An Offloading Strategy for Multi-User Energy Consumption Optimization in Multi-MEC Scene

Zhi Li [1,2] and Qi Zhu [1,2,*]

[1] Jiangsu Key Laboratory of Wireless Communications, Nanjing University of Posts and Telecommunications, Nanjing 210003, China; 18852853125@163.com
[2] Engineering Research Center of Health Service System Based on Ubiquitous Wireless Networks, Nanjing University of Posts and Telecommunications, Nanjing 210003, China
*Correspondence: zhuqi@njupt.edu.cn

Abstract

Mobile edge computing (MEC) is capable of providing services to smart devices nearby through radio access networks and thus improving service experience of users. In this paper, an offloading strategy for the joint optimization of computing and communication resources in multi-user and multi-MEC overlapping scene was proposed. In addition, under the condition that wireless transmission resources and MEC computing resources were limited and task completion delay was within the maximum tolerance time, the optimization problem of minimizing energy consumption of all users was created, which was then further divided into two subproblems, i.e. offloading strategy and resource allocation. These two subproblems were then solved by the game theory and Lagrangian function to obtain the optimal task offloading strategy and resource allocation plan, and the Nash equilibrium of user offloading strategy games and convex optimization of resource allocation were proved. The simulation results showed that the proposed algorithm could effectively reduce the energy consumption of users.

Keywords: Mobile-edge computing, Offload, Nash equilibrium, Convex optimization

1. Introduction

Various novel intelligent services keep emerging with the coming of 5G network, so hundreds of millions of intelligent terminal devices have a lot of tasks to deal with, but they are insufficient to fulfill all these tasks due to their limited resources. Mobile edge computing [1] enables terminal devices to offload computing tasks onto edge servers through radio access networks, which shortens the distance between server and user, thereby greatly reducing the task completion time and user energy consumption [2]. Task offloading strategy and resource allocation have become the research focus due to its direct influence on system's performance [3].

The complex intelligent services have higher demands for the energy consumption of terminal devices [4]. The optimal offloading strategy and resource allocation plan to minimize the energy consumption in the multi-user and single MEC scene was proposed in [5], and the differences in energy consumption between transmission modes TDMA and FDMA were compared. But it only considers one MEC in that paper, this does not conform to the actual scene. Meanwhile, the algorithm proposed in that paper can be optimized to make the total energy less than before. Paper [6] used the alternating direction multiplier method to achieve the satisfaction of minimum delay while minimizing the energy consumption. Paper [7] introduced the concept of task cache, created the problem of jointly optimizing task caching and offloading on edge servers, and proposed the alternative and iterative algorithm of task caching and offloading to solve the problem. A game theory based algorithm was proposed in [8] to jointly optimize channel bandwidth and MEC computing resources, so as to minimize total time and energy consumption. Paper [9] put forward a mobile edge computing framework with the functions of multi-user computing task distribution and transmission scheduling to maximize the social networking welfare (the whole network performance) and achieve the game equilibrium between mobile users. Paper [10] considered the existing challenges of establishing a multi-hop network based on the application protocol for instant messaging, and proposed a heuristic method and an iterative task allocation algorithm to optimize the collaborative network of computing task allocation in the multi-hop device. In [11], the optimal task offloading strategy and resource allocation plan were studied, an effective bisection method was proposed to solve the strong coupling problem of resource allocation, and Gibbs sampling algorithm was raised to obtain the optimal distribution decision. The caching strategy was added to the offloading model in [12], and the optimal edge computing offloading method with cache enhancement plan was proposed to reduce the overall user delay and energy consumption. Paper [13] mapped the user task upload to a queuing model to describe the network dynamics, and used the game theory method to seek the best plan for computing distribution and transmission scheduling. Although most of the above studies have taken the resource allocation and offloading strategy into consideration, they only researched on the situation that single user accessed to a single MEC scene at the same time, and there is lack of studies on multi-MEC scene accessed by single user at the same time.

If a user can access to the networks of multiple operators in the overlapping area of wireless networks of multiple operators, he/she can choose to offload the task onto the MEC of different operators. In this paper, the multi-user and multi-MEC offloading strategy was studied, together with which, computing and communication resources were jointly optimized to minimize the total energy consumption of user terminal devices. To summarize, our contributions are as follows:

- Under the condition that the wireless transmission resources and MEC computing resources were constrained and the task completion delay was within the maximum tolerance time, the optimization problem of minimizing the energy consumption of all users in the multi-user and multi-MEC overlapping scene was created;
- Because the proposed problem was nonconvex and thus hard to be solved, it was divided into two subproblems: offloading strategy and resource allocation. The offloading strategy was solved by using the game theory, and its Nash equilibrium (NE) was proved. Meanwhile, the resource allocation was proved to be a convex optimization problem, and its optimal value was obtained through Lagrangian method;
- The two subproblems were jointly iterated to obtain the optimal solution to minimize the total user energy consumption. The simulation results showed that the algorithm proposed in this paper could reduce the total user energy consumption to a lower level compared to [5]. For example, when there are 15 users, the total user energy consumption is reduced by about 12.1%.

The rest of this paper is arranged as follows: Section 2 presents the system model of the multi-user and multi-edge server and states the problems. The optimization problem is analyzed and an algorithm for joint optimization of user task offloading strategy and resource allocation is proposed in Section 3. Section 4 performs the simulation and performance analysis of the algorithm. This paper is summarized in Section 5.

2. System Model

The system model is shown in **Fig. 1**. Many smart mobile terminals(SMTs) can choose to access to the networks of two operators at the same time in the WiFi overlapping area of two operators. There are M user devices and N WiFi access points (WAPs) randomly distributed in the overlapping area, and each operator has an edge server. Assuming that K WAPs are connected to MEC1, and the rest WAPs are connected to MEC2, users can process their computing tasks locally or offload these tasks onto the edge server through WAP. The computing tasks that users m ($m = 1, 2, \mathsf{L} \ M$) need to process are $U_m = \{ f_{m,0}, t_m^{\max}, C_m, D_m \}$, where $f_{m,0}$ is the local computing resources of user devices(UDs), $t_m^{\max}$ is the maximum tolerance time of users, C_m is the size of computing resources required by users to process tasks, and D_m is the data size.
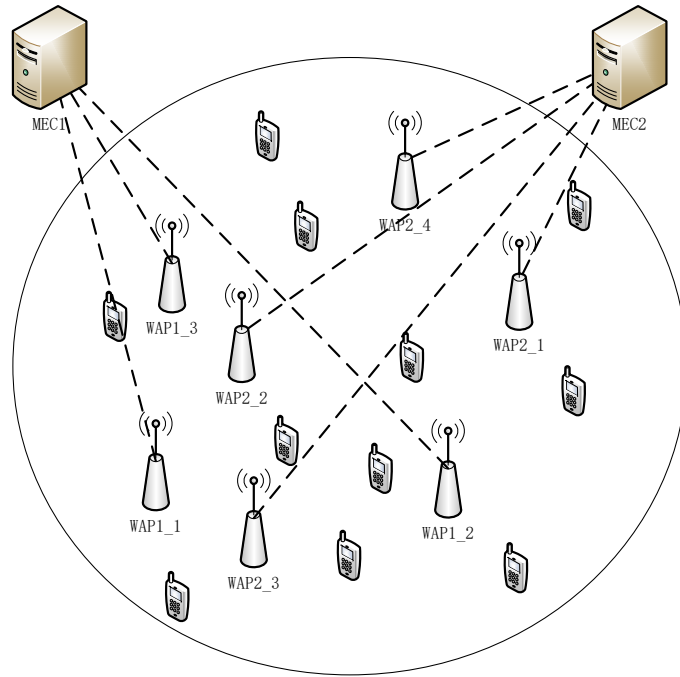
Fig. 1. System Model

2.1 User Local Processing Time and Energy Consumption

Each user can choose whether to offload tasks onto MEC for processing. If MEC has already carried a lot of load or because of the far distance between the user m and WAP, the transmission energy consumption is too high and the transmission time exceeds the user's maximum tolerance time for the task completion, the user m chooses to process the tasks locally, and the time it takes is [14]:

$$t_{m,local} = \frac{C_m}{f_{m,0}} \tag{1}$$

and the energy it consumes is:

$$E_{m,0} = k(f_{m,0})^3 t_{m,local} \tag{2}$$

where $k=10^{-24}$ is a constant. Substituting Formula (1) into Formula (2), we obtain:

$$E_{m,0} = k(f_{m,0})^2 C_m \tag{3}$$

2.2 Task Offloading Processing Time and Energy Consumption

The user offloads the data through frequency division multiplexing method. The bandwidth of the access point $i(i \in [1, N])$ is B^i and the bandwidth allocated to each user is $B_{m,i}$. Then, the rate of the user m offloading the task data through the access point i is:

$$r_{m,i} = B_{m,i} \log_2(1 + \frac{p_m g_{m,i}}{\delta^2}) \tag{4}$$

where p_m is the user transmission power, and δ^2 is the thermal noise power. The time for the user m to complete the processing of offloaded tasks includes the offloading communication time and computing time, which is expressed as:

$$t_{m,i} = \frac{C_m}{f_{m,i}} + \frac{D_m}{r_{m,i}} \qquad (i \in F_x) \tag{5}$$

where $i \in F_x$ means that WAPi can connect to MEC1(F_1) or MEC2(F_2), $\frac{C_m}{f_{m,i}}$ is the time that MEC takes to complete tasks, $f_{m,i}(i \in F_x, x \in [1,2])$ is the computing resources allocated to the user by MEC1 or MEC2, and $\frac{D_m}{r_{m,i}}$ is the communication time required for task uploading. After finishing processing the tasks, the edge server will send the processed data back to the user, which takes T_{back}. Compared with other time, T_{back} has little impact on the overall time [12], so it is ignored in this paper. Energy consumed by the user m for task offloading is:

$$E_m^i = p_m \frac{D_m}{r_{m,i}} + p_m^c \frac{C_m}{f_{m,i}} \qquad (i \in F_x) \tag{6}$$

where p_m is the transmission power, and p_m^c is the standby power for the user to wait for the completion of offloaded task processing. The total energy consumed by all users is:

$$E = \sum_{i=0}^{N} \sum_{m=1}^{M} I(a_m = i) * E_{m,i} \tag{7}$$

Each user m can choose to offload tasks through WAP $i(i \in [1,N])$ or process the tasks locally. All users' offloading decisions $a_m(a_m = i \in [0,N])$ constitute the offloading decision $A(A = \{a_1, a_2...a_m\})$, where $a_m = 0$ represents the local processing of tasks. In the summation process, $I(a_m = i)$ is 1 if WAPi is a_m, and otherwise it is 0. The goal of this paper is to minimize the total energy consumption E by jointly optimizing the user offloading decision A, bandwidth allocation $B_{m,i}$ and MEC computing resources $f_{m,x}$. The above optimization problem can be expressed as:

$$P1: \quad \min \quad E(A,F,B) = \sum_{i=0}^{N} \sum_{m=1}^{M} I(a_m = i) * E_{m,i}$$

$$s.t. \quad C1: f_{m,i} \geq 0, \qquad \forall m \in M, i \in F_x$$

$$C2: t_{m,i} \leq t_m^{\max}, \qquad \forall m \in M, \forall i \in (1,2...N)$$

$$C3: \sum_{m \in M} B_{m,i} \leq B^i, \quad \forall m \in M, \forall i \in (1,2...N) \qquad (8)$$

$$C4: \sum_{i=0}^{N} I(a_m = i) = 1, \forall m \in M, \forall i \in (0,1...N)$$

$$C5: \sum_{i \in F_x} \sum_{m \in M} f_{m,i} \leq F_x, \forall m \in M, \forall i \in (1,2...N)(x \in (1,2))$$

where A is the user offloading set, restrictive condition C1 represents that the computing resources allocated to users cannot be negative, C2 represents that the processing time after task offloading should be less than the maximum tolerance time of users, C3 represents that the total bandwidth resources allocated to each user cannot be greater than the channel bandwidth, C4 represents that the users choose to process tasks locally or offload the tasks to MEC1 or MEC2 for processing through any WAP, and C5 represents that the MEC computing resources allocated to each user cannot be greater than the total MEC computing resources. The notations are given in **Table 1**.

Table 1. Lists of symbols

Symbols	Descriptions
F_1, F_2	MEC1's computational resource and MEC2's computational resource
$B_{\max}$	The max channel bandwidth of WAP
C_m	Needed computing resource to complete a task
D_m	Data size of a task
f_m	UD's local computational capacity
M	UD numbers
p_m	The transmission power
p_m^c	The maintaining power
δ^2	Gaussian thermal noise power

3. Optimization Solution

Since offloading decision A is a discontinuous variable, Problem (8) is a nonconvex optimization problem, to which it's hard to find the optimal solution. However, it can be found that all the restrictive conditions of pare only related to computing and communication resources except C4, which is related to offloading decision. Therefore, an algorithm for the joint optimization of task offloading strategy and resource allocation was proposed in this study. It decomposed the original problem into two subproblems, i.e. offloading decision and resource allocation, which were then solved separately to minimize user energy consumption. The offloading decision and resource allocation were carried out under the condition of given computing and communication resources, and the allocation of bandwidth and computing resources could be obtained by iterative solution when the offloading strategies of all users were known.

3.1 Offloading Strategy

Problem (8) can be decomposed into two subproblems: offloading strategy and resource allocation. Since only $C4$ is related to offloading strategy, the optimization problem of offloading strategy can be expressed as：

$$P2: \quad \min \quad E(A) = \sum_{i=0}^{N} I(a_m = i) * E_{m,i}$$

$$s.t. \quad C4: \sum_{i=0}^{N} I(a_m = i) = 1, \forall m \in M, \forall i \in (0,1...N) \tag{9}$$

Game theory is used in this paper to resolve the optimization problem of offloading strategy because it could effectively resolve the conflicts of interest of users and enable the decision-making mechanism to find the optimal set of strategies [15]. a_m represents the user offloading strategy, $A = (a_1, a_2...a_m)$ represents the user offloading strategy set, $a_{-m} = (a_1,..., a_{m-1}, a_{m+1},..., a_M)$ represents the SMT strategy set excluding m, the user's utility is defined as $E_m(a_m, a_{-m})$, and the game process can be defined as $\theta = \{M, A, E_m(a_m, a_{-m})\}$. In order to make the game have the optimal solution, a user offloading strategy set A that satisfies the Nash equilibrium had to be found, that is:

$$E_m(a'_m, a'_{-m}) \le E_m(a_m, a'_{-m}) \quad \forall a_m \in \{0,1,2...N\} \tag{10}$$

We only need to prove that the game process is a potential game problem to justify that θ convergence is tenable [14] because potential game process must have Nash equilibrium point. Since all users change their strategy always for the same purpose (to improve their utility), if the change of each user's utility can be mapped to a potential game process, it will make the potential game also monotonous. To testify that θ is a potential game, we need to prove

$$K(0, a_{-m}) - K(j, a_{-m}) = E(0, a_{-m}) - E(j, a_{-m}) \tag{11}$$

where function K is expressed as [18]：

$$K(0,a) = \sum_{m=1}^{M} (E_{n,0} - E_{n,s}) I(a_m = 0) + (1 - I(a_m = 0)) *$$
$$(\sum_{i=1}^{N} I(a_m = i) * E_{m,i} + \sum_{n=1,n \ne m}^{M} (E_{n,0} - E_{n,s})) \tag{12}$$

Substituting Formula (12) into Formula (11), we obtain:

$$K(0, a_{-m}) - K(j, a_{-m})$$
$$= k(f_{m,0})^2 C_m - (p_m \frac{D_m}{r_{m,j}} + p_m^c \frac{C_m}{f_{m,i}}) \tag{13}$$
$$= E(0, a_{-m}) - E(j, a_{-m})$$

$K(0,a_{-m})-K(j,a_{-m})=E(0,a_{-m})-E(j,a_{-m})$ is obtained from Formula (12),(13), testifying that θ is a potential game, which means there exists the Nash equilibrium that enables the game process to converge to an optimal value. In the context of multi-user decision-making, the game theory was used to update the users' offloading strategies until NE was obtained in this paper. After the offloading strategy is given, the computing and communication resources are optimized to achieve the minimum energy consumption of all users.

3.2 Resource allocation

The allocation of computing and communication resources have to be optimized to achieve the minimum energy consumption of all users after determining the user offloading strategy. As mentioned before, Problem (8) is decomposed into two subproblems: offloading strategy and resource allocation [16], while $C1, C2, C3, C5$ are related to resource allocation, so the optimization problem of resource allocation can be represented as:

$$P3: \quad \min \quad E(F,B)=\sum_{m=1}^{M} p_m^c \frac{C_m}{f_{m,i}} + p_m \frac{D_m}{r_{m,i}}$$
$$=\sum_{m=1}^{M} p_m^c \frac{C_m}{f_{m,i}} + p_m \frac{D_m}{B_{m,i}\log_2(1+\frac{p_m g_m^i}{\delta^2})} \quad (14)$$
$$s.t. \quad C1,C2,C3,C5$$

Problem P3 is a binary function, and whether it is concave or convex is proved at first. According to the properties of Hessian matrix, if the determinant of Hessian matrix with $f_{m,i}$ and $B_{m,i}$ as variables is greater than zero and positive definite, the function is convex and has a minimum value [17]. The Hessian matrix of Formula (14) is expressed as:

$$H(p_m^c \frac{C_m}{f_{m,i}} + p_m \frac{D_m}{r_{m,i}}) = \begin{bmatrix} \dfrac{\partial^2(p_m^c \frac{C_m}{f_{m,i}} + p_m \frac{D_m}{r_{m,i}})}{\partial^2(f_m^i)}, & \dfrac{\partial^2(p_m^c \frac{C_m}{f_{m,i}} + p_m \frac{D_m}{r_{m,i}})}{\partial f_m^i \partial B_{m,i}} \\ \dfrac{\partial^2(p_m^c \frac{C_m}{f_{m,i}} + p_m \frac{D_m}{r_{m,i}})}{\partial B_{m,i}\partial f_{m,i}}, & \dfrac{\partial^2(p_m^c \frac{C_m}{f_{m,i}} + p_m \frac{D_m}{r_{m,i}})}{\partial^2(B_{m,i})} \end{bmatrix} \quad (15)$$

Simplifying Formula (15), we obtain the determinant of Hessian matrix:

$$H(p_m^c \frac{C_m}{f_m} + p_m \frac{D_m}{r_{m,i}}) = \begin{vmatrix} \dfrac{2p_m^c C_m}{(f_{m,i})^3}, & 0 \\ 0, & \dfrac{2D_m p_m^c \lg 2}{(B_{m,i})^3 * \log_2(1+\frac{p_m g_{m,i}}{\delta^2})} \end{vmatrix} \quad (16)$$

According to Formula (16), $\dfrac{2p_m^c C_m}{(f_{m,i})^3} > 0$ and |H|>0, so the Hessian matrix is positive

definite, proving that Problem P3 is a concave function. There must be a minimum value since it is a convex optimization problem, which, therefore, was solved by a Lagrange function in this paper:

$$L(f,B,\lambda,\beta) = \sum_{i \in F_x} \sum_{m \in M} p_m^c \frac{C_m}{f_m^i} + p_m \frac{D_m}{r_{m,i}} + \beta(\frac{C_m}{f_{m,i}} + \frac{D_m}{r_{m,i}} - t_m^{\max}) + \lambda(\sum_{i \in F_x} \sum_{m \in M} f_{m,i} - F_x)$$

(17)

Where β and λ are Lagrange multipliers. Since Formula (14) is a convex optimization problem, Formula (17) must also be a convex optimization problem. Take the derivatives of four variables separately and make the derivatives to be 0, as shown below:

$$\frac{\partial L}{\partial f} = 0 \Rightarrow -p_m^c \frac{C_m}{(f_{m,i})^2} - \beta \frac{C_m}{(f_{m,i})^2} + \lambda = 0$$

$$\frac{\partial L}{\partial B} = 0 \Rightarrow -(\frac{(p_m + \beta)D_m}{(B_{m,i})^2 \log_2(1 + \frac{p_m g_{m,i}}{\delta^2})}) = 0$$

(18)

$$\frac{\partial L}{\partial \lambda} = 0 \Rightarrow f_{m,i} - F_x = 0$$

$$\frac{\partial L}{\partial \beta} = 0 \Rightarrow \frac{C_m}{f_{m,i}} + \frac{D_m}{r_{m,i}} - t_m^{\max} = 0$$

Setting up Formula (18), the optimal values of f_m^* and B_m^i are:

$$f_m^* = \sqrt{\frac{(p_m^c + \beta_m)C_m}{\lambda}}$$

(19)

$$B_m^i = \sqrt{\frac{p_m * D_m}{\lambda \log_2(1 + \frac{p_m g_{m,i}}{\delta^2})}}$$

(20)

f_m^* and B_m^i can be obtained by iteration according to Lagrange multiplier method. The update rule of multipliers λ and β is as follows:

$$\lambda(v+1) = \max\{0, \lambda(v) + \delta(v)\nabla\lambda(v)\}$$

(21)

$$\beta(v+1) = \max\{0, \beta(v) + \delta(v)\nabla\beta(v)\}$$

(22)

where v is the index of the number of iterations, $\nabla\beta(v) = \frac{C_m}{f_{m,i}} + \frac{D_m}{r_{m,i}} - t_m^{\max}$,

$\nabla\lambda(v) = \sum_{i \in F_x} \sum_{m \in M} f_{m,i} - F_x$, $\delta(v)$ is the number of iterations and $\delta = 1/(10^5 * v)$.

Algorithm 1 in **Table 2** can be used to solve Problem (13).

Table 2. Optimal Algorithm for Solving Problem (14)

---

1. Initialize $U_m = \{f_{m,0}, T_m, t_m^{\max}, C_m\}$, precision ε and the number of iterations $v_{\max}$

2. **while** $v < v_{\max}$ **do :**

3.   Computing $f_{m,*}, B_{m,i}(m \in M)$

4. **end**
5. According to (20)(21), updating Lagrange multiplier λ, β
6. **if** $\left\| \lambda(v+1) - \lambda(v) \right\| < \varepsilon, \left\| \beta(v+1) - \beta(v) \right\| < \varepsilon$:
7. $f_{m,*} = f_m(t)$, $B_{m,i} = B_{m,i}(t)$
8.   **break**
9. **else**:
10. $v = v + 1$
11. **end if**
12. Output: $f_{m,*}, B_{m,i}$

3.3 Energy Consumption Optimization Algorithm Steps

Problem P1 is solved by cyclic iteration of two subproblems in this paper. At first, assuming a given offloading strategy, Algorithm 1 is used to find the optimal resource allocation strategy to minimize the total energy consumption of users. Then, after obtaining the optimal resource allocation values, the offloading strategy of P1 is optimized by non-cooperative game method. Algorithm 2 in **Table 3** shows the specific steps: first, put the initial offloading strategy set A to 0 (namely all users do not offload tasks), and take the opportunity to update their own offloading strategy to make $a_m = i, (i \in [0, N])$; then successively compare the energy consumption of WAPs that are connected to MEC1 and MEC2, and select the WAP with the lowest energy consumption as the user's connection point; next, update the strategy set A after all users update their own offloading strategy for one round, and repeat this process until all users' energy consumption converges to a certain value to get the final strategy set.

In algorithm 2 in **Table 3**, the potential game needs $O(KN)$ calculations to obtain the offloading update set of SMTs at the each iteration. Moreover, in Algorithm 1 in **Table 2**, the subgradient projection method needs $O(\frac{1}{\varepsilon^2})$ iterations to converge. Therefore, the computation complexity is $O(KN \frac{1}{\varepsilon^2} * num)$, where num is the number of iterations of Algorithm 2.

Table 3. Joint optimization algorithm for resource allocation and offloading strategy

1.   Initialize $U_m = \{ f_{m,0}, T_m, t_m^{\max}, C_m \}$, $I(a_m = 0) \forall m \in M$
2.   **for** all SMTs($m \in M$) **do**
3.   **for** all WAPs($n \in [0, K]$) **do**
4.   According to algorithm in **Table 2**, computing $m's$ optimal $f_{m,i}$, $B_{m,i}$ and $E_{m,n}$
5.   **end for**
6.   Choose the lowest $E_{m,n} (n \in [0, K])$,marked the WAP as temp1
7.   **for** all WAPs($n \in [N - K, N]$) **do**

8.   According to algorithm in **Table 2**, computing $m's$ optimal $f_{m,i}$, $B_{m,i}$ and $E_{m,n}$

9.   **end for**

10.   Choose the lowest $E_{m,n}(n \in [N-K,N])$,marked the WAP as temp2

11.   **if** ($E(temp1) > E_{m,0}$ and $E(temp2) > E_{m,0}$)

12.   do not offload

13.   **else if** ($E(temp1) < E(temp2)$)

14. $a_m = 0 \rightarrow a_m = temp1$

15.   **else**

16. $a_m = 0 \rightarrow a_m = temp2$

17.   update A

18. **end** for

19. **Repeat** until convergence, get A

20. Output: $A, f_{m,i}(m \in M, i \in F_x, x \in (1,2))$, $B_{m,i}(m \in M, i \in N)$

---

4. Simulation Results and Analysis

In this paper, we use Matlab to simulate the proposed algorithm, and the simulation process is shown in **Fig. 1**. Users and WAPs are randomly distributed within an area with the radius of 100m. Some WAPs are connected to MEC1, while the rest are connected to MEC2. **Table 4** shows the specific simulation parameters.

Table 4. Simulation parameters

Parameters	Value
MEC computational resource F_1, F_2	{100,125,150}GHz/s
Channel bandwidth B_{max}	{3,5,7}MHz
Needed computing resource to complete a task C_m	$C_m \sim N(3,0.6)$ GHz
Data size of a task D_m	$D_m \sim N(350,84)$ KB
UD's local computational capacity f_m	[0.8,1.2]GHz/s
UD numbers M	[0,30]
Transmission power p_m	0.4w
Maintaining power p_m^c	0.05w

δ^2	-114dBm/MHz
t_m^{max}	[10,20]s

Fig. 2 shows the simulation results of the total energy consumption of different numbers of users changing with the number of iterations when $B_{max} = 3MHz$, $F_{max} = 100GHz/s$. It can be seen from the figure that a dozen times of iterations can achieve convergence generally, and the times of iteration to convergence gradually increases with the increase of the number of users. The fewer of users, the faster the convergence speed.
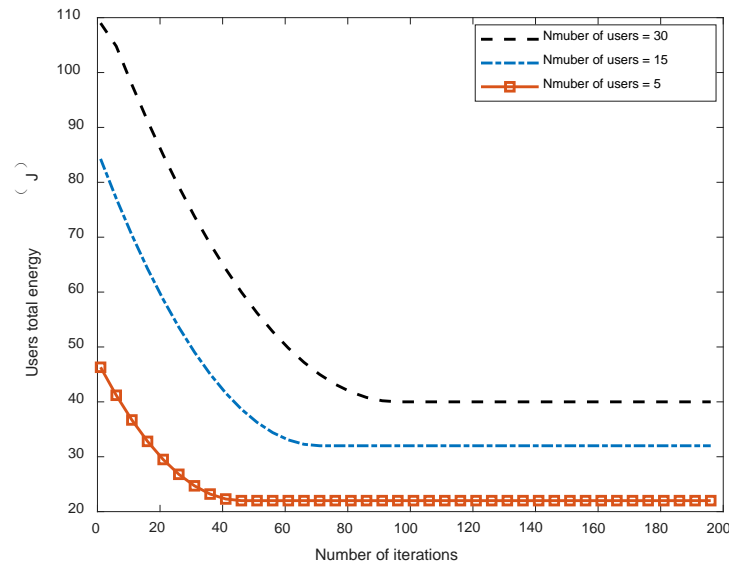
Fig. 2. Users' total energy varies with the number of iterations

Fig. 3 shows the simulation results of the total user energy consumption changing with the number of users when $B_{max} = 3MHz$, $F_{max} = 100GHz/s$, and compares the algorithm proposed in this paper with the algorithm in [5] and the local processing method. 5 randomly distributed WAPs connect to MEC1 and $N-5$ randomly distributed WAPs connect to MEC2, it does not optimize how UDs choose the best WAP to offload their tasks. Besides, it more considers the situation of a single MEC instead of multiple MECs, therefore, the scenario and problem presented in this paper is more authentic. It can be seen that the local processing method has the worst performance because the users' energy consumption rises linearly. It also can be concluded that the proposed algorithm is better than the algorithm in [5], and its advantages become more obvious with increase of the number of users. This is because the MEC accessed by users in [5] is fixed and cannot be selected, which will result in load unbalance between two operators and use efficiency. However, the proposed algorithm can choose a MEC to offload tasks through optimization decision, enabling two operators to share communication and computing resources, thereby improving the system performance.
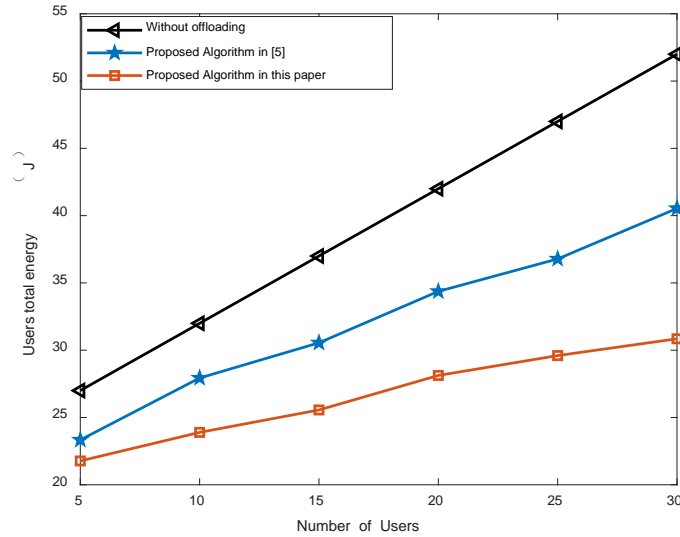
Fig. 3. Users' total energy varies with the number of users

The simulation results of user energy consumption changing with WAP bandwidth when $F = 100\mathrm{G}Hz$ are shown in **Fig. 4**, from which it can be seen that a wider WAP bandwidth will reduce the user energy consumption. This is because with the increase of allocable communication resources, the user transfer rate r_m^i becomes greater and the data transfer time $\dfrac{D_m}{r_m^i}$ gets smaller, so the standby energy consumption of the user end decreases. However, the advantage of wide bandwidth gradually weakens with the increase of the number of users because of the limitation of MEC's computing power.
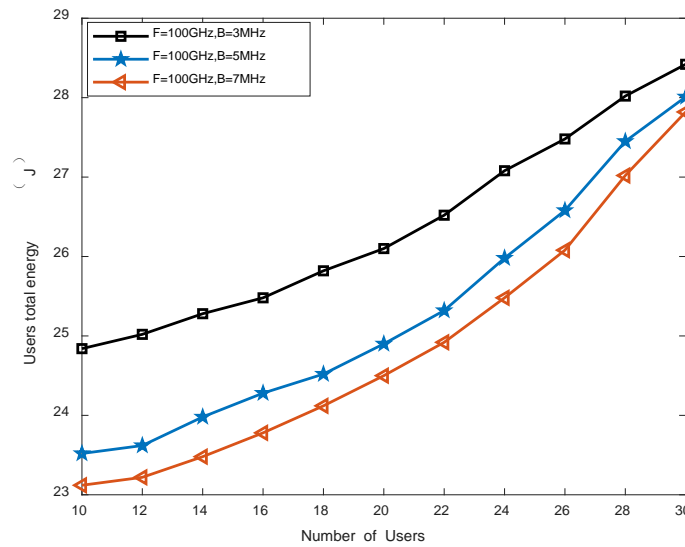

Fig. 4. Users' total energy varies with the Bandwidth

The simulation results of user energy consumption changing with the computing power of the edge server when $B = 3MHz$ are shown in **Fig. 5**, from which it can be seen that with the improvement of the computing power of the edge server, more users offload the tasks, and the energy consumption of users decreases. Besides, **Fig. 5** shows a greater decline trend in energy consumption than **Fig. 4**, which is due to the larger impact of computing power on user energy consumption and delay. However, the user energy consumption grows slightly when the computing resources of two MECs reach 150GHz, and the reason is that because the computing resources allocated to users are adequate, and bandwidth prevents the reduction of energy consumption.
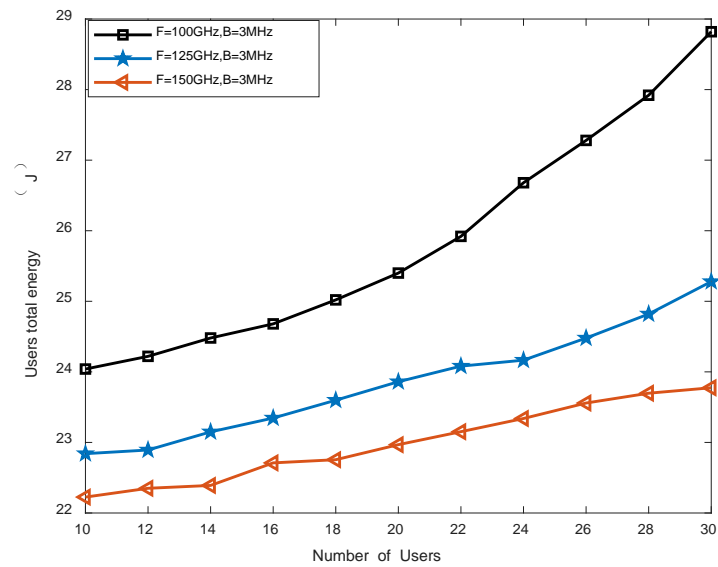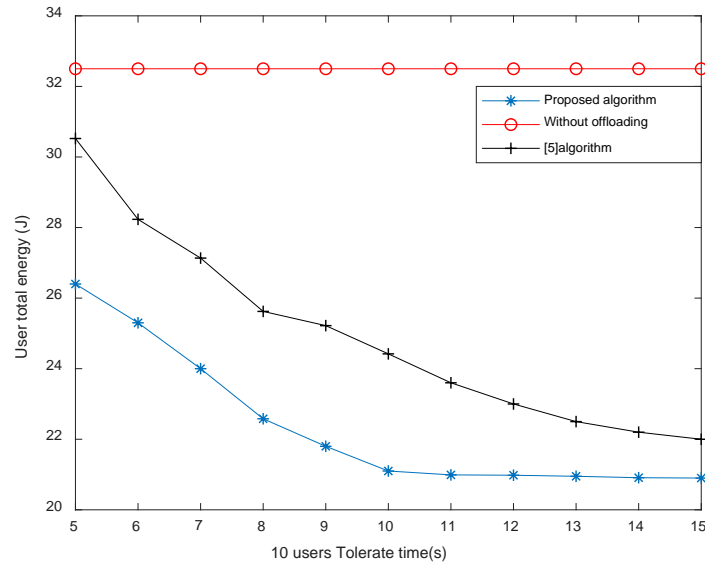

Fig. 5. Users' total energy varies with the Computing resource

The simulation results of user energy consumption changing with the tolerate time when UD numbers=10 are shown in **Fig. 6**, from which it can be seen that with the improvement of the tolerate time, more users offload the tasks, and the energy consumption of users decreases. Besides, **Fig. 6** shows a decline trend in energy consumption, but it seems that the total energy can achieve convergence with the max tolerate time increase.

Fig. 6. Users' total energy varies with the tolerate time

5. Conclusion

In this paper, a problem of minimizing total user energy consumption by optimizing user task offloading strategy and resource allocation in the multi-user and multi-MEC scene is proposed, which is decomposed into two subproblems, namely offloading strategy and resource allocation. Then, the existence of Nash equilibrium (NE) in offloading strategy is proved, and the resource allocation problem is testified to be a convex optimization problem, whose optimal values are obtained by Lagrangian method. Finally, the two subproblems are combined to obtain the optimal solution to minimize the total user energy consumption. Simulation results show that joint optimization of resource allocation and offloading strategy can reduce the total energy consumption compared with random allocation of offloading strategy. Besides, joint optimization of communication and computing resources can reduce more energy consumption compared with the separate optimization of bandwidth and MEC computing resources, which will also encounter bottlenecks. However, this paper has limitations, including a relatively simple scene, and multi-edge server and multiple operators not considered in the simulation. Moreover, data security is also a key problem in MEC. These problems need further research in the future.

References

[1]   Abbas N, Yan Z, Taherkordi A, et al., "Mobile Edge Computing: A Survey," *IEEE Internet of Things Journal*, 5(1), 450-465, 2018. Article (CrossRef Link)
[2]   Wang S, Xing Z, Yan Z, et al., "A Survey on Mobile Edge Networks: Convergence of Computing, Caching and Communications," *IEEE Access*, 5(12), 6757-6779, 2017. Article (CrossRef Link)
[3]   Xu X, Liu J, Tao X, "Mobile Edge Computing Enhanced Adaptive Bitrate Video Delivery with Joint Cache and Radio Resource Allocation," *IEEE Access*, 5(6), 16406-16415, 2017. Article (CrossRef Link)

[4]   Weijian Chen, Yejun He, Jian Qiao, "Cost Minimization for Cooperative Mobile Edge Computing Systems," in *Proc. of 2019 28th Wireless and Optical Communications Conference (WOCC)*, 2019. Article (CrossRef Link)

[5]   YOU, C, Huang, K, Chae, H., "Energy-Efficient Resource Allocation for Mobile-Edge Computation Offloading," *IEEE Transactions on Wireless Communications*, 16(3), 1397-1411, 2017. Article (CrossRef Link)

[6]   L. Li, X. Zhang, et al., "An Energy-Aware Task Offloading Mechanism in Multiuser Mobile-Edge Cloud Computing," *Mobile Information Systems*, 18(4), 1-12, 2018.
Article (CrossRef Link)

[7]   Hao Y, Chen M, Hu L, et al., "Energy Efficient Task Caching and Offloading for Mobile Edge Computing," *IEEE Acce*ss, 6, 11365-11373, 2018. Article (CrossRef Link)

[8]   Xu C, "Decentralized Computation Offloading Game for Mobile Cloud Computing," *Parallel & Distributed Systems IEEE Transactions on*, 26(4), 974-983, 2015. Article (CrossRef Link)

[9]   Yi C, Cai J, Su Z, "A Multi-User Mobile Computation Offloading and Transmission Scheduling Mechanism for Delay-Sensitive Applications," *IEEE Transactions on Mobile Computing*, 19(1), 29-43, 2020. Article (CrossRef Link)

[10] Funai C F, Tapparello C, Heinzelman W, "Computational Offloading for Energy Constrained Devices in Multi-hop Cooperative Networks," *IEEE Transactions on Mobile Computing*, 19(1), 60-73, 2020. Article (CrossRef Link)

[11] Yan J, Bi S, Zhang Y J A, et al., "Optimal Task Offloading and Resource Allocation in Mobile-Edge Computing with Inter-user Task Dependency," *IEEE Transactions on Wireless Communications*, 19(1), 235-250, 2020. Article (CrossRef Link)

[12] Yu S , Langar R , Fu X , et al., "Computation Offloading With Data Caching Enhancement for Mobile Edge Computing," *IEEE Transactions on Vehicular Technology*, 67(11), 11098-11112, 2018. Article (CrossRef Link)

[13] Yi C, Cai J, Su Z, "A Multi-User Mobile Computation Offloading and Transmission Scheduling Mechanism for Delay-Sensitive Applications," *IEEE Transactions on Mobile Computing*, 19(1), 29-43, 2020. Article (CrossRef Link)

[14] Mach, Pavel, Becvar, Zdenek, "Mobile Edge Computing: A Survey on Architecture and Computation Offloading," *IEEE Communications Surveys & Tutorials*, 19(3), 1628-1656, 2017.
Article (CrossRef Link)

[15] Daniel Nowak, Tobias Mahn, Hussein Al-Shatri, "A Generalized Nash Game for Mobile Edge Computation Offloading," in *Proc. of 2018 6th IEEE International Conference on Mobile Cloud Computing, Services, and Engineering (MobileCloud). IEEE*, 2018.
Article (CrossRef Link)

[16] Boyd, Vandenberghe, Faybusovich, "Convex Optimization," *IEEE Transactions on Automatic Control*, 51(11), 1859-1859, 2006. Article (CrossRef Link)

[17] Ranadheera, Shermila, Maghsudi, Setareh, Hossain, Ekram, "Mobile Edge Computation Offloading Using Game Theory and Reinforcement Learning,".

[18] H. Guo and J. Liu, "Collaborative computation offloading for multiaccess edge computing over fiber-wireless networks," *IEEE Trans. Veh*., 67(5), 4514-4526, 2018. Article (CrossRef Link)

[19] Li Q, Zhao J, Gong Y, "Cooperative Computation Offloading and Resource Allocation for Mobile Edge Computing," in *Proc. of 2019 IEEE International Conference on Communications Workshops (ICC Workshops). IEEE*, pp. 1-6, 2019.Article (CrossRef Link)

Zhi Li is with the School of Nanjing University of Posts and Telecommunications (NUPT), Nanjing, China. He received the bacheor's degree in JiangSu University, ZhenJiang, China in 2018. Currently, he is working toward M. Eng.degree in Information and Communication Engineering in NJUPT. His research interests includes the technology of next-generation communication, broadband wireless access, OFDM, channel and source coding, and dynamic allocation of radio resources.

Qi Zhu received the bacheor's and master's degree in radio engineering from Nanjing University of Posts and Telecommunications (NUPT), China, in 1986 and 1989, respectively, where she is currently full-time professor with the School of Telecommunication and Information Engineering of NUPT. Her research interests include the technology of next-generation communication, broadband wireless access, OFDM, channel and source coding, and dynamic allocation of radio resources.