PAPER • **OPEN ACCESS**

An Efficient Offloading Scheme For MEC System Considering Delay and Energy Consumption

To cite this article: Yanhua Sun *et al* 2018 *J. Phys.: Conf. Ser.* **960** 012002

An Efficient Offloading Scheme For MEC System Considering Delay and Energy Consumption

Yanhua Sun[1,2,3,a], Zhe Hao[1,2,3,b] and Yanhua Zhang[1,2,3]

[1]Beijing Advanced Innovation Center for Future Internet Technology, Beijing University of Technology, No 100, Pingleyuan, Beijing, China
[2]Beijing Laboratory of Advanced Information Networks, Beijing, China
[3]College of Information and Communications, Beijing University of Technology, No 100, Pingleyuan, Beijing, China

Email: [a]sunyanhua@bjut.edu.cn, [b]zoeahao@qq.com

Abstract. With the increasing numbers of mobile devices, mobile edge computing (MEC) which provides cloud computing capabilities proximate to mobile devices in 5G networks has been envisioned as a promising paradigm to enhance users experience. In this paper, we investigate a joint consideration of delay and energy consumption offloading scheme (JCDE) for MEC system in 5G heterogeneous networks. An optimization is formulated to minimize the delay as well as energy consumption of the offloading system, which the delay and energy consumption of transmitting and calculating tasks are taken into account. We adopt an iterative greedy algorithm to solve the optimization problem. Furthermore, simulations were carried out to validate the utility and effectiveness of our proposed scheme. The effect of parameter variations on the system is analysed as well. Numerical results demonstrate delay and energy efficiency promotion of our proposed scheme compared with another paper's scheme.

1. Introduction

In the foreseeable future 5G network, some applications require low latency computing but result in high energy consumption [1,2,3,4]. However, the computing capabilities and battery power of mobile devices (MD)are limited. These factors lead to conflicts between the applications and smart mobile devices, as well as, bring unprecedented challenges. Consequently, the idea of making the radio access network in close proximity to mobile devices that possess bandwidth and latency-critical data is becoming the consensus of the industry, and the MEC, which comes with the idea, has become the key technology to improve the users experience in 5G network [5]. MEC architecture allows MD to offload their computation-intensive tasks to MEC server via wireless cellular network [6].

It is well known that fog computing has been proposed to deploy computing resources closer to the terminals [7,8]. In recent years, studies on MEC architecture have become the hot spot gradually. In [9], a dynamic service migration problem in MEC were studied by authors. In [10], an analytical model for MEC was presented. In [11], a virtual network embedding problems were investigated, and network virtualization in the context of MEC networks were proposed by authors. In [12], an optimization problem to minimize the energy consumption of a MEC offloading system is researched by authors. In [13], a computation task scheduling policy of MEC offloading system is studied.

In this paper, a joint delay and energy efficient offloading scheme (JCDE) for multiuser MEC system is proposed, which takes into account both the latency demands and energy requirements of each MD in the system. Then, an optimization problem is formulated, in order to solve it, an iterative greedy algorithm is adopted. Numerical results show that proposed JCDE scenario achieves better performance compare with EECO scheme [12] and stochastic offloading scheme. Moreover, the impacts of different CPU capabilities of MEC server and MD on the offloading decisions are analyzed. Finally, the effects of MD's requirements on offloading assignments are investigated. The rest of the paper is organized as follows. The system model is introduced in Section2. The delay and energy efficient optimization problem is formulated in Section3. The JCDE is described in Section4. Then the numerical results are presented in Section5 and summary in Section6.

2. System model

As illustrated in figure 1, MD offload their tasks to MEC server through a 5G heterogeneous network. There is a Macro base station (MBS) equipped with a MEC server, besides there is a Small base station (SBS) whose service area is overlaid by that of the MBS. It is worth mentioning that MEC server has the ability to handle multiple tasks simultaneously.

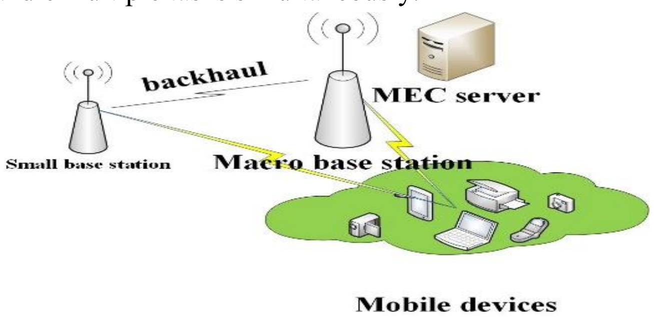

Figure 1. MEC system framework

We consider multiple MD which is denoted as $\mathscr{N} = \{1, 2, ..., N\}$. Each MD has a delay sensitive task or computation-intensive task. Each task can be described in three parameters as $T_i = \{I_i, O_i, c_i, t_i^{\max}\}$, $i \in \mathscr{N}$. I_i is the size of computational task (in bits), O_i is the size of output data which is computed by MEC server (in bits). c_i denotes the numbers of CPU cycles required for computing this task. For each MD i its task can either be executed locally on itself or accomplished on MEC server via offloading. $t_i^{\max}$ is the maximum latency requirement by the computational task.

3. Problem formulation

3.1. Key factors

In this system, some MD are running some applications that are sensitive to the delay and others that hope to save more energy. Considering different needs of different MD, we propose different weighting factors φ_i^T and φ_i^E for different MD in order to provide a better modeling flexibility and meet personalized user demands [8]. The former denotes latency factor and the latter denotes energy factor. $\varphi_i^T + \varphi_i^E = 1$ is satisfied. For MD can offload tasks to MEC server in two ways, $a_{i,j} = \{0,1\}$ is introduced as a task offloading decision indicator. Here $i \in \mathscr{N}$, when $a_{i,j} = 1$, the MD i selects mode j to process the task T_i. Otherwise, $a_{i,j} = 0$. $j = 1, 2, 3$ denotes the chosen modes. Specifically, $j = 1$ denotes the mode where the computational task T_i is executed locally, $j = 2$ corresponds to the mode where the task T_i is offloaded to MBS directly, $j = 3$ denotes the mode where the computational task is offloaded to MBS via the relay SBS. Thus, $a_{i,1} = 1$ represents that the MD i selects local computation. Similarly, $a_{i,2} = 1$ and $a_{i,3} = 1$ denote that MD i offload computational tasks to MEC server directly and via the relay SBS respectively.

3.2. Local problem formulation

Let F_i^L be the computational capability of MD i (in CPU cycles per second), different MD have different computational capabilities. Let δ_i^L be the energy consumption for one CPU cycle of MD. Thus, computational time of executing T_i locally is given as

$$t_{i,ex}^L = c_i / F_i^L \tag{1}$$

The energy consumption is calculated as

$$e_i^L = c_i \delta_i^L \tag{2}$$

Since there is no other time or energy consumption in the local execution, (1) represents the total delay of MD i accomplishing the task T_i locally. (2) represents the total energy consumption of MD i accomplishing T_i locally as well.

3.3. Offloading problem formulation

In our framework, a multi-user OFDMA in 5G heterogeneous networks is considered, the channels between the MD accessing the same base station are orthogonal to each other, therefore, there is only interference between the signals transmitted to the MBS and those to the SBS. The power consumption of the backhaul between the MBS and the SBS is neglected for this backhaul is shared with other communication infrastructures according to [12]. The backhaul transmission time is expressed as a proportional to the size of the data, and the coefficient is denoted by β.

When the MD i accesses the MBS, the uplink transmission rate can be given as

$$r_{i,u}^M = W \log_2 \left(1 + \frac{P_i^M g_i^M}{I_i^S + n_0} \right) \tag{3}$$

Where W denotes the bandwidth (in Hz). P_i^M denotes the power of MD i transmitting data to MBS, it can be controlled by MBS according to power control mechanism [14]. g_i^M denotes the channel gain between the MD i and the MBS. I_i^S denotes the interference at the MBS which is caused by other MD's uplink transmission to the SBS. n_0 denotes the background noise power.

Similar to the above, the MD i accesses the SBS, the uplink transmission rate is given as

$$r_{i,u}^S = W \log_2 \left(1 + \frac{P_i^S g_i^S}{I_i^M + n_0} \right) \tag{4}$$

Let F_i^S be the computational capability of MEC server and it is a constant for every task. Let δ_i^S be the energy cost of the MEC server for one unit CPU cycle, we assume $\delta_i^S < \delta_i^L$ on the basis of [12]. For MD i offloading T_i to MEC server directly, uplink transmission time is given as

$$t_{i,u}^M = I_i / r_{i,u}^M \tag{5}$$

Computing executing time is calculated as

$$t_{i,ex}^M = c_i / F_i^S \tag{6}$$

Downlink transmission time is calculated as

$$t_{i,d}^M = O_i / r_{i,d}^M \tag{7}$$

$r_{i,d}^M$ is the downlink transmission rate. For the size of output data O_i is much smaller than that of input data I_i [13]. $t_{i,d}^M$ is neglected ($t_{i,d}^M \approx 0$). Thus, (5) and (6) are the total time delay of MD i offloading T_i to MEC server through MBS directly.

Energy consumption of MD i offloading task to MBS directly includes two parts: uplink transmission energy, computational energy (ignoring downlink energy due to aforementioned reason similarly). They can be given respectively as

$$e_{i,u}^M = P_i^M I_i / r_{i,u}^M \tag{8}$$

$$e_{i,ex}^M = c_i \delta_i^S \tag{9}$$

According to the above, the total time and energy consumption of MD i offloading T_i to MEC server through MBS directly can be given as

$$t_i^M = t_{i,u}^M + t_{i,ex}^M \tag{10}$$

$$e_i^M = e_{i,u}^M + e_{i,ex}^M \tag{11}$$

Similarly, the total time consumption for the case where MD i selects to offload task through the SBS can be given as

$$t_i^S = t_{i,u}^S + t_i^{backhaul} + t_{i,ex}^S \tag{12}$$

Where $t_i^{backhaul}$ is the backhaul delay, calculated as

$$t_i^{backhaul} = I_i \beta \tag{13}$$

$t_{i,ex}^S = t_{i,ex}^M$ because computational execution time by MEC server for the same task is the same. Thus according to (5), (6), (8), (9), (10)and (11) the total duration of MD i selects offloading task through SS can be calculated as

$$t_i^S = I_i / r_{i,u}^S + I_i \beta + c_i / F_i^S \tag{14}$$

The total energy consumption can be calculated similarly as

$$e_i^S = P_i^S I_i / r_{i,u}^S + c_i \delta_i^S \tag{15}$$

Let introduce the task offloading decision indicator (3) and (4) can be rewritten as

$$r_{i,u}^M = W \log_2 \left(1 + \frac{P_i^M g_i^M}{\sum_{l=1, l \neq i}^{N} a_{l,3} P_l^S g_l^M + n_0}\right) \tag{16}$$

$$r_{i,u}^S = W \log_2 \left(1 + \frac{P_i^S g_i^S}{\sum_{l=1, l \neq i}^{N} a_{l,2} P_l^M g_l^S + n_0}\right) \tag{17}$$

3.4. System problem formulation

In summary, considering $a_{i,j}$, φ_i^T, φ_i^E, (1), (2), (5), (6), (8), (9), (10), (11), (12), (13), (14), (15), (16) and (17) the optimization problem is mathematically formulated to minimize the total time and energy consumption

$$\begin{aligned}
\min \sum_{i=1}^{N} \Bigg\{ & \varphi_i^T \Bigg[a_{i,1} \frac{c_i}{F_i^L} + a_{i,2} \Bigg(\frac{I_i}{W \log_2 \left(1 + \frac{P_i^M g_i^M}{\sum_{l=1, l \neq i}^{N} a_{l,3} P_l^S g_l^M + n_0}\right)} + \frac{c_i}{F_i^S} \Bigg) \\
& + a_{i,3} \Bigg(\frac{I_i}{W \log_2 \left(1 + \frac{P_i^S g_i^S}{\sum_{l=1, l \neq i}^{N} a_{l,2} P_l^M g_l^S + n_0}\right)} + I_i \beta + \frac{c_i}{F_i^S} \Bigg) \Bigg] \\
& + \varphi_i^E \Bigg[a_{i,1} c_i \delta_i^L + a_{i,2} \Bigg(\frac{P_i^M I_i}{W \log_2 \left(1 + \frac{P_i^M g_i^M}{\sum_{l=1, l \neq i}^{N} a_{l,3} P_l^S g_l^M + n_0}\right)} + c_i \delta_i^S \Bigg) + a_{i,3} \Bigg(\frac{P_i^S I_i}{W \log_2 \left(1 + \frac{P_i^S g_i^S}{\sum_{l=1, l \neq i}^{N} a_{l,2} P_l^M g_l^S + n_0}\right)} + c_i \delta_i^S \Bigg) \Bigg] \Bigg\}
\end{aligned}$$

$$
\begin{aligned}
&st. \\
&t_{i,ex}^{L} \le t_i^{\max}, t_i^{M} \le t_i^{\max}, t_i^{S} \le t_i^{\max} \\
&r_i^{M} \geqslant \frac{I_i}{t_i^{\max} - c_i / F_i^{s}}, r_i^{S} \geqslant \frac{I_i}{t_i^{\max} - ci / F_i^{S} - I_i \beta} \\
&a_{i,1} + a_{i,2} + a_{i,3} = 1 \\
&\varphi_i^{T} + \varphi_i^{E} = 1
\end{aligned}
\tag{18}
$$

4. JCDE algorithm

In this section, the details of JCDE scheme are given below:

1: **Initialization** of MD $i = \{1, 2, ..., N\}$, T_i, local mode $a_{i,1}$=0, MBS mode $a_{i,2}$=0, SBS mode $a_{i,3}$ =0, bandwidth, background noise, coefficient etc.
2: **repeat** for each MD in parallel (in a simulation, each MD is chosen as the first choice):
3: **while** three modes of devices $a_{i,1}$+ $a_{i,2}$+ $a_{i,3} \neq N$ **Do**
4: {choose a MD
5: compute the total consumption in three different modes
6: compare three results of this MD and choose the minimum consumption mode
7: choose another MD execute step4 under former MD's influence then go to 6}
8: **end while**
9: compare all results of parallel total consumptions
10: choose the minimum result and record its chosen mode $a_{i,1}$, $a_{i,2}$, $a_{i,3}$
11: **end**

5. Numerical results

In this section, we conduct simulations to validate the utility and effectiveness of proposed JCDE algorithm. Plenty of users are randomly distributed in a cell with a radius of 1000m, a MBS and a SBS are in this area. Bandwidth is 2MHz, the CPU capability of MEC server is {2,3,4,5} GHz/s, N= {10,20,30,40,50,60,70,80,90,100}, backhaul delay coefficient is 0.0001s/KB, delay constraint is 0.5s, CPU cycles required by computing a task are randomly distributed between 0.1 and 1.1GHz.

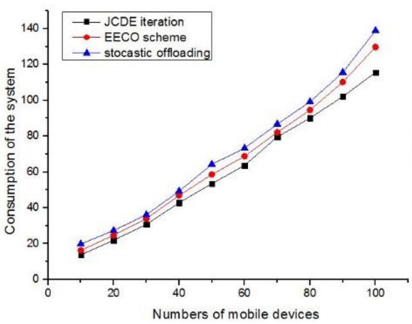

Figure 2. Consumption of system with different schemes

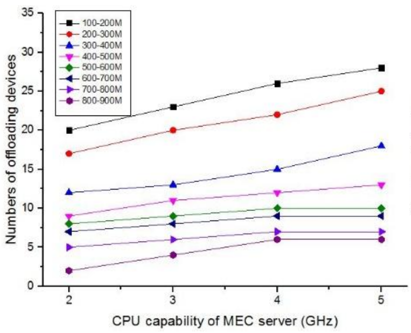

Figure 3. Influence of both CPU computing capabilities

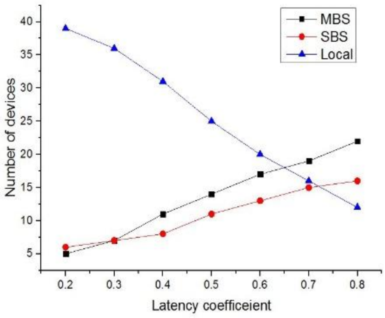

Figure 4. Influence of latency and energy coefficients

Figure 2 indicates the total consumption of the system with three different offloading schemes in terms of MD numbers. It is obvious that, regardless of MD numbers, consumptions of system in our JCDE scheme are always less than those in both EECO scheme [12] and stochastic offloading scheme. The simulation results validate the utility as well as effectiveness of our proposed scheme.

Figure 3 (N=50 in the simulations) shows that with the computational capability of MEC server CPU increasing and the computational capability of local CPU reducing, the number of MD which select to offload tasks is increases. Even though the computational capability of local CPU is strong, the computational capability MEC server CPU is weak, few MD still offload their tasks because they are close to the MBS and SBS and their tasks are delay-sensitive.

Figure 4 (the computational capability of MEC server CPU is 5GHz/s, N=50) shows latency and energy coefficient can affect offloading selections. It means the higher delay requirements of MD, the more assignments to offload, this is because the strong computing capability of MEC server. When the delay requirements of MD are low, most MD choose to perform computational tasks locally due to the large amount of energy consumed during offloading process.

6. Summary

In this paper, an optimization problem to minimize system consumption considering delay and energy for MEC system is formulated. To solve it, we design a JCDE algorithm and carry out simulations to validate the utility and effectiveness of proposed scheme. Numerical results demonstrate delay and energy efficiency promotion of our proposed scheme compared with EECO scheme and stochastic offloading scheme. In our future work we will study on NP-hard and more complex system model.

Acknowledgement

This work is jointly supported by the National Natural Science Foundation of China under Grant 61571021 and 61372089.

References

[1] Wang S and Dey S, Adaptive mobile cloud computing to enable rich mobile multimedia applications, *IEEE Trans. Multimedia,* vol 15, 4(2013), 870–883.
[2] http://www.ngmn.org/5g-white-paper.html.
[3] Shen J, Tan H,Wang J and Lee S, A novel routing protocol providing good transmission reliability in underwater sensor networks. *J. Internet Technol*, vol 16, 1(2015), 171-178.
[4] Pan Z, Zhang Y, Kwong S, Efficient motion and disparity estimation optimization for low complexity multiview video coding, *IEEE Trans. Broadcast*, vol 61, 2(2015), 166-176.
[5] 2016 Mobile Edge Computing– A Key Technology Towards 5G[EB/OL].
[6] ETSI 2014 *ETSI White Paper.*
[7] Zhanikeev M, A cloud visitation platform to facilitate cloud federation and fog computing, *computer,* vol 48, 5(2015), 80-83
[8] Sarkar S, Chatterjee S, Misra S, Assessment of the suitability of fog computing in the context of Internet of things, *IEEE Trans. Cloud Computing,* 99(2015).
[9] Wang S, Urgaonkar R, Zafer M, He T, Chan K, Leung K K, Dynamic service migration in mobile edge-clouds, *Proc. IFIP Netw.Conf*, 5(2015), 1–9
[10] Taleb T, Ksentini A, An analytical model for follow me cloud, *Proc. IEEE Global Commun Conf*, 12(2013), 1291–1296.
[11] Beck M T, Maier M, Mobile edge computing: Challenges for future virtual network embedding algorithms, *Proc. Adv. Eng. Comput.Appl.Sci*, 8(2014), 65–70.
[12] Zhang K, Mao Y M, Energy-efficient offloading for mobile edge computing in 5G heterogeneous networks, *IEEE Access*, vol 4(2016).
[13] Juan Liu, Yuyi Mao, Jun Zhang, and Khaled B. Letaief, Delay-optimal computation task scheduling for mobile-edge computing systems, *IEEE International Symposium on Information Theory,* 1451-1455
[14] Liu K W, Yang Z T, Xiong H X, Aplication Rsearch of Cmputers, vol 34(2017).