Joint Load Management and Resource Allocation in the Energy Harvesting Powered Small Cell Networks with Mobile Edge Computing

Fengxian Guo*, Liangde Ma+, Heli Zhang*, Hong Ji*, Xi Li*
*Key Laboratory of Universal Wireless Communications, Ministry of Education
Beijing University of Posts and Telecommunications Beijing, P.R. China
+China Academy of Railway Sciences

***Abstract*—Considering the advantages of small cell networks (SCNs) and mobile edge computing (MEC), it is natural to combine small cells with MEC, which can not only enhance the computation capacity of the mobile user equipment (UE) but also provide the users with a high quality of services (QoS). However, the high energy consumption becomes a rigorous problem to the operators and the environment with the small cell base stations (SBSs) densely deployed. Hence, it is crucial to exploit off-grid and green energy resources to power the SBSs, for which energy harvesting (EH) is one of the viable solutions. But there impose formidable challenges combining EH-based SCNs with MEC, due to the fact that the energy arrival and traffic are not uniformly distributed across space and time domain. In this work, we jointly study the green load management and resource allocation scheme in such a network for achieving green computation offloading under the constraints of network cost and users' QoS. First, we formulate the main problem. To reduce the complexity of the proposed problem, we adopt a distributed three-stage iteration approach to solve it. To avoid overload and make full use of the resources of SBSs, cell range expansion (CRE) is used to adjust the coverage of each SBS to match the resource state and traffic pattern. After that, channel allocation is performed considering interference migration. Based on the results of the former two steps, the MEC servers dynamically distribute the computation resources to the UEs. Simulation results verified the performance of the proposed algorithm with different system parameters.**

***Index Terms*—MEC, small cell networks, energy harvesting, computation offloading, load management**

I. Introduction

The mobile cellular network has experienced an explosive growth in the past decades. On one hand, tremendous users are connecting to the wireless network. Cisco predicted that there will be approximately 11.6 billion mobile-ready devices/connections, including M2M modules -exceeding the world's projected population at that time (7.8 billion) [1]. On the other hand, the users' demands on new applications become more urgent with the development of mobile devices, which may be not only latency-sensitive but also computation intensive. However, such a growth in computation ability is not matched with an equally fast improvement on mobile devices' batteries [2]. These pose new challenges to the next generation cellular network regarding the network capacity and ability to support low-latency computation offloading services.

Integrating mobile edge computing (MEC) with small cell networks (SCNs) has drawn much attention from the academic and industry, which can not only compensate the computation capacity of mobile user equipments (UEs) but also provide the UEs with high quality of services (QoS), due to the powerful capacity of edge servers at the small cell base stations (SBSs) and the short communication distance between the UEs and the SBSs. There have been many works on SCNs with MEC [3]–[6], which mainly focus on the optimal computation offloading strategy or resource management scheme. These works all assume that there exists a common cloud server in the network. In this work, we consider the computation offloading in an SCN where each SBS is deployed with one MEC server.

With the expansion of mobile users and densely deployed SBSs, the energy consumption in the cellular networks increases sharply nowadays [7]. First, it may not be possible to provide grid power to all SBSs in a cost-effective way [8]. Second, a dense deployment of SBSs will not only increase operators' operating expense (OPEX) but also lead to significant carbon emission. Thus, a green wireless communication technology becomes essential for both economic and environmental reasons, for which energy harvesting (EH) is one of the solutions. EH is an emerging technology, which enables the low-power SBSs to harvest energy from the solar or wind and use the green energy to serve the users. Hence, it is natural to combine EH-based SCNs with MEC.

However, the randomness of energy arrival in space and time domain results in non-uniformly distribution of energy across different SBSs. In addition, the traffic is not uniformly distributed, which may lead to load unbalance among the SBSs. Hence, it is important to manage the load in such a network, while concurrently ensuring the users' QoS. However, there still lack works on this problem in such a network.

In this paper, we investigate the joint load management and resource allocation problem in an EH-powered SCN with MEC. Regarding one time period, we aim at maximizing the number of offloading users utilizing the limited energy and computation resources, via managing the load and distributing the resources to the UEs. First, we formulate the main problem. To reduce the complexity of the proposed problem, we adopt a distributed three-stage iteration approach to solve it. To manage the load, cell range expansion (CRE) is used to adjust the coverage of each SBS according to the resource state and traffic pattern. After that, channel allocation is performed

978-1-5386-5979-3/18/$31.00 ©2018 IEEE

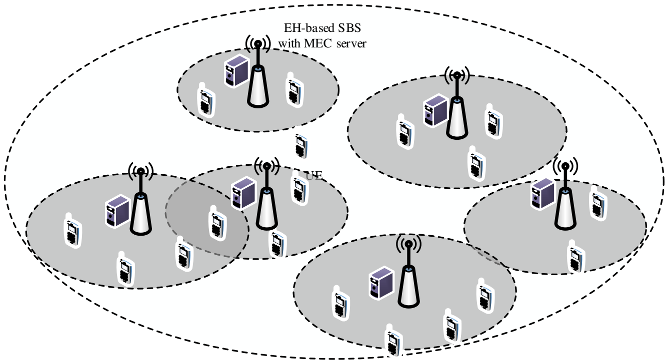

Fig. 1: Network Architecture (1). The SBSs in this network are equipped with MEC servers. (2). The SBSs are only EH powered. (3). Each of the UEs has a computation-intensive and delay-sensitive task to complete.

considering interference migration. The MEC servers use the outcomes of the former two steps to distribute the computation resources to the UEs dynamically. Simulation results show the effectiveness of the proposed method.

The rest of this paper is organized as follows. Section II presents the system model and problem formulation. The proposed method is presented in Section III. Section IV shows the simulation results. Finally, section V concludes this work.

II. System Model

In this paper, we consider a MEC system in a multi-user SCN, where each of the M SBSs is equipped with a MEC server. The set of SBSs is denoted by $\mathcal{M} = \{1, 2, ..., i..., M\}$. The SBSs are only EH-powered, which means the SBSs harvest energy from the wind or solar. In this network, there are N UEs, each of which has a computation task to complete. We denote the set of UEs as $\mathcal{N} = \{1, 2, ..., N\}$. The UEs can execute the task locally or offload it to the MEC servers on the SBSs. In this paper, we don't consider the mobility of the UEs as in [5]. We assume that time is slotted due to the SBSs are EH-powered, the energy state of which is related to time horizon. The length of one time slot is denoted by τ. In this paper, we only consider one time slot and ignore the time slot index t unless ambiguity rises.

A. Communication Model

We first introduce the communication model for computation offloading. We assume each SBS posses M orthogonal channels, the set of which is denoted by $\mathcal{C} = \{1, 2, ..., c, ..., C\}$. Furthermore, we design a parameter $a_n^m \in 0 \cup \mathcal{C}$ to depict the offloading decision, access and channel allocation. Specially, there is $a_n^m = c \in \mathcal{C}$ when UE n decides to offload to SBS m and SBS m allocates channel c to UE n. Otherwise, we have $a_n^m = 0$. Here we denote the offloading indicator of UE n as $\hat{a}_n = \{a_n^m, m \in \mathcal{M}\}$. Given the offloading indicator profile $A = \{\hat{a}_1, \hat{a}_2, ..., \hat{a}_N\}$ of all UEs, the data rate of UE n that chooses to offload to SBS m (i.e., $a_n^m > 0$) can be computed by

$$R_n^m(A, P) = I_{\{a_n^m > 0\}} w \\ log_2(1 + \frac{p_n |h_{n,m}|^2}{\sum\limits_{i \in \mathcal{N} \setminus \{n\}} \sum\limits_{j \in \mathcal{M} \setminus \{m\}} I_{\{a_i^j = a_n^m\}} p_i |h_{i,m}|^2 + N_0}) \quad (1)$$

where w is the channel bandwidth, and $h_{n,m}$ denotes the channel coefficient between UE n and SBS m. p_n is the transmission power of UE n. There exists $p_n < p_{max}$, where p_{max} is the maximum transmission power of the UEs. $I_{\{\cdot\}}$ is an indicator function, where there is $I_{\{\cdot\}} = 1$ when the event $\{\cdot\}$ is true. Otherwise, there is $I_{\{\cdot\}} = 0$.

In this paper, one UE can communicate with the SBSs only if it is in the coverage of the SBSs. Furthermore, we assume that each SBS can dynamically adjust its' coverage range to manage its' load based on the traffic and energy state.

Worth noting, the transmission power control is not considered in this work, and the reason is the same as [6]. In our future work, we will take into account of the power control scheme.

B. Computation Model

Give the computation task of UE n, which is a 3-tuple $I_n = (L_n, B_n, \tau_n)$. L_n (in bits) is the data size of input of UE n' task and B_n (in CPU cycles) is the number of total CPU cycles to complete this task. τ_n is the execution deadline, which means it must be completed within τ_n no matter which computation ways UE n chooses. Specially, we assume the UEs' applications are with execution deadline no greater than the time slot length, i.e., $\tau_n < \tau$, $\forall n \in \mathcal{N}$ [9].

1) Local Computing Model: Given the computation ability F_n (in Hz) of UE n, which is fixed when executing the task but varies over different UEs, the delay of local computing can be computed by

$$T_n^l = \frac{B_n}{F_n} \quad (2)$$

Accordingly, the energy consumption for local computing of UE n is given by

$$E_n^l = T_n^l P_n^{l,e} = B_n \kappa_n F_n^2 \quad (3)$$

where $P_n^{l,e} = \kappa_n F_n^3$ is the power consumption of UE n' CPU, and κ_n is a coefficient depending on the chip architecture [10].

2) Mobile Edge Computing Model: In this case, the UEs first transmit the task to the MEC servers to perform mobile-edge computing, which consumes energy from the UEs and causes transmission latency for offloading. Then the MEC servers execute the tasks, which consumes energy from the SBSs and results in execution delay for offloading. We assume that the transmission delay for feedback is ignored due to the small size of the output [4], [6]. Hence, the transmission delay and energy consumption offloading to SBS m ($a_n^m > 0$) can be computed by

$$T_{n,m}^{off,tr} = \frac{D_n}{R_n^m} \quad (4)$$

$$E_{n,m}^{off,tr} = p_n T_{n,m}^{off,tr} \quad (5)$$

The MEC servers are further assumed to be capacity-constrained due to the inherent space limitation on the SBSs. Denote the computation capacity (in Hz) of SBS m as F_m.

When UE n determines to offload to SBS m, SBS m allocates f_n^m computation resources to UE n. So we have the computation resource allocation profile $F = \{f_n^m, n \in \mathcal{N}, m \in \mathcal{M}\}$. Hence, the execution delay and energy consumption on SBS m can be given by

$$T_{n,m}^{off,e} = \frac{B_n}{f_n^m} \tag{6}$$

$$E_{m,n}^{off,e} = T_{n,m}^{off,e} P_m^{n,e} = B_n \kappa_m (f_n^m)^2 \tag{7}$$

where $P_m^{n,e} = \kappa_m (f_n^m)^3$ is the execution power when allocating f_n^m computation capacity to UE n.

So the delay that UE n experienced by offloading is $T_{n,m}^{off} = T_{n,m}^{off,tr} + T_{n,m}^{off,e}$. The energy consumption of UE n is $E_{n,m}^{off} = E_{n,m}^{off,tr}$ and the energy consumption of SBS m with UE n is $E_{m,n}^{off} = E_{m,n}^{off,e}$.

C. Energy Harvesting Model

Suppose that each SBS is powered solely by the energy harvested from the wind or solar and equipped with a rechargeable battery with finite capacity $E_{m,max}$. The stored energy can be available for executing the tasks of the offloading UEs and keeping the SBSs running, such as for cooling, battery backup and so on.

Let $\hat{E}_m[t]$ and $E_m[t]$ be the energy arrival during t_{th} time slot and battery energy state at the beginning of t_{th} time slot of SBS m. Hence, the dynamics of energy state of SBS m can be described as

$$E_m[t+1] = E_m[t] + \hat{E}_m[t] - U_m[t] \tag{8}$$

where $U_m[t]$ is the energy consumption of SBS m during time slot t, which can be expressed by

$$U_m[t] = \sum_{n \in \mathcal{N}} I_{\{a_n^m[t]>0\}} E_{m,n}^{off} + P_{m,static}\tau \tag{9}$$

where $P_{m,static}$ is the static expenditure of SBS m [11] including circuit power, battery backup, cooling and so on. As noted, the SBSs turn into sleep mode when the available energy can't support the static expenditure in the time slot.

D. Problem Formulation

According to the analysis aforementioned, the experienced delay of UE n is

$$T_n = \sum_{m \in \mathcal{M}} I_{\{a_n^m>0\}} T_{n,m}^{off} + \Big(1 - \sum_{m \in \mathcal{M}} I_{\{a_n^m>0\}}\Big) T_n^l. \tag{10}$$

The energy consumption of UE n is

$$E_n = \sum_{m \in \mathcal{M}} I_{\{a_n^m>0\}} E_{n,m}^{off} + \Big(1 - \sum_{m \in \mathcal{M}} I_{\{a_n^m>0\}}\Big) E_n^l. \tag{11}$$

The optimization problem is formulated to maximize the number of offloading UEs as the following

$$\begin{aligned}
P_0 : \max_{A,F} \quad & \sum_{n \in \mathcal{N}} I_{\{T_n<T_n^l \& E_n<E_n^l\}} \\
\texttt{s.t.} \quad & C1 : T_n \le \tau_n, \quad \forall n \in \mathcal{N} \\
& C2 : a_n^m \in 0 \cup \mathcal{C}, \quad \forall n \in \mathcal{N} \\
& C3 : a_n^m \ne a_{n'}^m (>0), \quad \forall m \in \mathcal{M} \\
& C4 : \sum_{m \in \mathcal{M}} I_{\{a_n^m>0\}} \in \{0,1\}, \quad \forall n \in \mathcal{N} \\
& C5 : 0 \le f_n^m \le F_m, \quad \forall n \in \mathcal{N} \\
& C6 : \sum_{n \in \mathcal{N}} I_{\{a_n^m>0\}} f_n^m \le F_m, \quad \forall m \in \mathcal{M} \\
& C7 : U_m \le E_m, \quad \forall m \in \mathcal{M}
\end{aligned} \tag{12}$$

Constraint $C1$ ensures the QoS of all UEs. Constraints $C2 \sim C3$ indicate the principles of channel allocation. $C2$ limits the range of channel allocation while $C3$ means that one channel in one cell can only be assigned to one UE. $C4$ indicates that one UE can be assigned at most one channel. $C5$ indicates the range of the amount of allocated computation resources. $C6$ reveals that the computation resources allocated to all UEs can't exceed that each SBS posses. According to $C7$, the energy consumption in one time slot won't exceed the stored energy at each SBS.

This problem is hard to optimize for a simple version of this problem given in [3] is proved to be NP-hard.

III. The Proposed Joint Load Management and Resource Allocation Approach

In this section, we present the proposed joint load management and resource allocation approach to optimize the capacity of the EH-powered SBSs to serve the UEs, which is quantified by the number of offloading UEs. Considering the proposed problem is NP-hard, the complexity of which may be extraordinarily high. Then we decompose the original problem into three sub-problems and propose a sub-optimal three-stage solution, taking into account of the performance of the UEs and the inherently limited resources in this network.

First, for a given channel allocation and computation resource allocation scheme, we manage the traffic load based on the energy state of the SBSs while considering the performance of the UEs. Second, based on the load traffic and a given computation resource allocation policy, the channel allocation is optimized aiming at minimizing the transmission energy consumption of the UEs. Third, for a given channel allocation policy and a set of offloading UEs (some given load traffic), the computation resources are distributed to the UEs to maximize the number of offloading UEs. Finally, an iteration based algorithm is adopted to obtain the joint load management and resource allocation solution.

For analysis, we divide a_n^m into two variables, $a_n \in 0 \cup \mathcal{M}$ to denote the selected SBS and $b_n \in 0 \cup \mathcal{C}$ to denote the allocated channel. Hence, the set of selected SBSs is $A^* =$

$\{a_n, n \in \mathcal{N}\}$ and the set of allocated channel is denoted by $B^* = \{b_n, n \in \mathcal{N}\}$. Hence, $U_m[t]$ is transformed into

$$U'_m[t] = \sum_{n \in \mathcal{N}} I_{\{a_n[t]=m\}} I_{\{b_n[t]>0\}} E^{off}_{m,n} + P_{m,static}\tau. \quad (13)$$

Also, the indicator function $I_{\{a^m_n>0\}}$ in expression (10) and (11) can be expressed by $I_{\{a_n=m\}} I_{\{b_n>0\}}$.

A. Load Management for the SBSs

Since the stored energy in the SBSs is limited, how to make full use of the energy is essential to the system performance. We first choose the proper UEs to manage the load on each SBS, while ensuring the QoS of the UEs. Hence, problem (12) can be transformed into

$$\begin{aligned} P_1 : \max_{A^*} \quad & \sum_{n \in \mathcal{N}} I_{\{T_n < T^l_n \& E_n < E^l_n\}} \\ \texttt{s.t.} \quad & C1, C2' : \sum_{n \in \mathcal{N}} I_{\{a_n=m\}} I_{\{b_n>0\}} \le C \\ & C3' : b_n \ne b_{n'}, if \quad a_n = a_{n'} \\ & C6' : \sum_{n \in \mathcal{N}} I_{\{a_n=m\}} f^m_n \le F_m \\ & C7' : U'_m \le E_m \end{aligned} \quad (14)$$

This problem (14) is a combinatorial optimization problem. Since the channel allocation and computation resource allocation policy are given, the delay T_n and energy consumption E_n can be computed by the expressions (1) $\sim$ (7), (10), (11). Thus whether one UE meets the requirement to be offloaded, i.e., $T_n < T^l_n \& E_n < E^l_n$, is obtained, which we called candidate offloading UEs. The set of candidate offloading UE is denoted by $\mathcal{A} = \{n', T_{n'} < T^l_{n'} \& E_{n'} < E^l_{n'}\}$. Concluded from the analysis before, problem (14) is to select the UEs that meets the energy constraint in $C8$ from the candidate offloading UE set. In this work, the knapsack algorithm is adopted to solve this sub-problem [12].

B. Channel Allocation for the SBSs

Given the load on the SBSs, the power selection, and the computation resource allocation, the SBSs allocate the channels to the UEs, managing the interference among the SBSs. Focusing on minimizing the energy consumption of the UEs, the optimal channel allocation can be obtained by solving the following problem

$$\begin{aligned} P_2 : \min_{B^*} \quad & \sum_{n \in \mathcal{N}} E_n \\ \texttt{s.t.} \quad & C1' : T_n \le T_{n,max} \\ & C2', C3', C7' \end{aligned} \quad (15)$$

where there is $T_{n,max} = min\{T^l_n, \tau_n\}$.

For the UEs to offload i.e., $a_n = m > 0$, the maximum interference can be represented by

$$Interference_{n,max} = \frac{p_n |h_{n,m}|^2}{2^{x_{n,m}}} - N_0 \quad (16)$$

where there is $x_{n,m} = D_n/(w(T_{n,max} - T^{off,e}_{n,m}))$. Since A^*, P, and F are given, $Interference_{n,max}$ can be obtained.

Given the other UEs' channel allocation B^*_{-n}, UE n' optimal choice is

$$b^*_n = \arg\min \sum_{i \in \mathcal{N} \backslash \{n\}, a_i \ne a_n} I_{\{b_i = b^*_n\}} p_i |h_{i,m}|^2 \quad (17)$$

Hence, this problem turns into distributing the channel with the lowest interference to the UEs. In this paper, we adopt a graph-based algorithm to solve this problem [13].

C. Computation Allocation for the UEs

We solve the original problem assuming the traffic load and channel allocation are known. For a given traffic load and channel allocation scheme, the UEs to offload are known and the transmission energy consumption and transmission delay can be computed. Hence, the minimal computation resources that an UE needs can be computed by the following equation

$$f^m_{n,min} = \frac{B_n}{T_{n,max} - T^{off,tr}_{n,m}} \quad (18)$$

Thus, this problem (12) can be rewritten as the following problem

$$\begin{aligned} P_3 : \max_{F} \quad & \sum_{n \in \mathcal{N}} I_{\{E_n < E^l_n\}} \\ \texttt{s.t.} \quad & C5' : f^m_n \ge f^m_{n,min} \quad or \quad f^m_n = 0, C6' \\ & C7' : I_{\{a_n=m\}} I_{\{b_n>0\}} (f^m_n)^2 \le \frac{E'_m}{B_n \kappa} \end{aligned} \quad (19)$$

where $E'_m = E_m - p_{m,static}\tau$. Thus this problem is reduced to choose the UEs for each SBS under the constraints listed in problem (19). Observing from this problem, we can find the optimal solution meets the following rule $f^{m*}_n \in \{f^m_{n,min}, 0\}$. To solve this problem, we adopt the priority based algorithm, in which the UEs are chosen according to the amount of computation resources they request.

D. Overall Algorithm

As mentioned above, we can solve the original problem (12) by a three-step iteration algorithm. By dividing the original NP-hard problem into three sub-problems: load management, channel allocation, and computation resource allocation. First, the load management problem is formulated as a combinatorial optimization problem, which can be solved by the knapsack algorithm. Second, the channel allocation is formulated to minimizing the energy consumption of the users. By distributing the channels with the lowest interference to the users, problem (15) can be solved by an improved graph-based algorithm. Third, a priority based algorithm is adopted to assign the computation resources to the users choose to offload and assigned with a certain channel. Finally, conduct the three algorithms iteratively until the overall algorithm converges or meeting the performance demand. Due to the space limitation, we omitted the detailed process of solving this problem. When the traffic load and resource allocation are fixed, the delay, energy consumption and number of offloading UEs can be computed with expression (10) (11) and (12).

TABLE I: The simulation parameters

Simulation parameters	Value
Power spectral density of noise	-174 dBm$/Hz$
The transmission power of the user p_n	200 mW
The bandwidth of each channel w	180 kHz
The input data size of the computation task L_n	$100 \sim 200$ KB
The CUP cycles needed by the computation task B_n	3000 per bite input
The energy coefficient of UE n κ_n	10^{-26}
The energy coefficient of SBS m κ_m	0.5×10^{-25}
The computation capacity of UE n F_n	$0.1 \sim 0.5$ GHz
The computation capacity of SBS m F_m	10 GHz

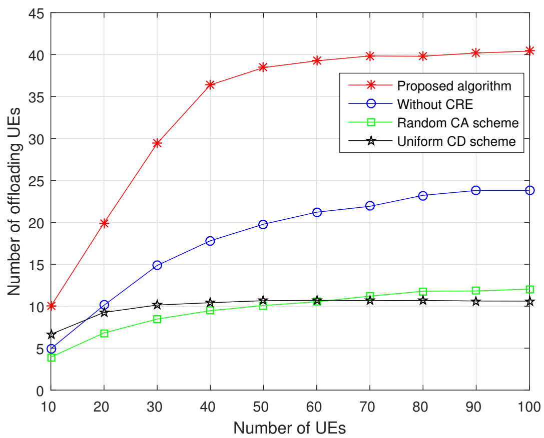

Fig. 2: The number of UEs versus the number of offloading UEs

IV. Simulation Results

In this section, we show the performance of the proposed algorithm in the densely deployed SCN via Monte Carlo simulation. The SBSs and UEs are distributed in the area of 300×300 m^2, and the original radius of each SBS is 100m. The channel gain is modeled by l^δ, where l denotes the distance between the UE and the SBS. $\delta = -3$ is the path loss exponent. The other main parameters are set as shown in Table I, unless ambiguity rises.

To evaluate the effectiveness of the proposed solution, we show its performance in comparison with the other baseline algorithms: 1) the same proposed approach without CRE, as labeled as "Without CRE"; 2) the same proposed solution with random channel assignment, as labeled as "Random CA scheme"; 3) the same proposed approach with computation resources uniformly assigned, as labeled as "Uniform CD scheme". In "Without CRE", the UEs can only access to the SBSs which cover the UEs. In "Uniform CD scheme", the offloading UEs are constrained by the available energy of the SBSs, so we assume that each UE is allocated with 1 GHz computation resources in the case that the energy is not enough when the number of the serving users is low.

Fig.2 plots the number of offloading UEs versus the number of total UEs from 10 to 100, where there are 10 SBSs and each of the SBSs owns 5 channels. First, the number of offloading UEs increases and intends to be steady with the number of UEs rising. That is because it maintains a chance to get a better solution when the UEs get denser. But the resources in the network is limited due to the number of SBSs is set to be fixed. Hence the number of offloading UEs intends to be steady when there are enough UEs. Second, the proposed algorithm maintains the best solution of all. In the solution without CRE, the SBSs with plenty energy can't hold more UEs, which causes severe unmatch between the traffic and energy, resulting in a poor performance. In "Random CA scheme", the channels are randomly assigned, which increases the transmission delay. In such cases, the UEs demand more computation resources and more execution energy from the SBSs. In "Uniform CD scheme", the computation resources are distributed uniformly among the UEs despite the requirement of the UEs. This leads to a low resource efficiency. Hence, the proposed algorithm obtains a higher number of offloading UEs compared with the other baseline algorithms.

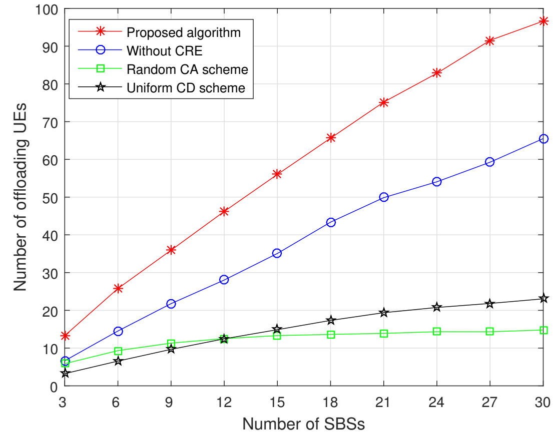

Fig. 3: The number of SBSs versus the number of offloading UEs

Fig.3 illustrates the number of offloading UEs versus the number of SBSs, ranging from 3 to 30. Here are 100 UEs. Concluded from Fig.3, the number of offloading UEs rises when there are more SBSs. The reason is that there are more resources (wireless resources, computation resources, and energy). Though the interference may become severer in this situations, the MEC servers can compromise this influence by assigning more computation resources. Furthermore, the proposed algorithm maintains the best performance compared with the other three baseline algorithms, then follows "Without CRE", "Uniform CD scheme", and "Random CA scheme". But when there are only several SBSs, "Random CA scheme" is better than "Uniform CD scheme". When the number of SBSs is small, the interference among the UEs has a lighter influence. As the wireless transmission condition plays a key role in the computation offloading, "Uniform CD scheme" obtains more offloading UEs than "Random CA scheme" when the SBSs get denser. The proposed algorithm and "Without CRE" conduct channel assignment and computation resource distribution wisely, these two get better performance than

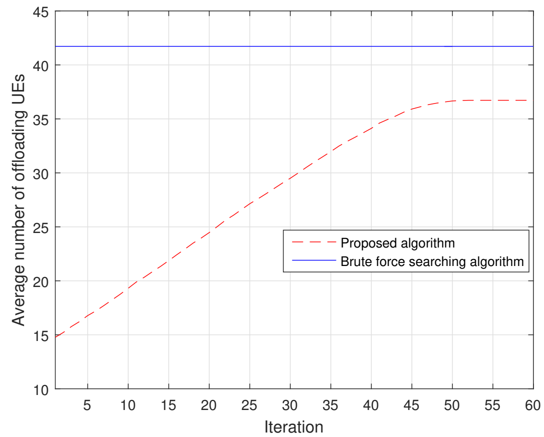

Fig. 4: The convergence of the proposed algorithm

"Random CA scheme" and "Uniform CD scheme". The proposed algorithm adopts CRE to match the traffic and energy, which increases the resource efficiency. Hence, the proposed algorithm is better than "Without CRE" in terms of the number of offloading UEs.

Next, the convergence of the proposed algorithm is shown in Fig.4, and the brute force searching (BFS) algorithm presents the obtained solution as a benchmark. Here are 60 UEs distributed around 10 SBSs, and each SBS posses 5 channels and 10 GHz computation resources. A good convergence speed can be observed from Fig.4. First, the proposed algorithm converges in almost 50 iterations, which is effective. Second, the performance obtained by the proposed algorithm is satisfying, and the gap between the proposed algorithm with BFS is not that big.

V. Conclusion

In this paper, the joint load management of resource allocation problem in an EH powered SCN with MEC is investigated. To manage the load and match the traffic load with the energy of the SBSs, CRE is adopted. First, we formulate the main problem, and the goal is to maximize the number of offloading UEs. To solve this problem, we divided it into three subproblems and proposed a distributed three step iteration algorithm. Finally, the convergence of the proposed algorithm is studied by simulation, and the performance of the proposed algorithm is verified by comparing with the other baseline algorithms. In our future work, we will consider a power control scheme and take account of the energy flow in the time horizon.

ACKNOWLEDGEMENT

This work is jointly supported by the National Natural Science Foundation of China (Grant No.61671088), the National Natural Science Foundation of China for the Young (Grant No.61501047), and the National Natural Science Foundation of China (Grant No.61771070).

References

[1] C. Index, "Cisco visual networking index: Global mobile data traffic forecast update, 2016-2021," *White Paper*, pp. C11–52 086, Feb 2017.

[2] M. R. Palacin, "Recent advances in rechargeable battery materials: a chemists perspective," *Chemical Society Reviews*, vol. 38, no. 9, pp. 2565–2575, 2009.

[3] X. Chen, L. Jiao, W. Li, and X. Fu, "Efficient multi-user computation offloading for mobile-edge cloud computing," *IEEE/ACM Transactions on Networking*, vol. 24, no. 5, pp. 2795–2808, October 2016.

[4] K. Zhang, Y. Mao, S. Leng, Q. Zhao, L. Li, X. Peng, L. Pan, S. Maharjan, and Y. Zhang, "Energy-efficient offloading for mobile edge computing in 5g heterogeneous networks," *IEEE Access*, vol. 4, pp. 5896–5907, 2016.

[5] S. Sardellitti, G. Scutari, and S. Barbarossa, "Joint optimization of radio and computational resources for multicell mobile-edge computing," *IEEE Transactions on Signal and Information Processing over Networks*, vol. 1, no. 2, pp. 89–103, June 2015.

[6] C. Wang, F. R. Yu, C. Liang, Q. Chen, and L. Tang, "Joint computation offloading and interference management in wireless cellular networks with mobile edge computing," *IEEE Transactions on Vehicular Technology*, vol. 66, no. 8, pp. 7432–7445, Aug 2017.

[7] Y. Zhang, Y. Xu, Y. Sun, Q. Wu, and K. Yao, "Energy efficiency of small cell networks: Metrics, methods and market," *IEEE Access*, vol. 5, pp. 5965–5971, Apr 2017.

[8] L. Chen, F. R. Yu, H. Ji, B. Rong, X. Li, and V. C. M. Leung, "Green full-duplex self-backhaul and energy harvesting small cell networks with massive mimo," *IEEE Journal on Selected Areas in Communications*, vol. 34, no. 12, pp. 3709–3724, Dec 2016.

[9] O. Mu?oz, A. Pascual-Iserte, and J. Vidal, "Optimization of radio and computational resources for energy efficiency in latency-constrained application offloading," *IEEE Transactions on Vehicular Technology*, vol. 64, no. 10, pp. 4738–4755, Oct 2015.

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

[11] P. S. Yu, J. Lee, T. Q. S. Quek, and Y. W. P. Hong, "Traffic offloading in heterogeneous networks with energy harvesting personal cells-network throughput and energy efficiency," *IEEE Transactions on Wireless Communications*, vol. 15, no. 2, pp. 1146–1161, Feb 2016.

[12] V. Cerone and F. D. Croce, "A continuous knapsack problem formulation for the robustness analysis of a polytope of polynomials," in *American Control Conference, Proceedings of the 1995*, vol. 5, Jun 1995, pp. 3856–3858 vol.5.

[13] L. Tan, Z. Feng, W. Li, Z. Jing, and T. A. Gulliver, "Graph coloring based spectrum allocation for femtocell downlink interference mitigation," in *2011 IEEE Wireless Communications and Networking Conference*, March 2011, pp. 1248–1252.