A Distributed Game Theoretic Algorithm for Minimizing Delay in Mobile Edge Computing

[†]Chit Wutyee Zaw, [*]Choong Seon Hong

Department of Computer Science and Engineering, Kyung Hee University,
Yongin, 446-701 Korea
{[†]cwyzaw, [*]cshong }@khu.ac.kr

Abstract

Mobile Edge Computing (MEC) can grant the low latency, high computation ability to resource constrained mobile users by providing servers at the edge of wireless networks. Because of the low latency applications, users require their task computation time as minimum. Delay in computation offloading is influenced by two factors, uplink offloading time and processing time. Since the uploading time is mainly affected by power of mobile users which is limited and the interference of others, there is a trade-off between the user's transmit power and uploading time. Moreover, the computational capability of the MEC server is restricted. Thus, users have to compete each other to access the computation resources. We formulate the uplink power allocation and computation resource allocation as Generalized Nash Equilibrium Problem (GNEP) where users want to minimize their delay which is controlled by their transmit power and the interference of other users while competing for the limited computation resources.

1. Introduction

Mobile Edge Computing (MEC) brings servers closer to users by deploying them at base stations (BSs). It can help users to process the real time applications such as Virtual Reality (VR) and Augmented Reality (AR) which require low latency. Users can save the energy of their mobile phones and their time by offloading the computation intensive tasks to servers. Since the servers are located at the base stations, the delay is much reduced than offloading to the traditional cloud servers which are located far from users.

MEC use cases and scenarios are surveyed in [1] where authors compared the difference between Mobile Cloud Computing MCC, where servers can only be accessed by the Internet, and MEC. [2] studied the energy efficient joint radio and computation resource management for multi-user MEC systems with a stochastic approach. A game theoretic formulation for offloading decisions can be found in [3]. The authors considered both delay and energy minimization but they do not include the limitation of computation resource allocation at the MEC server.

In this paper, we formulate the uplink power and computation resource allocation for minimizing delay as Generalized Nash Equilibrium Problem (GNEP). The reason why we use the GNEP is that the computation resource allocation is coupled among users which means users' strategy sets are dependent on each other [4]. We propose the distributed uplink power and computation resource allocation algorithm which uses the Nonlinear Jacobi-type method [4].

2. System Model

A macro base station (MBS) with a MEC server is considered in this paper. We assume that users have some tasks to offload to the MEC servers. Users share the same bandwidth for the uplink transmission so that there is an interference among them. We also assume that the MEC servers has the capability of processing multiple tasks simultaneously.

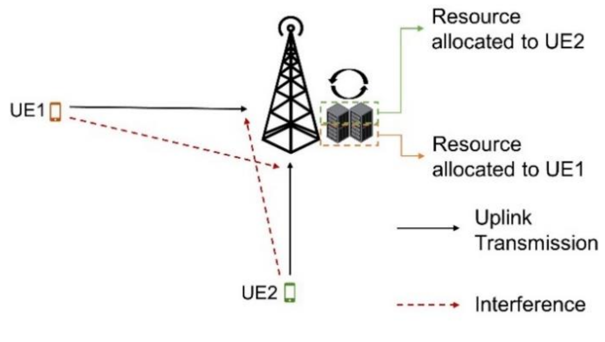

Fig. 1. MEC server collocated with MBS

A. Communication Model

The data rate of uplink transmission can be calculated as follows.

$$R_i = W \ \log_2(1 + \frac{p_i g_i}{\sum_{j \neq i} p_j g_j + n_0})$$

where W is the uplink bandwidth and p_i is the uplink transmit power of user i. Thus, the time to transmit the file with the size, b_i, is

$$t_i^{ul} = \frac{b_i}{R_i}$$

B. Computing Model

The time required to process the task which needs c_i CPU cycles can be calculated as

$$t_i^p = \frac{c_i}{f_i}$$

where f_i is the number of CPU cycles per second assigned to user i.

3. Game Formulation

The power control and computation resource allocation to users for minimizing delay is formulated as Generalized Nash Game. Players of the game are users and each user, i, solve the following optimization problem.

$$\min_{p_i, f_i} t_i^{ul} + t_i^p$$

s.t.

$$p_i \leq p_{max}$$

$$\sum_{i=1}^{N} f_i \leq f_{max}$$

The objective is to minimize the delay. The first constraint is the power budget of each user and the second constraint ensures that the computation resources allocated to all users must not exceed the resources of the MEC server.

This problem is formulated as GNEP because the objective function of user i depends on other players' strategies because of the interferences. Moreover, the user i's strategy set is coupled with other players' strategies in the CPU budget constraint.

The independent strategy set is defined as follows.

$$\mathcal{S}_i = \{p_i: \ p_i \leq p_{max}\}$$

The coupling constraint can also be written as

$$g_i(f_{-i}) := \sum_{i=1}^{N} f_i - f_{max} \leq 0$$

The coupled strategy set of user i is written as

$$\mathcal{Q}_i(f_{-i}) = \{p_i, f_i: \ p_i \leq p_{max}, g_i(f_{-i}) \leq 0 \ \}$$

The objective function is redefined as

$$h_i(p_i, f_i, f_{-i}) = t_i^{ul} + t_i^p$$

The Game can be formulated as

$$\min_{p_i, f_i \in \mathcal{Q}_i(f_{-i})} h_i(p_i, f_i, f_{-i})$$

$\mathcal{Q}_i(f_{-i})$ is called the moving set because it depends on the other players' strategies, f_{-i}.

Existence of GNE: The objective function, $h_i(p_i, f_i, f_{-i})$, is differentiable and convex in p_i and f_i. The player's independent strategy set, $\mathcal{S}_i$, is compact and convex and the coupling function, $g_i(f_{-i})$, is convex in f_i. So, there exists GNEs.

The GNEPs are difficult to solve because of their coupling constraints. This makes them different from NEPs where the players' strategy sets are fixed in NEPs. We propose the distributed algorithm in next section.

4. Distributed Power and Resource Allocation Algorithm

The uplink power and computation resource allocation algorithm is proposed using the Nonlinear Jacobi type method. Since each user's optimization problem is convex, we solve it using cvxpy [5].

Distributed Power and Resource Allocation Algorithm

1:	Choose the initial allocation $p^0 = (p_0^0, p_1^0, \ldots, p_N^0)$ and $f^0 = (f_0^0, f_1^0, \ldots, f_N^0)$
2:	k = k + 1
3:	Each user solves
4:	$\min_{p_i, f_i \in \mathcal{Q}_i(f_{-1}^{k-1})} h_i(p_i, f_i, f_{-i}^{k-1})$
5:	$p_i^k = p_i, \ f_i^k = f_i$
6:	Exchange $p_i^k, \ f_i^k$
7:	k = k + 1
8:	At each user's side,
9:	**while** ($h_i^k(p_i, f_i, f_{-i}^{k-1}) \geq h_i^{k-1}(p_i, f_i, f_{-i}^{k-2})$ or $\sum_{i=1}^{N} f_i^{k-1} - f_{max} \leq 0$)
10:	solve
11:	$\min_{p_i, f_i \in \mathcal{Q}_i(f_{-1}^{k-1})} h_i(p_i, f_i, f_{-i}^{k-1})$
12:	$p_i^k = p_i, \ f_i^k = f_i$
13:	Exchange $p_i^k, \ f_i^k$
14:	k = k + 1
15:	**end while**

5. Evaluation

In evaluation, one MBS and one MEC server with 2GHz CPU. The maximum transmit power of each user is 0.1 watt. We use the power density thermal noise as −174dBm/Hz and the long distance path loss model. The input file size is uniformly distributed between 35 KBs and 60 KBs. The required CPU cycles is taken from a uniform distribution of [1.5e2 kHz, 2.5e2 kHz].

We compare our distributed algorithm with a centralized solution which is an optimizer included in PyOpt package [6] and equal allocation where users transmit at their maximum power and CPU resources are shared equally.

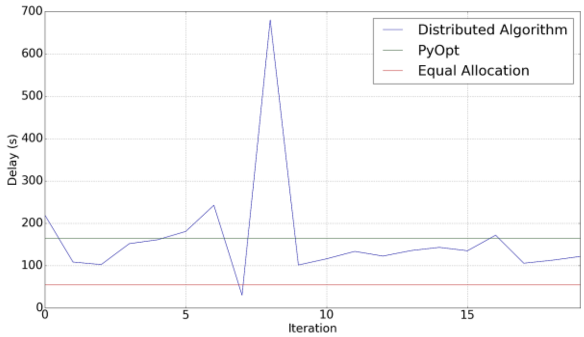

Fig. 2. Delay Comparison

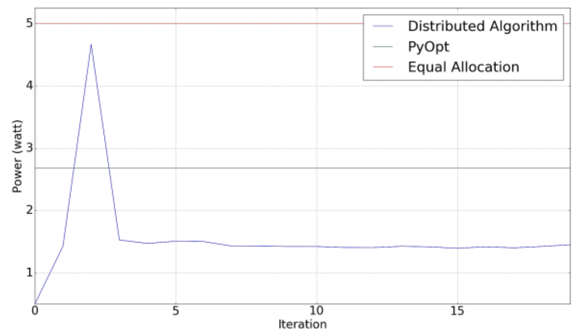

Fig. 3. Uplink Power Consumption Comparison

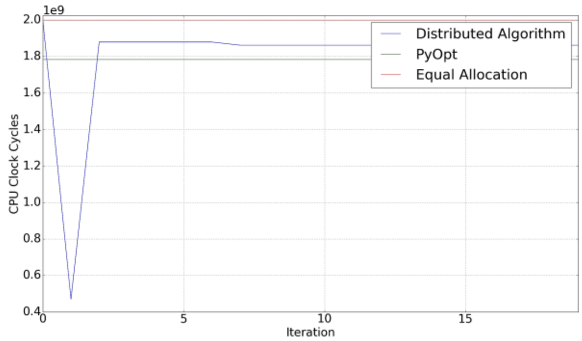

Fig. 5. CPU Clock Cycles Comparison

As we can see in figures, the centralized solution outperforms in resource allocation where users can need less power to transmit and CPU resource of the MEC server do not need to be fully allocated. But, the equal allocation outperforms in delay comparison which requires users to transmit at maximum power. Our distributed algorithm is comparable to the two approaches with less transmit power.

6. Conclusion

In this paper, we proposed a distributed power control and computation resource allocation for minimizing delay in Mobile Edge Computing environment. The problem is formulated as Generalized Nash Equilibrium Problem where both objective function and the strategy sets of players depend on each other. To solve the GNEP, we use the Nonlinear Jacobi type method for distributed algorithm. Simulation is performed to show the comparison between proposed distributed algorithm, a centralized solution and equal allocation for the coupled constraint.

Acknowledgement

This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT) (NRF-2017R1A2A2A05000995). *Dr. CS Hong is the corresponding author.

References

[1] P. Mach and Z. Becvar, "Mobile Edge Computing: A Survey on Architecture and Computation Offloading," *IEEE Communications Surveys & Tutorials,* vol. 19, no. 3, pp. 1628-1656, 2017.

[2] Yuyi Mao, Jun Zhang, S. H. Song, Khaled B. Letaief, "Stochastic Joint Radio and Computational Resource Management for Multi-User Mobile-Edge Computing Systems," *IEEE Transactions on Wireless Communications,* vol. 16, no. 9, pp. 5994-6009, 2017.

[3] 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, 2016.

[4] Francisco Facchinei, Christian Kanzow, "Generalized Nash Equilibrium Problems," *4OR,* vol. 5, no. 3, pp. 173-210, 2007.

[5] Steven Diamond and Stephen Boyd, "CVXPY: A Python-Embedded Modeling Language for Convex Optimization," *Journal of Machine Learning Research,* vol. 17, no. 83, pp. 1-5, 2016.

[6] Ruben E. Perez and Peter W. Jansen and Joaquim R. R. A. Martins, "pyOpt: A Python-Based Object-Oriented Framework for Nonlinear Constrained Optimization," *Structures and Multidisciplinary Optimization,* vol. 45, no. 1, pp. 101-118, 2012.