Power Control and Task Offloading for Cloud Assisted MEC in Vehicular Networks

Abstract—Cloud-assisted mobile-edge computing (C-MEC) has been witnessed as a novel solution for task offloading in vehicular networks, which is able to provide rich computing resources. In this paper, a robust power control scheme is proposed to offload the computation task and maximize the utility of C-MEC networks. However, an uncertain channel state seriously affects the stable transmission of the offloading task. The first-order Markov process is adopted to simulate channel uncertainty, where vehicular mobility is highly considered. Moreover, channel reusing is assumed due to the limited spectrum resources, which leads to complex co-channel interference and communication delay. To depress the above challenges, probability constraints of signal links are constructed to ensure communication quality. Furthermore, the Bernstein approximations method is adopted to transform the original constraints into solvable ones. Scrupulously, the block coordinate descent (BCD) method and the successive convex approximation (SCA) technique are further adopted to solve the nonconvex robust optimization framework. Furthermore, a joint robust power control and task offloading scheduling algorithm is proposed to approach the optimal solutions. Numerical simulations are performed to evaluate the system performances, and the results indicate that the proposed algorithm is effective and outperform the benchmarks, especially in communication environments with channel uncertainty.

Index Terms—Internet of Vehicle (IoV), Robust Power Control, Edge Computing, Computation Offloading,

I. INTRODUCTION

Mobile-edge computing (MEC) and mobile cloud computing (MCC), as two new architectures for the emerging 5G networks, have been increasingly developing to support task offloading for Internet of Things devices, especially providing the low-latency and high-reliability computing services [?]. At the edge of the network center, MEC can reduce the transmission delay and provide computing resources to the vehicle users to relieve the computing pressures [?]. However, the computing resources of MEC are still inadequately facing a large number of computing tasks. Considering the rich computing resources provided by cloud computing, cloud-based computing networks have been developed to address the explosive-growth requirements of computation offloading. However, cloud computing centers tend to be far from the road. In the high-dynamic Internet of Vehicles, the data produced by vehicles must be processed in a short period [?]. Therefore, the C-MEC is expected to solve this problem, a network architecture that has both rich computing resources and reduced transmission latency.

We explored and adopted a two-layer C-MEC vehicular network. Specifically, we consider a multi-cell network where each Base Station (BS) is equipped with a MEC server to provide computation offloading services for the mobile vehicles. For MEC layer, which has moderate computation capacity and deploys close to networks, can be used to assist the vehicles.

Cloud computing layer, can be used to process the large-scale and delay-insensitive data that the MEC layer can not process. We hope that such a network structure can solve the problem of task offloading in the Internet of Vehicles. Then the C-MEC vehicular network not only curtails the transmission delay of data but also frees the computing pressure of MEC.

However, for the current Mobile-Edge computing that enables vehicular networks, interference in the dense vehicle scenario often leads to acutely poor communication Quality of Service (QoS). In addition, a vehicles mobility leads to an uncertain channel state and further affects the stability of communication. So joint power control and computing resource allocation in multi-vehicles, the multi-MEC system will resolve the task offloading problem in a C-MEC vehicular network and will guarantee the QoS.

A. Related Works

Recently, some works have been devoted to an IoV edge computing network, consisting of a cloud computing layer and MEC layer vehicle network architectures. In [?], a hierarchical computing framework for vehicular networks is proposed which is composed of the control layer, the vehicular edge computing server layer, and the vehicular network layer. Dai et al. Investigate the service scenario of cooperative computation offloading in MEC-assisted service architecture, where multiple MEC servers and remote cloud offload computation-intensive tasks in a collaborative way [?]. Some papers focused on the problem of computation offloading in the C-MEC vehicular network scenario. Tan and Hu designed a joint communication, caching, and computing problem for achieving the operational excellence and cost efficiency of vehicular networks [?]. Wang et al. [?] formulated the problem as a generalized NE problem and presented a game theory algorithm to analyze the equilibrium problem. Wang et al. [?] develop a distributed clustering mechanism designed to classify vehicles into multiple cooperative edge servers and maximize the total revenue in the entire scheduling duration. Li et al. [?] construct an analytical model of the service cache at the edge of the vehicle, mainly considering the computational task offloading and task interdependence between RSUs. In summary, most of the existing works only optimized one of the two indexes power control and computing resource allocation. It is assumed in some research that the vehicles use a constant transmit power while our approach optimizes the vehicles transmit power and the computing resource allocation in a multi-vehicles, multi-MEC system considered. In this case, it caused a new problem because the objective function is difficult to handle. Nemirovski and Shapiro have proposed a convex approximation approach that can solve it [?]. Aiming

at the non-convex of the problem with two variables, some research decouples the original problem into two subproblems and deploys the block coordinate descent (BCD).

Unlike the traditional mobile communications networks with low mobility, the Doppler effect in the high mobility of vehicles poses a challenge to C-MEC communication, when the fast-moving vehicles communicate with different MEC servers, the deterministic channel state information (CSI) is no longer sufficient to describe the channel state in network scenarios with dynamic characteristics, the generated Doppler effect has a significant influence on the small-scale fading of CSI and thereby causing fast channel variations. In other words, the CSIs used are outdated. The First-order Gauss-Markov process is adopted to describe the impacts of the Doppler frequency shift on the channel in [?]. Moreover, in order to the outstanding performance with low communication delay and computing delays, vehicle equipment has a reduced tolerance for delay and transmission reliability, so higher requirements are put forward. Li et al. introduce the outage probability constraint to guarantee the reliability of vehicular links [?]. Considering that the exact expression contains the exponential integral function, to make it tractable, it is necessary to consider an approximate closed-form expression such that the computational complexity can be reduced.

In the C-MEC vehicular networks, authorized vehicles with spectrum resources can directly communicate with the RSU. However, the scarce spectrum resources appear inadequate in high-density vehicular networks [?]. Zhou et al. investigated dynamic sharing of the 5G spectrum and proposed a sharing architecture of DSRC and the 5G spectrum for immersive experience-driven vehicular communications [?]. Tran et al. design a holistic solution for joint task offloading and resource allocation in a multi-server MEC-assisted network [?]. It can be seen effective channel reusing is crucial when the spectrum resources are scarce [?]. But this can cause problems with interference, interference caused by channel reuse in the vehicle communication scenario often leads to acutely poor communication quality. Then, to deal with the interference constraint, the probability constraint is constructed to depress the uncertain co-channel interference, and the Bernstein approximation method is used to transform it into a solvable closed form, the method has commonly been used to deal with the difficult handling non-convex problem [?]. To deal with the outage probability constraint, Xiao et al. assume the CSIs are obtained through channel estimation [?]. Therefore, the outage constraint is transformed according to the Bernstein-type inequality to make it a deterministic optimization problem [?]. Based on the characteristics of our constraints, the Bernstein method is also used in this paper.

B. Contributions

In this paper, a robust power control and task offloading algorithm is proposed for the cloud-assisted MEC in vehicular networks, where the communication delay and computing delay are guaranteed by probabilistic constraints, and user QoS is also guaranteed. The main contributions of this paper are shown as follows:

- A computation offloading architecture named C-MEC vehicular network is developed. For MEC layer, which has moderate computation capacity and deploys close to networks, can be used to assist the vehicles. Cloud computing layer, can be used to process the large-scale, delay-insensitive data that the MEC layer can not process.
- Considering the channel uncertainty caused by the high-speed movement of vehicles in the scenario of the Internet of Vehicles, the first-order Markov process is introduced. A reasonable and feasible IoV network environment is constructed to more realistically describe the dynamic characteristics of the Internet of Vehicles. The Bernstein approximation method previously used in interference constraints is improved and generalized, and it is applied to the matrix form of interruption probability to deal with non-convex outage constraints in large-scale dynamic vehicle network environments to ensure the quality of network communication services.
- Our proposed algorithm considers cross-layer computation and communication resources to guarantee the vehicle QoS and various task requirements under C-MEC vehicular networks.

The remainder of this paper is organized as follows: the model of power control and task offloading for cloud-assisted MEC in vehicular networks is established in Section II. In Section III, the probability constraints and the objective function are formulated and then give the problem solutions. In Section IV, simulation results and performance analysis are presented. Finally, we draw a conclusion in Section V.

II. SYSTEM MODEL

In this work, the C-MEC vehicular network is shown in Fig. 1, which is composed of the MEC layer, and the cloud computing layer hierarchical architecture of computation offloading, numerous vehicles divided into multiple geographic zones within the RSUs coverage underlay a cell, each RSU is equipped with a MEC server to provide computation offloading services to the vehicles. We denote the set of vehicles and MEC servers in the mobile system as $\mathcal{V} = \{1, 2, ..., V\}$ and $\mathcal{M} = \{1, 2, ..., M\}$, respectively. The high-speed mobile wireless communication link is defined as V2RSU (V2R) link, and the fixed wired connection link is defined as RSU to Cloud (R2C) link. The detailed offloading process is described as follows. Firstly, the vehicles offload request messages by the wireless interface, which includes required communication resources, the task ID and submission time, and the maximum tolerable service times of the task to the cloud. Second, the MEC server makes scheduling according to the received request messages, including the task upload server and task computation server. Finally, after the task is uploaded, the task waits in the compute queue until the server's processor processes it. Furthermore, some notations are given in Table I.

Remark 1. *In this article, we consider only simplified cases within a time slot to arrive at a tractable solution. Nevertheless, the proposed solution can be easily extended to the multi-segment scenario by adopting a time division multiple access*

communication technology. That is, the vehicles in each RSU coverage communication are divided into different collections. Hence, time resource is divided into multi-frames, and each frame is divided into several time slots. Different vehicles access their time slots when they communicate with the RSU.
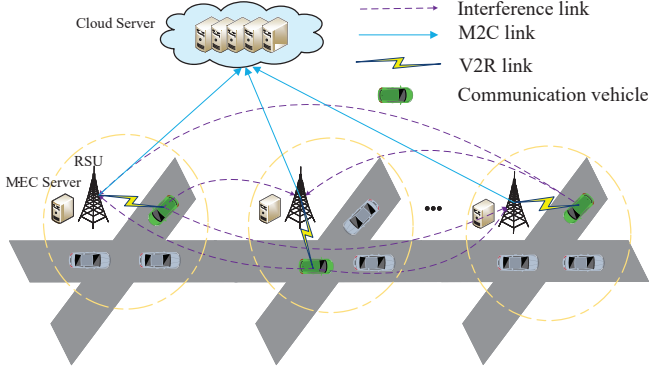

Fig. 1: System model.

TABLE I: Notations

$\Pr\{\cdot\}$	Probability function.
$\mathbb{R}^k$	Set of k-dimensional real vectors.
$\mathbf{f}$	Index set of computing resource $\mathbf{f}=[f_1,\cdots,f_i,\cdots,f_M]$.
$\mathbf{p}$	Index set of vehicle's power $\mathbf{p}=[p_1,\cdots,p_i,\cdots,p_M]$.
$\mathcal{M}$	Index set of vehicles over a time slot $\mathcal{M}=\{1,2,\cdots,M\}$.
$\mathcal{V}$	Index set of all active vehicles $\mathcal{V}=\{1,2,\cdots,V\}$.
$\mathrm{E}\{\cdot\}$	Mathematical expectation of a random variable.

A. Communication Model

Different from traditional cellular communication, Due to the fast mobility of vehicles, their CSIs are hard to be estimated precisely. In particular, RSU can only achieve accurate knowledge of large-scale fading L^2 of vehicular to RSU links while the small-scale fading h is greatly influenced by the fast channel variations caused by the Doppler effect. We assume the CSIs are obtained through channel estimation [?], Therefore, we model the small-scale fading channel estimation of h by using the first-order Gauss-Markov process [?] in each transmission time interval as follows,

$$h = \xi\widetilde{h} + \sqrt{1-\xi^2}\zeta, \tag{1}$$

We assume that the estimated channel gain $\widetilde{h}$ denotes the estimate of h and $\widetilde{h}^2$ is exponentially distributed with unit mean [?]. Furthermore, $\xi \in (0,1)$ represents the correlation coefficient over the V2R link, and ζ stands for the channel gain and follows a complex Gaussian distribution $\zeta \sim CN(0,\delta^2)$ and independent and uncorrelated of $\widetilde{h}$. The coefficient $(0 < \zeta < 1)$ quantifies the channel correlation between the two consecutive time slots and we assume that the time correlation coefficient ζ is the same for all VUEs. According to the Jakes statistical model for the fading channel [?], ζ is given as $\zeta = J_0(2\pi f_{max}T_s)$, where J_0 is the zero-order Bessel function of the first kind. $f_{max} = \bar{\nu}f_c/c$

is the maximum Doppler frequency, where $\bar{\nu}$ indicates the vehicle speed, f_c indicates the carrier frequency at 5.9 Ghz, and $c = 3 \times 10^8 m/s$, T_s is a period feedback latency, both transmitter vehicles and RSU can know the accurate ζ.

Based on the aforementioned discussion, the mobile V2I channel power gain of the effective links and interference links in kth time slot from ith transmitter to jth receiver can be expressed as a shared expression:

$$G_{i,j}^k = \widetilde{g}_{i,j}^k + \hat{g}_{i,j}^k, \tag{2}$$

Where $\widetilde{g}_{i,j}^k = L_{i,j}^2 \widetilde{h}_{i,j}^2 \xi_{i,j}^2$, $\hat{g}_{i,j}^k = L_{i,j}^2 \left(1-\xi_{i,j}^2\right)\zeta_{i,j}^2$, and $L_{i,j}^{k}{}^2$ denotes the kth time slot large-scale fading effects including shadow-fading and path loss from ith transmitter to jth receiver on the road section. Moreover, $\hat{g}_{i,j}^k$ is an observed value. $\widetilde{g}_{i,j}^k$ denotes an exponential random variable with parameter $\frac{1}{L_{i,j}^{k}{}^2(1-\zeta_{i,j}^{k}{}^2)}$.

To improve the spectrum utilization and realize multi-vehicles joint communication, V2R communications reuse the same uplink channel. In this case, the Signal-to-Interference-plus-Noise Ratio (SINR) of V2R link can be formulated as,

$$\gamma_i(\mathbf{p}) = \frac{p_i g_{i,j}}{\sum_{j=1,j\neq i}^M p_j g_{j,i} + \sigma^2} \tag{3}$$

where p_j denotes the transmit power of the jth vehicles, and σ^2 is the background noise. Therefore, the deterministic equivalent transmission rate of VUEs calculated by Shannons theorem is,

$$R_i(\mathbf{p}) = \log_2\left(1 + \frac{p_i g_{i,j}}{\sum_{j=1,j\neq i}^M p_j g_{j,i} + \sigma^2}\right) \tag{4}$$

then the transmission time of vehicle i when sending its task input $d_{i,up}$ in the uplink can be calculated as,

$$t_{i,up} = \frac{d_{i,up}}{R_i(\mathbf{p})} \tag{5}$$

therefore, the upload time of each V2I link can be formulated as,

$$t_{i,up} = \frac{d_{i,up}}{W\log_2\left(1 + \frac{p_i g_{i,j}}{\sum_{j=1,j\neq i}^M p_j g_{j,i}+\sigma^2}\right)} \tag{6}$$

where W is the bandwidth of the reused channel, and $d_{i,up}$ is the amount of input data including system settings, program codes, and input parameters, which is necessary to transfer the program execution.

Communication delay is another significant index that affects the performance of wireless networks. The packets to V2R receivers must be in the queue before they transmit at the speed of R_i. It is assumed that the process of a packet arriving at the ith V2R receiver is a Poisson process with parameter k_i, and the length of the data packet obeys the exponential distribution of parameter τ_i. We adopted the M/M/1 model instructions the relationship between the expected delay and transmission rate of the ith V2I links can be expressed as,

$$D_i = \frac{1}{\tau_i R_i - k_i} \tag{7}$$

B. Vehicle Computing Model

We consider that each vehicle $v \in \mathcal{V}$ has one different computation task at a time. denoted as c_v [cycles] specifies the workload, i.e. the amount of computation to accomplish the task, that is atomic and cannot be divided into subtasks. The values of c_v can be obtained through carefully profiling of the task execution [?]. Each task should be offloaded to the MEC server and then transmission to the cloud server. By offloading the computation task to the MEC server, the vehicles would get more computing resources, however, it would consume additional time for sending the task input in the uplink.

The MEC server at each RSU is able to provide computation offloading service to a vehicle at a time slot. The computing resources are quantified by the fixed rate $\bar{f}$, expressed in terms of number of CPU cycles/s. the vehicle ith uploads the input data of task to the nearest RSU, the RSU process the small-scale, delay-sensitive data first, then the RSU forward the remaining data to the remote cloud server, the cloud is able to provide computation offloading service to multiple RSU concurrently. The computing resources made available by cloud to be shared among the associating users are quantified by the computational rate f_i, it is still expressed in terms of number of CPU cycles/s. Thus, the latency for computing offloading can be written as,

$$t_{i,exe} = \frac{c_{i,e}}{\bar{f} + f_i} \tag{8}$$

C. Problem Definition

Given the computing resource allocation f_i, the total delay experienced by vehicle i when offloading its task is given by,

$$t_i = \frac{c_{i,e}}{\bar{f} + f_i} + T_c \tag{9}$$

The transmission latency between RSU and cloud server is defined as T_c, usually it is set to a fix value [?], so the relative improvement in task completion time is characterized by,

$$U_{i,exe} = \frac{t_{max} - t_{i,exe}}{t_{max}} \tag{10}$$

Where t_{max} is the maximum tolerable threshold of the task completion timeIf a task can be completed ahead of deadline t_{max}, the vehicle can get a higher utility, otherwise, it will produce the corresponding loss.Therefore, we define the offloading utility of vehicle u as,

$$\frac{U_{i,exe}}{t_{i,up}} \tag{11}$$

denote offloading time cost utilities at a unit.

The joint task offloading and resource allocation will be formulated as an optimization problem in this section. And the goal is to obtain the minimum total system cost composed of latency and transmission rate for all vehicles in the networks. For a given uplink power allocation $\mathbf{p}$, and computing resource allocation $\mathbf{f}$, we define the system utility as the weighted-sum of all the vehicles offloading utilities,

$$U = \max \sum_{i=1}^{M} \frac{U_{i,exe}}{t_{i,up}} \tag{12}$$

This utility means getting a more enormous execution time utility with a minor upload time cost. We now formulate the Joint Resource Allocation and Task Offloading Problem as a system utility maximization problem, i.e. The robust optimization problem is formulated as follows:

$$\mathbf{P} : U = \max \sum_{i=1}^{M} \frac{U_{i,exe}}{t_{i,up}} \tag{13a}$$

$$s.t. \begin{cases} \Pr\{\gamma_i \geq \gamma_{th}\} \geq 1 - \varepsilon_1, & (13b) \\ \Pr\left\{\frac{1}{\tau_i R_i - k_i} + \frac{c_{i,e}}{\bar{f} + f_i} \leq D_{max}\right\} \geq 1 - \varepsilon_2 & (13c) \\ \sum_{i=1}^{N} f_i \leq f_{total}, & (13d) \\ 0 \leq p_i \leq p_{max} & (13e) \end{cases}$$

Where U denotes the network utility, the constraints in the formulation above can be explained as follows: Constraints (13a) is used to guarantee the QoS requirements of VUEs, however, due to Large amount of computation caused by time varying network topologies, the real-time SINR is hard to obtain in vehicular communication scenario, and it can be replaced with the long-term SINR since the CSI feedback time interval is very small. γ_i denotes the average SINR of the ith V2I link when a small CSI feedback time interval is used, in order to ensure that the task is successfully offloaded to the RSU, the SINR should be guaranteed to be no less than the SINR threshold [?]. γ_{th} is the SINR threshold for successful detecting the V2I communication. $\Pr\{\cdot\}$ defines the probability of the input. In this case, we introduce the outage probability constraint (13a) to guarantee the reliability of vehicular links [?]. D_{max} is the delay bound of the ith V2I link in the process of data transmission. ε_1, ε_2 are the outage probability thresholds of SINR and delay constraint respectively, where $\varepsilon_1, \varepsilon_2 \in (0, 1)$. constraints (13c) state that each MEC server must allocate a positive computing resource to each user associated with it and that the total computing resources allocated to all the associated users must not excess the servers computing capacity, in another word, the number of applications served by a particular edge cloud should be within its capacity. (13b) denote the total latency of communication and computing should be guaranteed to be no less than the time threshold, p_{max} is the maximum transmit power of the transmit vehicle in vehicle communication network, and the transmit power is greater than zero in (13d).

III. PROBLEM SOLUTIONS

In this section, we proposed a BCD-based algorithm to solve the problem (13). The BCD method enables the complex original problem to be decomposed into a succession of simpler subproblems [?]. Motivated by this fact, all variables are divided into two blocks and optimized alternatively.

By fixing $\mathbf{f}$, the problem (13) can be transformed into the following problem.

$$\mathbf{P1} : U = \max \sum_{i=1}^{M} \frac{U_{i,exe}}{t_{i,up}} \tag{14a}$$

$$s.t. \begin{cases} \Pr\{\gamma_i \geq \gamma_{th}\} \geq 1 - \varepsilon_1 & \text{(14b)} \\ \Pr\left\{\frac{1}{\tau_i R_i - k_i} + \frac{c_{i,e}}{\tilde{f} + f_i} \leq D_{max}\right\} \geq 1 - \varepsilon_2 & \text{(14c)} \\ 0 \leq p_i \leq p_{max} & \text{(14d)} \end{cases}$$

A. Successive Convex Approximation of the Objective Function

Since the original problem is a non-convex and NP-hard because of the logarithmic function in the objective function, here, the method of successive convex approximation is adopted to relax the original problem and make objective function solvable. We can use the lower bound to approach the original function as follows,

$$\begin{cases} \alpha \ln(z) + \beta \leq \ln(1+z) \\ \quad \alpha = \frac{z_0}{1+z_0} \\ \quad \beta = \ln(1+z_0) - \frac{z_0}{1+z_0}\ln(z_0) \end{cases} \quad \text{(15)}$$

Each term of (14) can be represented by $A_k \ln\left(\gamma_k\left(e^{\tilde{\mathbf{P}}}\right)\right) + B_k$ through successive convex approximation, where A_k and B_k can be chosen as $A_k = \gamma_i / (1 + \gamma_i)$, $B_k = \ln(1 + \gamma_i) - A_k \ln(\gamma_i)$, $A_k = 1$, $B_k = 0$, and each term of objective function can be written as follows,

$$\frac{1}{\ln 2} \sum_{i=1}^{M} \frac{U_{i,exe}}{d_{i,up}} [A_k \ln(\gamma(p)) + B_k] \quad \text{(16)}$$

It is still hard to directly calculation because of fractional from of SINR, we use variable substitution, i.e. $\hat{p}_i = \ln p_i$, $p_i = e^{\hat{p}_i}$, then $\hat{p}_i \leq \ln p_{max}$, $\forall \ 1 \leq i \leq M$

$$U = \max \frac{1}{\ln 2} \sum_{i=1}^{M} \frac{U_{i,exe}}{d_{i,up}} \left[A_k \ln\left(\gamma\left(e^{\tilde{P}}\right)\right) + B_k\right] \quad \text{(17)}$$

B. Approximate of the Outage Probability Constraint

It is obvious that the constraint (14a) includes uncertainties and the objective function is a non-convex problem in (14), so the objective function and constraints are difficult to deal with when determining the optimal solutions. It is necessary to design an algorithm with lower complexity to solve the problem. In this paper, For the uncertain channel gain. Considering the fast fading. Two common forms are adopted to describe the uncertainty mentioned above, i.e. the statistical constraints and deterministic constraints. to pursue a simple form of (14a), a matrix form is introduced, the general form the channel gain is described as,

$$\Pr\left\{(G_m)^T e^{\tilde{p}} + \sigma^2 \leq 0\right\} \geq 1 - \varepsilon_1 \quad \text{(18)}$$

b ... Where $\mathbf{G}_m = \left[G_{1,m}, G_{2,m}, \ldots, -\frac{G_{m,m}}{\gamma_{th}}, \ldots, G_{M,m}\right]^T$, Furthermore, the Bernstein method is adopted to approximate the probability constraint with channel uncertainty

Theorem 1: The outage probability of all cochannel V2I links $\Pr\{\gamma_i \leq \gamma_{th}\} \geq 1 - \varepsilon_1$ is reformulated as the separable constraints (14a)

$$\sigma^2 + \sum_{i=1}^{M} \chi_{i,j} e^{\tilde{p}_i} + \sqrt{2\ln\left(\frac{1}{\varepsilon_1}\right) \left(\sum_{m=1}^{M} (\sigma_{i,j}\beta_{i,j}p_i)^2\right)^{\frac{1}{2}}} \leq 0, \quad \text{(19)}$$

Where $\chi_{i,j} = \mu_{i,j}^+ \alpha_{i,j} + \beta_{i,j} + g_{i,j}$, these parameters (i.e. $\sigma_{i,j}$ and $\alpha_{i,j}$) are deduced to be positive in [?]. Suppose that the truncated distributions of $G_{i,j}$ have bounded supports $\left[\tilde{g}_{i,j}^k + \alpha_{i,j}, \tilde{g}_{i,j}^k + \beta_{i,j}\right]$, $\tilde{g}_{i,j}^k$ is an estimate of $G_{i,j}$, Introduce constants $\alpha_{i,j} = \frac{1}{2}(b_{i,j} - a_{i,j})$, $\beta_{i,j} = \frac{1}{2}(b_{i,j} + a_{i,j})$ to normalize the supports to $[-1, 1]$ as follows,

$$\xi_{i,j} = \frac{G_{i,j} - \tilde{g}_{i,j}^k - \beta_{i,j}}{\alpha_{i,j}} \in [-1, 1], \quad \text{(20)}$$

In the last term of (19), the variables p are coupled nonlinearly. Hence, directly finding an acceptably good solution to (14a) by the Bernstein method is time consuming when the K increases and the number of vehicles is large. Therefore, it is necessary to Introduced a $l_2 - norm$ approximate problem for any $x \in \mathbb{R}^k$. Hence, the last term in (19) containing $l_2 - norm$ of the vector is further approximated by $\| x \|_1 \leq \| x \|_2$. Based on these, the constraint in (14a) is further formulated as fellow with lower complexity and higher reliability,

$$\sigma^2 + \sum_{i=1}^{M} \chi_{i,j} e^{\tilde{p}_i} + \sqrt{2\ln\left(\frac{1}{\varepsilon_1}\right)} \sum_{i=1}^{M} |\sigma_{i,j}\beta_{i,j}| e^{\tilde{p}_i} \leq 0 \quad \text{(21)}$$

To pursue a simple form of (21), we define

$$\Pi_i = \sigma^2 + \sqrt{2\ln\left(\frac{1}{\varepsilon_1}\right)} \sum_{i \neq j}^{M} |\sigma_{i,j}\beta_{i,j}| e^{\tilde{p}_i} \quad \text{(22)}$$

The constraint (14b) can be handled by an Integral transformation method, According to constraint (14b), where $D_{max} = D_1 + D_2$, $D_1 = \frac{1}{\tau_i R_i - k_i}$, $D_2 = \frac{c_{i,e}}{f_i}$, $X = \tilde{h}^2$ is an exponential random variable with unit mean, i.e., $X \sim exp(1)$, we can get the feasible power region of the communication delay probability as follows,

$$\left[\ln(1 - \varepsilon_2) - \hat{g}_{i,j}^k\right] e^{\tilde{p}_i} + D^* \leq 0 \quad \text{(23)}$$

The proof of the feasible region can be found as follow,

Proof: The probability constraint of (14b) can be transformed to the deterministic one according to the following inference

$$\Pr\left\{\frac{1}{\tau_i R_i - k_i} + \frac{c_{i,e}}{f_i} \leq D_{max}\right\}$$
$$= \Pr\left\{R_i \geq \frac{1}{R_i(D_{max} - D_2)} + \frac{k_i}{\tau_i}\right\}$$
$$\leq 1 - \Pr\left\{p_i \tilde{g}_{i,j}^k \leq \left(I_{th} + \sigma^2\right) 2^{\frac{1+k_i(D_{max}-D_2)}{\tau_i(D_{max}-D_2)}} - p_i \hat{g}_{i,j}^k\right\}$$
$$= 1 - \int_0^{\left(I_{th}+\sigma^2\right) 2^{\frac{1+k_i(D_{max}-D_2)}{\tau_i(D_{max}-D_2)}} - p_i \hat{g}_{i,j}^k} e^{-x} dx \geq 1 - \varepsilon_2 \quad \text{(24)}$$

Then, we have the equivalent result of the inequality function in (24) as

$$\left[\ln(1 - \varepsilon_2) - \hat{g}_{i,j}^k\right] e^{\tilde{p}_i} + D^* \leq 0 \quad \text{(25)}$$

Where, $D^* = \left(I_{th} + \sigma^2\right) 2^{\frac{1+k_i(D_{max}-D_2)}{\tau_i(D_{max}-D_2)}}$ ∎

In summary, we can obtain a deterministic optimization problem of robust power allocation by transforming the objective function, outage probability constraints, delay constraints.

It is expressed as,

$$\mathbf{P1} : \max \frac{1}{\ln 2} \sum_{i=1}^{M} \frac{U_{i,exe}}{d_{i,up}} \left[A_k \ln \left(\gamma \left(e^{\widetilde{P}} \right) \right) + B_k \right]$$

(26a)

$$s.t. \begin{cases} \sum_{i=1}^{M} \chi_{i,j} e^{\widetilde{p}_i} + \Pi_i \leq 0 & \text{(26b)} \\ \left[\ln \left(1 - \varepsilon_2 \right) - \hat{g}_{i,j}^k \right] e^{\widetilde{p}_i} + D^* \leq 0 & \text{(26c)} \\ -\infty \leq \widetilde{p}_i \leq \ln p_{i,max} & \text{(26d)} \end{cases}$$

C. Optimal Power Control Algorithm

To pursue an iterative algorithm for solving the problem, Lagrange method is used to maximize the lower bound of the original objective under given coefficients X_i and Y_i. It is noted that these two coefficients should be updated to guarantee a monotonic increase in the lower bound performance.

Hence, the Lagrangian function of (26) under fixed coefficients X_i and Y_i can be expressed as,

$$L \left(\widetilde{\mathbf{p}}, \lambda, \mu \right) = \frac{1}{\ln 2} \sum_{i=1}^{M} \frac{U_{i,exe}}{d_{i,up}} \left[A_k ln \left(\bar{\gamma}_k \left(e^{\widetilde{\mathbf{P}}} \right) \right) + B_k \right] \quad (27)$$

$$- \mu_k \left[\left(ln \left(1 - \varepsilon_2 \right) - \hat{g}_{i,j}^k \right) e^{\widetilde{p}_i} + D^* \right]$$

$$- \lambda_k \left[\sum_{i=1}^{M} \chi_{i,j} e^{\widetilde{p}_i} + \Pi_i \right],$$

Where λ_k and μ_k are the Lagrangian multipliers, and $\lambda_k \geq 0$, $\mu_k \geq 0$.

The power vector $\mathbf{p}$ iteration function is obtained by

$$\frac{\partial L \left(\mathbf{p}, \lambda, \mu \right)}{\partial p_i} = A_i - \left[\sum_{j=1, j \neq i}^{M} \left(A_j \frac{\bar{\gamma}_j \left(e^{\widetilde{p}} \right) \bar{G}_{k,j}}{e^{\widetilde{p}_j} \bar{G}_{j,j}} \right) \right. \quad (28)$$

$$\left. + \lambda_i \Pi_i e^{-\widetilde{p}_i} + \mu_i \hat{g}_{i,j}^k \right] e^{\widetilde{p}_i} = 0$$

Based on (28), the iteration for the power allocation , can be formulated as,

$$\widetilde{p}^{(t+1)} = \left[ln A_i + ln \left(\sum_{j=1, j \neq i}^{M} \left(A_j \frac{\bar{\gamma}_j \left(e^{\widetilde{p}} \right) \bar{G}_{k,j}}{e^{\widetilde{p}_j} \bar{G}_{j,j}} \right) \right. \right.$$

$$\left. \left. + \lambda_i \Pi_i e^{-\widetilde{p}_i} + \mu_i \hat{g} \right) \right]_{-\infty}^{ln p_{max}} \quad (29)$$

The Lagrangian multiplier λ, μ, are updated through the sub-gradient method, which are formulated as,

$$\lambda_i^{(t+1)} = \left[\lambda_i^{(t)} + K_\lambda^{(t)} \left(\sum_{j \neq i}^{M} \chi_{i,j} e^{\widetilde{p}_j} + \Pi_i \right) \right]^+ \quad (30)$$

$$\mu_{i,j}^{(t+1)} = \left[\mu_{i,j}^{(t)} + K_\mu \left(\left(\ln \left(1 - \varepsilon_2 \right) - \hat{g}_{i,j}^k \right) e^{\widetilde{p}_i} + D^* \right) \right]^+ \quad (31)$$

Where K_λ, K_μ denote the step-size, and $K_\lambda \geq 0$, $K_\mu \geq 0$. t denotes the iteration index. $[x]^+ = \max[0, x]$.

D. Computing Resource Allocation

After obtaining $\mathbf{p}$, the formulated problem with respect $\mathbf{f}$ reformulated by:

$$\mathbf{P2} : \max \sum_{i=1}^{N} \frac{U_{i,exe}}{t_{i,up}}$$

(32a)

$$s.t. \begin{cases} \Pr \left\{ \gamma_i \leq \gamma_{th} \right\} \geq 1 - \varepsilon_1 & \text{(32b)} \\ \Pr \left\{ \frac{1}{\tau_i R_i - k_i} + \frac{c_{i,e}}{\bar{f} + f_i} \leq D_{max} \right\} \geq 1 - \varepsilon_2 & \text{(32c)} \\ 0 \leq P_i \leq P_{i,max} & \text{(32d)} \end{cases}$$

Notice that the constraints in (32a) and (32b) are convex, by calculating the second-order derivatives of f_i, the Lagrangian function is constructed to seek the optimal powers. Hence, (32) is a convex optimization problem and can be solved using Karush-Kuhn-Tucker (KKT) conditions. The Lagrangian function of (32) is formulated as,

$$U \left(\mathbf{f}, \xi, \varphi \right) = \sum_{i=1}^{M} \frac{R_i \left(P \right)}{\ln 2 * d_{i,up}} \left[1 - \left(\frac{c_{i,e}}{t_{max} \left(\bar{f} + f_i \right)} + \frac{T_c}{t_{max}} \right) \right]$$

$$- \xi_k \left(\frac{1}{\tau_i R_i - \lambda_i} + \frac{c_{i,e}}{\bar{f} + f_i} \leq D_{max} \right) - \varphi_k \left[\sum_{i=1}^{M} f_i - f_{total} \right]$$

(33)

Based on (33), the iteration for the computing resource allocation, can be formulated as, In order to prove the concavity of (32), the following research is taken. The first-order derivative of $U \left(\mathbf{f}, \xi, \varphi \right)$ with respect to f_i is,

$$\frac{\partial U \left(\mathbf{f}, \xi, \varphi \right)}{\partial f_i} = \frac{c_{i,e}}{ln 2 * d_{i,up} t_{max} \left(\bar{f} + f_i \right)^2} = \frac{\Omega_i}{\left(\bar{f} + f_i \right)^2}$$

(34)

in which, for simplicity

$$\Omega_i = \frac{c_{i,e}}{\ln 2 * d_{i,up} * t_{max}}$$

The second-order derivative of is obtained further as,

$$\frac{\partial^2 U}{\partial f_i^2} = -\frac{2 \cdot \Omega_i}{\left(\bar{f} + f_i \right)^3} \leq 0$$

(35)

It is obvious that the second-order derivative of $U \left(\mathbf{f}, \xi, \varphi \right)$ with respect to f_i is always less than zero. Therefore, $U \left(\mathbf{f}, \xi, \varphi \right)$ is a concave function about f_i, Hence, (32) is a convex optimization problem and can be solved using Karush-Kuhn-Tucker (KKT) conditions.

$$\frac{\partial u \left(\mathbf{F}, \xi, \varphi \right)}{\partial f_i} = \frac{\Omega_i R_i \left(P \right)}{\left(\bar{f} + f_i \right)^2} - \xi_k \frac{c_{i,e}}{\left(\bar{f} + f_i \right)^2} - \sum_{i=1}^{N} \varphi_k = 0 \quad (36)$$

Let

$$\frac{\partial u \left(\mathbf{F}, \xi, \varphi \right)}{\partial f_i} = 0$$

the optimal computing resource allocation is:obtained by

$$f_i^* = \sqrt{\frac{\Omega_y R_i \left(P \right) - c_{i,e} \xi_k}{\sum_{i=1}^{N} \varphi_k}} - \bar{f} \quad (37)$$

The iterative expression is as follows,

$$\widetilde{f}^{(t+1)} = \left[\sqrt{\frac{\Omega_y R_i\left(P\right) - c_{i,e}\xi_k}{\sum_{i=1}^{M} \varphi_k}} - \bar{f} \right]_0^{f_{total}} \tag{38}$$

The Lagrangian multiplier η, φ, are updated through the subgradient method, which are formulated as,

$$\xi_i^{(t+1)} = \left[\xi_i^{(t)} + K_\xi^{(t)} \left(\frac{1}{\tau_i R_i - \lambda_i} + \frac{c_{i,e}}{\bar{f} + f_i} - D_{max} \right) \right]^+ \tag{39}$$

$$\varphi_{i,j}^{(t+1)} = \left[\varphi_{i,j}^{(t)} + K_\varphi \left(\sum_{i=1}^{M} f_i - f_{total} \right) \right]^+ \tag{40}$$

With the above efforts, we successfully transform the original problem into two convex subproblems. Then, an alternative iterative algorithm which is summarized in Algorithm 1 is proposed to solve them.

Algorithm 1 Joint Robust Power Control Task Offloading Scheduling Algorithm

---

1: **Input:** Set the maximal iterative number $\mathcal{T}_{max}$, and the iterative index $t = 0$.
2: **repeat**
3:    Given feasible points $\lambda, \mu, \mathbf{f}$.
4:    Solve problem P_1, and obtain the current optimal solution $\widetilde{p}^{(t+1)}$.
5:    Given feasible points $\xi, \varphi, \mathbf{p}$.
6:    Solve the problem P_2, and obtain the current optimal solution $\widetilde{f}^{(t+1)}$.
7: **until** synchronously converge to the optimal solutions or $t \geq \mathcal{T}_{max}$
8: **Output:** $\mathbf{f}, \mathbf{p}$.

Remark 2. *Algorithm1 contains a loop whose time complexity is described by the maximal loop count, $\mathcal{T}_{max}$. Since there are V clusters which use their power iterations for power optimization, the computational complexity of the Algorithm1 is $O(V\mathcal{T}_{max})$.*

IV. SIMULATION RESULTS AND PERFORMANCE ANALYSIS

In this section, numerical simulations are presented to evaluate the performance of the proposed Algorithm1. A MEC-based vehicular network system which includes five clusters under a certain time slot is selected as our fundamental simulation scenario. The major system parameters are listed in Table II. It is noted that the bandwidth W is set as 10 MHz in the numerical simulations. We assume that both the vehicles and RSUs use a single antenna for uplink transmission and reception, respectively,and the variation of the vehicles speed is negligible within the reference time interval. Unless stated otherwise, the parameter value of γ_{th} is set to 10^{-6}, the outage probability threshold $\varepsilon_1 = \varepsilon_2 = 0.1$.

TABLE II: System parameters

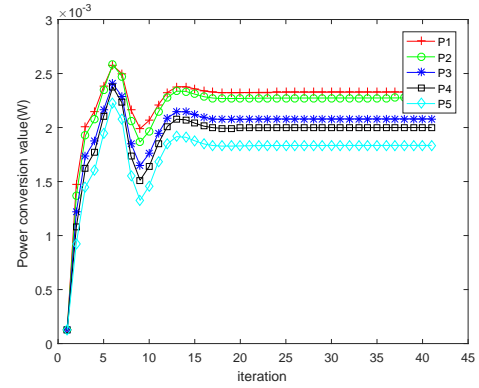| Parameter	Value
Carrier frequency (f_c)	5.9 GHz
Radio Range (R_a)	300 m
CSI feedback period of vehicle (T)	1 ms
Average speed of vehicle	30 m/s
Mean of background noise (σ^2)	-30 dBm
Maximum transmitter power ($p_{i,max}$)	0.05 W
Log-normal shadowing standard deviation	10 dB
Pathloss model	$d^{-\theta}$, d in m
Pathloss exponent (θ)	3

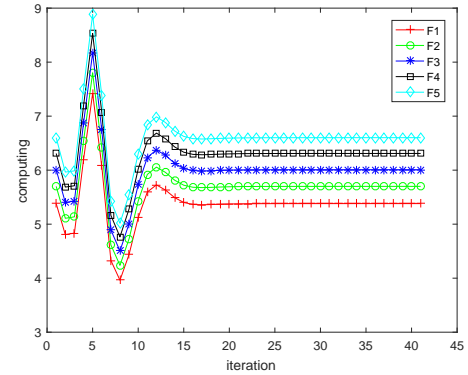Fig. 2: Power convergence performance.

Fig. 3: computing resource which cloud allocation to RSU.
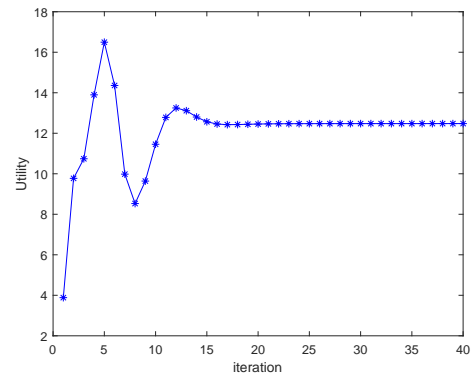

Fig. 4: Comparison of average system utility.

Fig. 1 and Fig. 2 show the power allocation of each vehicle

transmitter and the corresponding computing resource which cloud allocation to RSU in Algorithm1, respectively. It can be seen that the computing resources allocated in the cloud peak at the fifth iteration and begin to decline because of the limitation of total computing resources f_{total} from the cloud. The corresponding power resource allocation also changes due to computing resources allocation under Joint Robust Power Control and Task Offloading Scheduling.

This phenomenon is reasonable due to the definition of U in (12). R_i increases logarithmically as the power vector $\mathbf{p}$ increases, the upload time $t_{i,up}$, as the denominator of U, will decrease with the increase of power vector $\mathbf{p}$ and as the executive utility of the numerator part,$t_{i,exe}$ decreases inversely proportional with the increase of computing power $\mathbf{f}$,causing the numerator to increase with the increase of $\mathbf{f}$.

In the MEC-Enabled vehicular cloud system. It is unrealistic not to take into account the mobility of the vehicle, we then investigated the impact of vehicle mobility on system performance. Assume the variation of the vehicles speed is negligible within the reference time interval. In order to further illustrate the influence of speed-induced Doppler shift on system performance, the comparison experiment between the benchmark value and the increasing speed measurement under the condition of constant vehicle speed is simulated in the system.

Since the relative speed in the V-RSU link is zero, And the speed of all vehicles is the same in the same network there is no Doppler effect. Then the vehicle speed on the road is set to 20 m/s, 30 m/s, 40 m/s, 50 m/s and 60 m/s, respectivelyIt can be seen from Fig. 5 that with the increase of vehicle speed, the utility value of the vehicular network decreases, This is because the higher speed will cause a greater Doppler frequency shift in the network, increase channel uncertainty. The result also proves that methods tend to obtain better utility when the vehicle speed Is low.
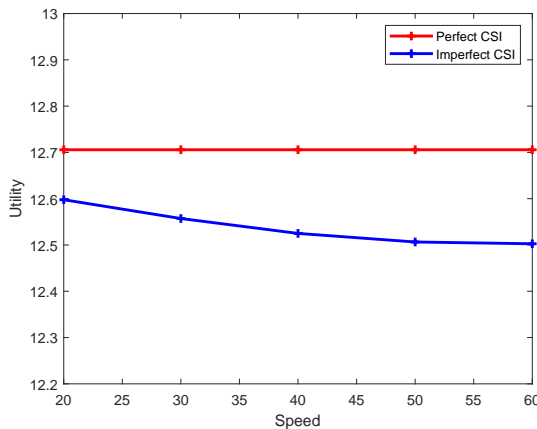

Fig. 5: Comparison of average system utility against different speed.

In order to further verify the performance of the proposed scheme after considering the mobility of the vehicle, the figure.6 describes the effect of the same speed and different speeds of each vehicle under different ε_1 on the total utility,

and it can be seen from the figure that with the change of ε_1, the system utility also changes, and the utility at different speeds of each vehicle is higher than that of all vehicles at the same speed, which characterizes the high robustness of the proposed method in complex dynamic vehicle networks.
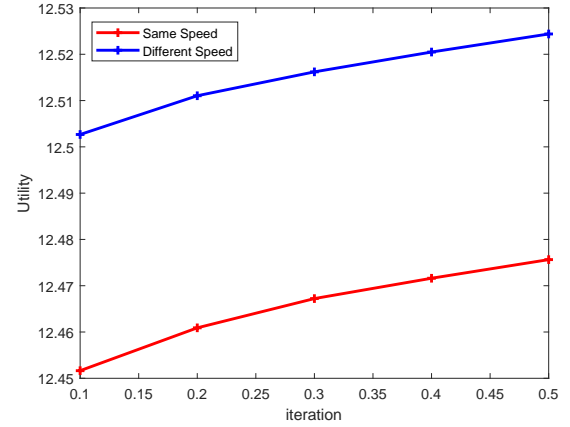

Fig. 6: Comparison of average system utility against different ε_1.

In terms of computing resources allocation, we choose the default task input size as $d_u = 420KB$ (following [?]),We now evaluate the system utility performance against different benchmark schemes. The purpose of this section is to show the convergence of our proposed algorithm and its performance is better than three benchmark schemes through some simulation results. The benchmark schemes are described as follow

1) "Independent Offloading and power control" (denoted as "IOP"):the vehicles independently make power control and the computing resources allocation.
2) "Without vehicle power control"(denoted as "Without-VPC"):The transmit power of the vehicles is set as average power during the offloading.
3) "Without computing resources allocation": (denoted as "Without-CRA"),The transmit power of the vehicles is set as average power during the offloading.

Fig. 6 is the iterative convergence of the total utility of the system in different cases, and it can be seen from the figure that the robust joint optimization performance is better than the other three cases. It can be seen that with the increase of the number of iterations, the four methods converge to a stable value, among which the performance of proposed scheme is the best.
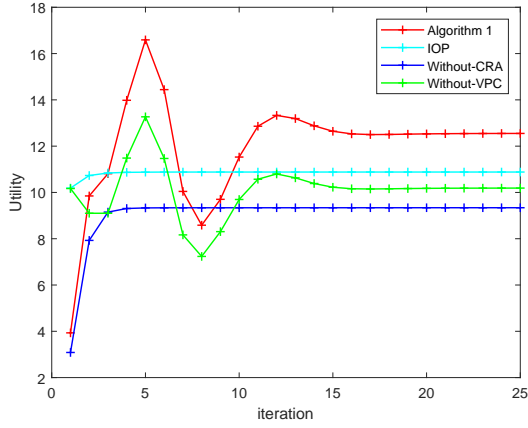
Fig. 7: transmission rate under different numbers.

Fig. 9: Comparison of average system utility against different f_{total}.

In order to reflect a more realistic situation, the CPU task lode (Megzcycles) required for each vehicle are often different, so we set the CPU task lode (Megzcycles) of the five vehicles to 1600, 1700, 1800, 1900, 2000.

As we can see, with the increase of the ratio of iteration, the average system utility of vehicles changes gradually and tends to be stable. In the independent optimization process, the computing resource allocation is carried out first, and the optimal power allocation is not known at this time, and the power and computing resource alternate optimization method is used, and the corresponding optimal value can be obtained for each iteration. Individual optimization is to first optimize the power **p**, and after obtaining the result, the result is used for the optimization of computing resources, and then the computing resources are optimized, and finally, the utility of the system is obtained.However, if joint optimization is used, then both variables can get the optimal value.
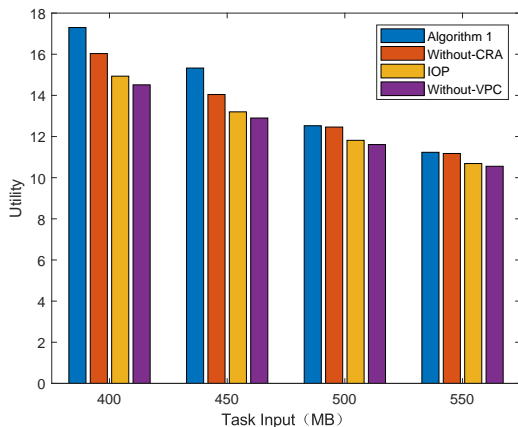

Fig. 10: Comparison of average system utility against different task workloads.
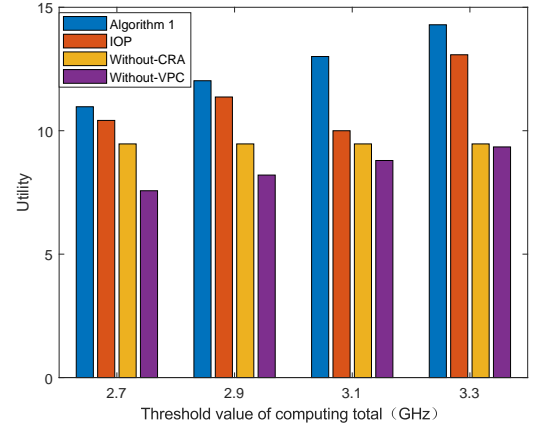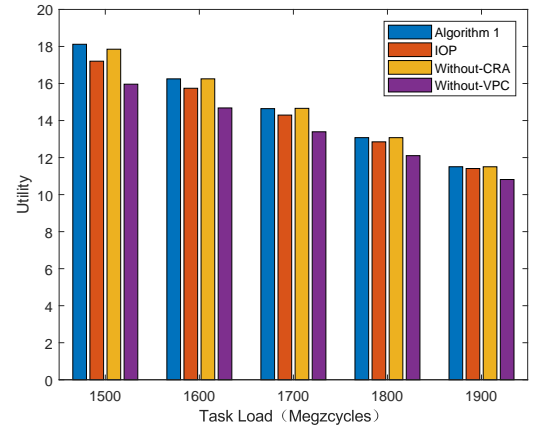
The average system utility of the four competing schemes are plotted in Fig.8 with different values of d_u. It can be seen that the average system utilities of all schemes decrease with the task input size. Moreover, we observe that the performance gains of the proposed scheme over the other schemes also follow the similar trend. The total system cost comparisons with different f_{total} are shown in Fig.9. Due to the limited computation capability at the cloud, when the computation capability is small, the system utility tend to small. We can clearly see that as the data size increases, the system utility tend to small. This is because when the tasks require more upload time with more data size.

V. CONCLUSIONS

This paper focuses on the Joint Task Offloading and Resource Allocation for Mobile-Edge Computing Enable Vehicular Networks with channel uncertainty and co-channel interference. The optimization scheme attempt to guarantees vehicles QoS when there exists a maximized utility requirement. Due to the existence of channel uncertainty, the probability forms of interference, delay, and delivery rate constraints are performed.

Fig. 8: Comparison of average system utility against different task input size.

The underlying optimization problem was formulated as a Mixed-Integer Non-linear Program (MINLP), which is very difficult to solve to optimal, then the SCA technique is applied to transform the non-convex problem of variables coupling into a treatable convex problem. The Task Offloading and power allocation algorithm is developed to achieve practical execution scheme. Simulation results showed that our heuristic algorithm performs closely to the optimal solution and significantly improves the average system offloading utility over traditional approaches.