# Joint Heterogeneous Tasks Offloading and Resource Allocation in Mobile Edge Computing Systems

Sihua Wang, Chunyu Pan, and Changchuan Yin
Beijing Laboratory of Advanced Information Network,
Beijing Key Laboratory of Network System Architecture and Convergence,
Beijing University of Posts and Telecommunicaitons, Beijing, P. R. China
Email: {sihuawang,cypan,ccyin}@bupt.edu.cn

***Abstract*—By offloading computationally intensive tasks to the edge cloud, the mobile edge computing (MEC) technique has the potential to realize the critical millisecond-scale latency requirement of next generation mobile services. In this paper, we study heterogeneous tasks offloading in an orthogonal frequency division multiple access (OFDMA) based cloud radio access (C-RAN) network with an integrated MEC server. A joint subcarrier, power allocation and tasks partition problem is formulated to minimize the delay of each user. In order to tackle the intractable optimization problem, an improved hybrid-fitness function evolutionary (HFFE) algorithm is proposed by relaxing the channel allocation indicators into continuous variables. Simulation results indicate that the performance of the proposed algorithm can approach the optimal result obtained by exhaustive search but with much lower complexity.**

## I. INTRODUCTION

The diversity of the mobile applications, such as online video, virtual reality, and interactive gaming, have different features which results in various timeliness requirements for the computation tasks in the wireless networks [1]. When a large number of multifarious tasks are integrated into the network, the traditional cloud computing will cause intolerable delay [2]. To address such a critical challenge, tasks can be offloaded to the mobile edge computing (MEC) server for the performance improvement. Unlike conventional cloud computing which is integrated with remote central clouds and has long latency and fragile wireless connections, MEC provides users with short delay and highly effective computing services by deploying computing nodes or servers at the edge of the network [3].

Tasks offloading and resource allocation in MEC systems have attracted significant attention in recent years. In general, the MEC paradigm can be divided into two categories, binary offloading [4]-[7] and partial offloading [8]-[11]. As for binary offloading, the computation tasks of users can not be partitioned but must be executed as a whole either locally or at the MEC server. For partial offloading, the computation tasks of users can be partitioned into several parts for local computing or for offloading to the MEC server at the same time. Recently, many research works studied the resource allocation in MEC systems [12]-[15]. The authors of [12] utilized queuing theory to study the energy consumption, execution delay and price cost of offloading process in a MEC system, jointly considered both wireless transmission and computing capabilities. In [13], the authors investigated the joint subcarrier and power allocation problem in an orthogonal frequency division multiple access (OFDMA) based MEC system to minimize the maximal delay. Similarly, the works in [14] proposed an algorithm named distributed potential game to provide optimal computation offloading, including uplink subcarrier allocation, transmission power allocation, and computation resource scheduling. The authors in [15] provided the challenges and opportunities of using machine learning techniques for resource management in mobile edge computing networks.

The above works demonstrated the performance enhancement achievable in the MEC system. However, the resource management strategy of the above works does not take the diversity of tasks' requirements into account. Different tasks may have different delay requirements [16] and different users may have different kinds of priority [17]. The authors in [16] introduced the allocation of both radio resources and computation resources of the MEC server to increase system effectiveness based on the different delay requirements. In [17], the authors analyzed the average energy consumption in a mobile edge computing system with two classes of mobile users, named high priority users and low priority users, and high priority users are allowed to use more resources at the server than the low priority users. Based on the diversity of tasks' requirements, the resource management strategy in MEC system will be more effective and more significant.

Considering the coexistence of various tasks in the network, in this paper we jointly optimize the offloading, subcarrier and transmission power allocation, where an OFDMA based C-RAN system integrated with a MEC server is considered to offload the various types of user tasks [18]. The difference of task types is caused by the computation requirements, such as image acquistion and processing must be excuted locally, while resource-intensive task task must be computed at the MEC server. This paper aims to investigate the latency minimization problem including the communication time and computation time. The main contributions of this work are summarized as follows:

- We propose a C-RAN system integrated with a MEC server to excute the users' tasks. Based on the proposed architecture, we not only consider the subcarrier and the transmission power, but also the partial task offloading to

978-1-5386-6119-2/18/$31.00 ©2018 IEEE

minimize the end-to-end latency of each user.

- We consider the coexistence of heterogeneous tasks: local computing task, edge computing task and cooperative computing task. Considering the limit of network resources, an optimization problem is formulated to minimize the sum delay of all users.
- We propose an improved HFFE algorithm to provide a suboptimal solution, which makes the original problem tractable. Compared the achieved suboptimal solution with the globally optimal solution obtained by exhaustive search, we show the accessible gap of the proposed algorithm and prove its convergence.

The rest of this paper is organized as follows. The system model and the problem formulation are introduced in Section II. The improved HFFE algorithm is developed in Section III. Section IV presents the simulation results and finally we conclude the paper in Section V.

## II. SYSTEM MODEL AND PROBLEM FORMULATION

In this paper, we consider the heterogeneous tasks that have different computation requirements. As for the computation requirement, we assume three kinds of common tasks, named local-computing task, MEC-computing task, and collaboration-computing task, which are defined as follows:

- **Local computing task**: The computing task must be processed locally. This kind of tasks are more common in image acquisition and processing, e.g., image acquisition devices need to compress the data before it can be transmitted due to bandwidth limitation.
- **Edge computing task**: Computing task must be processed on an MEC server. This kind of task are more common in extensive resource requirement task (e.g., video). The resources of processing these tasks exist in MEC's cache or other upper network [19], which must be computed in an MEC server.
- **Cooperative computing task**: This kind of task can be processed by an MEC server and users collaboratively. The computating task in this category can be arbitrarily divided in bitwise for partial local computing and partial offloading.

### A. System Model

We consider a C-RAN system integrated with an MEC server to execute the various types of tasks of the users. As shown in Fig. 1, the system consists of a BBU pool attached to an MEC server, $\mathcal{N}$={1,2,...,$N$} RRHs, $\mathcal{M}$={1,2,...,$M$} users, and high speed fronthaul communication links connecting RRHs to the BBU pool. There are $K$ orthogonal subcarriers in the system with index $\mathcal{K}$={1,2,...,$K$}, where the bandwidth of each subcarrier is $W$. Each subcarrier maintains a maximal transmission power $p_{max}$. To schedule the uplink and downlink transmissions simultaneously, the BBU pool needs to allocate subcarriers to each RRHs to serve the user.

We define a 3-dimensional index matrix $S$ with elements $s^k_{n,m}$, if RRH $n$ is scheduled to serve user $m$ using subcarrier $k$, $s^k_{n,m}$=1 and 0 otherwise. Each subcarrier is allocated to at most one user to avoid interference, as such we have

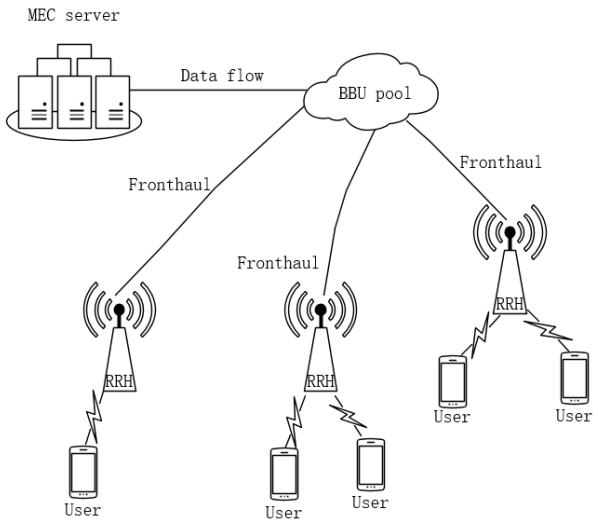


Fig. 1. C-RAN integrated with MEC server

$$\sum_{n\in\mathcal{N}}\sum_{m\in\mathcal{M}} s^k_{n,m} \le 1, \forall k \in \mathcal{K}. \tag{1}$$

In addition, every user in the system can use at least one subcarrier of each RRH to ensure that each user can transmit data with the MEC server, thus we have

$$\sum_{k\in\mathcal{K}} s^k_{n,m} \le 1, \forall n \in \mathcal{N}, \forall m \in \mathcal{M}. \tag{2}$$

For each subcarrier, we denote the instantaneous channel power gain between user $m$ and RRH $n$ as $h^k_{n,m}$, and the bandwidth of each subcarrier is $W$ . Then the channel capacity for subcarrier $k$ can be expressed as

$$r^k_{n,m} = W\log_2\left(1+\frac{p^k_{n,m}h^k_{n,m}}{\sigma^2_N}\right), \tag{3}$$

where $p^k_{n,m}$ is the transmission power between user $m$ and RRH $n$ on subcarrier $k$, $\sigma^2_N$ represents the variance of the additive white Gaussian noise (AWGN).

The overall data rate between user $m$ and RRH $n$ on subcarrier $k$ can be expressed as

$$r^k_{n,m} = s^k_{n,m}W\log_2\left(1+\frac{p^k_{n,m}h^k_{n,m}}{\sigma^2_N}\right). \tag{4}$$

As mentioned before, there are three kinds of tasks for each user $m$: Local computing task, edge computing task, and cooperative computing task. We consider a common calculating model, that is, the time consumption of computation depends on the CPU clock frequency and the number of CPU cycles required for computing 1-bit of task. Denoting the CPU clock

frequency for users $m$ and the MEC server as $f_m$ and $F$, respectively. Similarly, the number of CPU cycles required for computing 1-bit task in the MEC server and user $m$ is $\omega$ and $\omega_m$, respectively. We assume that the size of users' task data is $\lambda_m$. Therefore, the time consumption for each kind of task can be calculated as follows

- **Local computing task**: The task is computed locally, and then the result is transmitted to the MEC server. The time consumption for finishing the computation of local task can be written as

$$t_1 = t_{user} + t_{upload} = \frac{\omega_m \lambda_m}{f_m} + \frac{\lambda_m}{r^k_{n,m}}. \tag{5}$$

- **Edge computing task**: This kind of task is computed at the MEC server, and then the result is transmitted to the mobile user. The time consumption for this task is

$$t_2 = t_{MEC} + t_{download} = \frac{\omega \lambda_m}{F} + \frac{\lambda_m}{r^k_{n,m}}. \tag{6}$$

- **Cooperative computing task**: For this kind of task, we assume that the task data can be partitioned into two parts, part of it can be processed locally while the remaining needs to be processed at the MEC server in order to fully utilize the computation resource at both mobile users and the MEC server. Denote the proportion of the data processed locally as $\alpha_m$. The computing time is influenced by the tasks partition, the subcarrier and power allocation. The different tasks partition may lead to different computation time, and the subcarrier and power allocation scheme will influence the transmission time. These three factors are coupled and can not be considered separately. We ignore the time for the download-process because the data size of computation result is much less than the upload [20]. The time consumption for the computation of cooperative computing task is

$$\begin{aligned} t_3 &= \max\left\{t_{user}, t_{upload} + t_{MEC}\right\} \\ &= \max\left\{\frac{\omega_m \alpha_m \lambda_m}{f_m}, \frac{(1-\alpha_m)\lambda_m}{r^k_{n,m}} + \frac{\omega(1-\alpha_m)\lambda_m}{F}\right\}. \end{aligned} \tag{7}$$

### *B. Problem Formulation*

Based on the above model, we formulate the resource allocation optimization problem for the heterogeneous tasks to minimize the sum delay of each user. This optimization problem involves finding the subcarrier allocation indicator $s^k_{n,m}$, the transmission power $p^k_{n,m}$ and the task partition $\alpha_m$. The optimization problem can be mathematically formulated as

$$\min_{s^k_{n,m}, p^k_{n,m}, \alpha_m} \sum_{m=1}^{M} t_i, \tag{8}$$

subject to

$$\sum_{n\in\mathcal{N}} \sum_{m\in\mathcal{M}} s^k_{n,m} \le 1, \forall k \in \mathcal{K}, \tag{9}$$

$$\sum_{k\in\mathcal{K}} s^k_{n,m} \le 1, \forall n \in \mathcal{N}, \forall m \in \mathcal{M}, \tag{10}$$

$$s^k_{n,m} \in \{0,1\}, \forall n, m, k, \tag{11}$$

$$p^k_{n,m} < p_{\max}, \forall m, \tag{12}$$

$$\sum_{n,m} p^k_{n,m} \le P, \tag{13}$$

$$0 \le \alpha_m \le 1, \tag{14}$$

## III. PROBLEM SOLUTION BASED ON EVOLUTIONARY ALGORITHM

The formulated problem is a mixed integer nonlinear programming (MINLP), since the subcarrier allocation indicators are binary integers, which makes the problem quite difficult to obtain the globally optimal solution. Thus, we propose an improved HFFE algorithm to derive a suboptimal solution. In essence, the improved HFFE algorithm belongs to the genetic algorithm, demonstrating the principle 'survival of the fittest'. In a genetic algorithm, a number of individuals to an optimization problem is evolved toward better by mutation and alteration. In each evolutionary generation, the fitness of each individual in the population is evaluated, and then individuals are sorted according to the fitness. Fitness is a measure of the ability of an individual, to make sure the individual can produce viable offspring and contribute to future generations. Generally speaking, the individuals in a species that have the highest fitness will contribute disproportionately to the subsequent generations.

In this paper, each individual chromosome is expressed as a multi-dimensional vector and the first $N \times M$ elements of it indicate the binary channel allocation result. Hence, we propose a evolutionary algorithm by relaxing the binary variables into continuous ones. The rounding operation is applied for the first $N \times M$ variables before calculating the objective function and the penalty function. The rounding method is in accordance with the following formula

$$x = \begin{cases} 1, & \text{if } x \ge 0.5, \\ 0, & \text{if } x < 0.5. \end{cases} \tag{15}$$

The second $N \times M$ elements indicate the transmission power in each possible subcarrier allocation scheme. The last $m$ elements indicate the uploading proportion for collaboration computing task.

To ensure evolutionary uncertainty, we define fitness functions in piecewise terms as

$$F(\mathrm{x}) = \begin{cases} F_1(x), & \text{if } r \ge 0.5, \\ F_2(x), & \text{if } r < 0.5, \end{cases} \tag{16}$$

where $r$ is a variable randomly distributed between 0 and 1. $F_2(x)$ is the expression of the piecewise function $F_1(x)$ at the infeasible solutions.

The fitness function $F_1(\mathrm{x})$ is given by

$$F_1(\mathrm{x}) = \begin{cases} 2 * S + f_1(x), & x \in A, \\ \frac{g}{g+1}\left(\frac{n}{S} * 0.5 + 0.5\right) p_1(x) + f_1(x), & \text{otherwise}, \end{cases} \tag{17}$$

where $S$ is the size of population, $A$ is the collection of feasible solutions, $g$ is the current generation number, and $n$ is the number of feasible solutions in the current population. Objective satisfaction degree function $f_1(x)$ and penalty satisfaction degree function $p_1(x)$ are defined as follows

$$f_1(x) = \frac{f_{\max} - f(x)}{f_{\max} - f_{\min}} S, \tag{18}$$

$$p_1(x) = \frac{p_{\max} - p(x)}{p_{\max} - p_{\min}} S, \tag{19}$$

where $f_{max} = \max\{f(x)\}$, $f_{min} = \min\{f(x)\}$, and $p_{max} = \max\{p(x)\}$. In addition, penalty function $p(x)$ is defined as

$$p(x) = \sum_{j=1}^{J} q_j(x), \tag{20}$$

where $J$ is the number of constraints, and $q_j(x)$ is given by

$$q_j(x) = \begin{cases} [\max\{0, g_j(x)\}]^2, & j \in I, \\ [h_j(x)]^2, & j \in E, \end{cases} \tag{21}$$

where $I$ and $E$ are index sets of inequality and equality constraints in the above MINLP optimization problem.

In general, the improved HFFE algorithm starts with a number of random feasible solutions with uniform distribution. And then, the algorithm evaluating the solutions according to fitness function. Simplex crossover, particle swarm optimization (PSO) mutation, and local search operator in the framework is executed to produce offspring. Finally, the elitist selection scheme is used to select the next generation population. The proof of the convergence of the proposed HFFE algorithm is provided in the Appendix.

## IV. Simulation Results

In this section, simulations are carried out to evaluate the performance of our proposed algorithm. First, we simplify the model in order to obtain the globally optimal solution by exhaustive search method. Then, we present simulation results to validate the performance of the proposed improved HFFE algorithm. The simulation parameters are defined in Table I unless otherwise specified.

### A. Performance of the Proposed HFFE Algorithm

We simulate a C-RAN with $N$ = 3 RRHs to serve $M$ = 6 users which are randomly located in a circular area. Without loss of generality, we consider wireless fading channel where the channel gain $h_{n,m}^k$ is uniformly distributed between $[0.9, 1.1]$ in each time slot. $N_0$ = $10^{-15}$ mW/Hz denotes the power spectral density of additive white Guassian noise. The total bandwidth is 30 MHz which are equally divided into $K$ = 9 subcarriers. The transmission power of mobile users is less than 0.5 W, the maximum transmission power of RRH is 2 W. We assume that the length of the task $\lambda_m$ is uniformly distributed between $[100, 400]$ KB, the CPU frequencies of mobile users, $f_m$, and the MEC server, $F$, is 0.5GHz and 100GHz, respectively. And the required number of CPU cycles per bit for mobile users $\omega \in [1000, 2000]$ and MEC server $\omega_m$ is 1500.

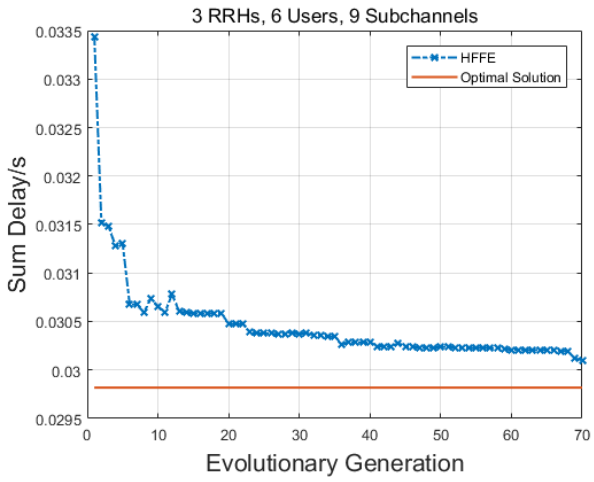


Fig. 2. Comparison of the optimal solution and the suboptimal solution obtained by the HFFE algorithm.

Fig. 2 shows that the solution of the proposed improved HFFE algorithm is close to the global optimal solution, and the average gap over 50 runs is $6\times10^{-5}$ seconds. This indicates that the algorithm is capable to obtain the global optimal solution. Moreover, it can be seen from the figure that the algorithm converges after several iterations and the proposed scheme can ensure the convergence effectively, although the sum delay is not monotonically decreasing due to uncertainty of mutation. On the contrary, exhaustive method need calculate $2^{N*M*K}\times10^{3*N*M}\times10^{3*M}$ times which is a huge cost in terms of time complexity. Although the proposed algorithm may not reach the a global optimal solution, however it has a much lower complexity.

### B. Performance of the Formulated Problem

We consider the scenario that a C-RAN network with $M$ = 27 users which are randomly distributed in the coverage of RRHs. We use the HFFE algorithm to show how the

TABLE I
SIMULATION PARAMETERS

| Parameter Notation | Value |
|---|---|
| $N$ | 3 |
| $M$ | 6, 27 |
| $K$ | 9, 50 |
| $p_{max}$ | user 1.5 W, RRH 3 W |
| $W$ | 4.5 MHz |
| $h_{n,m}^k$ | $[0.9, 1.1]$ |
| $N_0$ | $10^{-15}$ mW/Hz |
| $\omega$ | $[1000, 2000]$ |
| $\omega_m$ | 1500 |
| $\lambda_m$ | $[100, 400]$ KB |
| $F$ | 100 GHz |
| $f_m$ | 0.5 GHz |

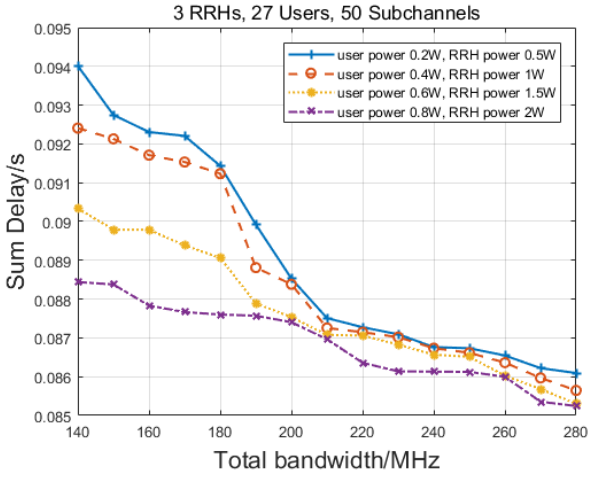


Fig. 3. The sum delay against bandwidth $W$ with $N$=50.

total bandwidth $W$ and the transmission power $p_{max}$ affect the delay of the considered system. Fig. 3 indicates that the sum delay $T_{sum}$ decreases with the increase of bandwidth and transmission power. Furthermore, the sum delay firstly drops rapidly when bandwidth is small and then slowly when bandwidth is larger enough. The reason is that when the bandwidth is quite small, the performance of the system is limited by the bandwidth. On the contrary, when the bandwidth increases gradually and becomes large enough, the dominant constraint is no longer the bandwidth but the transmission power. It also can be seen from Fig. 3 that when the bandwidth and the transmission power are both sufficient large, the delay will converge nearly to a same value. The reason is that the performance bottleneck is no longer due to the transmission resource but the computation resource, which means there is no need to set a large bandwidth and a large transmission power for the transmission system.

## V. Conclusion

In this paper, we investigated the joint resources allocation and heterogeneous tasks offloading problem in an OFDMA based C-RAN system integrated with an MEC server to minimize the sum delay of each user. We proposed an improved HFFE algorithm to obtain a suboptimal solution for the original intractable MINLP problem, and numerically proved that the obtained solution is close to the globally optimal solution. The simulation results show that the sum delay decreases with the bandwidth and the transmission power and the algorithm has a much lower complexity compared to the exhaustive search algorithm.

## Appendix

The convergence of the HFFE algorithm can be proved as follows.

We assume that $\{X_n\}$ is a sequence of random variables on the probability space $\{\Omega, F, P\}$, $X_n$ converges with probability 1 towards $X$ if there is a random variable $X$ which satisfies:

$$P\left\{\lim_{n\to\infty} X_n = X\right\} = 1. \tag{22}$$

We assume $y$ and $x$ are elements of the sequence $\{X_n\}$. If the probability that $x$ can evolve into $y$ through hybridization and mutation is greater than 0, then it means that $y$ is reachable from $x$, which can be expressed as

$$P\{cm(x) = y\} > 0, \tag{23}$$

where $cm$ means hybridization and mutation operations.

As for genetic algorithm, if $x$ can evolve into $y$ by variation and hybridization with the precision $\delta$, it can be expressed as

$$P\{||cm(x) - y|| \le \delta\} > 0. \tag{24}$$

We assume that $x$ generates a set of random offspring by hybridization, which is noted as $a$=$\{a_1, ..., a_n\}$. Then, we will prove the offspring $a$ can mutate into $y$ with the precision $\delta$, which can be shown as

$$P\{||cm(x) - y|| \le \delta\} = P\{||m(a) - y|| \le \delta\} > 0. \tag{25}$$

According to the continuity of the optimization problem, for any small positive number $\delta$, the HFFE algorithm can converge with probability 1 in precision $\delta$ towards the optimal solution, which is

$$P\left\{\lim_{g\to\infty} ||x_g^* - x^*|| \le \delta\right\} = 1, \tag{26}$$

where $x^*$ is the globally optimal solution, $x_g^*$ is the best solution gained by the $g$th-generation population.

To prove (26), we demonstrate (25) and figure out how the algorithm produces offspring.

Actually, $a$ produces a mutant offspring in the following way:

$$x_{i+1} = \begin{cases} x_i - \Theta(x_i - l_i), r < 0.5 \text{ or } x_i = u_i, \\ x_i + \Theta(u_i - x_i), r > 0.5 \text{ or } x_i = l_i, \end{cases} \tag{27}$$

where $u_i$ and $l_i$ represent the upper bound and lower bound of the $i$-th component, respectively. So the corresponding probability of producing a mutant offspring by $a$ is

$$\left(1 + \frac{1}{g+1}\right)\xi > \xi > 0. \tag{28}$$

As such the mutation operator can be divided into two cases:

1) $a_i$=$u_i$ or $a_i$=$l_i$, namely, $a_i$ is at the boundary, and if $a_i$=$u_i$, then

$$\begin{aligned} &P\{||x_{i+1} - y_i|| \le \delta\} \\ &= P\{||a_i - \Theta(a_i - l_i) - y_i|| \le \delta\} \\ &= P\left\{\Theta \in \left[\frac{a_i - y_i - \delta}{a_i - l_i}, \frac{a_i - y_i + \delta}{a_i - l_i}\right]\right\} \\ &= \frac{2\delta}{u_i - l_i} > \frac{\delta}{u_i - l_i}. \end{aligned} \tag{29}$$

Similarly, when $a_i$=$l_i$,

$$P\{||x_{i+1} - y_i|| \le \delta\} > \frac{\delta}{u_i - l_i}.$$

2)When $a_i \in (l_i,u_i)$, if $y_i \in (l_i,a_i)$, then

$$\begin{aligned} & P\{||m(a_i)-y_i|| \le \delta\} \\ &= \frac{1}{2} \times P\{||a_i - \Theta(a_i - l_i) - y_i|| \le \delta\} \\ &= \frac{1}{2} \times P\left\{\Theta \in \left[\frac{a_i - y_i - \delta}{a_i - l_i}, \frac{a_i - y_i + \delta}{a_i - l_i}\right]\right\} \\ &= \frac{1}{2} \times \frac{2\delta}{a_i - l_i} > \frac{\delta}{u_i - l_i}. \end{aligned} \tag{30}$$

Similarly, if $y_i \in (a_i,u_i)$, then

$$P\{||m(a_i)-y_i|| \le \delta\} = \frac{1}{2} \times \frac{2\delta}{u_i - x_{i+1}} > \frac{\delta}{u_i - l_i}. \tag{31}$$

Since the variables are independent, then

$$P\{||m(a)-y|| \le \delta\} > \prod_{i=1}^{n} \frac{\delta}{u_i - l_i} > 0. \tag{32}$$

For $g$th-generation, we have

$$P\{||cm(x)-y|| \le \delta\} > \xi \prod_{i=1}^{n} \frac{\delta}{u_i - l_i} > 0. \tag{33}$$

Therefore, we draw a conclusion that $x$ can evolve into $y$ by variation and hybridization with the precision $\delta$.

The population sequence of the algorithm can be considered as a Markov chain $\{P(t)\}$. The Markov chain has two states:

1)In the population sequence $\{P(t)\}$, there could be any individual $x_k$ which satisfies: $||x_k - x|| < \delta$,

2)In the population sequence $\{P(t)\}$, the individual $x_k$ doesn't satisfy state 1).

Because of the monotonicity of the population, the transfer probability from state *1* to state *2* is 0, which means that state *1* is an absorbing state. We have proved that if $x$ and $y$ are in the probability space, then $x$ can evolve into $y$ by variation and hybridization with the precision $\delta$. According to Markov chain theory, the following equation holds

$$P\left\{\lim_{g \to \infty} ||x_g^* - x^*|| \le \delta\right\} = 1 \tag{34}$$

Then we can conclude that the improved HFFE algorithm is convergent.

## ACKNOWLEDGMENT

This work was supported in part by the National Natural Science Foundation of China under Grants 61671086 and 61629101, in part by the 111 Project under Grant B17007, and in part by the Director Funds of Beijing Key Laboratory of Network System Architecture and Convergence under Grant 2017BKL-NSAC-ZJ-04.

## REFERENCES

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

[2] P. Mach, Z. Becvar, "Mobile edge computing: A survey on architecture and computation offloading," *IEEE Communications Surveys & Tutorials*, vol. 19, no. 3, pp. 1628-1656, Mar. 2017.

[3] S. Wang, X. Zhang, Y. Zhang, L. Wang, J. Yang, and W. Wang, "A survey on mobile edge networks: Convergence of computing, caching and communications," *IEEE Access*, vol. 5, no. 2, pp. 6757-6779, Mar. 2017.

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

[5] S. Bi and Y. J. Zhang, "Computation rate maximization for wireless powered mobile-edge computing with binary computation offloading," *IEEE Transactions on Wireless Communications*, vol. 17, no. 6, pp. 4177-4190, Jun. 2018.

[6] 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, Oct. 2016.

[7] 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.

[8] M. Liu, Y. Liu, "Price-based distributed offloading for mobile-edge computing with computation capacity constraints," *IEEE Wireless Communications Letters*, vol. 7, no. 3, pp. 420-423, Jun. 2018.

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

[10] J. Xu, L. Chen, and S. Ren, "Online learning for offloading and autoscaling in energy harvesting mobile edge computing," *IEEE Transactions on Cognitive Communications and Networking*, vol. 3, no. 3, pp. 361-373, 2017.

[11] C. You, K. Huang, H. Chae, and B. H. Kim, "Energy-efficient resource allocation for mobile-edge computation offloading," *IEEE Transactions on Wireless Communications*, vol. 16, no. 3, pp. 1397-1411, Mar. 2017.

[12] L. Liu, Z. Chang, X. Guo, and T. Ristaniemi, "Multi-objective optimization for computation offloading in mobile-edge computing," *IEEE Symposium on Computers and Communications*, pp. 832-837, Sept. 2017

[13] M. Li, S. Yang, Z. Zhang, J. Ren, and G. Yu, "Joint subcarrier and power allocation for OFDMA based mobile edge computing system," *in Proc. IEEE 28th Annual International Symposium on Personal, Indoor, and Mobile Radio Communications*, pp. 1-6, Oct. 2017.

[14] J. Zhang, W. Xia, F. Yan, and L. Shen, "Joint computation offloading and resource allocation optimization in heterogeneous networks with mobile edge computing," *IEEE Access*, vol. 6, pp. 19324-19337, Mar. 2018.

[15] M. Chen, M. Mozzaffari, W. Saad, C. Yin, M. Debbah, and C. S. Hong, "Caching in the sky: Proactive deployment of cache-enabled unmanned aerial vehicles for optimized quality-of-experience," *IEEE Journal on Selected Areas on Communications (JSAC), Special Issue on Human-In-The-Loop Mobile Networks*, vol. 35, no. 5, pp. 1046-1061, May. 2017.

[16] T. Y. Kan, Y. Chiang, and H. Y. Wei, "Task offloading and resource allocation in mobile edge computing system," *in Proc. Wireless and Optical Communication Conference*, pp. 1-4, Apr. 2018.

[17] K. Aljobory and M. A. Yazici, "Effect of queueing delay and service discrimination on offloading performance in two-class mobile edge computing systems," *in Proc. Signal Processing and Communications Applications Conference*, pp. 1-4, May. 2018.

[18] Y. Li and C. Yin, "Joint energy cooperation and resource allocation in C-RANs with hybrid energy sources," *in Proc. International Conference on Communications in China*, pp. 1-6, Oct. 2017.

[19] Y. Hao, M. Chen, L. Hu, M. S. Hossain, and A. Ghoneim, "Energy efficient task caching and offloading for mobile edge computing," *IEEE Access*, vol. 6, pp. 11365-11373, 2018.

[20] T. Yang and H. Zhang and H. Ji and X. Li, "Computation collaboration in ultra dense network integrated with mobile edge computing," *in Proc. IEEE 28th Annual International Symposium on Personal, Indoor, and Mobile Radio Communications*, pp. 1-5, Oct. 2017.