End-to-end Network Resource Slicing in Mobile Edge Computing for 5G New Radio

Yan Kyaw Tun, Shashi Raj Pandey and Choong Seon Hong
Department of Computer Science and Engineering, Kyung Hee University, South Korea
ykyawtun7@khu.ac.kr, shashiraj@khu.ac.kr, and cshong@khu.ac.kr

Abstract

Wireless network slicing is one of the promising technologies for 5G new radio network to enhance the network capacity and the peak data rate of the current LTE network. Meanwhile, the upcoming technology so called mobile edge computing (MEC) offloading offloads the computation intensive task on the mobile device to the cloud server located at the edge of the cellular network. By offloading, MEC network can enhance the the computation capacities of the mobile devices and prolong the battery lives. However, there are still several challenges to address before deploying the aforementioned technologies in telecommunication industry. Among them, the efficient resource allocation is the most important issue. Therefore, in the work, we propose the energy efficient ene-to-end network slicing problem in the MEC network. Then, we deploy the block coordinate descent (BCD) approach to address the proposed problem. Simulation results show that our proposed scheme outperforms the existing schemes.

Keywords - End-to-end network slicing, mobile-edge computing(MEC), block coordinate descent (BCD).

1. Introduction

In order to address problems introduced by the exponential growth of the mobile devices with the emerging mobile applications (e.g., face recognition, virtual reality (VR), Augmented reality (AR) and so on), researchers in the telecommunication industry are proposing new technologies such as wireless network slicing, mobile-edge computing. Network slicing enables decoupling the current cellular network into two entities, such as infrastructure provider (InP) and mobile virtual network operators (MVNOs) where InP provides infrastructures and wireless network resources to the MVNOs and MVNOs provide different services (i.e., ultra-reliable and low latency communication (URLLC), enhanced mobile broad-band (eMBB), and massive machine type communication (mMTC)) to their mobile users [1, 2, 3]. Meanwhile, mobile-edge computing becomes the new and key technology for 5G new radio. Thereby, MEC can reduce the delay experienced by the mobile devices when compared with the mobile cloud computing, and fog computing [4, 5, 6]. However, the computation capacity (i.e., CPU) of the MEC is limited. Therefore, how to efficiently offload the computation intensive tasks of the mobile devices to the sever at the edge of the radio access network becomes important issue in the MEC network.

2. System Model and Problem Formulation

As shown in Fig. 1, an infrastructure provider (InP) deploys the macro base station (BS) integrated with the edge server. So, we will use the term edge server and base station interchangebaly. Then, the physical infrastucture is split into multiple virtual networks to support heterogeneous mobile services requests that are categorized into MEC service slice and traditional cellular service slice. Then, the InP will allocate each resource slice including cellular resource slice and MEC resource slice to each mobile virtual network operator. So, we will use the term slice and MVNO interchangeably throughout the paper. Mobile virtual network operators, i.e., service providers, are providing multiple services to their mobile users where each service has its own QoS requirements.

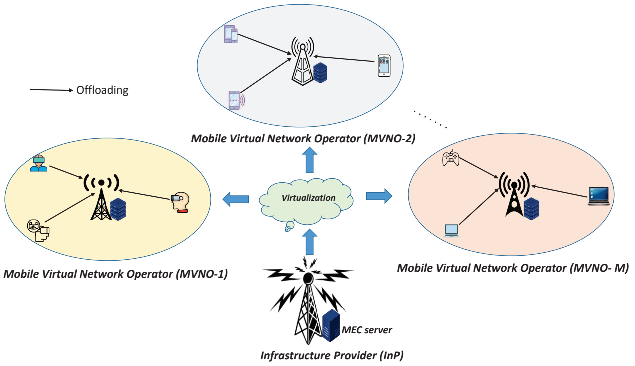
Offloading
Mobile Virtual Network Operator (MVNO-2)
Virtualization
Mobile Virtual Network Operator (MVNO-1)
Mobile Virtual Network Operator (MVNO- M)
MEC server
Infrastructure Provider (InP)

Figure 1: System Model

In this model, we assume that the base station is operating on the system bandwidth B and it is divided into the subchannels where each subchannel has the bandwidth ω. Moreover, an InP is providing the virtual networks/ resource slices (including communication resources such as sub-channel and transmit power, computation resource) to the M mobile virtual network operators/ service provider (SPs), denoted as a set $\mathcal{M} = \{1, 2, \dots, M\}$. Then, MVNOs are providing mobile services to their mobile users and each MVNO has U mobile users, denoted as a set $\mathcal{U} = \{1, 2, \dots, U\}$. Here we assume that each mobile user of each MVNO is generating a task a_{mu} to be executed. Then, the properties of a task of each mobile user in each MVNO can be represented as a tuple (d_{mu}, c_{mu}) where d_{mu} is the total data size (i.e., bits) of the task of user u of MVNO m, c_{mu} is the required CPU cycle to execute a bit of data. For each user of each MVNO, its computation task can be executed locally on mobile device or executed remotely at the MEC server. Therefore, let us introduce a binary variable x_{mu}, where $x_{mu} = 1$ indicates that the generated task is offloaded to the server, $x_{mu} = 0$ otherwise.

1) Local Computing: When the generated task of the user

u of the MVNO m is executed locally, the latency to complete the task execution can be formulated as follows:

$$t_{mu}^{L} = \frac{d_{mu}c_{mu}}{f_{mu}^{l}}, \forall u \in \mathcal{U}, \forall m \in \mathcal{M}, \tag{1}$$

where f_{mu}^{l} is the computation capacity of user u of the MVNO m. Moreover, the energy consumption of the user u can be expressed as follows:

$$E_{mu}^{L} = k(f_{mu}^{l})^{2}c_{mu}d_{mu}, \forall u \in \mathcal{U}, \forall m \in \mathcal{M}, \tag{2}$$

where $k = 10^{-26}$ and it depends on the chip architecture of the mobile device. Then, the local computation overhead of the user u of MVNO m can be formulated as follows:

$$G_{mu}^{L} = \lambda_{t}t_{mu}^{L} + \lambda_{e}E_{mu}^{L}, \forall u \in \mathcal{U}, \forall m \in \mathcal{M}, \tag{3}$$

where λ_t and λ_e of the weighted parameters for local computation latency and local energy consumption.

2) Remote Computing: When the generated task of the user u of the MVNO m is executed remotely, the mobile user firstly transmits the task (i.e., input data) to the edge server. Therefore, the achievable data rate of the user u of the MVNO m is as follows:

$$R_{mu} = \omega \log_{2}(1 + \frac{p_{mu}h_{mu}}{N_0}), \forall u \in \mathcal{U}, \forall m \in \mathcal{M}, \tag{4}$$

where p_{mu} is the transmit power of the user u of MVNO m, h_{mu} is the achievable channel gain, and N_0 is the additive white Gaussian noise power. Then, we can formulate the transmission latency/delay experienced by the user u of MVNO m as follows:

$$t_{mu}^{O} = \frac{d_{mu}}{R_{mu}}, \forall u \in \mathcal{U}, \forall m \in \mathcal{M}. \tag{5}$$

Moreover, in this work, we assume that the computation capacity of the edge server is infinite. Therefore, the execution delay of the task of the user u in MVNO m can be ignored. Then, the energy consumption for the uplink transmission as follows:

$$E_{mu}^{O} = \frac{p_{mu}d_{mu}}{\omega \log_{2}(1 + \frac{p_{mu}h_{mu}}{N_0})}, \forall u \in \mathcal{U}, \forall m \in \mathcal{M}, \tag{6}$$

Finally, we can formulate the remote computation overhead experienced by the user u of MVNO m as follows:

$$G_{mu}^{O} = \lambda_{t}t_{mu}^{O} + \lambda_{e}E_{mu}^{O}, \forall u \in \mathcal{U}, \forall m \in \mathcal{M}. \tag{7}$$

In this work, the efficient generated task offloading and power allocation for the end-to-end network slicing in the MEC network is formulated as the optimization problem. The objective is to minimize the computation overhead experienced by the users of all MVNOs. Under the efficient offloading decision constraint, and power budget constraint of each mobile user, the optimization problem can be expressed as follows:

$$\min_{\mathbf{x},\mathbf{p}} \left(\sum_{m=1}^{M}\sum_{u=1}^{U} x_{mu}G_{mu}^{O} + \sum_{m=1}^{M}\sum_{u=1}^{U}(1 - x_{mu})G_{mu}^{L} \right) \tag{8}$$

$$\text{s.t. C1} : x_{mu} \in \{0,1\}, \quad \forall u \in \mathcal{U}, \forall m \in \mathcal{M}, \tag{9}$$

$$\text{C2} : 0 \le p_{mu} \le P_{mu}^{\mathbf{max}}, \quad \forall u \in \mathcal{U}, \forall m \in \mathcal{M}, \tag{10}$$

where $P_{mu}^{\mathbf{max}}$ is the maximum transmit power of the user u of the MVNO m. Then, C1 represents the task offloading constraint, and C2 shows the power budget constraint of each mobile user in MVNOs. We can see that aforementioned optimization problem is the mixed integer and non-convex problem. Generally, it is difficult to solve. Therefore, we divide the original problem into two subproblems and each subproblem becomes convex. Then, we provide the close-form solution for each subproblem.

2.1 Task Offloading Problem (TOP)

In a given power allocation, the optimization problem expressed in the (8) can be transformed into the task offloading problem and it is as follows:

$$\min_{\mathbf{x}} \left(\sum_{m=1}^{M}\sum_{u=1}^{U} x_{mu}G_{mu}^{O} + \sum_{m=1}^{M}\sum_{u=1}^{U}(1 - x_{mu})G_{mu}^{L} \right) \tag{11}$$

$$\text{s.t. C1} : x_{mu} \in [0,1], \quad \forall u \in \mathcal{U}, \forall m \in \mathcal{M} \tag{12}$$

where we firstly relax the offloading variable (i.e., binary variable) into the continuous form and the task offloading problem becomes convex problem. So, we can use the CVX solver to solve it.

2.2 Power Allocation Problem (PAP)

In a given task offloading, the optimization problem mentioned in (8) can be transformed into the power alloation problem and it is as follows:

$$\min_{\mathbf{p}} \left(\sum_{m=1}^{M}\sum_{u=1}^{U} x_{mu}G_{mu}^{O} + \sum_{m=1}^{M}\sum_{u=1}^{U}(1 - x_{mu})G_{mu}^{L} \right) \tag{13}$$

$$\text{s.t. C2} : 0 \le p_{mu} \le P_{mu}^{\mathbf{max}}, \quad \forall u \in \mathcal{U}, \forall m \in \mathcal{M}, \tag{14}$$

where the above subproblem so called power allocation problem is also convex. Therefore, we can use the CVX solver to solve the aforementioned sub-problem in (13). However, we omit the proofs of convexity for two subproblems because of the limited space. We can get the solution of the optimization problem by solving the the above two subproblems alternatively.

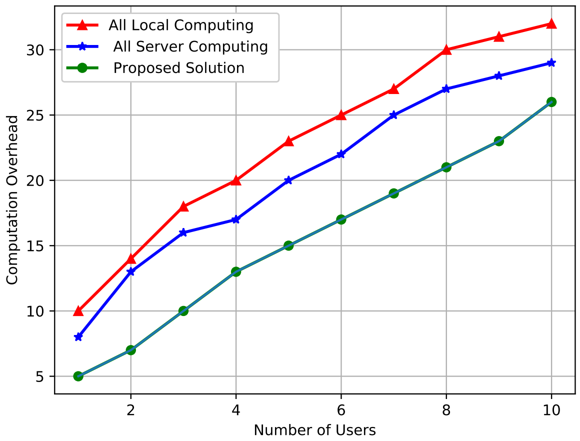

Figure 2: Tradeoff betwee the computation overhead and number of users

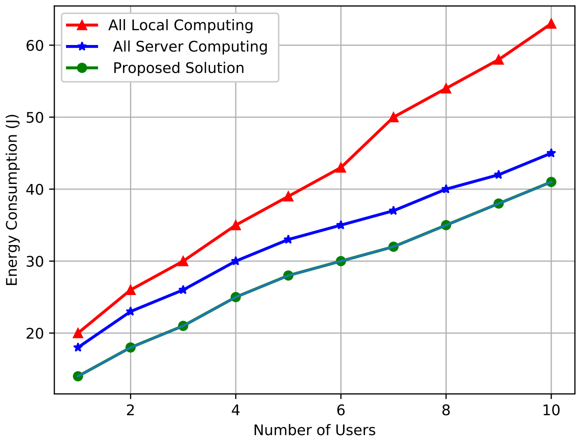

Figure 3: Tradeoff between energy consumption and number of users

3. Simulation Results

In our simulation section, we consider a single BS with the total system bandwidth 20MHz and the bandwidth of each subchannel is 180kHz. The additive white Gaussian noise power is $-174dBm/Hz$. The long distance path loss model in our simulation is $PL = 40\log_{10}(d_0) - 10\log_{10}(Gh_t^2h_r^2) + 10\gamma\log_{10}\frac{d}{d_0} + X_g$ where G is the gain product of transmitter and receiver, d_0 and d are the reference distance and actual distance between transmitter and receiver, h_t and h_r are heights of transmitter and receivers, and X_g is the random variable. Moreover, we consider the BS is serving 2 MVNOs and there are 5 mobile users in each MVNO. The maximum transmit power and the maximum computation (i.e., CPU) capacity of each mobile device is 1mW and 0.2GHz. The total input data size of each mobile device is 0.5MB and the required CPU capacity to execute one bit of input data is 500 cycles. In case of simplicity, in this simulation setting, we consider both λ_t and λ_e are 1. Fig. 2 shows the trade off between computation overhead and number of users. We can see that computation overhead depends on the number of users. When the number of users increases, the computation overhead also increases. Moreover, we compare our proposed algorithm with the all local computing and all server computing. It can be seen from Fig. 2 that our proposed solution outperforms other two schemes. The reason is that when each mobile user executes its computation task locally, more energy is consumed. Another one, the computation task offloaded to the edge-server has high latency. This is why, all local computing and all server computing is resulting in the high computation overhead.

Fig. 3 represents the trade-off between energy consumption and number of users of all MVNOs. From Fig. 3, we can see that the energy consumption is the highest when users of MVNOs execute their tasks locally (i.e., all local computing). Moreover, the energy consumption is higher than our proposed solution and lower than local computing when all users offload their computation tasks to the edge-server because of the energy consumption for uplink transmission.

4. Conclusion

In this work, we propose energy efficient computation task offloading and end-to-end network slicing scheme in the mobile edge computing (MEC). Then, we formulate the optimization problem as a minimizing computation overhead problem under the efficient offloading decision constraint, and power budget constraint of each mobile device. In the simulation section, we can see that our proposed solution outperforms other schemes. Moreover, our work is the very first work that consider both network slicing and mobile edge computing at the same time. In future, we will extend our work into multiple base stations (BSs) and multiple mobile-edge severs case.

References

[1] M. Alsenwi, N. H. Tran, M. Bennis, A. K. Bairagi, and C. S. Hong, "eMBB-URLLC resource slicing: A risk-sensitive approach," *IEEE Communications Letters*, vol. 23, no. 4, pp. 740–743, apr 2019.

[2] M. A. C. W. Z. C. S. H. Yan Kyaw Tun, Shashi Raj Pandey, "Weighted proportional allocation based power allocation in wireless network virtualization for future wireless networks," in *The 33rd International Conference on Information Networking (ICOIN 2019), IEEE*, 2019.

[3] Y. K. Tun, C. W. Zaw, and C. S. Hong, "DownlinK power allocation in virtualized wireless networks," in *2017 19th Asia-Pacific Network Operations and Management Symposium (APNOMS)*. IEEE, sep 2017.

[4] 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, 2017.

[5] 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.

[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.