Joint Computation Offloading, Resource Allocation and Content Caching in Cellular Networks with Mobile Edge Computing

Chenmeng Wang*, Chengchao Liang†, F. Richard Yu†, Qianbin Chen*, and Lun Tang*
*Chongqing Key Lab of Mobile Communications Technology,
Chongqing University of Posts and Telecomm. (CQUPT), Chongqing, 400065, P. R. China
†Dept. of Systems and Computer Eng., Carleton University, Ottawa, ON, Canada

***Abstract*—Mobile edge computing (MEC) has risen as a promising technology to augment computational capabilities of mobile devices. Meanwhile, in-network caching has become a natural trend of the solution of handling exponentially increasing Internet traffic. The important issues in these two networking paradigms are computation offloading and content caching strategies, respectively. In order to jointly tackle these issues, we formulate an optimization problem in wireless cellular networks with mobile edge computing, taking into consideration computation offloading decision, physical spectrum resource allocation, MEC computation resource allocation, and content caching strategy. Furthermore, we transform the original problem into a convex problem and then decompose it in order to solve it in a distributed and efficient way. Finally, with recent advances in distributed convex optimization, we develop an alternating direction method of multipliers (ADMM) based algorithm to solve the optimization problem. The effectiveness of the proposed scheme is demonstrated by simulation results with different system parameters.**

***Index Terms*—Mobile edge computing, small cell networks, computation offloading, resource allocation, in-network caching.**

I. Introduction

The increasing popularity of smart phones has led to an exponentially growing demand not only in *high data rate* but also in *high computational capability*.

In order to address the data rate issue, the heterogeneous network structure was recently proposed [1]. Nevertheless, severe inter-cell interference may be incurred due to spectrum reuse [2], [3], which will significantly deteriorate network performance. To address the spectrum allocation issue, the work in [4] proposed a graph colouring method to assign physical resource blocks (PRBs) to UEs.

On the other hand, to address the computational capability issue, *mobile edge computing* (MEC), a technique similar to the *fog computing*, has attracted great interest in wireless cellular networks recently [5]. MEC enables the mobile user's equipment (UEs) to perform *computation offloading* to the MEC server via wireless cellular networks. Then each UE is associated with a clone in the MEC server, which executes the computational tasks on behalf of that UE. A number of previous works have discussed the computation offloading problem [6], [7].

In addition, the server in MEC system can realize an in-network caching function [5], similar to the function provided by information-centric networking (ICN) [8], which is able to reduce replicate information transmissions. A number of works have been dedicated to caching strategies [9], [10].

The motivations behind this paper are based on the following observations.

- Computation offloading, resource allocation and content caching are all parts of the entire system, and they all contribute to the end-to-end user experience, which can hardly be guaranteed by the optimization of one single segment of the whole system [11].
- To prevent transmission interference and MEC server overload, some UEs should be selected to offload their computations, while others should execute their computations locally.
- Different amounts of spectrum and computation resources should be allocated to different UEs to fulfill different user demands.
- Different caching strategies should be applied upon different contents, in order to maximize the caching revenue.

The distinct features of this paper are as follows.

- We formulate the computation offloading, resource allocation, and content caching as an optimization problem.
- We transform the original non-convex problem into a convex problem and prove the convexity.
- We decompose the problem and apply alternating direction method of multipliers (ADMM) to solve the problem.
- Simulation results are presented.

The rest of this paper is organized as follows. The system model under consideration is described in Section II. The original optimization problem is formulated, transformed and decomposed in Section III. Section IV presents the progress of problem solving by ADMM. Simulation results are discussed in Section V. Finally, we conclude this study in Section VI.

II. System Model

A. Network Model

An environment of one macrocell and N small cells in the terminology of LTE standards is considered here. An MEC server is placed in the macro eNodeB (MeNB), and all the N small cell eNodeBs (SeNBs) are connected to the MeNB as well as the MEC server. The set of small cells is denoted by $\mathcal{N} = \{1, 2, ..., N\}$, and we use n to refer to the nth small

978-1-4673-8999-0/17/$31.00 ©2017 IEEE

cell (SeNB). It is assumed that SeNB n is associated with K_n mobile UEs. We let $\mathcal{K}_n = \{1, 2, ..., K_n\}$ denote the set of UEs associating with SeNB n, and k_n refers to the kth UE which associates with the nth SeNB.

We assume that each UE has a computation task to be completed. Each UE can offload the computation to the MEC server through the SeNB, or execute the task locally. UEs can request content from the Internet, then the Internet content will be transmitted through macro base station (MeNB) to UEs. Upon the first transmission, the MEC server can choose whether to store the content or not. For simplicity, user mobility and handover [12]–[14] are not considered. In this paper, we consider two logical roles in the network: *mobile network operator* (MNO) and *MEC system operator* (MSO). The MNOs possess and operate the radio resources and physical infrastructures of the wireless network, while the MSOs own the MEC servers, lease physical resources (like spectrum and backhaul) from MNO and provide mobile edge computing services to UEs. The MSO will charge the UEs for receiving mobile edge computing services.

B. Communication Model

We denote $a_{k_n} \in \{0, 1\}, \forall n, k$ as the computation offloading decision of UE k_n. Specifically, we have $a_{k_n} = 0$ if UE k_n was determined to compute its task locally on the mobile device. We have $a_{k_n} = 1$ if UE k_n was chosen to offload the computation to the MEC server via wireless access. So we have $\boldsymbol{a} = \{a_{k_n}\}_{k_n \in \mathcal{K}_n, n \in \mathcal{N}}$ as the offloading decision profile.

In this paper, we consider the case where spectrum used by small cells is overlaid, while spectrum within one small cell is orthogonally assigned to every UE. The whole available spectrum bandwidth is B Hz. The backhaul capacity between MeNB and MEC server is L bps, and the backhaul capacity of SeNB n is L_n bps. According to Shannon bound, the spectrum efficiency of UE k_n is given by,

$$e_{k_n} = \log_2 \left(1 + \frac{p_{k_n} G_{k_n,n}}{\sigma + \sum\limits_{m=1, m \neq n}^{N} \sum\limits_{i=1}^{K_m} p_{i_m} G_{i_m,n}} \right), \quad \forall n, k, \tag{1}$$

where p_{k_n} is the transmission power density of UE k_n, and $G_{k_n,n}$, $G_{i_m,n}$ stand for the channel gain between UE k_n and SeNB n, the channel gain between UE i_m and SeNB n, respectively. σ denotes the power spectrum density of additive white Gaussian noise.

We denote $s_{k_n} \in [0, 1], \forall n, k$ as the percentage of radio spectrum allocated to UE k_n by small cell n, thus $\sum_{k_n \in \mathcal{K}_n} s_{k_n} \leq 1, \forall n$. We have $\boldsymbol{s} = \{s_{k_n}\}_{k_n \in \mathcal{K}_n, n \in \mathcal{N}}$ as the radio spectrum allocation profile. Then the expected instantaneous data rate of UE k_n, R_{k_n} is calculated as

$$R_{k_n}(\boldsymbol{a}, \boldsymbol{s}) = a_{k_n} s_{k_n} B e_{k_n}, \quad \forall n, k. \tag{2}$$

The data rate cannot exceed the backhaul capacity of SeNB n, thus $\sum_{k_n \in \mathcal{K}_n} R_{k_n} \leq L_n, \forall n$ must hold. The total data rate of all the UEs cannot exceed the backhaul capacity of MeNB, thus $\sum_{n \in \mathcal{N}} \sum_{k_n \in \mathcal{K}_n} R_{k_n} \leq L$ must hold.

C. Computation Model

For the computation model, we consider that each UE k_n has a computation task $W_{k_n} \triangleq (Z_{k_n}, D_{k_n})$. Here Z_{k_n} stands for the size of input data, including program codes and input parameters, and D_{k_n} denotes the total number of CPU cycles required to accomplish the computation task W_{k_n}.

1) Local Computing: We denote $f_{k_n}^{(l)}$ as the computational capability (i.e., CPU cycles per second) of UE k_n. The computation execution time $T_{k_n}^{(l)}$ of task W_{k_n} executed locally by UE k_n is expressed as

$$T_{k_n}^{(l)} = D_{k_n} / f_{k_n}^{(l)}. \tag{3}$$

2) MEC Server Computing: The time costs for transmitting the computation input data of size Z_{k_n} are calculated as

$$T_{k_n,off}^{(e)}(\boldsymbol{a}, \boldsymbol{s}) = Z_{k_n} / R_{k_n}(\boldsymbol{a}, \boldsymbol{s}). \tag{4}$$

Let $f_{k_n}^{(e)}$ denote the computational capability (i.e., CPU cycles per second) of the MEC server assigned to UE k_n. Then the execution time of the MEC server on task W_{k_n} is given as

$$T_{k_n,exe}^{(e)} = D_{k_n} / f_{k_n}^{(e)}. \tag{5}$$

Then the total execution time of the task of UE k_n is given by

$$T_{k_n}^{(e)}(\boldsymbol{a}, \boldsymbol{s}) = T_{k_n,off}^{(e)}(\boldsymbol{a}, \boldsymbol{s}) + T_{k_n,exe}^{(e)}. \tag{6}$$

D. Caching Model

We denote $h_{k_n} \in \{0, 1\}, \forall n, k$ as the caching strategy for UE k_n. Specifically, we have $h_{k_n} = 1$ if the MEC server decides to cache the content requested by UE k_n and $h_{k_n} = 0$ otherwise. So we have $\boldsymbol{h} = \{h_{k_n}\}_{k_n \in \mathcal{K}_n, n \in \mathcal{N}}$ as the caching decision profile.

The alleviated backhaul bandwidth between macro cell and the Internet is adopted as the caching reward. Thus, the reward of caching the content requested by UE k_n can be given as,

$$Caching\ reward = q_{k_n} \bar{R} h_{k_n}, \tag{7}$$

where $\bar{R}$ is the average single UE data rate in the system, and q_{k_n} is the request rate (by other UEs) of the content first requested by UE k_n. According to the statistics of [15], the request rate of a content can be calculated as $q(i) = 1/i^{\beta}$, where i stands for the i-th most popular content, and β is a constant whose typical value is 0.56 [16].

The sum size of all the cached content cannot exceed the total storage capability of the MEC server. In other words, $\sum_{n \in \mathcal{N}} \sum_{k_n \in \mathcal{K}_n} h_{k_n} o_{k_n} \leq Y$ must hold, where Y is the total storage capability of the MEC server, and o_{k_n} is the size of the content first requested by UE k_n. In this paper, it is assumed that $o_{k_n} \forall k, n$ is constant and we adopt $o_{k_n} = 1$.

E. Utility Function

In this paper, we set the maximization of the revenue of MSO as our goal. MSO rents spectrum and backhaul from MNO, and the unit price for leasing spectrum from small cell n is defined as δ_n per Hz, while the unit price of backhaul between small cell n and macro cell is defined as η_n per bps.

The MSO will charge UEs for transmitting computation input data to MEC server, and the unit price being charged is defined as θ_n per bps. So the net revenue of MSO for assigning radio resources to UE k_n is calculated as

$$\iota_{k_n} = s_{k_n}\Psi_{k_n} = s_{k_n}(\theta_n B e_{k_n} - \delta_n B - \eta_n B e_{k_n}). \quad (8)$$

We next calculate the revenue of MSO for allocating computation resource to UEs. First, we define $c_{k_n} \in [0,1], \forall n,k$ as the percentage of MEC server computation resource allocated to UE k_n, thus $\sum_{n\in\mathcal{N}}\sum_{k_n\in\mathcal{K}_n} c_{k_n} \leq 1$. We have $\boldsymbol{c} = \{c_{k_n}\}_{k_n\in\mathcal{K}_n, n\in\mathcal{N}}$ as the computation resource allocation profile. Without losing generality, the different sizes of computation tasks and different local computation capability of different UEs should be taken into account. So we define that the MSO will charge UE k_n only for the difference between MEC computation resource allocated to every unit computation task and the local computation resource assigned to every unit computation task, and the unit price is λ_n for small cell n. Then the net revenue of allocating computation resource to UE k_n is given as

$$\Omega_{k_n} = \lambda_n(\frac{c_{k_n}F}{D_{k_n}} - \frac{f_{k_n}^{(l)}}{D_{k_n}}), \quad (9)$$

where F stands for the total computation resource of MEC server. Note that the reciprocal of $c_{k_n}F/D_{k_n}$ is the time consumption for MEC server executing computation task D_{k_n}, and the reciprocal of $f_{k_n}^{(l)}/D_{k_n}$ is the time consumption for UE k_n locally executing task D_{k_n}. This implies that the amount of computation resource assigned to every unit computation task can reflect the time consumption of executing this task.

We next discuss the revenue of MSO for caching Internet content requested by UEs. We define the unit price of leasing the backhaul between the macro cell and Internet is ζ per bps, and the cost in the memory for caching one content is ϖ. If the content first requested by UE k_n was stored by the MEC server, according to (7), the alleviated backhaul bandwidth in the future should be $\zeta q_{k_n}\bar{R}$. And the memory cost for storing that content is ϖ. So the long term revenue of caching the Internet content first requested by UE k_n is calculated as,

$$\Lambda_{k_n} = \zeta q_{k_n}\bar{R} - \varpi. \quad (10)$$

Next we formulate the utility function of MSO as

$$\begin{aligned} U &= \sum_{n\in\mathcal{N}}\sum_{k_n\in\mathcal{K}_n} u\left(a_{k_n}\iota_{k_n} + a_{k_n}\Omega_{k_n}\right) + h_{k_n}\Lambda_{k_n} \\ &= \sum_{n\in\mathcal{N}}\sum_{k_n\in\mathcal{K}_n} a_{k_n} u\left(s_{k_n}\Psi_{k_n} + c_{k_n}\frac{\lambda_n F}{D_{k_n}} - \frac{\lambda_n f_{k_n}^{(l)}}{D_{k_n}}\right) \\ &\quad + h_{k_n}\Lambda_{k_n}, \end{aligned} \quad (11)$$

where $u(\cdot)$ is an utility function which is nondecreasing and convex. Since $h_{k_n}\Lambda_{k_n}$ is always non-negative due to problem optimality, it can be put outside of the function $u(\cdot)$. It is equivalent to take a_{k_n} outside of the function $u(\cdot)$. Here the logarithmic function is adopted as the utility function.

Define

$$U' = \sum_{n\in\mathcal{N}}\sum_{k_n\in\mathcal{K}_n} a_{k_n} u\left(s_{k_n}\Psi_{k_n} + c_{k_n}\frac{\lambda_n F}{D_{k_n}}\right) + h_{k_n}\Lambda_{k_n}. \quad (12)$$

Because $\lambda_n f_{k_n}^{(l)}/D_{k_n}$ is constant, when U' reaches the maximum value, U reaches the maximum as well, i.e., the MSO reaches the maximum income. Let $\lambda_n F/D_{k_n} = \Phi_{k_n}$, next we will use

$$U' = \sum_{n\in\mathcal{N}}\sum_{k_n\in\mathcal{K}_n} a_{k_n} u\left(s_{k_n}\Psi_{k_n} + c_{k_n}\Phi_{k_n}\right) + h_{k_n}\Lambda_{k_n} \quad (13)$$

as our objective function of the optimization problem.

III. Problem Formulation, Transformation and Decomposition

A. Problem Formulation

The problem is formulated as

$$\begin{aligned} \underset{\boldsymbol{a},\boldsymbol{s},\boldsymbol{c},\boldsymbol{h}}{\text{Maximize}} \quad & \sum_{n\in\mathcal{N}}\sum_{k_n\in\mathcal{K}_n} a_{k_n} u\left(s_{k_n}\Psi_{k_n} + c_{k_n}\Phi_{k_n}\right) + h_{k_n}\Lambda_{k_n} \\ s.t. \quad C1: \quad & \sum_{k_n\in\mathcal{K}_n} a_{k_n}s_{k_n} \leq 1, \forall n \\ C2: \quad & \sum_{k_n\in\mathcal{K}_n} a_{k_n}s_{k_n}Be_{k_n} \leq L_n, \forall n \\ C3: \quad & \sum_{m\in\mathcal{N}/\{n\}}\sum_{k_m\in\mathcal{K}_m} a_{k_m}p_{k_m}G_{k_m,n} \leq I_n, \forall n \\ C4: \quad & \sum_{n\in\mathcal{N}}\sum_{k_n\in\mathcal{K}_n} a_{k_n}c_{k_n} \leq 1 \\ C5: \quad & a_{k_n}(\frac{c_{k_n}F}{D_{k_n}} - \frac{f_{k_n}^{(l)}}{D_{k_n}}) \geq 0, \forall k,n \\ C6: \quad & \sum_{n\in\mathcal{N}}\sum_{k_n\in\mathcal{K}_n} h_{k_n} \leq Y. \end{aligned} \quad (14)$$

Constraints (14) $C3$ are proposed to ensure that the interference on SeNB n caused by all offloading UEs which are served by other SeNBs doesn't exceed a predefined threshold, I_n. Because we have removed $-\lambda_n f_{k_n}^{(l)}/D_{k_n}$ in (11), we need constraints (14) $C5$ to guarantee that the computation resource allocated to each offloading UE k_n is no less than that of itself.

B. Problem Transformation

Problem (14) is a mixed discrete and non-convex optimization problem, and such problems are usually considered as NP-hard problems [17]. Therefore, a transformation and simplification of the original problem are necessary.

1) Binary variable relaxation: We need to relax binary variables $\boldsymbol{a}$ and $\boldsymbol{h}$ into real value variables as $0 \leq a_{k_n} \leq 1$, $0 \leq h_{k_n} \leq 1$ [17].

2) Substitution of the product term: Next we will propose a proposition of the equivalent problem of (14) to make the problem solvable.

Proposition 1: If we define $\tilde{s}_{k_n} = s_{k_n}a_{k_n}$, $\tilde{c}_{k_n} = c_{k_n}a_{k_n}$, and $a_{k_n}u[(\tilde{s}_{k_n}\Psi_{k_n} + \tilde{c}_{k_n}\Phi_{k_n})/a_{k_n}] = 0$ when $a_{k_n} = 0$, then substitute them into problem (14), it is equivalent to the original problem.

Proof: If we substitute $\tilde{s}_{k_n} = s_{k_n} a_{k_n}$ and $\tilde{c}_{k_n} = c_{k_n} a_{k_n}$ into (14), the objective function becomes

$$\underset{\boldsymbol{a},\tilde{\boldsymbol{s}},\tilde{\boldsymbol{c}},\boldsymbol{h}}{\text{Maximize}} \quad \sum_{n\in\mathcal{N}} \sum_{k_n\in\mathcal{K}_n} a_{k_n} u\left(\frac{\tilde{s}_{k_n}\Psi_{k_n} + \tilde{c}_{k_n}\Phi_{k_n}}{a_{k_n}}\right) + h_{k_n}\Lambda_{k_n} \tag{15}$$

We can recover the original optimization problem (14) except the point when $a_{k_n} = 0$. Next we will discuss about this point. Suppose $a_{k_n} = 0$, then $s_{k_n} = 0$ and $c_{k_n} = 0$ will certainly hold because of the problem optimality. Apparently, if UE k_n will not offload computation task to MEC server, SeNB n will not allocate any spectrum resource to UE n, and MEC server will not assign any computation resource to it, either. Now it's a one-to-one mapping. □

C. Convexity

Proposition 2: If the problem is feasible, it is jointly convex with respect to all the optimization variables $\boldsymbol{a}$, $\tilde{\boldsymbol{s}}$, $\tilde{\boldsymbol{c}}$ and $\boldsymbol{h}$.

Proof: $f(t,x) = x\log(t/x), t \geq 0, x \geq 0$ is the well-known perspective function of $f(x) = \log x$. Next we will give a proof of the continuity of the perspective function $f(t,x) = x\log(t/x), t \geq 0, x \geq 0$ on the point $x = 0$. Let $s = t/x$,

$$f(t,0) = \lim_{x\to 0} x\log\frac{t}{x} = \lim_{s\to\infty}\frac{t}{s}\log s = t\lim_{s\to\infty}\frac{\log s}{s} = 0. \tag{16}$$

So we have $a_{k_n}\log[(\tilde{s}_{k_n}\Psi_{k_n} + \tilde{c}_{k_n}\Phi_{k_n})/a_{k_n}] = 0$ for $a_{k_n} = 0$. Since $(\tilde{s}_{k_n}\Psi_{k_n} + \tilde{c}_{k_n}\Phi_{k_n})$ are linear with respect to $\tilde{s}_{k_n}$ and $\tilde{c}_{k_n}$, $\log(\tilde{s}_{k_n}\Psi_{k_n} + \tilde{c}_{k_n}\Phi_{k_n})$ is a concave function. Then $a_{k_n}\log[(\tilde{s}_{k_n}\Psi_{k_n} + \tilde{c}_{k_n}\Phi_{k_n})/a_{k_n}]$ is concave due to the fact that it is the perspective function of $\log(\tilde{s}_{k_n}\Psi_{k_n} + \tilde{c}_{k_n}\Phi_{k_n})$. The perspective function of a concave function is concave [18]. Furthermore, $h_{k_n}\Lambda_{k_n}$ is linear, and all the constraints of the problem are linear, so the problem is a convex optimization problem. □

A lot of methods could be applied to solve a convex optimization problem. But as far as our problem is concerned, the size of the problem becomes appreciably large as the number of small cells grows. Therefore, it will be more efficient to employ a distributed algorithm which is running on each SeNB as well as the MEC server.

D. Problem Decomposition

In order to make the problem separable, we need to introduce the local copies of the global variables. For small cell n, we denote $\hat{\boldsymbol{a}}^n = \{\hat{a}^n_{k_j}\}_{k_j\in\mathcal{K}_j, j\in\mathcal{N}, n\in\mathcal{N}}$ *, $\hat{\boldsymbol{c}}^n = \{\hat{c}^n_{k_j}\}_{k_j\in\mathcal{K}_j, j\in\mathcal{N}, n\in\mathcal{N}}$ and $\hat{\boldsymbol{h}}^n = \{\hat{h}^n_{k_j}\}_{k_j\in\mathcal{K}_j, j\in\mathcal{N}, n\in\mathcal{N}}$ as the local copies of $\boldsymbol{a}$, $\tilde{\boldsymbol{c}}$ and $\boldsymbol{h}$, respectively. We have

$$\begin{cases} \hat{a}^n_{k_j} = a_{k_j}, & \forall n,k,j, \\ \hat{c}^n_{k_j} = \tilde{c}_{k_j}, & \forall n,k,j, \\ \hat{h}^n_{k_j} = h_{k_j}, & \forall n,k,j. \end{cases} \tag{17}$$

Next we will use

$$U'' = \sum_{n\in\mathcal{N}}\sum_{k_n\in\mathcal{K}_n} \hat{a}^n_{k_n} u\left(\frac{\tilde{s}_{k_n}\Psi_{k_n} + \hat{c}^n_{k_n}\Phi_{k_n}}{\hat{a}^n_{k_n}}\right) + \hat{h}^n_{k_n}\Lambda_{k_n} \tag{18}$$

as the objective function of the equivalent global consensus version of problem (14).

For ease of description, we define the following set as the local variable *feasible set* of each small cell $n \in \mathcal{N}$:

$$\xi_n = \left\{ \begin{array}{l|l} \hat{\boldsymbol{a}}^n & \sum_{k_n\in\mathcal{K}_n} \tilde{s}_{k_n} \leq 1 \\ \tilde{\boldsymbol{s}}_n & \sum_{k_n\in\mathcal{K}_n} \tilde{s}_{k_n} Be_{k_n} \leq L_n \\ \hat{\boldsymbol{c}}^n & \sum_{j\in\mathcal{N}/\{n\}}\sum_{k_j\in\mathcal{K}_j} \hat{a}^n_{k_j} p_{k_j} G_{k_j,n} \leq I_n \\ \hat{\boldsymbol{h}}^n & \sum_{j\in\mathcal{N}}\sum_{k_j\in\mathcal{K}_j} \hat{c}^n_{k_j} \leq 1 \\ & \hat{c}^n_{k_j} F/D_{k_j} - \hat{a}^n_{k_j} f^{(l)}_{k_j}/D_{k_j} \geq 0, \forall k,j \\ & \sum_{j\in\mathcal{N}}\sum_{k_j\in\mathcal{K}_j} \hat{h}^n_{k_j} \leq Y \\ & \hat{a}^n_{k_n} \geq \tilde{s}_{k_n}, \hat{a}^n_{k_n} \geq \hat{c}^n_{k_n}, \forall k \end{array} \right\}, \forall n. \tag{19}$$

Note that ξ_n is proprietary for small cell n and is completely decoupled from other small cells.

Next we give the local utility function of each small cell $n \in \mathcal{N}$ as follows

$$v_n = \begin{cases} -\left[\sum_{k_n\in\mathcal{K}_n} \hat{a}^n_{k_n} u\left(\frac{\tilde{s}_{k_n}\Psi_{k_n} + \hat{c}^n_{k_n}\Phi_{k_n}}{\hat{a}^n_{k_n}}\right) + \hat{h}^n_{k_n}\Lambda_{k_n}\right], \\ \qquad when \quad \{\hat{\boldsymbol{a}}^n, \tilde{\boldsymbol{s}}_n, \hat{\boldsymbol{c}}^n, \hat{\boldsymbol{h}}^n\} \in \xi_n, \\ +\infty, \quad otherwise. \end{cases} \tag{20}$$

With (19) and (20), an equivalent formulation of the problem is given as

$$\begin{aligned} \underset{\{\hat{\boldsymbol{a}}^n,\tilde{\boldsymbol{s}},\hat{\boldsymbol{c}}^n,\hat{\boldsymbol{h}}^n\},\{\boldsymbol{a},\tilde{\boldsymbol{c}},\boldsymbol{h}\}}{\text{Minimize}} \quad & \sum_{n\in\mathcal{N}} v_n(\hat{\boldsymbol{a}}^n, \tilde{\boldsymbol{s}}_n, \hat{\boldsymbol{c}}^n, \hat{\boldsymbol{h}}^n) \\ s.t. \quad C1: \quad & \hat{a}^n_{k_j} = a_{k_j}, \forall n,k,j \\ C2: \quad & \hat{c}^n_{k_j} = \tilde{c}_{k_j}, \forall n,k,j \\ C3: \quad & \hat{h}^n_{k_j} = h_{k_j}, \forall n,k,j. \end{aligned} \tag{21}$$

IV. Problem Solving via Alternating Direction Method of Multipliers

A. Augmented Lagrangian and ADMM Sequential Iterations

According to [19], the augmented Lagrangian of problem (21) is given as

$$\begin{aligned} &L_\rho(\{\hat{\boldsymbol{a}}^n, \tilde{\boldsymbol{s}}_n, \hat{\boldsymbol{c}}^n, \hat{\boldsymbol{h}}^n\}_{n\in\mathcal{N}}, \{\boldsymbol{a}, \tilde{\boldsymbol{c}}, \boldsymbol{h}\}, \{\boldsymbol{\sigma}^n, \boldsymbol{\omega}^n, \boldsymbol{\tau}^n\}_{n\in\mathcal{N}}) = \\ &\sum_{n\in\mathcal{N}} v_n(\hat{\boldsymbol{a}}^n, \tilde{\boldsymbol{s}}_n, \hat{\boldsymbol{c}}^n, \hat{\boldsymbol{h}}^n) + \sum_{n\in\mathcal{N}}\sum_{j\in\mathcal{N}}\sum_{k_j\in\mathcal{K}_j} \sigma^n_{k_j}(\hat{a}^n_{k_j} - a_{k_j}) \\ &+ \sum_{n\in\mathcal{N}}\sum_{j\in\mathcal{N}}\sum_{k_j\in\mathcal{K}_j} \omega^n_{k_j}(\hat{c}^n_{k_j} - \tilde{c}_{k_j}) \\ &+ \sum_{n\in\mathcal{N}}\sum_{j\in\mathcal{N}}\sum_{k_j\in\mathcal{K}_j} \tau^n_{k_j}(\hat{h}^n_{k_j} - h_{k_j}) \\ &+ \frac{\rho}{2}\sum_{n\in\mathcal{N}}\sum_{j\in\mathcal{N}}\sum_{k_j\in\mathcal{K}_j} (\hat{a}^n_{k_j} - a_{k_j})^2 \\ &+ \frac{\rho}{2}\sum_{n\in\mathcal{N}}\sum_{j\in\mathcal{N}}\sum_{k_j\in\mathcal{K}_j} (\hat{c}^n_{k_j} - \tilde{c}_{k_j})^2 \\ &+ \frac{\rho}{2}\sum_{n\in\mathcal{N}}\sum_{j\in\mathcal{N}}\sum_{k_j\in\mathcal{K}_j} (\hat{h}^n_{k_j} - h_{k_j})^2, \end{aligned} \tag{22}$$

*Here we need to introduce another small cell index $j \in \mathcal{N}$ to indicate each small cell in the local copy of small cell n.

where $\boldsymbol{\sigma}^n = \{\sigma_{k_j}^n\}_{n\in\mathcal{N}}$, $\boldsymbol{\omega}^n = \{\omega_{k_j}^n\}_{n\in\mathcal{N}}$ and $\boldsymbol{\tau}^n = \{\tau_{k_j}^n\}_{n\in\mathcal{N}}$ are the Lagrange multipliers, and $\rho \in \mathbb{R}_{++}$ is the so called *penalty parameter*, which is a constant parameter intended for adjusting the convergence speed of ADMM [19].

With ADMM being applied to solving problem (21), the following sequential iterative optimization steps are presented as findings [19]. Local variables:

$$\{\hat{\boldsymbol{a}}^n, \tilde{\boldsymbol{s}}_n, \hat{\boldsymbol{c}}^n, \hat{\boldsymbol{h}}^n\}_{n\in\mathcal{N}}^{[t+1]} = \underset{\{\hat{a}_{k_j}^n, \tilde{s}_{k_n}, \hat{c}_{k_j}^n, \hat{h}_{k_j}^n\}}{\arg\min} \left\{ \begin{array}{l} v_n(\hat{\boldsymbol{a}}^n, \tilde{\boldsymbol{s}}_n, \hat{\boldsymbol{c}}^n, \hat{\boldsymbol{h}}^n) \\ + \sum\limits_{j\in\mathcal{N}} \sum\limits_{k_j\in\mathcal{K}_j} \sigma_{k_j}^{n[t]} \left(\hat{a}_{k_j}^n - a_{k_j}^{[t]}\right) \\ + \sum\limits_{j\in\mathcal{N}} \sum\limits_{k_j\in\mathcal{K}_j} \omega_{k_j}^{n[t]} \left(\hat{c}_{k_j}^n - \tilde{c}_{k_j}^{[t]}\right) \\ + \sum\limits_{j\in\mathcal{N}} \sum\limits_{k_j\in\mathcal{K}_j} \tau_{k_j}^{n[t]} \left(\hat{h}_{k_j}^n - h_{k_j}^{[t]}\right) \\ + \frac{\rho}{2} \sum\limits_{j\in\mathcal{N}} \sum\limits_{k_j\in\mathcal{K}_j} \left(\hat{a}_{k_j}^n - a_{k_j}^{[t]}\right)^2 \\ + \frac{\rho}{2} \sum\limits_{j\in\mathcal{N}} \sum\limits_{k_j\in\mathcal{K}_j} \left(\hat{c}_{k_j}^n - \tilde{c}_{k_j}^{[t]}\right)^2 \\ + \frac{\rho}{2} \sum\limits_{j\in\mathcal{N}} \sum\limits_{k_j\in\mathcal{K}_j} \left(\hat{h}_{k_j}^n - h_{k_j}^{[t]}\right)^2 \end{array} \right\} \tag{23}$$

Global variables:

$$\{\boldsymbol{a}\}^{[t+1]} = \underset{\{a_{k_j}\}}{\arg\min} \left\{ \begin{array}{l} \sum\limits_{n\in\mathcal{N}} \sum\limits_{j\in\mathcal{N}} \sum\limits_{k_j\in\mathcal{K}_j} \sigma_{k_j}^{n[t]} \left(\hat{a}_{k_j}^{n[t+1]} - a_{k_j}\right) \\ + \frac{\rho}{2} \sum\limits_{n\in\mathcal{N}} \sum\limits_{j\in\mathcal{N}} \sum\limits_{k_j\in\mathcal{K}_j} \left(\hat{a}_{k_j}^{n[t+1]} - a_{k_j}\right)^2 \end{array} \right\} \tag{24}$$

$$\{\tilde{\boldsymbol{c}}\}^{[t+1]} = \underset{\{\tilde{c}_{k_j}\}}{\arg\min} \left\{ \begin{array}{l} \sum\limits_{n\in\mathcal{N}} \sum\limits_{j\in\mathcal{N}} \sum\limits_{k_j\in\mathcal{K}_j} \omega_{k_j}^{n[t]} \left(\hat{c}_{k_j}^{n[t+1]} - \tilde{c}_{k_j}\right) \\ + \frac{\rho}{2} \sum\limits_{n\in\mathcal{N}} \sum\limits_{j\in\mathcal{N}} \sum\limits_{k_j\in\mathcal{K}_j} \left(\hat{c}_{k_j}^{n[t+1]} - \tilde{c}_{k_j}\right)^2 \end{array} \right\} \tag{25}$$

$$\{\boldsymbol{h}\}^{[t+1]} = \underset{\{h_{k_j}\}}{\arg\min} \left\{ \begin{array}{l} \sum\limits_{n\in\mathcal{N}} \sum\limits_{j\in\mathcal{N}} \sum\limits_{k_j\in\mathcal{K}_j} \tau_{k_j}^{n[t]} \left(\hat{h}_{k_j}^{n[t+1]} - h_{k_j}\right) \\ + \frac{\rho}{2} \sum\limits_{n\in\mathcal{N}} \sum\limits_{j\in\mathcal{N}} \sum\limits_{k_j\in\mathcal{K}_j} \left(\hat{h}_{k_j}^{n[t+1]} - h_{k_j}\right)^2 \end{array} \right\} \tag{26}$$

Lagrange multipliers:

$$\{\boldsymbol{\sigma}^n\}_{n\in\mathcal{N}}^{[t+1]} = \boldsymbol{\sigma}^{n[t]} + \rho(\hat{\boldsymbol{a}}^{n[t+1]} - \boldsymbol{a}^{[t+1]}) \tag{27}$$

$$\{\boldsymbol{\omega}^n\}_{n\in\mathcal{N}}^{[t+1]} = \boldsymbol{\omega}^{n[t]} + \rho(\hat{\boldsymbol{c}}^{n[t+1]} - \tilde{\boldsymbol{c}}^{[t+1]}) \tag{28}$$

$$\{\boldsymbol{\tau}^n\}_{n\in\mathcal{N}}^{[t+1]} = \boldsymbol{\tau}^{n[t]} + \rho(\hat{\boldsymbol{h}}^{n[t+1]} - \boldsymbol{h}^{[t+1]}), \tag{29}$$

where the superscript $[t]$ stands for the iteration index.

B. Local Variables $\{\hat{\boldsymbol{a}}^n, \tilde{\boldsymbol{s}}_n, \hat{\boldsymbol{c}}^n, \hat{\boldsymbol{h}}^n\}_{n\in\mathcal{N}}$ Update

After eliminating the constant terms, it is equivalent for SeNB $n \in \mathcal{N}$ to solve the following optimization problem at iteration $[t+1]$,

$$\begin{aligned} \underset{\{\hat{a}_{k_j}^n, \tilde{s}_{k_n}, \hat{c}_{k_j}^n, \hat{h}_{k_j}^n\}}{\text{Minimize}} \quad & v_n(\hat{\boldsymbol{a}}^n, \tilde{\boldsymbol{s}}_n, \hat{\boldsymbol{c}}^n, \hat{\boldsymbol{h}}^n) \\ & + \sum_{j\in\mathcal{N}} \sum_{k_j\in\mathcal{K}_j} \left[\sigma_{k_j}^{n[t]} \hat{a}_{k_j}^n + \frac{\rho}{2} \left(\hat{a}_{k_j}^n - a_{k_j}^{[t]}\right)^2\right] \\ & + \sum_{j\in\mathcal{N}} \sum_{k_j\in\mathcal{K}_j} \left[\omega_{k_j}^{n[t]} \hat{c}_{k_j}^n + \frac{\rho}{2} \left(\hat{c}_{k_j}^n - \tilde{c}_{k_j}^{[t]}\right)^2\right] \\ & + \sum_{j\in\mathcal{N}} \sum_{k_j\in\mathcal{K}_j} \left[\tau_{k_j}^{n[t]} \hat{h}_{k_j}^n + \frac{\rho}{2} \left(\hat{h}_{k_j}^n - h_{k_j}^{[t]}\right)^2\right] \\ s.t. \quad & \{\hat{\boldsymbol{a}}^n, \tilde{\boldsymbol{s}}_n, \hat{\boldsymbol{c}}^n, \hat{\boldsymbol{h}}^n\} \in \xi_n. \end{aligned} \tag{30}$$

Obviously, problem (30) is a convex problem, which can be solved using *primal-dual interior-point method*.

C. Global Variables $\{\boldsymbol{a}, \tilde{\boldsymbol{c}}, \boldsymbol{h}\}$ and Lagrange Multipliers $\{\boldsymbol{\sigma}^n, \boldsymbol{\omega}^n, \boldsymbol{\tau}^n\}_{n\in\mathcal{N}}$ Update

Since problem (24), (25) and (26) are unconstrained quadratic problems and are strictly convex, we can solve them by simply setting the *gradients* of $\boldsymbol{a}$, $\tilde{\boldsymbol{c}}$ and $\boldsymbol{h}$ to zeros, and this result in

$$a_{k_j}^{[t+1]} = \frac{1}{N\rho} \sum_{n\in\mathcal{N}} \sigma_{k_j}^{n[t]} + \frac{1}{N} \sum_{n\in\mathcal{N}} \hat{a}_{k_j}^{n[t+1]}, \quad \forall k, j \tag{31}$$

$$\tilde{c}_{k_j}^{[t+1]} = \frac{1}{N\rho} \sum_{n\in\mathcal{N}} \omega_{k_j}^{n[t]} + \frac{1}{N} \sum_{n\in\mathcal{N}} \hat{c}_{k_j}^{n[t+1]}, \quad \forall k, j \tag{32}$$

$$h_{k_j}^{[t+1]} = \frac{1}{N\rho} \sum_{n\in\mathcal{N}} \tau_{k_j}^{n[t]} + \frac{1}{N} \sum_{n\in\mathcal{N}} \hat{h}_{k_j}^{n[t+1]}, \quad \forall k, j. \tag{33}$$

The process of Lagrange multipliers $\{\boldsymbol{\sigma}^n, \boldsymbol{\omega}^n, \boldsymbol{\tau}^n\}_{n\in\mathcal{N}}$ updating is simple. With the current local variables received from each SeNB, the MEC server can easily obtain the Lagrange multipliers by calculating equations (27)–(29) in each iteration.

V. Simulation Results and Discussions

In this section, simulation results of the proposed scheme are presented in comparison with the centralized scheme and several baseline schemes. We consider 10-50 small cells that are randomly deployed in a 120×120 m^2 area. There are 4-10 UEs connected to one SeNB, as mentioned in Section II. The transmission power of single UE, P_n is set to 100 mW. The channel gain models presented in 3GPP standardization [20] are adopted here. The total size of the Internet content is 1000 files, and the storage capability of the MEC server is 1000 files. The main parameters employed in the simulations, unless mentioned otherwise, are summarized in table I.

Fig. 1 shows the revenue of MSO (utility value) with respect to the increasing number of small cells. The centralized algorithm and our proposed ADMM-based algorithm achieve relatively high revenue among all the six solutions. It can be seen that the gap between our proposed algorithm and the centralized algorithm is narrow.

Fig. 2 shows the average UE time consumption of ADMM based algorithm and other solutions, with respect to the number of small cells. As we can see, the time consumption

TABLE I: Simulation parameters

Parameter	Value
Bandwidth	20MHz
Transmission power of UE n, P_n	100 mWatts
Background noise σ^2	-100 dBm
Data size for computation offloading Z_{k_n}	420 KB
Number of CPU cycles of computation task D_{k_n}	1,000 Megacycles
Computation capability of UE n, $f_{k_n}^{(l)}$	0.7 GHz [21]
Computation capability of the MEC server F	100 GHz [21]

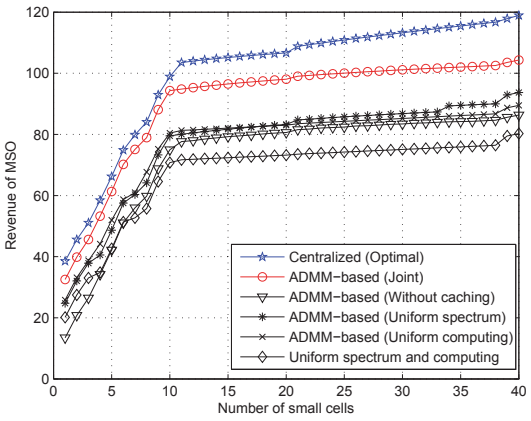

Fig. 1: MSO revenue versus the number of small cells.

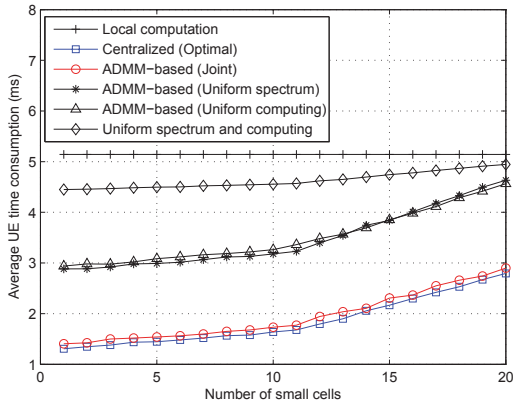

Fig. 2: Average UE time consumption versus the number of small cells.

performance of proposed ADMM based algorithm is very close to that of the centralized algorithm here.

VI. Conclusions and Future Work

In this paper, we formulated the computation offloading decision, spectrum resource allocation, MEC computation resource allocation, and content caching issues as an optimization problem. Then, in order to tackle this problem in an efficient way, we presented an ADMM-based distributed solution. Simulation results demonstrated that the proposed scheme can achieve better performance than other baseline solutions under various system parameters. Future work is in progress to consider wireless network virtualization in the proposed framework.

Acknowledgment

This work is jointly supported by the National Natural Science Foundation of China (Grant No. 61571073) and the National High Technology Research and Development Program of China (Grant No. 2014AA01A701).

References

[1] J. Hoadley and P. Maveddat, "Enabling small cell deployment with HetNet," *IEEE Wireless Commun. Mag.*, vol. 19, no. 2, pp. 4–5, Apr. 2012.

[2] R. Xie, F. R. Yu, H. Ji, and Y. Li, "Energy-efficient resource allocation for heterogeneous cognitive radio networks with femtocells," *IEEE Trans. Wireless Commun.*, vol. 11, no. 11, pp. 3910 –3920, Nov. 2012.

[3] Z. Li, F. R. Yu, and M. Huang, "A distributed consensus-based cooperative spectrum sensing in cognitive radios," *IEEE Trans. Veh. Tech.*, vol. 59, no. 1, pp. 383–393, Jan. 2010.

[4] A. Elsherif, W.-P. Chen, A. Ito, and Z. Ding, "Adaptive resource allocation for interference management in small cell networks," *IEEE Trans. Commun.*, vol. 63, no. 6, pp. 2107–2125, Jun. 2015.

[5] ETSI, "Mobile-edge computing: Introductory technical white paper," *ETSI White Paper*, Sept. 2014.

[6] O. Munoz, A. Pascual-Iserte, and J. Vidal, "Optimization of radio and computational resources for energy efficiency in latency-constrained application offloading," *IEEE Trans. Veh. Technol.*, vol. 64, no. 10, pp. 4738–4755, Oct. 2015.

[7] Z. Yin, F. R. Yu, S. Bu, and Z. Han, "Joint cloud and wireless networks operations in mobile cloud computing environments with telecom operator cloud," *IEEE Trans. Wireless Commun.*, vol. 14, no. 7, pp. 4020–4033, July 2015.

[8] K. Wang, F. R. Yu, and H. Li, "Information-centric virtualized cellular networks with device-to-device (D2D) communications," *IEEE Trans. Veh. Tech.*, vol. 65, no. 11, pp. 9319 – 9329, Nov. 2016.

[9] W. Chai, D. He, I. Psaras, and G. Pavlou, "Cache less for more in information-centric networks (extended version)," *Computer Commun.*, vol. 36, no. 7, pp. 758–770, Apr. 2013.

[10] C. Liang, F. R. Yu, and X. Zhang, "Information-centric network function virtualization over 5G mobile wireless networks," *IEEE Network*, vol. 29, no. 3, pp. 68–74, May 2015.

[11] Z. Chen and D. Wu, "Rate-distortion optimized cross-layer rate control in wireless video communication," *IEEE Trans. Circ. Sys. Video Tech.*, vol. 22, no. 3, pp. 352–365, Mar. 2012.

[12] L. Ma, F. Yu, V. C. M. Leung, and T. Randhawa, "A new method to support UMTS/WLAN vertical handover using SCTP," *IEEE Wireless Commun.*, vol. 11, no. 4, pp. 44–51, Aug. 2004.

[13] F. Yu and V. C. M. Leung, "Mobility-based predictive call admission control and bandwidth reservation in wireless cellular networks," in *Proc. IEEE INFOCOM'01*, Anchorage, AK, Apr. 2001.

[14] F. Yu and V. Krishnamurthy, "Optimal joint session admission control in integrated WLAN and CDMA cellular networks with vertical handoff," *IEEE Trans. Mobile Computing*, vol. 6, no. 1, pp. 126–139, Jan. 2007.

[15] C. Fricker, P. Robert, J. Roberts, and N. Sbihi, "Impact of traffic mix on caching performance in a content-centric network," in *Proc. IEEE INFOCOM'12*, Mar. 2012, pp. 310–315.

[16] N. Golrezaei, K. Shanmugam, A. Dimakis, A. Molisch, and G. Caire, "Femtocaching: Wireless video content delivery through distributed caching helpers," in *Proc. IEEE INFOCOM'12*, Mar. 2012.

[17] D. Fooladivanda and C. Rosenberg, "Joint resource allocation and user association for heterogeneous wireless cellular networks," *IEEE Trans. Wireless Commun.*, vol. 12, no. 1, pp. 248–257, Jun. 2013.

[18] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge, UK: Cambridge university press, 2009.

[19] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein, "Distributed optimization and statistical learning via the alternating direction method of multipliers," *Foundations and Trends in Machine Learning*, vol. 3, no. 1, pp. 1–122, Jan. 2011.

[20] 3rd Generation Partnership Project. (2012) Evolved universal terrestrial radio access (E-UTRA); further advancements for E-UTRA physical layer aspects (release 9) 3gpp ts 36.814.

[21] X. Chen, L. Jiao, L. Wenzhong, and F. Xiaoming, "Efficient multi-user computation offloading for mobile-edge cloud computing," *IEEE/ACM Trans. Netw.*, vol. PP, no. 99, pp. 1–14, 2015, IEEE early access article.