Resource Allocation for Multi-access Edge Computing with Coordinated Multi-Point Reception

Jian-Jyun Hung*, Wanjiun Liao*, and Yi-Han Chiang†

*Department of Electrical Engineering, National Taiwan University, Taipei, Taiwan

†Information Systems Architecture Science Research Division, National Institute of Informatics, Tokyo, Japan

Email: {r06942077,wjliao}@ntu.edu.tw, yhchiang@nii.ac.jp

***Abstract*—Multi-access edge computing (MEC) has emerged as a promising platform to provide user equipments (UEs) with timely computational services through the deployed edge servers. Typically, the size of an uplink task data (e.g., images or videos) required for processing is more pronounced than that of a downlink task result, and hence MEC offloading (MECO) plays a decisive role in the efficiency of MEC systems. In the light of an unprecedented growth of UEs in next-generation mobile networks, the reception of uplink signals at base stations (BSs) can be corrupted due to potential inter-user interference. To address this issue, coordinated multi-point (CoMP) reception which enables BSs to cooperatively receive uplink signals has evolved as an effective approach to enhance the received signal qualities. In this paper, we investigate a resource allocation problem for MECO with CoMP reception and formulate it as a mixed-integer non-linear program (MINLP). To solve this problem, we leverage the concept of interference graphs to characterize uplink inter-user interference, based on which we propose a resource allocation algorithm that consists of three phases: 1) computing resource allocation, 2) subcarrier allocation and cell clustering, and 3) subcarrier reuse and cell re-clustering. The simulation results show that our proposed solution can effectively enhance the delay performance of MECO through CoMP reception as compared with existing solution approaches under various system settings.**

***Index Terms*—Resource allocation, multi-access edge computing offloading (MECO), coordinated multi-point (CoMP) reception.**

I. INTRODUCTION

Multi-access edge computing (MEC) [1]–[4] is a promising solution for scenarios with low-delay requirements or to Internet of Things (IoT) devices with limited energy and computing capabilities. The idea behind MEC is to push computing services from the remote cloud to the proximity of user equipments (UEs), thereby reducing transmission delays to and from the cloud. One key feature of MEC is to enable UEs to offload computation-intensive tasks to nearby base stations (BSs). As such, UEs with limited computing resources and low battery lives could be complemented by MEC offloading (MECO) to facilitate emerging 5G services.

In MEC systems, UEs can offload their tasks to MEC servers, thereby accelerating task processing and prolonging their battery lives. In recent years, there have been many works on how to allocate communication and computing resources for single-cell MECO scenarios. In [5], You *et al.* studied resource allocation for multi-user single-BS MEC systems by considering both cases of infinite and finite cloud computation capacities. In [6], Mao *et al.* investigated the power-delay tradeoff in a multi-user MEC system. In [7], Li *et al.* investigated the joint subcarrier and power allocation problem in an orthogonal frequency division multiple access (OFDMA) based MEC system to minimize the maximal delay of mobile devices. In [8], Guo *et al.* considered an energy-efficient resource allocation for MEC where all UEs need to offload their computation-intensive tasks. In [9], Bi *et al.* studied a computation rate maximization problem in a multi-user MEC system with binary offloading policies. Though these works have been devoted to optimizing delay performance or energy efficiency for single-cell MECO scenarios, they neglect the multi-cell ones where inter-user interference in the uplink may adversely affect the efficiency of MECO.

In literature, several works have been focused on how to alleviate interference for multi-cell MECO scenarios. In [10], Guo *et al.* investigated the joint load management of resource allocation problem for multiple small BSs in an MEC system and the goal is to maximize the number of offloaded tasks. In [11], Wang *et al.* jointly considered computation offloading and interference management in a multi-cell heterogeneous MEC system. In [12], Chen *et al.* focused on an MECO problem in an ultra dense network, the goal of which is to minimize the effects of delay and interference. Despite the above works on multi-cell MECO scenarios, they do not tell us how to leverage coordinated multi-point (CoMP) reception [13], [14], which is to form a virtual antenna array composed of multiple geographically separated reception points (i.e. BSs) to jointly elevate uplink signal qualities.

In this paper, we investigate the resource allocation problem for MECO with CoMP reception (RAMCR) and formulate it as a mixed-integer non-linear program (MINLP). Then, we propose the interference graph based resource allocation (IGRA) algorithm to improve the efficiency of MECO in three phases: 1) computing resource allocation, 2) subcarrier allocation and cell clustering, and 3) subcarrier reuse and cell re-clustering. Our simulation results show that our proposed solution outperforms other comparison schemes in terms of total delays under various data sizes, BS and UE densities. The contribution of this paper is summarized as follows.

- We formulate the RAMCR problem as an MINLP and show its NP-hardness.
- We leverage the concept of interference graphs to characterize uplink inter-user interference for the RAMCR problem.

- We propose the 3-phase IGRA algorithm to allocate communication and computing resources, which is shown to effectively reduce total delays in our simulation results.

The rest of this paper is organized as follows. Sec. II describes the system model. Sec. III presents the RAMCR problem. In Sec. IV, we introduce the concept of interference graphs and present the IGRA algorithm. In Sec. V, we show the simulation results. Finally, Sec. VI concludes this paper.

II. System Model

We consider an OFDMA-based MECO system with CoMP reception in the uplink, where a set $\mathcal{M}$ of BSs is servicing a set $\mathcal{U}$ of UEs, and an MEC server is responsible for processing the offloaded tasks. The available network bandwidth is divided into a set $\mathcal{N}$ of subcarriers. To improve the efficiency of MECO, BSs are able to cooperate with each other to jointly receive uplink signals. In addition, we assume that the task of UE $u \in \mathcal{U}$ has a data size D_u (for communications) and a workload W_u (for computation).

With the uplink CoMP reception, the signal-to-interference-plus-noise (SINR) value experienced by UE u on subcarrier n is given (as [15]) by

$$\mathrm{SINR}_{un} = \frac{\sum_{m\in\mathcal{M}} \rho_{un} p_{un} h^2_{mun} x_{mu}}{n_0 + \sum_{m\in\mathcal{M}} x_{mu} \sum_{v\in\mathcal{U}\setminus\{u\}} \rho_{vn} p_{vn} h^2_{mvn}}, \quad \forall n \in \mathcal{N}, u \in \mathcal{U}, \tag{1}$$

where p_{un} is the transmission power allocated by UE u on subcarrier n, h^2_{mun} is the channel gain of UE u to BS m on subcarrier n, ρ_{un} indicates whether UE u uses subcarrier n, x_{mu} indicates whether UE u is served by BS m, and n_0 refers to the noise spectral density. The achievable transmission rate for each UE $u \in \mathcal{U}$ summarized over all subcarriers can be obtained as

$$R_u = B \sum_{n\in\mathcal{N}} \rho_{un} \log_2(1 + \mathrm{SINR}_{un}), \quad \forall u \in \mathcal{U}, \tag{2}$$

where B is the bandwidth of each subcarrier. Given (1) and (2), the transmission delay can be calculated as

$$t^{\mathrm{tran}}_u = \frac{D_u}{R_u}, \quad \forall u \in \mathcal{U}. \tag{3}$$

If the computing resources of the system are sufficient, each UE can offload its task to the MEC server, and the resulting execution delay is given by

$$t^{\mathrm{exe}}_u = \frac{W_u}{f_u}, \quad \forall u \in \mathcal{U}, \tag{4}$$

where f_u is the computing resource allocated to this task. If the system workload is heavy and the delay constraints of the UE cannot be satisfied when offloading, the UE will execute its task locally. The resulting local computing delay is

$$t^{\mathrm{loc}}_u = \frac{W_u}{f^{\mathrm{loc}}_u}, \quad \forall u \in \mathcal{U}, \tag{5}$$

where f^{loc}_u is the local computing power of UE u. The total delay of an upload task is the sum of the transmission delay and the execution delay, where the delay sent back from the MEC server is negligibly small [16].

TABLE I
Notation Summary

Notation	Definition
D_u, W_u	Data size and workload of UE u
$\mathcal{M}$	The set of BSs
$\mathcal{N}$	The set of subcarriers
$\mathcal{U}$	The set of UEs
B	Bandwidth of each subcarrier
$P^{\max}_u$	Maximum transmission power of UE u
ρ_{un}	Subcarrier assignment of UE u on subcarrier n
x_{mu}	Cell clustering policy from UE u to BS m
f_u	Computing resource allocation for UE u
f^{loc}_u	Local computing power of UE u
h^2_{mun}	Channel gain from UE u to BS m on subcarrier n
t^{tran}_u	Transmission delay of UE u
t^{exe}_u	Execution delay of UE u
t^{loc}_u	Local computing delay of UE u
F	Total available computing resource of the MEC server
T_u	Delay constraint of UE u
X	Cluster size limit of each UE

III. Problem Formulation

In this paper, we aim to seek an efficient resource allocation for MECO with CoMP reception. For brevity, we denote by

$$\boldsymbol{\rho} = \{\rho_{un}, \forall u \in \mathcal{U}, n \in \mathcal{N}\}, \quad \text{(subcarrier assignment)}$$
$$\boldsymbol{x} = \{x_{mu}, \forall m \in \mathcal{M}, u \in \mathcal{U}\}, \quad \text{(cell clustering policy)}$$
$$\boldsymbol{f} = \{f_u, \forall u \in \mathcal{U}\}. \quad \text{(computing resource allocation)}$$

Mathematically, we formulate the resource allocation problem for MECO with CoMP reception (RAMCR) to minimize the total delays of all tasks, as

$$\min_{\boldsymbol{\rho},\boldsymbol{x},\boldsymbol{f}} \sum_{u\in\mathcal{U}} t^{\mathrm{tran}}_u + t^{\mathrm{exe}}_u,$$
$$\text{s.t.} \sum_{u\in\mathcal{U}} \rho_{un} x_{mu} \le 1, \quad \forall m \in \mathcal{M}, n \in \mathcal{N}, \tag{6}$$
$$\sum_{n\in\mathcal{N}} p_{un}\rho_{un} \le P^{\max}_u, \quad \forall u, \tag{7}$$
$$\sum_{u\in\mathcal{U}} f_u \le F, \tag{8}$$
$$\sum_{u\in\mathcal{U}} x_{mu} \sum_{n\in\mathcal{N}} \rho_{un} \le |\mathcal{N}|, \quad \forall m \in \mathcal{M}, \tag{9}$$
$$\sum_{m\in\mathcal{M}} x_{mu} \le X, \quad \forall u \in \mathcal{U}, \tag{10}$$
$$t^{\mathrm{tran}}_u + t^{\mathrm{exe}}_u \le T_u, \quad \forall u \in \mathcal{U}, \tag{11}$$
$$\rho_{un} \in \{0, 1\}, \quad \forall n \in \mathcal{N}, u \in \mathcal{U}, \tag{12}$$
$$x_{mu} \in \{0, 1\}, \quad \forall m \in \mathcal{M}, u \in \mathcal{U}, \tag{13}$$
$$f_u \ge 0, \quad \forall u \in \mathcal{U}. \tag{14}$$

(6) ensures that each subcarrier can only be used by at most one UE in one BS. (7) is the UE transmission power constraint,

where $P_u^{\max}$ is the maximum transmission power of UE u. (8) is the MEC server computing resource constraint, where F is the total computing resource of the MEC server. (9) and (10) represent the constraints of the connection limits of BSs and the cluster limits of UEs, respectively, where X is the cluster size constraint of each UE. (11) is the delay constraint for each UE, where T_u is the delay constraint of UE u. (12)-(14) are the auxiliary constraints for the decision variables. Note that the RAMCR problem can be proved NP-hard by the reduction from the multiple knapsack problem (MKP) [17].

IV. Algorithm Design

To efficiently solve the RAMCR problem, we first introduce the interference graph concept in Sec. IV-A, and then design the interference graph based resource allocation (IGRA) algorithm for MECO with CoMP reception in Sec. IV-B. The time complexity of the IGRA algorithm is presented in Sec. IV-C.

A. Interference Graph

The interference graph is denoted by $G_n = (V, E)$ for each subcarrier $n \in \mathcal{N}$, where V is the set of nodes representing UEs and E is the set of edges representing inter-user interference. Although each subcarrier can only be used by one UE in a BS, it can be spatially reused by other BSs, as long as no severe interference takes place. For brevity, we denote by $\mathcal{M}_u$ the set of BSs that have UE u in their coverage areas. In addition, there is an edge between two nodes a and b in the interference graph if and only if $\mathcal{M}_a \cap \mathcal{M}_b \neq \varnothing$.

Fig. 1 illustrates how to construct an interference graph. In Fig. 1a, there are 4 BSs and 7 UEs, where each solid line indicates a UE located within the coverage of a BS. In this example, UE 1 is in the coverage of both BS 1 and BS 2, and hence $\mathcal{M}_1 = \{1, 2\}$. UE 4 is in the coverage of BSs 2 and 4, so $\mathcal{M}_4 = \{2, 4\}$. Since $\mathcal{M}_1 \cap \mathcal{M}_4 \neq \varnothing$, there is an edge connection between UEs 1 and 4, as depicted in Fig. 1b. Once if a subcarrier n is allocated to UE 1, UEs 2, 3, and 4 will be deleted from interference graph G_n, as these four nodes are all within the same coverage of BS 2. Therefore, we see in Fig. 1c that the subcarrier can be reused for UEs 5, 6 and 7.

B. IGRA – Interference Graph based Resource Allocation

If the delay constraint (11) of an uploaded task cannot be satisfied, the task will be executed locally. Otherwise, the UE will perform task offloading for computation. We present the IGRA algorithm in the following three phases.

1) Computing resource allocation: First, we consider the MEC computing resource allocation. In particular, we minimize the total execution delay of UEs' task under the MEC computing resource constraint (8). We allocate our MEC server computing resources by leveraging the arithmetic-geometric mean inequality. Specifically, we allocate the MEC computing resources proportionally to the requirements of all UEs. The computing resource allocation pseudo code is described in Algorithm 1.

Algorithm 1 Computing Resource Allocation

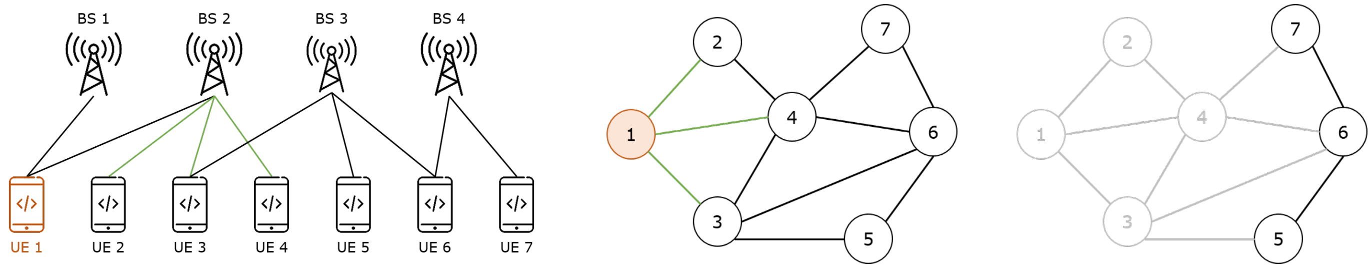
Input: Workloads of all UEs: $\{W_u\}$.
Output: Computing resource allocation: $\boldsymbol{f}$.
for $u \in \mathcal{U}$ **do**
  Set $f_u \leftarrow \frac{W_u F}{\sum_{u \in \mathcal{U}} W_u}$.
  Update t_u^{exe} by (4).

Algorithm 2 Subcarrier Allocation and Cell Clustering

Input: Set of all UEs $\mathcal{U}$, delay constraints of all UEs $\{T_u\}$, set of subcarrier $\mathcal{N}$, network topology.
Output: First-phase subcarrier assignment policy $\boldsymbol{\rho}$, clustering policy $\boldsymbol{x}$.
while $\mathcal{U} \neq \varnothing$ **do**
  Set $u^* \leftarrow \underset{u \in \mathcal{U}}{\text{argmin}} \{T_u^{\text{tran}}\}$;
  Set $n^* \leftarrow \underset{n \in \mathcal{N} \setminus \mathcal{N}'}{\text{argmax}} \left\{\bar{h}_{u^* n}\right\}$;
  Delete u^* and its neighboring nodes from G_{n^*};
  Set $\mathcal{N}_{u^*} \leftarrow \mathcal{N}_{u^*} \cup \{n^*\}$;
  Set $\mathcal{U} \leftarrow \mathcal{U} \setminus \{u^*\}$;
for $u \in \mathcal{U}$ **do**
  while $\mathcal{M}_u \neq \varnothing$ or $|\mathcal{X}_u| \leq X$ **do**
    Set $m^* \leftarrow \underset{m \in \mathcal{M}_u}{\text{argmax}} \left\{\sum_{n \in \mathcal{N}_u} h_{umn} | m \in \mathcal{M}_u\right\}$;
    Set $\mathcal{X}_u \leftarrow \mathcal{X}_u \cup \{m^*\}$;
  Update R_u by (2);
  Update t_u^{tran} by (3).

2) Subcarrier allocation and cell clustering: After the computing resource allocation, we proceed to allocate subcarriers and do cell clustering. Since every UE has a task to be processed, we need to allocate at least one subcarrier to each UE so as to ensure that the task can be successfully offloaded. For this, we first initialize the set of all UEs' transmission delay constraint $T_u^{\text{tran}} = T_u - t_u^{\text{exe}}$. To minimize the total delays of all UEs, we allocate the subcarrier n^* with the best channel gain $\bar{h}_{u^* n} = \frac{1}{|\mathcal{M}|} \sum_{m \in \mathcal{M}} h_{u^* mn}$ to the UE u^* based on the "earliest-deadline-first" principle (i.e., whose transmission delay constraint is tightest) since in this phase each UE will be associated with one subcarrier. Next, we construct an interference graph for each subcarrier. Whenever a subcarrier n^* is allocated to UE u^*, we delete node v and its neighbor nodes from the graph G_{n^*} to ensure that the neighbors of u^* will never use the subcarrier n^* that can cause severe interference to u^* and its neighbors. Therefore, other UEs cannot use the subcarrier that will cause severe interference. $\mathcal{N}'$ is a set of subcarriers that u^* cannot use since it will cause interference to others.

After the subcarrier allocation, we then form clusters for UEs to utilize uplink CoMP reception. Since we manage inter-user interference through interference graphs, each UE can choose its cluster by simply finding the best BS for UE's subcarrier set $\mathcal{N}_u$. Since we limit the size of each UE cluster set due to the constraint of cluster limit, namely $|\mathcal{X}_u| \leq X$, where $\mathcal{X}_u = \{m | x_{mu} = 1, m \in \mathcal{M}_u\}$. Then we update each



(a) Network topology and uplink inter-user interference

(b) Interference graph of UEs

(c) Interference-free UEs (w.r.t. UE 1)

Fig. 1. An example of interference graph construction.

Algorithm 3 Subcarrier Reuse and Cell Re-clustering

Input: Current delay of all UE t_u^{tran}, delay constraint of all UE T_u, set of subcarrier $\mathcal{N}$, network topology, first-phase subcarrier assignment policy $\boldsymbol{\rho}$, clustering policy $\boldsymbol{x}$.

Output: Final subcarrier assignment policy $\boldsymbol{\rho}$, clustering policy $\boldsymbol{x}$.

while $\mathcal{U} - \mathcal{U}' \neq \varnothing$ **do**
  Set $u^* \leftarrow \underset{u \in \mathcal{U}-\mathcal{U}'}{\text{argmax}} \{t_u^{\text{tran}} - T_u^{\text{tran}}\}$;
  Set $\mathcal{N}^\dagger$ to the set of available subcarriers for u^*;
  if UE u^* has available subcarrier to choose **then**
    Set $n^* \leftarrow \underset{n \in \mathcal{N}^\dagger}{\text{argmax}} \left\{\bar{h}_{u^*n}\right\}$;
    Delete u^* and his neighbor node from G_{n^*};
    Set $\mathcal{N}_{u^*} \leftarrow \mathcal{N}_{u^*} \cup \{n^*\}$;
    Do cell re-clustering and update R_{u*} and t_{u*}^{tran};
  if UE u^* has no available subcarrier or subcarrier limit is reached **then**
    Remove u^* from all graph G_n;
    Set $\mathcal{U}' \leftarrow \mathcal{U}' \cup \{u^*\}$.

UE transmission rate by (2) and calculate its delay by (3). The pseudo code is described in Algorithm 2.

3) Subcarrier reuse and cell re-clustering: Now, we need to allocate and reuse the rest of available subcarriers. We need to find the worst-case UE and the best subcarrier. Therefore, in each iteration, we find the most delay-sensitive UE and then allocate the subcarrier with best channel gain for the UE. Note that we need to look up the interference graph before choosing the best subcarrier to avoid the interference between UEs. After allocating the new subcarrier to the UE, we remove the UE from the interference graph as before and perform cell clustering again and update R_u and delay t_u^{tran}. We also consider the maximum transmission power of each UE so that each UE will limit the number of subcarriers to occupy. Therefore, we will check whether each UE has reached its subcarrier limit to prevent from allocating excessive subcarriers to the UE. The algorithm terminates when no more subcarrier is available for UEs or all UEs have reached their subcarrier limits. The pseudo code is described in Algorithm 3.

C. Complexity Analysis

In computing resource allocation, we allocate computing resource to UEs in each iteration, and spend $\mathcal{O}(|\mathcal{U}|)$ allocating computing resource to all UEs. In subcarrier allocation and cell clustering, to allocate the best subcarrier of the total subcarriers to UEs, one UE is chosen and associated with the best subcarrier in each iteration so as to obtain the highest channel gain. Therefore, it spends at most $\mathcal{O}(|\mathcal{U}||\mathcal{N}|)$. In cell clustering, each UE finds the best BSs from all BSs. Thus, it takes $\mathcal{O}(|\mathcal{U}||\mathcal{M}|)$ to find the cluster set for all UEs. In subcarrier reuse and cell re-clustering, the implementation is based on the fibonacci heap. After our derivation, the algorithm spends $\mathcal{O}(|\mathcal{U}||\mathcal{N}|)$ in the case that the number of UEs is more than that of subcarriers. The overall complexity of the IGRA algorithm is $\mathcal{O}(|\mathcal{U}||\mathcal{N}|)+\mathcal{O}(|\mathcal{U}||\mathcal{M}|)+\mathcal{O}(|\mathcal{U}||\mathcal{N}|)+\mathcal{O}(|\mathcal{U}|) = \mathcal{O}(|\mathcal{U}|(|\mathcal{N}| + |\mathcal{M}|))$.

V. Performance Evaluation

A. Benefit of CoMP Scenario

In this section, the performance of our algorithm (denoted by IGRA) is evaluated and compared against other schemes. We first compare the performance between our algorithm and the non-CoMP reception scheme (denoted by nCR). The simulation area is within a 500m×500m area. UEs and BSs are distributed according to a Point Poisson Process. The system bandwidth is set to 20 MHz and the carrier frequency is 800 MHz. We consider a Rayleigh fading channel model to characterize the heterogeneity of channel gain among subcarriers and the channel gain follows the free-space path loss model. The path loss exponent is set to 2.8. The BS coverage radius is set to 100m. The MEC server CPU speed is 30 GHz. The max transmission power of each UE is 0.4W. Other simulation settings are listed in Table II, as in [9], [18], [19].

Fig. 2a investigates the benefit of CoMP reception. As can be seen, nCR suffers from the severe interference problem because it lacks inter-cell coordination. On the other hand, our algorithm takes care of inter-user interference, therefore, the interference can be mitigated and transformed to effective signals. The gap between IGRA and nCR in terms of total delays is significant as shown in Fig. 2a.

TABLE II
SIMULATION SETTINGS I

Parameter	Value
BS density	600 BSs per km^2
UE density	900 UEs per km^2
Area	500×500 m^2
System bandwidth	20 MHz
Carrier frequency	800 MHz
Number of subcarriers	128
UE data size	1 kbit to 70kbits (uniform)
Task workload	10^5 to 10^6 cycles (uniform)

TABLE III
SIMULATION SETTINGS II

Parameter	Value
BS density	100-700 BSs per km^2
UE density	300 UEs per km^2
Area	500×500 m^2
UE data size	1 kbit to 10kbits (uniform)
Task workload	10^5 to 10^6 cycles (uniform)

B. Comparison with Other Schemes

To evaluate our heuristic algorithm, we compare with three resource allocation schemes.

- Maximum channel gain (denoted by maxCG): This algorithm allocates subcarriers to the UE who has the maximum channel gain in order to maximize the overall channel quality in the network [20]. We add our graph-based interference management into this benchmark to purely discuss the difference of the subcarrier assignment policy.
- Random allocation (denoted by Rand): This algorithm just randomly assigns subcarrier to UEs. Graph-based interference management is considered.
- Interference graph (denoted by IG): In this algorithm, we want to observe how well our graph-based interference management works, so we do not take into account inter-user interference management in this case. The simulation setup is summarized in Table III.

We now evaluate our algorithm when the BS density becomes larger. That implies that when the cells become denser, our algorithm always outperforms the others since we allocate subcarriers properly and take into account inter-user interference. The result is shown in Fig. 2b. We can also observe that the delay decreases when the cell density increases and our proposed algorithm has the best performance.

We now evaluate our algorithm when the UE density increases. This implies that when the number of UEs become larger and the resource will become insufficient. The BS density is $500/\text{km}^2$ and the UE density is 300 to $1100/\text{km}^2$ and the data size varies from 1 kbit to 30 kbits. Other settings remain the same as before. From Fig. 2c, we observe that the performance for the scenario IG becomes much worse when the UE density becomes extremely large. This is because this algorithm does not consider the interference among UEs. When the UE density is large, the effect of interference becomes a problem. UEs interfere with each other very heavily, resulting in very poor delay performance. However, in our heuristic algorithm, we allocate the resource of the MEC system properly and jointly consider the delay constraints of UEs. Therefore, our algorithm has the best delay performance. We also compare our algorithm with the optimality (denoted by OPT) in Fig. 2d. We conduct this simulation under a smaller network (250m×250m).

We now evaluate our algorithm when the UE's maximum data size and the variance of data size among UEs are large. The setting differences include: BS density is $600/\text{km}^2$ and UE density is $300/\text{km}^2$ and data size is from 1 kbits to 10 kbits and 1 kbits to 40 kbits. Other settings remain the same. From Fig. 2e, we observe that the performance for the scenario maxCG becomes much worse when the variance of data size grows. In maxCG, which only considers allocating the best subcarrier first and ignores the data size of each UE, its delay increases much more as the heterogeneity increases among UEs.

Now we take local computing mode into consideration. Assume that the computing capability is fixed at each UE and it may vary between UEs. Each UE can choose to execute the task locally if it can satisfy delay constraint. The simulation area is 250m×250m, and the BS and UE densities are 600 km^2 and 300 km^2, respectively. Data size varies from 1 kbit to 10 kbits. UE computation capacity is 0.1 GHz to 0.15 GHz. Other setting remains the same. From Fig. 2f, we can observe that the delay is smaller when UE can choose whether to offload or not. The reason is that if there are too many UEs in the MEC system, it will cause a heavy burden to the system and then result in the increase of delay. However, if some UE can execute his task locally and satisfy his delay constraint. There will be sufficient communication and computing resources for those really need. In the end, delay performance will be better when the resource of MEC system is tight.

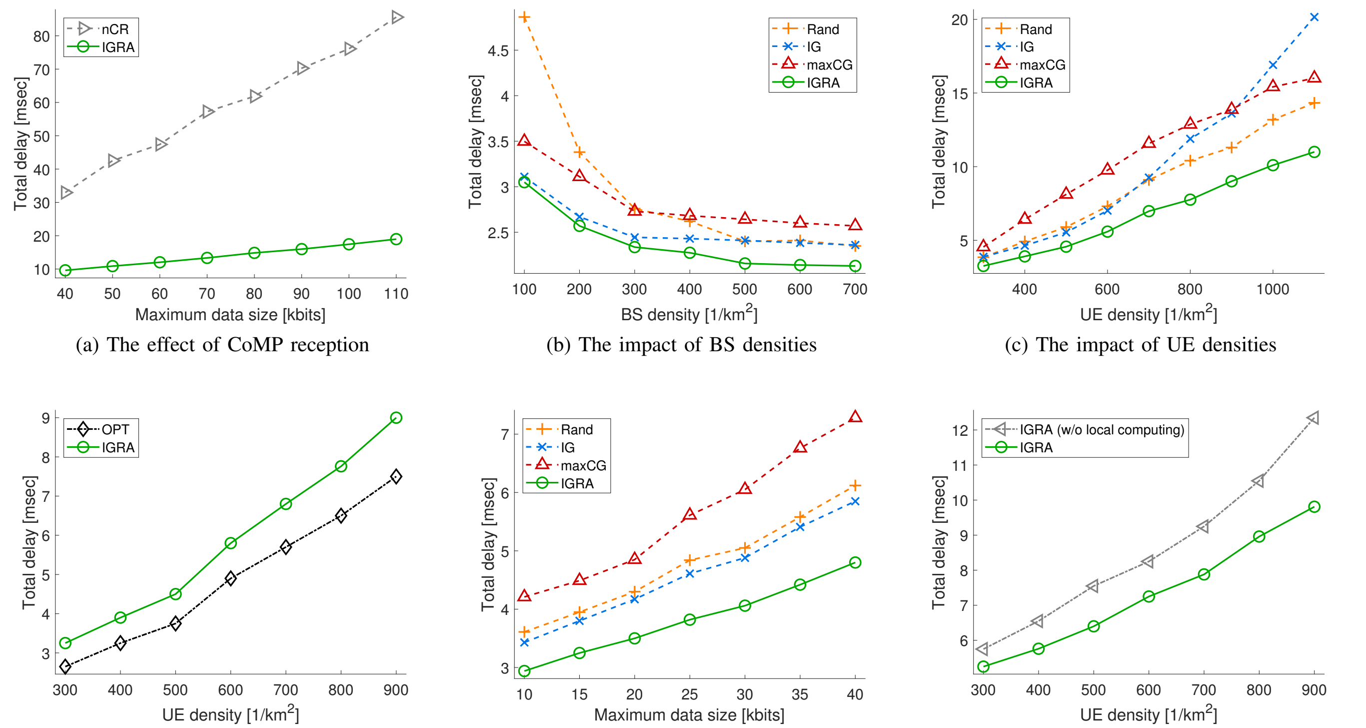
VI. CONCLUSION

In this paper, we investigate a resource allocation problem in MEC with uplink CoMP reception, thereby elevating uplink signal qualities. To solve the RAMCR problem, we propose the 3-phase IGRA algorithm which leverages interference graphs to allocate communication and computing resources and do cell clustering. The simulation results show that by means of CoMP reception, the delay performance can be significantly improved. Besides, our proposed graph-based interference management can cope with inter-user interference effectively. We also show that our performance outperforms other comparison schemes which do not take into account inter-user interference.

REFERENCES

[1] P. Mach and Z. Becvar, "Mobile edge computing: A survey on architecture and computation offloading," *IEEE Commun. Surveys Tuts.*, vol. 19, no. 3, pp. 1628–1656, 3rd Quart. 2017.

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



(a) The effect of CoMP reception (b) The impact of BS densities (c) The impact of UE densities

(d) The impact of UE densities (e) The impact of maximum data sizes (f) The impact of local computing

Fig. 2. The achieved total delay under various system settings.

[3] T. Taleb, K. Samdanis, B. Mada, H. Flinck, S. Dutta, and D. Sabella, "On multi-access edge computing: A survey of the emerging 5G network edge cloud architecture and orchestration," *IEEE Commun. Surveys Tuts.*, vol. 19, no. 3, pp. 1657–1681, 3rd Quart. 2017.

[4] P. Porambage, J. Okwuibe, M. Liyanage, M. Ylianttila, and T. Taleb, "Survey on multi-access edge computing for Internet of Things realization," *IEEE Commun. Surveys Tuts.*, vol. 20, no. 4, pp. 2961–2991, 4th Quart. 2018.

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

[6] Y. Mao, J. Zhang, S. H. Song, and K. B. Letaief, "Power-delay tradeoff in multi-user mobile-edge computing systems," in *Proc. IEEE GLOBECOM*, Washington D.C., USA, Dec. 2016.

[7] 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 PIMRC*, Montreal, QC, Canada, Oct. 2017.

[8] J. Guo, Z. Song, Y. Cui, Z. Liu, and Y. Ji, "Energy-efficient resource allocation for multi-user mobile edge computing," in *Proc. IEEE GLOBECOM*, Singapore, Dec. 2017.

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

[10] F. Guo, L. Ma, H. Zhang, H. Ji, and X. Li, "Joint load management and resource allocation in the energy harvesting powered small cell networks with mobile edge computing," in *Proc. IEEE INFOCOM WKSHPS*, Honolulu, HI, USA, Apr. 2018.

[11] C. Wang, F. R. Yu, C. Liang, Q. Chen, and L. Tang, "Joint computation offloading and interference management in wireless cellular networks with mobile edge computing," *IEEE Trans. Veh. Technol.*, vol. 66, no. 8, pp. 7432–7445, Aug. 2017.

[12] M. Chen and Y. Hao, "Task offloading for mobile edge computing in software defined ultra-dense network," *IEEE J. Sel. Areas Commun.*, vol. 36, no. 3, pp. 587–597, Mar. 2018.

[13] W. M. A. Osseiran, J.F. Monserrat, *Mobile and Wireless Communications for IMT-advanced and Beyond*. John Wiley & Sons, 2011.

[14] "Coordinated multi-point operation for LTE," 3GPP, TR 36.819, Sep. 2013. [Online]. Available: http://www.3gpp.org/ftp/Specs/html-info/36819.htm

[15] F. Guidolin, L. Badia, and M. Zorzi, "A distributed clustering algorithm for coordinated multipoint in LTE networks," *IEEE Wireless Commun. Lett.*, vol. 3, no. 5, pp. 517–520, Oct. 2014.

[16] 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.

[17] A. S. Fukunaga and R. E. Korf, "Bin completion algorithms for multicontainer packing, knapsack, and covering problems." *J. Artif. Intell. Res.*, vol. 28, pp. 393–429, Jan. 2007.

[18] H. Guo, J. Zhang, J. Liu, H. Zhang, and W. Sun, "Energy-efficient task offloading and transmit power allocation for ultra-dense edge computing," in *Proc. IEEE GLOBECOM*, Abu Dhabi, UAE, Dec. 2018.

[19] Y. Mao, J. Zhang, and K. B. Letaief, "Joint task offloading scheduling and transmit power allocation for mobile-edge computing systems," in *Proc. IEEE WCNC*, San Francisco, CA, USA, Mar. 2017.

[20] X. Zhang, Y. Li, and H. Ji, "An energy efficiency power and sub-carrier allocation for the downlink multi-user CoMP in multi-cell systems," in *Proc. IEEE ICC*, Ottawa, Canada, Jun. 2012.