C-V2X based Offloading Strategy in Multi-Tier Vehicular Edge Computing System

Weiyang Feng*, Siyu Lin*, Ning Zhang†, Gongpu Wang‡, Bo Ai*, and Lin Cai§
*School of Electronic and Information Engineering, Beijing Jiaotong University, Beijing 100044, China
†Department of Electrical and Computer Engineering, University of Windsor, ON, Canada
‡School of Computer and Information Technology, Beijing Jiaotong University, Beijing 100044, China
§Department of Electrical and Computer Engineering, University of Victoria, Victoria, Canada
Siyu Lin (Corresponding Author) Email: sylin@bjtu.edu.cn

GLOBECOM 2022 - 2022 IEEE Global Communications Conference | 978-1-6654-3540-6/22/$31.00 ©2022 IEEE | DOI: 10.1109/GLOBECOM48099.2022.10001050

***Abstract*—Many emerging intelligent transportation services are latency-sensitive with heavy demand for computing resources, which can be handled by a multi-tier computing system composed of vehicular edge computing (VEC) servers in the roadside and micro servers carried by vehicles. In multi-tier VEC system, the offloading of vehicle-to-vehicle (V2V) can be supported using the Cellular Vehicle-to-Everything (C-V2X) links, through Uu or PC5 interfaces. In this work, we investigate the offloading and resource strategy in C-V2X enabled multi-tier VEC system. The successful transmission probability of PC5 interface is modeled to characterize the normalized transmission rate of C-V2X link. We aim to minimize the total system latency of the task processing to optimize the offloading ratio matrix and packet transmit frequency of the PC5 interface, and computation resource allocation of vehicles and VEC server. Due to the non-convex and variables coupling, the latency minimization problem is decomposed into two subproblems, i.e., resource allocation and offloading strategy subproblems, and propose a PC5 interface based greedy offloading (PC5-GO) algorithm. Specifically, for the resource allocation subproblem, we derive the closed expressions of packet transmit frequency of PC5 interface and CPU computation frequency at vehicle and VEC server. For the offloading strategy subproblem, the offloading ratio matrix is obtained by the proposed PC5-GO algorithm. Simulation results are provided that the proposed PC5-GO algorithm can significantly enhance the system performance compared with other benchmark schemes by 5.88% at least.**

***Index Terms*—Multi-tier vehicular edge computing, C-V2X, PC5 interface, offloading strategy, resource allocation**

I. Introduction

The Internet of Vehicles (IoV) [1], [2] is emerging in support of providing network connections for intelligent transportation services, and computation resources are also needed to meet the latency requirement for intelligent transportation services. With the increasing number of vehicles and high computation demand of services, the computing resource of traditional edge computing cannot meet the performance requirement of transportation services [3], [4]. Thus edge computing and vehicular computing are combined to form a multi-tier vehicular edge computing (VEC) system [5], [6] to realize the decision-making of computation-intensive intelligent applications. As a new auxiliary computing and integration platform, the multi-tier VEC system can configure servers at the road side unit (RSU) or other vehicles on the edge side to provide computing for vehicles with insufficient computing capacity [7]. In the multi-tier VEC system, the vehicle can offload tasks to VEC servers or other vehicles for auxiliary computing to reduce the task processing latency.

In the literature, many works on task offloading in the VEC system have been reported. For latency minimization, Dai *et al.* [8] formulate a cooperative computation offloading problem to minimize the expected system service latency. The reverse offloading framework is proposed to relieve the burden of the VEC server and reduce the system latency [9], and the authors design offloading algorithms for binary offloading and partial offloading schemes to minimize the system latency. For other metrics, Wang *et al.* [10] propose a multiuser noncooperative computation offloading game and construct a distributed best response algorithm, where each vehicle can adjust its offloading probability to realize the maximum utility. Considering the dynamics of mobile vehicular networks, Kazmi *et al.* [11] propose a Deep Reinforcement Learning (DRL) based task-offloading approach to minimize the energy consumption subject to the network resources. To maximize system utility in VEC network, Dai *et al.* [12] formulate a load balancing and offloading problem by jointly optimizing RSU selection, offloading decision, and computation resource allocation.

However, some of the above studies only focus on the orthogonal resources allocation at Uu interface, which requires a large coverage of infrastructure, i,e., base stations (BS). In fact, due to the high mobility and uncertainty of the IoV, it is difficult to guarantee that vehicles are always covered by BS, and thus communication between vehicles is mainly realized through PC5 interface. The system based on Cellular Vehicle-to-Everything (C-V2X) can better cope with the changing network topology, where vehicles communicate with other vehicles through the LTE or NR PC5 interfaces while using the LTE or NR Uu interface to communicate with BSs. 3GPP Release 14 proposes a semi-persistent scheduling (SPS) algorithm based on the channel sensing in mode 4 in LTE V2X [13], and Release 16 further supplements it in mode 2

This work was supported in part by Beijing Natural Science Foundation under Grants 4212007, Fundamental Research Funds for the Central Universities, China under Grant 2022JBQY004, in part by the National Natural Science Foundation of China under Grants 61971030 and 61861002, and in part by the Project of China Railway Corporation under Grants P2021S005 and P2020G005.

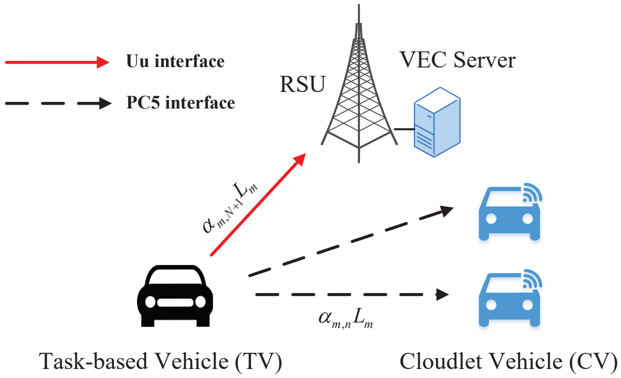


Fig. 1. System model

in NR V2X [14]. The PC5 interface can realize transmission between vehicles without reliable coverage, so it is more flexible than Uu interface. Jointly exploiting PC5 interface and Uu interface of C-V2X system can better support task offloading and computation of services in the multi-tier VEC system.

The traditional offloading mechanism does not consider the sensing failure between vehicles. In fact, due to the rapid change of the topology of the IoV, the vehicle sensing under the C-V2X architecture is not stable enough. Compared with the traditional offloading strategy, it is a big challenge to introduce C-V2X sensing success probability into system latency modeling. Besides, the different link characteristics of PC5 and Uu interfaces make it difficult to joint optimize the offloading strategy and resource allocation.

In this work, considering the sensing characteristics of C-V2X, we study the influence of successful transmission probability of V2V on the offloading strategy, communication and computation resource allocation. The vehicle generates a series of tasks, and the part of each task is partially offloaded to the VEC server by Uu interface, and the other part is offloaded to nearby vehicles by PC5 interface. Then the latency minimization problem is formulated by optimizing the offloading decision, packet transmit frequency of PC5 interface, and computation frequency. The problem is decomposed into two subproblems, where resource allocation subproblem is solved by KKT condition and offloading strategy subproblem can be addressed by greedy algorithm. Based on the above solution, we proposed a PC5-based offloading strategy to reduce the task processing latency of C-V2X aided multi-tier VEC system.

II. System Model

We consider a multi-tier computing network with multiple vehicles and a single RSU connected with VEC servers, as shown in Fig.1. These vehicles can be divided into two types. The first type is task-based vehicles (TV) demand more computation capacity, which generates a series of tasks. We consider a representative TV in a multi-tier VEC system. The second type is cloudlet vehicles (CV), denoted by the $\mathcal{N} = \{1, 2, \dots, N\}$, which generally provide transportation services, such as buses and taxis, and can deploy micro-computing servers. The multi-tier VEC system is composed of MEC and CVs, which are supported by Uu and PC5 interface respectively. The TV offloads a part of its task to the VEC server through the Uu interface, and the other part to the CV through the PC5 interface. After computation, the VEC server and CV will return the computation results to the TV.

A. Task Model

The TV generates M periodic tasks, denoted by the $\mathcal{M} = \{1, 2, \dots, M\}$, and the size of task is denoted by $L_m, m \in \mathcal{M}$. Considering that each task is separable, partial offloading can be adopted [15]. For each task L_m, one part is offloaded to RSU (VEC server) for computation, and the other part is offloaded to a nearby CV. For convenience of description, we index RSU as the $(N+1)$-th vehicle.

We define $\boldsymbol{\alpha}$ as the offloading ratio matrix, where the entry is $\alpha_{m,n} \in [0, 1], m \in \mathcal{M}, n \in \mathcal{N} \cup \{N + 1\}$. For computing at VEC server, i.e., $n = N + 1$, $\alpha_{m,N+1} > 0$ means the TV offloads a part of m-th task $\alpha_{m,N+1}L_m$ to the VEC server; otherwise, $\alpha_{m,N+1} = 0$ means the TV does not offload any part of m-th task to the VEC server. For computing at CVs, i.e., $n \in \mathcal{N}$, $\alpha_{m,n} > 0$ means the TV offloads a part of m-th task $\alpha_{m,n}L_m$ to the n-th CV; otherwise, $\alpha_{m,n} = 0$ means the TV does not offload any part of m-th task to the n-th CV. Note that for m-th task, only one CV computes it, i.e, for the first column to the n-th column of each row of the matrix $\boldsymbol{\alpha}$, only one element is greater than 0, and all other elements are equal to 0. Similarly, the value of the element $\alpha_{m,n}, n \in \mathcal{N}$ indicates that the part of the m-th task is offloaded to the n-th CV. According to the partial offloading mechanism, when $\alpha_{m,n} > 0$, we have $\alpha_{m,N+1} + \sum_{n=1}^{N} \alpha_{m,n} = 1$.

B. Communication Model

The communication mode is divided into two types based on Uu and PC5 interfaces.

- Uu interface: communication interface between vehicle and RSU.
- PC5 interface: communication interface between vehicles.

1) Uu interface: For Uu interface, the orthogonal frequency division multiple access (OFDMA) technique is adopted for the channel access. The channel gain between the TV and RSU is denoted by $g^{Uu}_{m,N+1}$. The transmission rate of m-th task between the TV and RSU is given by

$$r^{\mathrm{Uu}}_{m,N+1} = B_{m,N+1} \log_2 \left(1 + \frac{p^{Uu}_{m,N+1} g^{Uu}_{m,N+1}}{\sigma^2} \right) \tag{1}$$

where $B_{m,N+1}$ denotes the available bandwidth, σ^2 denotes the noise power, and $p^{Uu}_{m,N+1}$ denotes the transmission power of Uu interface of TV. The offloading latency of offloaded part of m-th task from TV to the VEC server can be given as

$$T^{ser-o}_m = \frac{\alpha_{m,N+1} L_m}{r^{\mathrm{Uu}}_{m,N+1}} \tag{2}$$

2) PC5 interface: The packet transmit frequency of PC5 interface at the TV is denoted by λ, which indicates the number of packets transmitted per second of TV. Here we consider that the transmit frequency can be adjusted adaptively. The practical transmit frequency of the local transmitter is less than the perfect Shannon channel capacity, denoted by $C^{v2v,\max}$.

Next, we introduce the influence of C-V2X characteristics on the normalized V2V transmission rate. Since the wireless transmission mode of C-V2X is half duplex, the half-duplex(HD) error is that the vehicle cannot receive the data packet when it transmits its own data packet in the same sub-frame. The HD error [16] can be given as $\delta_{HD} = \frac{\lambda}{N_{frame}}$, where N_{frame} is the number of sub-frames in one second. Besides, when the received signal power is higher than the sensing power threshold P_{Sen}^{th}, the receiver can decode data. The Packet Sensing Ratio(PSR) [16] can be given as $PSR(d) = \frac{1}{2}\left(1 + erf\left(\frac{P^{PC5} - PL(d) - P_{Sen}^{th}}{\sigma\sqrt{2}}\right)\right)$. Here $PL(d)$ is the pathloss at the distance d and P^{PC5} is the transmission power at the PC5 interface of the TV, which satisfies $P^{PC5} = \sum_{m=1}^{M}\sum_{n=1}^{N} p_{m,n}^{v2v}$. Here $p_{m,n}^{v2v}$ denotes the transmission power of the TV to transmit the m-th task to the n-th CV. The traffic density is denoted by β in vehicles per meter, and the successful sensing probability (SSP) can be given as

$$\mathcal{P}_{SEN} = 1 - \frac{K\beta \cdot 2d_{sen}}{N_{sub-f}N_{sub-c}} \tag{3}$$

where K is the conversion coefficient considering multiple lanes, d_{sen} is the sensing distance of TV, and N_{sub-f} and N_{sub-c} denote the number of sub-frames and sub-channels in the Selection Window (SW), respectively. Thus the successful transmission probability (STP) can be expressed as

$$\mathcal{P}_{STP} = (1 - \delta_{HD}) \cdot \mathcal{P}_{SEN} \cdot PSR(d) \tag{4}$$

The STP will further affect the task transmission rate, hence the normalized transmission rate of m-th task between the TV and CVs n is given by

$$r_{m,n}^{\mathrm{PC5}} = P_{STP}L_m\lambda \tag{5}$$

The offloading latency of offloaded part of m-th task from TV to the n-th CV can be given as

$$T_{m,n}^{veh-o} = \frac{\alpha_{m,n}L_m}{r_{m,n}^{\mathrm{PC5}}} \tag{6}$$

C. Computation Model

The computation workload of the m-th task of TV is denoted by C_m in cycles per bit, which indicates the number of CPU cycles for computing the one-bit task data.

We define the computation frequency of VEC server and of n-th CV assigned to the m-th task part as f_m^{ser} and $f_{m,n}^{veh}$ in cycles per second, respectively. The computation latency for the m-task at the VEC server can be expressed as

$$T_m^{ser-c} = \frac{\alpha_{m,N+1}L_mC_m}{f_m^{ser}} \tag{7}$$

The computation latency for the m-task at n-th CV can be expressed as

$$T_{m,n}^{veh-c} = \frac{\alpha_{m,n}L_mC_m}{f_{m,n}^{veh}} \tag{8}$$

Here we ignore the download latency of results due to the small data size.

III. Problem Formulation

Due to partial offloading, the processing latency of each task includes two parts, one is the sum of the offloading and computation latency from TV to the VEC server, the other is the sum of the offloading and computation latency from TV to CV. Thus, the processing latency of m-th task is the larger one between the two parts, which is given by

$$T_m^{tot} = \max\{T_m^{ser-o} + T_m^{ser-c}, T_{m,n}^{veh-o} + T_{m,n}^{veh-c}\} \tag{9}$$

We aim to minimize the total processing latency of all tasks, by jointly optimizing packet transmit frequency of PC5 interface $\boldsymbol{\lambda}$, VEC server frequency $\boldsymbol{f^{ser}}$, CPU frequency $\boldsymbol{f^{veh}}$ of CVs, and offloading ratio matrix $\boldsymbol{\alpha}$. Since tasks of TV are processed in serial, the total latency is the sum of latency of all tasks for TV. The system latency minimization problem is formulated as follows:

$$\min_{\boldsymbol{\lambda}, \boldsymbol{f^{ser}}, \boldsymbol{f^{veh}}, \boldsymbol{\alpha}} \quad \sum_{m=1}^{M} T_m^{tot} \tag{10}$$

$$\text{s.t.} \quad L_m\lambda \leq C^{v2v,\max} \tag{10a}$$

$$\lambda \in (0, \lambda^{max}] \tag{10b}$$

$$\sum_{m=1}^{M} f_m^{ser} \leq f_{ser}^{max} \tag{10c}$$

$$\sum_{m=1}^{M} \mathcal{I}(\alpha_{m,n}) f_{m,n}^{veh} \leq f_n^{veh,max} \tag{10d}$$

$$\alpha_{m,n}, \alpha_{m,N+1} \in [0,1], \forall n \in \mathcal{N} \tag{10e}$$

$$\alpha_{m,N+1} + \sum_{n=1}^{N} \alpha_{m,n} = 1, \forall n \in \mathcal{N} \tag{10f}$$

$$\sum_{n=1}^{N} \mathcal{I}\left(\alpha_{m,n}\right) = 1 \tag{10g}$$

$$f_m^{ser}, f_{m,n}^{veh} \geq 0 \tag{10h}$$

where $\lambda^{max} = N_{frame}$ is the maximum packet transmit frequency of PC5 interface, f_{ser}^{max} and $f_n^{veh,max}$ are maximum computation frequency of VEC server and n-th CV, respectively. Constraints (10a) and (10b) denote the packet transmit frequency constraints. Constraints (10c) and (10d) are the maximum frequency constraints of VEC server and n-th CV, respectively. Constraint (10e) indicates the offloading ratio belongs to the range of 0-1. Constraint (10f) indicates the sum of the ratio of the m-th task offloaded to the CV or VEC server is equal to one. Constraint (10g) ensures a part of m-th task is only offloaded to a CV.

The problem (10) is challenging to solve due to the following reasons. First, the form of the objective function includes the sum of multiple max functions due to the serial nature of tasks. Second, the optimization variable $\boldsymbol{\alpha}$ is coupled with other variables. Finally, different from traditional offloading, the characteristics of C-V2X, i.e., successful transmission probability, brings challenges to the design of offloading strategy.

IV. Joint Offloading and Resource Allocation Strategy

In this section, since problem (10) is non-convex, we transform the problem (10) into an equivalent problem. Then the problem is decomposed into resource allocation subproblem and offloading strategy subproblem and solved separately.

A. Problem Transformation

Since the objective is non-convex, it is difficult to solve, we transform the objective function by following Proposition.

Proposition 1: When $T_m^{ser-o}+T_m^{ser-c}=T_{m,n}^{veh-o}+T_{m,n}^{veh-c}$, the optimal solution $\boldsymbol{\lambda}^*, \boldsymbol{f^{ser,*}}, \boldsymbol{f^{veh,*}}, \boldsymbol{\alpha}^*$ can be obtained. The objective function of (10) can be transformed into

$$\sum_{m=1}^{M} T_{m,n}^{veh-o}+T_{m,n}^{veh-c} \tag{11}$$

Proof: The proof is similar to that of Proposition 2 in [9], and we omit it due to space limitation. □

Thus problem (10) can be equivalently transformed as

$$\begin{aligned} \min_{\boldsymbol{\lambda},\boldsymbol{f^{ser}},\boldsymbol{f^{veh}},\boldsymbol{\alpha}} \quad & \sum_{m=1}^{M} T_{m,n}^{veh-o}+T_{m,n}^{veh-c} \\ \text{s.t.} \quad & (10a),(10b),(10c),(10d),(10e),(10f) \\ & (10g),(10h) \end{aligned} \tag{12}$$

Since variable coupling exists, problem (12) is non-convex and challenging to solve. Thus we decouple the problem into resource allocation subproblem and offloading strategy subproblem, which decouple the resource allocation variables $\boldsymbol{\lambda}, \boldsymbol{f^{ser}}, \boldsymbol{f^{veh}}$ and offloading ratio matrix $\boldsymbol{\alpha}$.

B. Optimal Computation Resource Allocation

The resource allocation subproblem can be expressed as

$$\begin{aligned} \min_{\boldsymbol{\lambda},\boldsymbol{f^{ser}},\boldsymbol{f^{veh}}} \quad & \sum_{m=1}^{M} T_{m,n}^{veh-o}+T_{m,n}^{veh-c} \\ \text{s.t.} \quad & (10a),(10b),(10c),(10d),(10h) \end{aligned} \tag{13}$$

According to the convex optimization theory, Problem (13) is a convex with respect to $(\boldsymbol{\lambda}, \boldsymbol{f^{ser}}, \boldsymbol{f^{veh}})$, hence it can be solved by the KKT condition [17]. The partial Lagrangian of problem (13) can be given as

$$\begin{aligned} \mathcal{L}(\boldsymbol{\lambda},\boldsymbol{f^{ser}},\boldsymbol{f^{veh}},\boldsymbol{\eta},\boldsymbol{\zeta},\boldsymbol{\psi}) &= \sum_{m=1}^{M} \frac{\alpha_{m,n}L_m}{P_{STP}L_m\lambda}+\frac{\alpha_{m,n}L_mC_m}{f_{m,n}^{veh}} \\ &+\eta\left(L_m\lambda-C^{v2v,\max}\right)+\zeta\left(\sum_{m=1}^{M} f_m^{ser}-f_{ser}^{max}\right) \\ &+\sum_{n=1}^{N}\psi_n\left(\sum_{m=1}^{M}\mathcal{I}(\alpha_{m,n})f_{m,n}^{veh}-f_n^{veh,max}\right) \end{aligned} \tag{14}$$

By using the KKT conditions, the optimal computation resource allocation is derived in the following theorem.

Theorem 1: The optimal CPU computation frequency of the VEC server and of CVs is given by

$$f_m^{ser,*}=\begin{cases} \min\left\{\sqrt{\frac{\alpha_{m,N+1}L_mC_m}{\zeta}}, f_{ser}^{max}\right\}, & \alpha_{m,N+1}>0 \\ 0, & \alpha_{m,N+1}=0 \end{cases} \tag{15}$$

$$f_{m,n}^{veh,*}=\begin{cases} \min\left\{\sqrt{\frac{\alpha_{m,n}L_mC_m}{\psi_n}}, f_n^{veh,max}\right\}, & \mathcal{I}(\alpha_{m,n})=1 \\ 0, & \mathcal{I}(\alpha_{m,n})=0 \end{cases} \tag{16}$$

where $\forall n \in \mathcal{N}$.

We iteratively update the dual variables $(\boldsymbol{\eta},\boldsymbol{\zeta},\boldsymbol{\psi})$ with the optimal solution $(\boldsymbol{\lambda}^*, \boldsymbol{f^{ser,*}}, \boldsymbol{f^{veh,*}})$, the previous iterate $\eta^{(k)}$, $\zeta^{(k)}$, $\psi_n^{(k)}$, and the current iterate $\eta^{(k+1)}$, $\zeta^{(k+1)}$, $\psi_n^{(k+1)}$.

C. Optimal Packet Transmit Frequency of PC5 Interface

The optimization of packet transmit frequency of PC5 interface can be decoupled from Problem (13), since $\boldsymbol{\lambda}$ is independent when $\boldsymbol{\alpha}$ is fixed. Thus we can obtain the following subproblem

$$\begin{aligned} \min_{\boldsymbol{\lambda}} \quad & \sum_{m=1}^{M} \frac{\alpha_{m,n}L_m}{r_{m,n}^{\text{PC5}}} \\ \text{s.t.} \quad & (10a),(10b) \end{aligned} \tag{17}$$

The range of λ can be determined to be $\lambda \in (0, \lambda^{cons}]$, where $\lambda^{cons}=\min\{\lambda^{max}, \frac{C^{v2v,\max}}{L_m}\}$. For convenience, we define the objective function of (17) as $y(\lambda)$, i.e., $y(\lambda)=\sum_{m=1}^{M}\frac{\alpha_{m,n}L_m}{P_{STP}L_m\lambda}$. The second derivative of the function $y(\lambda)$ satisfies $\frac{\partial^2 y(\lambda)}{\partial\lambda^2}>0$. Besides, the constraints (10a), (10b) are all linear constraints, thus the Problem (17) is a convex problem according to the Second-order condition. By calculating $\frac{\partial y(\lambda)}{\partial\lambda}=0$, we have $\lambda^*=\frac{N_{frame}}{2}$. Considering the constraints of λ, i.e., $\lambda \in (0,\lambda^{cons}]$, the λ^* can be obtain when $\frac{N_{frame}}{2}\le\lambda^{cons}$. On the other hand, when $\frac{N_{frame}}{2}>\lambda^{cons}$, $\lambda^*=\frac{N_{frame}}{2}$ cannot be reached. We have $\frac{\partial y(\lambda)}{\partial\lambda}<0$, thus the objective function (17) is monotonically decreasing in the $(0,\lambda^{cons}]$. In this case, the optimal λ^* can be given by $\lambda^*=\lambda^{cons}$. Therefore, the optimal λ^* can be derived as following theorem.

Theorem 2: The optimal packet transmit frequency of PC5 interface is given by

$$\lambda^* = \begin{cases} \frac{N_{frame}}{2}, & \frac{N_{frame}}{2} \le \lambda^{cons} \\ \lambda^{cons}, & \frac{N_{frame}}{2} > \lambda^{cons} \end{cases} \tag{18}$$

where $\lambda^{cons} = \min\{\lambda^{max}, \frac{C^{v2v,\max}}{L_m}\}$.

D. Offloading Strategy Subproblem

The offloading strategy subproblem can be expressed as

$$\min_{\boldsymbol{\alpha}} \quad \sum_{m=1}^{M} T_{m,n}^{veh-o} + T_{m,n}^{veh-c} \tag{19}$$
$$\text{s.t.} \quad (10e), (10f), (10g)$$

By Proposition 1, when $T_m^{ser-o} + T_m^{ser-c} = T_{m,n}^{veh-o} + T_{m,n}^{veh-c}$, the optimal $\boldsymbol{\alpha}^*$ can be obtained. According to Proposition 1, by calculating $T_m^{ser-o} + T_m^{ser-c} = T_{m,n}^{veh-o} + T_{m,n}^{veh-c}$, we can obtain the optimal relational value of offloading ratio matrix as follows.

Theorem 3: The optimal relational value of offloading ratio matrix is given by

$$\alpha_{m,n} = \frac{1}{\Gamma + 1}$$
$$\alpha_{m,N+1} = \frac{\Gamma}{\Gamma + 1} \tag{20}$$

where $\Gamma = \frac{\frac{1}{P_{STP} L_m \lambda^*} + \frac{C_m}{f_{m,n}^{veh,*}}}{\frac{1}{B_{m,N+1} \log_2\left(1 + \frac{p_{m,N+1}^{Uu} g_{m,N+1}}{\sigma^2}\right)} + \frac{C_m}{f_m^{ser,*}}}$.

Based on the above explanation and analysis, the PC5-GO Algorithm algorithm is shown in Algorithm 1.

Algorithm 1 PC5 Interface based Greedy Offloading Algorithm

Require: initial value $\boldsymbol{\alpha}^{(0)}$, $\eta^{(0)}$, $\zeta^{(0)}$ and $\psi_n^{(0)}$ and candidate set $\mathcal{S}^{cloudlet} = \emptyset$.

```
repeat
    Update λ^(k), f^{ser,(k)} and f^{veh,(k)} by calculating (18), (15) and (16).
    Update dual variables η^(k), ζ^(k), and ψ_n^(k).
    Define the criteria Ω = P_STP · f_n^{veh,max}.
    for m = 1 to M do
        The CV selected to compute the m-th task is ω_i = arg max_{i∈N} Ω.
        Assign the index i to candidate set S^cloudlet.
        Update the Ω ← Ω \ Ω_{i∈S^cloudlet}.
        Update α^(k)_{m,i∈S^cloudlet} ⇐ 1,
        Update α^(k)_{m,n∈{N\i}} ⇐ 0.
        Update relational value of offloading ratio matrix by calculating (20).
    end for
    Update k + 1 ← k.
until |T^(k) − T^(k−1)| ≤ ϵ
return T and S^cloudlet
```

Parameters	Values
Number of tasks of TV M	20
Number of CVs in multi-tier VEC N	5
Mumber of lanes	4
Maximum vehicle density per lane β	0.8×84^{-1}
Number of RBs between RSU and TV	20
Number of sub-frames in 1 second N_{frame}	1000
Number of sub-frames in the SW N_{sub-f}	100
Number of sub-channels in the SW N_{sub-c}	4
Number of RBs per sub-channel	12
Sensing distance of TV d_{sen}	1 km
Maximum transmission power of TV P^{max}	23 dBm
The data size of m-th task L_m	$U(1,2) \times 10^4$ bits
Computation load of m-th task C_m	$U(500, 1000)$ cycle/bit
Maximal frequency of VEC server f_{ser}^{max}	2×10^9 cycle/s
Maximal frequency of CV $f_n^{veh,max}$	$U(1,2) \times 10^9$ cycle/s

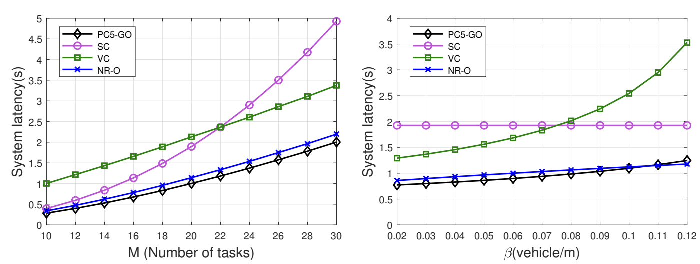
V. Performance Evaluation

A. Simulation Setup

We consider a vehicular computing networking with a single RSU and multiple vehicles, which includes a TV and N CVs. The pathloss model is $PL = 38.77 + 16.7\log_{10}(d) + 18.2\log_{10}(f_c)$ for NR V2X Sindelink(PC5 interface), where f_c denotes the carrier frequency in GHz and d denotes the Euclidean distance between a TX and a RX in 3D space in meters. Besides, the other simulation parameters are summarized in Table I [14]. For comparison, the benchmark schemes are set as follows.

- VEC server computing (SC): All tasks are offloaded to the VEC server by Uu interface for computation.
- Cloudlet vehicles computing (VC): All tasks are offloaded to CVs by PC5 interface for computation.
- NR-based offloading (NR-O) [12]: considering the traditional orthogonal channel allocation, one part of each task is offloaded to the RSU and the other part is offloaded to the CV.

B. Simulation Results



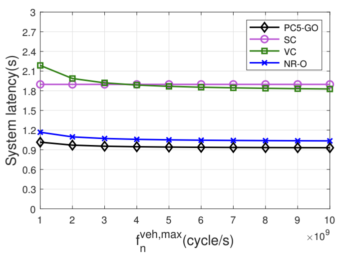
(a) Number of tasks (b) Vehicle density

Fig. 2. Impact of (a) the number of tasks and (b) the vehicle density on system latency.

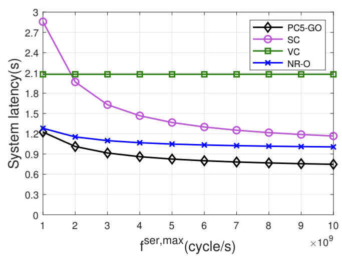
Fig. 2(a) shows the comparison of system latency with different number of tasks M. It can be observed that PC5-GO algorithm can achieve lower latency compared with other schemes, i.e., SC, VC, and NR-O. With the increase of M, the latency of all schemes increases, in which SC increases faster than VC. It is because the increase of the number of tasks brings a heavy burden to the centralized processing of

VEC server, but has relatively little influence on the distributed computing of CVs.

To investigate the relationship between the system latency and vehicle density β, Fig. 2(b) shows the system latency under different benchmark schemes, i.e., SC, VC, and NR-O. It is observed that the system latency of the PC5-GO and NR-O algorithms achieve lower system latency than SC and VC schemes, which shows the benefit of partial offloading. Besides, as the vehicle density increases, the system latency of the PC5-GO, NR-O, VC algorithms also increases correspondingly. Moreover, when the vehicle density is greater than 0.11, the latency of PC5-GO is larger than that of NR-O. It is because that the $\mathcal{P}_{STP}$ decreases with the increase of vehicle density.



(a) Vehicular frequency



(b) VEC frequency

Fig. 3. Impact of (a) computation frequency of cloudlet vehicles and (b) VEC server computation frequency on system latency.

Fig. 3(a) shows the comparison of system latency with respect to computation frequency of CVs. We consider that the computation frequency of all CVs is the same. Note that the system latency of PC5-GO algorithm is lower than SC, VC, and NR-O. The system latency of SC remains unchanged as $f_n^{veh,max}$ changes, which reason is that the SC only utilizes VEC server computation frequency. Moreover, the latency of the PC5-GO, NR-O, and VC decreases with the increasing of computation frequency of CVs.

Fig. 3(b) shows the system latency with VEC server computation frequency of different benchmark schemes. Note that the PC5-GO algorithm always realizes lower system latency than SC, VC, and NR-O. Besides, the system latency of VC does not change with the change of VEC server computation frequency. As the VEC server computation frequency increases, the system latency of the PC5-GO, NR-O, and SC are reduced accordingly. Moreover, with the increase of computation frequency, the latency of PC5-GO, NR-O, and SC tends to be stable, which is due to the limitation of wireless resources.

VI. Conclusion

In this paper, we have proposed a novel offloading strategy based C-V2X in the multi-tier vehicular edge computing system. The communication model of Uu and PC5 interface has been formulated and successful transmission probability is studied. The system latency minimization problem of task processing has been formulated by optimizing the offloading matrix and packet transmit frequency of PC5 interface, and the computation frequency of vehicles and VEC server. The problem can be solved by decomposing subproblems, i.e., resource allocation and offloading strategy subproblems, to solve non-convexity and variable coupling. The PC5-GO algorithm has been proposed to obtain the optimal communication and computation resource allocation, and the offloading strategy. The simulation results have shown that our PC5-GO algorithm can effectively reduce the system latency compared with the other benchmark schemes by 5.88% at least.

References

[1] N. Zhang, S. Zhang, P. Yang, O. Alhussein, W. Zhuang, and X. S. Shen, "Software defined space-air-ground integrated vehicular networks: Challenges and solutions," *IEEE Communications Magazine*, vol. 55, no. 7, pp. 101–109, 2017.

[2] S. Li, N. Zhang, S. Lin, L. Kong, A. Katangur, M. K. Khan, M. Ni, and G. Zhu, "Joint admission control and resource allocation in edge computing for Internet of Things," *IEEE Network*, vol. 32, no. 1, pp. 72–79, 2018.

[3] Q. Pan, S. Lin, W. Lu, Z. Wang, L. Wu, Y. Zou, B. Ai, and Z. Zhong, "Space-Air-Sea-Ground integrated monitoring network-based maritime transportation emergency forecasting," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 3, pp. 2843–2852, 2022.

[4] W. Feng, N. Zhang, S. Lin, S. Li, Z. Wang, B. Ai, and Z. Zhong, "Energy efficient collaborative offloading in NOMA-enabled fog computing for internet of things," *IEEE Internet of Things Journal*, pp. 1–1, 2022.

[5] K. Wang, W. Chen, J. Li, Y. Yang, and L. Hanzo, "Joint task offloading and caching for massive MIMO-aided multi-tier computing networks," *IEEE Transactions on Communications*, pp. 1–1, 2022.

[6] S. Li, S. Lin, L. Cai, W. Li, and G. Zhu, "Joint resource allocation and computation offloading with time-varying fading channel in vehicular edge computing," *IEEE Transactions on Vehicular Technology*, vol. 69, no. 3, pp. 3384–3398, 2020.

[7] Q. Gao, S. Lin, and G. Zhu, "Joint vehicular and static users multiplexing transmission with hierarchical modulation for throughput maximization in vehicular networks," *IEEE Transactions on Intelligent Transportation Systems*, vol. 21, no. 9, pp. 3835–3847, 2020.

[8] P. Dai, K. Hu, X. Wu, H. Xing, F. Teng, and Z. Yu, "A probabilistic approach for cooperative computation offloading in MEC-assisted vehicular networks," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 2, pp. 899–911, 2022.

[9] W. Feng, N. Zhang, S. Li, S. Lin, R. Ning, S. Yang, and Y. Gao, "Latency minimization of reverse offloading in vehicular edge computing," *IEEE Transactions on Vehicular Technology*, pp. 1–1, 2022.

[10] Y. Wang, P. Lang, D. Tian, J. Zhou, X. Duan, Y. Cao, and D. Zhao, "A game-based computation offloading method in vehicular multiaccess edge computing networks," *IEEE Internet of Things Journal*, vol. 7, no. 6, pp. 4987–4996, 2020.

[11] S. M. A. Kazmi, S. Otoum, R. Hussain, and H. T. Mouftah, "A novel deep reinforcement learning-based approach for task-offloading in vehicular networks," in *2021 IEEE Global Communications Conference (GLOBECOM)*, 2021, pp. 1–6.

[12] Y. Dai, D. Xu, S. Maharjan, and Y. Zhang, "Joint load balancing and offloading in vehicular edge computing and networks," *IEEE Internet Things J.*, vol. 6, no. 3, pp. 4377–4387, 2019.

[13] 3rd Generation Partnership Project, "Release 14: Technical specifications group services and system aspects," *3GPP TR 21.914*, 2017.

[14] M. H. C. Garcia, A. Molina-Galan, M. Boban, J. Gozalvez, B. Coll-Perales, T. Şahin, and A. Kousaridas, "A tutorial on 5G NR V2X communications," *IEEE Communications Surveys Tutorials*, vol. 23, no. 3, pp. 1972–2026, 2021.

[15] Z. Kuang, L. Li, J. Gao, L. Zhao, and A. Liu, "Partial offloading scheduling and power allocation for mobile edge computing systems," *IEEE Internet of Things Journal*, vol. 6, no. 4, pp. 6774–6785, 2019.

[16] M. Gonzalez-Martín, M. Sepulcre, R. Molina-Masegosa, and J. Gozalvez, "Analytical models of the performance of C-V2X Mode 4 vehicular communications," *IEEE Transactions on Vehicular Technology*, vol. 68, no. 2, pp. 1155–1166, 2019.

[17] C.-Y. Chi, W.-C. Li, and C.-H. Lin, *Convex optimization for signal processing and communications: from fundamentals to applications*. CRC press, 2017.