**University of Macau**

**Faculty of Science and Technology**



**Interim report of**

**Design of Efficient Mobile Edge Computing in Wireless Network**

***by***

**LI KIN TAK, *Student No: D-B4-2640-6***

Project Supervisor

Prof. Yuan Wu

TABLE OF CONTENTS

CHAPTER 1. Background 3

1.1.1 Abstract 3

1.1.2 Advantage of MEC 4

1.1.3 Application Scenarios 5

1.2.1 Research Status 7

1.2.2 Typical MEC-Related Models 8

1.2.3 Related Work 9

1.2.4 Paper Movtivation and Outline 12

CHAPTER 2. Schedule of this project 13

CHAPTER 3. Meeting summary 16

CHAPTER 4. Brief introduction of mean Modeling 18

CHAPTER 5. Tools and methods used in project 21

CHAPTER 6. Simulation result of EEO-Bot layer 21

CHAPTER 7. Member Contributions 22

CHAPTER 8. Conclusion and Future plan 22

CHAPTER 9. reference 22

### CHAPTER 1. Background

**1.1.1 Abstract**

With the rapid development of the Internet of Things, the number of IOT devices has increased rapidly. Cisco predicts that approximately 50 billion IOT devices will be added to the Internet in 2020 [1] . Most of these devices have limited communication and storage resources and cannot complete all tasks through its own resources, you need to rely on cloud or edge devices to enhance its functions. And because of the huge amount of data, it is impossible for the cloud computing center to handle all tasks alone. With the advent of the 5G era, the limitations of MCC, that is, the propagation distance from the end user to a remote cloud center is very long, and the network bandwidth and computing resources are limited, making cloud computing transmission and calculation take a certain time, causing it has caused a huge delay, which has led to the inability to meet the needs of more and more emerging applications. MEC can solve this problem. MEC deployed cloud computing capabilities to mobile networks to provide IT service environments at the edge of mobile networks and cloud computing resources, offloading computing tasks from mobile devices, making cloud computing tasks as close to the data producer as possible, and making haptic application experience possible. Migrating computing users to edge servers can not only reduce latency, but also reduce overall energy consumption due to factors such as short transmission distances [2] .

**1.1.2 Advantages of MEC**

Compared with MCC, MEC has lower latency, lower energy consumption, and also supports context awareness and stronger security.

**1.Delay:**

Mobile service delay is generally composed of transmission delay, calculation delay and communication delay. MEC's ​​information transmission distance is generally the distance from a small cellular network or device to the edge device. Compared to the distance between the MCC from the end user to the cloud computing center of the core network, the MEC has a lower distance. Transmission delay. For computing latency, although the edge device is not enough to compare with the huge computing power of the cloud computing center, since the number of users facing a single edge device is correspondingly reduced, users can enjoy more computing resources, which reduces the calculation. delay. And because the communication distance and the number of passing nodes are reduced, data will only be transmitted between the data origin and the edge server, reducing its communication delay.

**2. Energy consumption:**

Energy consumption is more important than edge devices with limited energy. The number of IoT devices is extremely large. It is impossible to frequently charge or replace the battery for each device. Through effective calculation and shunting, the computing tasks are diverted from IOT. When migrating to an edge server from a device, the energy consumption of IOT will be greatly reduced compared to the local computing tasks.

**3. Context-aware:**

In MEC, the distance between the edge device and the end user is short, so that edge computing can obtain the user's real-time information, such as its location, behavior, and environment, and then use this information to provide various services.

**4. Security:**

In MCC, user information is concentrated in the cloud computing data center, which easily leads to information security issues such as data leakage and loss. However, the scale of MEC is small, the stored data is small, and the attack value of edge servers is relatively higher than that of cloud computing centers. Small, thereby reducing the possibility of being attacked. MEC's ​​distributed deployment also improves its security.

**1.1.3 Application Scenarios**

The advantages of edge computing enable edge computing to play a huge role in 5G systems [3] . It has been applied to many application scenarios, such as VR, AR applications, cloud gaming platforms, ultra high-definition video streaming, etc., and it can also be applied to many emerging scenarios. Such as video analysis, real-time environmental monitoring [] and so on.

**1. Video analysis:**

In order to create a smart city, a large number of cameras are deployed in the city for vehicle license plate recognition, face recognition, and security monitoring. The transmission and analysis of video streams require a lot of computing and network resources, and most of the cameras are not sufficient. Computing power cannot perform video analysis locally, and edge servers that are close and have sufficient computing resources can just meet their needs. Edge computing can reduce its delay and reduce the energy consumption of sensor devices.

**2.VR / AR:**

VR and AR are very latency-sensitive applications that require extremely low latency to satisfy the user experience. The average VR / AR mobile device does not provide the computing power that satisfies the application experience. Offload computing tasks from the device to the edge. The server can greatly reduce its latency and improve the user experience.

**3.Internet of Things / Internet of Vehicles:**

With the development of the Internet of Things, machine learning and other technologies, vehicles can also be used as smart devices to access the Internet. The low latency and context awareness of edge computing enable smart vehicles to respond quickly to vehicle behavior, status, and surrounding environment, and to automatically Techniques such as driving, abnormal condition judgment, and even road condition analysis all help a lot.

**4. Environmental monitoring:**

Edge computing has location awareness capabilities. Edge computing can collect and process data at nearby geographic locations without transmitting to a cloud computing center. This feature can be applied to public management and scientific research, such as deploying sensors in various places to obtain information about the surrounding environment. To help scientists monitor geographic conditions and understand the climate change in the area. Data transmission in sensors and cloud computing centers will generate huge energy consumption and delays, affecting the real-time nature of information. Mobile edge computing allows people to make reaction.

**1.2.1 Research Status**

The huge advantages of edge computing make people work hard on research and development in the field of edge computing. Edge computing originated in the 1990s. Akamai proposed a content distribution network. A storage transmission node was set up near the end user to store static content and allow users to get the content they need nearby. In 2006, Amazon released the EC2 service, announcing the advent of the cloud computing era. With the development of cloud computing, in order to reduce the delay, reduce the computing load of the cloud computing center, and solve the problem of insufficient network transmission bandwidth, in 2009, Satyanarayanan and others proposed the concept of Cloudlet [5] , which deploys a trusted and resource-rich host on the network Like the cloud computing center, the edge provides users with IT and cloud computing services. In 2012, Cisco proposed the concept of "fog computing". In 2013, "edge computing" proposed by American scholars received great attention in academia and industry. Since then, relevant international conferences on edge computing have continued to develop and move to edge computing. Technology has entered a period of rapid growth. In 2016, Professor Shi Weisong's team gave a formal definition of edge computing and published a paper entitled "Edge Computing: Vision and Challenges", which was widely cited. In 2018, edge computing was introduced to the general public, and more people began to become familiar with this technology, which has increased the number of participants in this field and has become a hot topic in academia and industry.

**1.2.2 Typical MEC-Related Models**

This section will introduce some typical MEC calculation, communication and other models. Analysis based on these models can provide ideas for research and promote theoretical analysis.

**1.MEC calculation model**

Due to multiple factors such as delay, bandwidth, and utilization, it is very complicated to accurately model computing tasks, but there are also some simple models that are reasonable and mathematically easier to handle. The following two computing models are commonly used in the MCC and MEC literature, namely binary and partial computing migration, respectively.

**a.** **Binary migration task model:**

For highly integrated or relatively simple tasks, you can choose to complete all locally on the mobile device or uninstall the whole to the MEC server.

**b. Partial migration task model**

For a task consisting of multiple processes, the task can be divided into two parts, one is executed locally on the mobile device, and the other is migrated to the edge server for simultaneous execution.

**2. Communication model**

**a. Wireless bit pipe model:**

Different from the mobile pipe and cloud computing server in the MCC literature, the communication pipe is given as a bit pipe model with a constant rate and a random rate. MEC needs to consider the requirements of edge cloud and delay sensitive applications, so Consider some key attributes of wireless transmission, such as multi-path fading of wireless channels caused by scattering of objects in the environment, interference between wireless signals, and spectrum shortage.

**b. Wireless channel fading model:**

In the task migration of edge computing, the influence of channel gain on delay must be considered, and the migration must be performed on channels of good quality, so the model needs to be considered in conjunction with migration and wireless transmission.

**1.2.3 Related Work**

MEC helps terminal devices with limited resources to offload computing tasks to the edge server to complete the problem of insufficient intelligent IOT computing resources, which has aroused widespread concern in academia and industry [6]. In order to make MEC more efficient, the multi-access MEC paradigm was conceived [7]. In a multi-access MEC, a terminal device can offload single or multiple computing tasks to different edge servers at the same time, thereby better utilizing edge server resources. In this paper, starting from the task migration method of edge computing, this paper studies the design algorithm for the multi-access MEC energy efficiency optimization problem based on non-orthogonal multiple-access transmission, and finds the best task migration solution for single-user multi-tasking and multi-server. The academic achievements related to the research in this article are reviewed as follows:

Chen et al. First studied the multi-user joint offloading decision and channel allocation problem, and then further studied the multi-tasking scheme to simultaneously optimize the migration decision and the allocation of computing and communication resources in order to reduce the total energy and delay of all users in edge computing cost [8]. Huang et al. In "Deep Reinforcement Learning for Online Computation Offloading in Wireless Powered Mobile-Edge Computing Networks, "proposed a solution for wireless MEC online computing offload [9]. Guo et al. Proposed a greedy migration solution based on the multi-user ultra-dense MEC server [10].

In a large number of researches on edge computing, multiple-access edge computing based on non-orthogonal multiple-access technology (NOMA) can further reduce the delay, thereby meeting more requirements. Different from traditional frequency division, time division, etc., NOMA allows multiple mobile users to use the same resource block at the same time and provide services to multiple users at the same time on the same subchannel. In order to use NOMA to improve throughput and energy efficiency, many studies envision NOMA as a potential solution for multiple-access MEC, and use NOMA to simultaneously migrate computing tasks to multiple edge servers, further reducing transmission delays and energy consumption [11][12].

The relevant achievements of NOMA are reviewed as follows:

In order to improve the energy efficiency of non-orthogonal multiple-access transmission, scholars such as Zhang and Fang proposed various resource allocation schemes [13][14]. Ding et al. Studied the migration delay minimization problem of NOMA-MEC [15]. Wu et al. Studied the joint power allocation and service scheduling of NOMA relay networks [16]. Recent studies by some scholars have shown that NOMA can reduce the delay and energy consumption of MEC [17].

Mobile edge computing based on non-orthogonal multiple access transmission allows users to divide computing tasks into multiple parts and simultaneously migrate to different servers to complete, shortening the transmission distance, reducing latency and energy consumption, and greatly improving efficiency and application experience.

**1.2.4 Paper Motivation and Outline**

Edge computing's low latency, low energy consumption, high security, and support for context awareness make edge computing have considerable potential. Based on curiosity in this field, after reading the papers of seniors and learning related background knowledge, I chose the topic of designing mobile edge computing energy efficiency solutions for orthogonal multiple access transmission is arranged as follows:

The second chapter is the schedule of whole project.

The third chapter is the meeting summary.

Chapter 4 is the mean modeling in our project.

Chapter 5 is the tools we used to solve problems.

Chapter 6 is the simulation result of EEO-BOT layer.

Chapter 7 is the member contribution.

Chapter 8 is the conclusion and future plan.

And the last chapter is the reference.

### CHAPTER 2. SCHEDULE

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Objective | Task | Content | Schedule  YY/MM/DD | Completion |
| 1.  Background | 1.1 | Read paper to study background knowledge of edge computing. | 19/09/10~  19/09/17 | 100% |
| 1.2 | Read paper to study background knowledge of NOMA. | 19/09/17~  19/09/21 | 100% |
| 1.3 | Read paper and do summary base on the knowledge we got in 1.1~1.2 task. | 19/09/21~  19/09/27 | 100% |
| 1.4 | Base on the 1.1~1.3, choose direction and find the ideas to do this research. | 19/09/27~  19/10/03 | 100% |
| 2.  Modeling | 2.1 | Understanding the MEC computation-offloading model. | 19/10/03~  19/10/08 | 100% |
| 2.2 | Understanding the NOMA transmission model. | 19/10/08~  19/10/15 | 100% |
| 2.3 | Base on 2.1~2.2, combine them to NOMA-MEC model and optimize “t” “U**L**” “{S**i**}” parameter (EEO) in the model. | 19/10/15~  19/10/23 | 100% |
| 2.4 | Base on the model after optimization, optimize the ordering problem to minimize the energy consumption. (EEO-E) | 19/10/23~  19/10/29 | 100% |
| 3.  Solve the model &  Algorithm simulation | 3.1 | Break EEO-E into “EEO-Top” and “EEO-Bot” two layers to let EEO-E became solvable. | 19/10/29~  19/11/13 | 100% |
| 3.2 | Find the way to solve “EEO-Bot” convex problem and design the algorithm. | 19/11/13~  20/01/03 | 100% |
| 3.3 | Design the algorithm to optimize the result in 3.2 and solve “EEO-Top” problem. | 20/01/03~  20/01/22 | 50% |
| 3.4 | Design the algorithm to solve ordering problem (ORP) to find the minimization of energy consumption in single mobile user -multitasking -multi-server transmission base on NOMA-MEC. | 20/01/22~  20/02/01 | 0% |
| 4.  Results verification & visualization | 4.1 | Compare with the traditional transmission to verify advantage of this program. | 20/02/01~  20/02/05 | 0% |
| 4.2 | Design and code the GUI to make the result visualization. | 20/02/05~  20/02/20 | 0% |
| 4.3 | Test the Demo and Make video. | 20/02/20~  20/03/01 | 0% |
| 5.  Conclusion & Finish | 5.1 | Summary the work and finish the final report. | 20/03/01~  20/03/20 | 0% |
| 5.2 | Prepare the presentation of this project. | 20/03/20~  Finish | 0% |

### CHAPTER 3. Meeting Summary:

|  |  |  |  |
| --- | --- | --- | --- |
| Meeting No. | Date  YY/MM/DD | Member | Content |
| 1 | 19/09/10 | Prof.Wu  Kin | Matched the fyp project topic and the general direction of the work. |
| 2 | 19/09/21 | Prof.Wu  Kin | Talk more information about MEC and set the target to read two paper in next two week. |
| 3 | 19/09/27 | Prof.Wu  Kin | Explain some keywords in MEC and report the reading progress. |
| 4 | 19/10/03 | Prof.Wu  Kin | Talk some applications in MEC,and set the target to build the reference/pre-review table after reading. |
| 5 | 19/10/08 | Prof.Wu  Kin | Report the reading progress and get one new paper to read, begin to study the NOMA background knowledge. |
| 6 | 19/10/15 | Prof.Wu  Kin | Explain the concept about NOMA and set reading target <<Deisgn and implementation of edge computing algorithm based on non-orthogonal multiple access>> |
| 7 | 19/11/13 | Prof.Wu  Kin | Explain the algorithm in last paper. Begin to run the code in that paper. |
| 8 | 19/11/18 | Prof.Wu  Kin | Explain the formula about energy consumption in NOMA-MEC. |
| 9 | 19/11/19 | Prof.Wu  Kin | Complete the analysis of the EEO-B algorithm and set a target to do simulation of that algorithm. |
| 10 | 20/01/08 | Prof.Wu  Kin | Show the result of the simulation algorithm and discuss how to do the interim report. |
| 11 | 20/01/17 | Prof.Wu  Kin | Begin to do simulation of linear search algorithm, talk more about the interim report. |
| 12 | 20/01/18 | Prof.Wu  Kin | Talk more details of the report and make the overall idea clearer. |

### CHAPTER 4. Mean Modeling:

In the non-orthogonal multiple access transmission mode, {, , ...., } is simultaneously migrated to each edge server, and the power consumption will be affected by the channel power gain factor. We use to indicate the user to the edge server Channel power gain of i. We assume that the order of the edge servers is based on:

Assuming the transmission duration is t, according to this formula, we can get the minimum total power for the user to send to the edge server I:

Where W is the channel bandwidth, parameter represents the power density of the background noise, and is set to 0. That is, the total power of the user using NOMA transmission tasks is

Based on non-orthogonal multiple-access transmission, this paper studies the edge computing of single-user, multi-tasking, and multi-server. In order to find its energy-efficient optimization solution, that is, to minimize the total energy consumption of mobile users to complete computing tasks, we can do three aspects to get start:

1. How much migrates per task

2. Transfer time of each task migration to the edge server t

3. Which server should mobile user tasks be migrated to for calculation

If you want to solve problems 1 and 2, you need to optimize Si and transmission time t to minimize its energy consumption. This article assumes that there are I computing tasks on a mobile device of a single mobile user, which are represented by I = {1,2, ..., I}, where the total task amount of each task is represented by, because Part of the task amount is migrated, so each task i is divided into two parts, one part is the task amount that needs to be migrated to the edge server through non-orthogonal multiple access, and the other part is the local computing task amount .

For each edge server , there is a fixed calculation rate, expressed as , and is used to indicate the local calculation rate on the terminal device. In order to clearly indicate that we measure by and bits / s.

Because the power consumption of the processor can be modeled as a cubic relationship with respect to its calculation rate. So, we can get the total energy consumption of mobile users to complete all remaining tasks locally:

Where is used to represent the coefficient determined by the CPU chip architecture.

This formula is constrained by

Variables in this formula:

After optimizing according to the above conditions, we will get (EEO-E):

Then we drive into objective 3 and found that (EEO-E) can be decomposed into 2 layers to solve this non-convexity problem of (EEO-E).

### CHAPTER 5. Tools & Methods used in this project

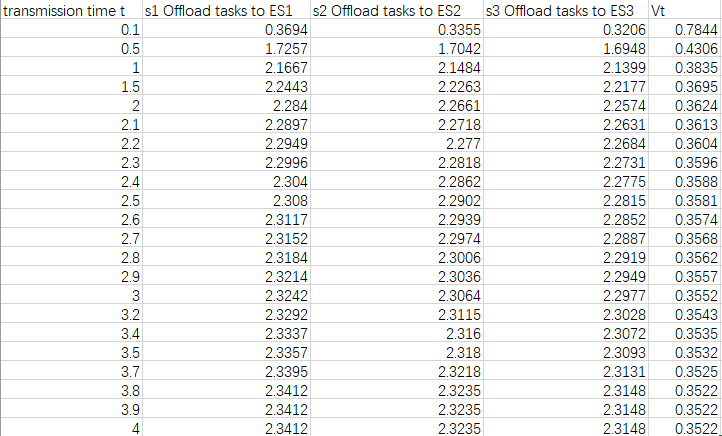
|  |  |  |
| --- | --- | --- |
| Task | Content | Tools |
| 3.2 | Find the way to solve “EEO-Bot” convex problem and design the algorithm. | 1.use CVXPY to solve the convex optimization problem of fix t to find each Vt at every t  <https://www.cvxpy.org/examples/index.html> |
| 3.3 | Design the algorithm to optimize the result in 3.2 and solve “EEO-Top” problem. | 1.Linear search  Because t is scope, so we can use linear search algorithm to adjust t to find the result in EEO-E-BOT. |
| 3.4 | Design the algorithm to solve ordering problem (ORP) to find the minimization of energy consumption in single mobile user -multitasking -multi-server transmission base on NOMA-MEC. | 1.A3C algorithm / Simulated annealing algorithm  Using these two algorithms to find the optimal matching scheme in the task-mapping problem to get the minimization energy consumption. |
| 4.2 | Design a UI interface to show the result | 1.PYQT  Use this package to design a GUI to show result of this project |

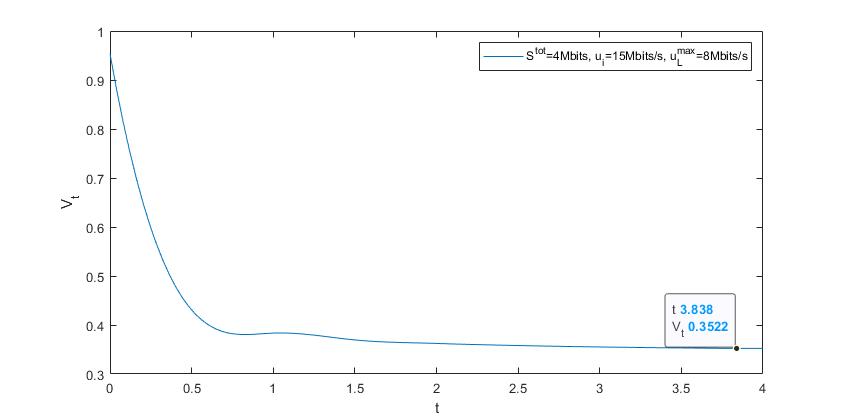
### CHAPTER 6. Simulation of EEO-Bot layer

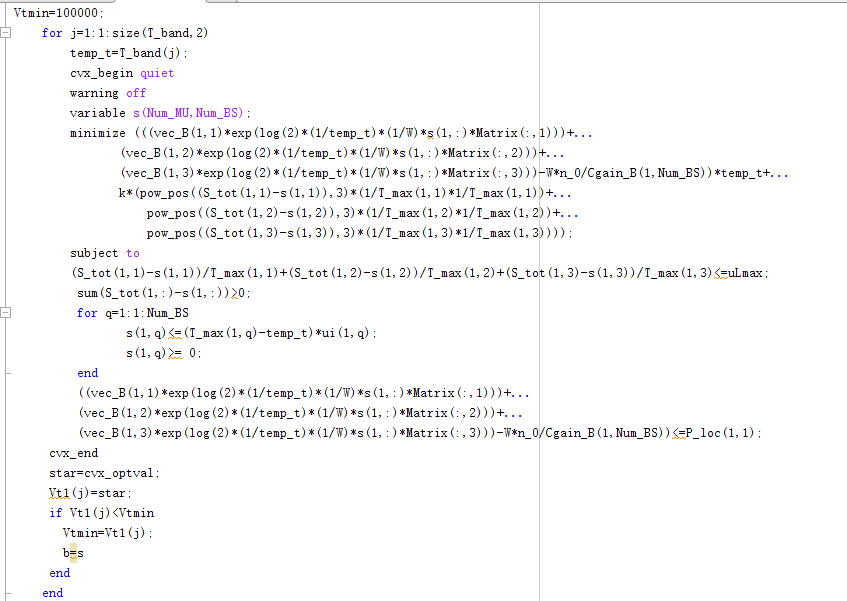
In order to verify the performance of the EM-i optimization model, we first set up a scenario with three base stations, where three base stations are located at (500,0), (-500,0), and (0,500). At the same time, mobile users are randomly distributed in a circle with (0,0) as the center and a radius of 100, and the continuous channel power gain to the base station is randomly generated according to the distance model.

Based on the above scenario, we will show one of the cases below. The channel gain is , the channel bandwidth is , and the power density of the background noise is. Set the calculation speed of the server . The maximum power budget for mobile users is. The total calculation amount for each task is. The maximum time limit for completing all tasks is .

When we give the transmission time , we can find and the corresponding .



In the following we will use a linear search method to change t with a very small step size. Since the range of t is limited, can be obtained to obtain the EEO-bot result.



### CHAPTER 7. Member contribution

Group J only has one member under the guidance of Professor Wu.

### CHAPTER 8. Conclusion and future plan

In general, the work is smooth. Although there are some challenges in the process, the task can be successfully completed after discussion and guidance with the professor. I will speed up to complete more tasks, and strive for more time to fill in gaps and make this project even better.

### CHAPTER 9. Reference

1.Mao, Y., You, C., Zhang, J., Huang, K., & Letaief, K. (2017). A Survey on Mobile Edge Computing: The Communication Perspective. IEEE Communications Surveys & Tutorials, 19(4), 2322-2358. doi: 10.1109/comst.2017.2745201

2.Shi, W., Cao, J., Zhang, Q., Li, Y., & Xu, L. (2016). Edge Computing: Vision and Challenges. IEEE Internet Of Things Journal, 3(5), 637-646. doi: 10.1109/jiot.2016.2579198

3.张平, 陶运铮, 张治. 5G 若干关键技术评述[J]. 通信学报, 2016, 37(7):15-29.

4.于天琪, 朱咏絮, 王现斌. 基于边缘计算的物联网监测系统中利用自编码神经网络

实现的异常检测[J]. 物联网学报, 2018, 2(04):18-25.

5.T. Beck, M. Werner, S. Feld, and S. Schimper, Mobile edge computing: A taxono

my[C], in Proc of the Sixth International Conference on Advances in Future Inter

net, 2014:1-7.

6.施巍松, 张星洲, 王一帆, et al. 边缘计算：现状与展望[J]. 计算机研究与发展, 201

9, 56(1):69-89.

7.F. Giust, X. Costa-Perez, and A. Reznik, “Multi-Access Edge Computing:

An Overview of ETSI MEC ISG,” IEEE 5G Tech Focus, vol. 1, no. 4,

Dec. 2017.

8.X. Chen, L. Jiao, W. Li, and X. Fu. Efficient multi-user computation offloading f

or mobile-edge cloud computing. IEEE/ACM Trans. Netw, 2016, 24(5):2795-2808.

9.L. Huang, S. Bi, and Y.J. Zhang, “Deep Reinforcement Learning

for Online Computation Offloading in Wireless Powered Mobile-Edge

Computing Networks,” IEEE Trans. on Mobile Computing, DOI:10.1109/

TMC.2019.2928811, July 2019.

10.J. Liu. G. Shou, Y. Liu, Y. Hu, and Z. Guo, “Performance Evaluation

of Integrated Multi-Access Edge Computing and Fiber-Wireless Access

Networks,” IEEE Access, vol. 6, pp. 30269-30279, May 2018.

11.Ding Z , Liu Y , Choi J , et al. Application of Non-Orthogonal Multiple Access

in LTE and 5G Networks[J]. IEEE Communications Magazine, 2017, 55(2):185-19

1.

12.Huang H , Xiong J , Yang J , et al. Rate Region Analysis in a Full-Duplex-Aide

d Cooperative Non-Orthogonal Multiple Access System[J]. IEEE Access, 2017, 5:1

7869–17880.

13.Zhang Y , Wang H M , Zheng T X , et al. Energy-efficient transmission design

in non-orthogonal multiple access [arXiv][J]. arXiv, 2017, 66(3):2852-2857.

14.Fang F , Zhang H , Cheng J , et al. Energy-Efficient Resource Allocation for Do

wnlink Non-Orthogonal Multiple Access (NOMA) Network[J]. IEEE Transactions o

n Communications, 2016, 64(9):3722-3723.

15.Z. Ding, D.W.K. Ng, R. Schober, and H.V. Poor, “Delay Minimization

for NOMA-MEC Offloading,” IEEE Signal Processing Letters, vol. 25,

no. 12, Dec. 2018.

16.Wu Y , Qian L P , Mao H , et al. Optimal Power Allocation and Scheduling for

Non-Orthogonal Multiple Access Relay-Assisted Networks[J]. IEEE Transactions o

n Mobile Computing, 2018, 17(11):2591–2606.

17.Ding Z, Fan P, and Poor H. Impact of non-orthogonal multiple access on the offl

oading of mobile edge computing. unpublished paper, 2018. [Online]. Available: h

ttps://arxiv.org/abs/1804.06712