Optimal Delay Constrained Offloading for Vehicular Edge Computing Networks

Ke Zhang, Yuming Mao and Supeng Leng
School of Communication and Information Engineering,
University of Electronic Science and Technology of China
Email: {zhangke,ymmao,spleng}@uestc.edu.cn

Sabita Maharjan
Simula Research Laboratory
Norway
Email: sabita@simula.no

Yan Zhang
University of Oslo
Norway
Email: yanzhang@ieee.org

***Abstract*—The increasing number of smart vehicles and their resource hungry applications pose new challenges in terms of computation and processing for providing reliable and efficient vehicular services. Mobile Edge Computing (MEC) is a new paradigm with potential to improve vehicular services through computation offloading in close proximity to mobile vehicles. However, in the road with dense traffic flow, the computation limitation of these MEC servers may endanger the quality of offloading service. To address the problem, we propose a hierarchical cloud-based Vehicular Edge Computing (VEC) offloading framework, where a backup computing server in the neighborhood is introduced to make up for the deficit computing resources of MEC servers. Based on this framework, we adopt a Stackelberg game theoretic approach to design an optimal multilevel offloading scheme, which maximizes the utilities of both the vehicles and the computing servers. Furthermore, to obtain the optimal offloading strategies, we present an iterative distributed algorithm and prove its convergence. Numerical results indicate that our proposed scheme greatly enhances the utility of the offloading service providers.**

I. Introduction

The advancements in Internet of Things (IoT) and wireless technologies help make vehicles smarter and support them to provide better services, such as autonomous driving and natural language processing [1]. Along with the advancements and such new applications come challenges associated with computing resource requirements on the already resource-constrained vehicles. Ongoing attempts to cope with the explosive application demands on vehicular terminals involves offloading the computation tasks to remote servers through cloud-based vehicular networks [2].

Although mobile cloud computing improves both resource utilization and computation performance, the long distance transmission of the task files between remote cloud servers and the mobile vehicles may bring considerable overhead [3]. Furthermore, the delay fluctuation of transmission may significantly degrade the offloading efficiency. To overcome this problem, Mobile Edge Computing (MEC) is envisioned as a promising solution, where the computing resources are pushed to the radio access network and offloading service is provided in the proximity of the vehicles.

This work is supported by the National Natural Science Foundation of China under Grant No. 61374189, the 111 project (B14039), the projects 240079/F20 funded by the Research Council of Norway, and the project IoTSec - Security in IoT for Smart Grids with number 248113/O70 part of the IKTPLUSS program funded by the Norwegian Research Council.

Corresponding author: Supeng Leng (e-mail: spleng@uestc.edu.cn)

Due to proximity, MEC servers are able to provide fast interactive response in the computation offloading process, and enrich users' experience for the delay-sensitive applications [4]. However, MEC servers always have finite computing resources, which makes the servers unable to fully meet the offloading requirements from the vehicles within the specified delay constraints, especially for dense traffic flow. Addressing this challenge demands new MEC offloading schemes that can fulfill the offloading requirements as well as satisfy the QoS requirements through dynamic resource assignment.

Cloud-enabled vehicular networks have been studied recently. In [5], the authors proposed a multiagent/multiobjective interaction game system to manage on-demand service provision in a vehicular cloud network. With the aid of vehicular cloud, the authors in [6] proposed a self-adaptive interactive navigation tool, which brings the navigation paths of the vehicles to global road traffic optimization. In [7], the authors investigated resource management in vehicular cloud, and demonstrated the benefits of reinforcement-learning-based techniques for resource provisioning.

A few work have been carried out focusing on the offloading schemes. In [8], the authors studied the offloading problem among multiple devices for mobile-edge cloud computing, and proposed a game theoretic approach for achieving the efficient offloading mechanisms. To minimize the energy consumption of the MEC offloading system, the authors in [9] proposed an energy-efficient computation offloading scheme, which jointly optimizes the offloading decisions and the radio resource allocation strategies. In the studies of MEC in vehicular networks, the authors in [10] proposed a contract-based offloading and computing resource allocation scheme, which maximizes the benefit of the MEC service providers while enhancing the utilities of the vehicular terminals. In [11], the combination of vehicular delay-tolerant networks and MEC paradigm is utilized to manage the handoff and the processing of large data sets in smart grid environment.

In all aforementioned studies, the MEC servers are always envisaged as independent computing resources, where the resource sharing between the servers has been ignored. Furthermore, the incentive-based approaches for improving the offloading efficiency have not been studied in these studies. In

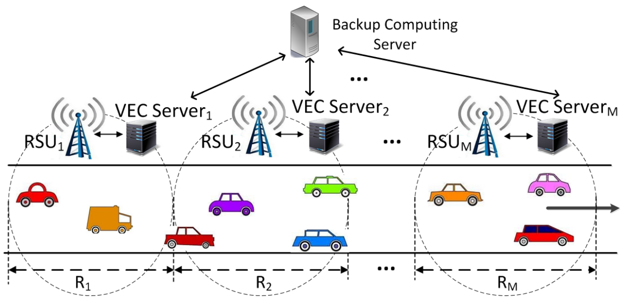

Fig. 1. The VEC offloading in a hierarchical cloud-based vehicular network.

this paper, we propose a hierarchical cloud-based Vehicular Edge Computing (VEC) offloading framework, where the computing resources of the VEC servers are dynamically assigned to the vehicles through an incentive way. The main contributions of this paper are listed below:

- We propose a new VEC offloading scheme, where the sharing of the backup server resources between the VEC servers is considered.
- We formulate the vehicular computation offloading process as a Stackelberg game, and introduce an incentive mechanism for the vehicles in the offloading server choosing and computing resource assignment.
- We develop an efficient distributed algorithm to reach the optimal computation offloading, where the revenue of the service providers is maximized while the delay constraints of the tasks are satisfied.

The rest of the paper is organized as follows. The system model is described in Section II. In Section III, we model the computation offloading between vehicles and VEC servers with a Stackelberg game approach. In Section IV, we prove the existence of the Stackelberg equilibrium, and propose an optimal offloading scheme. We present numerical results in Section V and conclude the paper in Section VI.

II. System Model

Fig. 1 shows our proposed VEC offloading scenario. We consider M Road Side Units (RSUs) located along a unidirectional road, whose id set is denoted as $\mathcal{M} = \{1, 2, ..., M\}$. Due to the different radio power of each RSU and the variety of the wireless environment, the size of the wireless coverage areas of these RSUs may be different [12]. Each RSU is equipped with a VEC server. The id of each VEC server is the same as that of the RSU containing it. The computation capacity of the VEC servers is $\{f_1^{max}, f_2^{max}, ..., f_M^{max}\}$, respectively.

A Backup Computing Server (BCS) that acts as a computing resource pool, is placed along the road. When the computing resources of a VEC server are not adequate to meet the vehicles' demands, the VEC server can further offload parts of the computation tasks to the BCS. The BCS sells resource units to the VEC servers with unit price y. The communication between the RSUs and the BCS is through a high bandwidth wired connection. The time delay of the wired communication is negligible compared to contemporary wireless technologies.

There are N vehicles arriving at the starting point of the road. All the vehicles are running at a constant speed v. Each vehicle has a computation task, which can be described in three terms as $T_i = \{d_i, b_i, t_i^{max}\}$, $i \in \mathcal{N} = \{1, 2, ..., N\}$. Here, d_i is the size of the input data for computation, b_i is the amount of the computing resources required to accomplish task T_i, and t_i^{max} is the maximum time delay constraint of T_i.

For each vehicle, its computation task can be accomplished either locally on its own computing resources or remotely on a VEC server through task offloading. To offload their tasks to the VEC servers, the vehicles should transmit their computation files to the VEC servers through the RSUs. We consider that each vehicle accesses the RSU with the strongest wireless signal. Considering various wireless coverage area of the M RSUs, the road can be divided into M segments, whose length is denoted as $\{R_1, R_2, ..., R_M\}$, respectively. The vehicles running within the mth segment only accesses the RSU located on the same segment.

In the system, we consider $\{T_i, i \in \mathcal{N}\}$ are delay sensitive tasks, such as interactive gaming and autonomous driving. Thus, the utility gained by each vehicle focuses on the delay reduction of the task accomplishment through offloading.

Let q_i be the amount of the computing resource owned by vehicle i. When vehicle i chooses to accomplish task T_i locally, the computation execution time is $t_{i,0} = b_i/q_i$. If vehicle i chooses to offload its task to VEC server k, the time cost can be divided into three parts. The first one is the time taken for vehicle i to reach VEC server k. As VEC server k is equipped on RSU k, this part of time equals the time cost for vehicle i arriving road segment k. The second part is the task file transmission time. The last one is the execution time of this task on the VEC server. Then, the total time cost for the task offloading from vehicle i to VEC server k is given as

$$t_{i,k} = \sum_{j=1}^{k-1} R_j/v + d_i/r_k + b_i/f_{i,k}, \tag{1}$$

where r_k is the data transmission rate for a vehicle accessing RSU k, and $f_{i,k}$ is the amount of the resources allocated from VEC server k to vehicle i for offloading the task.

During the vehicles running through the road, we consider that each vehicle at most can offload its task to one VEC server. Let $a_{i,k} = 1$ denote that vehicle i chooses to offload the task to VEC server k, and $a_{i,k} = 0$ otherwise. Thus, we have $\sum_{k=1}^{M} a_{i,k} \leq 1$, $i \in \mathcal{N}$. Considering both the VEC servers and the BCS are always operated and maintained by some operators, to gain revenues from providing computing services, the operators employ a pricing scheme such that the vehicles are charged according to their required computation resources. Furthermore, we consider a linear cost function for the servers providing the resources. The cost for server k providing z units of resources for computation offloading is formulated as

$$c_k(z) = \alpha_k z + \beta_k, \quad k \in \mathcal{M}, \tag{2}$$

where $\alpha_k > 0$ and β_k are the coefficients of the cost function of server k. For ease of analysis, we consider $\beta_k = 0$.

III. Modeling of VEC Offloading System: A Stackelberg Game Approach

Due to the computing resource limitation of the VEC servers, competitions may be incurred among the vehicles in choosing the offloading servers. As the offloading strategies of the vehicles are motivated by the pricing scheme, these vehicles are indirectly coupled through the resource price. Furthermore, the resource providers, i.e., the VEC servers, are also indirectly coupled through this price in the offloading process.

The vehicles only can offload tasks to the VEC servers, and the offloading decisions of the vehicles are response to the prices advertised by the VEC servers. Thus, a Stackelberg game is an appealing approach to model the multilevel offloading system [13]. In this game, VEC servers are the leaders, and the vehicles act as the followers by optimally reacting to the VEC servers' strategies. In this section, we define utilities for the VEC servers and the vehicles, and model the offloading system as a Stackelberg game.

A. Vehicle Side Analysis

The utility of vehicle i offloading its computation task to VEC server k is defined as

$$U_i = \sum_{k=1}^{M} a_{i,k}(\lambda_i(t_{i,0} - t_{i,k}) - x_k f_{i,k}), \tag{3}$$

where λ_i is a vehicle-specific parameter that shows vehicle i's sensitivity to the reduction of the task execution time, $\lambda_i > 0$. x_k is the price charged for a vehicle using a unit computing resource from VEC server k.

Since the vehicles are rational, they maximize their utilities through choosing the offloading target VEC servers. In the case where the price set $\{x_k, k \in \mathcal{M}\}$ is given, the optimization problem for vehicle i is

$$\begin{array}{ll} \underset{\{a_{i,k}, f_{i,k}, i\in\mathcal{N}, k\in\mathcal{M}\}}{max} & U_i \\ s.t. \quad \mathrm{C1}: & \sum_{k=1}^{M} a_{i,k} t_{i,k} \le t_i^{\max}, \quad i \in \mathcal{N}, \\ \qquad \mathrm{C2}: & \sum_{i=1}^{N} a_{i,k} f_{i,k} \le f_k^{max} + f_k^e, \quad k \in \mathcal{M}, \\ \qquad \mathrm{C3}: & a_{i,k} = \{0,1\}, \quad i \in \mathcal{N},\ k \in \mathcal{M}, \\ \qquad \mathrm{C4}: & \sum_{k=1}^{M} a_{i,k} \le 1, \quad i \in \mathcal{N}, \end{array} \tag{4}$$

where f_k^e is the amount of computing resource bought from the BCS by VEC server k.

The choice of the VEC servers by the vehicles not only depends on their own offloading demands, but also on the offloading strategies of other vehicles. Thus, there is coupling between the offloading decisions of these vehicles, which makes non-cooperative game an appropriate tool to model the decision process in this context. The players of this game are the vehicles $\{\mathcal{N}\}$. The strategy set of vehicle i can be denoted as $\boldsymbol{s}_i = \{s_{i,1}, s_{i,2}, ..., s_{i,M}\}$, where $s_{i,k} = a_{i,k} f_{i,k}$, $k \in \mathcal{M}$. Thus, the strategy space of the players can be given as $\boldsymbol{s} = \{\boldsymbol{s}_1 \times \boldsymbol{s}_2 \times ... \times \boldsymbol{s}_N\}$. Given $\boldsymbol{s}_{-i}$ as the strategies of all vehicles other than vehicle i, the payoff of vehicle i is denoted as $U_i(\boldsymbol{s}_i, \boldsymbol{s}_{-i})$, $i \in \mathcal{N}$.

Theorem 1. *The task offloading game between the vehicles is a concave multi-player game, a Nash Equilibrium (NE) exists for the game.*

Proof. In the case where vehicle i chooses to offload its task to VEC server k, as $f_{i,k} = [0, f_k^{max} + f_k^e]$, we have $s_{i,k} = [0, f_k^{max} + f_k^e]$, $i \in \mathcal{N}, k \in \mathcal{M}$. Replacing $f_{i,k}$ by $s_{i,k}$ in (3), we obtain $\partial^2 U_i / \partial {s_{i,k}}^2 = -\lambda_i b_i / s_{i,k}^3 < 0$. The payoff function $U_i(\boldsymbol{s}_i, \boldsymbol{s}_{-i})$ is strictly convex in terms of variable $s_{i,k}$ for the given offloading strategies of the vehicles other than vehicle i. This property holds for all the vehicles choosing to offload their tasks to any VEC servers. Thus, the task offloading game between the vehicles is a strictly concave multi-player game, which has a NE [14]. □

B. VEC Server Side Analysis

Being offloading service providers, the VEC servers aim to make more profit by selling computing resource to the vehicles. Since each vehicle can choose any VEC server as its offloading target under the specified delay constraint for the task, the servers play a non-cooperative price determination game with each other to decide their optimal resource price. Thus, there exists competition between the servers during the offloading process. Furthermore, each VEC server is able to buy computing resource from the BCS when its own resources cannot meet the requirements from the vehicles. As the total available computing resources affect the revenue of each VEC server, the amount of the resources bought from the BCS is also a strategy of each VEC server in the offloading competition. We consider the competition is imperfect, and each server determines its strategy set (x_k, f_k^e) based on its own available resource as well as the offloading demand, $k \in \mathcal{M}$.

As the computing resource from the backup source is normally more costly than the resource from the VEC servers themselves, we set $y > \alpha_k$, $k \in \mathcal{M}$. Thus, each VEC server prefers to utilize its own available resources first. Given the strategy sets of the other VEC servers, the revenue of VEC server k adopting strategy (x_k, f_k^e) is given as

$$\begin{aligned} & U_{\mathrm{VEC}}^{K}((x_k, f_k^e), (\boldsymbol{x_{-k}}, \boldsymbol{f^e_{-k}})) \\ & = x_k \sum_{i=1}^{N} f_{i,k} - \alpha_k \min(\sum_{i=1}^{N} f_{i,k}, f_k^{\max}) - y f_k^e, \end{aligned} \tag{5}$$

where $(\boldsymbol{x_{-k}}, \boldsymbol{f^e_{-k}})$ is the strategy sets of the VEC servers other than server k. Then, the revenue optimization problem for VEC server k can be formulated as

$$\begin{array}{ll} \max\limits_{x_k, f_k^e} & U_{\mathrm{VEC}}^{K}((x_k, f_k^e), (\boldsymbol{x_{-k}}, \boldsymbol{f^e_{-k}})) \\ s.t. \quad \mathrm{C1}: & x_k > 0,\ k \in \mathcal{M}, \\ \qquad \mathrm{C2}: & f_k^e \ge 0,\ k \in \mathcal{M}. \end{array} \tag{6}$$

Lemma 1. *For the computing resource selling price of each VEC server, there is an upper limit, i.e.,* $x_k \le x_k^{\max}$, $k \in \mathcal{M}$.

Proof. Let $C_{i,k} = t_{i,0} - \sum_{j=1}^{k-1} R_j / v - d_i / r_k$. According to (1) and (3), we get the utility of vehicle i when it offloads the task to VEC server k as $U_{i,k} = \lambda_i (C_{i,k} - b_i / f_{i,k}) - x_k f_{i,k}$.

Since the vehicles are rational, vehicle i may choose to offload its task to VEC server k only when $U_{i,k} > 0$. Thus, we have $x_k < \lambda_i(C_{i,k} - b_i/f_{i,k})/f_{i,k}$. Let $Q(f_{i,k}) = \lambda_i(C_{i,k} - b_i/f_{i,k})/f_{i,k}$. To prove the existence of x_k^{MAX}, we need to prove there is a maximum value of $Q(f_{i,k})$. The first-order derivative of $Q(f_{i,k})$ with respect to $f_{i,k}$ is $\partial Q(f_{i,k})/\partial f_{i,k} = 2\lambda_i b_i/f_{i,k}^3 - \lambda_i C_{i,k}/f_{i,k}^2$. We can see that $\partial Q(f_{i,k})/\partial f_{i,k} > 0$ when $f_{i,k} < 2b_i/C_{i,k}$. Thus, there exists a maximum value $Q_{i,k}^{max}$ of function $Q(f_{i,k})$. Considering each VEC server can offload the tasks from any vehicles under the delay constraints, the sufficient condition for VEC server k that is able to sell its computing resource to a vehicle is $x_k < x_k^{\max} = \max(Q_{1,k}^{\max}, Q_{2,k}^{\max}, ..., Q_{N,k}^{\max})$, $k \in \mathcal{M}$. Hence there is an upper limit of the resource price of each VEC server. □

Lemma 2. *For the amount of the computing resource purchased from the BCS by each VEC server, there exists an upper limit, i.e.,* $f_k^e \leq f_{k,\max}^e$, $k \in \mathcal{M}$.

Proof. According to (5), if the resource demand on VEC server k is less than its supply capability, i.e., $\sum_{i=1}^N f_{i,k} \leq f_k^{\max}$, in order to maximize its revenue, VEC server k should not purchase any resource from the BCS. On the contrary, when $\sum_{i=1}^N f_{i,k} > f_k^{\max}$, (5) can be written as $U_{\text{VEC}}^K((x_k, f_k^e), (\boldsymbol{x_{-k}}, \boldsymbol{f_{-k}^e})) = x_k \sum_{i=1}^N f_{i,k} - \alpha_k f_k^{\max} - y f_k^e$. As each VEC server is rational, it is required that $U_{\text{VEC}}^K((x_k, f_k^e), (\boldsymbol{x_{-k}}, \boldsymbol{f_{-k}^e})) \geq 0$. Thus, we have $f_k^e \leq (x_k \sum_{i=1}^N f_{i,k} - \alpha_k f_k^{\max})/y$. Due to the maximum selling price $x_k^{\max}$ proved in Lemma 1, we get $f_k^e \leq f_{k,\max}^e = (x_k^{\max} \sum_{i=1}^N f_{i,k} - \alpha_k f_k^{\max})/y$. □

Theorem 2. *A Nash equilibrium exists in the game of resource price determination and backup resource purchase between the VEC servers.*

Proof. For VEC server k, $k \in \mathcal{M}$, according to Lemma 1 and Lemma 2, its price strategy $x_k \in (0, x_k^{\max}]$ and the backup resource purchase strategy $f_k^e \in [0, f_{k,\max}^e]$. Thus, the spaces of the VEC servers' strategy sets $\{(x_k, f_k^e),\ k \in \mathcal{M}\}$ are nonempty, convex and compact. Further more, from (5), we can find that the revenue function of VEC server k is continuous and quasi-concave in terms of x_k and f_k^e. Thus the game possesses a Nash equilibrium [15]. □

IV. Stackelberg Equilibrium and Distributed Algorithm

In this section, we first prove the existence of Stackelberg Equilibrium (SE) of the game. Then, we propose an efficient distributed algorithm to obtain the optimal offloading strategies for the VEC servers.

A. Stackelberg Equilibrium

In the Stackelberg game of the vehicular task offloading process, the VEC servers are the leaders while the vehicles are the followers. The equilibrium strategies for the vehicles in this game are their optimal response for strategies announced by the VEC servers. Given strategy sets $\{(x_k, f_k^e),\ k \in \mathcal{M}\}$ of the VEC servers, according to (4), the equilibrium strategy $(a_{i,k}^*, f_{i,k}^*)$ of vehicle i should satisfy the following condition

$$U_i(s_{i,k}^*, \boldsymbol{s_{i,-k}^*}) \geq U_i(s_{i,k}, \boldsymbol{s_{i,-k}^*}), \quad i \in \mathcal{N}. \tag{7}$$

Similarly, the equilibrium strategies of the VEC server are their optimal strategies based on the known response of the vehicles. The strategy set (x_k^*, f_k^{e*}) is an equilibrium strategy of VEC server k, if

$$\begin{aligned} &U_{\text{VEC}}^K((x_k^*, f_k^{e*}), (\boldsymbol{x_{-k}^*}, \boldsymbol{f_{-k}^{e*}}), \boldsymbol{s_k}(x_k^*, f_k^{e*}; \boldsymbol{x_{-k}^*}, \boldsymbol{f_{-k}^{e*}})) \geq \\ &U_{\text{VEC}}^K((x_k, f_k^e), (\boldsymbol{x_{-k}^*}, \boldsymbol{f_{-k}^{e*}}), \boldsymbol{s_k}(x_k, f_k^e; \boldsymbol{x_{-k}^*}, \boldsymbol{f_{-k}^{e*}})),\ k \in \mathcal{M}. \end{aligned} \tag{8}$$

According to Theorem 1 and Theorem 2, the non-cooperative game between vehicles and the game between VEC servers have NE, respectively. Therefore, there exists a SE of the Stackelberg game [16].

B. Distributed Algorithm

As the VEC servers in the vehicular task offloading system may belong to different operators, it is impractical to manage these servers in a centralized manner. Thus, we propose a distributed algorithm for the VEC servers to select their optimal offloading strategies.

In this algorithm, each VEC server starts by randomly selecting both its resource selling price and the amount of resources to purchase from the BCS. Considering the VEC servers are rational, the price set by VEC server k is not less than the cost of the resource. Furthermore, according to Lemma 1, there is an upper limit $x_k^{\max}$ of the price. Thus, the randomly selected price of VEC server k should be in the interval $[\alpha_k, x_k^{\max}]$, $k \in \mathcal{M}$. Based on Lemma 2, the randomly selected amount of the purchased resources belongs to $[0, f_{k,\max}^e]$.

Following the announced strategies of the VEC servers, the vehicles are selected in a random sequence. Each selected vehicle determines its offloading target server and the amount of the resources needed in its task offloading by solving (4).

Based on the response of the vehicles, each VEC server firstly adjusts its resource selling price x_k, $k \in \mathcal{M}$. After this adjustment, if the demands on server k still exceed its available resource, it decides to purchase more resources from the BCS. Otherwise, server k decreases f_k^e until f_k^e reaches zero. The vehicles make their response to the strategy changes of the VEC servers. The strategies of the VEC servers are updated iteratively, until there is no change of the strategies compared to the previous iteration.

To improve the efficiency of the distributed algorithm, some impractical strategies of the VEC servers can be removed from the iterations, which are illustrated in the following Theorem.

Theorem 3. *In the case where the selling price of VEC server* k *satisfies* $\alpha_k \leq x_k \leq y$, *if the total resource demand from the vehicles on server* k *exceeds or equals its resource capacity limit* $f_k^{\max}$, *further reducing the price decreases the revenue of server* k, $k \in \mathcal{M}$.

Proof. As $\sum_{i=1}^{N} f_{i,k} \geq f_k^{\max}$, according to (5), we have $U_{\mathrm{VEC}}^{K}(x_k) = x_k \sum_{i=1}^{N} f_{i,k} - \alpha_k f_k^{\max} - y(\sum_{i=1}^{N} f_{i,k} - f_k^{\max})$. Let $U_{\mathrm{VEC}}^{K'}(x_k - \delta)$ denote the revenue of VEC server k that lowers its selling price with the reduction δ. Considering the rational characteristic of the vehicles, the decrease of the price attracts more vehicles offloading their tasks to VEC server k. Let σ be the increase of the computing resource requirements from the vehicles. We get $U_{\mathrm{VEC}}^{K'}(x_k - \delta) = (x_k - \delta)(\sum_{i=1}^{N} f_{i,k} + \sigma) - \alpha_k f_k^{\max} - y(\sum_{i=1}^{N} f_{i,k} + \sigma - f_k^{\max})$. The difference between the two revenues is $U_{\mathrm{VEC}}^{K'}(x_k - \delta) - U_{\mathrm{VEC}}^{K}(x_k) = (x_k - y)\sigma - \delta(\sum_{i=1}^{N} f_{i,k} + \sigma)$. Since $x_k \leq y$, we have $U_{\mathrm{VEC}}^{K'}(x_k - \delta) < U_{\mathrm{VEC}}^{K}(x_k)$. Thus, the decrease of price x_k reduces the revenue of server k. □

The details of the proposed distributed algorithm are illustrated in Algorithm 1.

Algorithm 1 Distributed algorithm for obtaining VEC servers' optimal offloading strategies

Initialization: The vehicles with computation tasks $\{T_i, i \in \mathcal{N}\}$; The computation resource capabilities $\{f_k^{\max}, k \in \mathcal{M}\}$ for the VEC servers; The BCS's resource price y.

```
VEC server k arbitrarily chooses its strategy set (x_k, f_k^e),
  where α_k ≤ x_k ≤ x_k^max, 0 ≤ f_k^e ≤ f_{k,max}^e, and k ∈ M;
Compute the demand response of the vehicles;
Loop
  For VEC server k, k ∈ M do
    if (Σ_{i=1}^N f_{i,k} ≥ f_k^max) & (x_k ≤ y) then
      Obtain the optimal price x'_k with increase strategy
        using (6);
    else
      Get the optimal x'_k either with increase or decrease
        strategies;
    end if
    Update the vehicles' resource demands;
    if (Σ_{i=1}^N f_{i,k} ≤ f_k^max & f_k^e > 0) || (Σ_{i=1}^N f_{i,k} >
      f_k^max & Σ_{i=1}^N f_{i,k} ≠ f_k^max + f_k^e) then
        f_k^{e'} = max(0, Σ_{i=1}^N f_{i,k} − f_k^max);
    end if
  End For
  if ∀ (x'_k, f_k^{e'}) == (x_k, f_k^e), k ∈ M then
    End loop;
  end if
End loop
```

V. Numerical Results

In this section, we evaluate the proposed hierarchical VEC offloading scheme. We consider a scenario where 5 VEC servers randomly locate in a 100-meter road. The computing resources of the VEC servers from the beginning of the road to the other end are $\{1415, 641, 985, 1070, 1228\}$, respectively. There are 120 arriving vehicles on the road, and they run at the speed 120 km/hr. The computing resource units required by the vehicles and the delay constraints of the tasks are randomly distributed on the intervals $(20, 70)$ and $(2, 5)$, respectively.

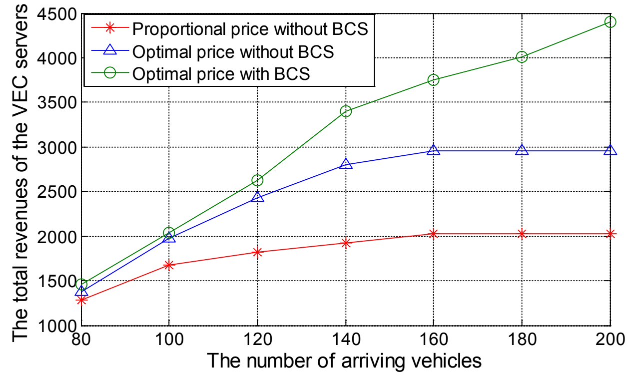

Fig. 2. The total revenues of the VEC servers with different offloading schemes.

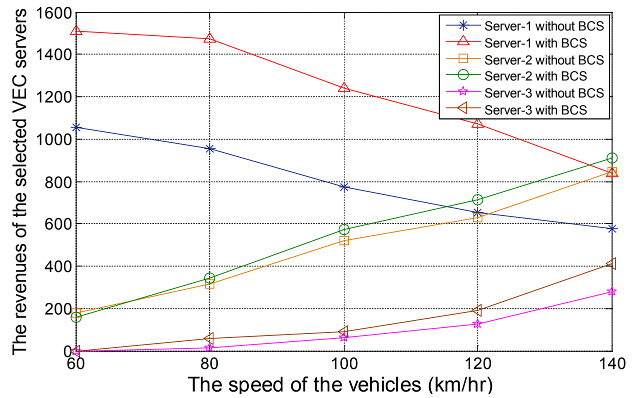

Fig. 3. The revenues of the selected VEC servers at different locations.

Fig. 2 evaluates the performance of the VEC offloading system with different computation offloading schemes. In this figure, the offloading system adopting the proportional price scheme has the lowest revenue. Since the price in this scheme is proportional to the VEC servers' resource costs, it cannot be dynamically adapted considering the supply and demand. On the contrary, in the two optimal price schemes, price is raised as the demand grows. This is why these two schemes get more profit than the proportional price scheme. However, constrained by their computing resource capacities, the revenues gained by the price schemes without the BCS are unable to grow continually with the increase of the number of the arriving vehicles. In our proposed hierarchical VEC offloading scheme, the price can be adjusted optimally as the BCS is utilized for making up the computing resource deficit. Thus, our proposed scheme gains the highest revenue compared to the other two schemes either with small or large number of vehicles.

Fig. 3 compares the revenues of the selected VEC servers with and without the BCS. Here, server 1 is the nearest server to the starting point of the road, and server-3 is the farthest one. From this figure, we can see that server 2 and server 3 obtain higher revenues with the increasing speed of the vehicles. However, the revenue of server 1 falls down as the speed grows higher. The reason is that when the speed is slow, a large

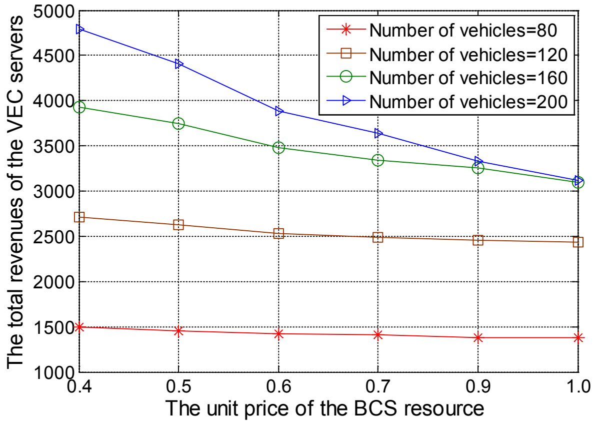

Fig. 4. The total revenues of the VEC servers affected by BCS resource price.

number of vehicles cannot reach the VEC servers located at the remote end of the road under the tasks' delay constraints. They prefer to offload their tasks to the servers being close to the starting point of the road. Thus, the revenue of server 1 is high. As the speed increases, vehicle travel delay is reduced. More farther VEC servers can be selected as the task offloading targets, and the competition between these VEC servers gets more intense. As a result, the revenue of server 1 decreases while the revenues of server 2 and server 3 are increasing.

In Fig. 3, we further find that the revenues gained by the servers with the BCS are generally greater than them gained by the servers without the BCS. The improvement of the revenue is brought by the backup resource supply from the BCS when the computing resource requirement on a VEC server exceeds its capacity. It is noteworthy that the revenue obtained by server 2 with the BSC is less than that of the same server without the BSC when vehicle speed is at 60 km/hr. With the aiding of the BCS, server 1 offloads more tasks. Since the speed of 60 km/hr is slow, lots of vehicles prefer to choose server 1 as the offloading target. Due to the competition between the VEC servers, the increase of the tasks offloaded to server 1 reduces the offloading to server 3.

Fig. 4 shows the effects of the BCS resource price on the revenue of the VEC servers in the cases with different vehicles. We can see that all the revenues of these cases fall down as the price increases. Due to the computing resource capacities of the VEC servers, more tasks may be offloaded to the BCS when the number of the vehicles becomes higher. Thus, the revenue in the case with more vehicles is more likely to be affected by the change of the BCS resource price, and decreases faster with the increase in price. It should be noted that when the price is 1.0, the revenue of the case with 200 vehicles is almost equal to that of the case where the number of the vehicles is 160. The reason is that according to Lemma 1, there is an upper limit of the VEC servers' resource selling price. When the price of the BCS is high, the VEC servers may not buy resource from the BCS. Furthermore, the volume of the VEC servers' offloading service is limited due to their resource limitations. Thus, the gained revenues of the servers are almost the same in the cases with high BCS price together with large number of vehicles.

VI. Conclusion

In this paper, we proposed a hierarchical VEC offloading framework in cloud-based vehicular networks. Based on the framework, we investigated the task offloading mechanism and formulated it as a Stackelberg game. We developed a distributed algorithm for obtaining the optimal strategies of the VEC servers, which maximize their revenues while ensure the delay constraints of the computation tasks. In addition, we validated the revenue enhancement in our proposed scheme through analytical and numerical results.

References

[1] C. Wang, Y. Li, D. Jin and S. Chen, "On the serviceability of mobile vehicular cloudlets in a large-scale urban environment," *IEEE Trans. Intelligent Transportation Systems*, vol. 17, no. 10, pp. 2960-2970, Oct. 2016.

[2] R. Yu, J. Ding, X. Huang, M. Zhou, S. Gjessing and Y. Zhang, "Optimal resource sharing in 5G-enabled vehicular networks: a matrix game approach," *IEEE Trans. Vehicular Technology*, vol. 65, no. 10, pp. 7844-7856, Oct. 2016.

[3] C. Shao, S. Leng, Y. Zhang, A. Vinel and M. Jonsson, "Performance analysis of connectivity probability and connectivity-aware MAC protocol design for platoon-based VANETs," IEEE Trans. Vehicular Technology, vol. 64, no.12, pp. 5596-5609, Dec. 2015.

[4] M. Deng, H. Tian and X. Lyu, "Adaptive sequential offloading game for multi-cell mobile edge computing," in *Proc. 23rd International Conference on Telecommunications (ICT)*, pp. 1-5, May 2016.

[5] 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, June 2016.

[6] J. Jeong, H. Jeong, E. Lee, T. Oh and D. H. C. Du, "SAINT: Self-adaptive interactive navigation tool for cloud-based vehicular traffic optimization," *IEEE Trans. Vehicular Technology*, vol. 65, no. 6, pp. 4053-4067, June 2016.

[7] M. A. Salahuddin, A. Al-Fuqaha and M. Guizani, "Reinforcement learning for resource provisioning in the vehicular cloud," *IEEE Wireless Communications*, vol. 24, no. 4, pp. 128-135, Aug. 2016.

[8] X. Chen, L. Jiao, W. Li and X. Fu, "Efficient multi-user computation offloading for mobile-edge cloud computing," *IEEE/ACM Transactions on Networking*, vol. 24, no. 5, pp. 2795 -2808, Oct. 2016.

[9] K. Zhang, Y. Mao, S. Leng, Q. Zhao, L. Li, X. Peng, L. Pan, S. Maharjan and Y. Zhang, "Energy-efficient offloading for mobile edge computing in 5G heterogeneous networks," *IEEE Access*, vol. 4, pp. 5896-5907, Aug. 2016.

[10] K. Zhang, Y. Mao, S. Leng, A. Vinel and Y. Zhang, "Delay constrained offloading for mobile edge computing in cloud-enabled vehicular networks," in *Proc. 11th International Workshop on Communication Technologies for Vehicles*, Sep. 2016.

[11] N. Kumar, S. Zeadally and J. J. P. C. Rodrigues, "Vehicular delay-tolerant networks for smart grid data management using mobile edge computing," *IEEE Communications Magazine*, vol. 54, no. 10, pp. 60-66, Oct. 2016.

[12] Y. Zhang, R. Yu, W. Yao, S. Xie, Y. Xiao and M. Guizani, "Home M2M networks: architectures, standards, and QoS improvement," *IEEE Communications Magazine*, vol. 49, no. 4, pp. 44-52, Apr. 2011.

[13] S. Maharjan, Q. Zhu, Y. Zhang, S. Gjessing and T. Basar, "Demand response management in the smart grid in a large population regime", *IEEE Trans. Smart Grid*, vol. 7, no. 1, pp. 189-199, Jan. 2016.

[14] J. B. Rosen, "Existence and uniqueness of equilibrium points for concave n-person games," Econometrica, vol. 33, no. 3, pp. 520- 534, Jul. 1965.

[15] G. Debreu, "A social equilibrium existence theorem," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 38, no. 10, pp. 886-893, Oct. 1952.

[16] S. Maharjan, Q. Zhu, Y. Zhang, S. Gjessing and T. Basar, "Dependable demand response management in the smart grid: a Stackelberg game approach," *IEEE Trans. Smart Grid*, vol. 4, no. 1, pp. 120-132, Mar. 2013.