IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received April 7, 2019, accepted May 3, 2019, date of publication May 9, 2019, date of current version May 24, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2915959

Computation Offloading for Mobile Edge Computing Enabled Vehicular Networks

JUN WANG[1], DAQUAN FENG[1], SHENGLI ZHANG[1], JIANHUA TANG[2], (Member, IEEE), AND TONY Q. S. QUEK[3], (Fellow, IEEE)

[1]Guangdong Key Laboratory of Intelligent Information Processing, Shenzhen University, Shenzhen 518060, China
[2]Institute of New Media and Communications (INMC), Seoul National University, Seoul 08826, South Korea
[3]Information Systems Technology and Design Pillar, Singapore University of Technology and Design, Singapore 487372

Corresponding author: Daquan Feng (fdquan@gmail.com)

This work was supported in part by the Natural Science Foundation of China under Grant 61701317 and Grant 61771315, in part by the Young Elite Scientists Sponsorship Program by the CAST under Grant 2018QNRC001, in part by the Guangdong Natural Science Foundation under Grant 2017A030310371, in part by the Guangdong Education Project under Grant 2016KZDXM006, in part by the Shenzhen Basic Research Program under Grant JCYJ20170302150006125, Grant JCYJ20160226192223251, Grant JCYJ20170412104656685, and Grant JCYJ20170302142312688, in part by the Start-up Fund of Shenzhen University under Grant 2017076, in part by the Tencent "Rhinoceros Birds" Scientific Research Foundation for Young Teachers of Shenzhen University, and in part by the Start-up Fund of the Peacock Project.

ABSTRACT The emergence of computation-intensive and delay-sensitive vehicular applications poses a great challenge for individual vehicles with limited computation resources. Mobile edge computing (MEC) is a new paradigm shift that can enhance vehicular services through computation offloading. However, the high mobility of vehicles will affect offloading performance. In this paper, we investigate the vehicular user (VU) computation overhead minimization problem in MEC-enabled vehicular networks by jointly optimizing the computation and communication resources' allocation (transmit power and uploading time for communication, and the offloading ratio and local CPU frequency for computation). This optimization problem is nonconvex and difficult to solve directly. To deal with this issue, we first transform the original problem into an equivalent one. Then, we decompose the equivalent problem into a two-level problem. In addition, we develop a low-complexity algorithm to obtain the optimal solution. The numerical results demonstrate that the proposed algorithm can significantly outperform benchmark algorithms in terms of computation overhead.

INDEX TERMS Mobile edge computing, vehicular networks, computation offloading, resource allocation.

I. INTRODUCTION

Along with the increasing number of connected autonomous vehicles, various computation-intensive and delay-sensitive applications are emerging, such as image-aided navigation and augmented reality (AR) driving. These applications require a significant amount of computation resources for real-time processing and analysis of the huge volume of sensing data, which imposes a great challenge to individual VUs with limited computation resources.

To address the problem, mobile cloud computing (MCC) is proposed as a promising approach, where the computation tasks are offloaded to remote cloud servers through wireless networks. Although MCC significantly improves computation performance and resource utilization, the delay fluctuation greatly reduces the offloading efficiency due to the long distance transmission between the VU and the cloud servers [1]. Mobile edge computing (MEC) is immediately introduced to cope with this issue, where computation-servers are deployed at the edge of radio access networks [2], [3]. Thus, with MEC-enabled computation offloading, the VU can get faster interactive response or lower delay. However, compared with traditional cloud servers with powerful computation capabilities, the MEC servers usually endure the computation resource limitation. On the other hand, computation offloading brings some communication overheads (i.e., bandwidth and power), which is similar to the data offloading in [4], [5]. As a result, it is vital that how to efficiently allocate communication and computation resources for MEC-based vehicular networks to guarantee VU good experience.

There have been many works focusing on computation offloading scheme designs and resource allocation for MEC-enabled networks [6]– [12]. These offloading schemes

The associate editor coordinating the review of this manuscript and approving it for publication was Eyuphan Bulut.

is generally divided into two categories: binary offloading and partial offloading. In [6], a binary offloading decision has been proposed to minimize the energy consumption by optimizing the local CPU frequency and the data transmission rate. In order to minimize the weighted sum energy consumption and delay, a joint optimization framework of binary offloading decision and local CPU frequency has been proposed in [7]. In [8], the weighted improvement of energy consumption and delay minimization problem has been considered through optimizing offloading decisions, local CPU frequency, and transmit power. Binary offloading scheme designs have been further extended to the wireless powered MEC systems in [9], [10], where energy consumption minimization or computation rate maximization problems were considered, respectively. However, for the data partitioned oriented applications, partial offloading schemes are more appropriate because it takes advantage of parallel process between the local users and MEC servers. In [11], two partial offloading schemes have been proposed to minimize the energy consumption subject to a delay constraint or minimize the delay subject to a energy consumption constraint, respectively. Furthermore, various machine learning-based approaches for MEC are also summarized in [12].

Recently, several offloading strategy designs have been extended to the MEC-enabled vehicular networks [13]–[18]. A stackelberg game theory based approach has been proposed in [13] to design an optimal multilevel offloading scheme, where the author aimed to maximize the utilities of both the vehicles and the VEC servers. In [14], a game theory based offloading scheme has been proposed to minimize the delay. In [15], the authors have proposed a joint load balancing and offloading solution to maximize system utility. Moreover, some deep reinforcement learning approaches have been proposed in [16]– [18] to determine the resource allocation policy for vehicular networks. In [16], a joint resource allocation of communication, caching and computing based on deep reinforcement learning has been proposed. This work has been further extended in [17], [18] by taking the vehicles' mobility and the hard service deadline into account.

Unfortunately, the aforementioned works, it is assumed that the wireless channels keep constant during computation offloading. In fact, this assumption is impractical, because the wireless channel may change when vehicles move fast, which may influences the offloading performance. Thus, in this paper, we consider more practical case that the wireless channel changes during computation offloading. Specifically, we study the computation overhead problem for MEC-enabled vehicular networks, and propose a joint allocation scheme of computation and communication resources in order to minimize the computation overhead. The main contributions of this paper are summarized as follows.

- With considering the impact of the channel change, we aim at minimizing the weighted sum of the latency and the energy consumption (referred to as *computation overhead*) of the VU by jointly optimizing transmit power, the uploading time, as well as the offloading ratio and local CPU frequency. This problem of interests is nonconvex and thus difficult to solve.
- In order to solve this problem, we first transform it into an equivalent form. Then, we decompose the equivalent problem into a two-level problem. In the lower-level problem, we jointly optimize the transmit power, offloading ratio, and the latency, given the uploading time while in the higher-level problem, optimizing the uploading time. Specifically, we derive the optimal solution in a semi-closed form for the lower-level problem by leveraging Lagrange duality method. In the high-level problem, one-dimensional line search method is used.
- In terms of the performance evaluation, we verify the performance of the proposed algorithm through extensive numerical simulations. Furthermore, we compare the proposed algorithm with three solutions: local computing only, partial offloading with fixed local CPU frequency, and SDR-based Method [7]. The results illustrate that the proposed algorithm can significantly achieve a performance improvement from computation offloading in terms of the computation overhead.

The rest of the paper is organized as follows. Section II presents the system model, computation model, and problem formulation. In Section III, we develop an efficient algorithm to solve the proposed formulation. Sections IV provides simulation results to verify the advantages of the proposed method. Finally, conclusion is given in Section V.

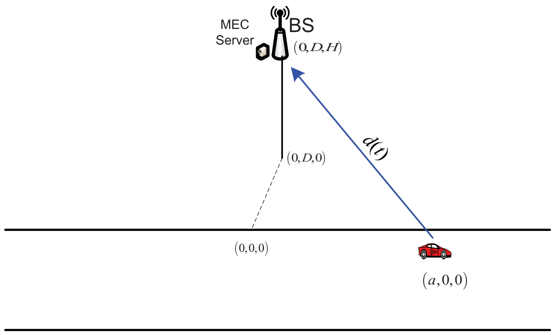


FIGURE 1. The System Model.

II. SYSTEM MODEL AND PROBLEM FORMULATION

In this section, we first introduce the system model. Then, computation model is presented. Finally, we formulate the optimization problem.

A. SYSTEM MODEL

Consider a MEC-enabled vehicular system as shown in Fig.1, where there exists a base station (BS) equipped with MEC server and a mobile vehicle within the coverage of the BS. The computing processor of vehicle is an on-chip microprocessor with low computing capability while the MEC server has a powerful processor. Thus, for computation-intensive and delay-sensitive task, the VU needs to offload partial task to the MEC server for fast processing due to its limited computation capacity.

For the sake of presentation, a three-dimensional Euclidean coordinate is adopted. The BS is assumed to be located at $(0, D, H)$, where D denote the distance between the BS and highway and H is the height of the BS antenna. Moreover, the VU unidirectionally moves along the highway from the location $(a, 0, 0)$ at speed v. Thus, the time-varying distance from the VU to the BS can be expressed as

$$d(t) = \sqrt{H^2 + D^2 + (a + vt)^2}. \tag{1}$$

As mentioned in [19], there will be many roadside units (served as BSs) located along the road to provide services for the VUs on the road in the future. Moreover, we consider the VUs adopt the orthogonal channels to transmit information and thus there is no interference between the VUs. Hence, transmission performance from the VU to the BS is mainly affected by the distance between them. Therefore, we here use the same channel model as in [19], [20]. Although the channel model is simple, the optimal solution obtained in our paper can be served as a performance benchmark. For more complicated channel models, we will leave it as our future work. So the channel power gain $G(t)$ between them is given by

$$G(t) = \rho_0 d(t)^{-\theta} = \frac{\rho_0}{\left[H^2 + D^2 + (a + vt)^2\right]^{\frac{\theta}{2}}}, \tag{2}$$

where ρ_0 is the channel power gain at a reference distance $d_0 = 1$ and θ is the path-loss exponent.

Let p and σ^2 denote the transmit power of the VU and the noise power at the BS receiver, respectively. Then, the instantaneous transmission rate $r(t)$ between the VU and the BS can be expressed as

$$r(t) = B\log_2\left(1 + \frac{pG(t)}{\sigma^2}\right), \tag{3}$$

where B denotes the channel bandwidth.

B. COMPUTATION MODEL

At the location of $(a, 0, 0)$, when the VU generates a computation-intensive and delay-sensitive task, it needs to offload to the BS/MEC server for fast processing. When the computation is finished, the computation results will be returned to the VU. In general, the task is modeled as a profile with three parameters, $\{L, C, T_{\max}\}$, where L, C, and T_{max} denote the task input-data size (in bits), computation intensity (in CPU cycles/bit), and maximum allowable delay (in s), respectively. All three parameters rely on the nature of the task and can be estimated through task profilers [21]. Similar to the assumption in [11], [15], [22], the task can be divided into two parts: $(1-\alpha)L$ bits for local computing and αL bits for MEC server computing, where α is the offloading ratio.

1) LOCAL COMPUTING AT THE VU

We first consider the $(1-\alpha)L$ bits data is processed at the VU. Let f_l denote the local CPU frequency (i.e. CPU cycles/s). The local computing delay can be then given by

$$T_l = \frac{(1-\alpha)LC}{f_l}. \tag{4}$$

As in [6], the energy consumption of local computing can be expressed as

$$E_l = \zeta f_l^3 T_l = \zeta LC f_l^2 (1-\alpha), \tag{5}$$

where ζ is the effective switched capacitance relating to chip architecture.

2) COMPUTATION OFFLOADING

When αL bits data is offloaded to the MEC server, the computing latency can be given by

$$T_r = t_{\text{up}} + t_{\text{comp}} + t_{\text{dn}}. \tag{6}$$

In (6), t_{comp} is computation time at the MEC server and given by $t_{\text{comp}} = \frac{\alpha LC}{F_{\text{mec}}}$, where F_{mec} is the computation capability of MEC. t_{up} and t_{dn} denote the uploading time and downloading time, respectively. Similar to [9], [10], and [23], the downloading time t_{dn} can be neglected since the data resulted from computation is usually with a small size. Moreover, t_{up} is determined by the integral of the transmission rate $\int_0^{t_{\text{up}}} r(\tau)d\tau = \alpha L$. The energy consumption of the VU in this process is caused by uploading task data and thus can be expressed as

$$E_r = pt_{\text{up}}. \tag{7}$$

C. PROBLEM FORMULATION

Since local computing and computation offloading take place simultaneously, the latency of the VU to execute the whole task can be given by

$$T = \max\{T_l, T_r\} = \max\left\{\frac{(1-\alpha)LC}{f_l}, t_{\text{up}} + \frac{\alpha LC}{F_{\text{mec}}}\right\}. \tag{8}$$

The energy consumption of the VU to finish the whole task can be expressed as

$$E = E_l + E_r = \zeta LC f_l^2 (1-\alpha) + pt_{\text{up}}. \tag{9}$$

Our goal is to minimize the computation overhead caused by communication and computation. Here, the computation overhead is defined as the weighted sum of the latency and the energy consumption, $\beta_T T + \beta_E E$, where β_T and β_E represent the weights of the latency and the energy consumption of the VU, respectively. As a result, the optimization problem of interests can be expressed as

$$\begin{aligned}
&\min_{p,\alpha,t_{\text{up}},f_l} \quad \beta_T T + \beta_E E && (10)\\
&\text{s.t.} \quad \varphi(p, t_{\text{up}}) = \alpha L, && (10a)\\
&\quad \sqrt{D^2 + (a + vT_r)^2} \le R_{\max}, && (10b)\\
&\quad 0 \le \alpha \le 1. && (10c)\\
&\quad 0 \le p \le P_{\max}, && (10d)\\
&\quad 0 \le f_l \le F_{\max}, && (10e)
\end{aligned}$$

where $\varphi\left(p, t_{\rm up}\right) \triangleq \int_0^{t_{\rm up}} r(\tau) d\tau$. In (10), the objective function could be considered as a tradeoff between the latency and energy consumption. The weights can be dynamically adjusted according to the remaining energy and maximum allowable delay. For example, a VU with less remaining energy can increase β_E to save more energy at the expense of longer task completion latency. Otherwise, if a VU is sensitive to processing latency, it can increase β_T to save more latency at the expense of high energy consumption. Constraint (10a) denotes the size of the task to be offloaded. Constraint (10b) ensures that the link between the VU and the BS is within the maximum transmission range. Constraints (10c), (10d), and (10e) guarantee that the offloading ratio, the transmit power, and local CPU frequency do not exceed their maximum values, respectively.

It can be observed that problem (10) is nonconvex due to the coupling of multiple variables, and hence is challenging to solve directly. In the next section, we will first transform this problem into a more tractable one and then derive the optimal solution.

III. A TWO-LEVEL SOLUTION APPROACH

In this section, we first transfer problem (10) into an equivalent form. Then, we derive the optimal solution of this equivalent problem.

A. PROBLEM TRANSFORMATION

Substituting (8) and (9) into (10), the problem can be rewritten as

$$\begin{aligned}
\min_{p,\alpha,t_{\rm up},f_l,T} \quad & \beta_T T + \beta_E \left(\varsigma L C f_l^2 (1-\alpha) + p t_{\rm up}\right) && (11)\\
\text{s.t.} \quad & \varphi\left(p, t_{\rm up}\right) = \alpha L, && (11a)\\
& \frac{(1-\alpha) LC}{f_l} \le T, && (11b)\\
& t_{\rm up} + \frac{\alpha LC}{F_{\rm mec}} \le T, && (11c)\\
& t_{\rm up} + \frac{\alpha LC}{F_{\rm mec}} \le c, && (11d)\\
& \text{(10c), (10d), and (10e)},
\end{aligned}$$

where $c \triangleq \frac{\sqrt{R_{\max}^2 - D^2} - a}{v}$.

It is easily observed that the objective function in (11) decreases monotonically with decrease of f_l. Moreover, from constraints (10e) and (11b), we have $\frac{(1-\alpha)LC}{T} \le f_l \le F_{\max}$. Therefore, the optimal f_l is given by

$$f_l^* = \frac{(1-\alpha) LC}{T} \qquad (12)$$

only when the following inequality holds

$$\frac{(1-\alpha) LC}{T} \le F_{\max}. \qquad (13)$$

Moreover, we also discover that constraint (11a) can be relaxed as

$$\varphi\left(p, t_{\rm up}\right) \ge \alpha L. \qquad (14)$$

In fact, (11a) in problem (11) can be equivalently replaced by (14). To prove this, we assume that $\left(p^*, \alpha^*, t_{\rm up}^*, f_l^*, T^*\right)$ is the optimal solution of problem (11) with the relaxed constraint (14). It is easy to see that if we decrease p^* while guaranteeing that all the other constraints in (11) are satisfied, the objective function will decrease. It contradicts the assumption that $\left(p^*, \alpha^*, t_{\rm up}^*, f_l^*, T^*\right)$ is the optimal solution. Hence, for problem (11) with the relaxed constraint (14), constraint (14) must be active at the optimum.

As a result, (11) can be equivalently transformed into the following problem

$$\begin{aligned}
\min_{p,\alpha,t_{\rm up},T} \quad & \beta_T T + \beta_E \left(\varsigma L^3 C^3 \frac{(1-\alpha)^3}{T^2} + p t_{\rm up}\right) && (15)\\
\text{s.t.} \quad & \alpha L \le \varphi\left(p, t_{\rm up}\right), && (15a)\\
& 1 - \alpha \le \frac{F_{\max}}{LC} T, && (15b)\\
& \text{(11c), (11d), (10c), and (10d).}
\end{aligned}$$

B. OPTIMAL SOLUTION

Although the original problem (10) is simplified to (15), it is still hard to solve due to the coupling of multiple variables. To cope with this challenge, we decompose (15) into a two-level problem. In the lower-level problem, we jointly optimize the transmit power p, offloading ratio α, and the latency T given the uploading time $t_{\rm up}$ while in the higher-level problem, optimizing the uploading time $t_{\rm up}$.

1) LOWER-LEVEL PROBLEM

For a given value of $t_{\rm up}$, the resulting lower-level problem can be expressed as,

$$\begin{aligned}
\min_{p,\alpha,T} \quad & \beta_T T + \beta_E \left(\varsigma L^3 C^3 \frac{(1-\alpha)^3}{T^2} + p t_{\rm up}\right) && (16)\\
\text{s.t.} \quad & 0 \le \alpha \le \overline{\alpha}, && (16a)\\
& \text{(15a), (15b), (11c), and (10d),}
\end{aligned}$$

where $\overline{\alpha} \triangleq \min\left\{1, \frac{(c - t_{\rm up}) F_{\rm mec}}{LC}\right\}$.

In the objective function of (16), it is easy to observe that the function $\varsigma L^3 C^3 \frac{(1-\alpha)^3}{T^2}$ is convex in (α, T) and the function $p t_{\rm up}$ is linear in p. Thus, the objective function is convex. For constraint (15a), the right-hand side $\varphi\left(p, t_{\rm up}\right)$ is concave due to the fact that the integral of a concave function with respect to p is still concave. Hence, constraint (15a) is convex. Therefore, problem (16) is convex, which can be solved by the interior point method [24]. However, to provide useful insights, we next exploit the Lagrange duality method to obtain the optimal solution in a semi-closed form for problem (16).

Let λ_1, λ_2, and λ_3 denote Lagrange multipliers associated with constraints (15a), (15b) and (11c), respectively. Define $\boldsymbol{\lambda} \triangleq (\lambda_1, \lambda_2, \lambda_3)$. Then the partial Lagrangian of (16) is expressed as

$$\mathcal{L}(p,\alpha,T,\boldsymbol{\lambda}) = \lambda_2+\lambda_3 t_{up}+\left(\beta_E p t_{\text{up}}-\lambda_1\varphi\left(p,t_{\text{up}}\right)\right) +\beta_E\zeta L^3C^3\frac{(1-\alpha)^3}{T^2} +\left(\lambda_1 L-\lambda_2+\lambda_3\frac{LC}{F_{\text{mec}}}\right)\alpha +\left(\beta_T-\lambda_2\frac{F_{\max}}{LC}-\lambda_3\right)T. \tag{17}$$

The dual function is given by

$$\Phi(\boldsymbol{\lambda}) = \min_{p,\alpha,T\geq 0} \mathcal{L}(p,\alpha,T,\boldsymbol{\lambda}) \tag{18}$$
$$\text{s.t.} \quad \text{(10d) and (16a)}. \tag{18a}$$

As a result, the dual problem is given by

$$\max_{\boldsymbol{\lambda}\geq 0} \Phi(\boldsymbol{\lambda}) \tag{19}$$

As problem (16) is convex and satisfies the Slater's condition, strong duality holds. Therefore, we can solve its dual problem (19) to obtain the optimal solution for problem (16). To solve dual problem (19), we need to evaluate $\Phi(\boldsymbol{\lambda})$ in (18) under any given $\boldsymbol{\lambda}$. Furthermore, we discover that (18) can be decomposed into two subproblems as follows

$$\Phi_p(\boldsymbol{\lambda}) = \min_{p} \beta_E p t_{\text{up}}-\lambda_1\varphi\left(p,t_{\text{up}}\right) \tag{20}$$
$$\text{s.t.} \quad \text{(10d)}. \tag{20a}$$

$$\Phi_{\alpha,T}(\boldsymbol{\lambda}) = \min_{\alpha,T\geq 0} \lambda_2+\lambda_3 t_{\text{up}}+\beta_E\zeta L^3C^3\frac{(1-\alpha)^3}{T^2} +\left(\lambda_1 L-\lambda_2+\lambda_3\frac{LC}{F_{\text{mec}}}\right)\alpha +\left(\beta_T-\lambda_2\frac{F_{\max}}{LC}-\lambda_3\right)T \tag{21}$$
$$\text{s.t.} \quad \text{(16a)}. \tag{21a}$$

In what follows, we separately evaluate $\Phi_p(\boldsymbol{\lambda})$ and $\Phi_{\alpha,T}(\boldsymbol{\lambda})$ and then combine them to obtain $\Phi(\boldsymbol{\lambda})$ together. The optimal solutions of (20) and (21) are given by the following two lemmas.

Lemma 1: For a given $\boldsymbol{\lambda}$, the optimal solution p^* to (20) is given by

$$p^* = \begin{cases} 0, & \text{if } \widehat{p}<0 \\ \widehat{p}, & \text{if } 0\leq\widehat{p}\leq P_{\max} \\ P_{\max}, & \text{if } \widehat{p}>P_{\max} \end{cases} \tag{22}$$

where $\widehat{p}$ is the root of the equation $\beta_E t_{up}-\lambda_1\varphi'\left(p,t_{up}\right)=0$.

Proof: Observe that the objective function in (20) is strictly convex in p. Thus, the equation $\beta_E t_{\text{up}}-\lambda_1\varphi'\left(p,t_{\text{up}}\right)=0$ has an unique root, denoted as $\widehat{p}$, where $\varphi'\left(p,t_{\text{up}}\right)\triangleq\frac{\partial\varphi(p,t_{\text{up}})}{\partial p}$. If $\widehat{p}<0$, the objective function increases monotonically in $[0,P_{\max}]$. In this case, $p^*=0$. If $\widehat{p}>P_{\max}$, the objective function decreases monotonically in $[0,P_{\max}]$. In this case, $p^*=P_{\max}$. If $0\leq\widehat{p}\leq P_{\max}$, the objective function increases monotonically in $[0,\widehat{p}]$ and decreases monotonically in $(\widehat{p},P_{\max}]$. In this case, $p^*=\widehat{p}$. ■

Algorithm 1 Solve Problem (16) Using Ellipsoid Method

1: **Initialize:** Give an initial ellipsoid $\varepsilon(\boldsymbol{\lambda},S)$ containing the optimal solution $\boldsymbol{\lambda}^*$ for (19).
2: **repeat**
3: Calculate (p^*,α^*,T^*) under given $\boldsymbol{\lambda}$ from Lemma 1 and Lemma 2;
4: Update $\boldsymbol{\lambda}$ based on the ellipsoid method [24];
5: **until** $\boldsymbol{\lambda}$ converges with a pre-defined threshold.
6: **Set** $\boldsymbol{\lambda}^*=\boldsymbol{\lambda}$.
7: **Output:** Calculate (p^*,α^*,T^*) by using Lemma 1 and Lemma 2 when $\boldsymbol{\lambda}=\boldsymbol{\lambda}^*$.

Moreover, note that $\beta_E t_{\text{up}}-\lambda_1\varphi'\left(p,t_{\text{up}}\right)=0$ is a transcendental equation with respect to p, we can find the root $\widehat{p}$ by the bisection search method.

Lemma 2: For a given $\boldsymbol{\lambda}$, the optimal solution (α^*,T^*) to (21) satisfies

$$\alpha^* = \begin{cases} 0, & \text{if } \sqrt{\frac{\lambda_1 L-\lambda_2+\lambda_3\frac{LC}{F_{\text{mec}}}}{3\beta_E\zeta L^3C^3}} > \frac{1}{T^*} \\ [0,\overline{\alpha}], & \text{if } \sqrt{\frac{\lambda_1 L-\lambda_2+\lambda_3\frac{LC}{F_{\text{mec}}}}{3\beta_E\zeta L^3C^3}} = \frac{1}{T^*} \\ \overline{\alpha}, & \text{if } \sqrt{\frac{\lambda_1 L-\lambda_2+\lambda_3\frac{LC}{F_{\text{mec}}}}{3\beta_E\zeta L^3C^3}} < \frac{1}{T^*} \end{cases} \tag{23}$$

$$T^* = \frac{1-\alpha^*}{\sqrt{\frac{\beta_T-\lambda_2\frac{F_{max}}{LC}-\lambda_3}{2\beta_E\zeta L^3C^3}}}. \tag{24}$$

Proof: See Appendix A. ■

Using Lemma 1 and Lemma 2, the dual function $\Phi(\boldsymbol{\lambda})$ is computed for any given $\boldsymbol{\lambda}$. Next, we solve (19). Nevertheless, the dual function $\Phi(\boldsymbol{\lambda})$ is generally concave but non-differentiable, so we can use the subgradient method or the ellipsoid method [24] to obtain the optimal solution $\boldsymbol{\lambda}^*$ for (19). In this paper, we use the ellipsoid method. Finally, by replacing $\boldsymbol{\lambda}$ in Lemma 1 and Lemma 2 as $\boldsymbol{\lambda}^*$, we have the optimal solution (p^*,α^*,T^*) for (16).

Until now, we have solved (16) and obtained the optimal solution (p^*,α^*,T^*) for a given t_{up}. The corresponding algorithm is summarized in Algorithm 1, where $\varepsilon(\boldsymbol{\lambda},S)$ denotes an ellipsoid with the center of $\boldsymbol{\lambda}$ and the volume of S.

2) HIGHER-LEVEL PROBLEM

Let $\phi\left(t_{\text{up}}\right)$ denote the optimal value of (16) for a given t_{up}. The higher-level problem aims at minimizing $\phi\left(t_{\text{up}}\right)$ as follows

$$\min_{t_{\text{up}}} \phi\left(t_{\text{up}}\right) \tag{25}$$
$$\text{s.t.} \quad 0\leq t_{\text{up}}\leq\bar{t}_{\text{up}}, \tag{25a}$$

where $\bar{t}_{\text{up}}\triangleq c$ can be obtained from constraint (11d) in problem (11).

Since problem (25) only involves a single-variable t_{up} within the interval $[0,c]$, we can adopt the one-dimensional linear search to solve it. With enough small step size of the

Algorithm 2 Solve Problem (25) Using One-Dimensional Search

Initialize: $\phi^{\text{temp}} = \text{Inf}$, $t_{\text{up}}^{\text{temp}} = 0$, and Δ.
for $t_{\text{up}} = 0 : \Delta : c$ **do**
  Calculate $\phi\left(t_{\text{up}}\right)$ by carrying out Algorithm 1.
  if $\phi\left(t_{\text{up}}\right) < \phi^{\text{temp}}$ **then**
    Set $\phi^{\text{temp}} = \phi\left(t_{\text{up}}\right)$ and $t_{\text{up}}^{\text{temp}} = t_{\text{up}}$.
  end if
end for
Set $t_{\text{up}}^{\text{opt}} = t_{\text{up}}^{\text{temp}}$.
Output: The optimal solution $t_{\text{up}}^{\text{opt}}$ to (25) and corresponding p^{opt}, α^{opt}.

TABLE 1. Default simulation parameters.

Parameter	Description	Value
H	Height of the BS antenna	25 m
D	Distance between the BS and highway	35 m
θ	Path-loss exponent	4
ρ_0	Reference channel power	-30 dB
B	Bandwidth	2 MHz
σ^2	Noise power	-104 dBm
$P_{\max}$	Maximum VU transmit power	23 dBm
v	Vehicle speed	100 Km/h
$R_{\max}$	Maximum transmission range	200 m
L	Task input-data size	1 MB
C	Task computation intensity	1900/8 cycles/bit
$F_{\max}$	Maximum local computation capability	1.2 GHz
F_{mec}	MEC computation capability	4 GHz
ζ	Energy consumption coefficient	1.25×10^{-26}
β_T	Weight of the latency	0.5

search, we can obtain the global optimal solution. The whole procedure is summarized in Algorithm 2.

IV. NUMERICAL RESULTS

A. EXPERIMENT SETUP

In this section, numerical results are provided to validate the performance of the proposed algorithm in the MEC-enabled vehicular networks. The default simulation parameters are listed in Table 1 [14], [25], unless mentioned otherwise. For comparison, we take the following three most related schemes as benchmarks.

- *Local computing only* (referred to as '***LC***'): This scheme corresponds to solving problem (11) (or problem (10)) by setting $\alpha = 0$. The resulting optimization problem is expressed as

$$\min_{f_l} \quad \beta_T \frac{LC}{f_l} + \beta_E \zeta LC f_l^2 \tag{26}$$
$$\text{s.t.} \quad 0 \le f_l \le F_{\max}. \tag{26a}$$

  Its optimal solution can be expressed as the following closed-form

$$f_l^{\text{opt}} = \begin{cases} \sqrt[3]{\frac{\beta_T}{2\beta_E \zeta}}, & \text{if } 0 \le \sqrt[3]{\frac{\beta_T}{2\beta_E \zeta}} \le F_{\max} \\ F_{\max}, & \text{otherwise} \end{cases} \tag{27}$$

- *Partial offloading with fixed local CPU frequency f_l* (referred to as '***PO with f_l***'): This scheme corresponds to solving problem (11) under fixed f_l. The resulting optimization problem can be solved by using similar method in this paper. Here, we set $f_l = 0.6F_{max}$.
- *SDR-based Method [7]*: In this scheme, the VU is static and corresponding wireless channel remains constant during the whole task execution process. To make a fair comparison, we make the following assumptions: 1) the VU is always located at the midpoint between the initial point and allowable maximum location; 2) the task data is transmitted at maximum power; 3) the VU has only one task which can be either executed locally or be offloaded to an AP.

In the proposed scheme and the above three schemes, the proposed scheme and PO with f_l belong to the type of partial offloading while SDR-based method [7] belongs to the type of binary offloading.

B. EFFECTIVE TASK OFFLOADING EVALUATION

1) IMPACT OF THE COMPUTATION TASK INPUT-DATA SIZE

Fig.2 shows the computation overheads of all the schemes for different task input-data size. It is observed that both the computation overheads of all the schemes increase with task input-data size L. As expected, the proposed scheme performs better than the other three schemes, since it takes fully advantages of partial offloading (PO) and dynamic voltage scaling (DVS) technology. Specifically, two partial offloading schemes, i.e., the proposed scheme and the PO with f_l scheme, outperform the other two schemes, which shows the superiority of PO. The reason is that two partial offloading schemes, computation task can be processed in parallel. Moreover, the proposed scheme surpasses the PO with f_l scheme, which confirms the benefit of DVS. This is because that the proposed scheme uses DVS to choose the optimal local CPU frequency so that more computation overhead is saved. Furthermore, we note that the computation overhead of the proposed scheme increases slowly with the task input-data size L. The reason for this is that a larger part of computation is offloaded to the MEC server as the input data size L increases, which leads to a small increase in computation overhead.

2) IMPACT OF THE TASK COMPUTATION INTENSITY

In Fig.3, we show the impact of the task computation intensity on the computation overhead. Here, the task computation intensity C is set as $C = 330/8, 1300/8, 1900/8, 5900/8, 8900/8$ cycles/bit [7], respectively. We observe that the computation overhead of all the schemes increases with the task computation intensity C. The proposed scheme performs better than the other three algorithms. It is mainly manifested in two aspects: one aspect is that the proposed scheme has the lowest computation overhead over different task computation intensity, the other is that the proposed scheme increases more slowly than the other three scheme

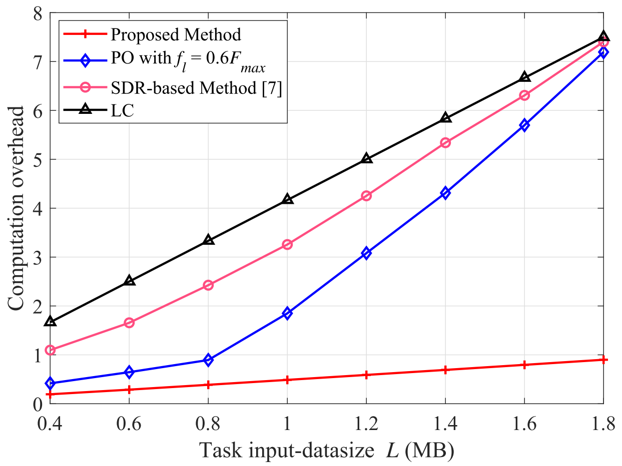


FIGURE 2. The computation overhead versus task input-data size L.

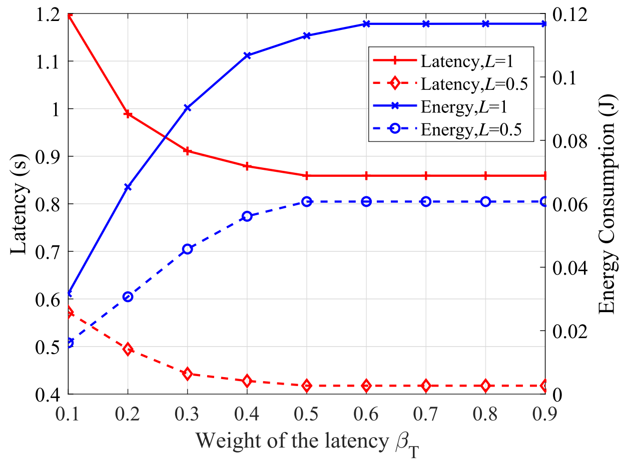


FIGURE 4. The proposed algorithm performance versus the weight of the latency β_T in terms of the latency and energy consumption.

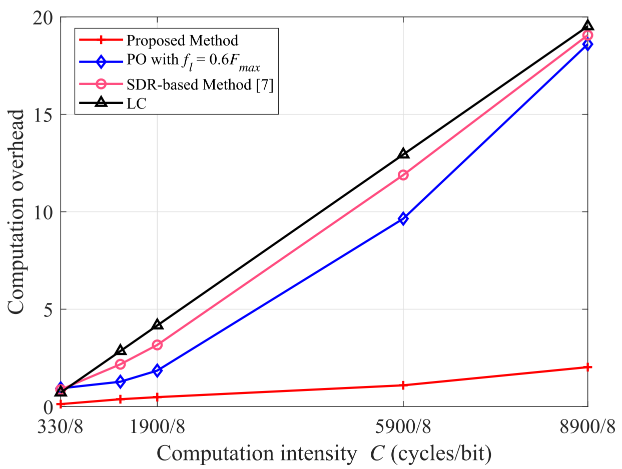


FIGURE 3. The computation overhead versus task computation intensity C.

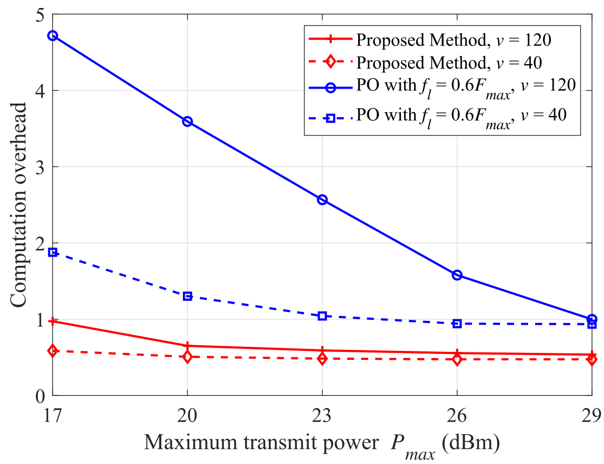


FIGURE 5. The computation overhead versus maximum transmit power of the VU.

in terms of the computation overhead. This possible reason for second advantage is that the larger the task computation intensity, the greater the percentage of computation offloaded to the MEC server. Thus, for the proposed scheme, increasing task computation intensity only brings a little growth of the computation overhead.

3) IMPACT OF THE WEIGHT OF THE LATENCY

Fig.4 shows the latency and energy consumption of the VU when the weight of the latency β_T increases from 0.1 and 0.9 meanwhile the weight of the energy consumption $\beta_E = 1 - \beta_T$ increases from 0.9 and 0.1. It is seen that the latency decreases when β_T increases, at the expense of larger energy consumption. In other words, the smaller the latency, the larger the energy consumption. It just exhibits the trade-off between the latency and energy consumption. Besides, we also observe that when $L = 0.5$, the VU experience a lower latency and energy consumption than in the case when $L = 1$. This observation agrees with the phenomenon shown in Fig.2.

4) IMPACT OF MAXIMUM TRANSMIT POWER

In Fig.5, we discuss the impacts of maximum transmit power on the performance of two schemes with partial offloading (i.e., the proposed scheme and the PO with f_l scheme). It is observed that as the maximum transmit power P_{max} increases, computation overhead decreases. In addition, when P_{max} is sufficiently large, the performance of the proposed scheme reaches saturation point under different speed v. This is because that 1) increasing maximum transmit power makes the VU offload larger part of computation to the MEC servers; 2) If the VU offloads more computation, the overhead caused by offloading is greater than the overhead by local computing. So the total computation overhead do not decrease with further increasing of P_{max}. Besides, we can also see that the faster the VU move, the larger the computation overhead.

In brief, the proposed method outperforms the other three methods in terms of computation overhead. It is because that the proposed method combines the advantages of partial offloading and dynamic voltage scaling technology. Moreover, there exists a tradeoff between the latency and energy consumption through adjusting the weights of the objective function.

V. CONCLUSION

In this paper, we investigate the computation overhead minimization problem by jointly optimizing communication and computation resources in MEC-enabled vehicular networks. The nonconvex problem is first transformed into an equivalent problem. Then, we decompose the equivalent problem into a two-level problem. Furthermore, we present a low-complexity algorithm to obtain the optimal solution. Numerical results show that the proposed scheme can achieve remarkable computation overhead saving.

APPENDIX A
PROOF OF LEMMA 2

Note that problem (21) is convex and satisfies the Slater's condition, so strong duality holds between it and its dual problem. Next, we can solve (21) via KKT conditions. The Lagrangian of problem (21) is given by

$$\begin{aligned}\widetilde{\mathcal{L}} &= \lambda_2 + \lambda_3 t_{\text{up}} - \eta_2\overline{\alpha} + \beta_E \zeta L^3 C^3 \frac{(1-\alpha)^3}{T^2} \\ &+ \left(\lambda_1 L - \lambda_2 + \lambda_3 \frac{LC}{F_{\text{mec}}} - \eta_1 + \eta_2\right)\alpha \\ &+ \left(\beta_T - \lambda_2 \frac{F_{\max}}{LC} - \lambda_3 - \vartheta\right) T \end{aligned} \tag{28}$$

where η_1, η_2, and ϑ denote the dual variables associated with constraints $\alpha \geq 0$, $\alpha \leq \overline{\alpha}$, and $T \geq 0$, respectively.

Let (α^*, T^*) and $\left(\eta_1^*, \eta_2^*\right)$ be the primal and dual optimal values, respectively. Then, according to KKT conditions, the following expressions hold

$$0 \leq \alpha^* \leq \overline{\alpha}, \ T^* \geq 0 \tag{29a}$$

$$\eta_1^* \geq 0, \ \eta_2^* \geq 0, \ \vartheta^* \geq 0 \tag{29b}$$

$$\eta_1^*\alpha^* = 0, \ \eta_2^*\left(\alpha^* - \overline{\alpha}\right) = 0, \ \vartheta^* T^* = 0 \tag{29c}$$

$$\begin{aligned}\frac{\partial \widetilde{L}}{\partial \alpha^*} &= -3\beta_E \zeta L^3 C^3 \frac{(1-\alpha^*)^2}{T^{*2}} + \lambda_1 L \\ &\quad -\lambda_2 + \lambda_3 \frac{LC}{F_{\text{mec}}} - \eta_1^* + \eta_2^* = 0 \end{aligned} \tag{29d}$$

$$\begin{aligned}\frac{\partial \widetilde{L}}{\partial T^*} &= -2\beta_E \zeta L^3 C^3 \frac{(1-\alpha^*)^3}{T^{*3}} + \beta_T - \lambda_3 \\ &\quad -\lambda_2 \frac{F_{\max}}{LC} - \vartheta^* = 0 \end{aligned} \tag{29e}$$

From (29d), we have

$$\frac{1-\alpha^*}{T^*} = \sqrt{\frac{\lambda_1 L - \lambda_2 + \lambda_3 \frac{LC}{F_{\text{mec}}} - \eta_1^* + \eta_2^*}{3\beta_E \zeta L^3 C^3}} \tag{30}$$

Next, we discuss the tightness of constraint (16a).

1) When $\alpha^* = \overline{\alpha}$, that is $\eta_1^* = 0$, $\eta_2^* > 0$, we have

$$\begin{aligned}\frac{1-\alpha^*}{T^*} &= \frac{1-\overline{\alpha}}{T^*} = \sqrt{\frac{\lambda_1 L - \lambda_2 + \lambda_3 \frac{LC}{F_{\text{mec}}} + \eta_2^*}{3\beta_E \zeta L^3 C^3}} \\ &> \sqrt{\frac{\lambda_1 L - \lambda_2 + \lambda_3 \frac{LC}{F_{\text{mec}}}}{3\beta_E \zeta L^3 C^3}} \end{aligned} \tag{31}$$

2) When $0 < \alpha^* < \overline{\alpha}$, that is $\eta_1^* = 0$, $\eta_2^* = 0$, we have

$$\frac{1-\alpha^*}{T^*} = \sqrt{\frac{\lambda_1 L - \lambda_2 + \lambda_3 \frac{LC}{F_{\text{mec}}}}{3\beta_E \zeta L^3 C^3}} \tag{32}$$

3) When $\alpha^* = 0$, that is $\eta_1^* > 0$, $\eta_2^* = 0$, we have

$$\begin{aligned}\frac{1-\alpha^*}{T^*} &= \frac{1}{T^*} = \sqrt{\frac{\lambda_1 L - \lambda_2 + \lambda_3 \frac{LC}{F_{\text{mec}}} - \eta_1^*}{3\beta_E \zeta L^3 C^3}} \\ &< \sqrt{\frac{\lambda_1 L - \lambda_2 + \lambda_3 \frac{LC}{F_{\text{mec}}}}{3\beta_E \zeta L^3 C^3}} \end{aligned} \tag{33}$$

Similarly, from (29e), we have

$$\frac{1-\alpha^*}{T^*} = \sqrt{\frac{\beta_T - \lambda_2 \frac{F_{\max}}{LC} - \lambda_3}{2\beta_E \zeta L^3 C^3}} \tag{34}$$

Based on (31)–(34) and with some algebraic operations, we have (23) and (24).

REFERENCES

[1] Y. Li, J. Liu, B. Cao, and C. Wang, "Joint optimization of radio and virtual machine resources with uncertain user demands in mobile cloud computing," *IEEE Trans. Multimedia*, vol. 20, no. 9, pp. 2427–2438, Sep. 2018.

[2] 5GAA White Paper. *Toward Fully Connected Vehicles: Edge Computing for Advanced Automotive Communications*. Accessed: Dec. 15, 2017. [Online]. Available: http://5gaa.org/news/toward-fully-connected-vehicles-edge-computing-for-advanced-automotive-communications/

[3] Q. Yuan, H. Zhou, J. Li, Z. Liu, F. Yang, and X. S. Shen, "Toward efficient content delivery for automated driving services: An edge computing solution," *IEEE Netw.*, vol. 32, no. 1, pp. 80–86, Jan./Feb. 2018.

[4] H. Zhou, H. Wang, X. Chen, X. Li, and S. Xu, "Data offloading techniques through vehicular ad hoc networks: A survey," *IEEE Access*, vol. 6, pp. 65250–65259, 2018.

[5] C.-M. Huang, Y.-F. Chen, S. Xu, and H. Zhou, "The vehicular social network (VSN)-based sharing of downloaded geo data using the credit-based clustering scheme," *IEEE Access*, vol. 6, pp. 58254–58271, 2018.

[6] W. Zhang, Y. Wen, K. Guan, D. Kilper, H. Luo and D. O. Wu, "Energy-optimal mobile cloud computing under stochastic wireless channel," *IEEE Trans. Wireless Commun.*, vol. 12, no. 9, pp. 4569–4581, Sep. 2013.

[7] 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.

[8] X. Lyu, H. Tian, C. Sengul, and P. Zhang, "Multiuser joint task offloading and resource optimization in proximate clouds," *IEEE Trans. Veh. Technol.*, vol. 66, no. 4, pp. 3435–3447, Apr. 2017.

[9] 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.

[10] S. Bi and Y. J. Zhang, "Computation rate maximization for wireless powered mobile-edge computing with binary computation offloading," *IEEE Trans. Wireless Commun.*, vol. 17, no. 6, pp. 4177–4190, Jun. 2018.

[11] 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.

[12] B. Cao, L. Zhang, Y. Li, D. Feng, and W. Cao, "Intelligent offloading in multi-access edge computing: A state-of-the-art review and framework," *IEEE Commun. Mag.*, vol. 57, no. 3, pp. 56–62, Mar. 2019.

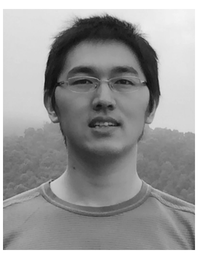
[13] K. Zhang, Y. Mao, S. Leng, S. Maharjan, and Y. Zhang, "Optimal delay constrained offloading for vehicular edge computing networks," in *Proc. ICC*, Ma 2017, pp. 1–6.

[14] Y. Liu, S. Wang, J. Huang, and F. Yang, "A computation offloading algorithm based on game theory for vehicular edge networks," in *Proc. ICC*, May 2018, pp. 1–6.

[15] Y. Dai, D. Xu, S. Maharjan, and Y. Zhang, "Joint Load Balancing and Offloading in Vehicular Edge Computing and Networks," *IEEE Internet Things J.*, to be published.

[16] Y. He, N. Zhao, and H. Yin, "Integrated networking, caching, and computing for connected vehicles: A deep reinforcement learning approach," *IEEE Trans. Veh. Technol.*, vol. 67, no. 1, pp. 44–55, Jan. 2018.
[17] L. T. Tan and R. Q. Hu, "Mobility-aware edge caching and computing in vehicle networks: A deep reinforcement learning," *IEEE Trans. Veh. Technol.*, vol. 67, no. 11, pp. 10190–10203, Nov. 2018.
[18] L. T. Tan, R. Q. Hu, and L. Hanzo, "Twin-timescale artificial intelligence aided mobility-aware edge caching and computing in vehicular networks," *IEEE Trans. Veh. Technol.*, vol. 68, no. 4, pp. 3086–3099, Apr. 2019.
[19] K. Zhang, S. Leng, X. Peng, L. Pan, S. Maharjan, and Y. Zhang, "Artificial intelligence inspired transmission scheduling in cognitive vehicular communications and networks," *IEEE Internet Things J.*, to be published.
[20] C. Zheng, D. Feng, S. Zhang, X.-G. Xia, G. Qian, and G. Y. Li, "Energy efficient v2x-enabled communications in cellular networks," *IEEE Trans. Veh. Technol.*, vol. 68, no. 1, pp. 554–564, Jan. 2019.
[21] S. Melendez and M. P. McGarry, "Computation offloading decisions for reducing completion time," in *Proc. IEEE CCNC*, Las Vegas, NV, USA, Jan. 2017, pp. 160–164.
[22] J. Ren, G. Yu, Y. Cai, and Y. He, "Latency optimization for resource allocation in mobile-edge computation offloading," *IEEE Trans. Wireless Commun.*, vol. 17, no. 8, pp. 5506–5519, Aug. 2018.
[23] C. You, K. Huang, and H. Chae, "Energy efficient mobile cloud computing powered by wireless energy transfer," *IEEE J. Sel. Areas Commun.*, vol. 34, no. 5, pp. 1757–1771, May 2016.
[24] S. Boyd and L. Vandenberghe,*Convex Optimization*. Cambridge, U.K.: Cambridge Univ. Press, 2004.
[25] L. Liang, G. Y. Li, and W. Xu, "Resource allocation for D2D-enabled vehicular communications," *IEEE Trans. Commun.*, vol. 65, no. 7, pp. 3186–3197, Jul. 2017.

JUN WANG received the M.S. degree in physical electronics from Huazhong Normal University, Wuhan, China, in 2008, and the Ph.D. degree in information and communication engineering from the Huazhong University of Science and Technology (HUST), Wuhan, in 2017. He was a Teaching Assistant with Huanggang Normal University, Huanggang, China, from 2008 to 2011, and an Assistant Professor with China Three Gorges University, Yichang, China, from 2017 to 2018, respectively. He is currently a Postdoctoral Research Fellow with Shenzhen University, Shenzhen, China. His current research interests include 5G, mobile edge computing, and the Internet of Things.

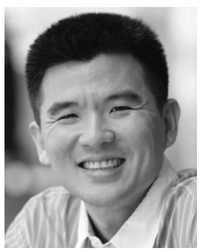


DAQUAN FENG received the Ph.D. degree in information engineering from the National Key Laboratory of Science and Technology on Communications, University of Electronic Science and Technology of China, Chengdu, China, in 2015. He had been a Visiting Student with the School of Electrical and Computer Engineering, Georgia Institute of Technology, USA, from 2011 to 2014. After graduation, he was a Research Staff with the State Radio Monitoring Center, Beijing, China, and then a Postdoctoral Research Fellow with the Singapore University of Technology and Design. He is currently an Assistant Professor with the Guangdong Key Laboratory of Intelligent Information Processing, College of Information Engineering, Shenzhen University, Shenzhen. His research interests include URLLC communications, LTE-U, and the massive IoT networks.

SHENGLI ZHANG received the B.Eng. degree in electronic engineering and the M.Eng. degree in communication and information engineering from the University of Science and Technology of China (USTC), Hefei, China, in 2002 and 2005, respectively, and the Ph.D. degree from the Department of Information Engineering, The Chinese University of Hong Kong (CUHK), in 2008.

After receiving the Ph.D. degree, he joined the Communication Engineering Department, Shenzhen University, where he is currently a Full Professor. From 2014 to 2015, he was a Visiting Associate Professor with Stanford University. He is the Pioneer of physical-layer network coding (PNC). He has published more than 20 IEEE top journal papers and ACM top conference papers, including IEEE JSAC, IEEE TWC, IEEE TMC, IEEE TCOMM, and ACM Mobicom. His research interests include physical layer network coding, interference cancellation, and cooperative wireless networks. He served as an Editor for IEEE TVT, IEEE WCL, and *IET Communications*. He has also served as a TPC Member in several IEEE conferences, including IEEE Globecom 2016, Globecom 2014, ICC 2015, ICC 2014, WCNC 2012, and WCNC 2014.

JIANHUA TANG (S'11–M'15) received the B.E. degree in communication engineering from Northeastern University, China, in 2010, and the Ph.D. degree in electrical and electronic engineering from Nanyang Technological University, Singapore, in 2015. He was a Postdoctoral Research Fellow with the Singapore University of Technology and Design, from 2015 to 2016. He is currently a Research Assistant Professor with the Department of Electrical and Computer Engineering, Seoul National University. His research interests include cloud computing, cloud radio access networks, and network slicing.



TONY Q. S. QUEK (S'98–M'08–SM'12–F'18) received the B.E. and M.E. degrees in electrical and electronics engineering from the Tokyo Institute of Technology, Tokyo, Japan, in 1998 and 2000, respectively, and the Ph.D. degree in electrical engineering and computer science from the Massachusetts Institute of Technology, Cambridge, MA, USA, in 2008.

Currently, he is a tenured Associate Professor with the Singapore University of Technology and Design (SUTD). He also serves as the Acting Head of ISTD Pillar and the Deputy Director of the SUTD-ZJU IDEA. He has coauthored the book *Small Cell Networks: Deployment, PHY Techniques, and Resource Allocation* (Cambridge University Press, 2013) and the book *Cloud Radio Access Networks: Principles, Technologies, and Applications* (Cambridge University Press, 2017). His current research interests include wireless communications and networking, the Internet-of-Things, network intelligence, wireless security, and big data processing.

Dr. Quek has been actively involved in organizing and chairing sessions, and has served as a member of the Technical Program Committee as well as the Symposium Chair in a number of international conferences. He is currently an elected member of the SPCOM Technical Committee, IEEE Signal Processing Society. He received the 2008 Philip Yeo Prize for Outstanding Achievement in Research, the IEEE Globecom 2010 Best Paper Award, the 2012 IEEE William R. Bennett Prize, the 2015 SUTD Outstanding Education Awards for Excellence in Research, the 2016 IEEE Signal Processing Society Young Author Best Paper Award, the 2017 CTTC Early Achievement Award, and the 2017 IEEE ComSoc AP Outstanding Paper Award, and he is the 2016–2018 Clarivate Analytics Highly Cited Researcher. He is a Distinguished Lecturer of the IEEE Communications Society. He was an Executive Editorial Committee Member of the IEEE Transactions on Wireless Communications, and an Editor of the IEEE Transactions on Communications and the IEEE Wireless Communications Letters.