Energy-efficient Computation Task Splitting for Edge Computing-enabled Vehicular Networks

Hewon Cho†, Ying Cui* and Jemin Lee†

†Daegu Gyeongbuk Institute of Science and Technology (DGIST), South Korea
*Shanghai Jiao Tong University, China
Email: nb00040@dgist.ac.kr, cuiying@sjtu.edu.cn, jmnlee@dgist.ac.kr

***Abstract*—In this paper, we develop the energy-efficient resource allocation scheme for an edge computing-enabled vehicular network. We first introduce the novel prescheduled task splitting and offloading scheme, which splits the vehicle's task into several subtasks and offloads to multiple road side units (RSUs), located ahead along the route. We then formulate the energy minimization problem subject to the communication performance and computation capabilities of the RSUs, and the deadline constraints to optimally split the task and allocate resources at the RSUs. We show that the optimization problem is a convex optimization problem and derive the closed form optimal solutions for the task splitting, the transmission power allocation, the CPU frequency allocation, and the transmission time allocation. Finally, through numerical results, by analyzing the impact of network parameters on the total energy consumption, we verify that our proposed scheme consumes lower energy than baseline schemes.**

I. Introduction

Recently, as the number of vehicles increases and various types of data are required by vehicles, the demand for an efficient vehicular network is increasing. With the emergence of various services such as autonomous driving, online video streaming, and road traffic management in vehicular communications, high computing resources and fast service response time are required. However, the computing resources of the vehicle are limited, so it is difficult to successfully perform the required service by the vehicle alone within the service completion deadline.

In order to solve this problem, the edge computing-enabled vehicular network, which brings the edge computing technology in the vehicular network, is considered as an appropriate solution to improve the service performance. The edge computing is the network architecture concept that enables computation to be performed at the edge of the network such as the cellular base stations (BSs) or other edge nodes (e.g., road side units (RSUs)) [1]. Applying edge computing technology can solve the limited computation capability problem of vehicles by offloading its computation tasks to the nearby BSs (or RSUs). However, there are some challenges in the edge computing-enabled vehicular network. As the vehicles are keep moving, a vehicle generally has short connection time to each BS (or RSU), so the available time for computing the task and downloading the computation result is also limited. Therefore, in the edge computing-enabled vehicular networks, efficient resource allocations for computation and communication are required.

Recently, various task offloading techniques have been presented for the edge computing-enabled vehicular networks [2]–[5]. The offloading technique, where fog nodes (e.g., vehicles) partially offload their computation tasks to the nearby mobile or fixed fog-access points (e.g., trucks or RSUs), has been proposed in [2]. The mobility-aware coded probabilistic caching and computational offloading scheme has been modeled in [3], where a vehicle offloads its computation task to the RSUs and the nearby vehicles under the delay constraint. A cooperative downloading mechanism involving RSUs and vehicles has also been presented in [4] with the consideration on different vehicle roles such as downloader, relay, and carrier vehicles. A cloud-based mobile edge computing (MEC) offloading framework has been presented for vehicular networks in [5]. Specifically, a predictive-mode transmission scheme is proposed, where vehicles send their task-input files to the MEC servers ahead of their running direction through vehicle-to-vehicle (V2V) transmissions. However, in those prior works, they assume performing computation or receiving the computation results of all task from one edge-computing server [2], [5] or other vehicle (e.g., carrier vehicle) [4]. This can be unrealistic in vehicular networks, especially for large size of the computation task, due to the short connection time to server or vehicle as vehicles are moving. Furthermore, the random jump mobility model is considered in [3], which may cause frequency handovers.

Therefore, in this paper, we develop the energy-efficient resource allocation scheme for edge computing-enabled vehicular networks, where the task of a vehicle is offloaded to multiple RSUs (i.e., edge computing servers), located ahead along the route of the vehicle. The main contributions of this work can be summarized as follows: 1) we introduce a novel prescheduled task splitting scheme, which assigns the computation tasks over multiple RSUs with considering their computation capabilities and communication performance; 2) after formulating the energy minimization problem, we obtain

This work was supported in part by the National Research Foundation of Korea (NRF) grant funded by the Korea government(MSIT) (NRF-2018R1A5A1060031)

978-1-7281-7440-2/20/$31.00 ©2020 IEEE

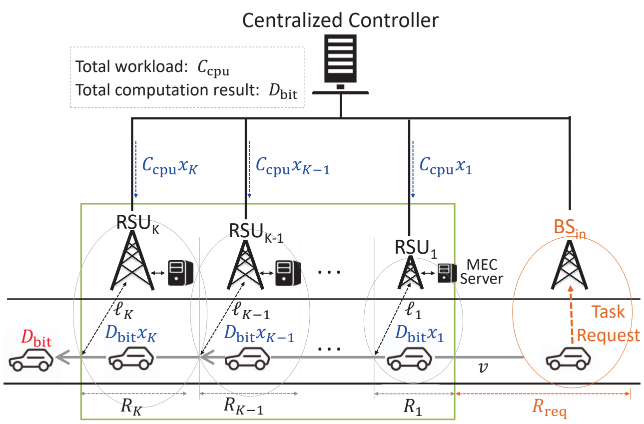

Fig. 1. System Model.

the optimal solutions for the task splitting, the transmission power, the CPU frequency, and the transmission time for each RSU; and 3) we analyze the impact of the computation result size, the vehicle's velocity, and the number of RSUs on the total energy consumption.

II. System Model

In this section, we present the system model of an edge computing-enabled vehicular network including the network description, communication model, computation model, and energy consumption model.

A. Network Description

We consider a unidirectional road, where RSUs are located along the road, as shown in Fig.1. We consider an edge computing-enabled vehicular network which consists of K RSUs, denoted by $\mathcal{K} \triangleq \{1, \cdots, K\}$. [1] Besides communication capability, each RSU has computation capability, since it is equipped with an edge computing server. Each RSU has M transmit antennas. The transmission power limit and the maximum computing capability (i.e., CPU frequency) of RSU k are denoted by P_k and F_k, respectively. The RSUs use the same spectrum of bandwidth B (in Hz). Note that, in reality, the road does not always pass through the center of each RSU's coverage area. Thus, for each RSU k, we introduce two measures, i.e., the length of the interval of the road covered by RSU k, denoted by R_k, and the maximum link length within RSU k's coverage interval, denoted by ℓ_k. We assume that all RSUs are connected to a controller, which is aware of the network parameters, P_k, F_k, R_k, ℓ_k, $k \in \mathcal{K}$ and M.

We consider one single-antenna vehicle on the road with velocity v (in m/s). The vehicle has a computation-intensive task, such as converting a large video file. Assuming the size of the task request is neglegible, the task is characterized by two parameters, i.e., the size of workload $C_{\text{cpu}} > 0$ (in number of CPU-cycles) and the size of the computation result $D_{\text{bit}} > 0$ (in bits) [6]. As C_{cpu} is large, the vehicle offloads the task to the edge computing-enabled vehicular network rather than computing it locally. In particular, the vehicle sends its velocity and location information and task parameters, C_{cpu} and D_{bit}, to the controller and offloads its task to the vehicular network, at a distance of R_{req} (in meters) from the 1st RSU in $\mathcal{K}$.

Since the vehicle keeps moving, it is hard for a single RSU to finish the computation of the whole task and transmit the computation result of the task to the vehicle. Thus, we consider task splitting across K RSUs that are located ahead of the vehicle. In particular, the task is splitted into K subtasks, which are then sent to the K RSUs, respectively. Let x_k denote the ratio of the kth subtask allocated to RSU k, where

$$x_k \geq 0, \quad \forall k \in \mathcal{K}, \tag{1}$$

$$\sum_{k=1}^{K} x_k = 1. \tag{2}$$

The task splitting is thus represented by $\mathbf{x} \triangleq (x_k)_{k\in\mathcal{K}}$. Assume that the sizes of the computational workload and the computation result of the kth subtask are $C_{\text{cpu}}x_k$ and $D_{\text{bit}}x_k$, respectively. Based on the known network parameters and the parameters of the vehicle, v and R_{req}, and its task, $C_{\text{cpu}}x_k$ and $D_{\text{bit}}x_k$, the controller will perform optimal task splitting and resource allocation, which will be further illustrated below.

B. Computation Model

Recall that the maximum computing capability (i.e., CPU frequency) of the RSU k is F_k (in number of CPU-cycles per second). To save the computation energy, RSU k may utilize smaller CPU frequency f_k, where

$$f_k \geq 0, \quad \forall k \in \mathcal{K}, \tag{3}$$

$$f_k \leq F_k, \quad \forall k \in \mathcal{K}. \tag{4}$$

Let $\mathbf{f} \triangleq (f_k)_{k\in\mathcal{K}}$ represent the CPU frequency allocation. Recall that the workload at RSU k has the size of $C_{\text{cpu}}x_k$ (in number of CPU-cycles). Thus, the computation time for the workload of size $C_{\text{cpu}}x_k$ at RSU k, $t_{\text{cp},k}(x_k)$, is given by

$$t_{\text{cp},k}(x_k) = \frac{C_{\text{cpu}}x_k}{f_k}, \quad \forall k \in \mathcal{K}. \tag{5}$$

For ease of implementation, we assume that the computation of the workload of size $C_{\text{cpu}}x_k$ at the RSU $k \in \mathcal{K}$ is completed before the vehicle enters the coverage of RSU k, i.e., $t_{\text{cp},k}(x_k) \leq \frac{1}{v}\left(R_{\text{req}} + \sum_{i=1}^{k-1} R_i\right)$. This leads to the following computation constraints

$$x_k \leq \frac{f_k}{vC_{\text{cpu}}}\left(R_{\text{req}} + \sum_{i=1}^{k-1} R_i\right), \quad \forall k \in \mathcal{K}. \tag{6}$$

The computation energy consumption for computing the workload of size $C_{\text{cpu}}x_k$ at RSU is given by [7]

$$E_{\text{cp},k}(x_k, f_k) = C_{\text{cpu}}x_k\kappa_k f_k^{\varphi_k - 1}, \quad \forall k \in \mathcal{K}, \tag{7}$$

where $\kappa_k \geq 0$ is the effective switched capacitance and $\varphi_k \geq 1$ is a positive constant.

[1] Note that the number of RSUs, K, is determined by the delay requirement of the task requested by the vehicle.

C. Communication Model

After the computation of the kth subtask, the M-antenna RSU k transmits the computation result to the single-antenna vehicle when the vehicle is in its coverage area. This corresponds to multi-input single-output (MISO) transmission. Let $t_{\text{cm},k}$ denote the transmission time in RSU k, where

$$t_{\text{cm},k} \geq 0, \quad \forall k \in \mathcal{K}, \tag{8}$$

$$t_{\text{cm},k} \leq \frac{R_k}{v}, \quad \forall k \in \mathcal{K}. \tag{9}$$

Let $\mathbf{t}_{\text{cm}} \triangleq (t_{\text{cm},k})_{k\in\mathcal{K}}$ represent the transmission time allocation. Recall that the transmission power limit for RSU k is P_k. To save the transmission energy, RSU k may transmit the computation result of the kth subtask to the vehicle with smaller power p_k, where

$$p_k \geq 0, \quad \forall k \in \mathcal{K}, \tag{10}$$

$$p_k \leq P_k, \quad \forall k \in \mathcal{K}. \tag{11}$$

Let $\mathbf{p} \triangleq (p_k)_{k\in\mathcal{K}}$ represent the transmission power allocation.

For tractability, we investigate the MISO transmission for RSU k in the worst case, i.e., with the maximum link length ℓ_k. The received signal of the vehicle from the RSU k, denoted as y_k, is given by [8]

$$y_k = \ell_k^{-\frac{\alpha}{2}} \mathbf{h}_k \mathbf{w}_k \sqrt{p_k} s_k + z, \quad \forall k \in \mathcal{K}, \tag{12}$$

where α is the path loss exponent, s_k is a information symbol with $\mathbb{E}\left[|s_k|^2\right] = 1$, $\mathbf{h}_k \in \mathcal{C}^{1\times M}$ is a vector channel from the RSU k to the vehicle, $\mathbf{w}_k \in \mathcal{C}^{M\times 1}$ is a normalized beamforming vector with $\|\mathbf{w}_k\| = 1$, and z is the complex Gaussian noise following $\mathcal{CN}(0, N_o)$. From (12), the received signal-to-noise ratio (SNR) of the vehicle is given by

$$\text{SNR}_k = \frac{p_k \left|\mathbf{h}_k \mathbf{w}_k\right|^2 \ell_k^{-\alpha}}{N_o}, \quad \forall k \in \mathcal{K}. \tag{13}$$

Given the perfect channel state information at each RSU, we consider maximal ratio transmission (MRT) beamforming, i.e., $\mathbf{w}_k = \frac{\mathbf{h}_k^{\dagger}}{\|\mathbf{h}_k\|}$, which maximizes the received signal power at the vehicle [8]. Here, $\mathbf{h}_k^{\dagger}$ denotes the conjugate transpose of $\mathbf{h}_k$. Thus, (13) becomes

$$\text{SNR}_k = \frac{p_k \left\|\mathbf{h}_k\right\|^2 \ell_k^{-\alpha}}{N_o}, \quad \forall k \in \mathcal{K}. \tag{14}$$

We can further obtain the maximum achievable data rate

$$C_{\max,k} = B \log\left(1 + \frac{p_k \left\|\mathbf{h}_k\right\|^2 \ell_k^{-\alpha}}{N_o}\right), \quad \forall k \in \mathcal{K}, \tag{15}$$

while the average data rate for RSU k to transmit computation result of size $D_{\text{bit}} x_k$ within $t_{\text{cm},k}$ is given by

$$C_k = \frac{D_{\text{bit}} x_k}{t_{\text{cm},k}}, \quad \forall k \in \mathcal{K}. \tag{16}$$

Note that $\mathbf{h}_k$ varies when the vehicle is passing through the coverage interval of RSU k and its values cannot be predicted ahead. Suppose the elements of $\mathbf{h}_k$ are independently and identically distributed (i.i.d.) according to $\mathcal{CN}(0, 1)$. As a result, we impose the following successful transmission constraints

$$\mathbb{P}\left[C_{\max,k} \geq C_k\right] = \mathbb{P}\left[\frac{D_{\text{bit}} x_k}{B \log\left(1 + \frac{p_k \|\mathbf{h}_k\|^2 \ell_k^{-\alpha}}{N_o}\right)} \leq t_{\text{cm},k}\right] \geq \theta_k, \ \forall k \in \mathcal{K}, \tag{17}$$

where θ_k denotes the target successful transmission probability (STP) at RSU k. Note that $\|\mathbf{h}_k\|^2 \sim \Gamma(M, 1)$. Let $G(x)$ denote the complementary cumulative distribution function (CCDF) of the gamma distribution, where

$$G(x) \triangleq \sum_{n=0}^{M-1} \frac{1}{n!} x^n e^{-x}, \tag{18}$$

and $G^{-1}(\cdot)$ denotes the inverse function of $G(\cdot)$, which can be numerically computed. Note that both $G(x)$ and $G^{-1}(y)$ are the decreasing functions. For the single-antenna RSU case, $G(x) = e^{-x}$ and $G^{-1}(\theta_k)$ can be replaced by $\ln(1/\theta_k)$. Then, we have

$$t_{\text{cm},k} \geq \frac{D_{\text{bit}} x_k}{B \log_2\left(1 + \frac{p_k \ell_k^{-\alpha}}{N_o} G^{-1}(\theta_k)\right)}, \quad \forall k \in \mathcal{K} \tag{19}$$

Now, we can obtain the communication energy consumption. The communication energy consumption of transmitting the computation result of size $D_{\text{bit}} x_k$ from RSU $k \in \mathcal{K}$ to the vehicle can be obtained as

$$E_{\text{cm},k}(p_k, t_{\text{cm},k}) = p_k t_{\text{cm},k}, \quad \forall k \in \mathcal{K}. \tag{20}$$

III. Problem Formulation

In this section, we formulate the energy minimization problem. The total energy consumption is given by

$$\begin{aligned} E_{\text{tot}}(\mathbf{x}, \mathbf{p}, \mathbf{f}, \mathbf{t}_{\text{cm}}) &= \sum_{k=1}^{K} \left\{E_{\text{cm},k}(p_k, t_{\text{cm},k}) + E_{\text{cp},k}(x_k, f_k)\right\} \\ &= \sum_{k=1}^{K} \left(p_k t_{\text{cm},k} + C_{\text{cpu}} x_k \kappa_k f_k^{\varphi_k - 1}\right). \end{aligned} \tag{21}$$

We would like to minimize the total energy consumption by optimizing the task splitting $\mathbf{x}$, transmission power allocation $\mathbf{p}$, the transmission time allocation $\mathbf{t}_{\text{cm}}$, and the CPU frequency allocation $\mathbf{f}$. Given the network parameters and the parameters of the vehicle and the task, this optimization can be handled by the controller.

Problem 1 (Energy Minimization):

$$\begin{aligned} E_{\text{tot}}^* \triangleq \min_{\mathbf{x},\mathbf{p},\mathbf{f},\mathbf{t}_{\text{cm}}} \quad & E_{\text{tot}}(\mathbf{x}, \mathbf{p}, \mathbf{f}, \mathbf{t}_{\text{cm}}) \\ \text{s.t.} \quad & (1), (2), (3), (4), (6), (8), (9), \\ & (10), (11), (19), \end{aligned}$$

where $E_{\text{tot}}(\mathbf{x}, \mathbf{p}, \mathbf{f}, \mathbf{t}_{\text{cm}})$ is given by (21). Let $(\mathbf{x}^*, \mathbf{p}^*, \mathbf{f}^*, \mathbf{t}_{\text{cm}}^*)$ denote an optimal solution of Problem 1.

In Problem 1, we can see that the objective function is a convex function, the inequality constraint functions in (1), (3), (4), (6), (8), (9), (10), (11), and (19) are convex functions, and finally the equality constraint function in (2) is an affine function. Therefore, Problem 1 is a convex problem. In the following section, we obtain an optimal solution of Problem 1.

IV. Optimal Solution

In this section, we derive the optimal solution of Problem 1. First, we charaterize opimality properties of Problem 1.

Lemma 1 (Optimal CPU Frequency Allocation and Communication Time Allocation): An solution of Problem 1 satisfies

$$f_k^* = \frac{vC_{\text{cpu}}x_k^*}{R_{\text{req}} + \sum_{i=1}^{k-1} R_i}, \quad \forall k \in \mathcal{K}, \tag{22}$$

$$t_{\text{cm},k}^* = \frac{D_{\text{bit}}x_k^*}{B\log_2\left(1 + \frac{p_k^*\ell_k^{-\alpha}}{N_o}G^{-1}(\theta_k)\right)}, \quad \forall k \in \mathcal{K}. \tag{23}$$

Proof: For given x_k and p_k, the objective function decreases with f_k and $t_{\text{cm},k}$, for all $k \in \mathcal{K}$. Thus, by contradiction, we can show that the inequality constraints in (6) and (19) are active at $(\mathbf{x}^*, \mathbf{p}^*, \mathbf{f}^*, \mathbf{t}_{\text{cm}}^*)$. Therefore, we complete the proof of Lemma 1. ■

Next, based on Lemma 1, we obtain an equivalent problem of Problem 1 with fewer variables and constraints. Define

$$\widetilde{E}_{\text{tot}}(\mathbf{x}, \mathbf{p}) \triangleq \sum_{k=1}^{K}\left(\widetilde{E}_{\text{cm},k}(x_k, p_k) + \widetilde{E}_{\text{cp},k}(x_k)\right), \tag{24}$$

where

$$\widetilde{E}_{\text{cm},k}(x_k, p_k) \triangleq \frac{D_{\text{bit}}x_k p_k}{B\log_2\left(1 + \frac{p_k\ell_k^{-\alpha}}{N_o}G^{-1}(\theta_k)\right)}, \tag{25}$$

$$\widetilde{E}_{\text{cp},k}(x_k) \triangleq x_k^{\varphi_k}C_{\text{cpu}}\kappa_k\left(\frac{vC_{\text{cpu}}}{R_{\text{req}} + \sum_{i=1}^{k-1} R_i}\right)^{\varphi_k - 1}. \tag{26}$$

Then, consider the following optimization problem.

Problem 2 (Master Problem - Task splitting):

$$\begin{aligned} E_{\text{tot}}^* \triangleq \min_{\mathbf{x}} \quad & \sum_{k=1}^{K}\left(E_{\text{cm},k}^*(x_k) + \widetilde{E}_{\text{cp},k}(x_k)\right) \\ \text{s.t.} \quad & (1), (2), \\ & x_k \leq \frac{R_k B}{D_{\text{bit}}v}\log_2\left(\frac{P_k G^{-1}(\theta_k)}{N_o\ell_k^{\alpha}} + 1\right), \forall k \in \mathcal{K} \quad (27) \\ & x_k \leq \frac{F_k}{vC_{\text{cpu}}}\left(R_{\text{req}} + \sum_{i=1}^{k-1} R_i\right), \ \forall k \in \mathcal{K}. \quad (28) \end{aligned}$$

Let $\mathbf{x}^\dagger$ denote an optimal solution of Problem 2. $E_{\text{cm},k}^*(x_k)$ is given by the following subproblem.

Problem 3 (Subproblem - Power Allocation at RSU k): For any x_k,

$$\begin{aligned} E_{\text{cm},k}^*(x_k) = \min_{p_k} \quad & \widetilde{E}_{\text{cm},k}(x_k, p_k) \\ \text{s.t.} \quad & (11), \\ & p_k \geq \frac{N_o\ell_k^{\alpha}}{G^{-1}(\theta_k)}\left(2^{\frac{D_{\text{bit}}x_k v}{R_k B}} - 1\right), \quad \forall k \in \mathcal{K} \quad (29) \end{aligned}$$

Let $p_k^\dagger(x_k)$ denote an optimal solution of Problem 3.

By Lemma 1, we can show that Problem 1 is equivalent to Problem 2 and Problem 3.

Theorem 1 (Equivalence of Problems 1, 2, and 3): The solutions of Problem 1, Problem 2 and Problem 3 satisfy

$$\mathbf{x}^\dagger = \mathbf{x}^*, \qquad \mathbf{p}^\dagger\left(\mathbf{x}^\dagger\right) = \mathbf{p}^*, \tag{30}$$

where $\mathbf{p}^\dagger\left(\mathbf{x}^\dagger\right) \triangleq \left(p_k^\dagger\left(x_k^\dagger\right)\right)_{k\in\mathcal{K}}$.

Proof: By Lemma 1, without loss of optimality, we can replace f_k and $t_{\text{cm},k}$ in Problem 1 with $\frac{vC_{\text{cpu}}x_k}{R_{\text{req}}+\sum_{i=1}^{k-1}R_i}$ and $\frac{D_{\text{bit}}x_k}{B\log_2\left(1+\frac{p_k\ell_k^{-\alpha}}{N_o}G^{-1}(\theta_k)\right)}$, respectively, and ignore the constraints in (6) and (19). Since it is obvious that both f_k and $t_{\text{cm},k}$ are non-negative, we can also ignore (3) and (8). Furthermore, (4) and (9) become (28) and (29), respectively. Here, we can see that (29) implies (10), and therefore, we can ignore (10). Then, Problem 1 can be equivalently transformed to

$$\begin{aligned} E_{\text{tot}}^* \triangleq \min_{\mathbf{x},\mathbf{p}} \quad & \widetilde{E}_{\text{tot}}(\mathbf{x}, \mathbf{p}) \\ \text{s.t.} \quad & (1), (2), (11), (28), (29). \end{aligned}$$

The above problem can be decomposed into two problems: the master problem in Problem 2, which optimizes the task splitting $\mathbf{x}$, and the subproblem in Problem 3, one for each k, which optimizes the power allocation p_k for given $\mathbf{x}$. Note that, in Problem 2, (27) is the feasibility condition that the lower bound of p_k in (29) should be smaller than the upper bound of p_k in (11). Therefore, we complete the proof of Theorem 1. ■

Due to the equivalence shown in Theorem 1, with slight abuse of notation, we will also use $\mathbf{p}^*(\mathbf{x})$ and $\mathbf{x}^*$ to represent the optimal solution of Problem 3 and Problem 2, respectively. By Lemma 1 and Theorem 1, to solve Problem 1, we can first solve Problem 2 and Problem 3 to obtain $(\mathbf{x}^*, \mathbf{p}^*)$, and then substitue $(\mathbf{x}^*, \mathbf{p}^*)$ into (23) and (22) to obtain $\mathbf{t}_{\text{cm}}^*$ and $\mathbf{f}^*$.

First, we solve Problem 3.

Lemma 2 (Optimal Solution of Problem 3):

$$p_k^*(x_k) = \frac{N_o\ell_k^{\alpha}}{G^{-1}(\theta_k)}\left(2^{\frac{D_{\text{bit}}x_k v}{R_k B}} - 1\right), \quad k \in \mathcal{K}. \tag{31}$$

Proof: For given x_k, the objective function decreases with p_k. Thus, by contradiction, we can show that the inequality constraint in (29) is active at $\mathbf{p}^*$. Therefore, we complete the proof of Lemma 2. ■

Next, we solve Problem 2. From Lemma 2, Problem 2 becomes:

Problem 4 (Task Splitting):

$$\min_{\mathbf{x}} \sum_{k=1}^{K} \left(\frac{R_k N_o \ell_k^{\alpha}}{v G^{-1}(\theta_k)} \left(2^{\frac{D_{\text{bit}} x_k v}{R_k B}} - 1 \right) + \widetilde{E}_{\text{cp},k}\left(x_k\right) \right)$$
$$\text{s.t.} \quad (1), (2), (27), (28).$$

Since the objective function is the convex function and all constraints are affine functions, Problem 4 is convex and strong duality holds. Thus, we can obtain an optimal solution of Problem 4 using Karush-Kuhn-Tucker (KKT) conditions [9].

Lemma 3 (Optimal solution of Problem 4):

$$x_k^* = \max\left\{0, \min\left\{ \frac{R_k B}{D_{\text{bit}} v} \log_2\left(\frac{P_k G^{-1}(\theta_k)}{N_o \ell_k^{\alpha}} + 1 \right), \frac{F_k}{v C_{\text{cpu}}}\left(R_{\text{req}} + \sum_{i=1}^{k-1} R_i \right), H^{-1}(\gamma^*) \right\}\right\}, \ k \in \mathcal{K}, \quad (32)$$

where γ^* satisfies $\sum_{k=1}^{K} x_k^* = 1$ and $H^{-1}(\cdot)$ is the inverse function of $H(\cdot)$ given by

$$H\left(x_k\right) \triangleq \frac{D_{\text{bit}} N_o \ell_k^{\alpha} \ln(2)}{B G^{-1}(\theta_k)} 2^{\frac{x_k v}{R_k B}} + x_k^{\varphi_k - 1} \varphi_k C_{\text{cpu}} \kappa_k \left(\frac{v C_{\text{cpu}}}{R_{\text{req}} + \sum_{i=1}^{k-1} R_i} \right)^{\varphi_k - 1}. \quad (33)$$

Proof: First, by relaxing the coupling constraint in (2), we obtain the partial Lagrange function

$$L\left(\mathbf{x}, \gamma\right) = \sum_{k=1}^{K} \left(\frac{R_k N_o \ell_k^{\alpha}}{v G^{-1}(\theta_k)} \left(2^{\frac{D_{\text{bit}} x_k v}{R_k B}} - 1 \right) + x_k^{\varphi_k} C_{\text{cpu}} \kappa_k \left(\frac{v C_{\text{cpu}}}{R_{\text{req}} + \sum_{i=1}^{k-1} R_i} \right)^{\varphi_k - 1} \right) + \gamma \left(1 - \sum_{k=1}^{K} x_k \right), \quad (34)$$

where γ denote the Lagrange multiplier with respect to the constraint in (2). Next, we obtain the derivative of the Lagrange function by

$$\frac{\partial L\left(\mathbf{x}, \gamma\right)}{\partial x_k} = x_k^{\varphi_k - 1} \varphi_k C_{\text{cpu}} \kappa_k \left(\frac{v C_{\text{cpu}}}{R_{\text{req}} + \sum_{i=1}^{k-1} R_i} \right)^{\varphi_k - 1} + \frac{D_{\text{bit}} N_o \ell_k^{\alpha} \ln(2)}{B G^{-1}(\theta_k)} 2^{\frac{x_k v}{R_k B}} - \gamma, \quad k \in \mathcal{K}. \quad (35)$$

Now we obtain the KKT conditions by

$$(1), \ (27), \ (28), \ 1 - \sum_{k=1}^{K} x_k = 0,$$
$$\frac{\partial L\left(\mathbf{x}, \gamma\right)}{\partial x_k} = 0, \quad k \in \mathcal{K}. \quad (36)$$

By substituting (35) into (36), we obtain $x_k = H^{-1}(\gamma)$, where $H^{-1}(\cdot)$ is the inverse function of $H(\cdot)$ in (33). Finally, by considering (1), (27) and (28), we can obtain the optimal task splitting at RSU $k \in \mathcal{K}$, x_k^*, in (32). Therefore, we complete the proof of Lemma 3. ■

Note that, as $H\left(x_k\right)$ is strictly increasing with x_k, $H^{-1}(\gamma^*)$ is also strictly increasing with γ^*. Thus, γ^* satisfying $\sum_{k=1}^{K} x_k^* = 1$ can be easily obtained using the bisection method. Then, we can obtain $\mathbf{x}^*$ according to (32).

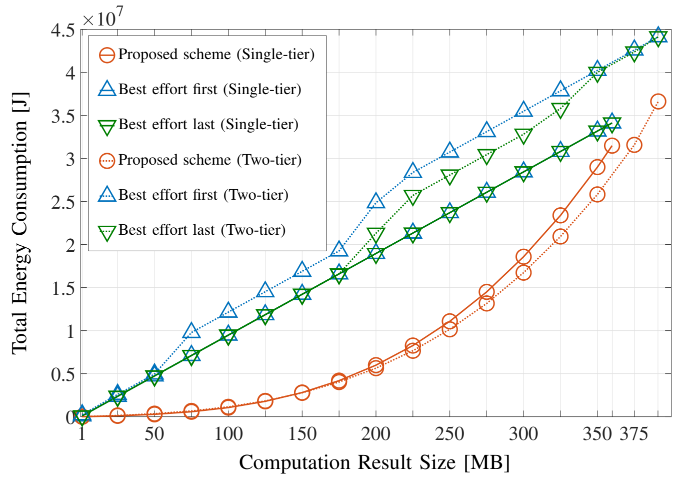

Fig. 2. Energy consumption versus the size of the computation result D_{bit} when $v = 100$km/h and $K = 20$.

V. Numerical Results

In this section, we present numerical results to examine the impacts of various environment parameters on the energy consumption of our proposed scheme. Throughout this section, we consider at most 20 RSUs, i.e., $K \leq 20$. We set $R_{\text{req}} = 300$m, $N_o = -80$dBm, $B = 5$MHz, $\alpha = 4$, $\theta_k = 0.95$, $C_{\text{cpu}} = 1000 D_{\text{bit}}$, $\kappa_k = 10^{-11}$, and $\varphi_k = 3$. We consider two scenarios: the single-tier network scenario and the two-tier network scenario. Set $P_{\text{RSU}} = 45$dBm, $P_{\text{BS}} = 55$dBm, $R_{\text{RSU}} = 400$m, $R_{\text{BS}} = 600$m, $F_{\text{RSU}} = 1.0$GHz and $F_{\text{BS}} = 1.2$GHz. In the single-tier network, $P_k = P_{\text{RSU}}$, $R_k = R_{\text{RSU}}$, and $F_k = F_{\text{RSU}}$, for all $k \in \mathcal{K}$. In the two-tier network, $P_k = P_{\text{BS}}$, $R_k = R_{\text{BS}}$, and $F_k = F_{\text{BS}}$, for all $k \in \{1, 8, 14\}$, and $P_k = P_{\text{RSU}}$, $R_k = R_{\text{RSU}}$, and $F_k = F_{\text{RSU}}$, for all $k \in \mathcal{K} \backslash \{1, 8, 14\}$. Note that RSUs 1, 8, 14 can be viewed as BSs.

To analyze the performance of our proposed scheme, we introduce two schemes: best effort first (BEF) scheme and best effort last (BEL) scheme. We can obtain the maximum ratio of the kth subtask $x_{k,\max}$ that can be handled by RSU k when utilizing the maximum transmission power and CPU frequency. Based on the maximum ratios, the BEF scheme splits the task according to $x_k = x_{k,\max}$ for all $k \in \{1, \cdots, \bar{k} - 1\}$, $x_{\bar{k}} = 1 - \sum_{i=1}^{\bar{k}} x_{i,\max}$, and $x_k = 0$ for all $k \in \{\bar{k} + 1, \cdots, K\}$, where $\bar{k}$ is the smallest k such that $\sum_{i=1}^{k} x_{i,\max} > 1$. On the other hand, the BEL scheme splits the task according to $x_k = x_{k,\max}$ for all $k \in \{\underline{k} + 1, \cdots, K\}$, $x_{\underline{k}} = 1 - \sum_{i=\underline{k}+1}^{K} x_{i,\max}$, and $x_k = 0$ for all $k \in \{1, \cdots, \underline{k} - 1\}$, where $\underline{k}$ is the largest k such that $\sum_{i=\underline{k}}^{K} x_{i,\max} > 1$.

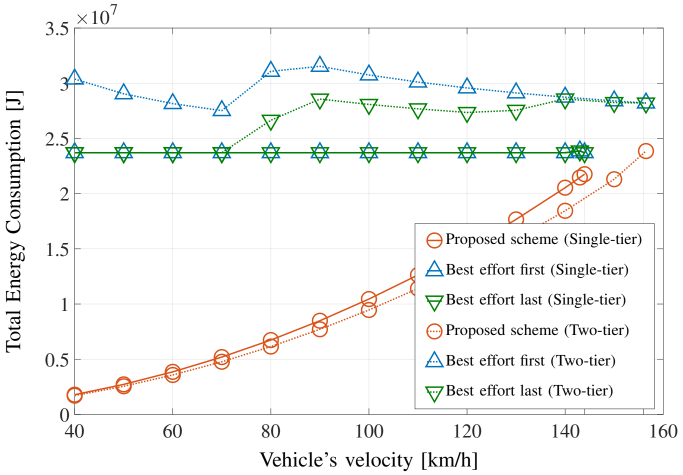

Fig. 3. Total energy consumption versus the vehicle's velocity when $D_{\text{bit}} =$ 250MB and $K = 20$.

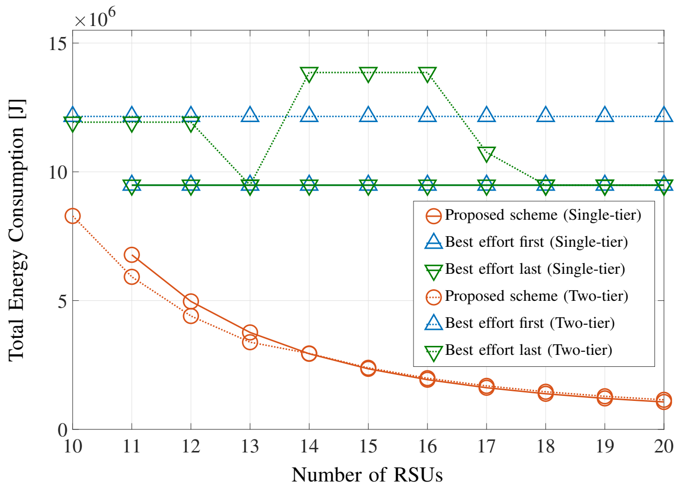

Fig. 4. Total energy consumption versus the number of RSUs when $D_{\text{bit}} =$ 100MB and $v =$ 100km/h.

Figure 2 shows the total energy consumption in the edge computing-enabled vehicular network versus the size of the computation result D_{bit}. We can see that our proposed scheme outperforms both baselines. In the single-tier network, when the total computation result exceeds 360MB, the problem is no longer feasible. Similarly, in the two-tier network, when the total computation result exceeds 391MB, the problem is also not feasible.

Figure 3 shows the total energy consumption in the edge computing-enabled vehicular network versus the vehicle's velocity v. Similar to Fig. 2, we can see that our proposed scheme consumes lower energy in both scenarios than the baseline schemes. Moreover, we can also see that there exists the maximum velocity of the vehicle that can be handled under the simulation setup.

Finally, Fig. 4 shows the total energy consumption in the edge computing-enabled vehicular network versus the number of RSUs K. We can see that our proposed scheme achieves the best performance compared to both baseline schemes. Moreover, we can see that there also exists the minimum number of RSUs so that the vehicle can be successfully served.

From Fig. 2, Fig. 3, and Fig. 4, we can see that, the total energy consumption of the proposed scheme increases with the size of the computation result and the vehicle's velocity, and decreases with the number of RSUs. The performance of each baseline scheme in the two-tier network is worse than that in the single-tier network, due to the existence of the macro RSUs which consumes higher energy and its heuristic way of splitting the task and allocate resource. Finally, when the problem is feasible, the proposed scheme, which is based on the optimal solution of Problem 1, is always better than the baseline schemes.

VI. Conclusion

This paper develops the energy-efficient computation task splitting scheme for an edge computing-enabled vehicular network. Specifically, after introducing the novel prescheduled task splitting and offloading scheme, we provide the closed form optimal solutions for the task splitting, the transmission power allocation, the CPU frequency allocation, and the transmission time allocation, which minimize the total energy consumption. Finally, through numerical results, we show that the total energy consumption of our proposed scheme increases with the size of the computation result and the vehicle's velocity, and decreases with the number of RSUs. We also show that there exist maximum computation result size, maximum vehicle's velocity, and minimum number of RSUs that the vehicle can be successfully served. In the feasible situations where the vehicle can be successfully served, we verify that our proposed scheme consumes lower energy than baseline schemes.

References

[1] "Mobile-edge computing introductory technical white paper," Dept. Mobile-Edge Comput. Ind. Initiative, ETSI, Sophia Antipolis, France, Sep. 2014, White Paper. [Online]. Available: https://portal.etsi.org/Portals/0/TBpages/MEC/Docs/Mobile-edge_Computing_-_Introductory_Technical_White_Paper_V1%2018-09-14.pdf

[2] A. Bozorgchenani, D. Tarchi, and G. E. Corazza, "Mobile edge computing partial offloading techniques for mobile urban scenarios," in *Proc. IEEE Global Telecomm. Conf.*, Abu Dhabi, United Arab Emirates, Dec. 2018, pp. 1–7.

[3] L. T. Tan and R. Q. Hu, "Mobility-aware edge caching and computing in vehicle networks: A deep reinforcement learning," *IEEE Trans. Veh. Technol.*, vol. 67, no. 11, pp. 10 190–10 203, Nov. 2018.

[4] Y. Sun, L. Xu, Y. Tang, and W. Zhuang, "Traffic offloading for online video service in vehicular networks: A cooperative approach," *IEEE Trans. Veh. Technol.*, vol. 67, no. 8, pp. 7630–7642, May 2018.

[5] 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.

[6] J. Guo, Z. Song, Y. Cui, Z. Liu, and Y. Ji, "Energy-efficient resource allocation for multi-user mobile edge computing," in *Proc. IEEE Global Telecomm. Conf.*, Singapore, Singapore, Dec. 2017, pp. 1–7.

[7] K. Wang, K. Yang, and C. S. Magurawalage, "Joint energy minimization and resource allocation in C-RAN with mobile cloud," *IEEE Trans. Cloud Comput.*, vol. 6, no. 3, pp. 760–770, Jul. 2018.

[8] N. Jindal, J. G. Andrews, and S. Weber, "Multi-antenna communication in ad hoc networks: Achieving MIMO gains with SIMO transmission," *IEEE Trans. Commun.*, vol. 59, no. 2, pp. 529–540, Feb. 2011.

[9] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge, UK: Cambridge University Press, 2004.