

Research Article

New Bargaining Game Model for Collaborative Vehicular Network Services

Sungwook Kim 

Department of Computer Science, Sogang University, 35 Baekbeom-ro (Sinsu-dong), Mapo-gu, Seoul 121-742, Republic of Korea

Correspondence should be addressed to Sungwook Kim; swkim01@sogang.ac.kr

Received 4 June 2018; Revised 14 July 2018; Accepted 7 February 2019; Published 7 March 2019

Academic Editor: Michael Vassilakopoulos

Copyright © 2019 Sungwook Kim. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

The wireless industry's evolution from fourth generation (4G) to fifth generation (5G) will lead to extensive progress in new vehicular network environments, such as crowdsensing, cloud computing, and routing. Vehicular crowdsensing exploits the mobility of vehicles to provide location-based services, whereas vehicular cloud computing is a new hybrid technology that instantly uses vehicular network resources, such as computing, storage, and Internet for decision-making. In this study, novel crowdsensing, cloud computing, and routing algorithms are developed for a next-generation vehicular network, and they are combined into a hybrid scheme. To provide an effective solution toward an appropriate interactive vehicular network process, we focus on the paradigm of learning algorithms and game theory. Based on individual cooperation, our proposed scheme is designed as a triple-plane game model to adapt to the dynamics of a vehicular network system. The primary advantage of our game-based approach is to provide self-adaptability and responsiveness to current network environments. The performance of our hybrid scheme is evaluated and analyzed using simulation experiments in terms of the cloud service success ratio, normalized dissemination throughput, and crowdsensing success probability.

1. Introduction

With the rapid development of mobile communication and intelligent transportation technologies, Vehicular Ad hoc Network (VANET) has emerged as an important part of the intelligent transportation system (ITS). VANET is a unique type of Mobile Ad hoc Network (MANET). The similarity between VANET and MANET is characterized by the movement and self-organization of nodes. However, the primary difference is that the nodes in VANETs are mobile vehicles; they move at much higher speeds and possess substantial power resources, which is an advantage over traditional MANETs. As a newly introduced technology, the goal of VANET research is to develop vehicle-to-vehicle (V2V) communications to enable quick and cost-efficient data sharing for enhanced vehicle safety and comfort [1–3].

Typically, vehicles send and receive messages using an on-board unit (OBU), which is a portable data collector hardware installed on the vehicle. By a combination of information technology and transport infrastructure, VANETs support a wide range of applications including traffic control, road

safety, route optimization, multimedia entertainment, and social interactions. With the advancement in the latest vehicular technology, automobile manufacturers and academia are heavily engaged in the blueprint of VANETs. Currently, 10% of moving vehicles on the road are wirelessly connected. By 2020, it is reported that 90% of vehicles will be connected [1, 4, 5].

In VANET systems, each vehicle can exchange messages in V2V communications while acting as a sender, receiver, and router. In addition, these vehicles perform vehicle-to-roadside (V2R) communications with roadside units (RSUs) [6]. The RSU operates as an access point to provide services to moving vehicles at anytime and anywhere while connecting to the backbone network. As Internet access points, RSUs provide the communication connectivity to passing vehicles. In VANET environments, RSUs and OBUs in vehicles repeatedly interact with each other and perform control decisions individually without centralized coordination [1,4–9].

During VANET operations, services can be categorized into three fundamental classes: vehicular cloud service (VCS), vehicular sensing service (VSS), and vehicular routing service

(VRS). Vehicular cloud service is a new paradigm that exploits cloud computing to serve VANETs with several computational services [10]. Currently, computing requirements for vehicular applications are increasing rapidly, particularly with the growing interest in embedding a new class of interactive applications and services. Within the communication range of each RSU, the RSU controls the VCS through interactions with the vehicles and provides computation capacity, storage, and data resources. Vehicles requiring resources can obtain the available resources via interconnected RSUs. Therefore, depending on the roadside infrastructure, the VCS provides a significant market opportunity and overcomes challenges, making the technology more cost-effective [10, 11].

With the rapid development of VANETs, VSSs have broad potential for enhancing the public sensing domain. Vehicles with OBUs can be considered as mobile sensors that collect the local information requested by the RSUs. As a server, the RSU pays for the VSS's participating vehicles according to their sensing qualities. As a type of vehicle-to-infrastructure communication, examples of VSSs are location-based road monitoring, real-time traffic condition reporting, pollution level measurements, and public information sharing. Owing to the mobility of the vehicles, the sensing coverage of location-based services can be expanded, and VSSs will benefit a wide scope of consumers [7, 8, 12].

In VANETs, vehicles are equipped with OBUs that can communicate with each other, also known as VRS. The VRS requires the suitable V2V communication technology, especially the routing method to address various challenges of VANETs. Typically, VANET is considered a subclass of MANET, and MANET-based routing protocols have been extensively studied. However, the VANET has several characteristics that distinguish it from MANET. To obtain adaptive vehicular routing protocols, high mobility, rapid network topology change, and an unlimited power supply must be considered to design an effective VRS algorithm [13–15].

In this study, we propose a new VANET control scheme, by considering the issues of VCS, VRS, and VSS. Based on the joint design of different vehicular operations, our integrated approach can obtain a synergy effect while attaining an appropriate performance balance. However, it is an extremely challenging task to combine the VCS, VRS, and VSS algorithms into a holistic scheme. Therefore, a new control paradigm is required. Currently, game theory has been widely applied to analyze the performance of VANET systems. More specifically, VANET system agents are assumed as game players, and the interaction among players can be formulated as a game model [6].

1.1. Motivation and Contribution. The aim of this study is to propose novel VCS, VRS, and VSS algorithms based on the different game models. For the VANET operations, game theory serves as an effective model of interactive situations among rational VANET agents. In the VCS algorithm, vehicles share the cloud resource in the RSU to increase the quality of service. The interactions among vehicles and RSUs

are modeled as a non-cooperative game model. In the VRS algorithm, vehicles attempt to increase the quality and reliability of routing services. In the VSS algorithm, vehicles act as mobile sensors to collect the required information. As system servers, RSUs operate to share and process several types of collected information. To collect information, RSUs use a learning-based incentive mechanism, and individual vehicles adopt the cooperative bargaining approach among their VCSs, VSSs, and VRSs to balance their performance.

The proposed VCS, VRS, and VSS algorithms may need to coexist and complement each other to meet the diverse requirements of VANETs. To investigate the strategic interactions among different control algorithms, we develop a new triple-plane game model across the VCS, VRS, and VSS algorithms. For effective VANET management, vehicles in the VSS algorithm select their strategies from the viewpoint of a bargaining solution. In a distributed manner, the idea of the classical Nash bargaining solution is used to implement a triple-plane bargaining process in each vehicle. The proposed triple-plane bargaining approach imitating the interactive sequential game process is suitable to negotiate their conflicts of interest while ensuring the system practicality.

The main contributions of this study can be summarized as follows: (i) we developed novel VCS, VRS, and VSS algorithms for VANETs; (ii) we employed a new game model to provide satisfactory services; (iii) we adopted a distributed triple-plane bargaining approach to balance contradictory requirements; (iv) self-adaptability is effectively implemented in each control decision process; and (v) we captured the dynamic interactions of game players depending on their different viewpoints. According to the reciprocal combination of optimality and practicality, our approach can provide a desirable solution for real-world VANET applications. To the best of our knowledge, little research has been conducted on integrated-bargaining-based algorithms for VANET control problems.

1.2. Organization. The remainder of this article is organized as follows. In the next section, we review some related VANET schemes and their problems. In section 3, we provide a detailed description of the VCS, VRS, and VSS algorithms before defining the triple-plane bargaining game model. In particular, this section provides fresh insights into the design benefit of our game-based approach. For convenience, the primary steps of the proposed scheme are listed. In section 4, we validate the performance of the proposed scheme by means of a comparison with some existing methods. Finally, we present our conclusion and discuss some ideas for the future work.

2. Related Work

There have been considerable researches into the design of VANET control schemes. Cheng et al. [16] investigate the opportunistic spectrum access for cognitive radio vehicular ad hoc networks. In particular, the spectrum access process is modeled as a non-cooperative congestion game. Therefore, the proposed scheme in [16] opportunistically accesses

licensed channels in a distributed manner and is designed to achieve a pure Nash equilibrium with high efficiency and fairness. By using the statistics of the channel availability, this approach can exploit the spatial and temporal access opportunities of the licensed spectrum. Through the simulation results, they conform that the proposed approach achieves higher utility and fairness compared to a random access scheme [16].

The paper [17] focuses on how to stimulate message forwarding in VANETs. To implement this mechanism, it is crucial to make sure that vehicles have incentives to forward messages for others. Especially, this study adopts the coalitional game theory to solve the forwarding cooperation problem and rigorously analyzing it in VANETs. The main goal is to ensure that whenever a message needs to be forwarded in a VANET, all involved vehicles gets their incentives to form a grand coalition. In addition, this approach is extended to taking the limited storage space of each node. Experiments on testbed trace data verify that the proposed method is effective for the stimulating cooperation of message forwarding in VANETs [17].

In [18], the main focus is to achieve a higher throughput by selecting the best data rate and the best network or channel based on the cognitive radio VANET mechanism. By changing wireless channels and data rate in heterogeneous wireless networks, the scheme in [18] is designed as a game theoretic model to achieve a higher throughput for vehicular users. In addition, a new idea adopted to find optimal number of APs for given VANET scenarios [18]. Even though the papers [16–18] have considered some game theory-based VANET control algorithms, they are designed as one-sided protocols while focusing a specific control issue. Therefore, the system interaction process is not fully investigated.

Recently, the vehicular microcloud technique has been considered one of the solutions to address the challenges and issues of VANETs. It is a new hybrid technology that has a remarkable impact on traffic management and road safety. By instantly clustering, cars help aggregating collected data that are transferred to some backend [19, 20]. The paper [19] introduces the concept of vehicular microcloud as a virtual edge server for efficient connections between cars and backend infrastructure. To realize such virtual edge servers, map-based clustering technique is needed to cope with the dynamicity of vehicular networks. This approach can optimize the data processing and aggregation before sending them to a backend [19].

F. Hagenauer et al. propose an idea that vehicular microcloud clusters of parked cars act as RSUs by investigating the virtual vehicular network infrastructure [20]. Especially, they focus on two control questions: (i) the selection of gateway nodes to connect moving cars to the cluster and (ii) a mechanism for seamless handover between gateways. For the first question, they select only a subset of all parked cars as gateways. This gateways selection helps significantly to reduce the channel load. For the second question, they enable driving cars to find better suited gateways while allowing the driving car to maintain connections [20]. The ideas in [19, 20] are very interesting. However, they can be used for special situations. Therefore,

it is difficult to apply these ideas in general VANET operation scenarios.

During VANETs' operations, there are two fundamental techniques to disseminate data for vehicular applications: Vehicle-to-infrastructure (V2I) and vehicle-to-vehicle (V2V) communications. Recently, Kim proposes novel V2I and V2V algorithms for the effective VANET management [6]. For V2I communications, a novel vertical game model is developed as an incentive-based crowdsensing algorithm. For V2V communications, a new horizontal game model is designed to implement a learning-based intervehicle routing protocol. Under dynamically changing VANET environments, the developed scheme in [6] is formulated as a dual-plane vertical-horizontal game model. By considering the real-world VANET environments, this approach imitates the interactive sequential game process by using self-monitoring and distributed learning techniques. It is suitable to ensure the system practicality while effectively maximizing the expected benefits [6].

In this paper, we develop a novel control scheme for vehicular network services. Our proposed scheme also includes sensing and routing algorithms for the operations of VANETs. Therefore, our proposed scheme may look similar to the method in [6]. First of all, the resemblance can be summarized as follows: (i) in the vehicular routing algorithm, link status and path cost estimation methods and each vehicle's utility function are designed in the same manner; (ii) in the crowdsensing algorithm, utility functions of vehicle and RSU are defined similarly; and (iii) in the learning process, strategy selection probabilities are estimated based on the Boltzmann distribution equation. While both schemes have some similarities, there are several key differences. First, the scheme in [6] is designed as a dual-plane game model, but our proposed scheme is extended as a triple-plane game model. Second, a new cloud computing algorithm is developed to share the cloud resource in the RSU while increasing the quality of service. Third, to reduce computation complexity, entropy-based routing route decision mechanism is replaced by an online route decision mechanism. Fourth, the scheme in [6] is modeled as a non-cooperative game approach. However, our proposed scheme adopts a dynamic bargaining solution. Therefore, in our proposed sensing algorithm, each individual vehicle decides his strategy as a cooperative game manner. Fifth, our proposed scheme fully considers cloud computing, sensing, and routing payoffs and calculates the bargaining powers to respond to current VANET situations. Most of all, the main difference between the scheme in [6] and our proposed scheme is a control paradigm. Simply, in the paper [6], V2I and V2V algorithms are combined merely based on the competitive approach. However, in this paper, sensing, cloud computing, and routing algorithms work together in a collaborative fashion and act cooperatively with each, other based on the interactive feedback process.

The *Multiobjective Vehicular Interaction Game (MVIg)* scheme [21] proposes a new game theoretic scheme to control the on-demand service provision in a vehicular cloud. Based on a game theoretic approach, this scheme can balance the overall game while enhancing vehicles' service

costs. The game system in the *MVIG* scheme differs from other conventional models, as it allows vehicles to prioritize their preferences. In addition, a quality-of-experience framework is also developed to provision various types of services in a vehicular cloud; it is a simple but an effective model to determine vehicle preferences while ensuring fair game treatment. The simulation results show that the *MVIG* scheme outperforms other conventional models [21].

The *Prefetching-Based Vehicular Data Dissemination (PVDD)* scheme [22] devises a vehicle route-based data prefetching approach while improving the data transmission accessibility under dynamic wireless connectivity and limited data storage environments. To put it more concretely, the *PVDD* scheme develops two control algorithms to determine how to prefetch a data set from a data center to roadside wireless access points. Based on the greedy approach, the first control algorithm is developed to solve the dissemination problem. Based on the online learning manner, the second control algorithm gradually learns the success rate of unknown network connectivity and determines an optimal binary decision matrix at each iteration. Finally, this study proves that the first control algorithm could find a suboptimal solution in polynomial time, and the optimal solution of the second control algorithm converges to a globally optimal solution in a certain number of iterations using regret analysis [22].

The *Cooperative Relaying Vehicular Cloud (CRVC)* scheme [23] proposes a novel cooperative vehicular relaying algorithm over a long-term evolution-advanced (LTE-A) network. To maximize the vehicular network capacity, this scheme uses vehicles for cooperative relaying nodes in broadband cellular networks. With new functionalities, the *CRVC* scheme can (i) reduce power consumption, (ii) provide a higher throughput, (iii) lower operational expenditures, (iv) ensure a more flexibility, and (v) increase a covering area. In a heavily populated urban area, the *CRVC* scheme is useful owing to a large number of relaying vehicles. Finally, the performance improvement is demonstrated through the simulation analysis [23]. In this paper, we compare the performance of our proposed scheme with the existing *MVIG* [21], *PVDD* [22], *CRVC* [23] schemes.

3. The Proposed VCS, VRS, and VSS Algorithms

In this section, the proposed VCS, VRS, and VSS algorithms are presented in detail. Based on the learning algorithm and game-based approach, these algorithms form a new triple-plane bargaining game model to adapt the fast changing VANET environments.

3.1. Game Models for the VCS, VRS, and VSS Algorithms. For the operation of a VANET system, we develop three different game models for the VCS, VRS, and VSS algorithms. As game players, vehicles and RSUs select their strategies based on the interactions of other players. In our proposed scheme, the VCS and VRS algorithms are formulated as non-cooperative game models, and the VSS algorithm is formulated as a triple-plane bargaining model

by cooperation, coordination, and collaboration of VCS, VRS, and VSS processes. First, for the VCS algorithm, we formally define the game model $\mathbb{G}^{\text{VCS}} = \{\mathbb{N}, \mathbb{C}, \{\mathbb{S}^{\mathcal{R}}, \mathbb{S}_{\mathcal{V}_i \in \mathcal{V}}^{\mathcal{V}}\}, \{\mathbb{U}^{\mathcal{R}}, \mathbb{U}_{\mathcal{V}_i \in \mathcal{V}}^{\mathcal{V}}\}, \mathbb{L}^{\mathbb{P}_j^{\mathcal{R}} \in \mathbb{S}^{\mathcal{R}}}, T\}$ at each time period of gameplay.

- (i) \mathbb{N} is the finite set of VCS game players $\mathbb{N} = \{\mathcal{R}, \mathcal{V}\}$, where \mathcal{R} represents a RSU and $\mathcal{V} = \{\mathcal{V}_1, \dots, \mathcal{V}_i, \dots, \mathcal{V}_k\}$ is a set of multiple vehicles in the \mathcal{R} 's coverage area
- (ii) \mathbb{C} is the cloud computation capacity, i.e., number of CPU cycles per second in the \mathcal{R}
- (iii) $\mathbb{S}^{\mathcal{R}} = \{\mathbb{P}_{\min}^{\mathcal{R}}, \dots, \mathbb{P}_j^{\mathcal{R}}, \dots, \mathbb{P}_{\max}^{\mathcal{R}}\}$ is a set of \mathcal{R} 's strategies, where $\mathbb{P}_j^{\mathcal{R}}$ means the price level to execute one basic cloud service unit (BCSU). $\mathbb{S}_{\mathcal{V}_i \in \mathcal{V}}^{\mathcal{V}}$ is the \mathcal{V}_i 's available strategies $\mathbb{S}_{\mathcal{V}_i}^{\mathcal{V}} = [\nu_{\min}^{\mathcal{V}_i}, \dots, \nu_k^{\mathcal{V}_i}, \dots, \nu_{\max}^{\mathcal{V}_i}]$, where $\nu^{\mathcal{V}_i}$ represents the amount of cloud services of the \mathcal{V}_i and is specified in terms of BCSUs
- (iv) $\mathbb{U}^{\mathcal{R}}$ is the payoff received by the RSU, and $\mathbb{U}_{\mathcal{V}_i \in \mathcal{V}}^{\mathcal{V}}$ is the payoff received by \mathcal{V}_i during the VCS process
- (v) $\mathbb{L}^{\mathbb{P}_j^{\mathcal{R}} \in \mathbb{S}^{\mathcal{R}}}$ is the \mathcal{R} 's learning value for the strategy $\mathbb{P}_j^{\mathcal{R}}$; \mathbb{L} is used to estimate the probability distribution ($\mathbb{P}^{\mathcal{R}}$) for the next \mathcal{R} 's strategy selection
- (vi) $T = \{\mathcal{H}_1, \dots, \mathcal{H}_t, \mathcal{H}_{t+1}, \dots\}$ denotes time, which is represented by a sequence of time steps with imperfect information for the VCS game process

For the VRS algorithm, we define the game model $\mathbb{G}^{\text{VRS}} = \{\mathbb{v}, \mathcal{N}_{\mathcal{V}_i}, \mathbb{A}_{\mathcal{V}_i \in \mathbb{v}}, U_{\mathcal{V}_i \in \mathbb{v}}, T\}$ at each time period of \mathbb{G}^{VRS} gameplay. \mathbb{G}^{VRS} can formulate the interactions of vehicles for VANET routing operations.

- (i) \mathbb{v} is the finite set of game players $\mathbb{v} = \{\mathcal{V}_1, \dots, \mathcal{V}_n\}$, where n is the number of vehicles for the \mathbb{G}^{VRS} game.
- (ii) $\mathcal{N}_{\mathcal{V}_i}$ is the set of \mathcal{V}_i 's one-hop neighboring vehicles.
- (iii) $\mathbb{A}_{\mathcal{V}_i \in \mathbb{v}} = \{a_{\mathcal{V}_i}^{\mathcal{V}_m} \mid \mathcal{V}_m \in \mathcal{N}_{\mathcal{V}_i}\}$ is the finite set of \mathcal{V}_i 's available strategies, where $a_{\mathcal{V}_i}^{\mathcal{V}_m}$ represents the selection of \mathcal{V}_m to relay a routing packet.
- (iv) $U_{\mathcal{V}_i \in \mathbb{v}}$ is the payoff received by the \mathcal{V}_i during the VRS process.
- (v) $T = \{\mathcal{H}_1, \dots, \mathcal{H}_t, \mathcal{H}_{t+1}, \dots\}$ is the same as the T in \mathbb{G}^{VCS} .

For the VSS algorithm, we formally define the game model $\mathbb{G}^{\text{VSS}} = \{\mathbb{N}, \mathfrak{S}, \{\mathbb{S}^{\mathcal{R}}, \mathbb{S}_{\mathcal{V}_i \in \mathcal{V}}^{\mathcal{V}}\}, \{u^{\mathcal{R}}, u_{\mathcal{V}_i \in \mathcal{V}}^{\mathcal{V}}\}, \mathfrak{Z}^{\mathbb{P}_j^{\mathfrak{S}_s \in \mathfrak{S}}}, T\}$ at each time period of gameplay.

- (i) $\mathbb{N} = \{\mathcal{R}, \mathcal{V}\}$ is the finite set of game players for the \mathbb{G}^{VSS} game; \mathbb{N} is the same as \mathbb{N} in \mathbb{G}^{VCS} .
- (ii) \mathfrak{S} is the finite set of sensing tasks $\{\mathfrak{S} = \mathfrak{S}_1, \dots, \mathfrak{S}_s\}$ in \mathcal{R} , where s is the number of total sensing tasks.
- (iii) $\mathbb{S}^{\mathcal{R}} = [\eta_{\mathcal{R}}^{\mathfrak{S}_1}, \dots, \eta_{\mathcal{R}}^{\mathfrak{S}_j}, \dots, \eta_{\mathcal{R}}^{\mathfrak{S}_s}]$ is a vector of \mathcal{R} 's strategies, where $\eta_{\mathcal{R}}^{\mathfrak{S}_j}$ represents the strategy set for the \mathfrak{S}_j . In the \mathbb{G}^{VSS} game, $\eta_{\mathcal{R}}^{\mathfrak{S}_j}$ means the set of price levels for the crowdsensing work for the task \mathfrak{S}_j during a time period ($\mathcal{H} \in T$). Like as the \mathbb{G}^{VCS}

- game, $\eta_{\mathcal{R}}^{\mathfrak{S}_j}$ is defined as $\eta_{\mathcal{R}}^{\mathfrak{S}_j} = \{\prod_{\mathfrak{S}_j \in \mathcal{N}} \mathfrak{P}_i^{\mathfrak{S}_j} \mid \mathfrak{P}_i^{\mathfrak{S}_j} \in [\mathfrak{P}_{\min}^{\mathfrak{S}_j}, \dots, \mathfrak{P}_l^{\mathfrak{S}_j}, \dots, \mathfrak{P}_{\max}^{\mathfrak{S}_j}]\}$.
- (iv) $\mathfrak{S}_{\mathcal{V}_i}^{\mathcal{V}} = [\mu_{\mathcal{V}_i}^{\mathfrak{S}_1}, \dots, \mu_{\mathcal{V}_i}^{\mathfrak{S}_j}, \dots, \mu_{\mathcal{V}_i}^{\mathfrak{S}_N}]$ is the vector of \mathcal{V}_i 's strategies, where $\mu_{\mathcal{V}_i}^{\mathfrak{S}_j}$ represents the strategy for \mathfrak{S}_j . In the \mathbb{G}^{VSS} game, $\mu_{\mathcal{V}_i}^{\mathfrak{S}_j}$ means the \mathcal{V}_i 's active VSS participation, i.e., $\mu_{\mathcal{V}_i}^{\mathfrak{S}_j} = 1$, or not, i.e., $\mu_{\mathcal{V}_i}^{\mathfrak{S}_j} = 0$, the \mathfrak{S}_j during \mathcal{H} .
- (v) $u^{\mathcal{R}}$ is the payoff received by the RSU, and $u_{\mathcal{V}_i}^{\mathcal{V}}$ is the payoff received by \mathcal{V}_i during the VSS process.
- (vi) $\mathfrak{Z}^{\mathfrak{P}_i^{\mathfrak{S}_j} \in \eta_{\mathcal{R}}^{\mathfrak{S}_j}}$ is the learning value for the strategy $\mathfrak{P}_i^{\mathfrak{S}_j}$; \mathfrak{Z} is used to estimate the probability distribution ($\mathbf{P}^{\mathcal{R}}$) for the next \mathcal{R} 's strategy selection.
- (vii) $T = \{\mathcal{H}_1, \dots, \mathcal{H}_t, \mathcal{H}_{t+1}, \dots\}$ is the same as T in \mathbb{G}^{VCS} and \mathbb{G}^{VRS} .

3.2. The VCS Algorithm Based on a Non-Cooperative Game Model. In the next-generation VANET paradigm, diverse and miscellaneous applications require significant computational resources. However, in general, the computational capabilities of the vehicles are limited. To address this resource restriction problem, vehicular cloud technology is widely considered as a new paradigm to improve the VANET service performance. In the VCS process, applications can be offloaded to the remote cloud server to improve the resource utilization and computation performance [24].

Vehicular cloud servers are located in RSUs and execute the computation tasks received from vehicles. However, the service area of each RSU may be limited by the radio coverage. Due to the high mobility, vehicles may pass through several RSUs during the task offloading process. Therefore, service results must be migrated to another RSU, which can reach the corresponding vehicle. From the viewpoints of RSUs and vehicles, payoff maximization is their main concern. To reach a win-win situation, they rationally select their strategies. During the VCS process, each RSU is a *proposer* and individual vehicles are

responders; they interact with each other for their objectives during the VCS process.

To design our VCS algorithm, we consider the VCS platform consisting of one RSU (\mathcal{R}) and a set of mobile vehicles (\mathcal{V}); they formulate the game model \mathbb{G}^{VCS} . As a *proposer*, the \mathcal{R} 's strategy ($\mathfrak{P}_j^{\mathcal{R}} \in \mathbb{S}^{\mathcal{R}}$) indicates the offered price for one BCSU process. \mathcal{R} has its own utility function, which represents its net gain while providing the VCS process. The \mathcal{R} 's utility function with $\mathfrak{P}_j^{\mathcal{R}}$ strategy at the t 'th time step \mathcal{H}_t ($\mathbb{U}^{\mathcal{R}}(\mathcal{V}, \mathfrak{P}_j^{\mathcal{R}}(\mathcal{H}_t))$) is given by

$$\mathbb{U}^{\mathcal{R}}(\mathcal{V}, \mathfrak{P}_j^{\mathcal{R}}(\mathcal{H}_t)) = \sum_{\mathcal{V}_i \in \mathcal{V}} \left(\xi_{\mathcal{V}_i}(\mathcal{H}_t) \times \left(\left\{ \mathfrak{P}_j^{\mathcal{R}}(\mathcal{H}_t) \times r_k^{\mathcal{V}_i}(\mathcal{H}_t) \right\} - \mathcal{C}^{\mathcal{R}}(r_k^{\mathcal{V}_i}(\mathcal{H}_t)) \right) \right),$$

$$\text{s.t. } \mathfrak{P}_j^{\mathcal{R}}(\mathcal{H}_t) \in \mathbb{S}^{\mathcal{R}},$$

$$\mathcal{C}^{\mathcal{R}}(r_k^{\mathcal{V}_i}(\mathcal{H}_t)) = \Theta_{\text{BCSU}}^{\mathcal{R}} \times r_k^{\mathcal{V}_i}(\mathcal{H}_t),$$

$$\Theta_{\text{BCSU}}^{\mathcal{R}} = \frac{(\theta \times \mathfrak{M})}{\mathbb{C}},$$

$$\xi_{\mathcal{V}_i}(\mathcal{H}_t) = \begin{cases} 1, & \mathcal{V}_i \text{ selects } r_k^{\mathcal{V}_i} \in \mathbb{S}_{\mathcal{V}_i}^{\mathcal{V}} \text{ at } \mathcal{H}_t, \\ 0, & \text{otherwise (i.e. no VCS of } \mathcal{V}_i), \end{cases} \quad (1)$$

where $\mathcal{C}^{\mathcal{R}}(r_k^{\mathcal{V}_i}(\mathcal{H}_t))$ is the cost function to execute the \mathcal{V}_i 's cloud service ($r_k^{\mathcal{V}_i}(\mathcal{H}_t)$) at time \mathcal{H}_t . $\Theta_{\text{BCSU}}^{\mathcal{R}}$ is the processing cost to execute one BCSU, and \mathfrak{M} is the currently using capacity of \mathbb{C} . θ is the coefficient factor of cost calculation. In practice, the actual sensing cost $\mathcal{C}^{\mathcal{R}}(\cdot)$ is usually unknown by vehicles.

As a *responder*, the vehicle \mathcal{V}_i 's strategy ($\mathbb{S}_{\mathcal{V}_i}^{\mathcal{V}}$) represents the amount of cloud service. The payoff of \mathcal{V}_i can be defined as a function of the task offload level ($r_k^{\mathcal{V}_i}$) and service price ($\mathfrak{P}_j^{\mathcal{R}}$). Therefore, the \mathcal{V}_i 's utility function with $\mathfrak{P}_j^{\mathcal{R}}$ and $r_k^{\mathcal{V}_i}$ strategies at time \mathcal{H}_t ($\mathbb{U}_{\mathcal{V}_i \in \mathcal{V}}^{\mathcal{V}}(\mathfrak{P}_j^{\mathcal{R}}(\mathcal{H}_t), r_k^{\mathcal{V}_i}(\mathcal{H}_t))$) is computed as follows:

$$\mathbb{U}_{\mathcal{V}_i \in \mathcal{V}}^{\mathcal{V}}(\mathfrak{P}_j^{\mathcal{R}}(\mathcal{H}_t), r_k^{\mathcal{V}_i}(\mathcal{H}_t)) = \varrho^{\mathcal{V}_i}(\mathcal{H}_t) \times \left(\left(\mathfrak{S}^{\mathcal{V}_i} \times r_k^{\mathcal{V}_i}(\mathcal{H}_t) \right) - \left(\mathfrak{P}_j^{\mathcal{R}}(\mathcal{H}_t) \times r_k^{\mathcal{V}_i}(\mathcal{H}_t) \right) \right),$$

$$\text{s.t. } \varrho^{\mathcal{V}_i}(\mathcal{H}_t) = \begin{cases} 1 \text{ and } \Gamma_{\mathcal{H}_{t+1}}^{\mathcal{V}_i} = \Gamma_{\mathcal{H}_t}^{\mathcal{V}_i} - \left(\mathfrak{P}_j^{\mathcal{R}}(\mathcal{H}_t) \times r_k^{\mathcal{V}_i}(\mathcal{H}_t) \right), & \text{if } \left(\Gamma_{\mathcal{H}_t}^{\mathcal{V}_i} \geq \left(\mathfrak{P}_j^{\mathcal{R}}(\mathcal{H}_t) \times r_k^{\mathcal{V}_i}(\mathcal{H}_t) \right) \right) \\ & \text{and } \left(r_k^{\mathcal{V}_i} \in \mathbb{S}_{\mathcal{V}_i}^{\mathcal{V}} \text{ is selected} \right), \\ 0, & \text{otherwise (i.e. } \Gamma_{\mathcal{H}_t}^{\mathcal{V}_i} < \left(\mathfrak{P}_j^{\mathcal{R}}(\mathcal{H}_t) \times r_k^{\mathcal{V}_i}(\mathcal{H}_t) \right) \right), \end{cases} \quad (2)$$

where $\mathfrak{S}^{\mathcal{V}_i}$ is the \mathcal{V}_i 's profit factor if one BCSU is processed and $\Gamma_{\mathcal{H}_t}^{\mathcal{V}_i}$ is the amount of \mathcal{V}_i 's virtual money at time \mathcal{H}_t ; if \mathcal{V}_i has enough money to pay the VCS price, \mathcal{V}_i can request its cloud service ($r_k^{\mathcal{V}_i}$). As the \mathbb{G}^{VCS} game players, the RSU and vehicles attempt to maximize their utility functions. Interactions among game players continue

repeatedly during the VCS process over time. In particular, the RSU should consider the reactions from vehicles to determine the price strategy ($\mathfrak{P}_j^{\mathcal{R}}$). In this study, we develop a new learning method to decide an effective price policy for cloud services. If the strategy $\mathfrak{P}_j^{\mathcal{R}}$ is selected at time \mathcal{H}_{t-1} , the RSU updates the strategy $\mathfrak{P}_j^{\mathcal{R}}$'s learning

value for the next time step ($\mathbb{L}_{\mathcal{H}_t}^{\mathfrak{P}_j^{\mathcal{R}}}$) according to the following method:

$$\mathbb{L}_{\mathcal{H}_t}^{\mathfrak{P}_j^{\mathcal{R}}} = \max \left\{ \left[\left((1-\chi) \times \mathbb{L}_{\mathcal{H}_{t-1}}^{\mathfrak{P}_j^{\mathcal{R}}} \right) + \left(\chi \times \log_2 \left(\frac{\mathbb{U}^{\mathcal{R}}(\mathcal{V}, \mathfrak{P}_j^{\mathcal{R}}(\mathcal{H}_{t-1}))}{\sum_{\mathfrak{P}_k^{\mathcal{R}} \in \mathbb{S}^{\mathcal{R}}} \left(\mathbb{L}_{\mathcal{H}_{t-1}}^{\mathfrak{P}_k^{\mathcal{R}}} / \|\mathbb{S}^{\mathcal{R}}\| \right)} \right) \right] \right], 0 \right\}, \quad (3)$$

where $\|\mathbb{S}^{\mathcal{R}}\|$ is the cardinality of $\mathbb{S}^{\mathcal{R}}$ and χ is a learning rate that models how the \mathbb{L} values are updated. To implement the price learning mechanism in the RSU, a strategy selection distribution ($\mathcal{P}^{\mathcal{R}} \in \mathbb{P}^{\mathcal{R}}$) for the VCS is defined based on the $\mathbb{L}(\cdot)$ values. During the \mathbb{G}^{VCS} game process, we sequentially determine $\mathbb{P}^{\mathcal{R}}(\mathcal{H}_t) = \{\mathcal{P}_{\mathfrak{P}_{\min}^{\mathcal{R}}}(\mathcal{H}_t), \dots, \mathcal{P}_{\mathfrak{P}_j^{\mathcal{R}}}(\mathcal{H}_t), \dots, \mathcal{P}_{\mathfrak{P}_{\max}^{\mathcal{R}}}(\mathcal{H}_t)\}$ as the probability distribution of \mathcal{R} 's strategy selection at time \mathcal{H}_t . The $\mathfrak{P}_j^{\mathcal{R}}$ strategy selection probability at time \mathcal{H}_t ($\mathcal{P}_{\mathfrak{P}_j^{\mathcal{R}}}(\mathcal{H}_t)$) is defined as follows:

$$\mathcal{P}_{\mathfrak{P}_j^{\mathcal{R}}}(\mathcal{H}_t) = \frac{\text{EXP}\left(\mathbb{L}_{\mathcal{H}_{t-1}}^{\mathfrak{P}_j^{\mathcal{R}}}\right)}{\sum_{k=\min}^{\max} \left(\text{EXP}\left(\mathbb{L}_{\mathcal{H}_{t-1}}^{\mathfrak{P}_k^{\mathcal{R}}}\right)\right)}. \quad (4)$$

3.3. The VRS Algorithm Based on a Non-Cooperative Game Model. The main goal of VRS is to transmit data from a source vehicle to a destination vehicle via wireless multihop transmission techniques. In the wireless multihop transmission technique, the intermediate vehicles in a routing path should relay data as soon as possible from the source to destination [6, 13, 25]. In this section, we develop a non-cooperative game model (\mathbb{G}^{VRS}) for vehicular routing services. As game players in \mathbb{G}^{VRS} , vehicles dynamically decide their routing routes. To configure the routing topology, a link cost ($\mathcal{L}\mathcal{C}$) is defined to relatively handle dynamic VANET conditions. In this study, we define a wireless link status ($\mathcal{L}\mathcal{C}_{\mathcal{V}_i^{\mathcal{V}_j}}^{\mathcal{V}_i}(\mathcal{H}_t)$) between \mathcal{V}_i and \mathcal{V}_j at time \mathcal{H}_t as follows:

$$\mathcal{L}\mathcal{C}_{\mathcal{V}_i^{\mathcal{V}_j}}^{\mathcal{V}_i}(\mathcal{H}_t) = \left((1-\alpha) \times \left(\frac{d_{\mathcal{V}_i^{\mathcal{V}_j}}^{\mathcal{V}_i}(\mathcal{H}_t)}{d_{\mathcal{V}_i}^M} \right) \right) + \left(\alpha \times \frac{\{\vec{v}_j(\mathcal{H}_t)\}}{\max_{\mathcal{V}_h \in \mathfrak{M}_{\mathcal{V}_i}^{\mathcal{V}_j}} \{\vec{v}_h(\mathcal{H}_t)\}} \right), \quad (5)$$

where $d_{\mathcal{V}_i}^M$ is the maximum coverage range of \mathcal{V}_i and $d_{\mathcal{V}_i^{\mathcal{V}_j}}^{\mathcal{V}_i}(\mathcal{H}_t)$ is the relative distance between \mathcal{V}_i and \mathcal{V}_j at time \mathcal{H}_t . Let $\mathfrak{M}_{\mathcal{V}_i}^{\mathcal{V}_j}$ be the set of the neighboring vehicles of \mathcal{V}_i at time \mathcal{H}_t and $\vec{v}_j(\mathcal{H}_t)$ is the relative velocity of \mathcal{V}_j at time \mathcal{H}_t . For the adaptive $\mathcal{L}\mathcal{C}_{\mathcal{V}_i^{\mathcal{V}_j}}^{\mathcal{V}_i}(\mathcal{H}_t)$ estimation, the parameter α controls the relative weight between the distance and the velocity of corresponding vehicle.

For the VRS, a source vehicle configures a multihop routing path by using the Bellman–Ford algorithm. As a routing game player, each source vehicle attempts to minimize total path cost ($\mathcal{P}\mathcal{C}$). From the source vehicle \mathcal{V}_s to the destination vehicle \mathcal{V}_d , the total path cost ($\mathcal{P}\mathcal{C}_{\mathcal{V}_s^{\mathcal{V}_d}}^{\mathcal{V}_s}(\mathcal{H}_t)$) at time \mathcal{H}_t is computed as the sum of all relay link costs as follows:

$$\mathcal{P}\mathcal{C}_{\mathcal{V}_s^{\mathcal{V}_d}}^{\mathcal{V}_s}(\mathcal{H}_t) = \min \left\{ \sum_{\mathcal{L}\mathcal{C}_h^{\mathcal{V}_c}(\mathcal{H}_t) \in \text{Path}_{\mathcal{V}_s^{\mathcal{V}_d}}^{\mathcal{V}_s}} \mathcal{L}\mathcal{C}_h^{\mathcal{V}_c}(\mathcal{H}_t) \right\}, \quad (6)$$

where $\text{Path}_{\mathcal{V}_s^{\mathcal{V}_d}}^{\mathcal{V}_s}$ is the routing path from \mathcal{V}_s to \mathcal{V}_d . Usually, a vehicle acting as a relay node has to sacrifice its energy and bandwidth. Therefore, incentive payment algorithm should be developed to guide selfish vehicles toward the cooperative intervehicle routing paradigm [6, 26]. By paying the incentive cost to relaying vehicles, the developed routing algorithm stimulates cooperative actions among selfish relay vehicles. As a \mathbb{G}^{VRS} game player, a source vehicle pays its virtual money (Γ) to disseminate the routing packet. If the source vehicle \mathcal{V}_i selects the $\text{Path}_{\mathcal{V}_s^{\mathcal{V}_d}}^{\mathcal{V}_s}$ according to (6), its utility function at time \mathcal{H}_t ($U_{\mathcal{V}_i}(\text{Path}_{\mathcal{V}_s^{\mathcal{V}_d}}^{\mathcal{V}_s}, \mathcal{H}_t)$) can be defined as

$$U_{\mathcal{V}_i}(\text{Path}_{\mathcal{V}_s^{\mathcal{V}_d}}^{\mathcal{V}_s}, \mathcal{H}_t) = \begin{cases} \mathcal{Q}_{\mathcal{H}_t}^{\mathcal{V}_i} - (\mathcal{I} \times (\mathcal{P}\mathcal{C}_{\mathcal{V}_s^{\mathcal{V}_d}}^{\mathcal{V}_s}(\mathcal{H}_t))) \text{ and } \Gamma_{\mathcal{H}_{t+1}}^{\mathcal{V}_i} = \Gamma_{\mathcal{H}_t}^{\mathcal{V}_i} - (\mathcal{I} \times (\mathcal{P}\mathcal{C}_{\mathcal{V}_s^{\mathcal{V}_d}}^{\mathcal{V}_s}(\mathcal{H}_t))), & \text{if } \Gamma_{\mathcal{H}_t}^{\mathcal{V}_i} \geq (\mathcal{I} \times (\mathcal{P}\mathcal{C}_{\mathcal{V}_s^{\mathcal{V}_d}}^{\mathcal{V}_s}(\mathcal{H}_t))), \\ 0, & \text{otherwise (i.e. } \Gamma_{\mathcal{H}_t}^{\mathcal{V}_i} < (\mathcal{I} \times (\mathcal{P}\mathcal{C}_{\mathcal{V}_s^{\mathcal{V}_d}}^{\mathcal{V}_s}(\mathcal{H}_t))), \end{cases} \quad (7)$$

where $\mathcal{Q}_{\mathcal{H}_t}^{\mathcal{V}_i}$ and $\Gamma_{\mathcal{H}_t}^{\mathcal{V}_i}$ are the outcome of \mathcal{V}_i 's routing and the amount of \mathcal{V}_i 's virtual money at time \mathcal{H}_t , respectively. \mathcal{I} is the coefficient factor to estimate the incentive payment for relaying vehicles.

3.4. The VSS Algorithm Based on the Triple-Plane Bargaining Game. Recently, the VSS has attracted great interests and

becomes one of the most valuable features in the VANET system. Some VANET applications involve generation of huge amount of sensed data. With the advance of vehicular sensor technology, vehicles that are equipped with OBUs are expected to effectively monitor the physical world. In VANET infrastructures, RSUs request a number of sensing tasks while collecting the local information sensed by vehicles. According to the requested tasks, OBUs in vehicles

sense the requested local information and transmit their sensing results to the RSU. Although some excellent researches have been done on the VSS process, there are still significant challenges that need to be addressed. Most of all, conducting the sensing task and reporting the sensing data usually consume resource of vehicles. Therefore, selfish vehicles should be paid by the RSU as the compensation for their VSSs. To recruit an optimal set of vehicles to cover the monitoring area while ensuring vehicles to provide their full sensing efforts, the RSU stimulates sufficient vehicles with proper incentives to fulfill VSS tasks [6, 8, 27].

In this sense, we design our VSS algorithm to determine the incentive for vehicles to complete their VSS tasks. Since vehicles may make different contributions of sensing work, the RSU may issue appropriate incentives in return for the collected sensing data [6]. The major goal of our algorithm is that the interactive trading between the RSU and vehicles should benefit both of them. This game situation can be modeled effectively though \mathbb{G}^{VSS} . In the VSS algorithm, we consider the VSS platform consisting of one RSU (\mathcal{R}), a set of mobile vehicles (\mathcal{V}), and sensing tasks (\mathfrak{S}); they formulate the sensing game model \mathbb{G}^{VSS} . To capture its own heterogeneity, a vehicle's strategy ($\mu \in \mathbf{S}_{\mathcal{V}_i}^{\mathfrak{S}}$) indicates the contribution for a specific task ($\mathfrak{S} \in \mathfrak{S}$). For example, \mathcal{V}_i can actually contribute for the task \mathfrak{S} during \mathcal{H}_t , i.e., $\mu_{\mathcal{V}_i}^{\mathfrak{S}} = 1$, or not, i.e., $\mu_{\mathcal{V}_i}^{\mathfrak{S}} = 0$, where $\mu_{\mathcal{V}_i}^{\mathfrak{S}} \in \mathbf{S}_{\mathcal{V}_i}^{\mathfrak{S}}$. Each vehicle has its own utility function, which represents its net gain. The \mathcal{V}_i 's utility function at time \mathcal{H}_t ($u_{\mathcal{V}_i}^{\mathfrak{S}}(\mathbf{S}_{\mathcal{V}_i}^{\mathfrak{S}}, \mathcal{H}_t)$) is given by

$$\begin{aligned} u_{\mathcal{V}_i}^{\mathfrak{S}}(\mathbf{S}_{\mathcal{V}_i}^{\mathfrak{S}} = [\mu_{\mathcal{V}_i}^{\mathfrak{S}_1}, \dots, \mu_{\mathcal{V}_i}^{\mathfrak{S}_j}, \dots, \mu_{\mathcal{V}_i}^{\mathfrak{S}_s}], \mathcal{H}_t) \\ = \sum_{\mathfrak{S}_j \in \mathfrak{S}_1}^{\mathfrak{S}_s} \left(\mu_{\mathcal{V}_i}^{\mathfrak{S}_j}(\mathcal{H}_t) \times \left(\eta_{\mathcal{R}}^{\mathfrak{S}_j}(\mathcal{H}_t) - \mathcal{C}_{\mathcal{V}_i}^{\mathfrak{S}_j}(\mathcal{H}_t) \right) \right), \end{aligned} \quad (8)$$

$$\text{s.t. } \eta_{\mathcal{R}}^{\mathfrak{S}_j}(\mathcal{H}_t) \in \mathbf{S}^{\mathcal{R}},$$

$$\mu_{\mathcal{V}_i}^{\mathfrak{S}_j}(\mathcal{H}_t) = \begin{cases} 1, & \text{if } \mathcal{V}_i \text{ contributes for } \mathfrak{S}_j \text{ at } \mathcal{H}_t, \\ 0, & \text{otherwise,} \end{cases}$$

where $\mathcal{C}_{\mathcal{V}_i}^{\mathfrak{S}_j}(\mathcal{H}_t)$ and $\eta_{\mathcal{R}}^{\mathfrak{S}_j}(\mathcal{H}_t)$ are the \mathcal{V}_i 's cost and the RSU's incentive payment for the \mathfrak{S}_j sensing at time \mathcal{H}_t , respectively. Usually, game players simply attempt to maximize their utility functions. However, vehicles in the \mathbb{G}^{VSS} game should consider not only the VSS but also the VCS and VRS. From the point of view of vehicles, this situation can be modeled effectively though a cooperative bargaining process. In this study, we adopt a well-known bargaining solution concept, called Nash bargaining solution (NBS), to effectively design the vehicle's VSS strategy decision mechanism; the NBS is an effective tool to achieve a mutually desirable solution among conflicting requirements.

In our proposed scheme, vehicles can earn the virtual money (Γ) from the VSS incentive payment and spend their Γ for their VCS and VRS. Due to this reason, the strategy decision in the VSS algorithm might directly affect the payoff

of \mathbb{G}^{VCS} and \mathbb{G}^{VRS} . During VANET operations, the individual vehicle \mathcal{V}_i decides his VSS's strategy ($\mathbf{S}_{\mathcal{V}_i}^{\mathfrak{S}}$) to maximize the combined payoffs ($CP_{\mathcal{V}_i}(\mathbf{S}_{\mathcal{V}_i}^{\mathfrak{S}}, \mathcal{H}_t)$) at time \mathcal{H}_t :

$$\begin{aligned} \max CP_{\mathcal{V}_i}(\mathbf{S}_{\mathcal{V}_i}^{\mathfrak{S}}, \mathcal{H}_t) &= \max_{\mathbf{S}_{\mathcal{V}_i}^{\mathfrak{S}} = [\mu_{\mathcal{V}_i}^{\mathfrak{S}_1}, \dots, \mu_{\mathcal{V}_i}^{\mathfrak{S}_j}, \dots, \mu_{\mathcal{V}_i}^{\mathfrak{S}_s}]} \prod_{1 \leq j \leq 3} (\mathbf{x}_j - d_j)^{\psi_j}, \\ &\sum_{j=1}^3 \psi_j = 1, \\ \text{s.t. } \begin{cases} \mathbf{x}_1 = \cup_{\mathcal{V}_i \in \mathcal{V}} (\mathfrak{P}_j^{\mathcal{R}}(\mathcal{H}_t), r_k^{\mathcal{V}_i}(\mathcal{H}_t)), \\ \mathbf{x}_2 = U_{\mathcal{V}_i}(\text{Path}_{\mathcal{V}_i}^{\mathfrak{S}_j}, \mathcal{H}_t), \\ \mathbf{x}_3 = u_{\mathcal{V}_i}^{\mathfrak{S}}(\mathbf{S}_{\mathcal{V}_i}^{\mathfrak{S}} = [\mu_{\mathcal{V}_i}^{\mathfrak{S}_1}, \dots, \mu_{\mathcal{V}_i}^{\mathfrak{S}_j}, \dots, \mu_{\mathcal{V}_i}^{\mathfrak{S}_s}], \mathcal{H}_t), \end{cases} \end{aligned} \quad (9)$$

where d_j is a disagreement point and a disagreement point $d_{i, 1 \leq i \leq n}$ represents a minimum payoff of each game model. Therefore, d is the least guaranteed payoff for game players (i.e., zero in our model). $\psi_{j, 1 \leq j \leq 3}$ is a bargaining power, which is the relative ability to exert influence over the bargaining process. Usually, the bargaining solution is strongly dependent on the bargaining powers. In the \mathbb{G}^{VSS} game, $\psi_{j, 1 \leq j \leq 3}$ can be estimated as follows:

$$\begin{aligned} \psi_{j, 1 \leq j \leq 3} &= \frac{\mathfrak{Y}_j}{(\mathfrak{Y}_1 + \mathfrak{Y}_2 + \mathfrak{Y}_3)}, \\ \text{s.t. } \begin{cases} \mathfrak{Y}_1 = \max \left\{ \left(RI_{\mathcal{H}_t}^{\text{VCS}(\mathcal{V}_i)} - UT_{\mathcal{H}_t}^{\text{VCS}(\mathcal{V}_i)} \right), \Gamma_{\mathcal{H}_t}^{\mathcal{V}_i} \right\}, \\ \mathfrak{Y}_2 = \max \left\{ \left(RI_{\mathcal{H}_t}^{\text{VRS}(\mathcal{V}_i)} - UT_{\mathcal{H}_t}^{\text{VRS}(\mathcal{V}_i)} \right), \Gamma_{\mathcal{H}_t}^{\mathcal{V}_i} \right\}, \\ \mathfrak{Y}_3 = \Gamma_{\mathcal{H}_t}^{\mathcal{V}_i}, \end{cases} \end{aligned} \quad (10)$$

where $RI_{\mathcal{H}_t}^{\text{VCS}(\mathcal{V}_i)}$ and $RI_{\mathcal{H}_t}^{\text{VRS}(\mathcal{V}_i)}$ are the requested virtual money to maximize the \mathcal{V}_i 's payoffs in \mathbb{G}^{VCS} and \mathbb{G}^{VRS} at time \mathcal{H}_t , respectively. $UT_{\mathcal{H}_t}^{\text{VCS}(\mathcal{V}_i)}$, $UT_{\mathcal{H}_t}^{\text{VRS}(\mathcal{V}_i)}$ are the used virtual money by \mathcal{V}_i in \mathbb{G}^{VCS} and \mathbb{G}^{VRS} at time \mathcal{H}_t , respectively. Finally, individual vehicles select their strategies $\mathbf{S}^{\mathfrak{S}}$ according to (9) and (10). In a distributed online fashion, each vehicle ensures a well-balanced performance among the VCS, VRS, and VSS processes.

In the VSS algorithm, the vehicles' actual sensing cost $\mathcal{C}_{\mathcal{V}_i}^{\mathfrak{S}}(\cdot)$ and each individual vehicle's situation is usually unknown by the RSU. Under this asymmetric information situation, the RSU should learn the vehicle's circumstance during the VSS process. Like as the VCS process, each RSU is a *proposer* and individual vehicles are *responders*, and they also interact with each other for their objectives based on the collaborative feedback procedure. For the RSU, the payoff can be defined as a function of price levels for tasks and vehicles' responses. Therefore, the RSU's utility function ($u^{\mathcal{R}}(\mathbf{S}^{\mathcal{R}}, \mathcal{H}_t)$) at time \mathcal{H}_t is computed as the sum of each task's outcome:

$$\begin{aligned}
u^{\mathcal{R}} \left(\mathbf{S}^{\mathcal{R}} = \left[\eta_{\mathcal{R}}^{\mathfrak{S}_1}, \dots, \eta_{\mathcal{R}}^{\mathfrak{S}_j}, \dots, \eta_{\mathcal{R}}^{\mathfrak{S}_s} \right], \mathcal{H}_t \right) \\
&= \sum_{\mathfrak{S}_h = \mathfrak{S}_1}^{\mathfrak{S}_s} \left(\left(\mathfrak{Z}^{\mathfrak{S}_h}(\mathcal{H}_t) \times \Phi^{\mathfrak{S}_h} \right) \right. \\
&\quad \left. - \sum_{\mathcal{V}_i \in \mathcal{V}} \left(\mu_{\mathcal{V}_i}^{\mathfrak{S}_h}(\mathcal{H}_t) \times \eta_{\mathcal{R}}^{\mathfrak{S}_h}(\mathcal{H}_t) \right) \right), \\
\text{s.t. } \mathfrak{Z}^{\mathfrak{S}_h}(\mathcal{H}_t) &= \begin{cases} 1, & \text{if } \sum_{\mathcal{V}_i \in \mathcal{V}} \left(\mu_{\mathcal{V}_i}^{\mathfrak{S}_h}(\mathcal{H}_t) \right) \geq \mathcal{L}_{\mathcal{R}}^{\mathfrak{S}_h}, \\ 0, & \text{otherwise,} \end{cases}
\end{aligned} \tag{11}$$

where $\Phi^{\mathfrak{S}_h}$ is the profit of \mathfrak{S}_h ; when the VSS for \mathfrak{S}_h is successfully completed, i.e., $\sum_{\mathcal{V}_i \in \mathcal{V}} \left(\mu_{\mathcal{V}_i}^{\mathfrak{S}_h}(\mathcal{H}_t) \right) \geq \mathcal{L}_{\mathcal{R}}^{\mathfrak{S}_h}$, $\Phi^{\mathfrak{S}_h}$ is obtained. $\mathcal{L}_{\mathcal{R}}^{\mathfrak{S}_h}$ is the predefined VSS requirement to complete the task \mathfrak{S}_h .

As \mathbb{G}^{VSS} game players, the RSU considers the interrelationship among tasks and vehicles' reactions to determine its incentive payment strategy. To select the best strategy, a novel learning method is needed. In this study, we develop a new learning method to determine an effective payment policy for each task. For the RSU, the strategy vector $\mathbf{S}^{\mathcal{R}}(\mathcal{H}_t) = [\eta_{\mathcal{R}}^{\mathfrak{S}_1}(\mathcal{H}_t), \dots, \eta_{\mathcal{R}}^{\mathfrak{S}_s}(\mathcal{H}_t)]$ represents each price strategy for each task at time \mathcal{H}_t . Let $\mathfrak{Z}_{\mathcal{H}_t}^{\mathfrak{S}_s}(\mathfrak{P}_i^{\mathfrak{S}_s}, \mathfrak{n}_{\mathcal{R}}^{-\mathfrak{S}_s})$ be the learning value of taking strategy $\mathfrak{P}_i^{\mathfrak{S}_s} \in \eta_{\mathcal{R}}^{\mathfrak{S}_s}$ with a vector of payment strategies except the strategy $\mathfrak{P}_i^{\mathfrak{S}_s}(\mathfrak{n}_{\mathcal{R}}^{-\mathfrak{S}_s})$. The RSU updates its $\mathfrak{Z}(\cdot)$ value over time according to the following method:

$$\begin{aligned}
\mathfrak{Z}_{\mathcal{H}_{t+1}}^{\mathfrak{P}_i^{\mathfrak{S}_s}}(\mathfrak{P}_i^{\mathfrak{S}_s} \in \eta_{\mathcal{R}}^{\mathfrak{S}_s}, \mathfrak{n}_{\mathcal{R}}^{-\mathfrak{S}_s}) &= \left((1 - \beta) \times \mathfrak{Z}_{\mathcal{H}_t}^{\mathfrak{P}_i^{\mathfrak{S}_s}}(\mathfrak{P}_i^{\mathfrak{S}_s}, \mathfrak{n}_{\mathcal{R}}^{-\mathfrak{S}_s}) \right) \\
&\quad + \left(\beta \times \left(\left((1 - \gamma^{\mathcal{R}}) \times \partial_{\mathcal{H}_t}^{\mathfrak{S}_s} \right) \right. \right. \\
&\quad \left. \left. + \left\{ \gamma^{\mathcal{R}} \times \sum_{\mathfrak{S}_h \in \mathfrak{S}^{-\mathfrak{S}_s}} \frac{\partial_{\mathcal{H}_t}^{\mathfrak{S}_h}}{(s-1)} \right\} \right) \right), \\
\text{s.t. } \partial_{\mathcal{H}_t}^{\mathfrak{S}_s} &= \left(\left(\mathfrak{Z}^{\mathfrak{S}_s}(\mathcal{H}_t) \times \Phi^{\mathfrak{S}_s} \right) \right. \\
&\quad \left. - \sum_{\mathcal{V}_i \in \mathcal{V}} \left(\mu_{\mathcal{V}_i}^{\mathfrak{S}_s}(\mathcal{H}_t) \times \eta_{\mathcal{R}}^{\mathfrak{S}_s}(\mathcal{H}_t) \right) \right).
\end{aligned} \tag{12}$$

During the \mathbb{G}^{VSS} game process, the RSU adaptively learns the current VSS state and sequentially adjusts $\mathfrak{Z}(\cdot)$ values for each VSS task. Based on the $\mathfrak{Z}(\cdot)$ values, we can determine $\mathbf{P}_{\mathfrak{S}_s \in \mathfrak{S}}^{\mathcal{R}}(\mathcal{H}_t) = \{P_{\mathfrak{S}_s}^{\mathfrak{P}_{\min}^{\mathfrak{S}_s}}(\mathcal{H}_t), \dots, P_{\mathfrak{S}_s}^{\mathfrak{P}_i^{\mathfrak{S}_s}}(\mathcal{H}_t), \dots, P_{\mathfrak{S}_s}^{\mathfrak{P}_{\max}^{\mathfrak{S}_s}}(\mathcal{H}_t)\}$ as the probability distribution of \mathcal{R} 's strategy selection for the \mathfrak{S}_s at time \mathcal{H}_t . The $\mathfrak{P}_i^{\mathfrak{S}_s}$ strategy selection probability for \mathfrak{S}_s at time \mathcal{H}_{t+1} ($P_{\mathfrak{S}_s}^{\mathfrak{P}_i^{\mathfrak{S}_s}}(\mathcal{H}_{t+1})$) is defined as

$$P_{\mathfrak{S}_s}^{\mathfrak{P}_i^{\mathfrak{S}_s}}(\mathcal{H}_{t+1}) = \frac{\exp\left(\mathfrak{Z}_{\mathcal{H}_t}^{\mathfrak{P}_i^{\mathfrak{S}_s}}(\mathfrak{P}_i^{\mathfrak{S}_s} \in \eta_{\mathcal{R}}^{\mathfrak{S}_s}, \mathfrak{n}_{\mathcal{R}}^{-\mathfrak{S}_s})\right)}{\sum_{\mathfrak{P}_j^{\mathfrak{S}_s}, j=\min}^{\max} \exp\left(\mathfrak{Z}_{\mathcal{H}_t}^{\mathfrak{P}_j^{\mathfrak{S}_s}}(\mathfrak{P}_j^{\mathfrak{S}_s} \in \eta_{\mathcal{R}}^{\mathfrak{S}_s}, \mathfrak{n}_{\mathcal{R}}^{-\mathfrak{S}_s})\right)}. \tag{13}$$

3.5. Main Steps of Proposed Triple-Plane Bargaining Algorithm. In this study, we design a novel triple-plane bargaining game model through a systematic interactive game process. In the proposed VCS, VRS, and VSS algorithms, the RSUs and vehicles are game players to maximize their payoffs. In particular, the RSUs learn the current VANET situation based on the learning methods, and the vehicles determine their best strategies while balancing their VCS, VRS, and VSS requirements. The primary steps of the proposed scheme are described as follows.

Step 1. Control parameters are determined by the simulation scenario (Table 1).

Step 2. At the initial time, the \mathfrak{Z} and \mathbb{L} learning values in the RSUs are equally distributed. This starting estimation guarantees that each RSU's strategy benefits similarly at the beginning of the \mathbb{G}^{VCS} and \mathbb{G}^{VSS} games.

Step 3. During the \mathbb{G}^{VCS} game, the *proposer* RSU selects its strategy $\mathfrak{P}^{\mathcal{R}} \in \mathbb{S}^{\mathcal{R}}$ to maximize its payoff ($\mathbb{U}^{\mathcal{R}}$) according to (1), (3), and (4). As *responders*, vehicles select their strategy $\mathfrak{r} \in \mathbb{S}^{\mathcal{V}}$ to maximize their payoffs ($\mathbb{U}^{\mathcal{V}}$) according to (2) while considering their current virtual money (Γ).

Step 4. At every time step (\mathcal{H}), the RSU adjusts the learning values ($\mathbb{L}(\cdot)$) and the probability distribution ($\mathbb{P}^{\mathcal{R}}$) based on equations (3) and (4).

Step 5. During the \mathbb{G}^{VRS} game, individual vehicles estimate the wireless link states ($\mathcal{L}\mathcal{E}$) according to equation (5). At each time period, the $\mathcal{L}\mathcal{E}$ values are estimated online based on the vehicle's relative distance and speed.

Step 6. During the \mathbb{G}^{VRS} game, the source vehicle configures a multihop routing path using the Bellman-Ford algorithm based on equation (6). The source vehicle's payoff (U) is decided according to (7) while considering their current virtual money (Γ).

Step 7. During the \mathbb{G}^{VSS} game, the *proposer* RSU selects its strategy $\mathbf{S}^{\mathcal{R}}$ to maximize its payoff ($u^{\mathcal{R}}$) according to (11). As *responders*, vehicles select their strategy $\mathbf{S}^{\mathcal{V}}$ to maximize their combined payoff (CP) according to (9) while adjusting each bargaining power (ψ) based on equation (10).

Step 8. At every time step (\mathcal{H}), the RSU adjusts the learning values ($\mathfrak{Z}(\cdot)$) and the probability distribution ($\mathbf{P}^{\mathcal{R}}$) according to equations (12) and (13).

Step 9. Based on the interactive feedback process, the dynamics of our \mathbb{G}^{VCS} , \mathbb{G}^{VRS} , and \mathbb{G}^{VSS} games cause a cascade of interactions among the game players, who choose their best strategies in an online distributed manner.

Step 10. Under the dynamic VANET environment, the individual game players constantly self-monitor for the next game process; go to Step 3.

TABLE 1: System parameters used in the simulation experiments.

Parameter	Value	Description
$[\mathfrak{P}_{\min}^{\mathcal{R}}, \dots, \mathfrak{P}_j^{\mathcal{R}}, \dots, \mathfrak{P}_{\max}^{\mathcal{R}}]$	0.2, 0.4, 0.6, 0.8, 1	Predefined price levels for cloud service (min = 1, max = 5)
$[\mathfrak{r}_{\min}^{\mathcal{R}}, \dots, \mathfrak{r}_k^{\mathcal{R}}, \dots, \mathfrak{r}_{\max}^{\mathcal{R}}]$	1, 2, 3, 4, 5	The amount of cloud services in terms of BCSUs
$[\mathfrak{P}_{\min}^{\mathcal{S}}, \dots, \mathfrak{P}_l^{\mathcal{S}}, \dots, \mathfrak{P}_{\max}^{\mathcal{S}}]$	0.2, 0.4, 0.6, 0.8, 1	Predefined price levels for sensing service (min = 1, max = 5)
θ	0.2	A coefficient factor of cost calculation
\mathfrak{S}	0.8	A profit factor of vehicle if one BCSU is processed
α	0.5	The weight control factor to the distance and velocity
χ	0.2	A learning rate to update the \mathbb{L} values
\mathcal{Q}	5	The routing outcome at each time period
\mathcal{F}	0.1	A coefficient factor to estimate the incentive payment
$\{\mathcal{C}^{\mathcal{S}_1}, \mathcal{C}^{\mathcal{S}_2}, \mathcal{C}^{\mathcal{S}_3}, \mathcal{C}^{\mathcal{S}_4}\}$	0.1, 0.2, 0.3, 0.4	Predefined cost for each task sensing
$\{\Phi^{\mathcal{S}_1}, \Phi^{\mathcal{S}_2}, \Phi^{\mathcal{S}_3}, \Phi^{\mathcal{S}_4}\}$	10, 10, 10, 10	Predefined profit for each sensing task
$\mathcal{L}^1, \mathcal{L}^2, \mathcal{L}^3, \mathcal{L}^4$	5, 5, 5, 5	Predefined requirement for each sensing task
β	0.3	A control parameter to estimate the learning value \mathfrak{Z}
$\gamma^{\mathcal{R}}$	0.3	A discount factor to estimate the learning value \mathfrak{Z}

4. Performance Evaluation

4.1. Simulation Setup. In this section, we evaluate the performance of our proposed protocol and compare it with that of the MVIG, PVDD, and CRVC schemes [21–23]. To ensure a fair comparison, the following assumptions and system scenario were used:

- (i) The simulated system was assumed to be a TDMA packet system for VANETs.
- (ii) The number of vehicles that passed over a RSU was the rate of the Poisson process (ρ). The offered range was varied from 0 to 3.0.
- (iii) Fifty RSUs were distributed randomly over the 100 km road area, and the velocity of each mobile vehicle was randomly selected to be 36 km/h, 72 km/h, or 108 km/h.
- (iv) The maximum wireless coverage range of each vehicle was set to 500 m.
- (v) Cloud computation capacity (C) is 5 GHz, and one BCSU is the minimum amount (e.g., 20 MHz/s in our system) of the cloud service unit.
- (vi) The number (s) of sensing task in each \mathcal{R} is 4, i.e., $\mathfrak{S} = \{\mathfrak{S}_1, \mathfrak{S}_2, \mathfrak{S}_3, \mathfrak{S}_4\}$.
- (vii) The source and destination vehicles were randomly selected. Initially, virtual money (Γ) in each vehicle was set to 100.
- (viii) At the source node, data dissemination was generated at a rate of λ (packets/s). According to this assumption, the time duration \mathcal{H} in our simulation model is one second.
- (ix) Network performance measures obtained on the basis of 100 simulation runs are plotted as functions of the vehicle distribution (ρ).

To demonstrate the validity of our proposed method, we measured the cloud service success ratio, normalized dissemination throughput, and crowdsensing success probability. Table 1 shows the control factors and coefficients used in the simulation. Each parameter has its own characteristics [6].

4.2. Results and Discussion. Figure 1 compares the cloud service success ratio of each scheme. In this study, the cloud service success ratio represents the rate of cloud services that was completed successfully. This is a key performance evaluation factor in the VCS operation. As shown in Figure 1, the cloud service success ratios of all schemes are similar to each other; however, the proposed scheme adopts an interactive environmental feedback mechanism, and the RSUs in our scheme adaptively adjust their VCS costs. This approach can improve the VCS performance than the existing MVIG, PVDD, and CRVC schemes. Therefore, we outperformed the existing methods from low to high vehicle distribution intensities.

Figure 2 compares the normalized dissemination throughput in VANETs. Typically, the network throughput is measured as bits per second of network access. In this study, the dissemination throughput is defined as the ratio of data amount successfully received in destination vehicles to the total generated data amount in the source vehicles. The throughput improvement achieved by the proposed scheme is a result of our \mathbb{G}^{VRS} game paradigm. During the VRS operations, each vehicle in the proposed scheme can select the most efficient routing path with real-time adaptability and self-flexibility. Hence, we attained a higher dissemination throughput compared to other existing approaches that are designed as lopsided and one-way methods and do not effectively adapt to the dynamic and diversified VANET conditions.

The crowdsensing success probability, which is shown in Figure 3, represents the efficiency of the VANET system. In the proposed scheme, we employed the learning-based triple-plane game model to perform control decisions in a distributed online manner. According to the interactive operations of the VCS, VSS, and VRS, our game-based approach can improve the crowdsensing success probability more effectively than the other schemes. The simulation results shown in Figures 1 to 3 demonstrate that the proposed scheme can attain an appropriate performance balance; in contrast, the MVIG [21], PVDD [22], and CRVC [23] schemes cannot offer this outcome under widely different and diversified VANET situations.

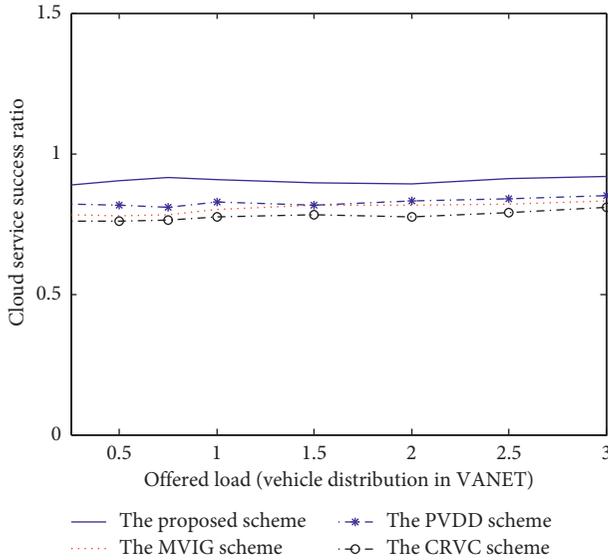


FIGURE 1: Cloud service success ratio. Y-axis: normalized cloud service success ratio. X-axis: offered numbers of vehicles that passed over a RSU; they are the rate of the Poisson process.

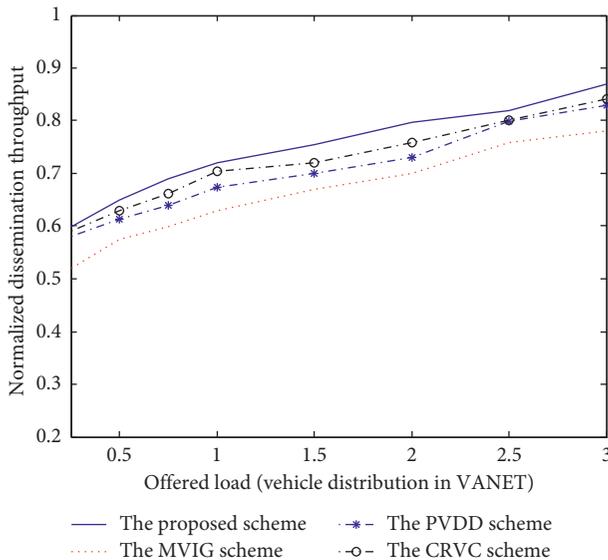


FIGURE 2: Normalized dissemination throughput. Y-axis: normalized dissemination throughput. X-axis: offered numbers of vehicles that passed over a RSU; they are the rate of the Poisson process.

5. Summary and Conclusions

The VANET using vehicle-based sensory technology is becoming more popular. It can provide vehicular sensing, routing, and clouding services for 5G network applications. Therefore, the design of next-generation VANET management schemes is important to satisfy the new demands. Herein, we focused on the paradigm of a learning algorithm and game theory to design the VCS, VRS, and VSS algorithms. By combining the VCS, VRS, and VSS algorithms, a new triple-plane bargaining game model was developed to

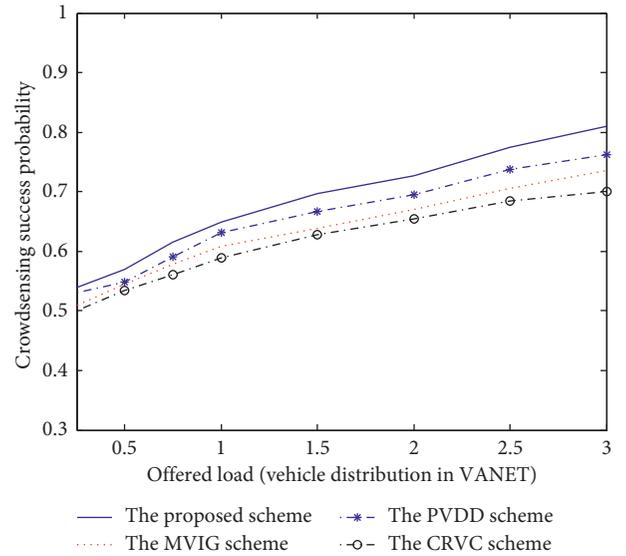


FIGURE 3: Crowdsensing success probability. Y-axis: normalized crowdsensing success probability. X-axis: offered numbers of vehicles that passed over a RSU; they are the rate of the Poisson process.

provide an appropriate performance balance. During the VANET operations, the RSUs learned their strategies better under dynamic VANET environments, and the vehicles considered the mutual-interaction relationships of their strategies. As game players, they considered the obtained information to adapt to the dynamics of the VANET environment and performed control decisions intelligently by self-adaptation. According to the unique features of VANETs, our joint design approach is suitable to provide satisfactory services under incomplete information environments. In the future, we would like to consider privacy issues such as the differential privacy during the VANET operation. Furthermore, we will investigate probabilistic algorithms to estimate the service quality of sensing, routing, and clouding services. In addition, we plan to investigate smart city applications, where the different sensory information of a given area can be combined to provide a complete view of the smart city development.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This research was supported by the MSIT (Ministry of Science and ICT), Korea, under the ITRC (Information Technology Research Center) support program (IITP-2018-2018-0-01799) supervised by the IITP (Institute for Information and Communications Technology Promotion)

grant funded by the Korea government (MSIT) (no. 2017-0-00498; A Development of Deidentification Technique Based on Differential Privacy).

References

- [1] Y. Wang, Y. Liu, J. Zhang, H. Ye, and Z. Tan, "Cooperative store-carry-forward scheme for intermittently connected vehicular networks," *IEEE Transactions on Vehicular Technology*, vol. 66, no. 1, pp. 777–784, 2017.
- [2] M. M. C. Morales, R. Haw, E.-J. Cho, C.-S. Hong, and S.-W. Lee, "An adaptable destination-based dissemination algorithm using a publish/subscribe model in vehicular networks," *Journal of Computing Science and Engineering*, vol. 6, no. 3, pp. 227–242, 2012.
- [3] Y. Kim, S. Atchley, G. R. Vallee, S. Lee, and G. M. Shipman, "Optimizing end-to-end big data transfers over terabits network infrastructure," *IEEE Transactions on Parallel and Distributed Systems*, vol. 28, no. 1, pp. 188–201, 2017.
- [4] J. Chen, G. Mao, C. Li, A. Zafar, and A. Y. Zomaya, "Throughput of infrastructure-based cooperative vehicular networks," *IEEE Transactions on Intelligent Transportation Systems*, vol. 18, no. 11, pp. 2964–2979, 2017.
- [5] 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, 2017.
- [6] S. Kim, "Effective crowdsensing and routing algorithms for next generation vehicular networks," *Wireless Networks*, 2017.
- [7] C. Wang, Z. Zhang, S. Lu, and M.C. Zhou, "Estimating travel speed via sparse vehicular crowdsensing data," in *Proceedings of IEEE World Forum on Internet of Things (WF-IoT)*, pp. 643–648, Reston, VA, USA, December 2016.
- [8] L. Xiao, T. Chen, C. Xie, H. Dai, and P. Vincent, "Mobile crowdsensing games in vehicular networks," *IEEE Transactions on Vehicular Technology*, vol. 67, no. 2, pp. 1535–1545, 2018.
- [9] B. Oh, N. Yongchan, J. Yang, S. Park, J. Nang, and J. Kim, "Genetic algorithm-based dynamic vehicle route search using car-to-car communication," *Advances in Electrical and Computer Engineering*, vol. 10, no. 4, pp. 81–86, 2011.
- [10] M. Chaqfeh, N. Mohamed, I. Jawhar, and J. Wu, "Vehicular cloud data collection for intelligent transportation systems," in *Proceedings of 2016 3rd Smart Cloud Networks and Systems (SCNS)*, pp. 1–6, Dubai, UAE, December 2016.
- [11] A. Ashok, S. Peter, and B. Fan, "Adaptive cloud offloading for vehicular applications," in *Proceedings of 2016 IEEE Vehicular Networking Conference (VNC)*, pp. 1–8, Columbus, OH, USA, December 2016.
- [12] J. Ahn, D. Shin, K. Kim, and J. Yang, "Indoor air quality analysis using deep learning with sensor data," *Sensors*, vol. 17, no. 11, pp. 1–13, 2017.
- [13] S. Kim, "Timed bargaining-based opportunistic routing model for dynamic vehicular ad hoc network," *EURASIP Journal on Wireless Communications and Networking*, vol. 2016, no. 14, pp. 1–10, 2016.
- [14] J.-H. Kim, K.-J. Lee, T.-H. Kim, and S.-B. Yang, "Effective routing schemes for double-layered peer-to-peer systems in MANET," *Journal of Computing Science and Engineering*, vol. 5, no. 1, pp. 19–31, 2011.
- [15] I. Jang, D. Pyeon, S. Kim, and H. Yoon, "A survey on communication protocols for wireless sensor networks," *Journal of Computing Science and Engineering*, vol. 7, no. 4, pp. 231–241, 2013.
- [16] N. Cheng, N. Zhang, N. Lu, X. Shen, J. W. Mark, and F. Liu, "Opportunistic spectrum access for CR-VANETs: a game-theoretic approach," *IEEE Transactions on Vehicular Technology*, vol. 63, no. 1, pp. 237–251, 2014.
- [17] T. Chen, L. Wu, F. Wu, and S. Zhong, "Stimulating cooperation in vehicular ad hoc networks: a coalitional game theoretic approach," *IEEE Transactions on Vehicular Technology*, vol. 60, no. 2, pp. 566–579, 2011.
- [18] D. B. Rawat, B. B. Bista, and G. Yan, "CoR-VANETs: game theoretic approach for channel and rate selection in cognitive radio VANETs," in *Proceedings of International Conference on Broadband, Wireless Computing, Communication and Applications*, pp. 94–99, Victoria, Canada, November 2012.
- [19] F. Hagenauer, C. Sommer, T. Higuchi, O. Altintas, and F. Dressler, "Vehicular micro clouds as virtual edge servers for efficient data collection," in *Proceedings of ACM International Workshop on Smart, Autonomous, and Connected Vehicular Systems and Services*, pp. 31–35, Snowbird, UT, USA, October 2017.
- [20] F. Hagenauer, C. Sommer, T. Higuchi, O. Altintas, and F. Dressler, "Vehicular micro cloud in action: on gateway selection and gateway handovers," *Ad Hoc Networks*, vol. 78, pp. 73–83, 2018.
- [21] M. Aloqaily, B. Kantarci, and H. T. Mouftah, "Multiagent/multiobjective interaction game system for service provisioning in vehicular cloud," *IEEE Access*, vol. 4, pp. 3153–3168, 2016.
- [22] R. Kim, H. Lim, and B. Krishnamachari, "Prefetching-based data dissemination in vehicular cloud systems," *IEEE Transactions on Vehicular Technology*, vol. 65, no. 1, pp. 292–306, 2016.
- [23] M. F. Feteiha and H. S. Hassanein, "Enabling cooperative relaying VANET clouds over LTE-A networks," *IEEE Transactions on Vehicular Technology*, vol. 64, no. 4, pp. 1468–1479, 2015.
- [24] K. Zhang, Y. Mao, S. Leng, Y. He, and Y. Zhang, "Mobile-edge computing for vehicular networks: a promising network paradigm with predictive off-loading," *IEEE Vehicular Technology Magazine*, vol. 12, no. 2, pp. 36–44, 2017.
- [25] C. T. Hieu and C.-S. Hong, "A connection entropy-based multi-rate routing protocol for mobile ad hoc networks," *Journal of Computing Science and Engineering*, vol. 4, no. 3, pp. 225–239, 2010.
- [26] S. Kim, "Adaptive ad-hoc network routing scheme by using incentive-based model," *Ad Hoc and Sensor Wireless Networks*, vol. 15, pp. 1–19, 2012.
- [27] K. Han, C. Chen, Q. Zhao, and X. Guan, "Trajectory-based node selection scheme in vehicular crowdsensing," in *Proceedings of IEEE/CIC International Conference on Communications in China*, pp. 1–6, Shenzhen, China, November 2015.



Hindawi

Submit your manuscripts at
www.hindawi.com

