



# Incentive Offloading with Communication and Computation Capacity Concerns for Vehicle Edge Computing

Chenliu Song, Ying Li<sup>(✉)</sup>, Jianbo Li, and Chunxin Lin

College of Computer Science and Technology, Qingdao University, Qingdao, China  
yingli2016@qdu.edu.cn

**Abstract.** With the popularity of intelligent vehicles, computation-intensive vehicle tasks rise dramatically. Vehicle edge computing (VEC) is a promising technology that offloads overloaded computation tasks of intelligent vehicles to the edge. However, VEC servers are constrained by their available computation capacity while dealing with numerous tasks. To this end, we propose multi-party cooperation to complete vehicle task offloading. Computation-assisted vehicles (CAVs) with free resources assist VEC servers to offload Computation-required vehicles (CRVs), which enables computation resources of VEC servers and CAVs for CRVs' task execution. To motivate positive participation of VEC servers and CAVs, we design a resource management and pricing mechanism by quantifying their gains and costs. Such design efficiently integrate and leverage the communication mode and computing mode among participants to describe their interactions, which composes two two-stage Stackelberg games. While Nash equilibrium (NE) for each Stackelberg game reaches, none of participants violates unilaterally. Simulation results demonstrate its effectiveness of the proposed model.

**Keywords:** Vehicle edge computing · Incentive mechanism · Stackelberg game · Resource allocation

## 1 Introduction

With the rapid development of Internet of Things (IoT), various transport system applications win growing popularity in recent years. Intelligent vehicle services generate high performance demands, such as low-latency communication and intensive computation. Vehicle edge computing (VEC) thus emerges for

---

Supported by National Natural Science Foundation of China under Grant No. 61802216, National Key Research and Development Plan Key Special Projects under Grant No. 2018YFB2100303, Shandong Province colleges and universities youth innovation technology plan innovation team project under Grant No. 2020KJN011, Program for Innovative Postdoctoral Talents in Shandong Province under Grant No. 40618030001.

supporting computing, storage and network bandwidth [1]. Many studies introduce a variety of VEC technology, including vehicle fog computing [2], hybrid combination of fog computing architecture and vehicle cloud computing [3] and so on. VEC servers execute computation tasks, generated by the vehicles at the end, while reducing the delay. To improve the QoS of vehicle applications, studies have investigated if peripheral vehicles with free resources cooperate with VEC server to complete computing tasks. Authors in [4] studied task scheduling scheme for enhancing computing resource utilization with system stability and low latency. Some other studies on vehicle edge computing are also discussed in [5, 6].

In this paper, we are motivated to achieve mutual cooperation for vehicle task offloading. Vehicles, constrained by limited resources, require VEC serves for task execution. A large number of computation tasks burden VEC servers. To improve VEC efficiency, we consider numerous vehicles with free resources to assist the computing work of VEC servers. However, due to energy usage, most vehicles unenthusiastically participate in task calculation. In doing so, a framework of resource allocation and pricing is designed to economically compensate their costs and stimulate their cooperation.

## 2 System Module and Game Formulation

### 2.1 System Model

Our model focuses on task offloading between one VEC server and a group of vehicles. The proposed model is illustrated. Assume that a VEC server connects  $m$  CRVs and  $n$  CAVs within its coverage, denoted by  $\mathcal{M} = \{1, 2, \dots, m\}$  and  $\mathcal{N} = \{1, 2, \dots, n\}$ . We also assume CRV  $i$ 's total computation tasks as  $w_i^r$ ,  $i \in \mathcal{M}$ . Let  $\mathbf{p}^r$  denote CRVs' transmit powers over the VEC server's sub-channels with  $\mathbf{p}^r = \{p_1^r, p_2^r, \dots, p_m^r\}$  while offloading tasks to VEC server. Specially,  $p_i^r$  is the transmit power from CRV  $i$  to VEC server,  $i \in \mathcal{M}$ . Similarly, the allocated powers for transmitting tasks to CAVs are denoted as  $\mathbf{p}^a = \{p_1^a, p_2^a, \dots, p_n^a\}$ , where  $p_j^a$  corresponds to the transmit power from VEC server to CAV  $j$ ,  $j \in \mathcal{N}$ .

According to the communication model in [7], the channel rate from CRV  $i$ ,  $i \in \mathcal{M}$  to VEC server is defined,

$$R_i^r = B_i^r \log_2 \left( 1 + \frac{p_i^r h_i^r}{\sigma_i^2} \right), \quad (1)$$

The channel rate from VEC server to CAV  $j$  is also defined,

$$R_j^a = B_j^a \log_2 \left( 1 + \frac{p_j^a h_j^a}{\sigma_j^2} \right), \quad (2)$$

where  $B_i^r$  and  $B_j^a$  represent the transmission bandwidth of the subchannel,  $\sigma_i$  and  $\sigma_j$  represent noise power, and  $h_i^r$  and  $h_j^a$  represent the subchannel power gain.

We here define the utility function of a CRV as the difference between its gains and its costs. Service satisfaction contributes to its gain. Local computation cost, incentive payment and task transmission cost incur its total costs. Let  $T_i^r$  represent the uplink transmission time of delivering CRV  $i$ 's tasks to VEC server, which indicates its task size of  $R_i^r T_i^r$ . Given the unit pricing  $u_0$  that VEC server charges CRVs, we define the utility of CRV  $i, i \in \mathcal{M}$ ,

$$U_i^r(u_0, p_i^r) = s_i^r R_i^r T_i^r - \varepsilon_i^r (w_i^r - R_i^r T_i^r) - u_0 p_i^r - \xi_i^r p_i^r, \quad (3)$$

where  $s_i^r$  is CRV  $i$ 's internal demand rate,  $\varepsilon_i^r$  is the unitary local computation cost and  $\xi_i^r$  is the unitary transmission cost. Accordingly, the utility of CAV  $j, j \in \mathcal{N}$  is outlined as,

$$U_j^a(u_j^a, p_j^a) = u_j^a p_j^a - \varepsilon_j^a R_j^a T_j^a, \quad (4)$$

where  $u_j^a$  is the incentive gain per unitary transmit power,  $\varepsilon_j^a$  is the unitary computation consumption. The parameter  $T_j^a$  represents computation tasks' uplink transmission time while delivering from VEC server to CAV  $j$ . It indicates the task size of  $R_j^a T_j^a$ , offloaded by CAV  $j$ . For the VEC server, its gain is the payment offered by CRVs. The total costs include local computation cost, incentive payment for CAVs, and power cost of transmitting task to the CAVs. So, the utility function  $U^{vec}$  of VEC server can be defined as:

$$U^{vec}(u_0, \mathbf{u}^a, \mathbf{p}^r, \mathbf{p}^a) = \sum_{i=1}^{\mathcal{M}} u_0 p_i^r - \sum_{j=1}^{\mathcal{N}} u_j^a p_j^a - \tau \sum_{j=1}^{\mathcal{N}} p_j^a - \varepsilon_0 \left( \sum_{i=1}^{\mathcal{M}} R_i^r T_i^r - \sum_{j=1}^{\mathcal{N}} R_j^a T_j^a \right), \quad (5)$$

where  $\varepsilon_0$  is the unitary computation cost of VEC server and  $\tau$  is the unit transmission cost. Denote  $\mathbf{u}^a$  as  $\mathbf{u}^a = \{u_1^a, u_2^a, \dots, u_n^a\}$ . For ease of illustration, we decompose the utility function in (5) into  $U^{vec} = U_1^{vec} + U_2^{vec}$ , in which,

$$U_1^{vec}(u_0, \mathbf{p}^r) = \sum_{i=1}^{\mathcal{M}} u_0 p_i^r - \varepsilon_0 \sum_{i=1}^{\mathcal{M}} R_i^r T_i^r, \quad (6)$$

$$U_2^{vec}(\mathbf{u}^a, \mathbf{p}^a) = \sum_{j=1}^{\mathcal{N}} \varepsilon_0 R_j^a T_j^a - \sum_{j=1}^{\mathcal{N}} (u_j^a + \tau) p_j^a. \quad (7)$$

## 2.2 Problem Formulation

In the three-tiered model, each participant has its own strategy and goal. Their different goals formulate different computation task management and pricing problems.

For player CRV  $i, \forall i \in \mathcal{M}$ , its strategy is to determine the transmit power  $p_i^r$  from it to VEC server in order to maximize the utility in (3). So, the CRV  $i$ 's optimization problem is stated as,

$$p_i^{r*} = \arg \max_{p_i^r > 0} U_i^r(u_0, p_i^r) \tag{8a}$$

$$\text{subject to } \underline{p}^r \leq p_i^r \leq \bar{p}^r, \tag{8b}$$

where  $\underline{p}^r$  and  $\bar{p}^r$  correspond to minimal and maximum transmit powers of CRVs.

For the player of VEC server, its strategy is to decide the pricing  $u_0$  of CRVs's computation tasks and transmit power  $\mathbf{p}^a$  from it to CAVs in order to maximize the utility in (5). So, the VEC server's optimization problem is stated as,

$$(\mathbf{p}^{a*}, u_0) = \arg \max_{(p_j^{a*}, u_0)} U^{vec}(u_0, \mathbf{u}^a, \mathbf{p}^r, \mathbf{p}^a) \tag{9a}$$

$$\text{subject to } \underline{p}^r \leq p_i^r \leq \bar{p}^r, \tag{9b}$$

$$\underline{p}^a \leq p_j^a \leq \bar{p}^a, \tag{9c}$$

where  $\underline{p}^a$  and  $\bar{p}^a$  correspond to minimal and maximum transmit powers of CAVs.

Since the utility function in (5) can be separately divided into  $U^{vec} = U_1^{vec} + U_2^{vec}$ , problem (9) is thus decomposed into two subproblems as follows,

$$u_0^* = \arg \max_{u_0} U_1^{vec}(u_0, \mathbf{p}^r) \tag{10a}$$

$$\text{subject to } \underline{p}^r \leq p_i^r \leq \bar{p}^r, \tag{10b}$$

and,

$$p_j^a = \arg \max_{p_j^a} U_2^{vec}(u_j^a, p_j^a) \tag{11a}$$

$$\text{subject to } \underline{p}^a \leq p_j^a \leq \bar{p}^a. \tag{11b}$$

For player CAV  $j, \forall j \in \mathcal{N}$ , its strategy is to determine the pricing  $u_j^a$  of VEC server's overloaded tasks for maximizing the utility in (4). So, the CAV  $j$ 's optimization problem is described as,

$$u_j^{a*} = \arg \max_{u_j^a > 0} U_j^a(u_j^a, p_j^a) \tag{12a}$$

$$\text{subject to } \underline{p}^a \leq p_j^a \leq \bar{p}^a. \tag{12b}$$

### 2.3 Stackelberg Game Building

We here build two consecutive Stackelberg games for describing the interactions among CRVs, CAVs and VEC server, shown in Fig. 2. The VEC server acts as the leader of CRVs to determine the power pricing  $u_0$  for CRVs' task. CRVs, as the followers of VEC server, then decide their transmit power strategy  $\mathbf{p}^r$ . The interactions between CRVs and VEC server formulates the first two-stage Stackelberg game. Problem (8) and Problem (10) jointly formulates such noncooperative game  $\mathcal{G}^{CRV} = \{\mathcal{M}, \{p_i^r\}_{i \in \mathcal{M}}, \{U_i^r\}_{i \in \mathcal{M}}\}$ , where  $\mathcal{M}$  is the set of CRVs,  $\{p_i^r\}_{i \in \mathcal{M}}$  is the strategy set and  $\{U_i^r\}_{i \in \mathcal{M}}$  is the cost function set.

CAVs, as the leaders, determine power pricing  $\mathbf{u}^a$  for VEC server's overloaded tasks. Accordingly, VEC server acts as the follower of CAVs in the second stage to determine its transmit power strategy  $\mathbf{p}^a$ . The interaction between CAVs and VEC server formulates the second two-stage Stackelberg game. In detail, Problem (11) and Problem (12) composes CAVs' game  $\mathcal{G}^{CAV} = \{\mathcal{N}, \{u_j^a\}_{j \in \mathcal{N}}, \{U_j^a\}_{j \in \mathcal{N}}\}$ , where  $\mathcal{N}$  is the set of CAVs,  $\{u_j^a\}_{j \in \mathcal{N}}$  is the strategy set and  $\{U_j^a\}_{j \in \mathcal{N}}$  is the set of profit functions.

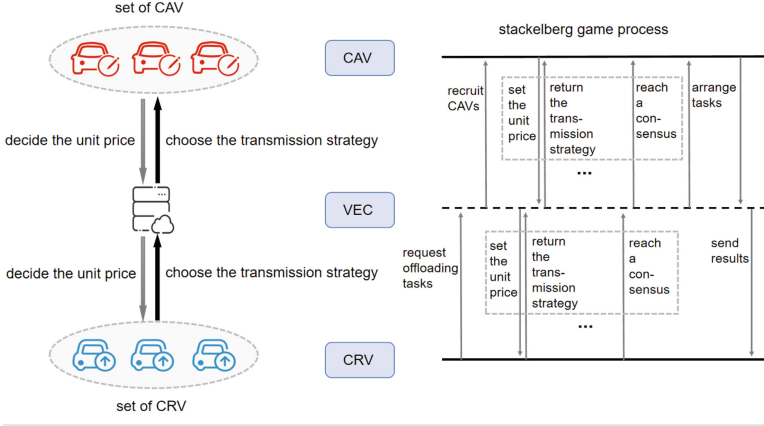


Fig. 1. Interactions in Stackelberg game

**Definition 1.** *Subgame Perfect Equilibrium (SPE) in  $\mathcal{G}^{CRV}$* : If the stratification strategy  $\{u_0^*, \mathbf{p}^{r*}\}$  represents SPE, it can reach NE in each stage of subgame, i.e.:

$$\begin{cases} \text{Stage I: } u_0^* = \arg \max_{u_0} U_1^{vec}(u_0, \mathbf{p}^r) \\ \text{Stage II: } \mathbf{p}_i^{r*} = \arg \max_{\mathbf{p}_i^r} U_i^r(u_0, \mathbf{p}_i^r) \forall i \in \mathcal{M} \end{cases} \quad (13)$$

**Definition 2.** *Subgame Perfect Equilibrium (SPE) in  $\mathcal{G}^{CAV}$* : If the stratification strategy  $\{\mathbf{u}^{a*}, \mathbf{p}^{a*}\}$  represents SPE, it can reach the NE in each stage of subgame, i.e.:

$$\begin{cases} \text{Stage I: } u_j^{a*} = \arg \max_{u_j^a} U_j^a(u_j^a, \mathbf{p}_j^a), \forall j \in \mathcal{N} \\ \text{Stage II: } \mathbf{p}^{a*} = \arg \max_{\mathbf{p}^a} U_2^{vec}(u_j^a, \mathbf{p}^{a*}) \end{cases} \quad (14)$$

Each game reaches SPE, which corresponds to a stable solution point of these problems. Under NE, no player has incentive to deviate. In detail, we use backward induction to analyze Stackelberg game. CRVs initiate the transmission strategy  $\mathbf{p}^r$  based on VEC server's pricing  $u_0$ . To finish CRVs' computation tasks, the VEC server also recruits computation-assisted vehicles. CAVs determine a power pricing for VEC server's overloaded tasks, on which VEC server

allocates the transmit power  $\mathbf{p}^a$  for CAVs. The process continues until Nash equilibrium reaches. The best strategy  $\{u_0^*, \mathbf{u}^a, \mathbf{p}^r, \mathbf{p}^a\}$  corresponds to the Nash equilibrium point, which is solved in the following theorems.

**Theorem 1.** *Given the unit price  $u_0$ , the optimal strategy  $\mathbf{p}^r$  of CRVs is calculated in (15).*

$$p_i^{r*} = \gamma_i^r(\mathbf{p}^r) = \begin{cases} \underline{p}^r & \text{if } \frac{(s_i^r + \varepsilon_i^r)B_i^r T_i^r}{\ln 2(u_0 + \xi_i^r)} - \frac{\sigma_i^2}{h_i^r} < \underline{p}^r, \\ \frac{(s_i^r + \varepsilon_i^r)B_i^r T_i^r}{\ln 2(u_0 + \xi_i^r)} - \frac{\sigma_i^2}{h_i^r} & \text{otherwise,} \\ \bar{p}^r & \text{if } \frac{(s_i^r + \varepsilon_i^r)B_i^r T_i^r}{\ln 2(u_0 + \xi_i^r)} - \frac{\sigma_i^2}{h_i^r} > \bar{p}^r. \end{cases} \quad (15)$$

**Theorem 2.** *VEC server achieves the unique optimal pricing strategy  $u_0^*$  by solving subproblem (10) while  $p_i^r \in [\underline{p}^r, \bar{p}^r], i \in \mathcal{M}$ .*

*Proof.* Substituting (15) into (6), we calculate the first derivative and second derivative of  $U_1^{vec}$  with respect to  $u_0$ ,

$$\frac{\partial U_1^{vec}}{\partial u_0} = \sum_{i=1}^{\mathcal{M}} \frac{(s_i^r + \varepsilon_i^r + \varepsilon_0) B_i^r T_i^r}{\ln 2(u_0 + \xi_i^r)} - \frac{\sigma_i^2}{h_i^r} - \frac{u_0 (s_i^r + \varepsilon_i^r) B_i^r T_i^r}{\ln 2(u_0 + \xi_i^r)^2}, \quad (16)$$

$$\frac{\partial^2 U_1^{vec}}{\partial u_0^2} = -\sum_{i=1}^{\mathcal{M}} B_i^r T_i^r \left[ \frac{2(s_i^r + \varepsilon_i^r) \xi_i^r + \varepsilon_0 (u_0 + \xi_i^r)}{\ln 2(u_0 + \xi_i^r)^3} \right]. \quad (17)$$

Obviously, the second derivative of  $U_1^{vec}$  on  $u_0$  is always negative. So,  $U_1^{vec}$  is a strict concave function with respect to  $u_0$ , which indicates the uniqueness of optimal solution  $u_0^*$ . Because of non-linear complexity of  $\frac{\partial U_1^{vec}}{\partial u_0}$  in (16), we approximate  $u_0^*$  by gradient ascent method.

**Theorem 3.** *VEC server achieves the unique optimal transmit power strategy  $\mathbf{p}^{a*}$  in (18) by solving subproblem (11) while  $p_j^a \in [\underline{p}^a, \bar{p}^a], j \in \mathcal{N}$ .*

$$p_j^{a*} = \delta_j^a(\mathbf{p}^a) = \begin{cases} \underline{p}^a & \text{if } \frac{\varepsilon_0 T_j^a B_j^a}{\ln 2(u_j^a + \tau)} - \frac{\sigma_j^2}{h_j^a} < \underline{p}^a, \\ \frac{\varepsilon_0 T_j^a B_j^a}{\ln 2(u_j^a + \tau)} - \frac{\sigma_j^2}{h_j^a} & \text{otherwise,} \\ \bar{p}^a & \text{if } \frac{\varepsilon_0 T_j^a B_j^a}{\ln 2(u_j^a + \tau)} - \frac{\sigma_j^2}{h_j^a} > \bar{p}^a. \end{cases} \quad (18)$$

**Theorem 4.** *CAV  $j, j \in \mathcal{N}$  reaches the optimal pricing  $u_j^{a*}$  in (19) by solving problem (12),*

$$u_j^{a*} = R_j^a(\mathbf{u}^a) = \frac{\sqrt{(\Delta_j^1 - \Delta_j^2 - 2\Delta_j^3 \tau)^2 + 4\tau \Delta_j^3 (\Delta_j^1 - \Delta_j^3 \tau)} - (2\Delta_j^3 + \Delta_j^2 - \Delta_j^1)}{2\Delta_j^3}, j \in \mathcal{N}, \quad (19)$$

with  $\Delta_j^1 = (\varepsilon_0 + \varepsilon_j^a) T_j^a B_j^a, \Delta_j^2 = T_j^s B_j^a \varepsilon_0$  and  $\Delta_j^3 = \ln 2 \frac{\sigma_j^2}{h_j^a}$ .

### 3 Performance Evaluation

#### 3.1 Simulation Setup

We consider a simulation scenario in which a VEC server provides services for  $m$  CRVs with assistance of  $n$  CAVs for task offloading. Each VEC server is able to cover about 10–30 CRVs. Referring to [8–10], the default model parameters are stated as follows. It is assumed that the bandwidth of each subchannel is 20 MHz and its channel gain is 53 dbm. The noise power of the channel of the relevant vehicles is 10 db. The task transmission time  $T_i^r$  and  $T_j^a$  is set from 0.5 ms to 1.5 ms. CRVs’ satisfaction on offloaded tasks follows a normal distribution  $N(e_0, \sigma_0)$ . The number of CRVs and CAVs are 10 and 10 by default.

#### 3.2 Simulation Results

We first investigate the effect of VEC server’s computation cost on its utility under different number of participants. We see from Fig. 2(a) and (b) that the increasing number of CRVs and CAVs is indeed helpful for enhancing VEC utility. While fixing the CRV number and CAV number, we start with observing a descending trend of VEC utility on computation cost. However, VEC server’s utility finally improves with incremental computation cost. Such situation is illustrated in Fig. 2(c) and (d). The growing computation cost incurs CRV’s decreasing contribution on VEC utility and CAV’s increasing contribution on VEC utility. While VEC server’s payoff increased by CRV completely cover that decreased by CAV, VEC utility turns into improvement.

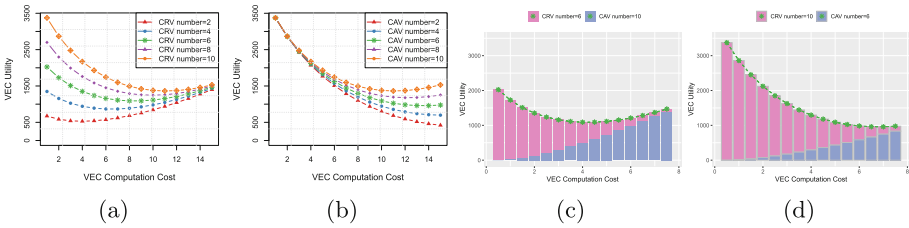


Fig. 2. Effect of computation cost on utility of VEC server

We next observe the comparison by altering the transmit power cost of VEC server. Figure 3(a) and (b) shows an ascending trend of VEC utility while more CRVs and CAVs participate in computation offloading. We also observe that the utility of VEC server tends to decline with the increasing VEC power cost in Fig. 3(a) and (b). Such result is explained in Fig. 3(c) and (d). VEC power cost has a negative effect on CAV’ contribution for VEC server’s payoff. However, CRV’s contribution on VEC server’s payoff don’t follow a significant change. Accordingly the utility of VEC server decreases.

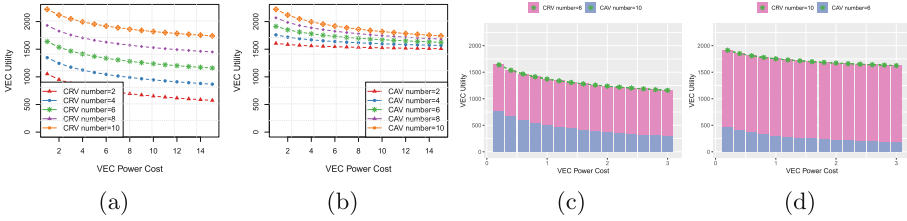


Fig. 3. Effect of VEC power cost on profit of VEC server

The CRV computation cost is also explored if it determines the performance of the proposed model. It can be observed in Fig.4(a) that local computation cost of CRV has no obvious effect on pricing, determined VEC server. With the increase of local computation cost, the steady pricing motivates the CRV to improve the transmit power for reducing local computation consumption in Fig.4(a). We see from Fig.4(b) that CRV utility depends on its computation cost, as well as task transmission time. Given a computation task, shorter task transmission time incurs higher local computation consumption. So, CRV utility tends to dwindle with the increasing local computation cost.

We finally evaluate CAV computation cost on the model performance. Figure4(c) shows the transmit power, allocated by VEC server, and power pricing, determined by CAV. The large CAV computation cost incurs CAV’s high pricing. Accordingly, VEC server reduces the transmit power for utility maximization. Therefore, Fig. 4(c) shows an increasing pricing for VEC power, as well as a decreasing VEC transmit power. Under such scenario, CAV utility gradually degrades with the increasing CAV computation cost in Fig. 4(d). While the power pricing is large enough, the VEC power decreases to zero. CAV utility accords with being zero. Given the computation cost, long task transmission time for CAV promotes the improvement of power pricing and transmit power, on which CAV utility increases.

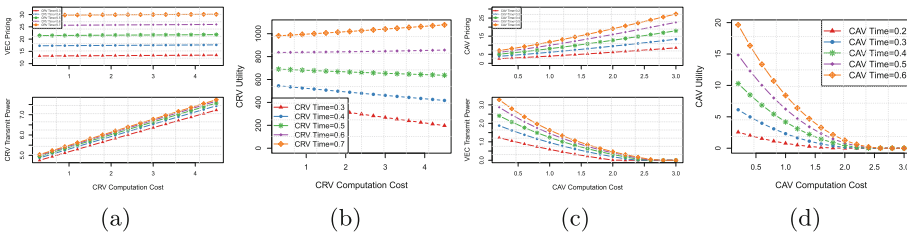


Fig. 4. Effect of CRV and CAV computation cost on utility of VEC server



## 4 Conclusion

In this paper, we study the incentive mechanism of task offloading based on vehicle edge computing. Vehicles, constrained by limited computation resources, request VEC server for executing computation tasks. To avoid more burdens on VEC server, vehicles with free resources are recruited to assist VEC server's computation work. We are motivated to quantify their gains and costs by defining utility function. Specially, the establishment of utility function considers the modes of task transmission and task computation among them, which is aimed to accurately describe their profits. Moreover, we build resource management and pricing framework by tools of Stackelberg game to ensure that each participant has appropriate incentives to participate in task offloading. Simulation results show that our incentive framework is stable and effective. In the future work, we will concentrate on computation resource management in extensive edge computing networks.

## References

1. Raza, S., Wang, S., Ahmed, M., Anwar, M.R.: A survey on vehicular edge computing: architecture, applications, technical issues, and future directions. *Wirel. Commun. Mob. Comput.* **2019**, 3159762:1–3159762:19 (2019)
2. Hou, X., Li, Y., Chen, M., Wu, D., Jin, D., Chen, S.L.: Vehicular fog computing: a viewpoint of vehicles as the infrastructures. *IEEE Trans. Veh. Technol.* **65**, 3860–3873 (2016)
3. Boussehham, M., Benamar, N., Addaim, A.: A new security mechanism for vehicular cloud computing using fog computing system. In: 2019 International Conference on Wireless Technologies, Embedded and Intelligent Systems (WITS) pp. 1–4 (2019)
4. Sun, F., et al.: Cooperative task scheduling for computation offloading in vehicular cloud. *IEEE Trans. Veh. Technol.* **67**, 11049–11061 (2018)
5. Zhang, L., Zhao, Z., Wu, Q., Zhao, H., Xu, H., Wu, X.: Energy-aware dynamic resource allocation in UAV assisted mobile edge computing over social internet of vehicles. *IEEE Access* **6**, 56700–56715 (2018)
6. Hochstetler, J., Padidela, R., Chen, Q., Yang, Q., Fu, S.: Embedded deep learning for vehicular edge computing. In: 2018 IEEE/ACM Symposium on Edge Computing (SEC), pp. 341–343 (2018)
7. Hao, Y., Chen, M., Hu, L., Hossain, M.S., Ghoneim, A.: Energy efficient task caching and offloading for mobile edge computing. *IEEE Access* **6**, 11365–11373 (2018)
8. Qian, J., Duan, B., Xie, W., Zhao, Y.: Edge computing for brightness and color temperature of smart streetlight. In: 2021 IEEE 2nd International Conference on Big Data, Artificial Intelligence and Internet of Things Engineering (ICBAIE), pp. 699–703 (2021)
9. Shang, B., Zhao, L., Chen, K.C., Chu, X.: An economic aspect of device-to-device assisted offloading in cellular networks. *IEEE Trans. Wireless Commun.* **17**, 2289–2304 (2018)
10. Shi, L., Zhao, L., Zheng, G., Han, Z., Ye, Y.: Incentive design for cache-enabled d2d underlaid cellular networks using Stackelberg game. *IEEE Trans. Veh. Technol.* **68**, 765–779 (2019)