Intelligent Task Offloading for Heterogeneous V2X Communications

Kai Xiong, Supeng Leng, *Member, IEEE*, Chongwen Huang, *Member, IEEE*, Chau Yuen, *Senior Member, IEEE*, and Yong Liang Guan, *Senior Member, IEEE*

***Abstract*— With the rapid development of autonomous driving technologies, it becomes difficult to reconcile the conflict between ever-increasing demands for high process rate in the intelligent automotive tasks and resource-constrained on-board processors. Fortunately, vehicular edge computing (VEC) has been proposed to meet the pressing resource demands. Due to the delay-sensitive traits of automotive tasks, only a heterogeneous vehicular network with multiple access technologies may be able to handle these demanding challenges. In this article, we propose an intelligent task offloading framework in heterogeneous vehicular networks with three Vehicle-to-Everything (V2X) communication technologies, namely Dedicated Short Range Communication (DSRC), cellular-based V2X (C-V2X) communication, and millimeter wave (mmWave) communication. Based on stochastic network calculus, this article firstly derives the delay upper bounds of different offloading technologies with certain failure probabilities. Moreover, we propose a federated Q-learning method that optimally utilizes the available resources to minimize the communication/computing budgets and the offloading failure probabilities. Simulation results indicate that our proposed algorithm can significantly outperform the existing algorithms in terms of resource cost and offloading failure probability.**

***Index Terms*— Vehicular edge computing, DSRC, C-V2X, mmWave, federated Q-learning.**

I. Introduction

MODERN transportation systems have evolved vehicles with the automotive artificial intelligence which can make decisions through on-board processors and shares road information by Vehicle-to-Everything (V2X) communications. The emergence of services like intelligent cruise scheduling, road traffic management, and cooperative driving with the shared on-board computing resources, has prompted the need for significant computing capacity and rigorous delay tolerance [1]. Previous literature [2] remarked that an autonomous vehicle could generate 1 GB of data per second from the on-board processor. Additionally, the volume of the data will exponentially grow with the increasing number of autonomous vehicles. Without enough communication and computing resources, road risk estimation becomes leggy and inaccurate that may incur life-threatening problems.

Due to the constrained on-board processing power and communication bandwidth in vehicular networks, it is impractical to raise the computing and communication capacity of an individual vehicle for automotive applications. To cope with the limited computing capability problem, Vehicular Edge Computing (VEC) technology has been proposed to support the computing-thirsty tasks in the intelligent transportation systems, in which the computing capability of a vehicle could be improved through offloading the overburden tasks to its adjacent vehicles or edge servers [3]. Previous work [4], [5] mainly pays attention to the task offloading schemes with sufficient communication resources. However, the performance of VEC is significantly constrained by the limited communication bandwidth in a practical vehicular network.

On the other hand, the use of heterogeneous V2X communications has great potential to enhance the communication capability for the large scale application of autonomous vehicles [6]. Vehicle-to-Vehicle (V2V) communications are the major connection form in the platoon-based edge computing paradigm, where a platoon of vehicles with sufficient on-board computing resources could provide additional mobile edge computing in the vicinity [7]. Specifically, there are three widely used types of V2V communication technologies, namely Dedicated Short Range Communications (DSRC) communication, cellular-based V2V (C-V2V) communication, and mmWave communication. Moreover, the edge servers in the roadside (e.g., cellular base stations) can also offer computing resource upon Vehicle-to-Infrastructure (V2I) communication, namely the infrastructure-based edge computing paradigm [8]. In this article, the cellular-based V2I (C-V2I) requires vehicles to stay within cellular coverage. Besides, DSRC and mmWave V2V operate on the license free bands [9], [10]. However, C-V2X (C-V2V and C-V2I) works on the licensed band to provide paid communication service [11].

In the context of VEC with heterogeneous V2X connections and computing servers, the offloading reliability is tightly affected by the selection of access technologies and offloading

Manuscript received March 31, 2020; revised June 29, 2020 and July 25, 2020; accepted July 31, 2020. This work was supported in part by the National Key Research and Development Program of China under Grant 2018YFE0117500; in part by the Science and Technology Program of Sichuan Province, China under Grant 2019YFG0520; in part by the EU H2020 Project COSAFE under Grant MSCA-RISE-2018-824019; in part by the China Scholarship Council; and in part by the A*STAR under its RIE2020 Advanced Manufacturing and Engineering (AME) Industry Alignment Fund-Pre Positioning (IAF-PP) under Grant A19D6a0053. The Associate Editor for this article was Y. Zhang. *(Corresponding author: Supeng Leng.)*

Kai Xiong and Supeng Leng are with the School of Information and Communication Engineering, University of Electronic Science and Technology of China (UESTC), Chengdu 611731, China (e-mail: spleng@uestc.edu.cn).

Chongwen Huang and Chau Yuen are with the Engineering Product Development, Singapore University of Technology and Design (SUTD), Singapore 487372.

Yong Liang Guan is with the School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore 639798.

Digital Object Identifier 10.1109/TITS.2020.3015210

targets. Malfunctions of any part (transmission may fail or task processing may be interrupted) deteriorate the VEC performance. However, no previous work has investigated the integrating communication and computing failure probability in the heterogeneous VEC furthermore incorporated this failure probability into the performance optimization objective. Therefore, with the constraints of the offloading failure probability and the cost of computing and communication, it is still an open challenge to attain reliable and economical VEC services in a heterogeneous vehicular network.

To fulfill these research gaps, our study exploits the heterogeneous VEC to optimize tasks offloading with multiple V2X technologies. First, we derive the upper bound of the task offloading delay with variable failure probability. Based on the analysis of the upper bounds, a federated learning-based intelligent offloading scheme is proposed to minimize the failure probability and the resource cost (communication and computing cost). Our federated Q-learning algorithm can parallelly exploit and share the local knowledge among vehicles. The main contributions are summarized as follows:

- We derive the upper bound of offloading delay for different V2X technologies by leveraging stochastic network calculus. This article gives a closed-form for the upper bound of offloading delay in VEC. These upper bounds can guide the design of optimal offloading scheduling.
- We propose a new optimization model taking accounts of the communication and computing budgets as well as the failure probability. This is the first work to consider the offloading failure probability into the heterogeneous VEC. Simulations show that the cellular-based V2X (C-V2X) communication has the best performance in terms of the failure probability under different traffic loads.
- We design a federated learning-based parallel scheme with high scalability and fast convergence. The optimization model is decoupled and parallelly trained by a set of local Q-learning processes, which accumulate global knowledge by exploiting diverse local action-state spaces, simultaneously. The proposed consensus Q-Table can suppress the heavy communication overhead of knowledge sharing.

The remainder of this article is organized as follows. Section II presents the related works. Section III introduces the system model. Section IV provides the platoon-based edge computing and infrastructure-based edge computing formations. Section V presents the optimization model and our two solutions on the V2X offloading selection. Section VI demonstrates the simulation results and the performance discussion. Finally, we draw the conclusion in Section VII.

II. Related Work

Existing work on VEC focuses on selecting the optimal offloading targets taking accounts of resource constraints. Dai *et al.* [4] investigated the multiple tasks offloading problem of VEC and proposed a two-step iterative algorithm to optimize the offloading ratio and computation assignment. Zhang *et al.* [12] proposed a Stackelberg game approach for the offloading candidates selection to improve the utilities of both vehicles and VEC servers. Zhang *et al.* [8] regarded vehicles as caching servers and proposed a caching service migration scheme, where the communication, computing, and caching resources at the network edge are jointly scheduled. However, these VEC solutions ignored the potential selection problem of the heterogeneous V2X access technologies for task offloading.

Selecting suitable V2X access technologies for different tasks offloading is reasonable since each V2X technology has its own limitations. DSRC is a typical competition-based communication that is not adequate for multiple tasks communication due to scalability issues [13]. The C-V2X technology eliminates the above drawbacks, while it charges the fee to vehicles [14]. In contrast, the mmWave communication works at the high-frequency unlicensed bands with large available bandwidths [6]. However, it needs antenna beam alignment that incurs additional link budget [15]. Consequently, the academic society has been exploited integrating V2X architecture to complement the shortcomings of individual V2X technology. Abboud *et al.* [16] discussed the interworking issue of DSRC and cellular communication solutions. Coll-Perales *et al.* [17] proposed a heterogeneous DSRC and mmWave vehicular network, which used side information of DSRC channel to speed up mmWave beam alignment. Prior work of Katsaros *et al.* [18] developed stochastic network calculus to derive the stochastic upper bound of the end-to-end delay for the DSRC combining with C-V2X communications.

However, to our best knowledge, there is no literature discussed the interworking of DSRC, C-V2X, and mmWave communications in the heterogeneous VEC system. In addition, existing work on VEC devoted to the computation offloading between the vehicle and the VEC servers. They ignored the offloading failure probability due to the communication and computing issues. To fill the research gap, our work attempts to addresses the problem of tasks offloading in the heterogeneous VEC taking accounts of the offloading failure probability and resource cost. This is a challenging problem due to the complicated dependency between multiple access technologies and offloading targets.

III. System Model

As shown in Fig. 1, the proposed VEC framework consists of vehicles, cellular base stations (BSs), and VEC servers, where VEC servers are located with the corresponding BSs at the roadside to reduce the end-to-end offloading latency. Since the C-V2I outperforms than other V2I technologies in terms of the wireless coverage and enhanced reliability [13], this article only regards the cellular BS as the roadside infrastructure. Besides, VEC servers can connect with each other through the X2 interfaces of the accompanied BSs to form a resource pool (VEC pool). The logical connection between the VEC pool and the connected VEC servers is denoted by the virtual link. In this article, the centralized resource management is implemented in the VEC pool that provides efficient resource utilization to the C-V2I offloading process. However, the resource management of V2V offloading is achieved by a resource-rich vehicle that is elected in its platoon, named platoon header [3].

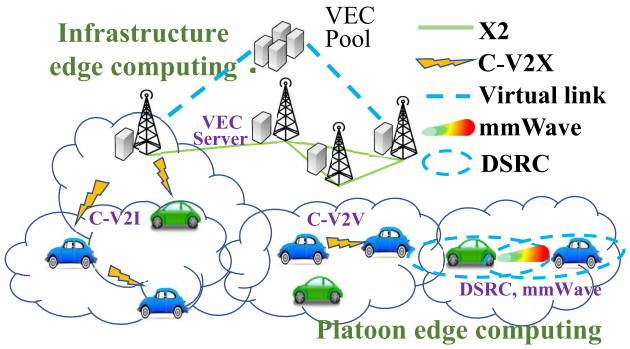


Fig. 1. Illustration of the heterogeneous VEC framework.

Furthermore, we propose the Synchronous Federated Q-Learning (Sync-FQL) algorithm that instructs the offloading way and direction of the network traffic taking accounts of offloading failure probability and resource cost. This algorithm is deployed in the VEC pool or platoon header. As shown in Fig. 2, the input of Sync-FQL algorithm includes delay requirements T_{max} of each task, traffic attributes (arrival rate λ and brustiness measure o) of each tele-traffic, as well as the service attributes (envelope service rate ξ and peak service capacity η) of each server [19]. Noted that we treat V2X communication and computing as a service.

Since the V2X technologies and the offloading targets (vehicles or VEC servers) are tightly coupled in series, it is a typically studied object of the stochastic network calculus theory. Recent advances in network performance researches have adopted network calculus to estimate the end-to-end delay [18], [20]. However, these previous works did not investigate the stochastic network calculus in the VEC system from the perspective of the offloading failure probability as well as communication and computing resources. In this article, we investigate the stochastic network calculus to obtain the offloading delay and failure probability, which are used to design the offloading scheduling.

In addition, offloading scheduling is impacted by the properties of the task tele-traffic. From the perspective of the network calculus, an arrival traffic curve $X(t)$ can be modelled as the cumulative volume of the input traffic during interval $(0, t]$. And, $X(\tau, t) = X(t-\tau) = X(t) - X(\tau)$ is the cumulative volume of the input traffic during $(\tau, t]$, where $\tau < t$. Moreover, $X(\tau, t)$ has a statistical envelope (λ, o), referred to the exponentially bounded burstiness model [19], which provides a validated inequality $P[X_i(t-\tau) \geq \lambda_i \cdot (t-\tau) + o_i] \leq \varepsilon$, where λ_i is an arrival rate of task i. o_i is the burstiness measure of task i [19], and ε is the violated probability. In virtue of Chernoff's bound [19], we get

$$P[X_i(t-\tau) \geq \lambda_i \cdot (t-\tau) + o_i] \leq e^{-\theta(\lambda_i \cdot (t-\tau)+o_i)} \mathrm{E}\left[e^{\theta X_i}\right], \tag{1}$$

where $\mathrm{E}(x)$ is the expectation of random variable x. θ is a constant parameter. We assume that there are K categories automotive tasks $\{1, \ldots, K\}$ in the transportation system, such as the vehicular Internet and infotainment, cooperative lane change assist, and cooperative adaptive cruise control [21]. Each task i generates its network traffic that is characterized by the arrival rate λ_i and brustiness measure o_i. Furthermore, the task can be divided into several parts and processed separately.

Hereafter, we investigate the dynamic service curve $Y(\tau, t)$, $t \geq \tau \geq 0$, which can be provided by a channel or a processor [22]. $Y(\tau, t)$ is non-negative and increased with time t. A dynamic service envelope is defined as $P[Y(t-\tau) \geq \xi \cdot (t-\tau) - \eta] \leq \varepsilon$ for all $t \geq \tau \geq 0$, where ξ is the envelope service rate and η is the peak service capacity of the server [19]. According to Chernoff's bound, the envelope is given as

$$P[Y(t-\tau) \leq \xi \cdot (t-\tau) - \eta] \leq e^{\theta(\xi \cdot (t-\tau)-\eta)} \mathrm{E}\left[e^{-\theta Y}\right]. \tag{2}$$

When the dynamic server $Y(s, t)$ deals with the input traffic $X(s, t)$, the system delay $d(\varepsilon)$ with a failure probability $\varepsilon(\eta)$ satisfies the following inequality [23]

$$\mathbb{P}\left[(X \oslash Y_{\text{net}})(t + d(\varepsilon), t) \geq 0\right] \leq \varepsilon(\eta), \tag{3}$$

where $X \oslash Y(s, t) = \sup_{\tau \in [0, s]}[X(\tau, t) - Y(\tau, s)]$.

IV. Performance Analysis on VEC

In this section, we derive the end-to-end delay upper bound of different networks in the platoon edge computing and infrastructure edge computing, respectively.

A. Platoon Edge Computing

Vehicles can form a platoon to share their computing resource with the surrounding vehicles for cooperatively processing. There are three kinds of communication technologies to support the platoon-based edge computing that is shown in Fig. 2. The C-V2V communication operates in a licensed band and needs the vehicles to pay a fee to the mobile network operator, while DSRC and mmWave are free to access.

1) Delay Upper Bound of DSRC: The DSRC standard is based on the 802.11p amendment to the IEEE 802.11 standard whose performance is governed by the exponential back-off algorithm. And, prior work of Cho and Jiang [24] proved that the summation of the backoff values generated per packet w^B has a Pareto-type tail with an exponent of $-\alpha$, formally $w^B \approx x^{-\alpha}$. Usually, the backoff value is rising with the number of active users while decreasing with the available DSRC bandwidth R^{dsrc}. To simplify the analysis, we assume $w^B \sim (\frac{\mathscr{V} R^{dsrc}}{N})^{-\alpha}$, where $\mathscr{V}$ is a normalized coefficient. Hence, the access delay of input traffic at the head-of-line is $\hat{t}_{\text{serv}} = w^B t_w + t_{TX}$ [18], where t_w is the average length of a back-off slot. And, $t_{TX} = \frac{o}{R^{dsrc}}$ is the largest length of a transmission slot. Without loss of generality, we assume $t_w = 1$ and $\frac{o}{R^{dsrc}}$ is negligible. Therefore, $\hat{t}_{\text{serv}} \sim u(R^{dsrc})^{-\alpha}$, where u is a constant.

Furthermore, we regard the transmission process of DSRC as a classical latency-rate service model $\beta(t) = R(t-s)^+$ where s is the access delay [22]. Thus, the latency-rate service curve of DSRC is expressed as

$$\beta^{dsrc}(t-\tau) = R^{dsrc}\left((t-\tau) - \hat{t}_{\text{serv}}\right)^+, \tag{4}$$

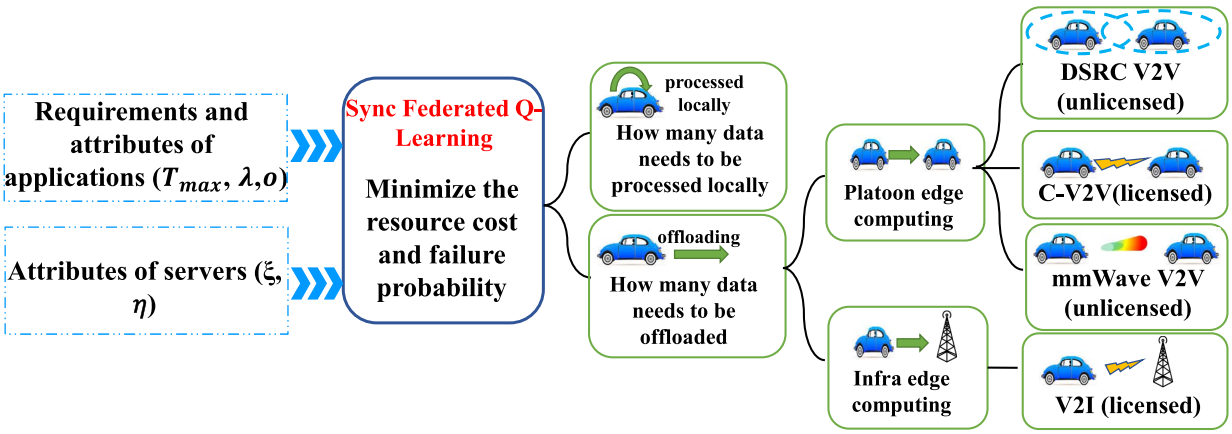


Fig. 2. Demonstration of the offloading scheduling.

where $(X)^+ = 0$ if $X < 0$, and $(X)^+ = X$ otherwise. By virtue of the superposition property [22], the whole input traffic except for task i's traffic can be treated as a background arrival curve $\alpha_i^{bg}(t)$ for task i:

$$\alpha^{bg}(t-\tau) = \sum_{j\neq i}^{K} \varrho_j^{dsrc} \mathscr{O}_j(t-\tau), \tag{5}$$

where ϱ_j^{dsrc} is the percentage of task j transmitted by DSRC. And, $\mathscr{O}_j(t-\tau)$ is the accumulated traffic volume of task j in interval $[\tau, t]$, i.e., $\mathscr{O}_j(t-\tau) = \lambda_j \cdot (t-\tau) + o_j$, where λ_j is an envelope arrival rate of task j; o_j is a burstiness measure of task j [19]. Then, upon the theory of Leftover Service [22], the service curve $S_i^{dsrc}(t-\tau)$ of DSRC for task i is

$$\begin{aligned} S_i^{dsrc}(t-\tau) &= (\beta^{dsrc} - \alpha_i^{bg})^+(t-\tau) \\ &= (R^{dsrc} - \sum_{j\neq i}^{K} \varrho_j^{dsrc}\lambda_j)(t-\tau) \\ &\quad - R^{dsrc}\hat{t}_{\text{serv}} - \sum_{j\neq i}^{K} \varrho_j^{dsrc} o_j. \end{aligned} \tag{6}$$

If task i has offloaded to the target vehicle, it has to compete with other offloaded tasks (i.e., offloaded by mmWave, DSRC, and C-V2V) and the local processed tasks for the on-board CPU cycling (computing resource). Similar to Eq. (5), the background on-board processing curve of task i can be expressed as

$$\phi_i^{bg}(t-\tau) = \frac{\mathscr{O}_j(t-\tau)}{N}\left[\sum_{j=1}^{K}(1-\varrho_j^{v2i}) - \varrho_i^{dsrc}\right], \tag{7}$$

where N is the number of vehicles in the road segment. ϱ_j^x is the percentage of the task j transmitted by x category offloading technology, where x represents one of the DSRC, C-V2I, C-V2V, mmWave, and local processing. Hence, for any task i, the equation $\varrho_i^{local} + \varrho_i^{mmw} + \varrho_i^{dsrc} + \varrho_i^{v2i} + \varrho_i^{cv2v} = 1$ holds. Again, according to the Leftover Service property [22], the on-board processing curve for task i through DSRC offloading is given as

$$\begin{aligned} &\Omega_i^{dsrc}(t-\tau) \\ &= (\Omega^{veh} - \phi_i^{bg})^+(t-\tau) \\ &= \left[\Theta^{veh} - \frac{1}{N}\left(\sum_{j\neq i}^{K}\varrho_j^{dsrc}\lambda_j + \sum_{j=1}^{K}(1-\varrho_j^{dsrc}+\varrho_j^{v2i})\lambda_j\right)\right] \\ &\quad \cdot (t-\tau) - \frac{1}{N}\left(\sum_{j\neq i}^{K}\varrho_j^{dsrc}o_j + \sum_{j=1}^{K}(1-\varrho_j^{dsrc}+\varrho_j^{v2i})o_j\right), \end{aligned} \tag{8}$$

where $\Omega^{veh}(t-\tau) = \Theta^{veh}\cdot(t-\tau)$, and Θ^{veh} is the computing capability of the on-board processor. Furthermore, according to the concatenated property [22], the total service curve Y_i^{dsrc} of DSRC offloading for task i is equal to the convolution of DSRC transmission and the on-board processing.

$$Y_i^{dsrc}(t-\tau) = (S_i^{dsrc} \otimes \Omega_i^{dsrc})(t-\tau), \tag{9}$$

which represents task i traversing the services of DSRC transmission and the on-board processor, where $(x\otimes y)(s,t) = \inf_{\tau\in[s,t]}[x(s,\tau)+y(\tau,t)]$. Hereafter, based on Eq. (3), the validated inequality of DSRC offloading is

$$\begin{aligned} &\mathbb{P}\{X_i \oslash Y_i^{dsrc}(t+d(\varepsilon),t) \geq 0\} \\ &\leq e^{\theta\cdot 0}\mathrm{E}e^{\theta[\sup_{\tau\in[0,t]} X_i(\tau,t) - Y_i^{dsrc}(\tau,t+d(\varepsilon))]} \\ &\leq e^{\theta\varrho_i^{dsrc}o_i}\sum_{\tau\in[0,t]} e^{\theta\lambda_i(t-\tau)}\sum_{\upsilon=\tau}^{t}\mathrm{E}\left[e^{-\theta S_{dsrc}^i(\tau,\upsilon)}\right] \\ &\quad \cdot \mathrm{E}\left[e^{-\theta\Omega_i^{dsrc}(\upsilon,t+d(\varepsilon))}\right] \\ &\leq \frac{e^{\theta\varrho_i^{dsrc}o_i+\theta(\eta_{dsrc}^{comp}+\eta^{dsrc})}e^{-\theta\xi_{dsrc}^{comp}d(\varepsilon)}}{1-e^{-\theta((\xi^{dsrc}-\xi_{dsrc}^{comp}))}}\sum_{\tau=0}^{\infty}e^{-\theta(\xi_{dsrc}^{comp}-\lambda_i)\tau} \\ &= \frac{e^{\theta\varrho_i^{dsrc}o_i+\theta(\eta_{dsrc}^{comp}+\eta^{dsrc})}e^{-\theta\xi_{dsrc}^{comp}d_i}}{(1-e^{-\theta((\xi^{dsrc}-\xi_{dsrc}^{comp}))})(1-e^{-\theta(\xi_{dsrc}^{comp}-\lambda_i)})}, \end{aligned} \tag{10}$$

where $\xi^{dsrc} = R^{dsrc}$, $\eta^{dsrc} = R^{dsrc}\hat{t}_{\text{serv}}$, $\xi_{dsrc}^{comp} = \Theta - \frac{1}{N}[\sum_{j\neq i}^{K}\varrho_j^{dsrc}\lambda_j + \sum_{j=1}^{K}(\varrho_j^{mmw} + \varrho_j^{cv2v} + \varrho_j^{local})\lambda_j]$, and

$\eta_{dsrc}^{comp} = \frac{1}{N}[\sum_{j\neq i}^{K} \varrho_j^{dsrc} o_j + \sum_{j=1}^{K} (\varrho_j^{mmw} + \varrho_j^{cv2v} + \varrho_j^{local}) o_j]$. The first inequality of Eq. (10) is attained by the Chernoff's bound. And, the second inequality is derived by the fact:

$$\mathrm{E}\left[e^{-\theta Y_i^{dsrc}(\tau,t)}\right] = \mathrm{E}\left[e^{-\theta \min_{\upsilon\in[\tau,t]}\{S_i^{dsrc}(\tau,\upsilon)+\Omega_i^{dsrc}(\upsilon,t)\}}\right] \leq \sum_{\upsilon=\tau}^{t} \mathrm{E}\left[e^{-\theta S_i^{dsrc}(\tau,\upsilon)}\right] \mathrm{E}\left[e^{-\theta \Omega_i^{dsrc}(\upsilon,t)}\right]. \tag{11}$$

The third inequality of Eq. (10) is based on the assumption $\mathrm{E}\left[e^{-\theta Y(\tau,t)}\right] \leq e^{-\theta(\xi(t-\tau)-\eta)}$ with $t \to \infty$. The last identity of Eq. (10) is obtained by the infinity geometric sum, where $(\xi^{dsrc} - \xi_{dsrc}^{comp}) > 0$ is assumed for convergence. In addition, the last equation is the offloading failure probability ε_i of task i:

$$\varepsilon_i = \frac{e^{\theta \varrho_i^{dsrc} o_i + \theta(\eta_{dsrc}^{comp} + \eta^{dsrc}) e^{-\theta \xi_{dsrc}^{comp} d_i}}}{(1 - e^{-\theta(\xi^{dsrc} - \xi_{dsrc}^{comp})})(1 - e^{-\theta(\xi_{dsrc}^{comp} - \lambda_i)})}, \tag{12}$$

where ε_i represents the probability that the offloading delay (transmission delay plus the on-board processing delay) exceeds the delay threshold d_i. Through a simple algebraic operation, we get the upper bound of the offloading delay d_i:

$$\begin{aligned} d_i &= \frac{1}{\xi_{dsrc}^{comp}}(\varrho_i^{dsrc} o_i + \eta_{dsrc}^{comp} + \eta^{dsrc}) - \frac{1}{\theta \xi_{dsrc}^{comp}} (\ln \varepsilon + \mathscr{U}) \\ &\leq \frac{1}{\xi_{dsrc}^{comp}}(\varrho_i^{dsrc} o_i + \eta_{dsrc}^{comp} + \eta^{dsrc} - \frac{\ln \varepsilon}{\theta}), \end{aligned} \tag{13}$$

where $\mathscr{U} = \ln(1 - e^{-\theta((\xi^{dsrc} - \xi_{dsrc}^{comp}))}) + \ln(1 - e^{-\theta(\xi_{dsrc}^{comp} - \lambda_i)})$. The inequality of Eq. (13) is because of $\mathscr{U} \leq 0$. Hence, the delay upper bound d_i^{dsrc} of DSRC offloading for task i with the offloading failure probability ε_i is given as

$$d_i^{dsrc} = \frac{\frac{1}{N}\sum_{j=1}^{K}[1 - \varrho_j^{\upsilon 2i} + (N-1)\varrho_j^{local}] o_j + u(R^{dsrc})^{1-\alpha} - \frac{\ln \varepsilon_i}{\theta}}{\Theta^{\upsilon eh} - \frac{1}{N}\sum_{j=1}^{K}[1 - \varrho_j^{\upsilon 2i} + (N-1)\varrho_j^{local}]\lambda_j + \varrho_i^{dsrc}\lambda_i}. \tag{14}$$

2) Delay Upper Bound of C-V2V: C-V2V communication is regarded as a reservation-based V2V technology [25]. Additionally, Release 14 of C-V2X standardization in the Third Generation Partnership Project defines two new modes (mode 3 and mode 4) for C-V2V communication [26]. Due to space limitations, this article focuses on the C-V2V communication upon mode 3. In mode 3, the communication resource for each V2V communication is pre-assigned by the nearby BS, which avoids the access competition in transmission. Hence, we obtain the transmission service curve of C-V2V $\beta_i^{cv2v} = R_i^{cv2v}(t-\tau)$, where R_i^{cv2v} is the reserved bandwidth for task i. Similar to the DSRC offloading, task i also has to compete with other offloaded tasks and the local processed tasks for the on-board CPU cycling. Again, according to the Leftover Service property, the on-board processing service curve of C-V2V offloading is

$$\begin{aligned} &\Omega_i^{cv2v}(t-\tau) \\ &= [\Theta^{\upsilon eh} + \varrho_i^{cv2v}\lambda_i - \frac{1}{N}\sum_{j=1}^{K}(1 - \varrho_j^{\upsilon 2i} + (N-1)\varrho_j^{local})\lambda_j] \\ &\quad \cdot (t-\tau) - \left\{\frac{1}{N}\sum_{j=1}^{K}(1 - \varrho_j^{\upsilon 2i} + (N-1)\varrho_j^{local}) o_j - \varrho_i^{cv2v} o_i\right\}. \end{aligned} \tag{15}$$

Hereafter, by virtue of the concatenated property, the total service curve of C-V2V offloading for task i is $Y_i^{cv2v}(t-\tau) = \beta_i^{cv2v} \otimes \Omega_i^{cv2v}(t-\tau)$. Therefore, the validated inequality of C-V2V offloading is given as

$$\mathbb{P}\{X_i \oslash Y_i^{cv2v}(t + d_i, t) \geq 0\} \leq \frac{e^{\theta \varrho_i^{cv2v} o_i + \theta(\eta_{cv2v}^{comp} + \eta^{cv2v}) e^{-\theta \xi_{cv2v}^{comp} d(\varepsilon)}}}{(1 - e^{-\theta((\xi^{cv2v} - \xi_{cv2v}^{comp}))})(1 - e^{-\theta(\xi_{cv2v}^{comp} - \lambda_i)})}. \tag{16}$$

where $\xi^{cv2v} = R_i^{cv2v}$, $\xi_{cv2v}^{comp} = \Theta^{\upsilon eh} + \varrho_i^{cv2v}\lambda_i - \frac{1}{N}\sum_{j=1}^{K}(1 - \varrho_j^{\upsilon 2i} + (N-1)\varrho_j^{local})\lambda_j$, $\eta_{cv2v}^{comp} = -\varrho_i^{cv2v} o_i + \frac{1}{N}\sum_{j=1}^{K}(1 - \varrho_j^{\upsilon 2i} + (N-1)\varrho_j^{local}) o_j$, and $\eta^{cv2v} = 0$. The right-hand-side of Eq. (16) can be regarded as the offloading failure probability ε_i. And, the delay upper bound d_i^{cv2v} of C-V2V offloading is

$$d_i^{cv2v} = \frac{\frac{1}{N}\sum_{j=1}^{K}(1 - \varrho_j^{\upsilon 2i} + (N-1)\varrho_j^{local}) o_j - \frac{\ln \varepsilon_i}{\theta}}{\Theta^{\upsilon eh} - \frac{1}{N}\sum_{j=1}^{K}(1 - \varrho_j^{\upsilon 2i} + (N-1)\varrho_j^{local})\lambda_j + \varrho_i^{cv2v}\lambda_i}, \tag{17}$$

3) Delay Upper Bound of mmWave: The implementation of mmWave communication needs the antenna beam alignment at first that will incur additional time overhead [27]. However, the usage of the control channel for the antenna beam alignment can significantly reduce the time overhead through additional communication overhead [17]. This control channel delivers the Request-To-Send like (RTS-like) and Clear-To-Send like (CTS-like) beacons that involve the kinetic information (location, speed, acceleration) and the request types. In addition, the sub-6GHz control channel for mmWave alignment can be a competition-based channel or a reservation-based channel. For convenient notations, the mmWave offloading with the competition-based channel aided mmWave alignment is denoted by CmmW. And, RmmW represents the mmWave offloading with the reservation-based channel aided mmWave alignment.

a) Delay Upper Bound of CmmW: MmWave communication includes two procedures: the beam alignment and data transmission. In this sub-section, we resort to the DSRC channel as the competition-based control channel for the beam alignment that is depicted in Fig. 3, in which vehicle A initiates the beam alignment through sending an RTS-like beacon that contains the kinetic information of vehicle A. Once vehicle B has received the RTS-like beacon, it returns a CTS-like beacon

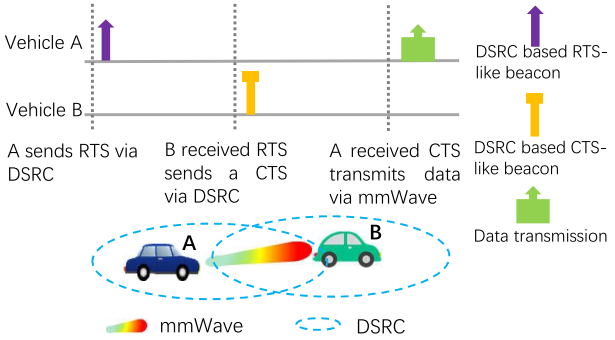


Fig. 3. Beam alignment of CmmW communications.

with its kinetic information to confirm the mmWave connection. So far the alignment procedure has completed [17]. Due to the concatenated property, the service curve of mmWave communication is the convolution of the beam alignment procedure and the data transmission procedure, i.e.,

$$\beta^{net}(t-\tau) = \left(\beta^{RTS} \otimes \beta^{CTS} \otimes \beta^{CmmW}\right)(t-\tau), \quad (18)$$

where β^{RTS} and β^{CTS} are the service curves of RTS-like traffic and CTS-like traffic transmissions, respectively. In general, the transmissions of RTS-like and CTS-like traffic are symmetrical. Thus, we assume the service curves of the RTS-like and CTS-like traffic transmissions are identity. However, due to the beam alignment using the DSRC channel, the original DSRC traffic could impact the alignment performance. According to the Leftover Service theory, the service curve of the DSRC channel for the RTS traffic is $\beta^{RTS} = [R_i^{dsrc} - (2\mathscr{O}^b + \sum_{j=1}^{N} \varrho_j^{dsrc}\mathscr{O}_j)]^+(t-\tau)$, where $\mathscr{O}^b$ is the brustiness measure of the RTS-like traffic that is identical to that of the CTS-like traffic. However, the brustiness measure of RTS/CTS-like traffic usually too small to interfere with the channel capacity compared with that of the original DSRC traffic. To simplify the analysis, the brustiness measure of RTS/CTS-like traffic is negligible, i.e., $\mathscr{O}^b = 0$. In addition, the service curve of mmWave data transmission is $\beta^{CmmW}(t-\tau) = R^{mmW} \times (t-\tau)$, where R^{mmW} is the channel capacity of mmWave. Therefore, according to Eq. (18), the service curve of mmWave communication is $\beta^{net}(t-\tau) = [R^{dsrc} - 2\sum_{j=1}^{N} \varrho_j^{dsrc}\mathscr{O}_j]^+ (t-\tau)$.

Hereafter, the task transmitted by CmmW also has to compete with other offloaded and local processed tasks for the on-board CPU cycling. Thus, the on-board processing service curve of task i that is offloaded by CmmW is expressed as

$$\begin{aligned}&\Omega_i^{CmmW}(t-\tau)\\ &= [\Theta^{veh} + \varrho_i^{CmmW}\lambda_i - \frac{1}{N}\sum_{j=1}^{K}(1-\varrho_j^{v2i}+(N-1)\varrho_j^{local})\lambda_j]\\ &\quad \cdot (t-\tau) - \left\{\frac{1}{N}\sum_{j=1}^{K}(1-\varrho_j^{v2i}+(N-1)\varrho_j^{local})o_j - \varrho_i^{CmmW}o_i\right\}.\end{aligned} \quad (19)$$

In the sequel, based on the concatenated property, we get the total service curve of the CmmW offloading, Y_i^{CmmW}:

$$Y_i^{CmmW}(t-\tau) = \beta^{net} \otimes \Omega_i^{CmmW}(t-\tau). \quad (20)$$

And, the validated inequality for the CmmW offloading is

$$\begin{aligned}&\mathbb{P}\{X_i \oslash Y_i^{CmmW}(t+d(\varepsilon),t) \geq 0\}\\ &\leq \frac{e^{\theta\varrho_i^{mmw}o_i + \theta(\eta_{mmw}^{comp}+\eta^{mmw})}e^{-\theta\xi_{mmw}^{comp}d(\varepsilon)}}{(1-e^{-\theta((\xi^{mmw}-\xi_{mmw}^{comp}))})(1-e^{-\theta(\xi_{mmw}^{comp}-\lambda_i)})},\end{aligned} \quad (21)$$

where $\xi^{mmw} = R^{dsrc} - 2\sum_{j=1}^{N}\varrho_j^{dsrc}\mathscr{O}_j$, $\xi_{mmw}^{comp} = \Theta^{veh} + \varrho_i^{mmw}\lambda_i - \frac{1}{N}\sum_{j=1}^{K}(1-\varrho_j^{v2i}+(N-1)\varrho_j^{local})\lambda_j$, $\eta_{mmw}^{comp} = -\varrho_i^{mmw}o_i + \frac{1}{N}\sum_{j=1}^{K}(1-\varrho_j^{v2i}+(N-1)\varrho_j^{local})o_j$, and $\eta^{mmw} = 2\sum_{j=1}^{K}\varrho_j^{dsrc}o_j$. We denote the right-hand-side of Eq. (21) as the failure probability ε_i. Thus, the upper bound of CmmW offloading delay $d_i^{mmw_c}$ for task i is

$$\begin{aligned}&d_i^{mmw_c}\\ &= \frac{2\sum_{j=1}^{K}\varrho_j^{dsrc}o_j + \frac{1}{N}\sum_{j=1}^{K}(1-\varrho_j^{v2i}+(N-1)\varrho_j^{local})o_j - \frac{\ln\varepsilon_i}{\theta}}{\Theta^{veh} - \frac{1}{N}\sum_{j=1}^{K}(1-\varrho_j^{v2i}+(N-1)\varrho_j^{local})\lambda_j + \varrho_i^{mmw}\lambda_i}.\end{aligned} \quad (22)$$

Compared with the C-V2V offloading, the mmWave offloading has an additional item $2\sum_{j=1}^{K}\varrho_j^{dsrc}o_j$ due to the influence of original DSRC traffic. This DSRC traffic deteriorates the performance of CmmW offloading.

b) Delay Upper Bound of RmmW: The reservation-based control channel is purchased from the mobile network operator or licensed by the standards in the future. Upon the same derivation of the CmmW offloading, the service curve of RTS-like/CTS-like traffic transmission in RmmW is $\beta_r^{RTS} = [R_i^{rc} - 2\sum_{j=1}^{N}\varrho_j^{mmw}\mathscr{O}_j^b]^+(t-\tau)$ where R^{rc} is the reserved bandwidth for control channel. Therefore, the service curve β_r^{net} of the RmmW transmission is

$$\beta_r^{net}(t-\tau) = [R^{rc} - 4\sum_{j=1}^{N}\varrho_j^{mmw}\mathscr{O}_j^b]^+(t-\tau). \quad (23)$$

Moreover, the on-board processing service curve of RmmW is the same as that of CmmW. Therefore, the delay upper bound of RmmW offloading $d_i^{mmw_r}$ for task i is

$$\begin{aligned}&d_i^{mmw_r}\\ &= \frac{4\sum_{j=1}^{N}\varrho_j^{mmw}o_j^b + \frac{1}{N}\sum_{j=1}^{K}(1-\varrho_j^{v2i}+(N-1)\varrho_j^{local})o_j - \frac{\ln\varepsilon_i}{\theta}}{\Theta^{veh} - \frac{1}{N}\sum_{j=1}^{K}(1-\varrho_j^{v2i}+(N-1)\varrho_j^{local})\lambda_j + \varrho_i^{mmw}\lambda_i},\end{aligned} \quad (24)$$

where o_j^b is the brustiness measure of the RTS/CTS-like traffic. Usually, the volume of the RTS/CTS-like traffic is constrained to a small amount for mitigating the communication budget. Hence, the constant $4\sum_{j=1}^{N}\varrho_j^{mmw}o_j^b$ is negligible. Then, the delay upper bound of RmmW offloading is similar to that of the C-V2V offloading, i.e., Eq.(17). Consequently, the performance of C-V2V offloading can be a good reference for that of RmmW. Therefore, we do not individually explore the RmmW offloading in the remainder of this article.

B. Infrastructure Edge Computing

As shown in Fig. 1, cellular base stations along the roadside collect data from vehicles and deliver the data to the VEC pool for processing. Thus, the C-V2I offloading is composed of two components: the uplink transmission and VEC pool processing. Since C-V2I uplink is typical reservation-based communication, its service curve is given as

$$\beta_i^{\text{uplink}}(t-\tau) = R_i^{v2i}(t-\tau), \tag{25}$$

where R_i^{v2i} is the assignment uplink bandwidth for task i. As for the VEC pool processing, the computing resources of the VEC pool are shared with all the upload tasks. Thus, the VEC pool processing service curve for task i is

$$\Omega_i^{v2i}(t-\tau) = (\Theta^{epc} - \sum_{j\neq i}^{K}\varrho_j^{v2i}\lambda_j)(t-\tau) - \sum_{j\neq i}^{K}\varrho_j^{v2i}o_j, \tag{26}$$

in which Θ^{epc} is the total computing capacity of the VEC pool that is larger than the computing capacity of on-board processor Θ^{veh}. And, the total service curve of C-V2I offloading for task i is expressed as

$$Y_i^{v2i}(t-\tau) = \beta_i^{\text{uplink}} \otimes \Omega_i^{v2i}(t-\tau). \tag{27}$$

Referring to the derivation of Eq. (10), the validated inequality of C-V2I offloading is

$$\mathbb{P}\{X_i \oslash Y_i^{v2i}(t+d(\varepsilon),t) \geq 0\} \leq \frac{e^{\theta\varnothing_i+\theta(\eta^{epc}+\eta^{uplink})}e^{-\theta\xi^{epc}d_i}}{(1-e^{-\theta((\xi^{uplink}-\xi^{epc}))})(1-e^{-\theta(\xi^{epc}-\lambda_i)})}, \tag{28}$$

where $\xi^{epc} = \Theta^{epc} - \sum_{j\neq i}^{K}\varrho_j^{v2i}\lambda_j$, $\xi^{uplink} = R_i^{v2i}$, $\eta^{epc} = \sum_{j\neq i}^{K}\varrho_j^{v2i}o_j$, and $\eta^{uplink} = 0$. Moreover, the delay upper bound of C-V2I offloading is:

$$d_i^{v2i} = \frac{\sum_{j=1}^{K}\varrho_j^{v2i}o_j - \frac{\ln\varepsilon_i}{\theta}}{\Theta^{epc} - \sum_{j\neq i}\varrho_j^{v2i}\lambda_j}. \tag{29}$$

C. Local Processing

Vehicular applications could be self-digested by the vehicle's own on-board processor. However, the local processed task i still has to compete with other offloaded and local tasks. According to the Leftover Service property, the local processing service curve of task i is

$$Y_i^{local}(t-\tau) = [\Theta^{veh} - \frac{1}{N}\sum_{j=1}^{K}(1-\varrho_j^{v2i}+(N-1)\varrho_j^{local})\lambda_j](t-\tau) - \frac{1}{N}\sum_{j=1}^{K}(1-\varrho_j^{v2i}+(N-1)\varrho_j^{local})o_j. \tag{30}$$

And, the corresponding validated inequality is

$$\mathbb{P}\{X_i \oslash Y_i^{local}(t+d(\varepsilon),t) \geq 0\} \leq \frac{e^{\theta\varrho_i^{local}o_i+\theta(\eta_{local}^{comp}+\eta^{local})}e^{-\theta\xi_{local}^{comp}d(\varepsilon)}}{(1-e^{-\theta((\xi^{local}-\xi_{local}^{comp}))})(1-e^{-\theta(\xi_{local}^{comp}-\lambda_i)})}, \tag{31}$$

where $\xi^{local} = 0$, $\xi_{local}^{comp} = \Theta^{veh} - \frac{1}{N}\sum_{j=1}^{K}(1-\varrho_j^{v2i}+(N-1)\varrho_j^{local})\lambda_j$, $\eta_{local}^{comp} = \frac{1}{N}\sum_{j=1}^{K}(1-\varrho_j^{v2i}+(N-1)\varrho_j^{local})o_j$, and $\eta^{local} = 0$. Letting the right-hand-side of Eq. (31) equal to ε_i, the delay upper bound d_i^{local} for local processing is given as

$$d_i^{local} = \frac{\frac{1}{N}\sum_{j=1}^{K}(1-\varrho_j^{v2i}+(N-1)\varrho_j^{local})o_j - \frac{\ln\varepsilon_i}{\theta}}{\Theta^{veh} - \frac{1}{N}\sum_{j=1}^{K}(1-\varrho_j^{v2i}+(N-1)\varrho_j^{local})\lambda_j}. \tag{32}$$

V. Model Optimization

This section proposes a new optimization model taking account of the communication and computing cost, as well as the failure probability. When vehicles have utilized C-V2X communication, the cellular operator will charge the fee (A_i per Mbps) to vehicles for transmission service. Additionally, the communication cost $C_i^{comm}(t-\tau)$ of task i is only generated by the C-V2X communication due to the licensed band, i.e.,

$$C_i^{comm}(t-\tau) = A_i(\varrho_i^{v2i}+\varrho_i^{cv2v})\mathscr{O}_i(t-\tau). \tag{33}$$

Regarding the computing cost, the unit computing costs of the VEC pool and on-board processor are $\mathscr{A}_i^{infra}$ per Mbps and $\mathscr{A}_i^{veh}$ per Mbps, respectively. The VEC pool computing cost is produced by C-V2I offloading, while the computing cost of the on-board processor is yielded by the DSRC offloading, the C-V2V offloading, and the mmWave offloading. Since the local processing only employs the local computing resource, it does not generate any cost of communication and computing. Thus, the computing cost for task i is

$$C_i^{comp}(t-\tau) = [\mathscr{A}_i^{infra}\varrho_i^{v2i} + \mathscr{A}_i^{veh}(\varrho_i^{cv2v}+\varrho_i^{mmw}+\varrho_i^{dsrc})] \times \mathscr{H}_i\mathscr{O}_i(t-\tau), \tag{34}$$

where $\mathscr{H}_i$ is the computation complexity of task i [28].

Furthermore, we originally treat the offloading failure probability as another system cost in the VEC because the failed offloading will tightly deteriorate the quality of service for automotive tasks. Therefore, the minimized cost model for the heterogeneous VEC is proposed in $P1$.

$$\begin{aligned}
\textbf{P1:}\ \min_{\boldsymbol{\varrho}}\ & \sum_{i=1}^{K} C_i^{comm}(\boldsymbol{\varrho}) + C_i^{comp}(\boldsymbol{\varrho}) + \psi\delta_i\varepsilon_i \\
\text{s.t.,}\ \ & \text{C1: } \varrho_i^h, \varepsilon_i \in [0,1], \quad \sum_h \varrho_i^h = 1,\ i \in \mathcal{I} \\
& \text{C2: } \sum_{i=1}^{K} (\varrho^{v2i} + \varrho^{cv2v})\mathscr{O}_i(t) \le R^{cv2x} \\
& \text{C3: } \sum_{i=1}^{K} \varrho_i^{dsrc}\mathscr{O}_i(t) \le R^{dsrc} \\
& \text{C4: } \bigcap_h \left\{ d_i^h \le T_i^{max} \right\}, \quad i \in \mathcal{I}
\end{aligned} \tag{35}$$

where $h \in \{dsrc, v2i, cv2v, mmw, local\}$. $\boldsymbol{\varrho} = [\varrho^{dsrc}, \varrho^{v2i}, \varrho^{cv2v}, \varrho^{mmw}, \varrho^{local}]$. T_i^{max} is the delay requirement of task i. The delay upper bound d_i^h and offloading failure probability ε_i of different access technologies are attained by Eq. (14), (17), (24), (22), (29), (32), respectively. $\boldsymbol{\delta} = [\delta_1, \delta_2, \ldots \delta_K]$ represents the priority of application i, and ψ is the weight coefficient of the failure probability. If the transportation system prefers to reduce the resource cost, ψ should be a small number. On the contrary, if the transportation system prefers to gain reliable V2X offloading, the value of ψ should be raised. This article first proposes the offloading failure probability as the optimization objective that is rational and neglected in previous literature. However, $P1$ is non-convex and is made particularly challenging by the presence of d_i^h and ε_i. In the sequel, we present two efficient approaches to tackle $P1$.

According to Eq. (14), (17), (22), (29), (32), ε_i is inversely proportional to d_i^h. When ε_i has been minimized, constraint $C4$ becomes tight, i.e., $d_i^h = T_i^{max}$. Thus, we can obtain $\ln \varepsilon_i^h$ by solving $d_i^h = T_i^{max}$ based on Eq. (14), (17), (22), (29), (32). Consequently, $C4$ can be removed through replacing $\ln \varepsilon_i$ as $\max_h \ln \varepsilon_i^h(\boldsymbol{\varrho})$. And, $P1$ transforms into $P2$, equivalently.

$$\begin{aligned}
\textbf{P2:}\ \min_{\boldsymbol{\varrho}}\ & \sum_{i=1}^{K} C_i^{comm}(\boldsymbol{\varrho}) + C_i^{comp}(\boldsymbol{\varrho}) + \psi\delta_i \max_h \ln \varepsilon_i^h(\boldsymbol{\varrho}) \\
\text{s.t.,}\ \ & \text{C1, C2, C3.}
\end{aligned} \tag{36}$$

Investigating the non-convex objective of $P2$, it is still difficult to directly solve. In this article, we propose two canonical solutions to address $P2$. The first is a parallel learning-based method, Federated Q-Learning (FQL) presented in Alg. 1, in which N_s is the training times of the federated Q-learning, and CQ-table is the consensus Q-table for aggregating. The second canonical solution is the Relaxation algorithm that applies the relaxation trick to transfer the original non-convex problem $P2$ to a convex problem with low complexity.

Algorithm 1 Sync Federated Q-Learning (Sync-FQL)

```
Initialize action-state Q-table Q_h; CQ-table CQ; ϱ
for j : 1 to N_s do
    for Traverse offloading technology h in parallel do
        Selecting an action with max Q_h or randomly
        rollout with a certain probability;
        Update reward r_i using Eq. (37); compute local
        update of Q_h using Eq. (38);
    if the aggregator receives all local updated Q_h(t)
    then
        Update global CQ(t+1) using Eq. (39);
        Update Q_h(t+1) ← CQ(t+1) for all h;
```

A. Federated Q-Learning

The V2X offloading selection is modeled as a markov decision process, which consists of a tuple $\{T, A, P, R\}$ where T and A are the set of states and actions, respectively. Transition probability $P(t_i|t_{i-1}, a_{i-1})$ is the probability of a transition occurred where an agent enters state t_i after taking action a_{i-1} at state t_{i-1}. Reward $R(t_i, a_i)$ is a feedback of selecting action a_i at state t_i [29].

1) State: $x_t = \{\boldsymbol{\varrho}(t), W_t\}$ is a state of the offloading assignment, where $\boldsymbol{\varrho}(t) = \{\varrho_i^{dsrc}, \varrho_i^{v2i} \ldots, \varrho_i^{local}\}$ is the proportion of task i transmitted by different offloading technologies. $W_t = \{R_r^{cv2x}, R_r^{dsrc}\}$ is the reserved bandwidth for C-V2X and DSRC. In addition, according to current proportion assignment $\boldsymbol{\varrho}$, the available bandwidth for C-V2X and DSRC are updated to $R_r^{cv2x} = R^{cv2x} - \sum_{i=1}^{K} (\varrho^{v2i} + \varrho^{cv2v})\mathscr{O}_i(t)$, and $R_r^{dsrc} = R^{dsrc} - \sum_{i=1}^{K} \varrho_i^{dsrc}\mathscr{O}_i(t)$, respectively.

2) Action: an action stands for a traffic assignment of different offloading technologies. When one kind of offloading technology increases/decreases 0.01 (or 0.1) percentage of traffic volume, the other offloading technologies will equally decrease/increase $\frac{0.01}{N_f^{total}-1}$ (or $\frac{0.1}{N_f^{total}-1}$) percentage of traffic volume, respectively. N_f^{total} is the total number of offloading technologies (including the local processing). The agent has a certain probability p to select the action with the maximum reward. or randomly chooses an action in the available action set with probability $1-p$.

3) Transition Probability: at each state, $P(t_i|t_{i-1}, a_{i-1})$ is non-zero except for the proportion ϱ_i^h of h^{th} offloading technology attains 1 or 0 that cannot increase or decrease anymore. Otherwise, the current proportion of the technology randomly adds one of the value in $\{0.1, 0.01, -0.1, -0.01\}$ to update its offloading proportion.

4) Reward: When the proportion of h^{th} offloading technology updates, the algorithm returns reward r_i that represents the gain of selecting the offloading proportion ϱ_i for task i. Noted that the reward is a negative form of the total cost, i.e.,

$$r_i = -\psi\delta_i \max_h \ln \varepsilon_i^h(\boldsymbol{\varrho}) - \left\{ \sum_{i=1}^{K} C_i^{comm}(\boldsymbol{\varrho}) + C_i^{comp}(\boldsymbol{\varrho}) \right\}. \tag{37}$$

5) Q-Table: The Q-Table resorts to an action-state performance index function. In this article, it is a three-dimensional matrix whose size is $N_f^{total} \times K \times N_{\boldsymbol{\varrho}}$, where N_f^{total} is the number of the offloading technology. K is the number of the application category. $N_{\boldsymbol{\varrho}}$ is the number of the feasible ϱ_i that is determined by the accuracy of ϱ_i. In this article, the accuracy is set to 0.01, which means that ϱ_i will increase/decrease with at least 0.01 in each iteration. Therefore, the number of feasible ϱ_i is 101, counting from 0 to 100. Then, the element $[i, h, k]$ of the Q-table represents the learning reward (according to Eq. (38)) attained by the state that k percentage of the i^{th} task is offloaded through the h^{th} offloading technology. Additionally, the Q-table is updated as following

$$Q_h(x_t, a) = \alpha \left\{ r_i(t) + \gamma \max_a \left[Q_h(x_{t-1}, a) \right] \right\} + (1-\alpha) Q_h(x_{t-1}, a), \quad (38)$$

where α is the learning rate. γ is the discount factor. Each offloading technology h has its own local Q-Table $Q_h(t)$, where t denotes the iteration time. At $t = 0$, the local Q-Tables of all technologies are initialized to the same value. The Q-learning for h^{th} offloading technology optimization is referred to as the local learning h. After multiple local updates, a global aggregation is performed through the aggregator (deployed at the VEC server or the platoon head).

Furthermore, we investigate two kinds of aggregations: the asynchronous aggregation and the synchronous aggregation. The synchronous aggregation implements the global aggregation when all local learnings have been updated. However, an asynchronous aggregation executes the global aggregation when one of the local learning has been updated. After the global aggregation, the aggregator will yield a consensus Q-table CQ that is composed of some local learning results.

$$CQ(t) = \frac{(N^{update} - 1) \times CQ(t-1) + \frac{1}{N_f} \sum_h^{N_f^{total}} Q_h(t)}{N^{update}}, \quad (39)$$

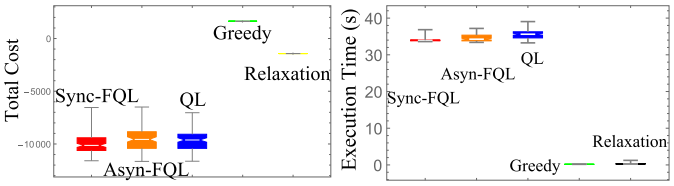
where $Q_h(t)$ is the local Q-table for offloading technology h at iteration t. N^{update} is the number of the updated local Q-Table that is equal to N_f^{total} in the synchronous aggregation. The asynchronous update represents that the global consensus Q-table $CQ(t)$ is updated once a local learning has updated its own Q-table. The main potential problem of the asynchronous aggregation is that each local update is achieved on the model that could be outdated. In the simulation part, we can see that the performance of synchronous federated learning outperforms that of asynchronous federated learning. Consequently, we mainly focus on the synchronous aggregation in this article. After the synchronous aggregation, the updated consensus Q-table $CQ(t)$ is broadcasted to each local Q-learning. Then, the local Q-table is revised by the consensus Q-table CQ. This procedure is shown in Alg. 1.

B. Relaxation Optimization

In this section, we propose an approximation programming, i.e., the Relaxation algorithm, to quickly obtain the suboptimal result. This Relaxation algorithm has a low computation complexity advantage that approximates $P2$ to a convex optimization problem $P3$:

$$\textbf{P3:} \quad \min_{\boldsymbol{\varrho}} \sum_{i=1}^{K} \left\{ C_i^{comm}(\boldsymbol{\varrho}) + C_i^{comp}(\boldsymbol{\varrho}) + \psi \delta_i \sum_h \ln \varepsilon_i^h(\boldsymbol{\varrho}) \right\} \quad \text{s.t.,} \ \ C1, C2, C3 \quad (40)$$

where the non-convex part $\max_h \ln \varepsilon_i^h$ of $P2$ is replaced by $\sum_h \ln \varepsilon_i^h$. Since the objective of $P3$ has a gap $\left\{ \sum_h \ln \varepsilon_i^h(\boldsymbol{\varrho}) - \max_h \ln \varepsilon_i^h(\boldsymbol{\varrho}) \right\}$ with the objective of $P2$, the total cost of $P3$ is worse (larger) than that of $P2$. However, $\ln \varepsilon_i^h(\boldsymbol{\varrho})$ is an affine function of $\boldsymbol{\varrho}$ that indicates the linear summation of the affine function $\sum_h \ln \varepsilon_i^h(\boldsymbol{\varrho})$ is convex. Moreover, $C1 - C3$ are linear constraints. Hence, $P3$ is a convex optimization problem. Although the optimal solution of $P3$ is worse than that of $P2$, the computational complexity is greatly simplified because the non-convex problem $P2$ is converted to the convex problem $P3$. It means that the Matlab-based CVX toolbox could solve $P3$ directly.



(a) *Total cost of algorithms.* **(b)** *Execution time of algorithms.*

Fig. 4. Performances of different algorithms.

VI. Performance Evaluation

To confirm the efficiency of the proposed synchronous federated Q-learning (Sync-FQL), we investigate the asynchronous federated Q-learning (Asyn-FQL), original Q-learning (QL), Greedy algorithm, and Relaxation algorithm for comparison. The Asyn-FQL method is substantially the same as the Sync-FQL, but the global generation of Asyn-FQL is performed once any local learning has been updated. As for the original Q-learning, it does not have the global aggregation process. The greedy algorithm is devised to always choose the maximum reward r_i in each action selection. The results of the Relaxation algorithm are obtained by directly solving $P3$ through CVX.

A. Performance of Offloading Algorithms

The box-plot of the total cost (i.e., the objective value of $P2$) versus different algorithms is illustrated in Fig. 4a, where the y-axis represents the total cost. The results are collected from 500 simulations. Each training period of the learning-based algorithms (i.e., Sync-FQL, Asyn-FQL, and QL) is set to 200 times. Additionally, there are 5 vehicular tasks generated by each vehicle and the number of vehicles is set to 5.

To simplify the analysis, the priorities of different applications are equally, i.e., $\boldsymbol{\delta} = [1, 1, \ldots, 1]$, and the weight

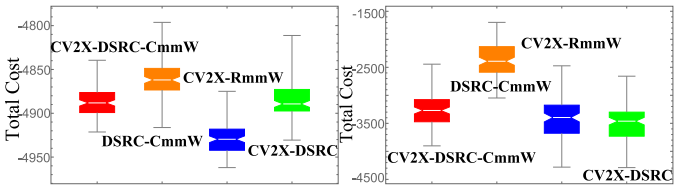
of failure probability ψ is set to 1. In addition, the computing resource of the service vehicle and VEC pool are 10^3 $Mbps$ and 10^4 $Mbps$, respectively. As for the communication bandwidth, the transmission rate of DSRC and C-V2X are 10^3 $Mbps$ and 10^4 $Mbps$, respectively. However, since mmWave communication could provide a significant large bandwidth compared to that of the sub-6 GHz technologies, we assume the transmission delay of mmWave communication is negligible. The arrival rates of the tasks are [20, 70, 30, 80, 8], respectively. And the corresponding burstiness measures of different tasks are [60, 400, 90, 380, 10]. Moreover, the delay requirement of tasks are [2, 6, 2.5, 3, 1], respectively.

As shown in Fig. 4a, the proposed Sync-FQL has the lowest total cost. And, the total cost of Greedy and Relaxation algorithms are much worse than that of the learning-based algorithms. The reason is that the proposed federated learning-based algorithm (Sync-FQL and Asyn-FQL) can obtain the learning results of different access technologies via the global aggregation. However, due to lacking global aggregation, the original Q-learning cannot obtain the diverse learning benefits from different access technologies.

Furthermore, Sync-FQL implements global aggregation upon all local learnings that have updated. However, Asyn-FQL applies global aggregation when each local learning has updated. Although asynchronous aggregation has proven to be faster than those synchronous, which often results in convergence to poor results. The main potential problem of the asynchronous aggregation is that the global update implements over potentially old parameters.

The execution time of different algorithms is illustrated in Fig. 4b. In the simulations, each algorithm runs 500 times to obtain stable statistical results. The simulation is implemented by Matlab on a laptop with i5-8300h CPU and 16G RAM. The Greedy and Relaxation algorithms have low execution time because of the simplicity. Regarding the practical vehicular applications, Campolo *et al.* [21] pointed out that not all the applications are delay-sensitive. For example, the latency requirements of the vehicular Internet, infotainment, and remote diagnostics are over 100ms. These delay-tolerant applications occupy almost whole vehicular transmission. Once Sync-FQL is trained on a powerful edge server at first, it could meet the requirement of the delay-tolerant applications. As for the delay-sensitive applications, we can apply the proposed Relaxation algorithm rather than the learning-based scheme. Although the performance of the Relaxation is worse than that of learning-based algorithms, its execution time is very short that is suitable for the delay-sensitive applications. In addition, delay-sensitive applications only occupy a small percentage of wireless communication bandwidths. Therefore, the overall V2X offloading efficiency will not decline significantly due to adapting the Relaxation offloading algorithm.

Hereafter, we investigate the Sync-FQL performance of heterogeneous offloading frameworks in Fig. 5. There are 4 different heterogeneous frameworks: the C-V2X combining with DSRC and CmmW (CV2X-DSRC-CmmW) framework; the DSRC combining with CmmW (DSRC-CmmW) framework; the C-V2X combining with RmmW (CV2X-RmmW) framework; and the C-V2X combining with DSRC (CV2X-DSRC) framework. Since the formation of Eq.(17) is the same as that of Eq.(24) if o_j^b is negligible, the performance of the CV2X-DSRC framework could be a good reference for that of the RmmW combing with DSRC (RmmW-DSRC) framework. In the *light traffic loads* scenario, there are 5 categories tasks with the arrival rate $\boldsymbol{\lambda} = [5, 5, 5, 5, 5]$, brustiness measure $\boldsymbol{o} = [2, 2, 2, 2, 2]$, and the delay requirement $\boldsymbol{T}_{max} = [1, 1, 1, 1, 1]$, respectively. While the parameters of the *heavy traffic loads* in Fig. 5b are set to $\boldsymbol{\lambda} = [100, 100, 100, 100, 100]$ and $\boldsymbol{o} = [50, 50, 50, 50, 50]$ with $\boldsymbol{T}_{max} = [1, 1, 1, 1, 1]$.



(a) *Performances in light loads.* **(b)** *Performances in heavy loads.*

Fig. 5. Performances of different frameworks.

Upon the *light traffic loads* (Fig. 5a), CV2X-RmmW offers the lowest total cost (resource cost and failure probability). In the *heavy traffic loads* (Fig. 5b), the performance of CV2X-DSRC is slightly better than that of CV2X-RmmW. Resorting to the DSRC-CmmW framework, it has the largest total cost. The reason is that CmmW offloading will compete with DSRC offloading for the bandwidth because the mmWave beam alignment needs the side information transferred by the DSRC channel. In addition, the RmmW offloading provides better service experience than that of DSRC in the *light traffic loads*. However, the situation converses in the *heavy traffic loads*. This is because the mmWave beam alignment requires extra control messages transmissions: the RTS-like and CTS-like transmissions. Respecting the DSRC offloading, it only has one data transmission. Therefore, the RmmW offloading suffers more deteriorating than that of DSRC in heavy traffic.

Remark 1: Upon the light traffic loads, C-V2X combining with the RmmW has the lowest total cost (resource cost and failure probability). However, C-V2X combining with DSRC has the lowest total cost in the heavy traffic loads.

B. Performance of Offloading Technologies

The offloading failure probability versus tasks delay requirement T_{max} for different V2X technologies is depicted in Fig. 6a-6e. In the *heavy traffic loads*, the failure probability is inverse with T_{max}. Besides, the Relaxation algorithm has the lowest failure probability. Fig. 6f presents the resource cost of different algorithms, where the resource cost is defined as the sum of the communication cost and the computing cost $\sum_{i=1}^{K} C_i^{comm}(\boldsymbol{\varrho}) + C_i^{comp}(\boldsymbol{\varrho})$. Because of the inherent stochastic of learning algorithms, the cost curves of learning-based algorithms are fluctuations, while the curves of Greedy and Relaxation are constant. Moreover, the resource cost of the Greedy and the Relaxation algorithms are larger than that of learning-based algorithms. Due to the complexity of

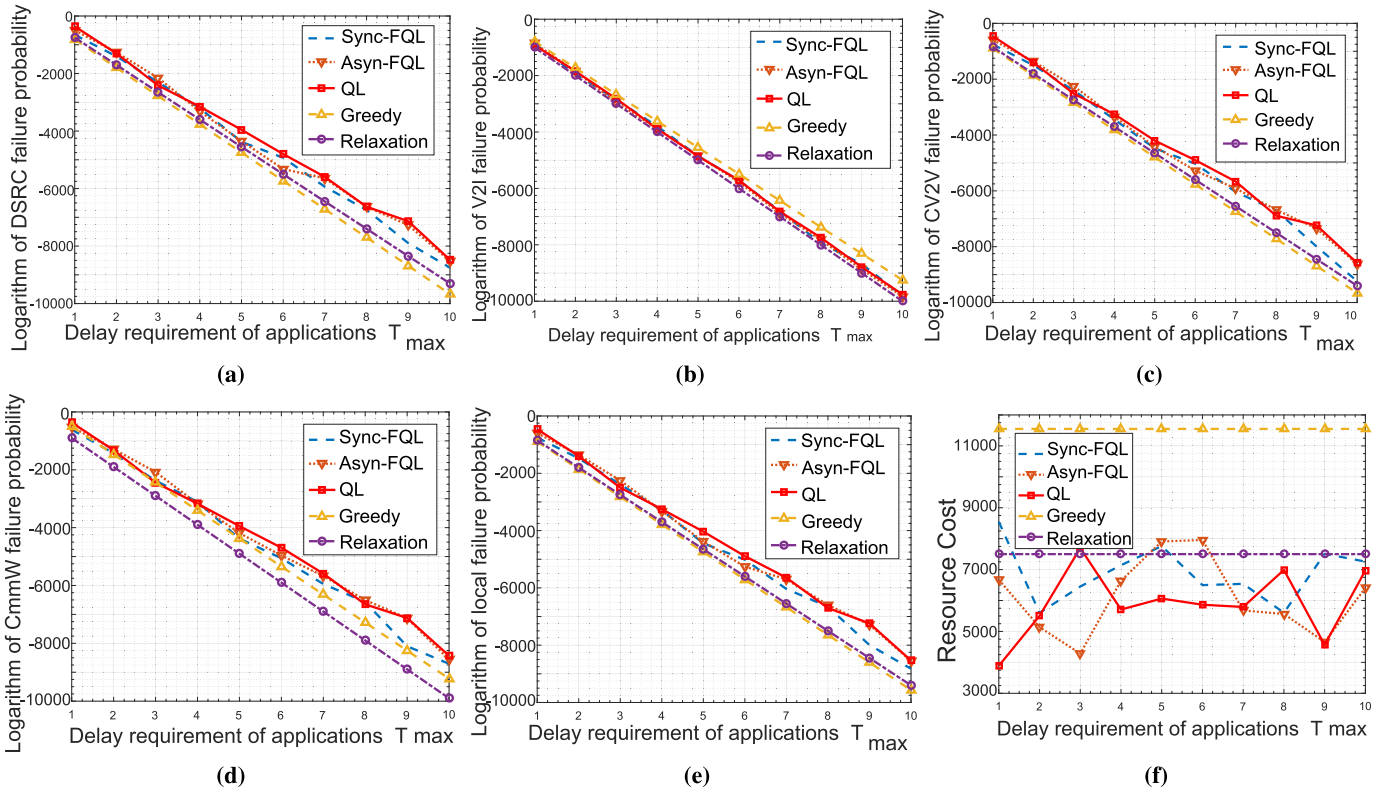
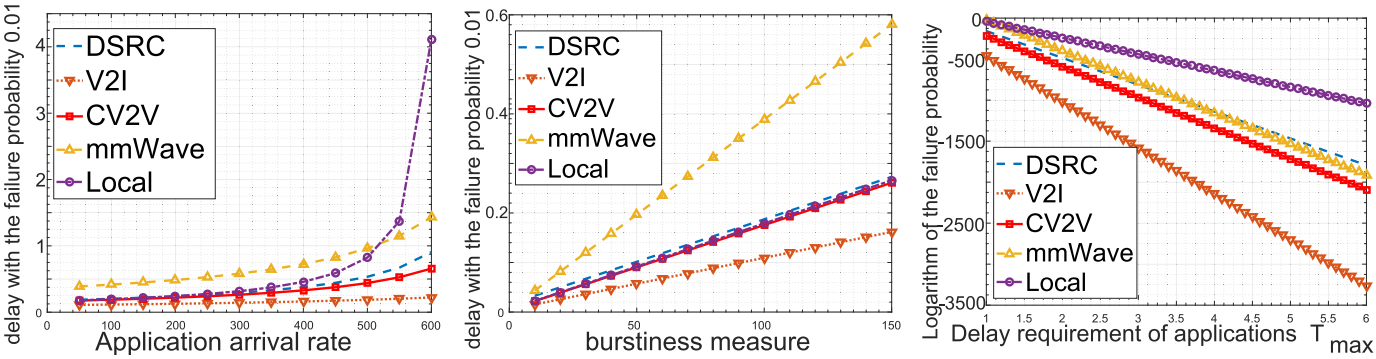


Fig. 6. (a)-(e) are the logarithm failure probabilities for DSRC, V2I, C-V2V, mmWave, and local processing, respectively. (f) is the resource cost for different algorithms.



(a) *Delay upper bound with different arrival rate* λ *under the failure probability* $\varepsilon = 0.01$ *and* $o = 100$.

(b) *Delay upper bound with different burstiness* o *under the failure probability* $\varepsilon = 0.01$ *and* $\lambda = 50$.

(c) *Logarithm failure probability* $\ln \varepsilon$ *with different delay requirements.*

Fig. 7. Performance of different offloading technologies under various tasks.

the $P2$, the Greedy algorithm cannot simultaneously optimize the failure probability and resource cost. As for the Relaxation algorithm, since the objective of $P3$ involves the sum of all failure probabilities, the Relaxation algorithm prefers to optimize the failure probability rather than the resource cost. This is the reason that the total cost of Greedy and Relaxation are bad in Fig. 4.

C. Performance with Traffic Loads

Fig. 7 verifies the impact of different traffic loads in VEC. Fig. 7a shows the delay upper bounds of different V2X technologies increased with the arrival rate λ of tasks. In this simulation, the acceptable failure probability is set to 0.01, and the burstiness is 100 Mbp. When the arrival rate excesses 500 $Mpbs$, the delay upper bound of the local process grows sharply. It reveals that offloading the heavy workloads to others (i.e., vehicles or VEC servers) instead of processing locally, is an efficient approach to reduce the offloading delay. Fig. 7b illustrates that the delay upper bounds of V2X technologies raised with the burstiness measure o. In addition, the delay upper bound of mmWave is larger than that of other V2X technologies. It means that the mmWave offloading is not suitable for the heavy burstiness traffic. However, in Fig. 7c, the C-V2I offloading has the lowest delay upper bound.

Thus, a task with the large burstiness trait prefers to C-V2I offloading.

Fig. 7c illustrates the failure probability decreased with the rising of delay requirements T_{max} of tasks. The mmWave offloading has the slowest decline, which demonstrates that the mmWave offloading is not the best choice for the delay-tolerant traffic. However, in Fig. 7c, C-V2V offloading (or RmmW offloading) has the lowest failure probability. Therefore, the delay-tolerant task prefers to C-V2V offloading for the lowest failure probability.

Remark 2: Vehicles should offload the large arrival rate traffic to others; the CmmW offloading is not proper to the heavy burstiness tasks. And, a delay-tolerant task prefers the C-V2V offloading due to the lowest failure probability.

VII. Conclusion

In this article, we have studied the task offloading in the heterogeneous VEC system with various V2X communications and edge servers. To minimize the offloading failure probability and resource cost, we first derive the upper bound of offloading delay with offloading failure probability through the stochastic network calculus. These upper bounds can be used as an indicator to guarantee the quality of services for automotive tasks. Hereafter, we devote to efficient offloading in the presence of task offloading failure (transmission failed and computing interrupted) and propose an intelligent offloading scheme to ensure offloading reliability while reducing resource cost. Simulations show that the C-V2X combining with RmmW offloading and the C-V2X combining with DSRC offloading have the best performance in the light and heavy traffic loads, respectively. Upon the traffic load with a high arrival rate, vehicles should offload the high arrival rate traffic to others. However, when tasks become more bursty, vehicles should avoid using CmmW communication. Besides, a delay-tolerant application should utilize the C-V2V offloading for the low offloading failure probability. In this article, we have studied the task offloading in the heterogeneous VEC system with various V2X communications and edge servers. To minimize the offloading failure probability and resource cost, we first derive the upper bound of offloading delay with offloading failure probability through the stochastic network calculus. These upper bounds can be used as an indicator to guarantee the quality of services for automotive tasks. Hereafter, we devote to efficient offloading in the presence of task offloading failure (transmission failed and computing interrupted) and propose an intelligent offloading scheme to ensure offloading reliability while reducing resource cost. Simulations show that the C-V2X combining with RmmW offloading and the C-V2X combining with DSRC offloading have the best performance in the light and heavy traffic loads, respectively. Upon the traffic load with a high arrival rate, vehicles should offload the high arrival rate traffic to others. However, when tasks become more bursty, vehicles should avoid using CmmW communication. Besides, a delay-tolerant application should utilize the C-V2V offloading for the low offloading failure probability.

References

[1] K. Zhang, Y. Mao, S. Leng, Y. He, and Y. Zhang, "Mobile-edge computing for vehicular networks: A promising network paradigm with predictive off-loading," *IEEE Veh. Technol. Mag.*, vol. 12, no. 2, pp. 36–44, Jun. 2017.

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

[3] G. Qiao, S. Leng, S. Maharjan, Y. Zhang, and N. Ansari, "Deep reinforcement learning for cooperative content caching in vehicular edge computing and networks," *IEEE Internet Things J.*, vol. 7, no. 1, pp. 247–257, Jan. 2020.

[4] 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, Jun. 2019.

[5] H. Chai *et al.*, "Proof-of-reputation based-consortium blockchain for trust resource sharing in Internet of vehicles," *IEEE Access*, vol. 7, pp. 175744–175757, 2019.

[6] X. Chen, S. Leng, Z. Tang, K. Xiong, and G. Qiao, "A millimeter wave-based sensor data broadcasting scheme for vehicular communications," *IEEE Access*, vol. 7, pp. 149387–149397, 2019.

[7] G. Qiao, S. Leng, K. Zhang, and Y. He, "Collaborative task offloading in vehicular edge multi-access networks," *IEEE Commun. Mag.*, vol. 56, no. 8, pp. 48–54, Aug. 2018.

[8] K. Zhang, S. Leng, Y. He, S. Maharjan, and Y. Zhang, "Cooperative content caching in 5G networks with mobile edge computing," *IEEE Wireless Commun.*, vol. 25, no. 3, pp. 80–87, Jun. 2018.

[9] I. Mavromatis, A. Tassi, and R. J. Piechocki, "Operating ITS-G5 DSRC over unlicensed bands: A city-scale performance evaluation," in *Proc. IEEE 30th Annu. Int. Symp. Pers., Indoor Mobile Radio Commun. (PIMRC)*, Sep. 2019, pp. 1–7.

[10] Y. Ghasempour *et al.*, "IEEE 802.11ay: Next-Generation 60 GHz Communication for 100 Gb/s Wi-Fi," *IEEE Commun. Mag.*, vol. 55, no. 12, pp. 186–192, Dec. 2017.

[11] S. Chen, J. Hu, Y. Shi, L. Zhao, and W. Li, "A vision of C-V2X: Technologies, field testing, and challenges with chinese development," *IEEE Internet Things J.*, vol. 7, no. 5, pp. 3872–3881, May 2020.

[12] K. Zhang, Y. Mao, S. Leng, S. Maharjan, and Y. Zhang, "Optimal delay constrained offloading for vehicular edge computing networks," in *Proc. IEEE Int. Conf. Commun. (ICC)*, May 2017, pp. 1–6.

[13] M. N. Sial *et al.*, "Stochastic geometry modeling of cellular V2X communication over shared channels," *IEEE Trans. Veh. Technol.*, vol. 68, no. 12, pp. 11873–11887, Dec. 2019.

[14] Q. Zheng, K. Zheng, L. Sun, and V. C. M. Leung, "Dynamic performance analysis of uplink transmission in cluster-based heterogeneous vehicular networks," *IEEE Trans. Veh. Technol.*, vol. 64, no. 12, pp. 5584–5595, Dec. 2015.

[15] C. Huang *et al.*, "Iterative channel estimation using LSE and sparse message passing for mmWave MIMO systems," *IEEE Trans. Signal Process.*, vol. 67, no. 1, pp. 245–259, Jan. 2019.

[16] K. Abboud *et al.*, "Interworking of DSRC and cellular network technologies for V2X communications: A survey," *IEEE Trans. Veh. Technol.*, vol. 65, no. 12, pp. 9457–9470, Dec. 2016.

[17] B. Coll-Perales, J. Gozalvez, and M. Gruteser, "Sub-6GHz assisted MAC for millimeter wave vehicular communications," *IEEE Commun. Mag.*, vol. 57, no. 3, pp. 125–131, Mar. 2019.

[18] K. Katsaros, M. Dianati, R. Tafazolli, and X. Guo, "End-to-end delay bound analysis for location-based routing in hybrid vehicular networks," *IEEE Trans. Veh. Technol.*, vol. 65, no. 9, pp. 7462–7475, Sep. 2016.

[19] M. Fidler and A. Rizk, "A guide to the stochastic network calculus," *IEEE Commun. Surveys Tuts.*, vol. 17, no. 1, pp. 92–105, 1st Quart., 2015.

[20] G. Yang, M. Xiao, H. Al-Zubaidy, Y. Huang, and J. Gross, "Analysis of millimeter-wave multi-hop networks with full-duplex buffered relays," *IEEE/ACM Trans. Netw.*, vol. 26, no. 1, pp. 576–590, Feb. 2018.

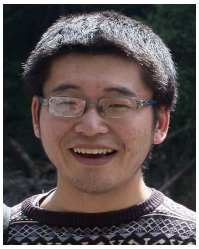
[21] C. Campolo, A. Molinaro, A. Iera, and F. Menichella, "5G network slicing for Vehicle-to-Everything services," *IEEE Wireless Commun.*, vol. 24, no. 6, pp. 38–45, Dec. 2017.

[22] Y. Jiang and Y. Liu, *Stochastic Network Calculus*. Springer, 2008.

[23] M. Fidler, "An End-to-End probabilistic network calculus with moment generating functions," in *Proc. 14th IEEE Int. Workshop Qual. Service*, Jun. 2006, pp. 261–270.

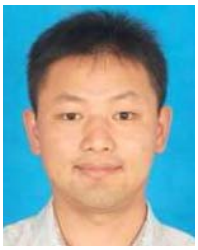
[24] J.-W. Cho and Y. Jiang, "Fundamentals of the backoff process in 802.11: Dichotomy of the aggregation," *IEEE Trans. Inf. Theory*, vol. 61, no. 4, pp. 1687–1701, Apr. 2015.

[25] S. Chen, J. Hu, Y. Shi, and L. Zhao, "LTE-V: A TD-LTE-Based V2X solution for future vehicular network," *IEEE Internet Things J.*, vol. 3, no. 6, pp. 997–1005, Dec. 2016.
[26] S. Chen *et al.*, "Vehicle-to-Everything (v2x) services supported by LTE-based systems and 5G," *IEEE Commun. Standards Mag.*, vol. 1, no. 2, pp. 70–76, Jul. 2017.
[27] C. Huang *et al.*, "Asymptotically optimal estimation algorithm for the sparse signal with arbitrary distributions," *IEEE Trans. Veh. Tech.*, vol. 67, no. 10, pp. 10070–10075, Oct. 2018.
[28] J. Liu and Q. Zhang, "Offloading schemes in mobile edge computing for ultra-reliable low latency communications," *IEEE Access*, vol. 6, pp. 12825–12837, 2018.
[29] H. Baier *et al.*, "Nested monte-carlo tree search for online planning in large MDPs," in *Proc. Eur. Conf. Artificial Intell.*, 2012, pp. 109–114.



Kai Xiong is currently pursuing the Ph.D. degree with the School of Communication and Information Engineering, University of Electronic Science and Technology of China (UESTC), Chengdu. He is currently a Visiting Student with Nanyang Technological University (NTU) and the Singapore University of Technology and Design (SUTD), Singapore. His research interests include design and optimization of next-generation wireless networks, Internet of Vehicles, and mobile edge computing.

Supeng Leng (Member, IEEE) received the Ph.D. degree from Nanyang Technological University (NTU), Singapore. He is currently a Full Professor and the Vice Dean with the School of Information and Communication Engineering, University of Electronic Science and Technology of China (UESTC). He is also the Leader with the Research Group, Ubiquitous Wireless Networks. He has been working as a Research Fellow with the Network Technology Research Center, NTU. He has published over 150 research articles in recent years. His research interests include spectrum, energy, routing and networking in the Internet of Things, vehicular networks, broadband wireless access networks, smart grid, and the next generation mobile networks. He serves as an organizing committee chair and a TPC member for many international conferences, and a reviewer for over ten international research journals.



Chongwen Huang (Member, IEEE) received the B.Sc. degree from Nankai University, Binhai College, in 2010, the M.Sc. degree from the University of Electronic Science and Technology of China (UESTC), Chengdu, in 2013, and the Ph.D. degree from the Singapore University of Technology and Design (SUTD), Singapore, in September 2019. Before this, he joined the Institute of Electronics, Chinese Academy of Sciences (IECAS), Beijing, as a Research Engineer in July 2013. In September 2015, he started his Ph.D. journey with SUTD and CentraleSupélec University, Paris, France, under the supervision of Prof. Chau Yuen and Prof. Mérouane Debbah. From September 2019, he has become a Post-Doctoral Researcher with SUTD. His main research interests include holographic MIMO surface/reconfigurable intelligence surface, 5G/6G wireless communication, deep learning for 5G/6G technologies, and statistics and optimization for wireless communication and intelligent network systems. He was a recipient of the Singapore Government Ph.D. Scholarship and the Partenariats Hubert Curien Merlion Ph.D. Grant from 2016 to 2019 for studying at CentraleSupélec, France. In addition, he also received more than ten outstanding scholarships coming from China and industries, which involves the Tang Lixin Overseas Scholarship, the Tang Lixin Scholarship, the National Postgraduate Scholarships, and the National Second Prize for National Undergraduate Electronic Design.

Chau Yuen (Senior Member, IEEE) received the B.Eng. and Ph.D. degrees from Nanyang Technological University (NTU), Singapore, in 2000 and 2004, respectively. He was a Post-Doctoral Fellow with Lucent Technologies Bell Labs, Murray Hill, in 2005, and a Visiting Assistant Professor with The Hong Kong Polytechnic University in 2008. From 2006 to 2010, he was with the Institute for Infocomm Research (I2R), Singapore, where he was involved in an industrial project on developing an 802.11n Wireless LAN system and participated actively in 3Gpp long-term evolution (LTE) and LTE-Advanced (LTE-A) standardization. Since 2010, he has been with the Singapore University of Technology and Design. He was a recipient of the Lee Kuan Yew Gold Medal, the Institution of Electrical Engineers Book Prize, the Institute of Engineering of Singapore Gold Medal, the Merck Sharp and Dohme Gold Medal, and twice a recipient of the Hewlett Packard Prize. He received the IEEE Asia-Pacific Outstanding Young Researcher Award in 2012 and the IEEE VTS Singapore Chapter Outstanding Service Award in 2019. He serves as an Editor for the IEEE TRANSACTIONS ON COMMUNICATIONS and the IEEE TRANSACTIONS ON VEHICULAR TECHNOLOGY, where he was awarded as the Top Associate Editor from 2009 to 2015. He served as the Guest Editor for several Special Issues, including the IEEE JOURNAL ON SELECTED AREAS IN COMMUNICATIONS, *IEEE Communications Magazine*, and the IEEE TRANSACTIONS ON COGNITIVE COMMUNICATIONS AND NETWORKING. He is a Distinguished Lecturer of the IEEE Vehicular Technology Society.

Yong Liang Guan (Senior Member, IEEE) received the B.Eng. degree (Hons.) from the National University of Singapore and the Ph.D. degree from the Imperial College of London, U.K. He is currently a tenured Associate Professor with the School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore. His research interests include modulation, coding and signal processing for communication systems, and data storage systems. He is an Associate Editor of the IEEE TRANSACTIONS ON VEHICULAR TECHNOLOGY.