Energy-Efficient Cooperative Offloading for Edge Computing-Enabled Vehicular Networks

Hewon Cho, *Graduate Student Member, IEEE*, Ying Cui, *Member, IEEE*, and Jemin Lee, *Member, IEEE*

***Abstract*— Edge computing technology has great potential to improve various computation-intensive applications in vehicular networks by providing sufficient computation resources for vehicles. However, inappropriate task offloading to roadside units (RSUs) can lead to large energy consumption, which will result in negative economic, environmental, and performance impacts. Therefore, in this paper, we develop the energy-efficient cooperative offloading scheme for edge computing-enabled vehicular networks. We first establish a novel cooperative offloading model to multiple RSUs for given batch of moving vehicles, different from existing works that consider single vehicle only or static users. Then, we consider the total energy minimization by optimizing the task splitting ratio, computation resource, and communication resource, which is a challenging non-convex problem, and provide optimal solutions for multi-vehicle case and single-vehicle case, respectively. Furthermore, we extend our proposed scheme to the one for a more realistic scenario (i.e., online scenario), where batches of vehicles sequentially approach the RSUs. Finally, through numerical results, the impact of network parameters on the total energy consumption is analyzed, and we verify that our proposed solution consumes lower energy than baseline schemes.**

***Index Terms*— Vehicular networks, edge computing, task splitting, resource allocation, convex optimization.**

Manuscript received 31 October 2021; revised 7 March 2022 and 10 June 2022; accepted 10 June 2022. Date of publication 4 July 2022; date of current version 12 December 2022. This work was supported in part by the National Research Foundation of Korea (NRF) funded by the Korean Government (MSIP) under Grant 2020R1A2C2008878; in part by the Institute for Information & Communications Technology Promotion (IITP) funded by the Korean Government (MSIP) through the Development of Attack Response and Intelligent RSU Technology for Vehicle Security Threat Prevention under Grant 2021-0-01277; in part by the National Research Foundation of Korea (NRF) funded by the Korean Government (MSIT) under Grant NRF-2018R1A5A1060031; and in part by the Ministry of Science and ICT (MSIT), South Korea, under the Information Technology Research Center (ITRC) Support Program Supervised by the Institute of Information & Communications Technology Planning & Evaluation (IITP) under Grant IITP-2020-0-01795. An earlier version of this paper was presented in part at the IEEE International Conference on Communications, Dublin, Ireland, in June 2020 [DOI: 10.1109/ICCWorkshops49005.2020.9145345]. The associate editor coordinating the review of this article and approving it for publication was D. Niyato. *(Corresponding author: Jemin Lee.)*

Hewon Cho is with the Department of Electrical Engineering and Computer Science, Daegu Gyeongbuk Institute of Science and Technology, Daegu 42988, South Korea (e-mail: nb00040@dgist.ac.kr).

Ying Cui is with the Department of Electronic Engineering, Shanghai Jiao Tong University, Shanghai 200240, China, and also with the Henan Joint International Research Laboratory of Intelligent Networking and Data Analysis, School of Information Engineering, Zhengzhou University, Zhengzhou, Henan 450001, China.

Jemin Lee is with the Department of Electrical and Computer Engineering, Sungkyunkwan University (SKKU), Suwon 16419, Republic of Korea (e-mail: jemin.lee@skku.edu).

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TWC.2022.3186590.

Digital Object Identifier 10.1109/TWC.2022.3186590

I. Introduction

DUE to the rapid development of vehicular networks, various computation-intensive applications, including road safety applications, traffic efficiency applications, infotainment applications, etc., are emerging. Implementing such applications requires enormous computing resources for data processing [2]–[5]. However, the computing resource of a vehicle is usually limited, making it hard for the vehicle itself to complete a computation task by the service completion deadline. For tackling this issue, the edge computing technology has been regarded as a promising solution [6]–[11]. The edge computing technology is to make mobile users compute the large tasks by offloading them to the nearby edge computing servers such as base stations (BSs) and roadside units (RSUs) instead of the central server.

There have been extensive works on an optimal offloading design for static users in edge computing-enabled networks [12]–[21]. For instance, a static user offloads its task to one edge computing server [12]–[18] or multiple edge computing servers [19]–[21]. More specifically, by optimizing the offloading scheduling and the resource allocation for computation and communication such as transmission power and CPU frequency at the RSU side, the authors maximize the weighted sum of the offloading rates [12], minimize the total latency for computing and transmitting a task [13], [19], or minimize the total energy consumption for local computing and offloading [14]–[18], [20], [21]. However, since user mobility is not considered, the solutions for the static scenario in [12]–[21] may not be effective for vehicular networks, where vehicles move most of the time. For example, the solution obtained for the static scenario is no longer optimal when vehicles' mobility is considered.

Recently, various task offloading techniques have been presented for edge computing-enabled vehicular networks [22]–[28]. For example, by optimizing the offloading scheduling and resource allocation for communication and computation, the system utility is maximized in [22]–[25], the total latency is minimized in [26], [27], and the total energy consumption is minimized in [28]. Most of the works, including [22]–[28], assume that a task is offloaded to only one RSU (i.e., an edge computing server). However, in a practical vehicular network, a vehicle may have a short connection time to single RSU [29], which makes such assumption unrealistic. Therefore, when the size of the task is large and the velocity of a vehicle is high, offloading a task to single RSU is obviously not energy-efficient or even not feasible.

To tackle the issue caused by the short connection time to each RSU and the large size of the computation-intensive tasks, such as HTTP adaptive streaming, video transcoding, and local dynamic map, the cooperation among multiple RSUs has been recently considered in [30]. Specifically, optimal task splitting ratios that minimizes the latency have been obtained for multiple vehicles in [30]. However, the optimization of the resource allocation for computation and communication is not considered, which results in inefficient edge computing-enabled vehicular networks. Moreover, the energy efficiency is not considered in [30], although the cost from the energy consumption such as severe carbon pollution and expensive electricity bills is not negligible [31], [32]. Furthermore, none of the aforementioned works can be extended to a more realistic scenario where vehicles sequentially enter the network and their information is unknown ahead.

To address the limitations mentioned above, we jointly consider the task splitting and the resource allocation for computation and communication in edge computing-enabled vehicular networks. Specifically, for a computation-intensive task, we consider the cooperative offloading, which splits each vehicle's task into multiple subtasks and offloads them to different RSUs (i.e., edge computing servers) located ahead along the route of the vehicle.[1] Since the energy-efficient design is important in edge computing-enabled vehicular networks [14]–[18], [20], [21], [28], [31], [32], we find an optimal task splitting and resource allocation that minimizes the energy consumption for given batch of vehicles. We further extend the cooperative offloading models to the one for a more realistic scenario (i.e., online scenario) where batches of vehicles sequentially approach the RSUs and their information, including locations, velocities, and task parameters, is unknown ahead. The main contributions of this work can be summarized as below.

- We establish novel cooperative offloading models for given batch of vehicles in edge computing-enabled vehicular network. The models are much more complex than the ones in the static scenario as the subtasks of each vehicle are offloaded to multiple RSUs and multiple vehicles pass through the coverage area of each RSU. To the best of our knowledge, this is the first work providing energy-efficient cooperative offloading models for multiple moving vehicle scenario (i.e., the vehicles with different velocities and different locations at the time of scheduling).
- For given batch of vehicles, we formulate the total energy minimization with respect to the task splitting ratio, computation resource, and communication resource. It is a challenging non-convex problem with considerably more variables and constraints than the one for the static scenario. To address this issue, we equivalently transform the original non-convex problem to a convex problem and solve it with standard convex optimization algorithms.
- For the single-vehicle case, we further decompose the equivalent convex problem into several subproblems of much smaller sizes controlled by a master problem. We then provide the closed-form optimal solutions for the subproblems and semi-closed-form optimal solutions for the master problem.
- We extend the cooperative offloading models to the one for the online scenario, which is more challenging due to sequential arrivals of vehicles. Specifically, we formulate the total energy minimization for both new arriving vehicles and leftover vehicles (i.e., vehicles that still exist in the network when new arriving vehicles enter the network). We show that the underlying problem structure for the online scenario is similar to the one for the multi-vehicle case and can be solved using the same method.
- Finally, by numerical simulations, we analyze the impacts of vehicles' velocities and computation result sizes on the total energy consumption in the single-vehicle case and the multi-vehicle case, respectively. Furthermore, we show that the proposed solutions achieve significant gains in the total energy consumption over all baseline schemes.

The remainder of this paper is organized as follows. Section II describes the system model. Section III formulates the energy minimization problem for given batch of vehicles and proposes low complexity optimal solutions for both the multi-vehicle case and single-vehicle case. We then formulate the energy minimization problem in the online scenario and present its solution method in Section IV. Numerical results are provided in Section V. Finally, conclusions are given in Section VI.

Notation: The notation used throughout the paper is reported in Table I.

II. System Model

In this section, we present the network model and cooperative offloading model of an edge computing-enabled vehicular network when a batch of vehicles is known ahead. Then, in Section IV-A, we will extend the cooperative offloading models to the one for the online scenario where sequential batches of vehicles are unknown ahead.

A. Network Model

We consider an edge computing-enabled vehicular network, which consists of K multi-antenna RSUs, denoted by $\mathcal{K} \triangleq \{1, \cdots, K\}$, located along a unidirectional road, as shown in Fig. 1.[2] For given batch of vehicles, we consider U single-antenna vehicles, denoted by $\mathcal{U} \triangleq \{1, \cdots, U\}$. At time 0, the U vehicles enter the edge computing-enabled network and pass RSUs $1, 2, \cdots, K$ successively. Each RSU is equipped with an edge computing server. Thus, each RSU has both communication capability and computation capability. In reality,

[1]For example, in case of utilizing edge computing servers for HTTP adaptive streaming [33], the server stores the video content in a segmented form, called slices, and transfers them to multiple edge computing servers.

[2]Note that the number of RSUs, K, is determined by the delay requirement of the tasks requested by the vehicles. Specifically, for given the required delay τ, we have $K = \max\left\{\bar{k} | \sum_{i=1}^{\bar{k}} \frac{R_i}{v_u} \leq \tau\right\}$.

TABLE I
NOTATIONS USED THROUGHOUT THE PAPER

Notation	Definition
$a_{k,m}$	Index of the vehicle that is m-th one entering RSU k's coverage area
C_u	Size of workload of vehicle u's task
D_u	Size of computation result of vehicle u's task
F_k	Maximum computing capability of RSU k
$f_{k,u}$; $\mathbf{f}$	CPU frequency used for executing vehicle u's subtask k at RSU k; CPU frequency allocation
P_k	Maximum transmission power of RSU k
$p_{k,u}$; $\mathbf{p}$	transmission power used for transmitting vehicle u's subtask k at RSU k; Transmission power allocation
R_k	Length of the interval of the road covered by RSU k
$R_{\text{req},u}$	Distance from RSU 1 to the point where vehicle u requested the task
v_u	Velocity of vehicle u
$s_{\text{cp},k,u}$; $\mathbf{s}_{\text{cp}}$	Computation starting time of vehicle u's subtask k at RSU k; computation starting time allocation
$s_{\text{cm},k,u}$; $\mathbf{s}_{\text{cm}}$	Transmission starting time of vehicle u's subtask k at RSU k; transmission starting time allocation
$T^{(\text{a})}_{k,u}$; $T^{(\text{d})}_{k,u}$	Arrival time of vehicle u at RSU k; departure time of vehicle u at RSU k
$x_{k,u}$; $\mathbf{x}$	Fraction of task u that is executed at RSU k; task splitting factor
$t_{\text{cp},k,u}$; $\mathbf{t}_{\text{cp}}$	Computation time of vehicle u's subtask k at RSU k; computation time allocation
$t_{\text{cm},k,u}$; $\mathbf{t}_{\text{cm}}$	Transmission time of vehicle u's subtask k at RSU k; transmission time allocation
$E_{\text{cp},k,u}$	Computation energy consumption for executing vehicle u's subtask k at RSU k
$E_{\text{cm},k,u}$	Communication energy consumption for transmitting the computation result from RSU k to vehicle u
E_{tot}	Total energy consumption

the road does not always pass through the center of each RSU's coverage area. As a result, for each RSU $k \in \mathcal{K}$, 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. For tractability, we assume that the K road intervals covered by the K RSUs form a partition of the road.

Each vehicle $u \in \mathcal{U}$ moves at a constant velocity, denoted by v_u (in m/s), along the road, and has a computation-intensive task, called task u [34]. Due to limited local computation capability, each vehicle offloads its task and requests to obtain the result of its task before it leaves the edge computing-enabled vehicular network.[3] Assume that at time 0, each vehicle has sent the input data of its task to the edge computing-enabled network. Thus, we characterize a task by two parameters, i.e., the size of workload $C_u > 0$ (in number of CPU-cycles) and the size of the computation result $D_u > 0$ (in bits).

Let $R_{\text{req},u}$ (in meters) denote the distance to the first RSU in the network at time 0. The arrival time and the departure

[3]Our work can be extended by considering local computing at each vehicle.

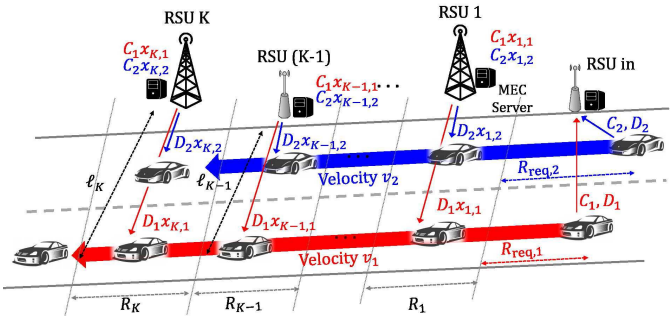


Fig. 1. Illustration of the edge computing-enabled vehicular networks.

time of vehicle u at RSU k are given as follows:

$$T^{(\text{a})}_{k,u} = \frac{1}{v_u}\left(R_{\text{req},u} + \sum_{i=1}^{k-1} R_i\right), \quad k \in \mathcal{K},\ u \in \mathcal{U}, \tag{1}$$

$$T^{(\text{d})}_{k,u} = \frac{1}{v_u}\left(R_{\text{req},u} + \sum_{i=1}^{k} R_i\right), \quad k \in \mathcal{K},\ u \in \mathcal{U}. \tag{2}$$

Let $a_{k,m}$ denote the index of the vehicle that is the m-th one entering RSU k's coverage area, where $m \in \mathcal{O} \triangleq \{1, \cdots, U\}$. Thus, $T^{(\text{a})}_{k,a_{k,1}} < T^{(\text{a})}_{k,a_{k,2}} < \cdots < T^{(\text{a})}_{k,a_{k,U}}$. As the U vehicles have different velocities and initial distances to RSU 1, the vehicle orders for different RSUs can differ.

B. Cooperative Offloading Model

Each vehicle may move very fast and single RSU can hardly finish the computation of its entire task before the vehicle leaves its coverage area. Hence, we consider cooperative offloading that makes K RSUs jointly compute the offloaded task.

1) Computation Model: Each vehicle's task is split into K subtasks of different sizes, which are offloaded to K RSUs. The size of each subtask is determined by the task splitting factor $\mathbf{x} \triangleq (x_{k,u})_{k\in\mathcal{K},u\in\mathcal{U}}$. Here, $x_{k,u}$ denotes the fraction of task u offloaded to RSU k, where

$$x_{k,u} \geq 0, \quad k \in \mathcal{K},\ u \in \mathcal{U}, \tag{3}$$

$$\sum_{k\in\mathcal{K}} x_{k,u} = 1, \quad u \in \mathcal{U}. \tag{4}$$

Note that when $x_{k,u} = 0$, no subtask is offloaded to RSU k. Assume that the sizes of the computation workload and the computation result of vehicle u's subtask at RSU k are $C_u x_{k,u}$ and $D_u x_{k,u}$, respectively.[4] In later sections, the task splitting will be optimized.

The CPU frequency of each RSU can be adjusted by using dynamic voltage and frequency scaling technology [20]. We denote $f_{k,u}$ as the CPU frequency used for executing vehicle u's subtask k at RSU k, where

$$f_{k,u} \geq 0, \quad k \in \mathcal{K},\ u \in \mathcal{U}, \tag{5}$$

$$f_{k,u} \leq F_k, \quad k \in \mathcal{K},\ u \in \mathcal{U}. \tag{6}$$

[4]Note that our work can be easily extended to the case where the size of the computation result is not proportional to the size of the computation workload.

Here, F_k represents the maximum computing capability (i.e., CPU frequency) of RSU k. Then, the computation energy consumption for executing vehicle u's subtask k at RSU k is given by [35]:

$$E_{\mathrm{cp},k,u}\left(x_{k,u}, f_{k,u}\right) = \kappa_k C_u x_{k,u} f_{k,u}^{\varphi_k - 1}, \quad k \in \mathcal{K},\ u \in \mathcal{U}, \tag{7}$$

where $\kappa_k > 0$ represents the effective switched capacitance depending on the chip architecture and $\varphi_k > 1$. Moreover, the computation time for executing vehicle u's subtask k at RSU k is $\frac{C_u x_{k,u}}{f_{k,u}}$.

For ease of implementation, we assume that RSU k processes U subtasks from U vehicles sequentially in the order of arrivals, i.e., $a_{k,1}, \cdots, a_{k,U}$.[5] We denote $s_{\mathrm{cp},k,a_{k,m}}$ as the computation starting time of the k-th subtask of the m-th vehicle entering RSU k's coverage area. Hence, we have the following computation constraints.

$$s_{\mathrm{cp},k,a_{k,m}} \geq 0, \quad k \in \mathcal{K},\ m \in \mathcal{O}, \tag{8}$$

$$s_{\mathrm{cp},k,a_{k,m}} + \frac{C_{a_{k,m}} x_{k,a_{k,m}}}{f_{k,a_{k,m}}} \leq s_{\mathrm{cp},k,a_{k,m+1}}, \quad k \in \mathcal{K},\ m \in \mathcal{O}\backslash\{U\}. \tag{9}$$

The constraints in (9) indicate that the computation of the m-th vehicle's subtask k at RSU k must complete before the start of the computation of the $(m+1)$-th vehicle's subtask k at RSU k. In other words, the constraints in (9) represent the waiting time constraint for $(m+1)$-th vehicle's task at RSU k. We assume that the computation of vehicle u's subtask at RSU k completes before vehicle u enters the coverage area of RSU k. According to the assumption, we have the following computation constraint.

$$s_{\mathrm{cp},k,a_{k,m}} + \frac{C_{a_{k,m}} x_{k,a_{k,m}}}{f_{k,a_{k,m}}} \leq T^{(\mathrm{a})}_{k,a_{k,m}}, \quad k \in \mathcal{K},\ m \in \mathcal{O}. \tag{10}$$

Note that this assumption is used to get tractable forms of the constraints on the communication and computation resources in this dynamic network environments with multiple vehicles.

2) Communication Model: After computing the k subtask of vehicle u, RSU k transmits the computation result to vehicle u when vehicle u is in its coverage area. Each RSU has M transmit antennas, and each vehicle has single transmit antenna. Thus, this corresponds to multi-input single-output (MISO) transmission. To avoid severe interference from nearby RSUs, we assume the exclusive channel usage among nearby RSUs.[6] Here, the interference from far away RSUs can exist, which can be treated as noise [36], [37]. Let g_k denote the large-scale fading power of the channel between RSU k and vehicle u with the maximum link length ℓ_k. We consider a narrow band slotted system of bandwidth B (in Hz) and adopt the block fading model for small-scale fading. Let $\mathbf{h}^{\mathrm{H}}_{k,u}$ denote the small-scale fading coefficients of the channel between RSU k and vehicle u in an arbitrary time slot, where $\mathbf{h}_{k,u} \in \mathcal{C}^{M\times 1}$. Here, $\mathbf{H}$ denotes conjugate transpose. Suppose that the elements of $\mathbf{h}_{k,u}$ are independently and identically distributed (i.i.d.) according to $\mathcal{CN}(0,1)$.

The transmission power limit of RSU k is denoted by P_k. We denote $p_{k,u}$ as the transmission power used for transmitting vehicle u's subtask k from RSU k to vehicle u, where

$$p_{k,u} \geq 0, \quad k \in \mathcal{K},\ u \in \mathcal{U}, \tag{11}$$

$$p_{k,u} \leq P_k, \quad k \in \mathcal{K},\ u \in \mathcal{U}. \tag{12}$$

Consider an arbitrary slot. The received signal of vehicle u from RSU k, denoted as $y_{k,u}$, is given by [38]:

$$y_{k,u} = \mathbf{h}^{\mathrm{H}}_{k,u} \mathbf{w}_{k,u} \sqrt{p_{k,u} g_k} s_{k,u} + z, \quad k \in \mathcal{K},\ u \in \mathcal{U}, \tag{13}$$

where $s_{k,u}$ is an information symbol with $\mathbb{E}\left[|s_{k,u}|^2\right] = 1$, $\mathbf{w}_{k,u} \in \mathcal{C}^{M\times 1}$ is a normalized beamforming vector with $\|\mathbf{w}_{k,u}\| = 1$, and z is the additive white Gaussian noise (AWGN) following the complex normal distribution $\mathcal{CN}(0, N_o)$. From (13), the received signal-to-noise ratio (SNR) of vehicle u is given by:

$$\mathrm{SNR}_{k,u} = \frac{p_{k,u} \left|\mathbf{h}_{k,u} \mathbf{w}_{k,u}\right|^2 g_k}{N_o}, \quad k \in \mathcal{K},\ u \in \mathcal{U}. \tag{14}$$

Assume that each RSU and vehicle have perfect channel state information (CSI) when the vehicle enters the RSU's coverage area. We consider maximal ratio transmission (MRT) beamforming, i.e., $\mathbf{w}_{k,u} = \frac{\mathbf{h}_{k,u}}{\|\mathbf{h}_{k,u}\|}$, which maximizes the received signal power at vehicle u [38]. The achievable instantaneous data rate for user u at RSU k is given by:

$$C_{k,u} = B \log\left(1 + \frac{p_{k,u} \left\|\mathbf{h}_{k,u}\right\|^2 g_k}{N_o}\right). \tag{15}$$

It is generally impossible to know the exact CSI during the resource allocation, as it is done before vehicles reach the coverages of RSUs. Hence, using the statistics of CSI (not the instantaneous CSI), we define the maximum target data rate $\bar{\epsilon}_{k,u}$ as the one guaranteeing the target successful transmission probability (STP) of vehicle u, θ_u, as[7]

$$\mathbb{P}\left[C_{k,u} \geq \bar{\epsilon}_{k,u}\right] \geq \theta_u, \quad k \in \mathcal{K},\ u \in \mathcal{U}. \tag{16}$$

From (16), the time to transmit the computation result of size $D_u x_{k,u}$ to vehicle u from RSU k, denoted by $t_{\mathrm{cm},k,u}$, is determined based on $\bar{\epsilon}_{k,u}$ as

$$t_{\mathrm{cm},k,u} = \frac{D_u x_{k,u}}{\bar{\epsilon}_{k,u}}, \quad k \in \mathcal{K},\ u \in \mathcal{U}. \tag{17}$$

$$t_{\mathrm{cm},k,u} \geq 0, \quad k \in \mathcal{K},\ u \in \mathcal{U}. \tag{18}$$

[5]Note that without changing the underlying problem structure, our work can be easily extended to the case where each RSU consists of U queues and U virtual servers (like multi-core CPU systems) to process U subtasks simultaneously. However, such modeling results in longer computation latency compared to the current single-queue single-server model.

[6]If there exists the interference from nearby RSUs, the transmission power of one RSU affects the interference experienced by vehicles in the coverage of the other RSUs. As a result, the problems cannot be transformed into convex problems and hence do not have efficient solutions. However, the proposed solution framework can be used to approximately solve the problem for the scenario with interference by exploiting the average maximum interference power determined by assuming all RSUs transmit with the transmission power limit P_k, and incorporating it into the noise power.

[7]Note that the retransmission from communication failure is not considered, since it can be ignored by setting high θ_u (i.e., ignorable communication failures).

Since $\|\mathbf{h}_{k,u}\|^2 \sim \Gamma(M,1)$, (16) is equivalent to:

$$\mathbb{P}\left[\|\mathbf{h}_{k,u}\|^2 \geq \frac{N_o}{p_{k,u}g_k}\left(2^{\frac{D_u x_{k,u}}{Bt_{\mathrm{cm},k,u}}}\right)\right] = G\left(\frac{N_o}{p_{k,u}g_k}\left(2^{\frac{D_u x_{k,u}}{Bt_{\mathrm{cm},k,u}}}\right)\right) \geq \theta_u, \quad (19)$$

where $G(y) \triangleq \sum_{n=0}^{M-1} \frac{1}{n!} y^n e^{-y}$ denotes the complementary cumulative distribution function (CCDF) of $\Gamma(M,1)$.

The transmission of the computation result from an RSU to a vehicle cannot start before the vehicle enters the coverage area of the RSU and must complete before the vehicle leaves the coverage area of the RSU. We denote $s_{\mathrm{cm},k,a_{k,m}}$ as the transmission starting time of the k-th subtask of the m-th vehicle entering RSU k's coverage area, respectively, where

$$s_{\mathrm{cm},k,a_{k,m}} \geq T^{(\mathrm{a})}_{k,a_{k,m}}, \quad k \in \mathcal{K}, \ m \in \mathcal{O}, \quad (20)$$

$$s_{\mathrm{cm},k,a_{k,m}} + t_{\mathrm{cm},k,a_{k,m}} \leq T^{(\mathrm{d})}_{k,a_{k,m}}, \quad k \in \mathcal{K}, \ m \in \mathcal{O}. \quad (21)$$

Similar to the computation model, it is assumed that RSU k transmits the U subtasks of the U vehicles sequentially in the order of arrival of the U vehicles. Hence, we have the following communication constraints.

$$s_{\mathrm{cm},k,a_{k,m}} + t_{\mathrm{cm},k,a_{k,m}} \leq s_{\mathrm{cm},k,a_{k,m+1}}, \quad k \in \mathcal{K}, \quad m \in \mathcal{O}\backslash\{U\}. \quad (22)$$

The constraints in (22) indicate that RSU k starts the transmission of subtask k of the $(m+1)$-th vehicle entering its coverage area after finishing the transmission of subtask k of the m-th vehicle entering its coverage area. Besides, the communication energy consumption for transmitting the computation result from RSU k to vehicle u is given by:

$$E_{\mathrm{cm},k,u}(p_{k,u}, t_{\mathrm{cm},k,u}) = p_{k,u} t_{\mathrm{cm},k,u}, \quad k \in \mathcal{K}, \ u \in \mathcal{U}. \quad (23)$$

3) Total Energy Consumption: The total energy consumed at the K RSUs for serving the U vehicles is given by:

$$E_{\mathrm{tot}}(\mathbf{x},\mathbf{f},\mathbf{p},\mathbf{t}_{\mathrm{cm}}) = \sum_{k\in\mathcal{K}}\sum_{u\in\mathcal{U}}\left(E_{\mathrm{cp},k,u}(x_{k,u}, f_{k,u}) + E_{\mathrm{cm},k,u}(p_{k,u}, t_{\mathrm{cm},k,u})\right) = \sum_{k\in\mathcal{K}}\sum_{u\in\mathcal{U}}\left(\kappa_k C_u x_{k,u} f_{k,u}^{\varphi_k - 1} + p_{k,u} t_{\mathrm{cm},k,u}\right), \quad (24)$$

where $\mathbf{x} \triangleq (x_{k,u})_{k\in\mathcal{K},u\in\mathcal{U}}$, $\mathbf{f} \triangleq (f_{k,u})_{k\in\mathcal{K},u\in\mathcal{U}}$, $\mathbf{p} \triangleq (p_{k,u})_{k\in\mathcal{K},u\in\mathcal{U}}$, $\mathbf{t}_{\mathrm{cm}} \triangleq (t_{\mathrm{cm},k,u})_{k\in\mathcal{K},u\in\mathcal{U}}$, and the last equation is due to (7) and (23).

We assume that all RSUs are connected to a controller, which is aware of the network parameters, M, K, P_k, F_k, R_k, ℓ_k, g_k, $k \in \mathcal{K}$. For given batch of vehicles, we further assume that the controller also knows the vehicle parameters, i.e., U, v_u, $R_{\mathrm{req},u}$, $C_u x_{k,u}$, and $D_u x_{k,u}$, at time 0. Under the above assumptions, the controller is aware of the expression of $E_{\mathrm{tot}}(\mathbf{x},\mathbf{f},\mathbf{p},\mathbf{t}_{\mathrm{cm}})$ at time 0.

Remark 1 (Differences Between Proposed Cooperative Offloading Model and Cooperative Offloading Model in [30]): The merits of the proposed cooperative offloading model compared to the existing one [30] are summarized as follows. Firstly, the task splitting and computation and communication resource allocation are jointly considered in the proposed model, whereas only task splitting is considered in [30]. Secondly, the limitation on the communication capability is considered in the proposed model via (11), (12), (18)–(22), but not in [30]. Thirdly, it is guaranteed to complete the execution of each subtask at each RSU before the corresponding vehicle enters the coverage area of the RSU in the proposed model via (8)–(10), but not in [30] (where outage of task is allowed). Finally, the computation and communication operation orders for the subtasks of all moving vehicles are reflected in the newly specified constraints (9), (10), (21), and (22), but not in [30]. To the best of our knowledge, this is the first cooperative offloading model for edge computing-enabled vehicular networks, which reflects the computation and communication resource limitation, allows efficient utilization of computation and communication resources, and provides performance guarantees for computation-intensive applications.

III. Total Energy Minimization

In this section, we consider the total energy minimization for given batch of vehicles. First, we formulate the total energy minimization as a non-convex problem. Then, we obtain a globally optimal solution for the multi-vehicle case. Finally, we obtain a globally optimal solution for the single-vehicle case.

A. Problem Formulation

For given batch of vehicles, we would like to minimize the total energy consumption $E_{\mathrm{tot}}(\mathbf{x},\mathbf{f},\mathbf{p},\mathbf{t}_{\mathrm{cm}})$ in (24) by optimizing the task splitting factor $\mathbf{x}$, the CPU frequency allocation $\mathbf{f}$, the transmission power allocation $\mathbf{p}$, the transmission time allocation $\mathbf{t}_{\mathrm{cm}}$, the computation starting time allocation $\mathbf{s}_{\mathrm{cp}} \triangleq (s_{\mathrm{cp},k,u})_{k\in\mathcal{K},u\in\mathcal{U}}$, and the transmission starting time allocation $\mathbf{s}_{\mathrm{cm}} \triangleq (s_{\mathrm{cm},k,u})_{k\in\mathcal{K},u\in\mathcal{U}}$. The optimization problem is formulated as follows.

Problem 1 (Total Energy Minimization):

$$E^*_{\mathrm{tot}} \triangleq \min_{\mathbf{x},\mathbf{f},\mathbf{p},\mathbf{t}_{\mathrm{cm}},\mathbf{s}_{\mathrm{cp}},\mathbf{s}_{\mathrm{cm}}} E_{\mathrm{tot}}(\mathbf{x},\mathbf{f},\mathbf{p},\mathbf{t}_{\mathrm{cm}}) \quad \text{s.t. } (3),(4),(5),(6),(8),(9),(10),(11),(12),(18),(19),(20),(21),(22),$$

where $E_{\mathrm{tot}}(\mathbf{x},\mathbf{f},\mathbf{p},\mathbf{t}_{\mathrm{cm}})$ is given by (24). Let $(\mathbf{x}^*,\mathbf{f}^*,\mathbf{p}^*,\mathbf{t}^*_{\mathrm{cm}},\mathbf{s}^*_{\mathrm{cp}},\mathbf{s}^*_{\mathrm{cm}})$ denote an optimal solution of Problem 1. Note that the constraints in (8), (9), (10), (20), (21), and (22) are affected by the mobilities of all vehicles (e.g., the velocities and the locations of all vehicles at the time of scheduling).

Given the network parameters and the vehicle parameters, the controller can solve Problem 1. In Problem 1, the objective function is non-convex, the inequality constraint functions in (9), (10), and (19) are non-convex, the inequality constraint functions in (3), (5), (6), (8), (11), (12), (18), (20), (22), and (21) are convex, and the equality constraint function in (4) is affine. Therefore, Problem 1 is a non-convex problem with $6KU$ variables and $(13KU - 2K + U)$ constraints. In general, it is hard to obtain a globally optimal solution for a non-convex problem analytically or numerically with

effective and efficient methods. Besides, Problem 1 is more challenging than those for cooperative offloading in the static scenario [12]–[21] and the vehicular networks [30]. This is because Problem 1 involves considerably more variables and constraints to specify the limitation on the computation and communication capability, and the computation and communication operation orders for the subtasks of all moving vehicles. In Section III-B and Section III-C, we solve Problem 1 in the multi-vehicle case $(U > 1)$ and single-vehicle case $(U = 1)$, respectively. Specifically, in each case, we transform Problem 1 into a convex problem with fewer variables and constraints by characterizing and utilizing an optimality property and obtain a globally optimal solution using convex optimization techniques.

B. Optimal Solution in Multi-Vehicle Case

In this subsection, we consider the multi-vehicle case, i.e., $U > 1$. First, we characterize an optimality property of Problem 1.

Lemma 1 (Optimal Transmission Power Allocation): An optimal solution of Problem 1 satisfies:

$$p_{k,u}^{*} = \frac{N_o}{g_k G^{-1}(\theta_u)} \left(2^{\frac{D_u x_{k,u}^{*}}{B t_{\mathrm{cm},k,u}^{*}}} - 1 \right), \quad k \in \mathcal{K},\ u \in \mathcal{U}, \tag{25}$$

where $G^{-1}(\cdot)$ denotes the inverse function of $G(\cdot)$.[8]

Proof: We equivalently convert the constraints in (19) to:

$$p_{k,u} \geq \frac{N_o}{g_k G^{-1}(\theta_u)} \left(2^{\frac{D_u x_{k,u}}{B t_{\mathrm{cm},k,u}}} - 1 \right), \quad k \in \mathcal{K},\ u \in \mathcal{U}. \tag{26}$$

Besides, for given $x_{k,u}$, $f_{k,u}$, $t_{\mathrm{cm},k,u}$, $s_{\mathrm{cp},k,u}$, and $s_{\mathrm{cm},k,u}$, $E_{\mathrm{tot}}(\mathbf{x}, \mathbf{f}, \mathbf{p}, \mathbf{t}_{\mathrm{cm}})$ in (24) increases with $p_{k,u}$, $k \in \mathcal{K}, u \in \mathcal{U}$. Thus, by contradiction, we can show that the inequality constraints in (26) are active at $(\mathbf{x}^{*}, \mathbf{f}^{*}, \mathbf{p}^{*}, \mathbf{t}_{\mathrm{cm}}^{*}, \mathbf{s}_{\mathrm{cp}}^{*}, \mathbf{s}_{\mathrm{cm}}^{*})$. Therefore, we complete the proof of Lemma 1. ∎

Next, we equivalently convert the non-convex problem in Problem 1 to a convex problem with fewer variables and constraints by changing variables and using Lemma 1. We introduce the computation time for executing vehicle u's subtask k with the workload of size $C_u x_{k,u}$ at RSU k, denoted by $t_{\mathrm{cp},k,u}$, which satisfies:

$$t_{\mathrm{cp},k,u} = \frac{C_u x_{k,u}}{f_{k,u}}, \quad k \in \mathcal{K},\ u \in \mathcal{U}. \tag{27}$$

Let $\mathbf{t}_{\mathrm{cp}} \triangleq (t_{\mathrm{cp},k,u})_{k \in \mathcal{K}, u \in \mathcal{U}}$. Define

$$\widetilde{E}_{\mathrm{tot}}(\mathbf{x}, \mathbf{t}_{\mathrm{cp}}, \mathbf{t}_{\mathrm{cm}}) \triangleq \sum_{k=1}^{K} \sum_{u=1}^{U} \Big(\widetilde{E}_{\mathrm{cp},k,u}(x_{k,u}, t_{\mathrm{cp},k,u}) + \widetilde{E}_{\mathrm{cm},k,u}(x_{k,u}, t_{\mathrm{cm},k,u}) \Big), \tag{28}$$

where

$$\widetilde{E}_{\mathrm{cp},k,u}(x_{k,u}, t_{\mathrm{cp},k,u}) \triangleq \kappa_k C_u^{\varphi_k} x_{k,u}^{\varphi_k} t_{\mathrm{cp},k,u}^{1-\varphi_k}, \tag{29}$$

$$\widetilde{E}_{\mathrm{cm},k,u}(x_{k,u}, t_{\mathrm{cm},k,u}) \triangleq \frac{N_o t_{\mathrm{cm},k,u}}{g_k G^{-1}(\theta_u)} \left(2^{\frac{D_u x_{k,u}}{B t_{\mathrm{cm},k,u}}} - 1 \right). \tag{30}$$

By (27) and Lemma 1, we can transform Problem 1 into the following problem with $5KU$ variables and $(9KU - K + U)$ constraints.

Problem 2 (Equivalent Problem of Problem 1 in Multi-Vehicle Case):

$$\begin{aligned}
\widetilde{E}_{\mathrm{tot}}^{*} \triangleq \min_{\substack{\mathbf{x}, \mathbf{t}_{\mathrm{cp}}, \mathbf{t}_{\mathrm{cm}}, \\ \mathbf{s}_{\mathrm{cp}}, \mathbf{s}_{\mathrm{cm}}}} \ & \widetilde{E}_{\mathrm{tot}}(\mathbf{x}, \mathbf{t}_{\mathrm{cp}}, \mathbf{t}_{\mathrm{cm}}) \\
\text{s.t. } & (3), (4), (8), (20), (21), (22), \\
& t_{\mathrm{cm},k,u} - \frac{D_u x_{k,u}}{B \log_2 \left(1 + \frac{P_k g_k G^{-1}(\theta_u)}{N_o} \right)} \geq 0, \\
& k \in \mathcal{K},\ u \in \mathcal{U}, && (31) \\
& x_{k,u} - \frac{F_k}{C_u} t_{\mathrm{cp},k,u} \leq 0, \quad k \in \mathcal{K},\ u \in \mathcal{U}, && (32) \\
& s_{\mathrm{cp},k,a_{k,m-1}} + t_{\mathrm{cp},k,a_{k,m-1}} \leq s_{\mathrm{cp},k,a_{k,m}}, \\
& k \in \mathcal{K},\ m \in \mathcal{O}, && (33) \\
& s_{\mathrm{cp},k,a_{k,m}} + t_{\mathrm{cp},k,a_{k,m}} \leq T_{k,a_{k,m}}^{(\mathrm{a})}, \\
& k \in \mathcal{K},\ m \in \mathcal{O}, && (34)
\end{aligned}$$

where $\widetilde{E}_{\mathrm{tot}}(\mathbf{x}, \mathbf{t}_{\mathrm{cp}}, \mathbf{t}_{\mathrm{cm}})$ is given by (28). Let $\left(\mathbf{x}^{\dagger}, \mathbf{t}_{\mathrm{cp}}^{\dagger}, \mathbf{t}_{\mathrm{cm}}^{\dagger}, \mathbf{s}_{\mathrm{cp}}^{\dagger}, \mathbf{s}_{\mathrm{cm}}^{\dagger} \right)$ denote an optimal solution of Problem 2.

Then, we show that Problem 1 is equivalent to Problem 2 with fewer variables and constraints.

Theorem 1 (Equivalence Between Problem 1 and Problem 2 in Multi-Vehicle Case): In the multi-vehicle case, the solutions of Problem 1 and Problem 2 satisfy:

$$\mathbf{x}^{\dagger} = \mathbf{x}^{*}, \quad \mathbf{t}_{\mathrm{cp}}^{\dagger} = \left(\frac{C_u x_{k,u}^{*}}{f_{k,u}^{*}} \right)_{k \in \mathcal{K}, u \in \mathcal{U}}, \quad \mathbf{t}_{\mathrm{cm}}^{\dagger} = \mathbf{t}_{\mathrm{cm}}^{*},$$
$$\mathbf{s}_{\mathrm{cp}}^{\dagger} = \mathbf{s}_{\mathrm{cp}}^{*}, \quad \mathbf{s}_{\mathrm{cm}}^{\dagger} = \mathbf{s}_{\mathrm{cm}}^{*}.$$

Furthermore, the optimal values of the two problems are identical, i.e., $E_{\mathrm{tot}}^{*} = E_{\mathrm{tot}}^{\dagger}$.

Proof: See Appendix A. ∎

Based on Theorem 1, we can solve Problem 2 instead of Problem 1 without loss of optimality [40]. All constraints in Problem 2 are convex. In the following lemma, we show that the objective function of Problem 2 is also convex.

Lemma 2 (Convexity of Objective Function of Problem 2): $\widetilde{E}_{\mathrm{tot}}(\mathbf{x}, \mathbf{t}_{\mathrm{cp}}, \mathbf{t}_{\mathrm{cm}})$ in (28) is convex in $(\mathbf{x}, \mathbf{t}_{\mathrm{cp}}, \mathbf{t}_{\mathrm{cm}})$.

Proof: Define $c_{k,u}^{(\mathrm{cp})}(y) = \kappa_k C_u^{\varphi_k} y^{\varphi_k}$, which is a convex function of y. Since the perspective operation preserves convexity, $\widetilde{E}_{\mathrm{cp},k,u}(x_{k,u}, t_{\mathrm{cp},k,u}) = t_{\mathrm{cp},k,u} c_{k,u}^{(\mathrm{cp})}(x_{k,u}/t_{\mathrm{cp},k,u})$ is also a convex function of $(x_{k,u}, t_{\mathrm{cp},k,u})$. Similarly, we can show that $\widetilde{E}_{\mathrm{cm},k,u}(x_{k,u}, t_{\mathrm{cm},k,u})$ is a convex function of $(x_{k,u}, t_{\mathrm{cp},k,u})$. Therefore, $\widetilde{E}_{\mathrm{tot}}(\mathbf{x}, \mathbf{t}_{\mathrm{cp}}, \mathbf{t}_{\mathrm{cm}})$ is a convex function of $(x_{k,u}, t_{\mathrm{cp},k,u})$. ∎

By Lemma 2, we know that Problem 2 is a convex problem, and hence can be solved optimally using an interior-point method of computational complexity $\mathcal{O}\left(K^3 U^3\right)$.

[8] $G^{-1}(\cdot)$ can be numerically computed [39].

C. Optimal Solution in Single-Vehicle Case

In this subsection, we consider the single-vehicle case, i.e., $U=1$. Note that in the single-vehicle case, we remove index u for simplicity. First, consider the following problem with $4K$ variables and $(9K+1)$ constraints.

Problem 3 (Equivalent Problem of Problem 1 in Single-Vehicle Case):

$$E^{\star}_{\text{single}} \triangleq \min_{\mathbf{x},\mathbf{f},\mathbf{p},\mathbf{t}_{\text{cm}}} E_{\text{tot}}\left(\mathbf{x},\mathbf{f},\mathbf{p},\mathbf{t}_{\text{cm}}\right)$$
$$\text{s.t. } (3),(4),(5),(6),(11),(12),(18),(19),$$
$$Cx_k - f_k T^{(\text{a})}_k \le 0, \quad k\in\mathcal{K} \tag{35}$$
$$t_{\text{cm},k} \le T^{(\text{d})}_k - T^{(\text{a})}_k, \quad k\in\mathcal{K}. \tag{36}$$

Let $(\mathbf{x}^{\star},\mathbf{f}^{\star},\mathbf{p}^{\star},\mathbf{t}^{\star}_{\text{cm}})$ denote an optimal solution of Problem 3.[9]

We can show that Problem 1 is equivalent to Problem 3 with fewer variables and constraints.

Theorem 2 (Equivalence Between Problem 1 and Problem 3 in Single-Vehicle Case): In the single-vehicle case, $s^{*}_{\text{cp},k}=0$ and $s^{*}_{\text{cm},k}=T^{(\text{a})}_k$, and the optimal solutions of Problem 1 and Problem 3 satisfy:

$$\mathbf{x}^{\star}=\mathbf{x}^{*}, \quad \mathbf{f}^{\star}=\mathbf{f}^{*}, \quad \mathbf{p}^{\star}=\mathbf{p}^{*}, \quad \mathbf{t}^{\star}_{\text{cm}}=\mathbf{t}^{*}_{\text{cm}}. \tag{37}$$

Proof: See Appendix B. ∎

Adopting the primal decomposition method, we decompose Problem 3 into several subproblems, each with 1 variable and 2 constraints, controlled by a master problem with K variables and $(3K+1)$ constraints [41].

Problem 4 (Master Problem - Task Splitting):

$$E^{\ddagger}_{\text{single}} \triangleq \min_{\mathbf{x}} \sum_{k=1}^{K}\left(\kappa_k C^{\varphi_k}\left(T^{(\text{a})}_k\right)^{1-\varphi_k} x_k^{\varphi_k} + E^{\ddagger}_{\text{cm},k}\left(x_k\right)\right)$$
$$\text{s.t. } (3),(4),$$
$$x_k \le \frac{F_k T^{(\text{a})}_k}{C}, \quad k\in\mathcal{K}, \tag{38}$$
$$x_k \le \frac{B\left(T^{(\text{d})}_k - T^{(\text{a})}_k\right)\log_2\left(1+\frac{P_k g_k G^{-1}(\theta)}{N_o}\right)}{D}, \quad k\in\mathcal{K}. \tag{39}$$

Let $\mathbf{x}^{\ddagger}$ denote an optimal solution of Problem 4.

Problem 5 (Subproblem - Power Allocation at RSU $k\in\mathcal{K}$): For any x_k,

$$E^{\ddagger}_{\text{cm},k}\left(x_k\right) \triangleq \min_{p_k} \frac{Dx_k p_k}{B\log_2\left(1+\frac{p_k g_k}{N_o}G^{-1}(\theta)\right)} \tag{40}$$
$$\text{s.t. } (12),$$
$$p_k \ge \frac{N_o}{g_k G^{-1}(\theta)}\left(2^{\frac{Dx_k}{B\left(T^{(\text{d})}_k - T^{(\text{a})}_k\right)}}-1\right). \tag{41}$$

Let $p^{\ddagger}_k\left(x_k\right)$ denote an optimal solution of Problem 5.

We can show that Problem 3 is equivalent to Problem 4 and Problem 5 which have fewer variables and constraints.

[9]Note that by replacing $T^{(\text{a})}_k$ in Problem 3 with $T^{(\text{a})}_k - T^{(\text{w})}_k$, the waiting time for the vehicle's subtask at RSU k, denoted by $T^{(\text{w})}_k$, can also be considered.

Theorem 3 (Equivalence Between Problem 3 and Problems 4 and 5): The solution of Problem 3 and the solution of Problem 4 and Problem 5 satisfy:

$$\mathbf{x}^{\star}=\mathbf{x}^{\ddagger}, \quad \mathbf{p}^{\star}=\left(p^{\ddagger}_k\left(x^{\ddagger}_k\right)\right)_{k\in\mathcal{K}}, \quad \mathbf{f}^{\star}=\left(\frac{Cx^{\ddagger}_k}{T^{(\text{a})}_k}\right)_{k\in\mathcal{K}},$$
$$\mathbf{t}^{\star}_{\text{cm}}=\left(\frac{Dx^{\ddagger}_k}{B\log_2\left(1+\frac{g_k p^{\ddagger}_k\left(x^{\ddagger}_k\right)}{N_o}G^{-1}(\theta)\right)}\right)_{k\in\mathcal{K}}.$$

Furthermore, the optimal values of the two problems are identical, i.e., $E^{*}_{\text{tot}}=E^{\star}_{\text{single}}$.

Proof: See Appendix C. ∎

Due to the equivalence shown in Theorem 3, we also use $\mathbf{x}^{*}$ and $\left(p^{*}_k\left(x_k\right)\right)_{k\in\mathcal{K}}$ to represent the optimal solutions of Problem 4 and Problem 5, respectively, with a slight abuse of notation. Furthermore, based on Theorem 3, we can obtain the solution of Problem 3 by solving Problem 4 and Problem 5 instead without loss of optimality [40].

First, we obtain a closed-form expression of an optimal solution for each subproblem in Problem 5.

Lemma 3 (Optimal Solution of Problem 5):

$$p^{\ddagger}_k\left(x_k\right)=\frac{N_o}{g_k G^{-1}(\theta)}\left(2^{\frac{Dx_k}{B\left(T^{(\text{d})}_k - T^{(\text{a})}_k\right)}}-1\right), \quad k\in\mathcal{K}. \tag{42}$$

Proof: For given x_k, the objective function decreases with p_k. Thus, by contradiction, we can show that the inequality constraint in (41) is active at p^{*}_k. Therefore, we complete the proof of Lemma 3. ∎

Next, we solve Problem 4. By (40) and Lemma 3, we have:

$$E^{\ddagger}_{\text{cm},k}\left(x_k\right)=\frac{N_o\left(T^{(\text{d})}_k - T^{(\text{a})}_k\right)}{g_k G^{-1}(\theta)}\left(2^{\frac{Dx_k}{B\left(T^{(\text{d})}_k - T^{(\text{a})}_k\right)}}-1\right). \tag{43}$$

Since the objective function is convex and all constraints are affine functions, Problem 4 is convex. As strong duality holds for Problem 4, we can obtain a semi-closed-form optimal solution of Problem 4 using Karush-Kuhn-Tucker (KKT) conditions [40], which is summarized as follows.

Lemma 4 (Optimal Solution of Problem 4):

$$x^{\ddagger}_k=\min\left[\frac{B\left(T^{(\text{d})}_k - T^{(\text{a})}_k\right)}{D}\log_2\left(1+\frac{P_k g_k G^{-1}(\theta)}{N_o}\right), \frac{F_k T^{(\text{a})}_k}{C}, \max\left\{H^{-1}_k\left(\gamma^{*}\right), 0\right\}\right], \quad k\in\mathcal{K}, \tag{44}$$

where γ^{*} satisfies $\sum_{k=1}^{K} x^{\ddagger}_k = 1$ with $x^{\ddagger}_k$ given by (44) and $H^{-1}_k(\cdot)$ is the inverse function of $H_k(\cdot)$ given by:

$$H_k(x)\triangleq \varphi_k\kappa_k C^{\varphi_k}\left(T^{(\text{a})}_k\right)^{1-\varphi_k} x^{\varphi_k-1} + \frac{DN_o\ln(2)}{g_k B G^{-1}(\theta)}2^{\frac{Dx}{B\left(T^{(\text{d})}_k - T^{(\text{a})}_k\right)}}. \tag{45}$$

Proof: See Appendix D. ∎

Since $\varphi_k \geq 1$ in (45), the derivative of $H_k(x)$ is positive, and therefore, $H_k(x)$ is an increasing function. As $H_k(x)$ is strictly increasing with x, $H_k^{-1}(\gamma^*)$ is also strictly increasing with γ^*. As $H_k^{-1}(\gamma^*)$ is strictly increasing with γ^*, we can easily obtain γ^* satisfying $\sum_{k=1}^{K} x_k^{\ddagger} = 1$ using the bisection method. Therefore, based on Lemma 4, we can readily obtain the semi-closed-form optimal solution of Problem 4.

From Lemma 4, we can obtain the closed-form optimal solution for a homogeneous setup with $R_k = R$, $\theta_k = \theta$, $g_k = g$, $\kappa_k = \kappa$, $\varphi_k = \varphi$, $k \in \mathcal{K}$ as follows.

Corollary 1 (Optimal Solution of Problem 4 for Homogeneous Setup): If $F_k \geq \frac{CH_k^{-1}(\gamma^*)}{T_k^{(\mathrm{a})}}$ and $P_k \geq \frac{N_o}{gG^{-1}(\theta)}\left(2^{\frac{DH_k^{-1}(\gamma^*)}{B\left(T_k^{(\mathrm{d})}-T_k^{(\mathrm{a})}\right)}}-1\right)$, the optimal solution is given by:

$$x_k^{\ddagger} = \max\left\{\tilde{H}_k^{-1}(\gamma^*),\ 0\right\}, \quad k \in \mathcal{K} \tag{46}$$

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

$$\tilde{H}_k(x) \triangleq \varphi\kappa C^{\varphi}\left(\frac{v}{R_{\mathrm{req}}+(k-1)R}\right)^{\varphi-1} x^{\varphi-1} + \frac{DN_o\ln(2)}{gBG^{-1}(\theta)}2^{\frac{vx}{BR}}. \tag{47}$$

Proof: First, we prove that $x_k^{\ddagger}$ is represented by (46). If $P_k \geq \frac{N_o}{gG^{-1}(\theta)}\left(2^{\frac{DH_k^{-1}(\gamma^*)}{B\left(T_k^{(\mathrm{d})}-T_k^{(\mathrm{a})}\right)}}-1\right)$, then $\min\left[\frac{B\left(T_k^{(\mathrm{d})}-T_k^{(\mathrm{a})}\right)}{D}\log_2\left(1+\frac{P_k gG^{-1}(\theta)}{N_o}\right), \max\left\{H_k^{-1}(\gamma^*), 0\right\}\right] = \max\left\{H_k^{-1}(\gamma^*),\ 0\right\}$. Moreover, if $F_k \geq \frac{CH_k^{-1}(\gamma^*)}{T_k^{(\mathrm{a})}}$, then $\min\left[\frac{F_k T_k^{(\mathrm{a})}}{C},\ \max\left\{H_k^{-1}(\gamma^*),\ 0\right\}\right] = \max\left\{H_k^{-1}(\gamma^*),\ 0\right\}$. Therefore, by (44), we have (46). Therefore, we complete the proof of Corollary 1. ∎

Analogously, by Corollary 1, we can readily obtain the semi-closed-form optimal solution of Problem 4 using the bisection methods.

We can solve K subproblems by Lemma 3, each with computational complexity $\mathcal{O}(1)$. The master problem can be solved by Lemma 4 with computational complexity $\mathcal{O}(K)$. Therefore, in the single-vehicle case, the overall computational complextiy for solving Problems 4 and 5 is $\mathcal{O}(K)$, which is lower than that in the multi-vehicle case.

IV. Extension to Online Scenario

So far, the proposed solutions work only for the offline scenario where a batch of vehicles is known ahead. However, in reality, sequential batches of vehicles enter the network and their information is usually unknown ahead. In this section, we extend the cooperative offloading models to the one for the online scenario by considering two types of vehicles, new arriving vehicles and leftover vehicles. Then, we formulate the energy minimization problem in the online scenario and discuss its solution method.

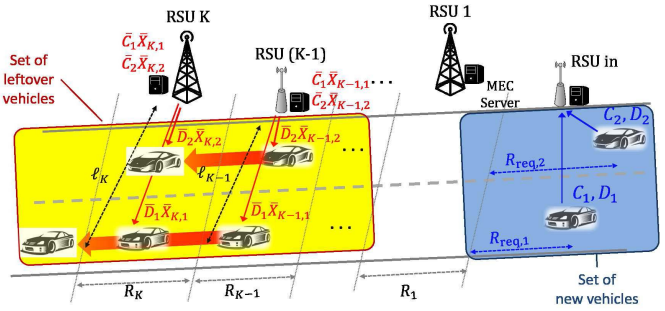


Fig. 2. Illustration of the online scenario.

A. Network Model and Cooperative Offloading Models in Online Scenario

In the online scenario, we consider the same edge computing-enabled vehicular network consisting of K multi-antenna RSUs as in the offline scenario. The notations for the network parameters are the same as those in the offline scenario. Unlike the offline scenario, at any time instant $t > 0$ when some vehicles, referred to as *new arrivals*, are about to enter the network, some vehicles, called *leftovers*, already exist in the network. Let U and $\bar{U}$ denote the numbers of new arrivals and leftovers, respectively. Denote $\mathcal{U} \triangleq \{1, \cdots, U\}$ and $\bar{\mathcal{U}} \triangleq \{\bar{1}, \cdots, \bar{U}\}$ as the sets of new arrivals and leftovers, respectively.[10] For ease of exposition, we assume that the minimum distance between a leftover and a new arrival at any time instant is large enough so that no leftovers and new arrivals will appear in an RSU's coverage area before they leave the network. In the online scenario, the mobility, task, computation, and communication models of each new arrival and leftover are the same as those of a vehicle in the offline scenario. Besides, in the online scenario, the notations for the vehicle parameters, task parameters, computation parameters, and communication parameters of each new arrival are identical to those of a vehicle in the offline scenario. For all $q \in \bar{\mathcal{U}}$, let $\bar{v}_q$, $\bar{C}_q$, and $\bar{D}_q$ denote the velocity, the size of the workload, and the size of the computation result for leftover q in the online scenario, respectively.

At time t, the task splitting among the K RSUs for the leftovers in $\bar{\mathcal{U}}$ has already been determined, and the computations and communications for some subtasks have been completed. We denote $\bar{x}_{k,q}$ as the fraction of task $q \in \bar{\mathcal{U}}$ that has not yet been executed at RSU $k \in \mathcal{K}$. Thus, the sizes of the remaining computation workload and the computation result of leftover q's subtask at RSU k are $\bar{C}_q\bar{x}_{k,q}$ and $\bar{D}_q\bar{x}_{k,q}$, respectively, where $\bar{x}_{k,q} \in [0, 1]$. The arrival and departure times of leftover q at RSU k are denoted by $\bar{T}_{k,q}^{(\mathrm{a})}$ and $\bar{T}_{k,q}^{(\mathrm{d})}$, respectively. Let $\bar{a}_{k,n}$ denote the index of the leftover that is the n-th one entering RSU k's coverage area for all $k \in \mathcal{K}$ and $n \in \bar{\mathcal{U}}$. As in the offline scenario, we assume that the controller knows the

[10]Note that in the offline scenario, the set $\bar{\mathcal{U}}$ is the empty set and therefore does not have to be considered.

task parameters of leftovers and new arrivals, i.e., $\bar{C}_{k,q}\bar{x}_{k,q}$, $\bar{D}_{k,q}\bar{x}_{k,q}$ and $C_{k,u}x_{k,u}$, $D_{k,u}x_{k,u}$, and the vehicle parameters for leftovers and new arrivals, i.e., $\bar{v}_q$, $\bar{T}^{(\mathrm{a})}_{k,q}$, $\bar{T}^{(\mathrm{d})}_{k,q}$ and v_u, $T^{(\mathrm{a})}_{k,u}$, $T^{(\mathrm{d})}_{k,u}$.

B. Online Optimization

Let $\mathbf{x} \triangleq (x_{k,u})_{k\in\mathcal{K},u\in\mathcal{U}}$, $\mathbf{f} \triangleq (f_{k,u})_{k\in\mathcal{K},u\in\mathcal{U}}$, $\mathbf{p} \triangleq (p_{k,u})_{k\in\mathcal{K},u\in\mathcal{U}}$, $\mathbf{t}_{\mathrm{cm}} \triangleq (t_{\mathrm{cm},k,u})_{k\in\mathcal{K},u\in\mathcal{U}}$, $\mathbf{s}_{\mathrm{cp}} \triangleq (s_{\mathrm{cp},k,u})_{k\in\mathcal{K},u\in\mathcal{U}}$, and $\mathbf{s}_{\mathrm{cm}} \triangleq (s_{\mathrm{cm},k,u})_{k\in\mathcal{K},u\in\mathcal{U}}$ denote the task splitting, the CPU frequency allocation, the transmission power allocation, the transmission time allocation, the computation starting time allocation, and the transmission starting time allocation for the new arrivals, respectively. Then, as in the offline scenario, $\mathbf{x}$, $\mathbf{f}$, $\mathbf{p}$, $\mathbf{t}_{\mathrm{cm}}$, $\mathbf{s}_{\mathrm{cp}}$, and $\mathbf{s}_{\mathrm{cm}}$ satisfy the constraints in (3), (4), (5), (6), (9), (10), (11), (12), (18), (20), (21), (22), and (19). However, unlike the offline scenario, $s_{\mathrm{cp},k,u}$ satisfies:

$$s_{\mathrm{cp},k,u} \geq t, \quad k \in \mathcal{K},\ u \in \mathcal{U}, \tag{48}$$

rather than (8), since the new arrivals enter the network at time t rather than time 0.

For the leftovers, the task splitting has been completed and cannot be changed, the communication resource allocation at the RSUs does not have to be updated under the assumption that no leftovers and new arrivals will appear in an RSU's coverage area, and the computation resource allocation at the RSUs should be updated given the new arrivals at time t. Let $\bar{p}_{k,q} \in [0, P_k]$ and $\bar{t}_{\mathrm{cm},k,q} \in [0, T^{(\mathrm{d})}_{k,q} - T^{(\mathrm{a})}_{k,q}]$ denote the transmission power and the transmission time allocated for transmitting leftover q's subtask k from RSU k to leftover q, respectively, which have been determined. Let $\bar{\mathbf{f}} = \left(\bar{f}_{k,q}\right)_{k\in\mathcal{K},q\in\bar{\mathcal{U}}}$ and $\bar{\mathbf{s}}_{\mathrm{cp}} = (\bar{s}_{\mathrm{cp},k,q})_{k\in\mathcal{K},q\in\bar{\mathcal{U}}}$ denote the updated CPU frequency allocation and computation starting time allocation for the leftovers, respectively. As in the offline scenario, we have the following computation constraints for $\bar{\mathbf{f}}$ and $\bar{\mathbf{s}}_{\mathrm{cp}}$:

$$\bar{f}_{k,q} \geq 0, \quad k \in \mathcal{K},\ q \in \bar{\mathcal{U}}, \tag{49}$$

$$\bar{f}_{k,q} \leq F_k, \quad k \in \mathcal{K},\ q \in \bar{\mathcal{U}}, \tag{50}$$

$$\bar{s}_{\mathrm{cp},k,\bar{a}_{k,n}} + \frac{\bar{C}_{\bar{a}_{k,n}}\bar{x}_{k,\bar{a}_{k,n}}}{\bar{f}_{k,\bar{a}_{k,n}}} \leq \bar{s}_{\mathrm{cp},k,\bar{a}_{k,n+1}}, \quad k \in \mathcal{K}, \quad n \in \bar{\mathcal{U}}\backslash\{\bar{U}\}, \tag{51}$$

$$\bar{s}_{\mathrm{cp},k,\bar{a}_{k,n}} + \frac{\bar{C}_{\bar{a}_{k,n}}\bar{x}_{k,\bar{a}_{k,n}}}{\bar{f}_{k,\bar{a}_{k,n}}} \leq \bar{T}^{(\mathrm{a})}_{k,\bar{a}_{k,n}}, \quad k \in \mathcal{K},\ n \in \bar{\mathcal{U}}. \tag{52}$$

Different from the offline scenario, $\bar{\mathbf{s}}_{\mathrm{cp}}$ satisfies:

$$\bar{s}_{\mathrm{cp},k,q} \geq t, \quad k \in \mathcal{K},\ q \in \bar{\mathcal{U}}, \tag{53}$$

rather than (8). Furthermore, assuming that no leftovers and new arrivals will appear in an RSU's coverage area, we have the following additional computation constraints for $\mathbf{s}_{\mathrm{cp}}$ and $\bar{\mathbf{s}}_{\mathrm{cp}}$:

$$\bar{s}_{\mathrm{cp},k,\bar{a}_{k,\bar{U}}} + \frac{\bar{C}_{\bar{a}_{k,n}}\bar{x}_{k,\bar{a}_{k,\bar{U}}}}{\bar{f}_{k,\bar{U}}} \leq s_{\mathrm{cp},k,a_{k,1}}, \quad k \in \mathcal{K}. \tag{54}$$

The total energy consumed at the K RSUs for serving the U new arrivals, denoted by $E_{\mathrm{tot}}(\mathbf{x}, \mathbf{f}, \mathbf{p}, \mathbf{t}_{\mathrm{cm}})$, is given by (24). The total energy consumed at the K RSUs for serving the $\bar{U}$ leftovers from time t, denoted by $\bar{E}_{\mathrm{tot}}(\bar{\mathbf{f}})$, is given by:

$$\bar{E}_{\mathrm{tot}}(\bar{\mathbf{f}}) = \sum_{k\in\mathcal{K}}\sum_{q\in\bar{\mathcal{U}}}\left(\kappa_k \bar{C}_q \bar{x}_{k,q} \bar{f}_{k,q}^{\varphi_k - 1} + \bar{p}_{k,q}\bar{t}_{\mathrm{cm},k,q}\right). \tag{55}$$

Therefore, in the online scenario, the total energy consumed at the RSUs for serving both leftovers and new arrivals from time t is given by:

$$E_{\mathrm{online}}(\mathbf{x}, \mathbf{f}, \mathbf{p}, \mathbf{t}_{\mathrm{cm}}, \bar{\mathbf{f}}) = E_{\mathrm{tot}}(\mathbf{x}, \mathbf{f}, \mathbf{p}, \mathbf{t}_{\mathrm{cm}}) + \bar{E}_{\mathrm{tot}}(\bar{\mathbf{f}}), \tag{56}$$

where $E_{\mathrm{tot}}(\mathbf{x}, \mathbf{f}, \mathbf{p}, \mathbf{t}_{\mathrm{cm}})$ and $\bar{E}_{\mathrm{tot}}(\bar{\mathbf{f}})$ are given by (24) and (55), respectively.

In the online scenario, we would like to minimize $E_{\mathrm{online}}(\mathbf{x}, \mathbf{f}, \mathbf{p}, \mathbf{t}_{\mathrm{cm}}, \bar{\mathbf{f}})$ in (56) by optimizing $\mathbf{x}$, $\mathbf{f}$, $\mathbf{p}$, $\mathbf{t}_{\mathrm{cm}}$, $\mathbf{s}_{\mathrm{cp}}$, $\mathbf{s}_{\mathrm{cm}}$, $\bar{\mathbf{f}}$, and $\bar{\mathbf{s}}_{\mathrm{cp}}$ under the constraints in (3)–(6), (9)–(12), (18)–(22), (19), and (48)–(54). Therefore, the optimization problem is formulated as follows:

Problem 6 (Energy Minimization in Online Scenario):

$$E^*_{\mathrm{online}} \triangleq \min_{\substack{\mathbf{x},\mathbf{f},\mathbf{p},\mathbf{t}_{\mathrm{cm}},\mathbf{s}_{\mathrm{cp}},\mathbf{s}_{\mathrm{cm}} \\ \bar{\mathbf{f}},\bar{\mathbf{s}}_{\mathrm{cp}}}} E_{\mathrm{online}}(\mathbf{x}, \mathbf{f}, \mathbf{p}, \mathbf{t}_{\mathrm{cm}}, \bar{\mathbf{f}})$$
$$\text{s.t. } (3)(6), (9)(12), (18)(22), (19), (48)(54).$$

Given the network parameters and the parameters of both new arrivals and leftovers, the controller can handle this optimization. The structure of Problem 6 for the online scenario is similar to the structure of Problem 1 for the offline scenario, as problems have identical constraints, i.e., (3)–(6), (9)–(12), (18)–(22), (19), and the structures of (48)–(54) are very similar to those of (5), (6), (8)–(10). Similar to the multi-vehicle case in the offline scenario, we can obtain an optimal transmission power allocation for new arrivals as in Lemma 1. Then, we equivalently convert the non-convex problem in Problem 6 to a convex problem, which can be solved optimally using an interior-point method of computational complexity $\mathcal{O}\left(\left(5KU + 2K\bar{U}\right)^3\right)$. Therefore, Problem 6 can be solved using the proposed methods for the offline scenario.[11] We omit the details due to the page limitation. Note that the higher arrival rate of the vehicles increases the number of the new arrivals in a batch, U, which increases the computational complexity. Therefore, the proposed online algorithm can be applied to the network that gives affordable computational complexity of the algorithm.

V. Numerical Results

In this section, we numerically show the total energy consumptions of our proposed solutions for the single-vehicle and the multi-vehicle cases in the offline scenario. Unless otherwise specified, we set $K = 20$, $N_o = -80$dBm, $B = 5$MHz, $\alpha = 4$, $\kappa_k = 10^{-11}$, $\varphi_k = 3$, $k \in \mathcal{K}$, and $C_u = 1000D_u$, and $\theta_u = 0.95$, $u \in \mathcal{U}$ [35]. In the single-vehicle case, we set $R_{\mathrm{req},1} = 300$m, $D_1 = 300$MB, and $v_1 = 75$km/h. In the multi-vehicle case, we set $U = 3$, $R_{\mathrm{req},1} = 300$m, $R_{\mathrm{req},2} = 400$m, $R_{\mathrm{req},3} = 500$m, $D_1 = (D - \delta_D)$MB, $D_2 =$

[11]Note that conventional optimization techniques (e.g., Markov decision process and Lyapunov optimization) are not suitable to the system model in this paper which is complex and highly dynamic.

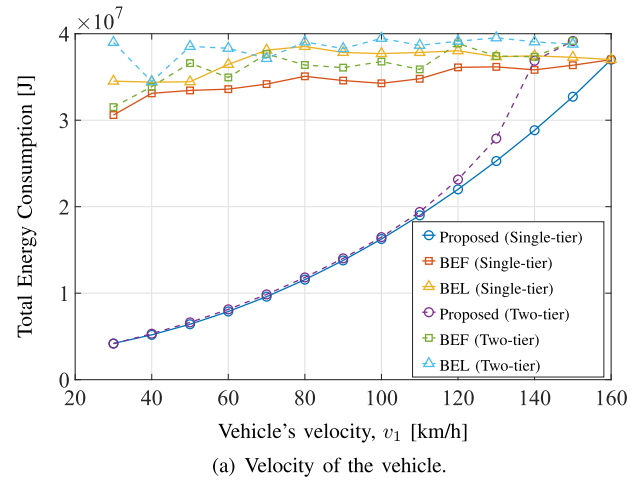
D MB, $D_3 = (D + \delta_D)$MB, $v_1 = (v - \delta_v)$km/h, and $v_2 = v$ km/h, $v_3 = (v + \delta_v)$km/h, where $D = 50$MB, $\delta_D = 0$MB, $v = 80$km/h, and $\delta_v = 5$km/h, unless otherwise specified. It is worth noting that D and v represent the average computation result size and velocity of the three vehicles, respectively, and δ_D and δ_v represent the differences in the computation result sizes and velocities of the three vehicles, respectively.[12] We consider two network setups in each case, i.e., the single-tier network and the two-tier network. In the single-tier network, we set $R_k = 500$m, $P_k = 50$dBm, $F_k = 1.1$GHz, $k \in \mathcal{K}$. In the two-tier network, we set $R_k = 600$m, $P_k = 55$dBm, $F_k = 1.2$GHz, $k \in \{1, 3, \cdots, 19\}$ and $R_k = 400$m, $P_k = 45$dBm, $F_k = 1.0$GHz, $k \in \{2, 4, \cdots, 20\}$ [43]. Note that both network setups have identical $\sum_{k\in\mathcal{K}} P_k$ and $\sum_{k\in\mathcal{K}} F_k$ for a fair comparison.

A. Single-Vehicle Case

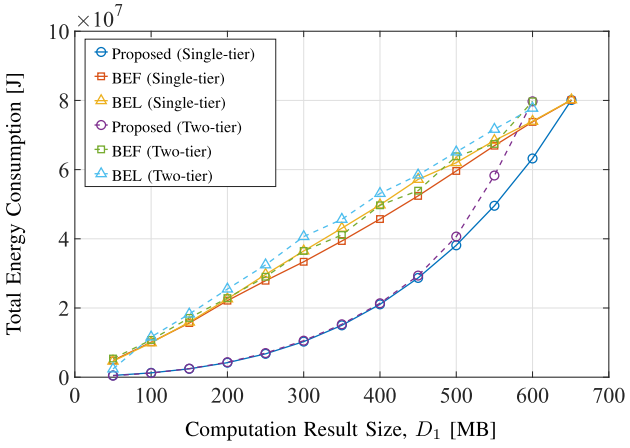
To assess the total energy consumption of our proposed solution in the single-vehicle case, we consider two baseline schemes, namely, the best effort first (BEF) scheme and the best effort last (BEL) scheme. From the constraints in (38) and (39) of Problem 4, we have $x_{k,1} \le x_{k,1}^{(\mathrm{m})}$, $k \in \mathcal{K}$, where $x_{k,1}^{(\mathrm{m})} \triangleq \min\left\{\frac{F_k T_{k,1}^{(\mathrm{a})}}{C_1}, \frac{B\left(T_{k,1}^{(\mathrm{d})} - T_{k,1}^{(\mathrm{a})}\right)}{D_1}\log_2\left(1 + \frac{P_k g_k G^{-1}(\theta)}{N_o}\right)\right\}$ represents the maximum fraction of the task that can be handled by RSU k under the simulation setup. The BEF scheme splits the task according to $x_{k,1} = x_{k,1}^{(\mathrm{m})}$, $k \in \{1, \cdots, \bar{k} - 1\}$, $x_{\bar{k},1} = 1 - \sum_{i=1}^{\bar{k}} x_{i,1}^{(\mathrm{m})}$, and $x_{k,1} = 0$, $k \in \{\bar{k} + 1, \cdots, K\}$, where $\bar{k} \triangleq \min\left\{k \in \mathcal{K} \,|\, \sum_{i=1}^{K} x_{i,1}^{(\mathrm{m})} > 1\right\}$. On the other hand, the BEL scheme splits the task according to $x_{k,1} = x_{k,1}^{(\mathrm{m})}$, $k \in \{\underline{k}+1, \cdots, K\}$, $x_{\underline{k}} = 1 - \sum_{i=\underline{k}+1}^{K} x_{i,1}^{(\mathrm{m})}$, and $x_{k,1} = 0$, $k \in \{1, \cdots, \underline{k} - 1\}$, where $\underline{k} \triangleq \max\left\{k \in \mathcal{K} \,|\, \sum_{i=1}^{K} x_{i,1}^{(\mathrm{m})} > 1\right\}$.

Fig. 3(a) and 3(b) illustrate the total energy consumption versus the vehicle's velocity v_1 and the size of the computation result D_1, respectively. Note that for the single-tier network, the problem is not feasible when $D_1 = 300$MB, $v_1 > 150$km/h or $D_1 > 650$MB, $v_1 = 75$km/h. Moreover, for the two-tier network, the problem is not feasible when $D_1 = 300$MB, $v_1 > 160$km/h or $D_1 > 600$MB, $v_1 = 75$km/h. From Fig. 3(a), we can see that the total energy consumption of the proposed solution decreases as v_1 decreases, whereas the total energy consumption of each baseline scheme does not show a noticeable trend with v_1. This is because the time for serving the vehicle's task increases as v_1 decreases, but the baseline schemes do not effectively adapt to v_1 as the proposed solution. From Fig. 3(b), we can see that the total energy consumption of each scheme increases with D_1. This is because as D_1 increases, each RSU allocates more CPU frequency and transmission power to execute and transmit the subtasks, respectively. Additionally, from Fig. 3, we can see that our proposed solution outperforms both baseline schemes in each network setup. This is because the proposed solution optimally splits the vehicle's task and allocates resources at the RSUs according to the network parameters in the single-vehicle case. Moreover, we can see that the total energy consumption of each scheme in the two-tier network is larger than that in the single-tier network, indicating that heterogeneity influences resource utilization efficiency.

[12]Note that the practical vehicle simulation such as the simulation of urban mobility (SUMO) [42] can also be used for the performance evaluation by taking the location information of vehicles of a batch.



(a) Velocity of the vehicle.

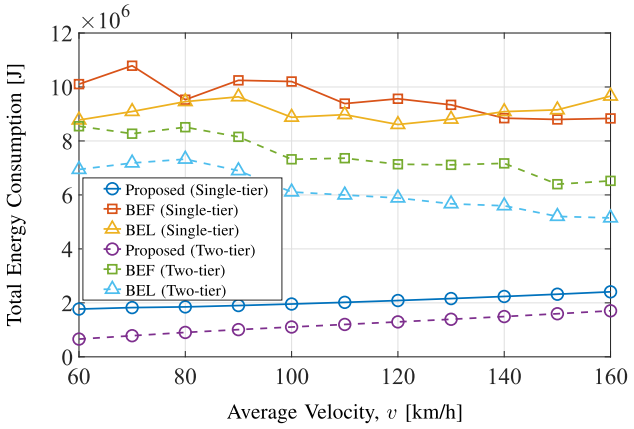


(b) Size of the computation result.

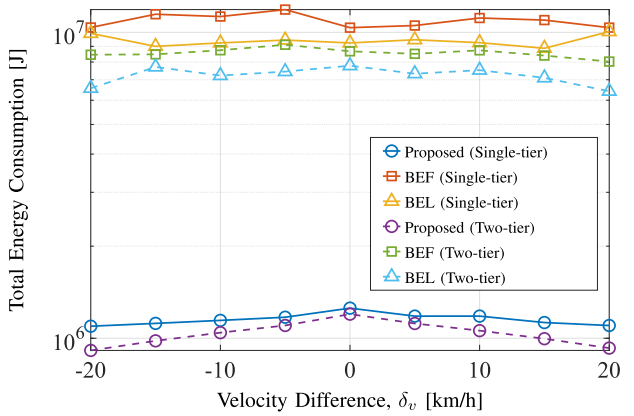
Fig. 3. Total energy consumption versus the velocity of the vehicle v_1 and the computation result size of the vehicle D_1 in the single-tier and two-tier network.

B. Multi-Vehicle Case

In the multi-vehicle case, we consider two baseline schemes that are generalized from those in the single-vehicle case and are also termed the BEF scheme and the BEL scheme for ease of exposition. Each RSU serves the vehicles in their arriving order in both baseline schemes as in the proposed solution. Specifically, from the constraints in (38) and (39) of Problem 4, we have $x_{k,a_{k,1}} \le x_{k,a_{k,1}}^{(\mathrm{m})}$, $k \in \mathcal{K}$ for vehicle $a_{k,1}$, where $x_{k,a_{k,1}}^{(\mathrm{m})} \triangleq \min\left\{\frac{F_k T_{k,a_{k,1}}^{(\mathrm{a})}}{C_{a_{k,1}}}, \frac{B\left(T_{a_{k,1}}^{(\mathrm{d})} - T_{a_{k,1}}^{(\mathrm{a})}\right)}{D_{a_{k,1}}}\log_2\left(1 + \frac{P_k g_k G^{-1}(\theta)}{N_o}\right)\right\}$ represents the maximum fraction of vehicle $a_{k,1}$'s task that can be handled by RSU k under the simulation setup. On the other hand, from the constraints in (9), (10), (21), and (22) of Problem 1, we have $x_k \le x_{k,a_{k,2}}^{(\mathrm{m})}$, $k \in \mathcal{K}$

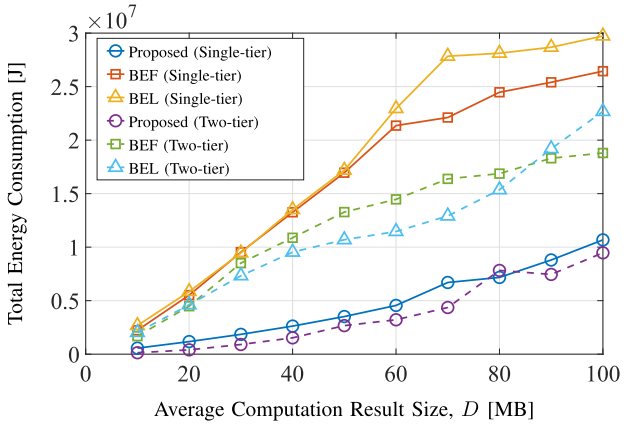


(a) Average velocity of the three vehicles.

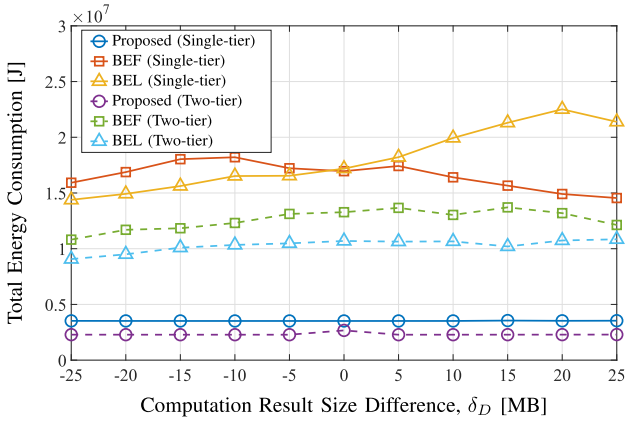


(b) Velocity difference between the three vehicles.

Fig. 4. Total energy consumption versus the average velocity v and the velocity difference δ_v in the single-tier and two-tier network.



(a) Average computation result size of the three vehicles.



(b) Computation result size difference between the three vehicles.

Fig. 5. Total energy consumption versus the average computation result size D and the computation result size difference δ_D in the single-tier and two-tier network.

for vehicle $a_{k,2}$, where $x^{(\mathrm{m})}_{k,a_{k,2}} \triangleq \min\left\{\frac{F_k\left(T^{(\mathrm{a})}_{k,a_{k,2}}-T^{(\mathrm{a})}_{k,a_{k,1}}\right)}{C_{a_{k,2}}}, \frac{\max\left\{0,B\left(T^{(\mathrm{d})}_{k,a_{k,2}}-\max\left(T^{(\mathrm{a})}_{k,a_{k,2}},T^{(\mathrm{d})}_{k,a_{k,1}}\right)\right)\right\}\log_2\left(1+\frac{P_k g_k G^{-1}(\theta)}{N_o}\right)}{D_1}\right\}$.

Based on $x^{(\mathrm{m})}_{k,a_{k,1}}$ and $x^{(\mathrm{m})}_{k,a_{k,2}}$ defined above, the BEF scheme and the BEL scheme split the tasks as in the multi-vehicle case.

Figs. 4(a) and 4(b) show the total energy consumption versus the average velocity of the three vehicles, v, and the velocity difference of the three vehicles, δ_v, respectively. Analogous to Fig. 4(a), we can see that the total energy consumption of the proposed solution decreases as v decreases, whereas the total energy consumption of each baseline scheme does not show an evident trend with v. Moreover, from Fig. 4(b), we can see that the total energy consumption of the proposed solution decreases with $|\delta_v|$, whereas the total energy consumption of each baseline scheme does not show an apparent trend with $|\delta_v|$. This is because the time for serving the first arriving vehicle at each RSU increases with $|\delta_v|$, but the baseline schemes do not effectively adapt to $|\delta_v|$ as the proposed solution. Figs. 5(a) and 5(b) show the total energy consumption versus the average computation result size of the three vehicles, D, and the computation result size difference between the three vehicles, δ_D, respectively. Analogously, from Fig. 5(a), we can see that the total energy consumption of each scheme increases with D, and from Fig. 5(b), we can see that the total energy consumption of our proposed solution is less dependent on δ_D compared to the baseline schemes. Moreover, Figs. 4 Fig. 5 show that our proposed solution, which represents the optimal offloading design, outperforms

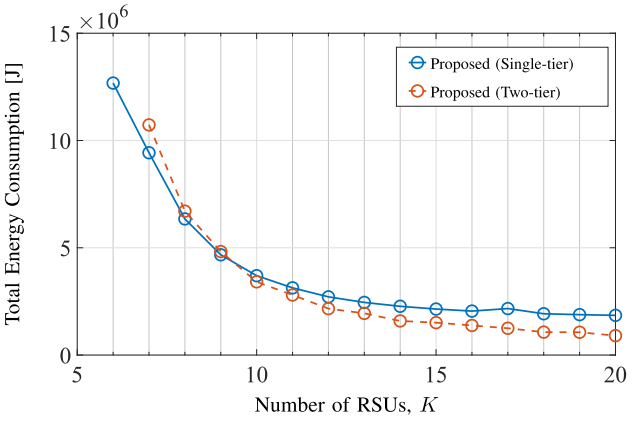


Fig. 6. Total energy consumption versus the number of RSUs K in the single-tier and two-tier network.

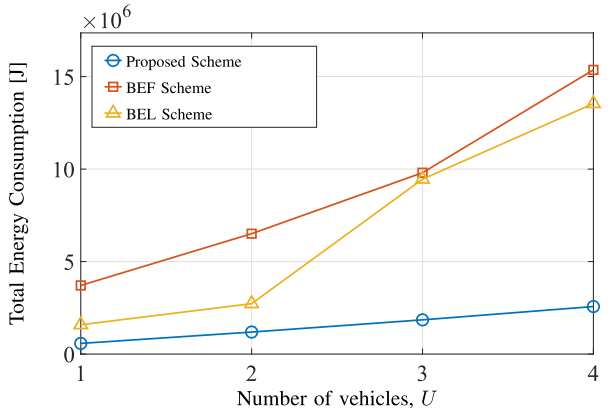


Fig. 7. Total energy consumption versus the number of vehicles U in the low traffic density case.

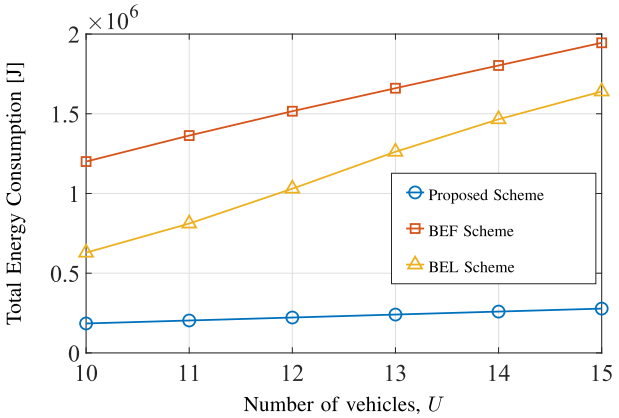


Fig. 8. Total energy consumption versus the number of vehicles U in the high traffic density case.

the baseline schemes in the single-tier and two-tier networks. Fig. 6 shows the total energy consumption versus the number of RSUs, K. We can see that the total energy consumption decreases with K since the required delay of the task increases with K.

Now, we present the total energy consumption for different number of vehicles, U in the two-tier network. Figs. 7 and 8 show the total energy consumption versus the number of vehicles, U, for low traffic density case and high traffic density case, respectively. For the simulation setup, we use $v_u \in$ [70, 90]km/h, and $R_{\text{req},u} \in$ [100, 1000]m, $u \in \mathcal{U}$. To satisfy the computation and communication constraints of U vehicles in a batch with the limited capabilities of K RSUs, we use $D_u = 30$MB for Fig. 7 and $D_u = 1$MB for Fig. 8.[13] From Figs. 7 and 8, we can see that the total energy consumption increases with U, and the proposed scheme consumes less energy compared to the baseline schemes for all network configurations. Therefore, we can see that the superiority of the proposed solution still remains for both low and high traffic densities.

[13]Note that according to the statistics in [44], the average traffic densities are 10^4 vehicles/hour and 2.5×10^3 vehicles/hour for Provincial Road No. 301 (63km) and Incheon International Airport Expressway (36.5km), respectively. Hence, considering the batch size of our proposed scheme, which is defined as the number of vehicles in 1km, the traffic densities, considered in Figs. 7 and 8, are in a realistic range.

VI. Conclusion

This paper develops the energy-efficient cooperative offloading scheme for edge computing-enabled vehicular networks. Specifically, we first establish the cooperative offloading model for given batch of vehicles. Then, we formulate the total energy consumption minimization problem, which is a challenging non-convex problem with a considerably large number of variables and constraints. We show that the original optimization problem can be equivalently transformed into a convex problem with fewer variables and constraints, and obtain the optimal solutions in both multi-vehicle and single-vehicle cases. We further extend the cooperative offloading model to the one for the online scenario. We show that the underlying problem structure is similar and hence can be solved using the same method. Finally, numerical results show that the proposed solutions achieve a significant gain in the total energy consumption over all baseline schemes. We show that the total energy consumption of the proposed solutions decreases with the velocity difference of vehicles and increases with the vehicle's velocity and the size of the computation results. In a future work, this work can be extended by considering the mutual interference between RSUs and allowing the transfer of the computation result between RSUs. Moreover, we will further apply coded distributed computing (CDC) scheme, which leverages redundant computations to reduce the communication load of distributed computing.

Appendix

A. Proof of Theorem 1

Since (3), (18), and (19) imply (11), we can ignore (11). Lemma 1 implies that (19) holds with equality at the optimal solution. Thus, we can replace $p_{k,u}$ in Problem 1 with $\frac{N_o}{g_k G^{-1}(\theta_u)}\left(2^{\frac{D_u x_{k,u}}{B t_{\text{cm},k,u}}} - 1\right)$, for all $k \in \mathcal{K}$, $u \in \mathcal{U}$, and ignore the constraint in (19). Furthermore, by substituting $p_{k,u} = \frac{N_o}{g_k G^{-1}(\theta_u)}\left(2^{\frac{D_u x_{k,u}}{B t_{\text{cm},k,u}}} - 1\right)$ into (12), we have (31). Since (3) and (31) imply (18), we can ignore (18). Consequently, Problem 1 can be equivalently transformed to the following problem.

Problem 7 (Equivalent Problem of Problem 1 in Multi-Vehicle Case):

$$\min_{\substack{\mathbf{x},\mathbf{f},\mathbf{t}_{\text{cm}},\\ \mathbf{s}_{\text{cp}},\mathbf{s}_{\text{cm}}}} \sum_{k=1}^{K}\sum_{u=1}^{U}\Big(E_{\text{cp},k,u}\left(x_{k,u}, f_{k,u}\right) + \widetilde{E}_{\text{cm},k,u}\left(x_{k,u}, t_{\text{cm},k,u}\right)\Big)$$

$$\text{s.t. } (3), (4), (5), (6), (8), (9), (10), (20), (21), (22), (31).$$

where $\widetilde{E}_{\text{cm},k,u}\left(x_{k,u}, t_{\text{cm},k,u}\right)$ is obtained from (30).

By substituting $f_{k,u} = \frac{C_u x_{k,u}}{t_{\text{cp},k,u}}$ into the objective function of Problem 7, we have (28). By replacing $f_{k,u}$ in (6), (9), and (10) with $\frac{C_u x_{k,u}}{t_{\text{cp},k,u}}$, we have (32), (33), and (34), respectively. Since (3) and (32) imply (5), we can ignore (5). Therefore, Problem 7 is equivalent to Problem 2, and thus Problem 2 is equivalent to Problem 1. Moreover, by substituting $\mathbf{x}^* = \mathbf{x}^\dagger$,

$\mathbf{f}^* = \left(\frac{C_u x^*_{k,u}}{t^\dagger_{\text{cp},k,u}}\right)_{k\in\mathcal{K},u\in\mathcal{U}}$, $\mathbf{t}^*_{\text{cm}} = \mathbf{t}^\dagger_{\text{cm}}$, $\mathbf{s}^*_{\text{cp}} = \mathbf{s}^\dagger_{\text{cp}}$, $\mathbf{s}^*_{\text{cm}} = \mathbf{s}^\dagger_{\text{cm}}$, and $p^*_{k,u}$ in (25) into E^*_{tot}, we have $E^*_{\text{tot}} = E^\dagger_{\text{tot}}$.

B. Proof of Theorem 2

First, we prove that $s^*_{\text{cp},k} = 0$ and $s^*_{\text{cm},k} = T^{(\text{a})}_k$. For any $\mathbf{s}_{\text{cp}}$ and $\mathbf{s}_{\text{cm}}$ satisfying (8) and (20), respectively, the feasible set of the other variables $(\mathbf{x}, \mathbf{f}, \mathbf{p}, \mathbf{t}_{\text{cm}})$ is given by:

$$\mathcal{Z}(\mathbf{s}_{\text{cp}}, \mathbf{s}_{\text{cm}}) \triangleq \{(\mathbf{x}, \mathbf{f}, \mathbf{p}, \mathbf{t}_{\text{cm}}) \,|\, (3), (4), (5), (6), (10), (11), (12), (18), (21), (19)\}. \quad (57)$$

Note that the constraints in (9) and (22) in Problem 1 are void in the single-vehicle case. Then, we can express Problem 1 in the following equivalent form.

Problem 8 (Equivalent Problem of Problem 1):

$$\min_{\mathbf{x},\mathbf{f},\mathbf{p},\mathbf{t}_{\text{cm}},\mathbf{s}_{\text{cp}},\mathbf{s}_{\text{cm}}} E_{\text{tot}}(\mathbf{x}, \mathbf{f}, \mathbf{p}, \mathbf{t}_{\text{cm}})$$
$$\text{s.t. } (8), (20),$$
$$(\mathbf{x}, \mathbf{f}, \mathbf{p}, \mathbf{t}_{\text{cm}}) \in \mathcal{Z}(\mathbf{s}_{\text{cp}}, \mathbf{s}_{\text{cm}}).$$

Note that $\mathbf{s}_{\text{cp}}$ and $\mathbf{s}_{\text{cm}}$ affect $\mathcal{Z}(\mathbf{s}_{\text{cp}}, \mathbf{s}_{\text{cm}})$ only through the constraints in (10) and (21). From (10) and (21), $\mathcal{Z}(\mathbf{s}_{\text{cp}}, \mathbf{s}_{\text{cm}})$ achieves the largest when $s_{\text{cp},k} = 0$ and $s_{\text{cm},k} = T^{(\text{a})}_k$. Therefore, we can prove that $s^*_{\text{cp},k} = 0$ and $s^*_{\text{cm},k} = T^{(\text{a})}_k$ are optimal.

Next, we show that Problem 1 is equivalent to Problem 3. For given $s^*_{\text{cp},k} = 0$ and $s^*_{\text{cm},k} = T^{(\text{a})}_k$, it remains to minimize $E_{\text{tot}}(\mathbf{x}, \mathbf{f}, \mathbf{p}, \mathbf{t}_{\text{cm}})$ with respect to $(\mathbf{x}, \mathbf{f}, \mathbf{p}, \mathbf{t}_{\text{cm}})$. Moreover, (10) and (21) become (35) and (36), respectively. Therefore, we can prove that the optimal solutions of Problem 1 and Problem 3 satisfy (37).

C. Proof of Theorem 3

First, we characterize optimality properties of Problem 3. For given x_k and p_k, the objective function increases with $t_{\text{cm},k}$ and f_k, for all $k \in \mathcal{K}$. Besides, the constraints in (3), (4), (11), and (12) are not related to $t_{\text{cm},k}$, f_k, $k \in \mathcal{K}$. Thus, by contradiction, we can show that the inequality constraints in (19) and (35) are active at $(\mathbf{x}^\star, \mathbf{f}^\star, \mathbf{p}^\star, \mathbf{t}^\star_{\text{cm}})$. Therefore, a solution of Problem 3 satisfies:

$$t^\star_{\text{cm},k} = \frac{Dx^\star_k}{B\log_2\left(1 + \frac{p^\star_k g_k}{N_o} G^{-1}(\theta)\right)}, \quad k \in \mathcal{K}, \quad (58)$$

$$f^\star_k = \frac{Cx^\star_k}{T^{(\text{a})}_k}, \quad k \in \mathcal{K}. \quad (59)$$

Thus, without loss of optimality, we can replace $t_{\text{cm},k}$ and f_k in Problem 3 with $\frac{Dx_k}{B\log_2\left(1+\frac{p_k g_k}{N_o}G^{-1}(\theta)\right)}$ and $\frac{Cx_k}{T^{(\text{a})}_k}$ respectively, for all $k \in \mathcal{K}$, $u \in \mathcal{U}$, and ignore the constraints in (19) and (35). Moreover, by substituting $f_k = \frac{Cx_k}{T^{(\text{a})}_k}$ and $t_{\text{cm},k} = \frac{Dx_k}{B\log_2\left(1+\frac{p_k g_k}{N_o}G^{-1}(\theta)\right)}$ into (6) and (36), we have (38) and (41). For Problem 3 to be feasible, the lower bound of p_k in (41) must be smaller than the upper bound in (12), and thus, we have (39). Since (3) and (41) imply (11), we ignore (11). Moreover, (3) implies (5) and (18), and therefore, we can ignore (5) and (18). Therefore, Problem 3 can be equivalently transformed to the following problem.

Problem 9 (Equivalent Problem of Problem 3):

$$\min_{\mathbf{x},\mathbf{p}} \sum_{k=1}^{K} \left(\kappa_k C^{\varphi_k}\left(T^{(\text{a})}_k\right)^{1-\varphi_k} x_k^{\varphi_k} + \frac{Dx_k p_k}{B\log_2\left(1+\frac{p_k g_k}{N_o}G^{-1}(\theta)\right)}\right)$$
$$\text{s.t. } (3), (4), (12), (38), (39), (41).$$

The constraints in (3), (4), (38), and (39) are only related to x_k, while the constraint in (12) is only related to p_k. Thus, we can equivalently convert Problem 9 to a master problem in Problem 4, which optimizes $\mathbf{x}$, and K subproblems in Problem 5, which optimize p_k, $k \in \mathcal{K}$, respectively. Furthermore, by substituting $x^\star_k = x^\ddagger_k$, $p^\star_k = p^\ddagger_k\left(x^\ddagger_k\right)$, $f^\star_k = \frac{Cx^\ddagger_k}{T^{(\text{a})}_k}$, and $t^\star_{\text{cm},k} = \frac{Dx^\ddagger_k}{B\log_2\left(1+\frac{g_k p^\ddagger_k\left(x^\ddagger_k\right)}{N_o}G^{-1}(\theta)\right)}$ into $E^\star_{\text{single}}$, we have $E^\star_{\text{single}} = E^\ddagger_{\text{single}}$. Therefore, we complete the proof of Theorem 3.

D. Proof of Lemma 4

First, by relaxing the coupling constraint in (4), we obtain the partial Lagrange function:

$$L(\mathbf{x}, \gamma) = \gamma\left(1 - \sum_{k=1}^{K} x_k\right) + \sum_{k=1}^{K}\left\{\kappa_k C^{\varphi_k}\left(T^{(\text{a})}_k\right)^{1-\varphi_k} x_k^{\varphi_k} \quad (60)$$

$$+ \frac{N_o\left(T^{(\text{d})}_k - T^{(\text{a})}_k\right)}{g_k G^{-1}(\theta)}\left(2^{\frac{Dx_k}{B\left(T^{(\text{d})}_k - T^{(\text{a})}_k\right)}} - 1\right)\right\}, \quad (61)$$

Now we obtain the KKT conditions:

$$(3),\ (38),\ (39), \quad 1 - \sum_{k=1}^{K} x_k = 0,$$

$$\frac{\partial L(\mathbf{x}, \gamma)}{\partial x_k} = \varphi_k \kappa_k C^{\varphi_k}\left(T^{(\text{a})}_k\right)^{1-\varphi_k} x_k^{\varphi_k - 1} + \frac{DN_o \ln(2)}{g_k B G^{-1}(\theta)} 2^{\frac{Dx_k}{B\left(T^{(\text{d})}_k - T^{(\text{a})}_k\right)}} - \gamma = 0, \quad k \in \mathcal{K}. \quad (62)$$

From (62), we obtain $x_k = H^{-1}_k(\gamma)$, where $H_k(\cdot)$ is given by (45). Thus, we have

$$x^\ddagger_k = \max\left[\min\left\{\frac{B\left(T^{(\text{d})}_k - T^{(\text{a})}_k\right)}{D}\log_2\left(1 + \frac{P_k g_k G^{-1}(\theta)}{N_o}\right), \frac{F_k T^{(\text{a})}_k}{C}, \ H^{-1}_k(\gamma^*)\right\}, \ 0\right], \quad k \in \mathcal{K}. \quad (63)$$

It is obvious that $\frac{B\left(T^{(\text{d})}_k - T^{(\text{a})}_k\right)}{D}\log_2\left(1 + \frac{P_k g_k G^{-1}(\theta)}{N_o}\right)$ and $\frac{F_k T^{(\text{a})}_k}{C}$ in (63) are non-negative values. Thus, we can represent (63) by (44). Therefore, we complete the proof of Lemma 4.

REFERENCES

[1] H. Cho, Y. Cui, and J. Lee, "Energy-efficient computation task splitting for edge computing-enabled vehicular networks," in *Proc. IEEE Int. Conf. Commun. Workshops (ICC Workshops)*, Dublin, Ireland, Jun. 2020, pp. 1–6.

[2] R. Li, Q. Ma, J. Gong, Z. Zhou, and X. Chen, "Age of processing: Age-driven status sampling and processing offloading for edge-computing-enabled real-time IoT applications," *IEEE Internet Things J.*, vol. 8, no. 19, pp. 14471–14484, Oct. 2021.

[3] J. Liu, J. Wan, B. Zeng, Q. Wang, H. Song, and M. Qiu, "A scalable and quick-response software defined vehicular network assisted by mobile edge computing," *IEEE Commun. Mag.*, vol. 55, no. 7, pp. 94–100, Jul. 2017.

[4] Y. He, M. Chen, B. Ge, and M. Guizani, "On WiFi offloading in heterogeneous networks: Various incentives and trade-off strategies," *IEEE Commun. Surveys Tuts.*, vol. 18, no. 4, pp. 2345–2385, 4th Quart., 2016.

[5] L. Nkenyereye, L. Nkenyereye, S. M. R. Islam, C. A. Kerrache, M. Abdullah-Al-Wadud, and A. Alamri, "Software defined network-based multi-access edge framework for vehicular networks," *IEEE Access*, vol. 8, pp. 4220–4234, 2020.

[6] H. T. Dinh, C. Lee, D. Niyato, and P. Wang, "A survey of mobile cloud computing: Architecture, applications, and approaches," *Wireless Commun. Mobile Comput.*, vol. 13, no. 18, pp. 1587–1611, Dec. 2011.

[7] M. Patel *et al.*, "Mobile-edge computing: Introductory technical white paper," Dept. Mobile-Edge Comput. Ind. Initiative, ETSI, Sophia Antipolis, France, White Paper V1 18-09-14, Sep. 2014.

[8] Y. Mao, C. You, J. Zhang, K. Huang, and K. B. Letaief, "A survey on mobile edge computing: The communication perspective," *IEEE Commun. Surveys Tuts.*, vol. 19, no. 4, pp. 2322–2358, 4th Quart., 2017.

[9] N. Hassan, K.-L. A. Yau, and C. Wu, "Edge computing in 5G: A review," *IEEE Access*, vol. 7, pp. 127276–127289, 2019.

[10] S. Liu, L. Liu, J. Tang, B. Yu, Y. Wang, and W. Shi, "Edge computing for autonomous driving: Opportunities and challenges," *Proc. IEEE*, vol. 107, no. 8, pp. 1697–1716, Aug. 2019.

[11] C. Park and J. Lee, "Mobile edge computing-enabled heterogeneous networks," *IEEE Trans. Wireless Commun.*, vol. 20, no. 2, pp. 1038–1051, Feb. 2021.

[12] Z. Liang, Y. Liu, T.-M. Lok, and K. Huang, "Multiuser computation offloading and downloading for edge computing with virtualization," *IEEE Trans. Wireless Commun.*, vol. 18, no. 9, pp. 4298–4311, Sep. 2019.

[13] P. F. Wang, C. Yao, Z. Zheng, G. Sun, and L. Song, "Joint task assignment, transmission, and computing resource allocation in multilayer mobile edge computing systems," *IEEE Internet Things J.*, vol. 6, no. 2, pp. 2872–2884, Apr. 2018.

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

[15] F. Wang, J. Xu, X. Wang, and S. Cui, "Joint offloading and computing optimization in wireless powered mobile-edge computing systems," *IEEE Trans. Wireless Commun.*, vol. 17, no. 3, pp. 1784–1797, Mar. 2018.

[16] J. Bi, H. Yuan, S. Duanmu, M. Zhou, and A. Abusorrah, "Energy-optimized partial computation offloading in mobile-edge computing with genetic simulated-annealing-based particle swarm optimization," *IEEE Internet Things J.*, vol. 8, no. 5, pp. 3774–3785, Mar. 2021.

[17] H. Li, H. Xu, C. Zhou, X. Lu, and Z. Han, "Joint optimization strategy of computation offloading and resource allocation in multi-access edge computing environment," *IEEE Trans. Veh. Technol.*, vol. 69, no. 9, pp. 10214–10226, Sep. 2020.

[18] K. Cheng, Y. Teng, W. Sun, A. Liu, and X. Wang, "Energy-efficient joint offloading and wireless resource allocation strategy in multi-MEC server systems," in *Proc. IEEE Int. Conf. Commun. (ICC)*, May 2018, pp. 1–6.

[19] T. Q. Dinh, J. Tang, Q. D. La, and T. Q. S. Quek, "Offloading in mobile edge computing: Task allocation and computational frequency scaling," *IEEE Trans. Commun.*, vol. 65, no. 8, pp. 3571–3584, Aug. 2017.

[20] Y. Wang, M. Sheng, X. Wang, L. Wang, and J. Li, "Mobile-edge computing: Partial computation offloading using dynamic voltage scaling," *IEEE Trans. Commun.*, vol. 64, no. 10, pp. 4268–4282, Oct. 2016.

[21] Q. Gu, G. Wang, J. Liu, R. Fan, D. Fan, and Z. Zhong, "Optimal offloading with non-orthogonal multiple access in mobile edge computing," in *Proc. IEEE Global Commun. Conf. (GLOBECOM)*, Abu Dhabi, United Arab Emirates, Dec. 2018, pp. 1–7.

[22] J. Zhao, Q. Li, Y. Gong, and K. Zhang, "Computation offloading and resource allocation for cloud assisted mobile edge computing in vehicular networks," *IEEE Trans. Veh. Technol.*, vol. 68, no. 8, pp. 7944–7956, Aug. 2019.

[23] J. Wang, D. Feng, S. Zhang, J. Tang, and T. Q. S. Quek, "Computation offloading for mobile edge computing enabled vehicular networks," *IEEE Access*, vol. 7, pp. 62624–62632, 2019.

[24] 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.

[25] J. Sun, Q. Gu, T. Zheng, P. Dong, A. Valera, and Y. Qin, "Joint optimization of computation offloading and task scheduling in vehicular edge computing networks," *IEEE Access*, vol. 8, pp. 10466–10477, 2020.

[26] U. Saleem, Y. Liu, S. Jangsher, Y. Li, and T. Jiang, "Mobility-aware joint task scheduling and resource allocation for cooperative mobile edge computing," *IEEE Trans. Wireless Commun.*, vol. 20, no. 1, pp. 360–374, Jan. 2021.

[27] J. Zhang, H. Guo, J. Liu, and Y. Zhang, "Task offloading in vehicular edge computing networks: A load-balancing solution," *IEEE Trans. Veh. Technol.*, vol. 69, no. 2, pp. 2092–2104, Feb. 2020.

[28] Y. Zhu, T. Yang, Y. Hu, W. Gao, and A. Schmeink, "Optimal-delay-guaranteed energy efficient cooperative offloading in VEC networks," in *Proc. IEEE Global Commun. Conf.*, Dec. 2020, pp. 1–6.

[29] L. Liu, C. Chen, Q. Pei, S. Maharjan, and Y. Zhang, "Vehicular edge computing and networking: A survey," *Mobile Netw. Appl.*, vol. 26, pp. 1145–1168, Jul. 2020.

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

[31] M. Dayarathna, Y. Wen, and R. Fan, "Data center energy consumption modeling: A survey," *IEEE Commun. Surveys Tuts.*, vol. 18, no. 1, pp. 732–794, 1st Quart., 2015.

[32] 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.

[33] H. Tang, L. Chen, Y. Wang, N. Wang, and X. Li, "Stalling assessment for wireless online video streams via ISP traffic monitoring," in *Proc. IEEE Wireless Commun. Netw. Conf. (WCNC)*, Mar. 2017, pp. 1–6.

[34] W. Tang, X. Zhao, W. Rafique, L. Qi, W. Dou, and Q. Ni, "An offloading method using decentralized P2P-enabled mobile edge servers in edge computing," *J. Syst. Archit.*, vol. 94, pp. 1–13, Mar. 2019.

[35] 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.

[36] H. ElSawy, A. Sultan-Salem, M. S. Alouini, and M. Z. Win, "Modeling and analysis of cellular networks using stochastic geometry: A tutorial," *IEEE Commun. Surveys Tuts.*, vol. 19, no. 1, pp. 167–203, 1st Quart., 2017.

[37] M. Di Renzo, A. Guidotti, and G. E. Corazza, "Average rate of downlink heterogeneous cellular networks over generalized fading channels: A stochastic geometry approach," *IEEE Trans. Commun.*, vol. 61, no. 7, pp. 3050–3071, Jul. 2013.

[38] 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.

[39] D. Bachrathy and G. Stépán, "Bisection method in higher dimensions and the efficiency number," *Periodica Polytechnica Mech. Eng.*, vol. 56, no. 2, pp. 81–86, Aug. 2012.

[40] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge, U.K.: Cambridge Univ. Press, 2004.

[41] D. P. Palomar and M. Chiang, "A tutorial on decomposition methods for network utility maximization," *IEEE J. Sel. Areas Commun.*, vol. 24, no. 8, pp. 1439–1451, Aug. 2006.

[42] M. Behrisch, L. Bieker, J. Erdmann, and D. Krajzewicz, "SUMO—Simulation of urban mobility: An overview," in *Proc. 3rd Int. Conf. Adv. Syst. Simul. (SIMUL)*, ThinkMind, 2011, pp. 63–68.

[43] H. Zhang, J. Lee, T. Q. Quek, and I. Chih-Lin, *Ultra-Dense Networks: Principles and Applications*. Cambridge, U.K.: Cambridge Univ. Press, 2020.

[44] *Road Traffic Statistics*. Accessed: Jul. 6, 2022. [Online]. Available: https://gits.gg.go.kr/web/main/index.do

Hewon Cho (Graduate Student Member, IEEE) received the B.S. degree in electronics engineering from Kyungpook National University (KNU), Daegu, South Korea, in 2017, and the M.S. degree from the Department of Information and Communication Engineering, Daegu Gyeongbuk Institute of Science and Technology (DGIST), Daegu, in 2019, where she is currently pursuing the Ph.D. degree with the Department of Information and Communication Engineering. Her current research interests include vehicular networks, edge computing enabled wireless networks, and intelligent networking.

Ying Cui (Member, IEEE) received the B.E. degree in electronic and information engineering from Xi'an Jiao Tong University, China, in 2007, and the Ph.D. degree in electronic and computer engineering from The Hong Kong University of Science and Technology (HKUST), Hong Kong, in 2012. She held visiting positions with Yale University, USA, in 2011, and Macquarie University, Australia, in 2012. From 2012 to 2013, she was a Post-Doctoral Research Associate with the Department of Electrical and Computer Engineering, Northeastern University, Boston, MA, USA. From 2013 to 2014, she was a Post-Doctoral Research Associate with the Department of Electrical Engineering and Computer Science, Massachusetts Institute of Technology (MIT), Cambridge, MA, USA. Since 2015, she has been an Associate Professor with the Department of Electronic Engineering, Shanghai Jiao Tong University, China. Her current research interests include optimization, learning, mobile edge caching and computing, and delay-sensitive cross-layer control. She was selected to the Thousand Talents Plan for Young Professionals of China in 2013. She received Best Paper Awards from IEEE ICC 2015 and IEEE GLOBECOM 2021. She serves as an Editor for IEEE TRANSACTIONS ON WIRELESS COMMUNICATIONS.

Jemin Lee (Member, IEEE) received the B.S. (Hons.), M.S., and Ph.D. degrees in electrical and electronic engineering from Yonsei University, Seoul, South Korea, in 2004, 2007, and 2010, respectively. She was a Post-Doctoral Fellow with the Massachusetts Institute of Technology (MIT), Cambridge, MA, USA, from 2010 to 2013, a Temasek Research Fellow with the iTrust, Centre for Research in Cyber Security, Singapore University of Technology and Design (SUTD), Singapore, from 2014 to 2016, and an Associate Professor with the Department of Information and Communication Engineering, Daegu Gyeongbuk Institute of Science and Technology (DGIST), Daegu, South Korea, from 2016 to 2021. She is currently an Associate Professor with the Department of Electrical and Computer Engineering, Sungkyunkwan University (SKKU), Suwon, South Korea. Her current research interests include wireless communications, wireless security, intelligent networking, and blockchain networks.

She received the Haedong Young Engineering Researcher Award in 2020, the IEEE ComSoc Young Author Best Paper Award in 2020, the IEEE ComSoc AP Outstanding Paper Award in 2017, the IEEE ComSoc AP Outstanding Young Researcher Award in 2014, the Temasek Research Fellowship in 2013, and the Chun-Gang Outstanding Research Award in 2011. She serves as the Chair for the IEEE Communication Society (ComSoc) Radio Communications Technical Committee (RCC). She has been serves as the Symposium Chair/the Track Chair for a number of international conferences, including 2020 IEEE Globecom and 2020 IEEE WCNC. She is currently an Editor of the IEEE TRANSACTIONS ON COMMUNICATIONS, the IEEE WIRELESS COMMUNICATIONS LETTERS, and the IEEE OPEN JOURNAL OF VEHICULAR TECHNOLOGY. She was an Editor of the IEEE TRANSACTIONS ON WIRELESS COMMUNICATIONS and the IEEE COMMUNICATION LETTERS from 2014 to 2019.