DRL-Assisted Resource Allocation for NOMA-MEC Offloading with Hybrid SIC

Haodong Li, Fang Fang, Zhiguo Ding

Abstract

Multi-access edge computing (MEC) and non-orthogonal multiple access (NOMA) have been regarded as promising technologies to improve computation capability and offloading efficiency of the mobile devices in the sixth generation (6G) mobile system. This paper mainly focuses on the hybrid NOMA-MEC system, where multiple users are first grouped into pairs, and users in each pair offload their tasks simultaneously by NOMA, and then a dedicated time duration is scheduled to the more delay-tolerable user for uploading the remaining data by orthogonal multiple access (OMA). For the conventional NOMA uplink transmission, successive interference cancellation (SIC) is applied to decode the superposed signals successively according to the channel state information (CSI) or the quality of service (QoS) requirement. In this work, we integrate the hybrid SIC scheme which dynamically adapts the SIC decoding order among all NOMA groups. To solve the user grouping problem, a deep reinforcement learning (DRL) based algorithm is proposed to obtain a close-to-optimal user grouping policy. Moreover, we optimally minimize the offloading energy consumption by obtaining the closed-form solution to the resource allocation problem. Simulation results show that the proposed algorithm converges fast, and the NOMA-MEC scheme outperforms the existing orthogonal multiple access (OMA) scheme.

Index Terms

depth reinforcement learning (DRL); multi-access edge computing (MEC); resource allocation; sixth-generation (6G); user grouping

I. INTRODUCTION

With fifth-generation (5G) networks being available now, the sixth-generation (6G) wireless network is currently under research, which is expected to provide superior performance to satisfy growing demands of mobile equipment, such as latency sensitive, energy hungry and computationally intensive services and applications [1], [2]. For example, the Internet of Things (IoT) networks are being developed rapidly, where massive numbers of nodes are supposed to be connected together, and IoT nodes can not only communicate with each others but also process acquired data [3]–[5]. However, such IoT and many other terminal devices are constrained by the battery life and computational capability, and thereby these devices cannot support computationally intensive tasks. A conventional approach to improve the computation capability of mobile devices is mobile cloud Computing (MCC), where

H. Li and Z. Ding are with the Department of Electrical and Electronic Engineering, The University of Manchester, M13 9PL, UK. F. Fang is with the Department of Engineering, Durham University, Durham DH1 3LE, UK (e-mail: haodong.li@manchester.ac.uk, fang.fang@durham.ac.uk, zhiguo.ding@manchester.ac.uk).
computation intensive tasks are offloaded to the central cloud servers for data processing [6], [7]. However, MCC will cause significant delays due to the long propagation distances. To address the offloading delay issue, especially for delay sensitive applications in the future 6G networks, multi-access edge computing (MEC) has been emerged as a decentralized structure to provide the computation capability close to the terminal devices, which are generally implemented at the base stations to provide cloud-like task processing service. [7]–[10].

From the communication perspective, non-orthogonal multiple access (NOMA) has been recognized as a promising technology to improve the spectral efficiency and massive connections, which enables multiple users to utilize the same resource block such as time and frequency for transmissions [11], [12]. Take power domain NOMA as an example, the signals of multiple users are multiplexed in power domain by the superposition coding, and at the receiver side, successive interference cancellation (SIC) is adopted to remove the multiple access interference successively [13]. Hence, integrating NOMA with MEC can potentially improve the service quality of MEC including low transmission latency and massive connections compared to the conventional orthogonal multiple access (OMA).

A. Related Works

The integration of NOMA and MEC has been well studied so far, and researchers have proposed various approaches on optimal resource allocation to minimize the offloading delay and energy consumption. In [14], the author minimized the offloading latency for a multi-user scenario, in which the power allocation and task partition ratio were jointly optimized. The partial offloading policy can determine the amount of data to be offloaded to the server, and the remainder is processed locally. The author of [15] proposed a iterative two-user NOMA scheme to minimize the offloading latency, in which two users offload their tasks simultaneously by NOMA. Since one of the users suffers performance degradation introduced by NOMA, instead of forcing two users to complete offloading at the same time, the remaining data is offloaded in together with the next user during the following time slot. Moreover, many existing works investigate the energy minimization of NOMA-MEC networks. For example, the joint optimization of central processing unit (CPU) frequency, task partition ratio and power allocation for a NOMA-MEC heterogeneous network were considered in [16], [17]. In [18], the author considered a multi-antenna NOMA-MEC network, and presented an approach to minimize the weighted sum energy consumption by jointly optimizing the computation and communication resource.

In addition to the existing works on pure NOMA schemes as aforementioned, a few works also combine NOMA and OMA in together, which is denominated as hybrid NOMA [19]. In this paper, the author proposed a two-user hybrid NOMA scenario, in which one user is less delay tolerable than the other. The two users offload during the first time slot by NOMA, and the user with longer deadline offloads the remaining data during an additional time duration by OMA. This configuration presents significant benefits, which outperforms both OMA and pure NOMA in terms of energy consumption since the energy can be saved for the delay tolerable user instead of finishing offloading at the same time in pure NOMA networks. In [20], [21], the hybrid NOMA scheme is extended to multi-user scenarios, in which a two-to-one matching algorithm is utilized to pair every two users into a group, and each group offload through a sub-carrier.
For the resource allocation in NOMA-MEC networks, user grouping is a non-convex problem, which is solved by exhaustive search or applying matching theory. Deep reinforcement learning (DRL) is recognized as a novel approach to this problem, which is a powerful tool to solve the real-time decision-making tasks, and only handful papers utilized it for user grouping and sub-channel assignment such as [22], [23] which output the user grouping policy for uplink and downlink NOMA networks respectively.

Moreover, in most of the NOMA works, the SIC decoding order is prefixed, which can either be determined by the channel state information (CSI) or the quality of service (QoS) requirements of users [24]–[26]. A recent work [27] has proposed a hybrid SIC scheme to switch the SIC decoding order dynamically, which has shown significant performance improvement in uplink NOMA networks. The author of [28] integrated the hybrid SIC scheme with an MEC network to serve two uplink users, and the results reveals that the hybrid SIC outperforms the QoS based decoding order.

B. Motivation and Contributions

Motivated by the existing research on MEC-NOMA, in this paper, we investigate the energy minimization for the uplink transmission in multi-user hybrid NOMA-MEC networks with hybrid SIC. More specifically, a DRL based framework is proposed to generate a user grouping policy, and the power allocation, time allocation and task partition assignment are jointly optimized for each group. The DRL framework collects experience data including CSI, deadlines, energy consumption as labeled data to train the neural networks (NNs). The main contributions of this paper are summarized as follows:

- A hybrid NOMA-MEC network is proposed, in which an MEC server is deployed at the base station to serve multiple users. All users are divided into pairs, and each pair is assigned into one sub-channel. The users in each group adopt NOMA transmission with the hybrid SIC scheme in the first time duration, and the user with longer deadline transmits the remaining data by OMA in the following time duration. We propose a DRL-assisted user grouping framework with joint power allocation, time scheduling, and task partition assignment to minimize the offloading energy consumption under transmission latency and offloading data amount constraints.

- By assuming that the user grouping policy is given, the energy minimization problem for each group is non-convex due to the multiplications of variables and a 0-1 indicator function, which indicates two cases of decoding orders. The solution to the original problem can be obtained by solving each case separately. A multilevel programming method is proposed, where the energy minimization problem is decomposed into three sub-problems including power allocation, time scheduling, and task partition assignment. By carefully analyzing the convexity and monotonicity of each sub-problem, the solutions to all three sub-problems are obtained optimally in closed-form. The solution to the energy minimization problem for each case can be determined optimally by adopting the decisions successively from the lower level to the higher level (i.e., from the optimal task partition assignment to the optimal power allocation). Therefore, the solution to the original problem can be obtained by comparing the numerical results of those two cases and selecting the optimal solution with lower energy consumption.
A DRL framework for user grouping is designed based on a deep Q-learning algorithm. We provide a training algorithm for the NN to learn the experiences based on the channel condition and delay tolerance of each user during a period of slotted time, and the user grouping policy can be learned gradually at the base station by maximizing the negative of the total offloading energy consumption.

Simulation results are provided to illustrate the convergence speed and the performance of this user grouping policy by comparing with random user grouping policy. Moreover, compared with the OMA-MEC scheme, our proposed NOME-MEC scheme can achieve superior performance with much lower energy consumption.

C. Organizations

The rest of the paper is structured as follows. The system model and the formulated energy minimization problem for our proposed NOMA-MEC scheme are described in Section II. Section III presents the optimal solution to the energy minimization problem. Following that, the DRL based user grouping algorithm is introduced in Section IV. Finally, the simulation results of the convergence and average performance for the proposed scheme are shown in Section V, and Section VI concludes this paper.

II. System Model and Problem Formulation

A. System Model

In this paper, we consider a NOMA-MEC network, where a base station is equipped with an MEC server to serve $K$ resource-constrained users. During one offloading cycle, each user offloads its task to the MEC server and then obtains the results which processed at the MEC server. Generally, the data size of the computation results is relatively smaller than the offloaded data in practical, thus, the time for downloading the results can be omitted [18]. Moreover, since the MEC server has much higher computation capability than mobile devices, the data processing time at the MEC server can be ignored compared to the offloading time [14]. Therefore, in this work, the total offloading delay is approximated to the time consumption of data uploading to base station.

We assume that all $K$ users are divided into $\Phi$ groups to transmit signals at different sub-channels, and each group $\phi$ contains two users such that $K = 2\Phi$. In each group, we denote the user with short deadline by $U_{m,\phi}$, and the user with relevantly longer deadline by $U_{n,\phi}$, which indicates $\tau_{m,\phi} \leq \tau_{n,\phi}$, where $\tau_{i,\phi}$ is the latency requirement of $U_{i,\phi}, \forall i \in \{m, n\}$ in group $\phi$. Because $U_{m,\phi}$ has a tighter deadline, it is assumed that the whole duration $\tau_{m,\phi}$ will be used up, which means that the offloading time $t_{m,\phi} = \tau_{m,\phi}$.

In this system model, we adopt the block channel model which indicates that the channel condition remains static during each time slot. With the small scale fading, the channel gain of a user in group $\phi$ can be expressed as

$$H_{i,\phi} = \tilde{h}_{i,\phi}d_{i,\phi}^{-\alpha}, \quad \forall i \in \{m, n\}, \forall \phi,$$

where $\tilde{h}_{i,\phi} \sim CN(0, 1)$ is the Rayleigh fading coefficient, $d_{i,\phi}$ is the distance between $U_{i,\phi}$ to the base station, and $\alpha$ is the pass loss exponent. The channel gain is normalized by the additive white Gaussian noise (AWGN) power with zero-mean and $\sigma^2$ variance, which can be written as

$$H_{i,\phi} = \tilde{h}_{i,\phi}d_{i,\phi}^{-\alpha}, \quad \forall i \in \{m, n\}, \forall \phi,$$
\[ h_{i,\phi} = \frac{|H_{i,\phi}|^2}{\sigma^2}, \quad \forall i \in \{m, n\}, \forall \phi. \] (2)

As shown in Fig. 1, since those two users have different delay tolerance, it is natural to consider that the \( U_{n,\phi} \) is unnecessary to finish offloading within \( \tau_{m,\phi} \) via NOMA transmission, and potentially to save energy if \( U_{n,\phi} \) can utilize the spare time \( \tau_{n,\phi} - \tau_{m,\phi} \). Hence, our proposed hybrid NOMA scheme enables \( U_{n,\phi} \) to offload part of its data when \( U_{m,\phi} \) offloading its task during \( \tau_{m,\phi} \), an additional time duration \( t_{r,\phi} \) is scheduled within each time slot to transmit \( U_{n,\phi} \)'s remaining data. The task transmission for \( U_{m,\phi} \) should be completed within \( \tau_{n,\phi} \), i.e.,

\[ t_{r,\phi} \leq \tau_{n,\phi} - \tau_{m,\phi}, \forall \phi. \] (3)

As aforementioned, the users in each group will occupy the same sub-channel to upload their data to the base station simultaneously via NOMA. In NOMA uplink transmission, SIC is adopted at the base station to decode the superposed signal. Conventionally, the SIC decoding order is based on either user’s CSI or the QoS requirement [27]. For the QoS based case, to guarantee \( U_{m,\phi} \) can offload its data by \( \tau_{m,\phi} \), \( U_{n,\phi} \) is set to be decoded first, and the data rate is

\[ R_{n,\phi} = B \ln \left( 1 + \frac{P_{n,\phi}|h_{n,\phi}|^2}{P_{m,\phi}|h_{m,\phi}|^2 + 1} \right), \] (4)

where \( B \) is the bandwidth of each sub-channel. \( P_{n,\phi} \) and \( P_{m,\phi} \) are the transmission power of \( U_{n,\phi} \) and \( U_{m,\phi} \) during NOMA transmission respectively. Based on the NOMA principle, the signal of \( U_{m,\phi} \) can then be decoded if (4) is satisfied, and the data rate for \( U_{m,\phi} \) can be written as

\[ R_{m,\phi} = B \ln \left( 1 + P_{m,\phi}|h_{m,\phi}|^2 \right). \] (5)

If \( U_{n,\phi} \) is decoded first according to the CSI principle, the achievable rate is same as (4) since \( U_{n,\phi} \) treat the signal of \( U_{m,\phi} \) as noise power. In contrast, \( U_{m,\phi} \) can be decoded first if the following condition holds:

\[ R_{m,\phi} \leq B \ln \left( 1 + \frac{P_{m,\phi}|h_{m,\phi}|^2}{P_{n,\phi}|h_{n,\phi}|^2 + 1} \right). \] (6)

Then the data rate of \( U_{n,\phi} \) can be obtained by removing the information of \( U_{m,\phi} \), which is

\[ R_{n,\phi} = B \ln \left( 1 + P_{n,\phi}|h_{n,\phi}|^2 \right). \] (7)

If the same power is allocated to \( U_{n,\phi} \) for both QoS and CSI scheme, it is evident that the achievable rate in (7) is higher than that in (4), and the decoding order in (7) is preferred in this case. However, since the constraint (6) cannot be always satisfied, the system has to dynamically change the decoding order accordingly to achieve better performance, which motivated us to utilize the hybrid SIC scheme.

In addition, during \( t_{r,\phi} \), \( U_{n,\phi} \) adopts OMA transmission, and the data rate can be expressed as

\[ R_{r,\phi} = B \ln \left( 1 + P_{r,\phi}|h_{n,\phi}|^2 \right). \] (8)
where $P_{r,\phi}$ represents the transmission power of $U_{n,\phi}$ during the second time duration $t_{n,\phi}$.

In this work, the data length of each task is denoted by $L$, which is assumed to be bitwise independent, and we propose a partial offloading scheme in which each task can be processed locally and remotely in parallel. An offloading partition assignment coefficient $\beta_{\phi} \in [0, 1]$ is introduced, which indicates how much amount of data is offloaded to the MEC server, and the rest can be executed by the local device in parallel. Thus, for each task, the amount of data for offloading to the server is $\beta_{\phi}L$ and $(1 - \beta_{\phi})L$ is the data processed locally.

$U_{n,\phi}$ can take the advantage of local computing by executing $(1 - \beta_{\phi})L$ data locally during the scheduled NOMA and OMA time duration $t_{m,\phi} + t_{r,\phi}$. Therefore, the energy consumption for $U_{n,\phi}$’s local execution, which is denoted by $E_{n,\phi}^{loc}$, can be expressed as

$$E_{n,\phi}^{loc} = \kappa_0 \frac{C(1 - \beta_{\phi})L^3}{(t_{m,\phi} + t_{r,\phi})^2},$$

(9)

where $\kappa_0$ denotes the coefficient related to the mobile device’s processor and $C$ is the number of CPU cycles required for computing each bit.

The total energy consumed by $U_{n,\phi}$ per task involves three parts, including the energy consumed by local computing, and transmission during NOMA and OMA offloading. The power for offloading is scheduled separately during these scheduled two time duration according to the hybrid SIC scheme, and thereby the offloading energy consumption $E_{n,\phi}^{off}$ can be expressed as

$$E_{n,\phi}^{off} = t_{m,\phi}P_{n,\phi} + t_{r,\phi}P_{r,\phi}.$$  

(10)

Hence, the total energy consumption can be expressed as

$$E_{\phi}^{tot} = E_{n,\phi}^{loc} + E_{n,\phi}^{off}.$$  

(11)

B. Problem Formulation

We assume that the resource allocation of $U_{m,\phi}$ is given as a constant in each group since $U_{m,\phi}$ is treated as the primary user whose requirement need to be guaranteed in priority, and we only focus on the energy minimization for $U_{n,\phi}$ during both NOMA and OMA duration. Given the user grouping policy which will be solved in Section IV, the energy minimization problem for each pair can be formulated as
\[
(P_1) : \min_{\tau_{r,\phi}, \beta_{\phi}} \quad \kappa_0 \frac{[C(1 - \beta_{\phi})L]^3}{(\tau_{m,\phi} + \tau_{r,\phi})^2} + \tau_{m,\phi} P_{n,\phi} + \tau_{r,\phi} P_{r,\phi}
\]
\[
\text{s.t.} \quad \tau_{m,\phi} P_{n,\phi}^H + \tau_{r,\phi} B \ln \left( 1 + P_{r,\phi} |h_{n,\phi}|^2 \right) \geq \beta_{\phi} L \quad (12a)
\]
\[
\tau_{m,\phi} B \ln \left( 1 + \frac{P_{m,\phi} |h_{m,\phi}|^2}{P_{n,\phi} |h_{n,\phi}|^2} \right) \geq 1 \quad (12b)
\]
\[
P_{n,\phi} \geq 0, P_{r,\phi} \geq 0 \quad (12c)
\]
\[
0 \leq \tau_{r,\phi} \leq \tau_{n,\phi} - \tau_{m,\phi} \quad (12d)
\]
\[
0 \leq \beta_{\phi} \leq 1, \quad (12e)
\]

where \( R_{n,\phi}^H = 1_{n,\phi} B \ln \left( 1 + P_{n,\phi} |h_{n,\phi}|^2 \right) + (1 - 1_{n,\phi}) B \ln \left( 1 + \frac{P_{m,\phi} |h_{m,\phi}|^2}{P_{n,\phi} |h_{n,\phi}|^2} + 1 \right) \). \( 1_{n,\phi} \) is the indicator function. When \( 1_{n,\phi} = 1 \), \( U_{m,\phi} \) is decoded first and vice versa. Constraint (12b) and (12c) ensure all the users should complete offloading the designated amount of data within the given deadline. The constraint (12d) limits the additionally scheduled time slot should not beyond \( U_{n,\phi} \)’s delay tolerance. Constraints (12d) (12f) set the feasible range of the transmission power and offloading coefficient.

The problem \((P_1)\) is non-convex due to the multiplication of several variables. Therefore, in the following section, we propose a multilevel programming algorithm to address the energy minimization problem optimally by obtaining the closed-form solution.

### III. ENERGY MINIMIZATION FOR NOMA-MEC WITH HYBRID SIC SCHEME

In this section, a multilevel programming method is introduced to decompose the problem \((P_1)\) into three sub-problems, i.e., power allocation, time slot scheduling and task assignment, which can be solved optimally by obtaining the closed-form solution. The optimal solution to the original problem \((P_1)\) can thereby be found by solving those three sub-problems successively, which are provided in the below subsections.

#### A. Power Allocation

Let \( \tau_{r,\phi} \) and \( \beta_{\phi} \) be fixed, the problem \((P_1)\) is regarded as a power allocation problem which can be rewritten as

\[
(P_2) : \min_{P_{n,\phi}, P_{r,\phi}} \quad \kappa_0 \frac{[C(1 - \beta_{\phi})L]^3}{(\tau_{m,\phi} + \tau_{r,\phi})^2} + \tau_{m,\phi} P_{n,\phi} + \tau_{r,\phi} P_{r,\phi}
\]
\[
\text{s.t.} \quad \tau_{m,\phi} P_{n,\phi}^H + \tau_{r,\phi} B \ln \left( 1 + P_{r,\phi} |h_{n,\phi}|^2 \right) \geq \beta_{\phi} L \quad (13a)
\]
\[
\tau_{m,\phi} B \ln \left( 1 + \frac{P_{m,\phi} |h_{m,\phi}|^2}{P_{n,\phi} |h_{n,\phi}|^2} \right) \geq 1 \quad (13b)
\]
\[
P_{n,\phi} \geq 0, P_{r,\phi} \geq 0 \quad (13c)
\]

Since there exists an indicator function, \((P_2)\) is solved in two different cases, i.e., when \( 1_{n,\phi} = 1 \) and when \( 1_{n,\phi} = 0 \). The following theorem provides the optimal solution of both cases.
Theorem 1. The optimal power allocation to \((P^2)\) is given by the following two cases according to the indicator function:

1) For \(1_{n,\phi} = 1\), \(U_{m,\phi}\) is decoded first, and the power allocation for this decoding order is presented as follows:

   a) When \(P_{n,\phi} \neq 0\) and \(P_{r,\phi} \neq 0\), \(U_{n,\phi}\) offloads in both time duration, which is termed as hybrid NOMA, and the power allocation is given in the following two cases:

      i) If \(P_{m,\phi} > |h_{m,\phi}|^{-2} e^{\frac{\beta_{m,\phi}}{r_{m,\phi} + r_{r,\phi}}} \left(e^{\frac{\beta_{r,\phi}}{r_{m,\phi} + r_{r,\phi}}} - 1\right)\),

         \[
P_{n,\phi}^* = P_{r,\phi}^* = |h_{n,\phi}|^{-2} \left(e^{\frac{\beta_{r,\phi}}{r_{m,\phi} + r_{r,\phi}}} - 1\right).
\]

      ii) If \(|h_{m,\phi}|^{-2} \left(e^{\frac{\beta_{r,\phi}}{r_{m,\phi} + r_{r,\phi}}} - 1\right) \leq P_{m,\phi} \leq |h_{m,\phi}|^{-2} e^{\frac{\beta_{m,\phi}}{r_{m,\phi} + r_{r,\phi}}} \left(e^{\frac{\beta_{r,\phi}}{r_{m,\phi} + r_{r,\phi}}} - 1\right)\),

         \[
P_{n,\phi}^* = |h_{n,\phi}|^{-2} \left\{e^{\frac{\beta_{r,\phi}}{r_{m,\phi} + r_{r,\phi}}} - 1\right\},
\]

         \[
P_{r,\phi}^* = |h_{n,\phi}|^{-2} \left(e^{\frac{\beta_{r,\phi}}{r_{m,\phi} + r_{r,\phi}}} ln\left[P_{m,\phi}|h_{m,\phi}|^2 \left(e^{\frac{\beta_{r,\phi}}{r_{m,\phi} + r_{r,\phi}}} - 1\right)^{-1}\right] - 1\right\}.
\]

   b) When \(U_{n,\phi}\) only offloads during the first time duration \(\tau_{m,\phi}\), this scheme is termed as pure NOMA, and the power allocation is obtained as

   if \(P_{m,\phi} \geq |h_{m,\phi}|^{-2} e^{\frac{\beta_{m,\phi}}{r_{m,\phi} + r_{r,\phi}}} \left(e^{\frac{\beta_{r,\phi}}{r_{m,\phi} + r_{r,\phi}}} - 1\right)\),

   \[
P_{n,\phi}^* = |h_{n,\phi}|^{-2} \left(e^{\frac{\beta_{r,\phi}}{r_{m,\phi} + r_{r,\phi}}} - 1\right),
\]

   \[
P_{r,\phi}^* = 0.
\]

   c) When \(P_{n,\phi}^* = 0\), \(U_{n,\phi}\) chooses to offload solely during the section time duration \(t_{r,\phi}\), and the optimal power allocation is:

   if \(P_{m,\phi} \geq |h_{m,\phi}|^{-2} \left(e^{\frac{\beta_{r,\phi}}{r_{r,\phi}}} - 1\right)\),

   \[
P_{n,\phi}^* = 0,
\]

   \[
P_{r,\phi}^* = |h_{n,\phi}|^{-2} \left(e^{\frac{\beta_{r,\phi}}{r_{r,\phi}}} - 1\right).
\]

2) For \(1_{n,\phi} = 0\):

   1) When \(P_{n,\phi} \neq 0\) and \(P_{r,\phi} \neq 0\), \(U_{n,\phi}\), the hybrid NOMA power allocation is given by

      if \(P_{m,\phi} \leq |h_{m,\phi}|^{-2} \left(e^{\frac{\beta_{r,\phi}}{r_{r,\phi}}} - 1\right)\),

      \[
P_{n,\phi}^* = |h_{n,\phi}|^{-2} \left(P_{m,\phi}|h_{m,\phi}|^2 + 1\right) e^{\frac{\beta_{r,\phi}}{r_{m,\phi} + r_{r,\phi}}} \ln\left|\frac{P_{m,\phi}|h_{m,\phi}|^{2+\tau}}{r_{m,\phi} + r_{r,\phi}}\right| - 1\right\},
\]

      \[
P_{r,\phi}^* = |h_{n,\phi}|^{-2} \left(P_{m,\phi}|h_{m,\phi}|^2 + 1\right) e^{\frac{\beta_{r,\phi}}{r_{m,\phi} + r_{r,\phi}}} \ln\left|\frac{P_{m,\phi}|h_{m,\phi}|^{2+\tau}}{r_{m,\phi} + r_{r,\phi}}\right| - 1\right\}.
\]
2) When $P_{r,\phi} = 0$, the pure NOMA case can be obtained as

$$P_{n,\phi}^* = |h_{n,\phi}|^{-2} \left( P_{m,\phi} |h_{m,\phi}|^2 + 1 \right) \left( e^{\frac{\beta_{\phi} L}{\tau_{m,\phi}}} - 1 \right). \quad (19)$$

3) When $P_{n,\phi} = 0$, the OMA case is:

$$P_{n,\phi} = 0, \quad (20a)$$

$$P_{r,\phi} = |h_{n,\phi}|^{-2} \left( e^{\frac{\beta_{\phi} L}{\tau_{r,\phi}}} - 1 \right). \quad (20b)$$

**Proof.** Refer to Appendix A. ■

**Remark 1.** Theorem 1 provides the optimal power allocation for both two decoding sequences, i.e., $U_{m,\phi}$ is decode first when $1_{n,\phi} = 1$, and $U_{n,\phi}$ is decode first when $1_{n,\phi} = 0$. The optimal solution to (1) is obtained by numerical comparison between these two cases in terms of energy consumption. Both cases can be further divided into three offloading scenarios including hybrid NOMA, pure NOMA and OMA based on different power allocation. For hybrid NOMA case, $U_{n,\phi}$ transmits during both $\tau_{m,\phi}$ and $\tau_{r,\phi}$, which indicates $P_{n,\phi} > 0$, $P_{r,\phi} > 0$ and $t_{r,\phi} > 0$. Pure NOMA scheme indicates that $U_{n,\phi}$ only transmits simultaneously with $U_{m,\phi}$ during $\tau_{m,\phi}$, and therefore, $P_{r,\phi} = 0$ and $t_{r,\phi} = 0$. In addition, the OMA case represents that $U_{m,\phi}$ occupies $\tau_{m,\phi}$ solely, and $U_{m,\phi}$ only transmit during $t_{r,\phi}$.

**Remark 2.** Appendix A provides the proof for the case $1_{n,\phi} = 1$. The proof for the case $1_{n,\phi} = 0$ similarly, and it can be referred to the previous work in [21]. Thus, the proof for the case $1_{n,\phi} = 0$ is omitted for this and the following two sub-problems.

In this subsection, the optimal power allocation for the hybrid NOMA scheme is obtained when $t_{r,\phi}$ is fixed, and then the optimization of $t_{r,\phi}$ is further studied to minimize $E_{n,\phi}^{\text{tot}}$ in the following subsection.

**B. Time Scheduling**

The aim of this subsection is to find the optimal time allocation for the second time duration $t_{r,\phi}$ which is solely utilized by $U_{n,\phi}$ for OMA transmission. As aforementioned in Theorem 1, the optimal power allocation for hybrid NOMA scheme is given as a function of $t_{r,\phi}$ and $\beta_{\phi}$. Hence, by fixing $\beta_{\phi}$, (P1) is rewritten as

$$\begin{align*}
\text{(P3)}: \min_{t_{r,\phi}} & \quad \frac{\kappa_0 |C(1 - \beta_{\phi})L|^3}{(\tau_{m,\phi} + t_{r,\phi})^2} + \tau_{m,\phi} P_{n,\phi}^* + t_{r,\phi} P_{r,\phi}^* \\
\text{s.t.} & \quad 0 \leq t_{r,\phi} \leq \tau_{n,\phi} - \tau_{m,\phi} \quad (21a)\end{align*}$$

**Proposition 1.** The offloading energy consumption (21a) is monotonically decreasing with respected to $t_{r,\phi}$ for both $1_{n,\phi} = 1$ and $1_{n,\phi} = 0$ cases. To minimize the energy consumption, the optimal time allocation is to schedule the entire available time before the deadline $\tau_{n,\phi}$, i.e.,

$$t_{r,\phi}^* = \tau_{n,\phi} - \tau_{m,\phi} \quad (22)$$
Proof. Refer to Appendix B.

By assuming all the data is offloaded to the MEC server, the following lemma studies the uplink transmission energy efficiency of the two hybrid NOMA-MEC schemes for $1_{n,\phi} = 0$ and $1_{n,\phi} = 1$.

**Lemma 1.** Assume all data are offloaded to the MEC server, i.e., $\beta_{\phi} = 1$, the solution in (18) for the case $1_{n,\phi} = 0$ has higher energy consumption than the solution in (14) for the case $1_{n,\phi} = 1$, if $|h_{m,\phi}|^{-2} \left( e^{\frac{L}{\tau_{m,\phi}}} - 1 \right) \leq P_{m,\phi} \leq |h_{n,\phi}|^{-2} \left( e^{\frac{L}{\tau_{n,\phi}}} - 1 \right)$.

**Proof.** Without considering local computing, the energy consumption for (14) can be written as

$$E_{1} = \tau_{n,\phi} |h_{n,\phi}|^{-2} \left( e^{\frac{\tau_{n,\phi}}{\tau_{n,\phi}}} - 1 \right),$$

and the energy consumption for the case (18) is given as

$$E_{2} = \tau_{m,\phi} |h_{n,\phi}|^{-2} \left( P_{m,\phi} |h_{m,\phi}|^{2} + 1 \right) \left( e^{\frac{L-(\tau_{n,\phi}-\tau_{m,\phi})}{\rho_{m,\phi}}} \frac{|h_{m,\phi}|^{2} + 1}{e^{\frac{\tau_{m,\phi}}{\tau_{m,\phi}}}} - 1 \right)$$

$$+ (\tau_{n,\phi} - \tau_{m,\phi}) |h_{n,\phi}|^{-2} \left( P_{m,\phi} |h_{m,\phi}|^{2} + 1 \right) e^{\frac{L-(\tau_{n,\phi}-\tau_{m,\phi})}{\rho_{m,\phi}}} \frac{|h_{m,\phi}|^{2} + 1}{e^{\frac{\tau_{m,\phi}}{\tau_{m,\phi}}}} - 1 \right).$$

To proof that $E_{2} \geq E_{1}$, the inequality can be rearranged as

$$- \tau_{m,\phi} |h_{m,\phi}|^{2} + \tau_{n,\phi} e^{\frac{L}{\rho_{m,\phi}}} \left( P_{m,\phi} |h_{m,\phi}|^{2} + 1 \right) \frac{\tau_{m,\phi} - \tau_{n,\phi}}{\tau_{n,\phi}} + 1 \geq \tau_{n,\phi} e^{\frac{L}{\rho_{n,\phi}}}. $$

Define $\zeta(x) = -\tau_{m,\phi} x + \tau_{n,\phi} e^{\frac{L}{\rho_{n,\phi}}} (x + 1) \frac{\tau_{m,\phi} - \tau_{n,\phi}}{\tau_{n,\phi}} + 1$, the first order derivative of $\zeta(x)$ is given as

$$\zeta'(x) = -\tau_{m,\phi} + \left( \frac{\tau_{n,\phi} - \tau_{n,\phi}}{B\tau_{n,\phi}} + 1 \right) \tau_{n,\phi} e^{\frac{L}{\rho_{n,\phi}}} (x + 1) \frac{\tau_{m,\phi} - \tau_{n,\phi}}{\tau_{n,\phi}}. $$

Therefore, $\zeta'(x)$ is monotonically decreasing since $\tau_{m,\phi} < \tau_{n,\phi}$, and the following inequality holds:

$$\zeta'(x) \geq \zeta' \left( e^{\frac{\tau_{n,\phi}}{\tau_{m,\phi}}} - 1 \right) = 0. $$

Hence for $0 \leq x \leq e^{\frac{\tau_{n,\phi}}{\tau_{m,\phi}}} - 1$, $\zeta(x)$ is monotonically increasing, and $\zeta(0) = \tau_{n,\phi} e^{\frac{L}{\rho_{n,\phi}}}$, which illustrates that $E_{2} \geq E_{1}$.

**C. Offloading Task Assignment**

In this subsection, we focus on the optimization of the task assignment coefficient for $U_{n,\phi}$ in each group $\phi$. Given the optimal power allocation and time arrangement, (P1) is reformulated as

$$(P1) : \min_{\beta_{\phi}} \ \ \ \ \ \ \frac{\kappa_{0} \left[ C(1 - \beta_{\phi})L \right]^{3}}{\tau_{m,\phi} + t_{r,\phi}^{*} P_{r,\phi}} + \tau_{m,\phi} P_{n,\phi} + t_{r,\phi}^{*} P_{r,\phi} $$

s.t. $0 \leq \beta_{\phi} \leq 1.$
Proposition 2. The above problem is convex, and the optimal task assignment coefficient can be characterized by those three optimal power allocation schemes for the hybrid NOMA model in (14), (15), and (18), which is given by

$$\beta^*_\phi = 1 - \frac{2}{z_{2,\phi}} \mathcal{W} \left( \frac{1}{2} z_{1,\phi} \frac{z_{2,\phi}}{z_{2,\phi} e^{z_{2,\phi}}} \right),$$

(29)

where $\mathcal{W}$ denotes the single-valued Lambert W function, and $z_{1,\phi}$ and $z_{2,\phi}$ are determined by the different power allocation schemes, which are presented as follows:

(a) $1_{n,\phi} = 1$:

If (14) is adopted:

$$\begin{align*}
z_1 &= \frac{3\kappa_0 B C^3 L^2 |h_{n,\phi}|^2}{\tau^4_{n,\phi}}, \\
z_2 &= \frac{L}{B \tau_{n,\phi}},
\end{align*}$$

(30)

If (15) is adopted:

$$\begin{align*}
z_1 &= \frac{3\kappa_0 [h_{n,\phi}]^2 C^3 L^2 e^{u_{\phi}}}{\tau^4_{n,\phi}}, \\
z_2 &= \frac{L}{B (\tau_{n,\phi} - \tau_{m,\phi})}
\end{align*}$$

(31)

where $u_{\phi} = \frac{\tau_{m,\phi}}{\tau_{n,\phi} - \tau_{m,\phi}} \ln \left( P_{m,\phi} |h_{m,\phi}|^2 \left( e^{\frac{\tau_{n,\phi}}{P_{m,\phi} |h_{m,\phi}|^2} - 1 \right)^{-1} \right)$.

(b) $1_{n,\phi} = 0$:

$$\begin{align*}
z_{1,\phi} &= \frac{3\kappa_0 B C^3 L^2 |h_{n,\phi}|^2 e^{(\tau_{n,\phi} - \tau_{m,\phi}) \ln(P_{m,\phi} |h_{m,\phi}|^2 + 1)}}{\tau_{n,\phi} \left( P_{m,\phi} |h_{m,\phi}|^2 + 1 \right)^{\frac{3}{2}}}, \\
z_{2,\phi} &= \frac{L}{B \tau_{n,\phi}}
\end{align*}$$

(32)

Proof. Refer to Appendix C

Remark 3. Problem (P4) is the lowest level of the proposed multilevel programming method, which provides three task assignment solutions corresponding to the three power allocation schemes (14), (15), and (18) respectively. The final solution to the energy minimization problem (P1) can be obtained by substituting the optimal task assignment into the corresponded power allocation schemes. Then the most energy efficient scheme is selected among (14), (15), and (18) by comparing the numerical energy consumption for each scheme.

IV. DEEP REINFORCEMENT LEARNING FRAMEWORK FOR USER GROUPING

In the previous section, it is assumed that the user grouping is given, and the optimal resource allocation is obtained in closed-form. The optimal user grouping can be obtained by exploring all possible user grouping combinations and find the one with the lowest energy consumption. Although this method can obtain the optimal user pairing scheme, the complexity of the exhaustive search method is high, and it is not possible to output real time decisions. Therefore, we propose a fast converge user pairing training algorithm based on DQN to obtain the user grouping policy, which is introduced in the following subsection, in which the state space, action space and reward function are defined. Subsequently, the training algorithm for the user grouping policy is provided.
A. The DRL Framework

The optimization of user grouping is modeled as a DRL task, where the base station is treated as the agent to interact with the environment which is defined as the MEC network. In each time slot $t$, the agent takes an action $a_t$ from the action space $A$ to assign users into pairs according to an optimal policy which is learned by the DNN. The action taken under current state $s_t$ results in an immediate reward $r_t$, which is obtained at the beginning of the next time slot, and then move to the next state $s_{t+1}$. In this problem, the aforementioned terms are defined as follows.

1) State Space: The state $s_t \in S$ is characterized by the current channel gains and offloading deadlines of all users since the user grouping is mainly determined by those two factors. Therefore, the state $s_t$ can be expressed as

$$s_t = \{h_1[t], h_2[t], \ldots, h_k[t], \ldots, h_K[t]; \tau_1[t], \tau_2[t], \ldots, \tau_k[t], \ldots, \tau_K[t]\}.$$ (33)

2) Action Space: At each time slot $t$, the agent takes a action $a_t \in A$, which contains all the possible user grouping decisions $j_{k,\phi}$. The action is defined as

$$a_t = \{j_{1,1}[t], \ldots, j_{k,\phi}[t], \ldots, j_{K,\Phi}[t]\},$$ (34)

where $j_{k,\phi} = 1$ indicates that $U_k$ is assigned to group $\phi$. In our proposed scheme, each group can only be assigned with two different users.

3) Rewards: The immediate reward $r_t$ is described by the sum of the energy consumption of each groups after choosing the action $a_t$ under state $s_t$. The numerical result of the energy consumption in each group can be obtained by solving the problem (P1). Therefore, the reward is defined as

$$r_t = - \sum_{\phi=1}^{\Phi} E_{\phi}^{\text{tot}}[t]$$ (35)

The aim of the agent is to find an optimal policy that maximizes the long-term discounted reward, which can be written as

$$R_t = r_t + \gamma r_{t+1} + \gamma^2 r_{t+2} + \ldots$$

$$= \sum_{i=0}^{\infty} \gamma^i r_{t+i},$$ (36)

where $\gamma \in [0, 1]$ is the discount factor which balance the immediate reward and the long-term reward.

B. DQN-based NOMA User Grouping Algorithm

To accommodate the reward maximization problem, a DQN-based user grouping algorithm is proposed in this paper, illustrated in Fig. 2. In the conventional Q-learning, Q-table is obtained to describe the quality of an action for a given state, and the agent chooses actions according to the Q-values to maximize the reward. However, it will be slow for the system to obtain Q-values for all the state-action pairs if the state space and action space are large. Therefore, to speed up the learning process, instead of generating and processing all possible Q-values,
DNNs are introduced to estimate the Q-values based on the weight of DNNs. We utilize a DNN to estimate the Q-value denoted by Q-network, which the Q-estimation is represented as $Q(s_t, a_t; \theta)$, and an additional DNN with the same setting to generate the target network with $Q(s_t, a_t; \theta^-)$ for training, where $\theta$ and $\theta^-$ are the weights of the DNNs.

We adopt $\epsilon$-greedy policy with $0 < \epsilon < 1$ to balance the exploration of new actions and the exploitation of known actions by either randomly choosing an action $a_t \in A$ with probability $\epsilon$ to avoid the agent sticking on non-optimal actions or picking the best action with the probability $1 - \epsilon$ such that [29]:

$$a_t = \arg \max_{a_t \in A} Q(s_t, a_t; \theta).$$

Generally, the threshold $\epsilon$ is fixed, which indicates the probability of choosing random action remains the same throughout the whole learning period. However, it brings fluctuation when the algorithm converges and may lead to diverge again in extreme cases. In this paper, we adopt an $\epsilon$-greedy decay scheme, which a large $\epsilon^+$ (more greedy) is given at the beginning, and then the it decays with each training step until a certain small probability $\epsilon^-$. The above policy encourages the agent to explore the never-selected actions at the beginning, and then the agent intends to take more large reward-guaranteed actions when the network is already converged.

The target network only updates every certain iterations, which provides a relatively stable label for the estimation network. The agent stores the tuples $(s_t, a_t, r_t, s_{t+1})$ as experiences to a memory buffer $R$, and a mini-batch of samples from the memory are fed into the target network to generate the Q-values labels, which is given by...
Algorithm 1 DQN-based User Grouping Algorithm

1: **Parameter initialization:**

2: Initialize Q-network $Q(s_i, a_i; \theta)$ and target network $Q(s_i, a_i; \theta^-)$.

3: Initialize Reply memory $\mathcal{R}$ with size $|\mathcal{R}|$, and memory counter.

4: Initialize $\gamma$, $\epsilon^+$, $\epsilon^-$, decay step, batch size, target network update interval $\delta_{up}$.

5: **Training Phase:**

6: for $episode = 1, 2, ..., N_{episode}$ do

7: for $time step = 1, 2, ..., N_t$s do

8: Input state $s_t$ into Q-network and obtain Q-values for all actions.

9: Take the user grouping decision as action $a_t$ based on the $\epsilon$-greedy decay policy.

10: Agent receive the reward $r_t$ based on (35) and the observation to next state $s_{t+1}$.

11: Store the experience tuple $(s_t, a_t, r_t, s_{t+1})$ into the memory $\mathcal{R}$.

12: if memory counter $>|\mathcal{R}|$ then

13: Remove the old experiences from the beginning.

14: end if

15: Randomly sample a mini-batch of the experience tuples $(s_t, a_t, r_t, s_{t+1})$ with batch size and feed into the DNNs.

16: Update the Q-network weights $\theta$ by calculating the Loss function (39).

17: Replace $\theta^-$ by $\theta$ after every $\delta_{up}$ steps.

18: end for

19: end for

\[
y_t = r_t + \max_{a_{i+1} \in A} Q(s_{i+1}, a_{i+1}; \theta^-), \quad \forall i \in \mathcal{R} \tag{38}\]

Hence, the loss function for the Q-network can be expressed as

\[
Loss(\theta) = (y_i - Q(s_i, a_i; \theta)), \quad \forall i \in \mathcal{R} \tag{39}\]

The Q-network can be trained by minimizing the loss function to obtain the new $\theta$, and the weights of the target network is updated after $\delta_{up}$ steps by replacing $\theta^-$ with $\theta$. The whole DQN-based user grouping framework is summarized in Algorithm 1.

V. SIMULATION RESULTS

In this section, several simulation results are presented to evaluate the convergence and effectiveness of the proposed joint resource allocation and user grouping scheme. Specifically, the impact of learning rate, user number, offloading data length, and delay tolerance are investigated. Moreover, the proposed hybrid SIC scheme is compared to some benchmarks including QoS based SIC scheme and other NOMA and OMA schemes.
TABLE I: System parameters

| Parameter                                    | Value     |
|----------------------------------------------|-----------|
| Effective capacitance coefficient            | $10^{-28}$|
| Number of CPU cycles required per bit        | $10^3$    |
| Transmission bandwidth $B$                   | 2 MHz     |
| Path loss exponent $\alpha$                 | 3.76      |
| Noise spectral density $N_0$                 | $-174$ dBm/Hz |
| Maximum cell radius                         | 1000 m    |
| Minimum distance to base station             | 50 m      |

TABLE II: Hyper-parameters

| Parameter                              | Value   |
|----------------------------------------|---------|
| $\epsilon$-greedy coefficient         | 0.5 – 0.01 |
| $\epsilon$-greedy decay steps         | 2000    |
| Discount factor $\gamma$               | 0.7     |
| Reply memory size $R$                  | 20000   |
| Batch size                             | 64      |
| Target network update interval $\delta_{up}$ | 10     |
| Number of episode $N_{episode}$        | 150     |
| Number of time steps $N_{ts}$          | 500     |

The system parameters are set up as follows. All users are distributed uniformly and randomly in a disc-shape cell where the base station located in the cell center. The total number of users is six, and each of them has a task contains 2 Mbit of data for offloading. As aforementioned, the delay sensitive primary user $U_{m,\phi}$ is allocated with a predefined power which is $P_{m,\phi} = 1$ W for all groups in the simulation. The delay tolerance for each user is given randomly between [0.2, 0.3] seconds. In addition, the rest of the system parameters are listed in Table I.

To implement the DQN algorithm, the two DNNs are configured with the same settings, where each of them consists of four fully connected layers, and two of which are hidden layers with 200 and 100 neurons respectively. The activation function we adopted for all hidden layers is Rectified Linear Unit (ReLU), i.e., $f(x) = \max(0, x)$, and the final output layer is activated by Tanh of which the range is $(-1, 1)$ [30]. The Adaptive moment estimation optimizer (Adam) method is used to learn the DNN weight $\theta$ with given learning rate [31]. The rest of the hyper-parameters are listed in Table II. All simulation results are obtained with PyTorch 1.70 and CUDA 11.1 on Python 3.8 platform.

A. Convergence of Framework

In this part, we evaluate the convergence of the proposed DQN based user pairing algorithm. Fig. 3 compares the convergence rate of the average reward for each episode under different learning rate, which is described by the average energy consumption. Learning rate controls how much it should be to adjust the weights of a DNN based on the network loss, and we set the learning rate $= [0.1, 0.01, 0.001]$ to observe its influence to the convergence. The network with 0.1 learning rate converges slightly faster than the one with 0.01 learning rate, and both of them
Fig. 3: Average energy consumption versus training episodes with different learning rates.

Fig. 4: Average energy consumption versus training episodes with different numbers of users.

converge much faster than the network with 0.001 learning rate. However, when the learning rate is 0.1, even though the large learning has better convergence, it overshoots the minimum and therefore has higher energy consumption after converge than other two plots. Therefore, the most suitable learning rate for our proposed DQN algorithm is 0.01, which is adopted to obtain the rest of simulation results in this paper.

Fig. 4 illustrates the effectiveness of the DQN user grouping algorithm proposed in this paper. By setting the numbers of users to \([6, 8, 10]\), the algorithm shows a similar performance that the average energy consumption decreases over training and converges within the first 20 episodes for the all three cases. Moreover, more users in the network can result in higher energy consumption, and the algorithm shows the superior performance over the
random policy, which reduces the energy consumption significantly.

The $\epsilon$-greedy decay policy to the convergence performance is further investigated in Fig. 5. The $\epsilon$-greedy coefficient for the blue curve is set to 0.1 while the red curve adopts the $\epsilon$-greedy decay policy following the parameters in Table II. Since the decay policy starts with large $\epsilon$, the network is more likely to choose the random action at the beginning, and hence the energy consumption is higher at the beginning. With $\epsilon$ decays over episode, the network chooses the actions which have been selected before that guarantees large rewards, and therefore it is more stable afterwards. Meanwhile, the network without the decay policy has significant fluctuations during the training since it has a greater chance to choose the random actions throughout the training. However, if a very small $\epsilon$ is adopted, the network will be less likely to explore some actions, which may stick on the non-optimal actions.

**B. Average Performance of Proposed Scheme**

In this part, we present the average performance of the proposed NOMA-MEC scheme to show the impact of $P_{m,\phi}$, offloading data length, and maximum delay tolerance. Meanwhile, our proposed scheme is compared with the one without task assignment and OMA offloading to show the superior performance gap. As shown in Fig. 6, the energy consumption of both hybrid-SIC schemes raises and then decreases as $P_{m,\phi}$ increases. Since $P_{m,\phi}$ is relatively small at the beginning, $U_{m,\phi}$ is not likely to be decoded first to satisfy the constraint (12c) in the case $1_{n,\phi} = 1$. Therefore, $U_{n,\phi}$ is more likely to be decoded in prior, and increasing $P_{m,\phi}$ causes more interference to $U_{n,\phi}$. After the power indicated by the arrows, the case $1_{n,\phi} = 1$ becomes feasible for both with and without task assignment schemes, and it is evident that the case $1_{n,\phi} = 1$ has better energy efficiency compared to the case $1_{n,\phi} = 0$. Moreover, the hybrid-SIC scheme with task assignment outperforms the one without task assignment in the blue line. The one with task assignment have a wider lower-bound of the feasible range of the power allocation for case $1_{n,\phi} = 1$ in (14), which means that it can adopt the $1_{n,\phi} = 1$ case with smaller $P_{m,\phi}$. In addition, both
Fig. 6: Average energy consumption versus training episodes with different numbers of users.

hybrid SIC schemes has lower energy consumption than the OMA scheme. In Fig. 7, the energy consumption is presented as a function of the offloading data length. As the data length increases, the average energy consumption also grows. Our proposed hybrid-SIC scheme reduces the energy consumption significantly especially when the data length is large. Moreover, Fig. 8 reveals the energy consumption comparisons versus the maximum delay tolerance for $U_{n, \phi}$. With tight deadlines, the energy consumption of the hybrid-SIC scheme is much lower than OMA scheme, and more portion of data is processed locally to save energy compared to the fully offloading curve.
This paper studied the resource allocation problem for a NOMA-assisted MEC network to minimize the energy consumption of users’ offloading activities. The hybrid NOMA scheme has two duration during each time slot, in which NOMA is adopted to serve the both users simultaneously during the first time duration, and a dedicate time slot is scheduled to offload the remaining part of the delay tolerable user solely by OMA. Upon fixing the user grouping, the non-convex problem was decomposed into three sub-problems including power allocation, time allocation and task assignment, which were all solved optimally by studying the convexity and monotonicity. The hybrid SIC scheme selects the SIC decoding order dynamically by a numerical comparison of the energy consumption between different decoding sequences. Finally, after solving those sub-problems, we proposed a DQN based user grouping algorithm to obtain the user grouping policy and minimize the long-term average offloading energy consumption. By comparing with various benchmarks, the simulation results proved the superiority of the proposed NOMA-MEC scheme in terms of energy consumption.
APPENDIX

A. Proof of Theorem 1

By fixing $t_{r,\phi}$ and $\beta_{\phi}$, the above problem in the case $1_{n,\phi} = 1$ can be rewritten as:

\[ (P5) : \min_{P_{n,\phi}, P_{r,\phi}} \frac{\kappa_0 [C(1 - \beta_{\phi})L]^3}{(\tau_{m,\phi} + t_{r,\phi})^2} + \tau_{m,\phi} P_{n,\phi} + t_{r,\phi} P_{r,\phi} \]

s.t.

\[ \tau_{m,\phi} B \ln \left( 1 + P_{n,\phi} |h_{n,\phi}|^2 \right) + t_{r,\phi} B \ln \left( 1 + P_{r,\phi} |h_{n,\phi}|^2 \right) \geq \beta_{\phi} L \]

\[ \tau_{m,\phi} B \ln \left( 1 + \frac{P_{m,\phi} |h_{m,\phi}|^2}{P_{n,\phi} |h_{n,\phi}|^2 + 1} \right) \geq L \]

\[ P_{n,\phi} \geq 0, P_{r,\phi} \geq 0 \]

It is evident that the problem is convex, and by rearranging (40d) as

\[ P_{n,\phi} |h_{n,\phi}|^2 - P_{m,\phi} |h_{m,\phi}|^2 \left( e^{\frac{L}{\tau_{m,\phi}}} - 1 \right)^{-1} + 1 \leq 0, \]

the Lagrangian function can be obtained as follows:

\[ \mathcal{L}(P_{n,\phi}, P_{r,\phi}, \lambda) = \kappa_0 \frac{[C(1 - \beta_{\phi})L]^3}{(\tau_{m,\phi} + t_{r,\phi})^2} + \tau_{m,\phi} P_{n,\phi} + t_{r,\phi} P_{r,\phi} - \lambda_1 P_{n,\phi} - \lambda_2 P_{r,\phi} + \lambda_3 \beta_{\phi} L \]

\[ - \lambda_3 \tau_{m,\phi} B \ln \left( 1 + P_{n,\phi} |h_{n,\phi}|^2 \right) - \lambda_3 t_{r,\phi} B \ln \left( 1 + P_{r,\phi} |h_{n,\phi}|^2 \right) \]

\[ + \lambda_4 \left( P_{n,\phi} |h_{n,\phi}|^2 - P_{m,\phi} |h_{m,\phi}|^2 \left( e^{\frac{L}{\tau_{m,\phi}}} - 1 \right)^{-1} + 1 \right), \]

where $\lambda \triangleq [\lambda_1, \lambda_2, \lambda_3, \lambda_4]$ are the Lagrangian multipliers. The stationary conditions are given as

\[ \frac{\partial \mathcal{L}}{\partial P_{n,\phi}} = \tau_{m,\phi} - \lambda_1 - \lambda_3 \tau_{m,\phi} B \frac{|h_{n,\phi}|^2}{P_{n,\phi} |h_{n,\phi}|^2 + 1} + \lambda_4 |h_{n,\phi}|^2 = 0 \]

\[ \frac{\partial \mathcal{L}}{\partial P_{r,\phi}} = t_{r,\phi} - \lambda_2 - \lambda_3 t_{r,\phi} B \frac{|h_{n,\phi}|^2}{P_{r,\phi} |h_{n,\phi}|^2 + 1} = 0 \]
The Karush–Kuhn–Tucker (KKT) conditions [32] can be obtained as

\[
\begin{align*}
\beta \phi L - \tau_{m,\phi} B \ln \left( 1 + P_{n,\phi} |h_{n,\phi}|^2 \right) - t_{r,\phi} B \ln \left( 1 + P_{r,\phi} |h_{n,\phi}|^2 \right) &\leq 0 \quad (44a) \\
P_{n,\phi} |h_{n,\phi}|^2 - P_{m,\phi} |h_{m,\phi}|^2 \left( e^{\frac{\beta \phi L}{B}} - 1 \right)^{-1} + 1 &\leq 0 \quad (44b) \\
-P_{n,\phi} &\leq 0, -P_{r,\phi} \leq 0 \quad (44c) \\
\lambda_i &\geq 0, \quad i \in \{1, 2, 3, 4\} \quad (44d) \\
\lambda_1 P_{n,\phi} &= 0 \quad (44e) \\
\lambda_2 P_{r,\phi} &= 0 \quad (44f) \\
\lambda_3 \beta \phi L - \lambda_3 \tau_{m,\phi} B \ln \left( 1 + P_{n,\phi} |h_{n,\phi}|^2 \right) - \lambda_3 t_{r,\phi} B \ln \left( 1 + P_{r,\phi} |h_{n,\phi}|^2 \right) &\leq 0 \quad (44g) \\
\lambda_4 \left( P_{n,\phi} |h_{n,\phi}|^2 - P_{m,\phi} |h_{m,\phi}|^2 \right) \left( e^{\frac{\beta \phi L}{B}} - 1 \right)^{-1} + 1 &\leq 0 \quad (44h) \\
\tau_{m,\phi} - \lambda_1 - \lambda_3 \tau_{m,\phi} B \frac{|h_{n,\phi}|^2}{P_{n,\phi} |h_{n,\phi}|^2 + 1} &+ \lambda_4 |h_{n,\phi}|^2 = 0 \quad (44i) \\
t_{r,\phi} - \lambda_2 - \lambda_3 t_{r,\phi} B \frac{|h_{n,\phi}|^2}{P_{r,\phi} |h_{n,\phi}|^2 + 1} &\leq 0 \quad (44j)
\end{align*}
\]

The power allocation schemes can be obtained by different Lagrangian multipliers decisions as follows

- **Hybrid NOMA**: \( \lambda_1 = 0, \lambda_2 = 0, \) and \( \lambda_3 \neq 0 \).
  - If \( \lambda_4 = 0 \):
    \[
    P_{n,\phi}^* = P_{r,\phi}^* = |h_{n,\phi}|^{-2} \left( e^{\frac{\beta \phi L}{B}} - 1 \right) \quad (45)
    \]
    \[
    P_{m,\phi} |h_{m,\phi}|^2 \geq e^{\frac{\beta \phi L}{B} \tau_{m,\phi}} \left( e^{\frac{\beta \phi L}{B} t_{r,\phi}} - 1 \right) \quad (46)
    \]
  - If \( \lambda_4 \neq 0 \):
    \[
    P_{n,\phi}^* = |h_{n,\phi}|^{-2} \left[ P_{m,\phi} |h_{m,\phi}|^2 \left( e^{\frac{\beta \phi L}{B} \tau_{m,\phi}} - 1 \right)^{-1} - 1 \right] \quad (47a)
    \]
    \[
    P_{r,\phi}^* = |h_{n,\phi}|^{-2} \left\{ e^{\frac{\beta \phi L}{B} t_{r,\phi}} - \frac{\tau_{m,\phi} P_{m,\phi} |h_{m,\phi}|^2}{P_{r,\phi} |h_{n,\phi}|^2 + 1} \left[ e^{\frac{\beta \phi L}{B} \tau_{m,\phi}} - 1 \right]^{-1} \right\} \quad (47b)
    \]
    where \( e^{\frac{\beta \phi L}{B} \tau_{m,\phi}} \left( e^{\frac{\beta \phi L}{B} t_{r,\phi}} - 1 \right) \geq P_{m,\phi} |h_{m,\phi}|^2 \geq e^{\frac{\beta \phi L}{B} \tau_{m,\phi}} - 1. \)

- **Pure NOMA**: \( \lambda_1 = 0, \lambda_2 \neq 0 \):
  \[
  P_{n,\phi} = |h_{n,\phi}|^{-2} \left( e^{\frac{\beta \phi L}{B \tau_{m,\phi}}} - 1 \right) \quad (48a)
  \]
  \[
  P_{r,\phi} = 0 \quad (48b)
  \]
  where \( P_{m,\phi} |h_{m,\phi}|^2 \geq e^{\frac{\beta \phi L}{B \tau_{m,\phi}}} \left( e^{\frac{\beta \phi L}{B \tau_{m,\phi}}} - 1 \right). \)
B. Proof of Proposition 1

The total energy consumption can be expressed as:

\[
E_{H1} = \frac{\kappa_0 [C(1 - \beta_\phi)L]^3}{(\tau_{m,\phi} + t_{r,\phi})^2} + t_{m,\phi}|h_{n,\phi}|^{-2} \left[ P_{m,\phi}|h_{m,\phi}|^2 \left( e^{\frac{L}{\tau_{m,\phi}} - 1} \right)^{-1} - 1 \right] + t_{r,\phi}|h_{n,\phi}|^{-2} \left\{ \frac{\beta_\phi L}{m,\phi} - \frac{\tau_{m,\phi}}{r,\phi} \ln \left[ P_{m,\phi}|h_{m,\phi}|^2 \left( e^{\frac{L}{\tau_{m,\phi}} - 1} \right)^{-1} \right] - 1 \right\},
\]

where \( a_\phi = \frac{\beta_\phi L - B r,\phi}{B} \ln \left[ P_{m,\phi}|h_{m,\phi}|^2 \left( e^{\frac{L}{\tau_{m,\phi}} - 1} \right)^{-1} \right] \).

Define \( g(x) = e^{\frac{a_\phi}{x}} - \frac{a_\phi}{x} e^{\frac{a_\phi}{x}} - 1 \),

\[
g'(x) = \frac{a_\phi^2 e^{\frac{a_\phi}{x}} - a_\phi^2 e^{\frac{a_\phi}{x}}}{x^3} \geq 0, \quad \forall x > 0.
\]

Hence, \( g(x) \) is monotonically increasing for \( x > 0 \), and \( g(t_{r,\phi}) \leq g(\infty) = 0 \). Therefore, \( \frac{dE_{H1}}{dt_{r,\phi}} \leq 0 \), which is monotonically decreasing. Hence, the larger \( t_{r,\phi} \) is scheduled, the less energy is consumed, and the optimal situation is when \( t_{r,\phi}^* = \tau_{n,\phi} - \tau_{m,\phi} \).

For the power allocation scheme in (15), the energy consumption is given as

\[
E_{H2} = \frac{\kappa_0 [C(1 - \beta_\phi)L]^3}{(\tau_{m,\phi} + t_{r,\phi})^2} + (\tau_{m,\phi} + t_{r,\phi})|h_{n,\phi}|^{-2} \left( e^{\frac{\beta_\phi L}{n,\phi} + t_{r,\phi}} - 1 \right) - 1) \right) - 1)
\]

By obtaining the derivative with respect to \( t_{r,\phi} \),

\[
\frac{\partial E_{H2}}{\partial t_{r,\phi}} = -\frac{2\kappa_0 [C(1 - \beta_\phi)L]^3}{(\tau_{m,\phi} + t_{r,\phi})^2} + |h_{n,\phi}|^{-2} \left( e^{\frac{\beta_\phi L}{n,\phi} + t_{r,\phi}} - 1 \right) - 1) \right) - 1)
\]

Define \( g_2(x) \equiv e^{\frac{\beta_\phi L}{n,\phi} + t_{r,\phi}} - x \frac{\beta_\phi L}{B (\tau_{m,\phi} + t_{r,\phi})} e^{\frac{\beta_\phi L}{n,\phi} + t_{r,\phi}} - 1 \), and the derivative of \( g_2(x) \) is

\[
g'_2(x) = \frac{(\beta_\phi L)^2}{B^2 (\tau_{m,\phi} + t_{r,\phi})^2} e^{\frac{\beta_\phi L}{n,\phi} + t_{r,\phi}} \geq 0, \quad \forall x > 0.
\]

Thus, \( g_2(x) \) is monotonically increasing for \( x > 0 \), and \( g(t_{r,\phi}) \leq g(\infty) = 0 \), which indicates \( \frac{dE_{H2}}{dt_{r,\phi}} \leq 0 \). Similar to the previous case, the energy function is monotonically decreasing with respect to \( t_{r,\phi} \), and the optimal time allocation is \( t_{r,\phi}^* = \tau_{n,\phi} - \tau_{m,\phi} \).
C. Proof to Proposition 2

\[
\min_{\beta_\phi} \quad \frac{\kappa_0 \left[C(1 - \beta_\phi)L\right]^3}{\left(\tau_{m,\phi} + t^*_{r,\phi}\right)^2} + \tau_{m,\phi}P^*_{n,\phi} + t^*_{r,\phi}P^*_{r,\phi} 
\]
\[\text{s.t.} \quad 0 \leq \beta_\phi \leq 1. \quad (56a)\]

The Lagrangian is given as

\[
L(\beta_\phi, \lambda_5, \lambda_6) = \frac{\kappa_0 \left[C(1 - \beta_\phi)L\right]^3}{\left(\tau_{m,\phi} + t^*_{r,\phi}\right)^2} + \tau_{m,\phi}P^*_{n,\phi} + t^*_{r,\phi}P^*_{r,\phi} - \lambda_5 \beta_\phi + \lambda_6 (\beta_\phi - 1) 
\]
\[\text{(57)}\]

• For the case \(P_{m,\phi} = P_{n,\phi}\) in (14), the stationary condition is obtained as

\[
\frac{\partial L}{\partial \beta_\phi} = \frac{-3\kappa_0 (CL)^3 (1 - \beta_\phi)^2}{\tau_{n,\phi}^2} + \frac{L}{B} |h_{n,\phi}| - 2 e^{\beta_\phi \frac{\beta_\phi L}{\tau_{n,\phi}}} - \lambda_5 + \lambda_6 = 0.
\]
\[\text{(58)}\]

Therefore, the KKT conditions can be written as follows:

\[
-\beta_\phi \leq 0 \quad (59a)
\]
\[
\beta_\phi - 1 \leq 0 \quad (59b)
\]
\[
\lambda_5 \beta_\phi = 0 \quad (59c)
\]
\[
\lambda_6 (\beta_\phi - 1) = 0 \quad (59d)
\]

\[
\frac{-3\kappa_0 (CL)^3 (1 - \beta_\phi)^2}{\tau_{n,\phi}^2} + \frac{L}{B} |h_{n,\phi}| - 2 e^{\beta_\phi \frac{\beta_\phi L}{\tau_{n,\phi}}} - \lambda_5 + \lambda_6 = 0 \quad (59e)
\]

For \(\beta_\phi > 0, \lambda_5 = \lambda_6 = 0, \) and (59e) can be rewritten as

\[
\frac{3\kappa_0 (CL)^3 (1 - \beta_\phi)^2}{\tau_{n,\phi}^2} = \frac{L}{B} |h_{n,\phi}| - 2 e^{\beta_\phi \frac{\beta_\phi L}{\tau_{n,\phi}}}.
\]
\[\text{(60)}\]

Define \(z_{1,\phi} = \frac{3\kappa_0 BC^3 L^2 |h_{n,\phi}|}{\tau_{n,\phi}^2}, \) \(z_{2,\phi} = \frac{L}{B \tau_{n,\phi}}, \) and \(b_\phi = (1 - \beta_\phi),\) the optimal task assignment coefficient can be derived as

\[
z_{1,\phi} b_\phi^2 = e^{z_{2,\phi}(1 - b_\phi)},
\]
\[\text{(61)}\]

\[
b_\phi = \frac{2}{z_{2,\phi} W \left(\frac{1}{2} z_{1,\phi} z_{2,\phi} e^{\frac{z_{2,\phi}}{2}}\right)}.
\]
\[\text{(62)}\]

The optimal task assignment ratio can be expressed as

\[
\beta^*_\phi = 1 - b = 1 - \frac{2}{z_{2,\phi} W \left(\frac{1}{2} z_{1,\phi} z_{2,\phi} e^{\frac{z_{2,\phi}}{2}}\right)}.
\]
\[\text{(63)}\]

• For the case \(P_{m,\phi} \neq P_{n,\phi}\) in (15):

The stationary condition can be expressed as

\[
\frac{\partial L}{\partial \beta_\phi} = \frac{-3\kappa_0 (CL)^3 (1 - \beta_\phi)^2}{\tau_{n,\phi}^2} + |h_{n,\phi}| - 2 e^{\beta_\phi \frac{\beta_\phi L}{B (\tau_{n,\phi} - \tau_{m,\phi})}} - \lambda_5 + \lambda_6 = 0,
\]
\[\text{(64)}\]
where \( u_{\phi} = \frac{\tau_{m,\phi}}{(\tau_{n,\phi} - \tau_{m,\phi})} \ln \left[ P_{m,\phi} |h_{m,\phi}|^2 \left( e^{\frac{\tau_{m,\phi}}{\tau_{n,\phi}}} - 1 \right)^{-1} \right] \).

\[- \beta_{\phi} \leq 0 \quad (65a)\]
\[\beta_{\phi} - 1 \leq 0 \quad (65b)\]
\[\lambda_5 \beta_{\phi} = 0 \quad (65c)\]
\[\lambda_6 (\beta_{\phi} - 1) = 0 \quad (65d)\]
\[-3\kappa_0 (CL)^3 (1 - \beta_{\phi})^2 + |h_{n,\phi}|^{-2} \frac{L}{B} e^{-u_{\phi}} e^{\frac{\beta_{\phi} L}{B(\tau_{n,\phi} - \tau_{m,\phi})}} - u_{\phi} - \lambda_5 + \lambda_6 = 0 \quad (65e)\]

For \( \beta_{\phi} > 0, \lambda_5 = \lambda_6 = 0 \), constraint (65e) can be rearranged as
\[\frac{3\kappa_0 (CL)^3 (1 - \beta_{\phi})^2}{\tau_{n,\phi}^2} + |h_{n,\phi}|^{-2} \frac{L}{B} e^{-u_{\phi}} e^{\frac{\beta_{\phi} L}{B(\tau_{n,\phi} - \tau_{m,\phi})}} = u_{\phi} \quad (66)\]
\[3\kappa_0 B|h_{n,\phi}|^2 (CL)^3 e^{2u_{\phi}} (1 - \beta_{\phi})^2 = e^{\frac{\beta_{\phi} L}{B(\tau_{n,\phi} - \tau_{m,\phi})}} \quad (67)\]

Define \( z_{1,\phi} = \frac{3\kappa_0 B|h_{n,\phi}|^2 (CL)^3 e^{2u_{\phi}}}{\tau_{n,\phi}^2}, z_{2,\phi} = \frac{L}{B(\tau_{n,\phi} - \tau_{m,\phi})} \), the above equation can be rewritten as
\[z_{1,\phi} b_{\phi}^2 = e^{z_{2,\phi}(1 - b_{\phi})} \quad (68)\]
\[b_{\phi} = \frac{2}{z_{2,\phi}} W \left( \frac{1}{2} z_{1,\phi} z_{2,\phi} e^{\frac{z_{2,\phi}}{2}} \right) \quad (69)\]

Hence the optimal task partition assignment ratio is:
\[\beta_{\phi}^* = 1 - b_{\phi} = 1 - \frac{2}{z_{2,\phi}} W \left( \frac{1}{2} z_{1,\phi} z_{2,\phi} e^{\frac{z_{2,\phi}}{2}} \right) \quad (70)\]

REFERENCES

[1] M. Nduwayezu, Q. Pham, and W. Hwang, “Online computation offloading in NOMA-based multi-access edge computing: A deep reinforcement learning approach,” IEEE Access, vol. 8, pp. 99 098–99 109, 2020.
[2] W. Sun, H. Zhang, R. Wang, and Y. Zhang, “Reducing offloading latency for digital twin edge networks in 6G,” IEEE Trans. Veh. Technol., vol. 69, no. 10, pp. 12 240–12 251, 2020.
[3] R. Zhao, X. Wang, J. Xia, and L. Fan, “Deep reinforcement learning based mobile edge computing for intelligent internet of things,” Physical Communication, vol. 43, p. 101184, 2020.
[4] L. Li, Q. Cheng, X. Tang, T. Bai, W. Chen, Z. Ding, and Z. Han, “Resource allocation for NOMA-MEC systems in ultra-dense networks: A learning aided mean-field game approach,” IEEE Trans. Wireless Commun., vol. 20, no. 3, pp. 1487–1500, 2021.
[5] T. Bai, C. Pan, Y. Deng, M. Elkashlan, A. Nallanathan, and L. Hanzo, “Latency minimization for intelligent reflecting surface aided mobile edge computing,” IEEE J. Sel. Areas Commun., vol. 38, no. 11, pp. 2666–2682, 2020.
[6] H. T. Dinh, C. Lee, D. Niyato, and P. Wang, “A survey of mobile cloud computing: architecture, applications, and approaches,” Wireless Commun. Mobile Comput., vol. 13, no. 18, pp. 1587–1611, 2013.
[7] N. Abbas, Y. Zhang, A. Taherkordi, and T. Skeie, “Mobile edge computing: A survey,” IEEE Internet Things J., vol. 5, no. 1, pp. 450–465, 2018.
[8] Y. Huang, Y. Liu, and F. Chen, “NOMA-aided mobile edge computing via user cooperation,” IEEE Trans. Commun., vol. 68, no. 4, pp. 2221–2235, 2020.
[9] A. Chen, Z. Yang, B. Lyu, and B. Xu, “System delay minimization for NOMA-based cognitive mobile edge computing,” IEEE Access, vol. 8, pp. 62 228–62 237, 2020.
[10] Y. Mao, C. You, J. Zhang, K. Huang, and K. B. Letaief, “A survey on mobile edge computing: The communication perspective,” IEEE Commun. Surveys Tutorials, vol. 19, no. 4, pp. 2322–2358, 2017.

[11] N. N. Dao, Q. V. Pham, N. H. Tu, T. T. Thanh, V. N. Q. Bao, D. S. Lakew, and S. Cho, “Survey on aerial radio access networks: Toward a comprehensive 6G access infrastructure,” IEEE Commun. Surveys Tutorials, pp. 1–1, 2021.

[12] B. Makki, K. Chitti, A. Behravan, and M. S. Alouini, “A survey of NOMA: Current status and open research challenges,” IEEE Open Journal of the Communications Society, vol. 1, pp. 179–189, 2020.

[13] V. Mnih, K. Kavukcuoglu, D. Silver, A. A. Rusu, J. Veness, M. G. Bellemare, A. Graves, M. A. Riedmiller, A. Fidjeland, G. Ostrovski, S. Petersen, C. Beattie, A. Sadik, I. Antonoglou, H. King, D. Kumaran, S. Legg, and D. Hassabis, “Human-level control through deep reinforcement learning.” Nature, vol. 518, no. 7540, pp. 529–533, 2015.

[22] V. Mnih, K. Kavukcuoglu, D. Silver, A. A. Rusu, J. Veness, M. G. Bellemare, A. Graves, M. A. Riedmiller, A. Fidjeland, G. Ostrovski, S. Petersen, C. Beattie, A. Sadik, I. Antonoglou, H. King, D. Kumaran, D. Wierstra, S. Legg, and D. Hassabis, “Human-level control through deep reinforcement learning.” Nature, vol. 518, no. 7540, pp. 529–533, 2015.

[30] A. F. Agarap, “Deep learning using rectified linear units (ReLU),” 2018, cite arxiv:1803.08375 Comment: 7 pages, 11 figures, 9 tables. [Online]. Available: http://arxiv.org/abs/1803.08375

[31] ———, “Unveiling the importance of SIC in NOMA systems—part II: State of the art and recent findings,” IEEE Commun. Lett., vol. 24, no. 11, pp. 2373–2377, 2020.

[32] D. Kingma and J. Ba, “Adam: A method for stochastic optimization,” International Conference on Learning Representations, 12 2014.

[33] S. Boyd and L. Vandenberghe, Convex Optimization. USA: Cambridge University Press, 2004.