_{k,q})D_q}{d_{k,q}}$ **then**
      $\widetilde{\mathbb{K}} \leftarrow k$.
    end if
  end for
  if $\widetilde{K} \geq 1$ **then**
    $\ell_{i,q} = 1$.
    if $\widetilde{K} > 1$ **then**
      for $k=1; k \leq \widetilde{K}$ **do**
        Calculate $\lambda_{k,q}$ using (13).
        Send $\lambda_{k,q}$ to the MCBS.
      end for
      Select the SCBS k^* by $k^* = \arg\min\{\lambda_{k,q}\}$.
      Calculate $B_{k^*,q}$ using (15).
      Calculate $C_{MC,q}$ using (2).
    else
      Calculate $B_{k^*,q}$ using (18).
      Calculate $C_{MC,q}$ by $\lambda_{k^*,q} + \frac{\overline{s_q}+\underline{s_q}}{r_i}\varsigma$.
    end if
  else
    Cancel the task offloading service.
  end if
end if
//**Stage2** :
if $\ell_{i,q} = 1$ **then**
  MCBS calculates $\mathrm{P}_{i,q}|_1$ using (24)
  MCBS makes the optimal strategy using (25).
  if $C_{i,q} > \frac{\mathrm{P}_{i,q}|_1}{2-\psi_{MC}\frac{1}{(2-\psi_i)}} + C_{M,q}$ **then**
    AV i accepts $\Gamma_{MC,q}|_1$.
  else
    MCBS makes the optimal strategy using (27).
    if $C_{i,q} > \frac{\mathrm{P}_{i,q}|_1}{4-\psi_{MC}\frac{2}{(2-\psi_i)}} + C_{M,q}$ **then**
      AV i accepts $\Gamma_{MC,q}|_2$.
    else
      AV i rejects $\Gamma_{MC,q}|_2$.
      The task offloading service is canceled.
    end if
  end if
end if

V. Simulation Results

In this section, we evaluate the performance of the proposed collaborative task offloading scheme using a simulator imple-

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TVT.2020.3041587, IEEE Transactions on Vehicular Technology



(a) (b) (c)

Fig. 2. Task completion rate by changing different parameters. (a) Number of SCBSs. (b) Maximum value of the resources that the task needs to use. (c) Maximum value of the task's TTL.

TABLE II
SIMULATION PARAMETERS

Parameters	Values
$\mathbb{K}$	{1,2,...,20}
$\mathbb{Q}$	{1,2,...,150}
$D_{\min}$	10
$D_{\max}$	{25,50,75,100,125,150,175,200}
$d_{\min}, d_{\max}$	10, 200
$s_{\min}, s_{\max}$	[0.1,1] MBytes
$T_{\min}$	1
$T_{\max}$	{1,2,3,4,5,6,7,8,9,10}
ϕ	0.9
r_i	[5,15] Mbit/s
ς	10
$\gamma_{i,q}$	0.5
ψ_{MC}, ψ_i	0.8, 0.8

mented in Matlab. We first introduce the simulation scenario, followed by the simulation results and discussions.

A. Simulation Scenario

In the simulation, we consider the scenario that an AV intends to offload its task to the MCBS, where the input size and the output size of the requested task are selected from [0.1, 1] MBytes. The number of SCBSs in the communication coverage of the MCBS varies from 1 to 20. The computing ability of each SCBS is randomly selected from [10, 200]. The task results cached in the MCBS and the SCBSs are based on the popularity, where the popularity of the task result follows the Zipf-like distribution. Specifically, we assume that the MCBS stores the complete results of the 20 most popular tasks. Apart from these tasks, each SCBS in the coverage of the MCBS stores the 10 most popular tasks, where the portion of each task result is selected from [0, 1]. The parameters used in the simulation are listed in Table II.

To evaluate the performance of the proposed collaborative task offloading scheme, we compare the proposed scheme with the conventional schemes shown as follows.

- **RSS+RBS:** The MCBS allocates the task computing service to a randomly selected SCBS. In addition, the bidding strategy of the MCBS in each round of the bargaining game is randomly selected.
- **RSS+OBS:** The MCBS allocates the task computing service to a randomly selected SCBS while the bidding strategy of the MCBS in each round of the bargaining game is determined by the proposed scheme.
- **OSS+RBS:** The MCBS allocates the task computing service to the optimally selected SCBS. However, the bidding strategy of the MCBS in each round of the bargaining game is randomly selected.

Under these conditions, the metrics used for the performance evaluation include:

- **Task completion rate:** The task completion rate is computed by the number of completed tasks divided by the total number of tasks requested by AVs. We evaluate the task completion rate (i.e., Fig. 2) with different schemes by changing K, $D_{\max}$ and $T_{\max}$, respectively.
- **Utilities of the participants:** We evaluate the utilities of the MCBS, the selected SCBS and the AV (i.e., Figs. 3-5) which are involved in the task offloading service with our proposed scheme and the conventional schemes by changing K, $D_{\max}$ and $T_{\max}$, respectively.

B. Simulation Results

Fig. 2 shows the task completion rate with different task offloading schemes by changing K, $D_{\max}$ and T_q, respectively. From this figure, we can see that the proposed scheme can achieve a higher task completion rate than the conventional schemes. In comparison, the RSS+RBS results in the lowest task completion rate compared with other schemes. In Fig. 2(a), the task completion rate increases with an increase of K in the proposed scheme and OSS+OBS. This is because the more SCBSs there are, the more candidate SCBSs with a high computing ability are available to execute the task. As a result, the probability that the task can be completed on time becomes high. For RSS+RBS and RSS+OBS, the increase in the number of SCBSs hardly affects the task completion rate because the two schemes always randomly select the SCBS to execute task. In Fig. 2(b), the task completion rate for the task offloading service in all the schemes decrease with an increase of $D_{\max}$. With the increase of $D_{\max}$, the task has a low probability to be completed within the TTL, thus resulting in a low task completion rate. Fig. 2(c) is the task completion rate with increasing the value of $T_{\max}$. In this figure, with an increase of $T_{\max}$, there are more time available for computing the task. Therefore, it can be seen that the task completion rate has the increasing trend in all the schemes. In addition, we can see that the increase of the task completion rate in the proposed scheme and OSS+RBS is higher than that in RSS+RBS and RSS+OBS. The reason for this is that the proposed scheme

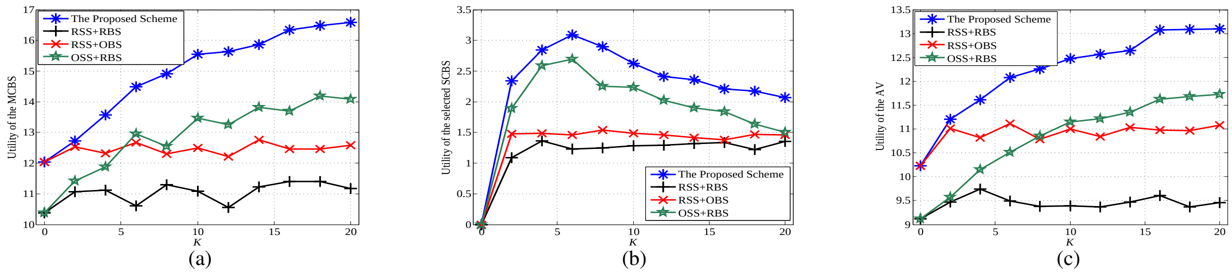


Fig. 3. Utility of the participants versus number of SCBSs. (a) Utility of the MCBS. (b) Utility of the selected SCBS. (c) Utility of the AV.

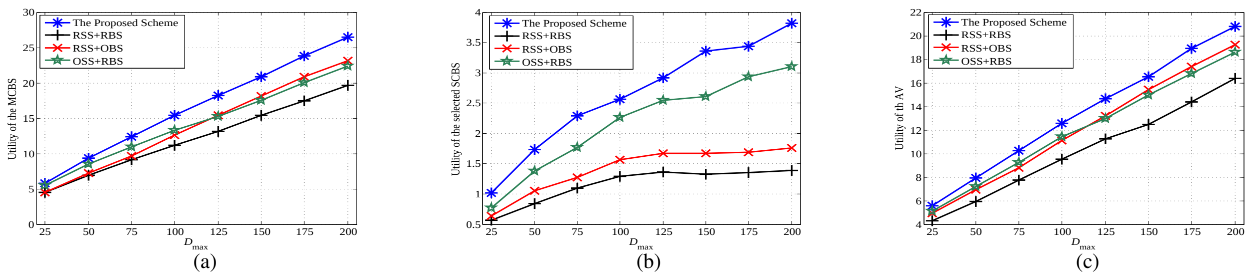


Fig. 4. Utility of the participants versus maximum value of the resources that the task needs to use. (a) Utility of the MCBS. (b) Utility of the selected SCBS; (c) Utility of the AV.

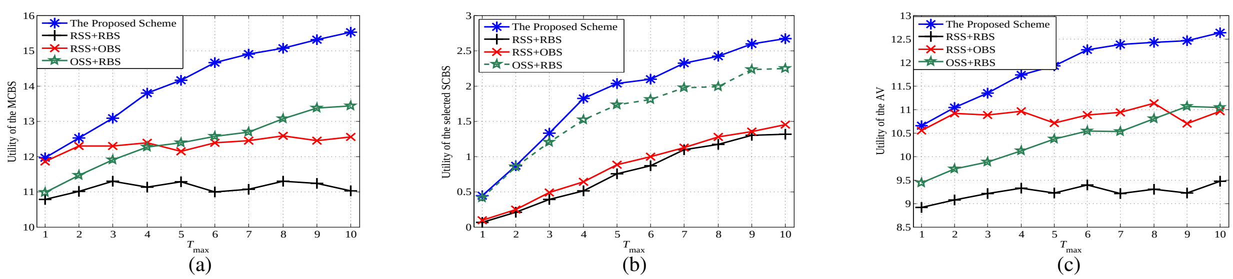


Fig. 5. Utility of the participants versus maximum value of task's TTL. (a) Utility of the MCBS. (b) Utility of the selected SCBS. (c) Utility of the AV.

and OSS+RBS can select the optimal SCBS to execute the task based on the required TTL.

Figs. 3-5 are the utilities of the MCBS, the selected SCBS and the AV involved in the task offloading service with different values of K, $D_{\max}$ and $T_{\max}$. It can be seen from these figures that the proposed scheme can bring the highest utilities to all the participants compared with the conventional schemes. This is because that the proposed scheme can allocate the task more efficient and therefore has a high probability to complete the task offloading service. As a result, the participants involved in the task offloading process can obtain more utilities.

In Fig. 3(a) and (c), the utilities of the MCBS and the AV increase with an increase of K in the proposed scheme and OSS+RBS. The reasons are as follows. First, the task completion rate is increased with an increase of K. Thus, the profits generated by the completed tasks can be shared to enhance their utilities. Second, with the increase of K, the task can be computed with a low cost which also brings high utilities to the MCBS and the AV. For the utility of the selected SCBS, as shown in Fig. 3(b), it first increases and then decreases with the increase of K in the proposed scheme and OSS+RBS. The reason for the increase is that the number of tasks that can be completed by SCBSs increases with the increase of K. For the decrease, the reason is that the increase in the number of SCBSs makes the competition more intense and thus reduces the utility.

Fig. 4 shows the utilities of the participants with increasing $D_{\max}$. As seen in this figure, the utilities of all the participants increase with an increase of the value of $D_{\max}$. This is because when $D_{\max}$ is larger, more computing resources will be used for computing the task and the reserve price of the AV for the service becomes higher. As a result, the utility of the selected SCBS is increased to compensate for the resources spent on computing the task. In addition, due to the increased profits (i.e., $P_{i,q} - C_{M_q}$) that can be shared between the MCBS and the AV, the utilities of the MCBS and the AV are increased.

Fig. 5 shows the utilities of the participants by changing the value of $T_{\max}$. From Fig. 5(a) and (c), we can see that the utilities of the MCBS and the AV keep increasing in the proposed scheme and OSS+RBS. This is because with the increase of $T_{\max}$, the probability that the task can be completed within the TTL requested by the AV becomes high so that the MCBS and the AV can obtain more utilities. However, in RSS+RBS and RSS+OBS, the MCBS randomly selects a SCBS to execute the task. Consequently, their utilities are hardly affected by the increase in time. As for the utility of the selected SCBS, we can see in Fig. 5(b) that it increases with the increase of the value of $T_{\max}$ in all the schemes. This mainly accounts for the fact that with the increase of the value of $T_{\max}$, more time can be used for executing the requested task. As a result, the task that allocated to the selected SCBS can be completed with a high probability.

VI. Conclusion

In this paper, we have proposed a game theoretic scheme for collaborative vehicular task offloading in 5G HetNets to improve the task completion rate and the utilities of participants. Specifically, by considering the imbalanced distribution of computing requests, the reuse of task results and the available resources owned by base stations, a two-stage task offloading mechanism has been designed to promote the cooperation among the participants involved in the task offloading service. In the first stage, we have formulated an auction model to help the MCBS select the optimal SCBS to execute the offloaded task. With the task computing cost bade by the optimal SCBS, a bargaining game has been adopted to model the interactions between the MCBS and the AV to reach an agreement of the task offloading service in the second stage. Compared with the conventional schemes, the simulation results have shown that the proposed scheme can obtain the highest task completion rate for the offloading service and bring the highest utilities to the participants.

About the future work, we plan to extend this work in two aspects. One is the optimization of the caching strategies of the MCBS and the SCBSs to further improve the task completion rate and the utilities of the participants. The other is the analysis of the security of vehicular task offloading by integrating the proposed scheme with blockchain.

Appendix A
Proof of Theorem 1

As the SCBSs in the networks are rational individuals, the bidding price of SCBS k should larger than its cost to obtain profits. We have $\lambda_{k,q} < C_{k,q}(\lambda_{k,q})$. Then, based on the value of the price declared by SCBS $\widetilde{k}(\widetilde{k} \in \mathbb{K}/k)$, we prove this theorem considering different cases.

Case1: $\lambda_{\widetilde{k},q} > C_{k,q}(\lambda_{k,q}) > \lambda_{k,q}$

In this case, $\lambda_{\widetilde{k},q}$ is the second lowest price bade by SCBS $\widetilde{k}$. SCBS k is the winner of the game if its bidding strategy is $\lambda_{k,q}(\lambda_{\widetilde{k},q} > \lambda_{k,q})$. As a result, SCBS k will obtain $\lambda_{\widetilde{k},q}$ after completing the task offloading service. If SCBS k bids by $C_{k,q}(\lambda_{k,q})$ instead of $\lambda_{k,q}$, we have $\lambda_{\widetilde{k},q} > C_{k,q}(\lambda_{k,q})$. The SCBS can also win the SPSA game and obtain the same utility. Therefore, in this case, the bidding strategy $\lambda_{k,q}$ is as good as $C_{k,q}(\lambda_{k,q})$.

Case2: $C_{k,q}(\lambda_{k,q}) > \lambda_{k,q} > \lambda_{\widetilde{k},q}$

In this case, SCBS $\widetilde{k}$ is the lowest bidder and wins the SPSA game. If the bidding strategy of SCBS k is $\lambda_{k,q}$, we have $\lambda_{k,q} > \lambda_{\widetilde{k},q}$. On the other hand, if the bidding strategy of the SCBS is $C_{k,q}(\lambda_{k,q})$, we have $C_{k,q}(\lambda_{k,q}) > \lambda_{\widetilde{k},q}$. As such, the SCBS will lose the chance to compute the task no matter whether its bidding strategy is $\lambda_{k,q}$ or $C_{k,q}(\lambda_{k,q})$. Consequently, in this case, there is no difference between the bidding strategy $\lambda_{k,q}$ and the bidding strategy $C_{k,q}(\lambda_{k,q})$.

Case3: $C_{k,q}(\lambda_{k,q}) > \lambda_{\widetilde{k},q} > \lambda_{k,q}$

In this case, if the bidding strategy of SCBS k is $C_{k,q}(\lambda_{k,q})$, we have $C_{k,q}(\lambda_{k,q}) > \lambda_{\widetilde{k},q}$. As such, the SCBS will lose the chance to compute the task and obtain the profits. However, if the bidding strategy of the SCBS is $\lambda_{k,q}$, the SCBS will win the game and gain the utility $\lambda_{\widetilde{k},q} - \lambda_{k,q} > 0$. Therefore, the bidding strategy $\lambda_{k,q}$ is better than $C_{k,q}(\lambda_{k,q})$ in this case.

Since the above analysis contains all the cases, we thus can conclude that $C_{k,q}(\lambda_{k,q}) = \lambda_{k,q}$ is the optimal bidding strategy of SCBS $k(k \in \mathbb{K})$ to maximize its utility in the SPSA game. The theorem is proved. ■

Appendix B
Proof of Theorem 2

We use the reverse induction method to analyze the optimal strategies of the AV and the MCBS in the bargaining game. We focus on round two and assume that AV i rejects $\digamma_{MC,q}|_1$ in the first round. In round two, the MCBS estimates that the profits for computing task q (i.e., $C_{i,q} - C_{MC,q}$) follows a uniform distribution with $[0, \mathrm{P}_{i,q}|_2 - C_{MC,q}]$. In this way, the goal of the MCBS is to decide $\digamma_{MC,q}|_2$ to maximize its utility. If the AV accepts the proposal, the utility of the MCBS is $\psi_{MC}\digamma_{MC,q}|_2$. Otherwise, the game is over and the utility of the MCBS becomes zero. The maximization problem of the MCBS thus can be formulated as

$$\begin{aligned} &\max_{\digamma_{MC,q}|_2} X_{MC,q}|_2 \\ &= \max_{\digamma_{MC,q}|_2} \psi_{MC}\left(p_a|_2 \cdot \digamma_{MC,q}|_2 + p_r|_2 \cdot 0\right), \end{aligned} \tag{29}$$

where $p_a|_2$ and $p_r|_2$ are the probabilities that AV i accepts and rejects $\digamma_{MC,q}|_2$, respectively. We have

$$\begin{aligned} p_a|_2 &= \Pr\left\{C_{i,q} - C_{MC,q} \geq \digamma_{MC,q}|_2\right\} \\ &= \frac{\mathrm{P}_{i,q}|_2 - C_{MC,q} - \digamma_{MC,q}|_2}{\mathrm{P}_{i,q}|_2 - C_{MC,q}}. \end{aligned} \tag{30}$$

Combining (29) and (30), the maximization problem becomes

$$\begin{aligned} \max_{\digamma_{MC,q|_2}} X_{MC,q}|_2 = \max_{\digamma_{MC,q|_2}} \Bigg\{ &\psi_{MC} \\ &\times \frac{\mathrm{P}_{i,q}|_2 - C_{MC,q} - \digamma_{MC,q}|_2}{\mathrm{P}_{i,q}|_2 - C_{MC,q}} \digamma_{MC,q}|_2 \Bigg\}. \end{aligned} \tag{31}$$

By taking the first derivative of (31) with respect to $\digamma_{MC,q}|_2$, we have the optimal bidding strategy of the MCBS in round two, shown as

$$\digamma_{MC,q}|_2 = \frac{\mathrm{P}_{i,q}|_2 - C_{MC,q}}{2}. \tag{32}$$

If AV i accepts $\digamma_{MC,q}|_2$, the transaction price is $\mathbf{P}_{i,q} = C_{MC,q} + \digamma_{MC,q}|_2$ and the utility of the AV in the game can be expressed as

$$\begin{aligned} X_{i,q}|_2 &= \psi_i\left(C_{i,q} - \mathbf{P}_{i,q}\right) \\ &= \psi_i\left(C_{i,q} - \frac{\mathrm{P}_{i,q}|_2 - C_{MC,q}}{2} - C_{MC,q}\right). \end{aligned} \tag{33}$$

For the AV, as it knows that its utility is $X_{k,q}|_2$ in round two, the AV will accept $\Gamma_{MC,q}|_1$ in the first round if

$$X_{i,q}|_2 \leq X_{i,q}|_1 = C_{i,q} - \Gamma_{MC,q}|_1 - C_{MC,q}. \tag{34}$$

Combining (33) and (34), we have

$$\psi_i\left(C_{i,q} - \frac{\mathrm{P}_{i,q}|_2 - C_{MC,q}}{2} - C_{MC,q}\right) \leq C_{i,q} - \Gamma_{MC,q}|_1 - C_{MC,q}. \tag{35}$$

Simplify (35), we can obtain the condition that AV i accepts the proposal, shown as

$$C_{i,q} - C_{MC,q} \geq \frac{\Gamma_{MC,q}|_1 - \psi_i \frac{\mathrm{P}_{i,q}|_2 - C_{MC,q}}{2}}{1-\psi_i}. \tag{36}$$

If (36) holds, the AV will accept $\Gamma_{MC,q}|_1$ in the first round. It means that if $C_{i,q} - C_{MC,q} < \frac{\Gamma_{MC,q}|_1 - \psi_i \frac{\mathrm{P}_{i,q}|_2 - C_{MC,q}}{2}}{1-\psi_i}$, the AV will reject $\Gamma_{MC,q}|_1$ in the first round and the game will move to round two. In this way, we can know that the maximum value of $C_{i,q}$ in round two estimated by the MCBS is $\frac{\Gamma_{MC,q}|_1 - \psi_i \frac{\mathrm{P}_{i,q}|_2 - C_{MC,q}}{2}}{1-\psi_i}$. Recall that the MCBS estimates that $C_{i,q}$ follows the uniform distribution with $[0, \mathrm{P}_{i,q}|_2 - C_{MC,q}]$, we thus have

$$\mathrm{P}_{i,q}|_2 - C_{MC,q} = \frac{\Gamma_{MC,q}|_1 - \psi_i \frac{\mathrm{P}_{i,q}|_2 - C_{MC,q}}{2}}{1-\psi_i}. \tag{37}$$

Simplify (37), we have the maximum price of $C_{i,q} - C_{MC,q}$ estimated by the MCBS in round two, shown as

$$\mathrm{P}_{i,q}|_2 - C_{MC,q} = \frac{2\Gamma_{MC,q}|_1}{2-\psi_i}. \tag{38}$$

Combining (32) and (38), we have

$$\Gamma_{MC,q}|_2 = \frac{\Gamma_{MC,q}|_1}{2-\psi_i}. \tag{39}$$

Based on (36) and (37), the MCBS can optimally select $\Gamma_{MC,q}|_1$ to maximize its utility in the first round. The problem is formulated as

$$\max_{\Gamma_{MC,q}|_1} X_{MC,q}|_1 = \max_{\Gamma_{MC,q}|_1} \Bigg(\Gamma_{MC,q}|_1 \cdot p_a|_1 + \psi_{MC}\Gamma_{MC,q}|_2 \cdot \overline{p_a|_2} + p_r|_2 \cdot 0\Bigg), \tag{40}$$

where $p_a|_1$ is the probability that the MCBS accepts $\Gamma_{MC,q}|_1$ in the first round. From the above analysis, we can know that if $C_{i,q} \geq \mathrm{P}_{i,q}|_2$, the AV will accept $\Gamma_{MC,q}|_1$ in the first round. We thus have

$$\begin{aligned}
p_a|_1 &= \Pr\{C_{i,q} \geq \mathrm{P}_{i,q}|_2\} \\
&= \frac{\mathrm{P}_{i,q}|_1 - C_{MC,q} - (\mathrm{P}_{i,q}|_2 - C_{MC,q})}{\mathrm{P}_{i,q}|_1 - C_{MC,q}} \\
&= \frac{\mathrm{P}_{i,q}|_1 - C_{MC,q} - \frac{2\Gamma_{MC,q}|_1}{2-\psi_i}}{\mathrm{P}_{i,q}|_1 - C_{MC,q}} \\
&= \frac{(2-\psi_i)\left(\mathrm{P}_{i,q}|_1 - C_{MC,q}\right) - 2\Gamma_{MC,q}|_1}{(2-\psi_i)(\mathrm{P}_{i,q}|_1 - C_{MC,q})} \\
&= 1 - \frac{2\Gamma_{MC,q}|_1}{(2-\psi_i)(\mathrm{P}_{i,q}|_1 - C_{MC,q})}.
\end{aligned} \tag{41}$$

$\overline{p_a|_2}$ is the probability that the AV rejects the proposal in the first round and accepts $\Gamma_{MC,q}|_2$ in the second round. It can be given by

$$\begin{aligned}
\overline{p_a|_2} &= \Pr\{C_{i,q} < \mathrm{P}_{i,q}|_2\} \\
&\quad \times \Pr\{C_{i,q} - C_{MC,q} \geq \Gamma_{MC,q}|_2\} \\
&= \frac{\mathrm{P}_{i,q}|_2 - C_{MC,q}}{\mathrm{P}_{i,q}|_1 - C_{MC,q}} \times \frac{\mathrm{P}_{i,q}|_2 - C_{MC,q} - \Gamma_{MC,q}|_2}{\mathrm{P}_{i,q}|_2 - C_{MC,q}} \\
&= \frac{\mathrm{P}_{i,q}|_2 - C_{MC,q} - \Gamma_{MC,q}|_2}{\mathrm{P}_{i,q}|_1 - C_{MC,q}} \\
&= \frac{\frac{2\Gamma_{MC,q}|_1}{2-\psi_i} - \frac{\Gamma_{MC,q}|_1}{2-\psi_i}}{\mathrm{P}_{i,q}|_1 - C_{MC,q}} \\
&= \frac{\Gamma_{MC,q}|_1}{(2-\psi_i)(\mathrm{P}_{i,q}|_1 - C_{MC,q})}.
\end{aligned} \tag{42}$$

By substituting (41) and (42) into (40), the maximization problem is changed by

$$\max_{\Gamma_{MC,q}|_1} X_{MC,q}|_1 = \max_{\Gamma_{MC,q}|_1} \left\{(\mathrm{P}_{i,q}|_1 - C_{MC,q})\Gamma_{MC,q}|_1 - \frac{2\Gamma_{MC,q}|_1{}^2}{2-\psi_i} + \psi_{MC}\frac{\Gamma_{MC,q}|_1{}^2}{(2-\psi_i)^2}\right\}. \tag{43}$$

By taking the first derivative of (43) with respect to $\Gamma_{MC,q}|_1$, we have the optimal strategy of the MCBS in the first round, shown as

$$\Gamma_{MC,q}|_1 = \frac{\mathrm{P}_{i,q}|_1 - C_{MC,q}}{\frac{4}{2-\psi_i} - \psi_{MC}\frac{2}{(2-\psi_i)^2}}. \tag{44}$$

The theorem is proved. ■

Appendix C
Proof of Theorem 3

From (34)-(39), the AV can know the optimal strategy of the MCBS in round two, i.e., $\Gamma_{MC,q}|_2 = \frac{\Gamma_{MC,q}|_1}{2-\psi_i}$. In addition, we can know from (36) and (37) that the AV will accept $\Gamma_{MC,q}|_1$ in the first round if

$$C_{i,q} - C_{MC,q} \geq \mathrm{P}_{i,q}|_2 - C_{MC,q}. \tag{45}$$

Combining (38) and (45), the condition that the AV accepts $\Gamma_{MC,q}|_1$ in the first round becomes

$$\begin{aligned}
C_{i,q} - C_{MC,q} &> \mathrm{P}_{i,q}|_2 - C_{MC,q} \\
&= \frac{2}{2-\psi_i}\frac{\mathrm{P}_{i,q}|_1 - C_{MC,q}}{\frac{4}{2-\psi_i} - \psi_{MC}\frac{2}{(2-\psi_i)^2}} \\
&= \frac{\mathrm{P}_{i,q}|_1 - C_{MC,q}}{2 - \psi_{MC}\frac{1}{(2-\psi_i)}}.
\end{aligned} \tag{46}$$

Namely, we have $C_{i,q} = \frac{\mathrm{P}_{i,q}|_1 - C_{MC,q}}{2-\psi_{MC}\frac{1}{(2-\psi_i)}} + C_{MC,q}$. The theorem is proved. ■

Appendix D
Proof of Theorem 4

From theorem 2, we can know that the proposal of the MCBS in the second round is related to that in the first round, namely,

$$\Gamma_{MC,q}|_2 = \frac{\Gamma_{MC,q}|_1}{2-\psi_i}. \tag{47}$$

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TVT.2020.3041587, IEEE Transactions on Vehicular Technology

By substituting (44) into (47), the optimal strategy of the MCBS in the second round can be expressed as

$$\begin{aligned} \digamma_{MC,q}|_2 &= \frac{\digamma_{MC,q}|_1}{2-\psi_i} \\ &= \frac{1}{2-\psi_i} \frac{\mathrm{P}_{i,q}|_1 - C_{MC,q}}{\frac{4}{2-\psi_i} - \psi_{MC}\frac{2}{(2-\psi_i)^2}} \\ &= \frac{\mathrm{P}_{i,q}|_1 - C_{MC,q}}{4 - \psi_{MC}\frac{2}{(2-\psi_i)}}. \end{aligned} \tag{48}$$

The theorem is proved. ■

APPENDIX E
PROOF OF THEOREM 5

In the second round of the bargaining game, if the AV rejects $\digamma_{MC,q}|_2$, its utility will be zero. Therefore, the condition that the AV accepts $\digamma_{MC,q}|_2$ is

$$C_{i,q} - C_{MC,q} > \digamma_{MC,q}|_2. \tag{49}$$

Combining (48) and (49), the condition that the AV accepts $\digamma_{MC,q}|_2$ becomes

$$\begin{aligned} C_{i,q} - C_{MC,q} &> \digamma_{MC,q}|_2 \\ &= \frac{\mathrm{P}_{i,q}|_1 - C_{MC,q}}{4 - \psi_{MC}\frac{2}{(2-\psi_i)}}. \end{aligned} \tag{50}$$

From (50), we have $C_{i,q} = \frac{\mathrm{P}_{i,q}|_1 - C_{MC,q}}{4-\psi_{MC}\frac{2}{(2-\psi_i)}} + C_{MC,q}$. The theorem is proved. ■

REFERENCES

[1] J. Wang, J. Liu, and N. Kato, "Networking and communications in autonomous driving: A survey," *IEEE Communications Surveys & Tutorials*, vol. 21, no. 2, pp. 1243–1274, Secondquarter 2019.

[2] Z. Su, Y. Hui, and T. H. Luan, "Distributed task allocation to enable collaborative autonomous driving with network softwarization," *IEEE J. Sel. Areas in Commun.*, vol. 36, no. 10, pp. 2175–2189, Oct. 2018.

[3] H. Peng, Q. Ye, and X. S. Shen, "Sdn-based resource management for autonomous vehicular networks: A multi-access edge computing approach," *IEEE Wireless Communications*, vol. 26, no. 4, pp. 156–162, Aug. 2019.

[4] Y. Wu, L. P. Qian, H. Mao, X. Yang, H. Zhou, X. Tan, and D. H. K. Tsang, "Secrecy-driven resource management for vehicular computation offloading networks," *IEEE Network*, vol. 32, no. 3, pp. 84–91, May 2018.

[5] M. Du, K. Wang, Y. Chen, X. Wang, and Y. Sun, "Big data privacy preserving in multi-access edge computing for heterogeneous internet of things," *IEEE Commun. Mag.*, vol. 56, no. 8, pp. 62–67, Aug. 2018.

[6] Y. Zhou, F. R. Yu, J. Chen, and Y. Kuo, "Resource allocation for information-centric virtualized heterogeneous networks with in-network caching and mobile edge computing," *IEEE Transactions on Vehicular Technology*, vol. 66, no. 12, pp. 11 339–11 351, Dec. 2017.

[7] Q. Xu, Z. Su, Q. Zheng, M. Luo, B. Dong, and K. Zhang, "Game theoretical secure caching scheme in multihoming edge computing-enabled heterogeneous networks," *IEEE Internet of Things Journal*, vol. 6, no. 3, pp. 4536–4546, June 2019.

[8] Y. Hui, Z. Su, and T. H. Luan, "Collaborative content delivery in software-defined heterogeneous vehicular networks," *IEEE/ACM Transactions on Networking*, vol. 28, no. 2, pp. 575–587, Apr. 2020.

[9] J. Li, S. Chu, F. Shu, J. Wu, and D. N. K. Jayakody, "Contract-based small-cell caching for data disseminations in ultra-dense cellular networks," *IEEE Transactions on Mobile Computing*, vol. 18, no. 5, pp. 1042–1053, May 2019.

[10] L. Wang, F. Tian, T. Svensson, D. Feng, M. Song, and S. Li, "Exploiting full duplex for device-to-device communications in heterogeneous networks," *IEEE Communications Magazine*, vol. 53, no. 5, pp. 146–152, May 2015.

[11] J. Feng, Z. Liu, C. Wu, and Y. Ji, "Ave: Autonomous vehicular edge computing framework with aco-based scheduling," *IEEE Transactions on Vehicular Technology*, vol. 66, no. 12, pp. 10 660–10 675, Dec. 2017.

[12] K. Wang, H. Yin, W. Quan, and G. Min, "Enabling collaborative edge computing for software defined vehicular networks," *IEEE Network*, vol. 32, no. 5, pp. 112–117, Otc. 2018.

[13] J. Zhang and K. B. Letaief, "Mobile edge intelligence and computing for the internet of vehicles," *Proceedings of the IEEE*, vol. 108, no. 2, pp. 246–261, Feb. 2020.

[14] Y. Hui, Z. Su, T. H. Luan, and J. Cai, "Content in motion: An edge computing based relay scheme for content dissemination in urban vehicular networks," *IEEE Transactions on Intelligent Transportation Systems*, vol. 20, no. 8, pp. 3115–3128, Aug. 2019.

[15] Z. Zhou, H. Yu, C. Xu, Z. Chang, S. Mumtaz, and J. Rodriguez, "Begin: Big data enabled energy-efficient vehicular edge computing," *IEEE Communications Magazine*, vol. 56, no. 12, pp. 82–89, Dec. 2018.

[16] Z. Zhou, B. Wang, M. Dong, and K. Ota, "Secure and efficient vehicle-to-grid energy trading in cyber physical systems: Integration of blockchain and edge computing," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 50, no. 1, pp. 43–57, Jan. 2020.

[17] X. Huang, R. Yu, J. Kang, Y. He, and Y. Zhang, "Exploring mobile edge computing for 5g-enabled software defined vehicular networks," *IEEE Wireless Communications*, vol. 24, no. 6, pp. 55–63, Dec. 2017.

[18] H. Peng, Q. Ye, and X. Shen, "Spectrum management for multi-access edge computing in autonomous vehicular networks," *IEEE Transactions on Intelligent Transportation Systems*, vol. 21, no. 7, pp. 3001–3012, July 2020.

[19] Y. Hui, Z. Su, T. H. Luan, and C. Li, "Reservation service: Trusted relay selection for edge computing services in vehicular networks," *IEEE J. Sel. Areas in Commun.*, vol. 38, no. 12, pp. 2734–2746, Dec. 2020.

[20] K. Wang, H. Yin, W. Quan, and G. Min, "Enabling collaborative edge computing for software defined vehicular networks," *IEEE Network*, vol. 32, no. 5, pp. 112–117, Sep. 2018.

[21] H. Li, K. Ota, and M. Dong, "Eccn: Orchestration of edge-centric computing and content-centric networking in the 5g radio access network," *IEEE Wireless Communications*, vol. 25, no. 3, pp. 88–93, June 2018.

[22] J. Li, G. Luo, N. Cheng, Q. Yuan, Z. Wu, S. Gao, and Z. Liu, "An end-to-end load balancer based on deep learning for vehicular network traffic control," *IEEE Internet Things J.*, vol. 6, no. 1, pp. 953–966, Feb. 2019.

[23] H. Guo, J. Zhang, and J. Liu, "Fiwi-enhanced vehicular edge computing networks: Collaborative task offloading," *IEEE Vehicular Technology Magazine*, vol. 14, no. 1, pp. 45–53, Mar. 2019.

[24] Y. Li, D. Jin, Z. Wang, L. Zeng, and S. Chen, "Coding or not: Optimal mobile data offloading in opportunistic vehicular networks," *IEEE Transactions on Intelligent Transportation Systems*, vol. 15, no. 1, pp. 318–333, Feb. 2014.

[25] F. Sun, F. Hou, N. Cheng, M. Wang, H. Zhou, L. Gui, and X. Shen, "Cooperative task scheduling for computation offloading in vehicular cloud," *IEEE Transactions on Vehicular Technology*, vol. 67, no. 11, pp. 11 049–11 061, Nov. 2018.

[26] Z. Zhou, H. Liao, X. Zhao, B. Ai, and M. Guizani, "Reliable task offloading for vehicular fog computing under information asymmetry and information uncertainty," *IEEE Transactions on Vehicular Technology*, vol. 68, no. 9, pp. 8322–8335, Sept. 2019.

[27] J. Zhao, Q. Li, Y. Gong, and K. Zhang, "Computation offloading and resource allocation for cloud assisted mobile edge computing in vehicular networks," *IEEE Transactions on Vehicular Technology*, vol. 68, no. 8, pp. 7944–7956, Aug. 2019.

[28] Y. Sun, X. Guo, J. Song, S. Zhou, Z. Jiang, X. Liu, and Z. Niu, "Adaptive learning-based task offloading for vehicular edge computing systems," *IEEE Transactions on Vehicular Technology*, vol. 68, no. 4, pp. 3061–3074, Apr. 2019.

[29] H. Cao and J. Cai, "Distributed multiuser computation offloading for cloudlet-based mobile cloud computing: A game-theoretic machine learning approach," *IEEE Transactions on Vehicular Technology*, vol. 67, no. 1, pp. 752–764, Jan. 2018.

[30] M. Liwang, S. Dai, Z. Gao, Y. Tang, and H. Dai, "A truthful reverse-auction mechanism for computation offloading in cloud-enabled vehicular network," *IEEE Internet of Things Journal*, vol. 6, no. 3, pp. 4214–4227, June 2019.

[31] N. Cheng, N. Lu, N. Zhang, X. Zhang, X. S. Shen, and J. W. Mark, "Opportunistic wifi offloading in vehicular environment: A game-theory approach," *IEEE Transactions on Intelligent Transportation Systems*, vol. 17, no. 7, pp. 1944–1955, July 2016.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TVT.2020.3041587, IEEE Transactions on Vehicular Technology

[32] K. Wang, F. C. M. Lau, L. Chen, and R. Schober, "Pricing mobile data offloading: A distributed market framework," *IEEE Transactions on Wireless Communications*, vol. 15, no. 2, pp. 913–927, Feb. 2016.

[33] J. Zheng, Y. Cai, Y. Wu, and X. Shen, "Dynamic computation offloading for mobile cloud computing: A stochastic game-theoretic approach," *IEEE Transactions on Mobile Computing*, vol. 18, no. 4, pp. 771–786, April 2019.

[34] Z. Hong, W. Chen, H. Huang, S. Guo, and Z. Zheng, "Multi-hop cooperative computation offloading for industrial iotcedgeccloud computing environments," *IEEE Transactions on Parallel and Distributed Systems*, vol. 30, no. 12, pp. 2759–2774, Dec. 2019.

[35] X. Peng, K. Ota, and M. Dong, "Multiattribute-based double auction toward resource allocation in vehicular fog computing," *IEEE Internet of Things Journal*, vol. 7, no. 4, pp. 3094–3103, Apr. 2020.

[36] F. Lyu, H. Zhu, N. Cheng, H. Zhou, W. Xu, M. Li, and X. Shen, "Characterizing urban vehicle-to-vehicle communications for reliable safety applications," *IEEE Transactions on Intelligent Transportation Systems*, vol. 21, no. 6, pp. 2586–2602, 2020.

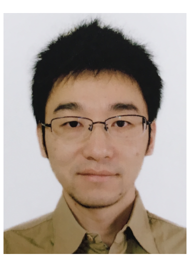
[37] S. Yu, R. Langar, X. Fu, L. Wang, and Z. Han, "Computation offloading with data caching enhancement for mobile edge computing," *IEEE Transactions on Vehicular Technology*, vol. 67, no. 11, pp. 11 098–11 112, Nov. 2018.

[38] W. Wu, N. Zhang, N. Cheng, Y. Tang, K. Aldubaikhy, and X. Shen, "Beef up mmwave dense cellular networks with d2d-assisted cooperative edge caching," *IEEE Transactions on Vehicular Technology*, vol. 68, no. 4, pp. 3890–3904, Apr. 2019.

[39] Z. Su, Y. Hui, and Q. Yang, "The next generation vehicular networks: A content-centric framework," *IEEE Wireless Communications*, vol. 24, no. 1, pp. 60–66, Feb. 2017.

[40] S. Zhang, P. He, K. Suto, P. Yang, L. Zhao, and X. Shen, "Cooperative edge caching in user-centric clustered mobile networks," *IEEE Transactions on Mobile Computing*, vol. 17, no. 8, pp. 1791–1805, Aug. 2018.

Yilong Hui received the Ph.D. degree in control theory and control engineering from Shanghai University, Shanghai, China, in 2018. He is currently a lecturer with the State Key Laboratory of Integrated Services Networks, and with the School of Telecommunication Engineering, Xidian University, China. He has published over 30 scientific articles in leading journals and international conferences. His research interests include mobile edge computing, vehicular networks, intelligent transportation systems and autonomous driving.



Zhou Su received the Ph.D. degree from Waseda University, Tokyo, Japan, in 2003. Dr. Su is an Associate Editor of IET Communications. He is the Chair of the Multimedia Services and Applications over Emerging Networks Interest Group (MENIG) of the IEEE Comsoc Society and the Multimedia Communications Technical Committee. He has also served as the Co-Chair of several international conferences including IEEE VTC Spring 2016, IEEE CCNC 2011, etc. He is a TPC Member of some flagship conferences including IEEE INFOCOM, IEEE ICC, IEEE Globecom, etc. His research interests include multimedia communication, wireless communication and network traffic. He received the Best Paper Award of International Conference IEEE CyberSciTech2017, WiCon2016, CHINACOM2008 and Funai Information Technology Award for Young Researchers in 2009.

Tom H. Luan is a professor with the School of Engineering, Xidian University, Xi'an, China. He received the B.E. degree from the Xi'an Jiaotong University, China, in 2004, the Master degree from the Hong Kong University of Science and Technology, Hong Kong, in 2007, and the Ph.D. degree from the University of Waterloo, Canada, in 2012, all in Electrical and Computer Engineering. Dr. Luan's research mainly focuses on the content distribution and media streaming in the vehicular ad hoc networks and peer-to-peer networking, protocol design and performance evaluation of wireless cloud computing and fog computing. Dr. Luan has authored/coauthored around 40 journal papers and 30 technical papers in conference proceedings, and awarded one US patent. He served as a TPC member for IEEE Globecom, ICC, PIMRC and the technical reviewer for multiple IEEE Transactions including TMC, TPDS, TVT, TWC and ITS.

Changle Li (M'09-SM'16) received the Ph.D. degree in communication and information system from Xidian University, China, in 2005. He was a Post-Doctoral Researcher in Canada and the National Institute of Information and Communications Technology, Japan. He was a Visiting Scholar with the University of Technology Sydney. He is currently a Professor with the State Key Laboratory of Integrated Services Networks, Xidian University. His research interests include intelligent transportation systems, vehicular networks, mobile ad hoc networks, and wireless sensor networks.

Guoqiang Mao (S'98-M'02-SM'08-F'18) has published over 200 papers in international conferences and journals, which have been cited more than 9000 times. He is an editor of the IEEE Transactions on Intelligent Transportation Systems (since 2018), IEEE Transactions on Wireless Communications (2014-2019), IEEE Transactions on Vehicular Technology (since 2010) and received Top Editor award for outstanding contributions to the IEEE Transactions on Vehicular Technology in 2011, 2014 and 2015. He was a co-chair of IEEE Intelligent Transport Systems Society Technical Committee on Communication Networks. He has served as a chair, co-chair and TPC member in a number of international conferences. His research interest includes intelligent transport systems, applied graph theory and its applications in telecommunications, Internet of Things, wireless sensor networks, wireless localization techniques and network modeling and performance analysis.

Weigang Wu received the B.Sc. degree in 1998 and the M.Sc. degree in 2003, both from Xi'an Jiaotong University, China. He received the Ph.D. degree in computer science in 2007 from Hong Kong Polytechnic University. He is currently a full professor at the School of Data and Computer Science, Sun Yat-sen University, China. His research interests include distributed systems and wireless networks, especially cloud computing platforms and ad hoc networks. He has published more than 60 papers in major conferences and journals. He has served as a member of editorial board of two international journals, Frontiers of Computer Science, and Ad Hoc & Sensor Wireless Networks.