Power Efficient User Cooperative Computation to Maximize Completed Tasks in MEC Networks

Yijin Pan*†, Cunhua Pan‡, Kezhi Wang§, Huiling Zhu†, and Jiangzhou Wang†
*National Mobile Communications Research Laboratory, Southeast University, China
†School of Engineering and Digital Arts, University of Kent, UK.
‡School of Electronic Engineering and Computer Science, Queen Mary University of London,UK.
§Department of Computer and Information Sciences, Northumbria University, UK.
Email: panyj@seu.edu.cn, c.pan@qmul.ac.uk,kezhi.wang@northumbria.ac.uk,{Y.Pan,H.Zhu,J.Z.Wang}@kent.ac.uk

***Abstract*—In this paper, the user cooperative task computation is explored by sharing the computing capability of the user equipments (UEs) so as to enhance the performance of mobile edge computing (MEC) networks. The number of completed tasks is maximized while minimizing the total power consumption of the UEs by jointly optimizing the user task offloading decision, the computational speed for the offloaded task and the transmit power for task offloading. An iterative algorithm based on the linear programming relaxation is proposed to solve the formulated mixed integer non-linear problem. The simulation results show that the proposed user cooperative computation scheme can achieve a higher completed tasks ratio than the non-cooperative scheme.**

I. Introduction

To allow the mobile user equipments (UEs) to operate computation-intensive and delay-sensitive applications such as real-time online gaming and virtual reality, mobile edge computing (MEC) has been proposed to bring the cloud-based IT servers closer to the end UEs. The MEC networks are usually constrained by the energy budget and delay requirements, and extensive efforts have been devoted to the design of efficient joint radio-and-computation resource managements [1], [2]. However, as the industry foresees that as many as 20.4 billion of potential IoT devices will be in service by 2020 [3], the limited computation capability of the MEC server has to be shared by intensive workloads [4]. This will lead to the server congestion issues so that a number of tasks may not be accomplished, resulting in the so-called infeasible tasks.

To cope with this issue, the cooperative task computation is proposed to exploit device-to-device (D2D) communications and seek for computation resources sharing among UEs. It has been shown that cooperative computation can help balance the heterogeneous distribution of computation resources and the uneven transmission conditions among different UEs [5]. For the fine-grained tasks, various partial computation offloading schemes have been proposed on the resource sharing and cooperative computing among UEs, such as [6], [7]. However, the tasks that are highly integrated or relatively simple cannot be partitioned and have to be executed as a whole [8]. In this case, the offloading decisions on which task should be offloaded and which UE should offload to are required in the multiple user cooperative MEC networks [9]. For instance, the binary task offloading decisions to minimize the total energy consumption were investigated in [10] and [11].

However, an important issue which is not addressed in the existing literature is simultaneously maximizing the number of completed tasks while minimizing the power consumption of mobile devices. In fact, the cooperative computation is based on the short-distance D2D transmissions, so that the task offloading delay can be significantly reduced. In this way, the previous infeasible tasks can become feasible by exploiting the computation resources in cooperative UEs. Furthermore, the power constraints of mobile devices are ignored in the existing user cooperative schemes for the sake of simplicity [9]–[11], which greatly restricts the practicability of the proposed approaches.

Against the above background, the computing capability of mobile devices is exploited in this paper to maximize the number of completed tasks while minimizing the total power consumption of UEs. By taking into account the maximum power constraints and the CPU frequency constraints, a mixed integer non-linear problem (MINLP) is formulated to jointly optimize the task offloading decision, transmit power for task offloading and the serving computational speed. To efficiently solve the non-convex problem, an equivalent tractable form is first presented by transforming the nonconvex constraints. Then, an iterative algorithm based on the relaxation of the integer constraint is proposed to efficiently solve the MINLP. Finally, the simulation results show that the proposed cooperative scheme can achieve a higher completed tasks ratio than the non-cooperative case.

II. System Model and Problem Formulation

Consider that there are N UEs, and each UE has a computation task to be executed. Define the UE set as $\mathcal{N} = \{1, 2, \cdots, N\}$, and the computational intensive task of UE i is denoted by $\mathcal{U}_i$. The network access point is connected to the MEC server, which enables UEs to offload their tasks for remote execution. Similar to [12], the task $\mathcal{U}_i$ of UE i is modeled as

$$\mathcal{U}_i = (F_i, D_i, T_i^{max}), \forall i \in \mathcal{N}, \quad (1)$$

where F_i is the required CPU cycles of $\mathcal{U}_i$ for computation, D_i denotes the data size of $\mathcal{U}_i$ for transmitting and T_i^{max} is the latency constraint of $\mathcal{U}_i$.

As shown in Fig. 1, as the computation capacity of edge cloud is limited so that it cannot afford to compute the tasks for all users at the same time. Therefore, to enhance the computation capacity of this MEC network, the computing capabilities of UEs are exploited by conducting D2D transmissions. For instance, in Fig. 1, UE i offloads its computational intensive task to UE j with higher computing capability via D2D link for cooperative task computation. At the same time, UE j's task can also be offloaded to its nearby device UE k.

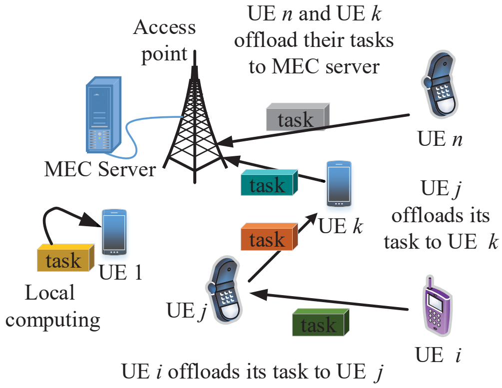

Fig. 1. An example of the cooperative task computation by UEs.

We use $\mathcal{M} = \{0, \mathcal{N}\}$ to represent the set of places that UEs can offload its task to, which includes the MEC server and all UEs. Define the indicator $a_{i,j}$, $i \in \mathcal{N}$, $j \in \mathcal{M}$ to represent the task decision, where

$$C1 : a_{i,j} = \{0, 1\}, \forall i \in \mathcal{N}, \forall j \in \mathcal{M}, \tag{2}$$

in which $a_{i,j} = 1$ denotes that UE i offloads its task to UE j ($j \neq 0$), or to the MEC server ($j = 0$). Note that $a_{i,i} = 1$ means that the task is executed by UE i itself. Also, each task can only be executed in one place, which can be expressed as

$$C2 : \sum_{j \in \mathcal{M}} a_{i,j} \leq 1, i \in \mathcal{N}. \tag{3}$$

It is worthy pointing out that some tasks may not be able to be completed anywhere in required time due to the lack of communication or computation resources.

If UE i decides to offload its task $\mathcal{U}_i$ to device j, the achievable data rate can be given as

$$r_{i,j} = B \log_2 \left(1 + \frac{p_{i,j}^T h_{i,j}}{\sigma^2}\right), \forall i \in \mathcal{N}, j \in \mathcal{M}, \tag{4}$$

where we assume that all the users have the same bandwidth and are allocated with the orthogonal frequency bands, $h_{i,j}$ is the channel gain from to UE i to UE j, σ^2 describes the white Gaussian noise power, B is the allocated bandwidth and $p_{i,j}^T$ is the transmit power. Then, the execution time of the task is

$$T_{i,j}^C = \frac{F_i}{f_{i,j}}, \forall i \in \mathcal{N}, \forall j \in \mathcal{M}, \tag{5}$$

where $f_{i,j}$ is the computation speed (CPU cycles per second) of the device j adopted to execute task $\mathcal{U}_i$. The time for task offloading transmission is

$$T_{i,j}^T = \frac{D_i}{r_{i,j}}, \quad \forall i \in \mathcal{N}, j \in \mathcal{M}, i \neq j. \tag{6}$$

The total time consumption should satisfy the latency constraint:

$$C3 : \sum_{j \neq i, j \in \mathcal{M}} a_{i,j} \left(\frac{D_i}{r_{i,j}} + \frac{F_i}{f_{i,j}}\right) + a_{i,i} \frac{F_i}{f_{i,i}} \leq T_{i,max}, i \in \mathcal{N}. \tag{7}$$

The computing power consumption for UE j to execute the task $\mathcal{U}_i$ at computational speed $f_{i,j}$ can be modeled as

$$p_{i,j}^C = \kappa_j (a_{i,j} f_{i,j})^{\nu_j}, \forall j \in \mathcal{N}, \tag{8}$$

where $\kappa_j \geq 0$ is the effective switched capacitance and $\nu_j \geq 1$ is a positive constant, which depends on the CPU chip structure.

For the MEC server and other devices, one has the computing constraints as

$$C4 : \sum_{i=1}^{N} a_{i,j} f_{i,j} \leq f_j^{max}, j \in \mathcal{M}. \tag{9}$$

Furthermore, for the mobile device, it has the limited power. Therefore, one has the following power constraint for UE i as

$$C5 : p_i = \sum_{k \in \mathcal{N}} a_{k,i} p_{k,i}^C + \sum_{j \neq i, j \in \mathcal{M}} a_{i,j} p_{i,j}^T \leq p_i^{max}, i \in \mathcal{N}. \tag{10}$$

Our target is to maximize the number of completed tasks while minimizing the total power consumption of the UEs by optimizing the task offloading decision $\{a_{i,j}\}$, transmit power for task offloading $\{p_{i,j}^T\}$ and the serving computational speed $\{f_{i,j}\}$. Then, this problem can be formulated as

$$\min_{\{a_{i,j}\}, \{f_{i,j}\}, \{p_{i,j}^T\}} \quad \sum_{i \in \mathcal{N}} p_i - \phi \sum_{i \in \mathcal{N}} \sum_{j \in \mathcal{M}} a_{i,j} \tag{11a}$$

$$\text{s.t.} \quad C1 - C5, \tag{11b}$$

where the positive constant ϕ is introduced to combine the different two objectives. It is readily to see that Problem (11) is a nonconvex MINLP, which is non-convex and NP-hard in general. In the following section, we propose an efficient method to solve Problem (11) by exploiting the relaxation of the integer constraint and the dual method.

III. Solution Analysis

As Problem (11) is not mathematically tractable due to the non-convex constraints $C3$, the constraint $C3$ is transformed into a tractable form in the following.

If the task $\mathcal{U}_i$ is offloaded to device j, i.e., $a_{i,j} = 1$, $i \neq j$. Then, by observing that the objective (11a) is an increasing function of $p_{i,j}^T$, it is inferred that the following equation holds for the optimal solution:

$$\frac{D_i}{r_{i,j}} + \frac{F_i}{f_{i,j}} = T_i^{max}, i \neq j, i \in \mathcal{N}, j \in \mathcal{M}. \tag{12}$$

After some algebraic transformation, (12) is transformed into

$$r_{i,j} = \frac{D_i f_{i,j}}{T_i^{max} f_{i,j} - F_i} \triangleq G_i(f_{i,j}). \tag{13}$$

In addition, the transmitting power $p_{i,j}^T$ can be represented as a function of $r_{i,j}$ according to (4), which is

$$p_{i,j}^T = \frac{\sigma^2}{h_{i,j}} \left(\exp\left(\frac{\ln(2)}{B} r_{i,j} \right) - 1 \right) \triangleq H_{i,j}(r_{i,j}). \tag{14}$$

The functions $G_i(x)$ and $H_{i,j}(x)$ are defined for simplicity.

Then, by denoting $f_i^{min} = \frac{F_i}{T_i^{max}}$ and $U_{i,j}(x) = H_{i,j}(G_i(x))$, an equivalent reformulation of Problem (11) is given by

$$\min_{\substack{\{a_{i,j}\}\\ \{f_{i,j}\}}} \sum_{i\in\mathcal{N}} \sum_{\substack{j\neq i\\ j\in\mathcal{M}}} a_{i,j} U_{i,j}(f_{i,j}) + \sum_{i\in\mathcal{N}} \kappa_i \sum_{k=1}^{N} (a_{k,i} f_{k,i})^{\nu_i} - \phi \sum_{i\in\mathcal{N}} \sum_{j\in\mathcal{M}} a_{i,j} \tag{15a}$$

$$\text{s.t.} \sum_{j\neq i, j\in\mathcal{M}} a_{i,j} U_{i,j}(f_{i,j}) + \kappa_i \sum_{k=1}^{N} (a_{k,i} f_{k,i})^{\nu_i} \leq p_i^{max}, i\in\mathcal{N}, \tag{15b}$$

$$a_{i,j} f_{i,j} \geq a_{i,j} f_i^{min}, i\in\mathcal{N}, j\in\mathcal{M}, \tag{15c}$$

$$C1, C2, C4.$$

The equivalence between Problem (15) and Problem (11) can be verified easily, which is omitted due to the limited space.

To efficiently solve Problem (15), we introduce the following conditions to reduce the feasible region. First of all, according to (15b), if UE j spends all its power executing the task $\mathcal{U}_i$, we have

$$f_{i,j} \leq \left(\frac{p_j^{max}}{\kappa_i} \right)^{\frac{1}{\nu_i}} \triangleq f_{i,j}^{max}, j\in\mathcal{N}. \tag{16}$$

If UE i uses all its power to transmit task $\mathcal{U}_i$ to UE j, we have

$$f_{i,j} \geq \frac{F_i}{T_i^{max} - \frac{D_i}{R_{i,j}^{max}}} \triangleq f_{i,j}^{min}, i\neq j, j\in\mathcal{M}, \tag{17}$$

where

$$R_{i,j}^{max} = B \log_2 \left(1 + \frac{p_i^{max} h_{i,j}}{\sigma^2} \right).$$

Then, we define $f_i^{min} = \frac{F_i}{T_i^{max}}$. Obviously, when $T_i^{max} > \frac{D_i}{R_{i,j}^{max}}$, we have $f_{i,j}^{min} > f_i^{min}$ for all $i \neq j$ due to the time consumption for offloading transmission. For notation simplicity, we define

$$f_{i,i}^{min} = f_i^{min}, f_{i,0}^{max} = f_0^{max} \tag{18}$$

$$f_{i,j}^{U} = \min\{f_{i,j}^{max}, f_j^{max}\}, f_{i,j}^{D} = \max\{f_{i,j}^{min}, f_i^{min}\}. \tag{19}$$

Then, we introduce the variable $x_{i,j} = a_{i,j} f_{i,j}$, and temporarily relax the integer constraints. Consequently, Problem (15) is transformed to

$$\min_{\substack{x_{i,j}\\ a_{i,j}}} \sum_{i\in\mathcal{N}} \sum_{\substack{j\neq i\\ j\in\mathcal{M}}} a_{i,j} U_{i,j}\left(\frac{x_{i,j}}{a_{i,j}}\right) + \sum_{i\in\mathcal{N}} \kappa_i \sum_{k=1}^{N} (x_{k,i})^{\nu_i} - \phi \sum_{i\in\mathcal{N}} \sum_{j\in\mathcal{M}} a_{i,j} \tag{20a}$$

$$\text{s.t.} \sum_{j\neq i, j\in\mathcal{M}} a_{i,j} U_{i,j}\left(\frac{x_{i,j}}{a_{i,j}}\right) + \kappa_i \sum_{k=1}^{N} (x_{k,i})^{\nu_i} \leq p_i^{max}, i\in\mathcal{N}, \tag{20b}$$

$$\sum_{i=1}^{N} x_{i,j} \leq f_j^{max}, j\in\mathcal{M}, \tag{20c}$$

$$\sum_{j\in\mathcal{M}} a_{i,j} \leq 1, i\in\mathcal{N}, \tag{20d}$$

$$a_{i,j} f_{i,j}^{D} \leq x_{i,j} \leq a_{i,j} f_{i,j}^{U}, \tag{20e}$$

$$0 \leq a_{i,j} \leq 1. \tag{20f}$$

According to (13)-(14), $H_{i,j}(x)$ is nondecreasing convex function with respect to (w.r.t) x, and $G_i(x)$ is convex. As a result, $U_{i,j}(x)$ is convex w.r.t x, and its perspective function $tU_{i,j}(x/t)$ is convex w.r.t (x,t). Consequently, it is concluded that Problem (20) is a convex problem, which can be optimally solved by the dual method. The Lagrangian of Problem (20) is given by

$$\begin{aligned}\mathcal{L} =& \sum_{i\in\mathcal{N}} (1+\mu_i) \sum_{j\neq i, j\in\mathcal{M}} a_{i,j} U_{i,j}\left(\frac{x_{i,j}}{a_{i,j}}\right) + \sum_{i\in\mathcal{N}} (1+\mu_i)\kappa_i \sum_{k=1}^{N} (x_{k,i})^{\nu_i} \\ & - \phi \sum_{i\in\mathcal{N}} \sum_{j\in\mathcal{M}} a_{i,j} + \sum_{j\in\mathcal{M}} v_j \left(\sum_{i=1}^{N} x_{i,j} - f_j^{max} \right) \\ & - \sum_{i\in\mathcal{N}} \mu_i p_i^{max} + \sum_{i\in\mathcal{N}} s_i \left(\sum_{j\in\mathcal{M}} a_{i,j} - 1 \right),\end{aligned} \tag{21}$$

where μ_i,v_j and s_i are the non-negative dual variables associated with the constraints (20b), (20c), and (20d), respectively.

Then, taking the derivatives of $\mathcal{L}$ w.r.t $x_{i,j}$ and $a_{i,j}$ respectively, we have

$$\frac{\partial\mathcal{L}}{\partial x_{i,0}} = (1+\mu_i) U'_{i,0}\left(\frac{x_{i,0}}{a_{i,0}}\right) + v_0, i\in\mathcal{N}, \tag{22}$$

$$\begin{aligned}\frac{\partial\mathcal{L}}{\partial x_{i,j}} =& (1+\mu_i) U'_{i,j}\left(\frac{x_{i,j}}{a_{i,j}}\right) + v_j \\ & + (1+\mu_j)\kappa_j \nu_i (x_{i,j})^{\nu_j - 1}, \forall i\neq j, i,j\in\mathcal{N},\end{aligned} \tag{23}$$

$$\frac{\partial\mathcal{L}}{\partial x_{i,i}} = (1+\mu_i)\kappa_i \nu_i (x_{i,i})^{\nu_i - 1} + v_i, i\in\mathcal{N}, \tag{24}$$

$$\begin{aligned}\frac{\partial\mathcal{L}}{\partial a_{i,j}} =& (1+\mu_i) \left(U_{i,j}\left(\frac{x_{i,j}}{a_{i,j}}\right) - \frac{x_{i,j}}{a_{i,j}} U'_{i,j}\left(\frac{x_{i,j}}{a_{i,j}}\right) \right) \\ & - \phi + s_i, \forall i\neq j, i, \in\mathcal{N}, j\in\mathcal{M},\end{aligned} \tag{25}$$

$$\frac{\partial\mathcal{L}}{\partial a_{i,i}} = -\phi + s_i, i\in\mathcal{N}, \tag{26}$$

where $U'_{i,j}(x)$ represents the first-order derivative of $U_{i,j}(x)$ w.r.t x. In addition, according to (13) and (14), it is easy to infer that $U'_{i,j}(x) < 0$ and the second order derivative $U''_{i,j}(x) > 0$ for $x \in \left[f_i^{min}, +\infty\right)$. In addition, it is inferred that there is only one solution in $x \in \left[f_i^{min}, +\infty\right)$ for the equation $\frac{\partial \mathcal{L}}{\partial x_{i,j}} = 0, i \neq j$.

For simplicity, we denote the solution to the equation $\frac{\partial \mathcal{L}}{\partial x_{i,j}} = 0$ in the interval $x \in \left[f_i^{min}, +\infty\right)$ as $\Gamma_{i,j}$, and denote the optimal solution to Problem (20) as $(a_{i,j}^*, x_{i,j}^*)$. Obviously, if $x_{i,j}^* = 0$, then $a_{i,j}^* = 0$, which is due to the constraints that $x_{i,j} \in [a_{i,j} f_{i,j}^D, a_{i,j} f_{i,j}^U]$. In the following analysis, we first consider the case that $x_{i,j}^* = 0$, and then consider the case that $x_{i,j}^* \neq 0$.

First, to ensure that the solution is feasible, according to (16) and (17), the set $\mathcal{J}_{i,j}$ is defined as

$$\mathcal{J}_{i,j} = \left\{ j | f_{i,j}^D \geq f_{i,j}^U, T_i^{max} \leq \frac{D_i}{R_{i,j}^{max}}, j \in \mathcal{M} \right\}. \quad (27)$$

Then, it is easy to infer that

$$x_{i,k}^* = 0, a_{i,k}^* = 0, \forall k \in \mathcal{J}_{i,k}. \quad (28)$$

In addition, according to the Lagrangian function [13]–[15], if the following condition holds, then $x_{i,j}{}^* = 0$, $a_{i,j}^* = 0$, $\forall i \in \mathcal{N}, j \in \mathcal{M}$:

$$\mathcal{L}|_{x_{i,j}=0, a_{i,j}=0} < \min \left\{ \mathcal{L}|_{x_{i,j}=f_{i,j}^U, a_{i,j}=1}, \mathcal{L}|_{x_{i,j}=f_{i,j}^D, a_{i,j}=1}, \mathcal{L}|_{x_{i,j}=\Gamma_{i,j}^*, a_{i,j}=1} \right\}, \quad (29)$$

where $\Gamma_{i,j}^* = [\Gamma_{i,j}]_{f_{i,j}^D}^{f_{i,j}^U}$. The term $y = [x]_a^b$ means that if $x \leq a$, then $y = a$, if $x \geq b$, then $y = b$. Otherwise, $y = x$. Then, we define $\mathcal{K}_{i,j}$ as the set of all the UEs that satisfy the condition (29).

Then, we consider the case that $x_{i,i}^* \neq 0$. In this case, according to (24), as the dual variables are non-negative, we have

$$\frac{\partial \mathcal{L}}{\partial x_{i,i}} > 0, \forall i = j, i \in \mathcal{N}. \quad (30)$$

Then, it is inferred that if the task is executed locally, the device should compute in the least computation speed that can satisfy the delay constraint, i.e.,

$$x_{i,i}{}^* = a_{i,i}^* f_{i,i}^D.$$

Furthermore, if $x_{i,j}^* \neq 0$ and $a_{i,j}^* \neq 0$, $\forall i \neq j$, according to the Karush-Kuhn-Tucker (KKT) conditions [16], we can conclude the following conditions:

$$\frac{\partial \mathcal{L}}{\partial a_{i,j}} \begin{cases} = 0, & \text{if } a_{i,j}{}^* \in (0,1), \\ < 0, & \text{if } a_{i,j}{}^* = 1. \end{cases} \quad (31)$$

$$\begin{cases} \begin{array}{l} \frac{\partial \mathcal{L}}{\partial x_{i,j}}(x) \mid_{x \in \left[f_{i,j}^D, f_{i,j}^U\right]} > 0, \text{ and} \\ \mathcal{L}|_{x_{i,j}=0, a_{i,j}=0} \geq \mathcal{L}|_{x_{i,j}=f_{i,j}^D, a_{i,j}=1} \end{array}, & \text{if } x_{i,j}^* = f_{i,j}^D; \\ \begin{array}{l} \frac{\partial \mathcal{L}}{\partial x_{i,j}}(x) \mid_{x \in \left[f_{i,j}^D, f_{i,j}^U\right]} = 0, \text{ and} \\ \mathcal{L}|_{x_{i,j}=0, a_{i,j}=0} \geq \mathcal{L}|_{x_{i,j}=\Gamma_{i,j}^*, a_{i,j}=1}, \end{array}, & \text{if } x_{i,j}{}^* \in (f_{i,j}^D, f_{i,j}^U); \\ \begin{array}{l} \frac{\partial \mathcal{L}}{\partial x_{i,j}}(x) \mid_{x \in \left[f_{i,j}^D, f_{i,j}^U\right]} < 0, \text{ and} \\ \mathcal{L}|_{x_{i,j}=0, a_{i,j}=0} \geq \mathcal{L}|_{x_{i,j}=f_{i,j}^U, a_{i,j}=1} \end{array}, & \text{if } x_{i,j}{}^* = f_{i,j}^U. \end{cases} \quad (32)$$

Then we have

$$x_{i,j}^* = a_{i,j}^* \Gamma_{i,j}^*, \forall i \neq j, i \in \mathcal{N}, j \in \mathcal{M}. \quad (33)$$

To determine the task decision $a_{i,j}$, we define $I_{i,j}$ as

$$I_{i,j} = (1 + \mu_i) \left(U_{i,j}\left(\Gamma_{i,j}^*\right) - \Gamma_{i,j}^* U'_{i,j}\left(\Gamma_{i,j}^*\right) \right), \quad \forall i \neq j, i \in \mathcal{N}, j \in \tilde{\mathcal{M}}_i, \quad (34)$$

where $\tilde{\mathcal{M}}_i = \mathcal{M} - \{\mathcal{K}_{i,j} \cup \mathcal{J}_{i,j}\}$.

Note that $U'_{i,j}(x) \leq 0$ for all $x \in [f_i^{min}, +\infty]$. Consequently, according to (25) and (26), if $x_{i,i} \neq 0$ and $x_{i,j} \neq 0$, the following inequality always holds

$$\frac{\partial \mathcal{L}}{\partial a_{i,i}} < \frac{\partial \mathcal{L}}{\partial a_{i,j}}. \quad (35)$$

Finally, according to constraint (20d), the task decision $a_{i,j}$ is concluded as

$$\begin{cases} \text{if } \tilde{\mathcal{M}}_i = \varnothing, & a_{i,j}^* = 0, \forall j \in \mathcal{M}, \\ \text{if } i \in \tilde{\mathcal{M}}_i, & a_{i,i}^* = 1, a_{i,j}{}^* = 0, j \neq i, j \in \mathcal{M}, \\ \text{else}, & a_{i,k}^* = 1, a_{i,j}{}^* = 0, j \neq k, j \in \mathcal{M}. \end{cases} \quad (36)$$

where $k = \arg \min_{j \in \tilde{\mathcal{M}}_i} I_{i,j}$.

The transmit power can be readily obtained according to (14). Note that the value of the dual variables μ_i and v_j can be determined by the sub-gradient method. The updating of μ_i, and v_j in the $(t+1)$-th iteration are

$$\mu_i^{(t+1)} = \left[\mu_i^{(t)} + \theta_i^{(t)} \left(\sum_{j \neq i, j \in \mathcal{M}} a_{i,j}^{(t)} U_{i,j}\left(\Gamma_{i,j}^{*(t)}\right) + \kappa_i \sum_{k=1}^{N} a_{k,i}^{(t)} \left(\Gamma_{k,i}^{*(t)}\right)^{\nu_i} - p_i^{max} \right) \right]^+, i \in \mathcal{N}, \quad (37)$$

$$v_j^{(t+1)} = \left[v_j^{(t)} + \zeta_j^{(t)} \left(\sum_{i=1}^{N} x_{i,j}^{(t)} - f_j^{max} \right) \right]^+, j \in \mathcal{M}, \quad (38)$$

where $[a]^+ = \max\{0, a\}$, and $\theta_i^{(t)}$ and $\zeta_j^{(t)}$ are the positive step sizes in the t-th iteration. According to [17, Proposition 6.3.1], the sub-gradient method converges to the optimal solution to Problem (20) for sufficiently small step sizes. Overall, the above analysis is summarized in Algorithm 1.

IV. Simulation Results

In this section, the simulation results are presented to show the performance gains achieved by the proposed cooperative MEC offloading scheme. Consider a 500 m × 500 m square cell with the BS in the center, and UEs are uniformly distributed. For the computation task, the size of each task is uniformly generated in the range $[0.1, 1.1]$ Mbits, the required computation frequency is uniformly distributed in [0.006G,14G] cycles/second, the maximum task execution time is uniformly drawn from $[40, 50]$ ms and the maximum CPU frequency of each UE is uniformly distributed in the range [0.002G,10G] cycles/second. The other simulation parameters are set to $B =$ 2 MHz, the noise power density is -174 dBm/Hz,$\kappa_i = 10^{-24}$, $\epsilon = 10^{-3}$ and $\nu_i = 3$. All the results are averaged over

Algorithm 1 Linear Programming Relaxation Based Iterative (LPRBI) Algorithm

Initialize ${x_{i,j}}^{(0)} = 0, {a_{i,j}}^{(0)} = 0, \forall i \in \mathcal{N}, j \in \mathcal{M}$ and the precision ϵ.
Initialize $\mu_i^{(0)}$, $v_j^{(0)}$, $\theta_i^{(0)}$, $\zeta_j^{(0)}$, $\forall i \in \mathcal{N}, j \in \mathcal{M}$.
repeat
  for $i \in \mathcal{N}$,$j \in \mathcal{M}$ **do**
    Calculate ${x_{i,j}}^{(t)}$ and ${a_{i,j}}^{(t)}$ according to (33) and (36), respectively;
  end for
  Update $\mu_i^{(t)}$, $v_j^{(t)}$ according to (37) and (38), respectively;
  Update the objective $\mathcal{O}^{(t)}$ according to (20a) ;
until $|\mathcal{O}^{(t)} - \mathcal{O}^{(t-1)}| < \epsilon$
Calculate $f_{i,j}$, p_{tot} =$\sum_{i \in \mathcal{N}} p_i$, and N_a =$\sum_{i \in \mathcal{N}} \sum_{j \in \mathcal{M}} a_{i,j}^{(t)}$;
Output: $\{a_{i,j}^{(t)}\}$, $\{f_{i,j}\}$, p_{tot}, N_a.

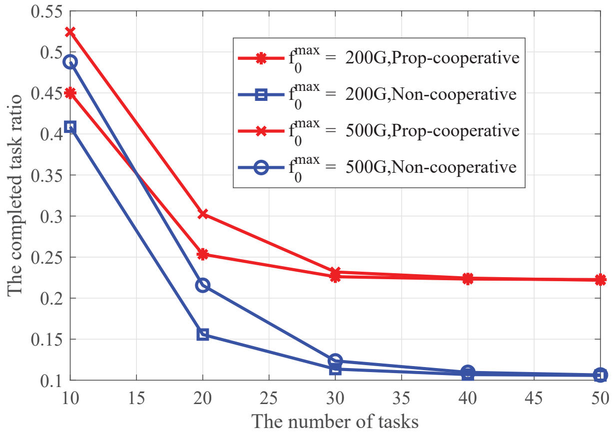

Fig. 3. The performances of the completed task ratio versus the task number.

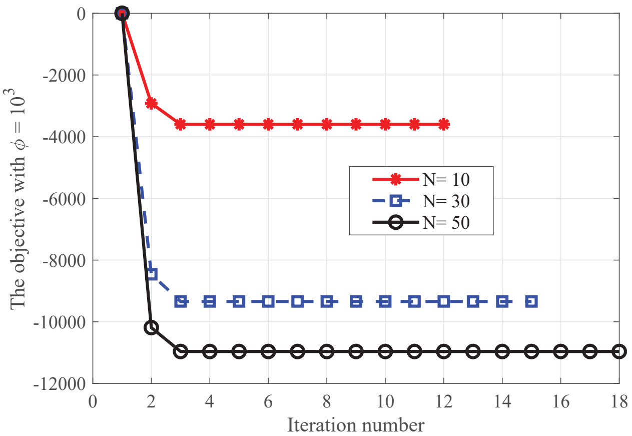

Fig. 2. The convergence performance of the proposed LPRBI algorithm.

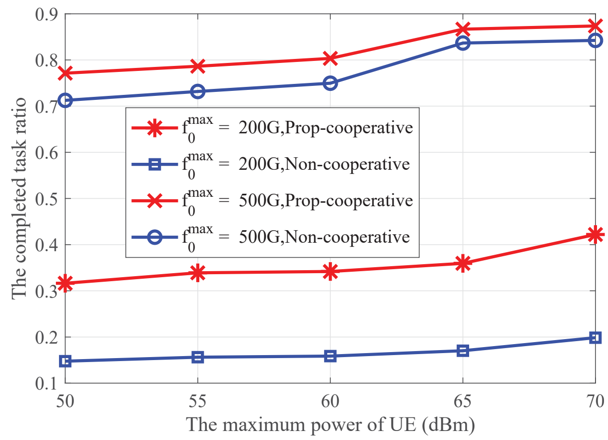

Fig. 4. The completed task ratio versus the maximum power of UEs.

1000 random realizations of users' locations, tasks, and fading channels [18], [19]. For comparison, we adopt the computation scheme that no cooperation is conducted between UEs, which is labeled as "Non-cooperative". The proposed cooperative computation scheme is labeled as "Prop-cooperative".

Fig. 2 shows the convergence performance of the proposed LPRBI algorithm. In Fig. 2, the maximum CPU frequency of MEC server is set to $f_0^{max} = 10^{12}$ cycles/second, and the maximum transmit power of each UE is set to $p_i^{max} = 60$dBm. As expected, it is seen from the figure that the objective value monotonically decreases during the initial iterations and then converges within 18 iterations for all considered cases. Consequently, the effectiveness of proposed LPRBI algorithm can be verified.

Fig. 3 compares the completed task ratio achieved by "Prop-cooperative" scheme and the "Non-cooperative" scheme versus the total number of tasks. In Fig. 3, $\phi = 10^8$, and other parameters are the same as those of Fig. 2. It is shown that the proposed "Prop-cooperative" scheme outperforms the "Non-cooperative" scheme in terms of the completed task ratios for all considered cases. When the user number or task size increases, the completed task ratio decreases due to the limited computation and power resources. Moreover, the performance gaps among "Prop-cooperative" scheme and the "Non-cooperative" scheme first increase and then becomes fixed, meanwhile, the performance gap between the different settings in each scheme gradually shrinks. The reason is that the critical factor for limiting the task completion rate, in this case, is the maximum power limit of the mobile UEs.

Fig. 4 illustrates the completed task ratios versus the maximum power of the mobile UEs. It is observed that the proposed "Prop-cooperative" scheme can always achieve a higher task completed ratio than the "Non-cooperative" scheme. For all the considered cases, more tasks can be completed as the maximum power limit of UEs increases. In the case of $f_0^{max} = 200$G, the task ratio is mainly limited by the computation capacity of the MEC server, so that the growth of the completed task ratio is slow. Meanwhile, in the case of $f_0^{max} = 500$G, the task ratio is mainly limited by the power limit of UEs, so that the gap between the proposed "Prop-cooperative" scheme and the "Non-cooperative" scheme decreases.

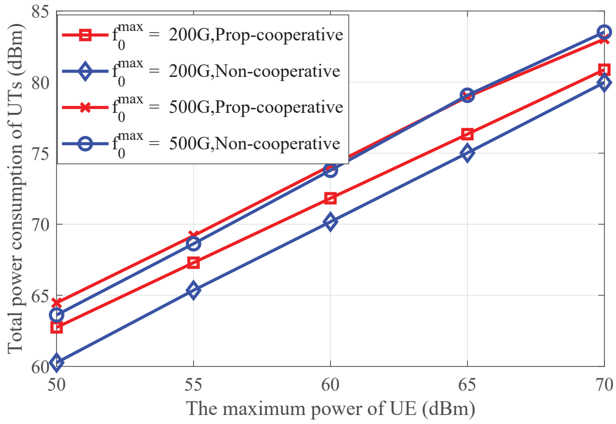

Fig. 5. Total power consumption of UEs versus UEs' maximum power limit.

Fig. 5 shows the total power consumption of UEs versus the maximum power of the mobile UEs. In the case of $f_0^{max} = 500$G, it is shown that the power consumed by the proposed "Prop-cooperative" scheme is larger than that of the "Non-cooperative" scheme. This is due to the fact that more UEs can get their task being accomplished. In the case of $f_0^{max} = 200$G, the power consumed by the proposed "Prop-cooperative" scheme is comparable with that of the "Non-cooperative" scheme, while the proposed "Prop-cooperative" scheme can achieve a higher completed task ratio according to Fig. 4.

V. Conclusions

A power efficient user cooperative task computation scheme has been investigated in this paper to maximize the number of accomplished tasks while minimizing the total power consumption of UEs. It has been shown that the ratio of completed tasks is greatly affected by the computation resources of the network and the maximum power of UEs. As the computation resource of the potential cooperative UEs is exploited and the task transmission distance is reduced by D2D transmission, the proposed cooperative computation scheme allows more tasks to be accomplished than the traditional non-cooperative computation scheme.

Acknowledgment

This work was supported in part by the National Science and Technology Major Project under Grant No. 2016ZX03001016-003, the National Natural Science Foundation of China under Grants No. 61871128, No. 61521061, No. 61571125, the Central University Basic Research fund under Grant No. 2242019K40202, the UK Royal Society Newton International Fellowship under Grant NIF\ R1\180777.

References

[1] 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.

[2] A. Al-Shuwaili and O. Simeone, "Energy-efficient resource allocation for mobile edge computing-based augmented reality applications," *IEEE Wireless Commun. Lett*, vol. 6, no. 3, pp. 398–401, June 2017.

[3] R. van der Meulen, "Gartner foresees releases," https://www.gartner.com/en/newsroom/press-releases/201702-07/, feb 7, 2017.

[4] M. Liu and Y. Liu, "Price-based distributed offloading for mobile-edge computing with computation capacity constraints," *IEEE Wireless Commun. Lett*, vol. 7, no. 3, pp. 420–423, June 2018.

[5] X. Cao, F. Wang, J. Xu, R. Zhang, and S. Cui, "Joint computation and communication cooperation for energy-efficient mobile edge computing," *IEEE Internet of Things Journal (Early Access)*, pp. 1–1, 2019.

[6] Z. Sheng, C. Mahapatra, V. C. Leung, M. Chen, and P. K. Sahu, "Energy efficient cooperative computing in mobile wireless sensor networks," *IEEE Transactions on Cloud Computing*, vol. 6, no. 1, pp. 114–126, 2018.

[7] C. You and K. Huang, "Mobile cooperative computing: Energy-efficient peer-to-peer computation offloading," *arXiv preprint arXiv:1704.04595*, 2017.

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

[9] M. Wang, H. Jin, C. Zhao, and D. Liang, "Delay optimization of computation offloading in multi-hop ad hoc networks," in *2017 IEEE International Conference on Communications Workshops (ICC Workshops)*. IEEE, 2017, pp. 314–319.

[10] J. Song, Y. Cui, M. Li, J. Qiu, and R. Buyya, "Energy-traffic tradeoff cooperative offloading for mobile cloud computing," in *2014 IEEE 22nd International Symposium of Quality of Service (IWQoS)*. IEEE, 2014, pp. 284–289.

[11] X. Chen, L. Pu, L. Gao, W. Wu, and D. Wu, "Exploiting massive D2D collaboration for energy-efficient mobile edge computing," *IEEE Wireless Communications*, vol. 24, no. 4, pp. 64–71, 2017.

[12] K. Wang, K. Yang, and C. Magurawalage, "Joint energy minimization and resource allocation in C-RAN with mobile cloud," *IEEE Transactions on Cloud Computing*, vol. PP, no. 99, pp. 1–1, 2016.

[13] H. Zhu and J. Wang, "Chunk-based resource allocation in ofdma systems - part i: chunk allocation," *IEEE Transactions on Communications*, vol. 57, no. 9, pp. 2734–2744, Sep. 2009.

[14] ——, "Chunk-based resource allocation in ofdma systems—part ii: Joint chunk, power and bit allocation," *IEEE Transactions on Communications*, vol. 60, no. 2, pp. 499–509, February 2012.

[15] H. Zhu, "Radio resource allocation for ofdma systems in high speed environments," *IEEE Journal on Selected Areas in Communications*, vol. 30, no. 4, pp. 748–759, May 2012.

[16] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge university press, 2004.

[17] D. P. Bertsekas, *Nonlinear programming*. Athena scientific Belmont, 1999.

[18] H. Zhu, "Performance comparison between distributed antenna and microcellular systems," *IEEE Journal on Selected Areas in Communications*, vol. 29, no. 6, pp. 1151–1163, June 2011.

[19] J. Wang, H. Zhu, and N. J. Gomes, "Distributed antenna systems for mobile communications in high speed trains," *IEEE Journal on Selected Areas in Communications*, vol. 30, no. 4, pp. 675–683, May 2012.