Demand Response in NOMA-Based Mobile Edge Computing: A Two-Phase Game-Theoretical Approach

Guangming Cui, Qiang He, *Senior Member, IEEE*, Xiaoyu Xia, Feifei Chen, *Member, IEEE*, Tao Gu, *Senior Member, IEEE*, Hai Jin, *Fellow, IEEE*, and Yun Yang, *Senior Member, IEEE*

Abstract—Mobile edge computing (MEC), as a key technology that facilitates 5G networks, provides a new and prospective mobile computing paradigm that allows the deployment of edge servers at base stations geographically close to mobile users to reduce their end-to-end network latency. Similar to cloud servers, edge servers running 24/7 in an MEC system consume a large amount of energy, contribute a significant proportion of global carbon emissions, and thus require demand response management. Demand response has been widely employed to reduce energy consumption at data centers. However, existing demand response approaches for data centers are rendered obsolete by the new and unique characteristics of MEC systems: 1) proximity constraint - mobile users can be served by neighbor edge servers only; 2) latency constraint - mobile users' workloads should be processed by their neighbor edge servers to ensure low latency; and 3) capacity constraint - edge servers have limited computing and communication resources to serve mobile users. Demand response for MEC is further complicated by the non-orthogonal multiple access (NOMA) scheme - the emerging radio access scheme for 5G. Communication resources like channels and transmit power in the NOMA-based MEC system must be systematically considered with computing resources like CPU, memory and storage to fulfill mobile users' resource demands. This paper makes the first attempt to tackle this Edge Demand Response (EDR) problem. We first formulate this problem and prove its $\mathcal{NP}$-hardness. Then, we propose a two-phase game-theoretical approach, named EDRGame, to solve the EDR problem. Its performance is theoretically analyzed and experimentally evaluated against three baseline approaches and two state-of-the-art approaches on a widely-used real-world dataset. The results show that it solves the EDR problem effectively and efficiently.

Index Terms—Demand response, mobile edge computing, energy consumption, game theory, potential game

1 Introduction

In THE last decade, many mega-size data centers have been built to power various mobile applications over the Internet. These data centers are major consumers of the world's electricity [1], consuming about 2% of electricity worldwide at the moment and 8% by 2030 [2]. Consequently, they contribute a significant proportion of the world's carbon emissions - about 0.3% in 2018 [1]. To reduce data centers' energy consumption, demand response has been widely investigated and implemented [3]. Many approaches have been proposed to strike a balance between data centers' workloads and smart grid's electricity supply [4].

- *Guangming Cui, Qiang He, and Yun Yang are with the Department of Computing Technologies, Swinburne University of Technology, Hawthorn, VIC 3122, Australia. E-mail: {gcui, qhe, yyang}@swin.edu.au.*
- *Xiaoyu Xia and Feifei Chen are with the School of Information Technology, Deakin University, Burwood, VIC 3125, Australia. E-mail: {xiaoyu.xia, feifei.chen}@deakin.edu.au.*
- *Tao Gu is with the Department of Computing, Macquarie University, Macquarie Park, NSW 2109, Australia. E-mail: tao.gu@mq.edu.au.*
- *Hai Jin is with the Services Computing Technology and System Lab, Big Data Technology and System Lab, Cluster and Grid Computing Lab, School of Computer Science and Technology, Huazhong University of Science and Technology, Wuhan, Hubei 430074, China. E-mail: hjin@hust.edu.cn.*

Manuscript received 27 Aug. 2020; revised 28 May 2021; accepted 20 Aug. 2021. Date of publication 30 Aug. 2021; date of current version 3 Feb. 2023. (Corresponding author: Qiang He.)
Digital Object Identifier no. 10.1109/TMC.2021.3108581

In recent years, mobile edge computing (MEC), i.e., a key technology that facilitates 5G networks, has emerged to extend mobile cloud computing (MCC) by pushing computing capacities to the edge of the mobile network [5]. In an MEC-enabled environment, edge servers are deployed at base stations that are geographically close to mobile users [6]. Mobile app vendors like YouTube and Uber can deploy their applications on edge servers to serve their users with low network latency. MEC offers many innovation opportunities and also raises a grand challenge - massive edge servers distributed all over the world with a density of up to 50 per km^2 in future 5G deployments [7] consume a large amount of electricity and further increase the energy consumption of the global IT infrastructure. To tackle this challenge, *edge demand response* (EDR) has been recently studied in MEC systems [8] and fog computing systems [9]. These approaches generally follow the same idea of data center demand response - transferring workloads across computation nodes to minimize the overall system energy consumption.

However, an MEC system has unique characteristics that have not been considered systematically in existing research on EDR, in particular, proximity constraint, latency constraint and capacity constraint [10], [11]. In general, edge servers in an MEC system are geographically distributed to cover different areas wirelessly. A mobile user can only be served by a *neighbor edge server* covering the mobile user. This *proximity*

constraint fundamentally differentiates MEC systems from MCC systems where a mobile user's workloads can be processed by any servers in the remote cloud. In addition, mobile users in an MEC system demand for low latency, known as the *latency constraint*. Transferring workloads across edge servers inevitably incurs extra latency which conflicts with MEC's pursuit of low latency. Thus, a mobile user's workloads must be processed by its neighbor edge servers to ensure low latency. Furthermore, unlike cloud servers, edge servers have limited computing resources due to their limited physical sizes [12]. An MEC system should ensure that the overall resource demand of mobile users allocated to an edge server must not exceed its capacity, known as *capacity constraint*. This constraint also applies to the communication resources in the MEC system, including channels and transmit power, in particular when Non-Orthogonal Multiple Access (NOMA) is enabled. The NOMA scheme has been acknowledged as a promising multiple access scheme for 5G [13]. Compared with conventional Orthogonal Multiple Access (OMA), NOMA can significantly improve spectral efficiency and provide massive user connectivity by allowing non-zero cross correlation signals [14]. In a NOMA-based MEC system, mobile users' data rates can be ensured through appropriate transmit power allocation based on their channel conditions [13]. However, compared with computation resources like CPU, memory and storage, the allocation of communication resources in a NOMA-based MEC system is much more complex and complicates the EDR problem significantly.

Typically, EDR is implemented when not all physical machines in an MEC system are needed to accommodate all the mobile users. It works as follows. In an MEC system, each edge server is normally powered by multiple physical machines [6]. While a minimum number of physical machines will be allocated to execute mobile users' workloads, idle machines can then be powered off to save energy. However, powering on/off machines frequently incurs considerable switch penalty such as start-up delay, system oscillation, and hardware wear-and-tear [8], which significantly reduces the energy-saving benefit. In addition, environmental transitions that trigger EDR, like a significant increase in the number of mobile users in an area, usually do not complete instantly. Thus, EDR is implemented periodically or on-demand by powering off idle machines in a period of hours or days. From the perspective of the edge infrastructure provider, e.g., Verizon and Amazon, the objectives of EDR in a NOMA-based MEC system are threefold: *Objective #1*) to serve the most mobile users[1] in the next period of time; *Objective #2*) minimize the system energy consumption; and *Objective #3*) maximize mobile users' overall data rate.

To achieve these objectives while fulfilling the three unique constraints in a NOMA-based MEC system, a typical EDR approach will go through two main phases. *Phase #1 Preparation*: based on the *estimated* maximum number of mobile users in the system and their distribution within each base station's coverage, it determines the physical machines needed to power each of the edge servers for serving the maximum number of mobile users with the minimum data rate and the minimum energy consumption. In this phase, the EDR approach first pursues Objective #1 to accommodate the most users' basic computing and networking resource demands. Then, it pursues Objective #2 to minimize the corresponding system energy incurred. *Phase #2 Operation*: *actual* mobile users are allocated to meet their demands of computing resources and maximize the overall data rate through channel and transmit power allocations based on NOMA. Objective #3 is pursued in this phase.

In real-world NOMA-based MEC systems, finding centralized optimal solutions to large-scale EDR problems may be intractable due to the complication in the EDR problem, especially in the allocation of communication resources. To tackle this challenge, this paper proposes EDRGame, a novel two-phase game-theoretical approach specifically-designed for formulating EDR strategies for NOMA-based MEC systems considering the impact of NOMA. EDRGame models the EDR problem as a game and makes allocation decisions for individual mobile users simultaneously in a decentralized manner. This allows EDR strategies to be formulated efficiently. The main contributions of this paper are summarized as follows.

- This is the first attempt to study the EDR problem in NOMA-based MEC systems, considering the unique characteristics of MEC and the impact of the NOMA scheme.
- We formulate the EDR problem as a constrained optimization problem and analyze its problem hardness.
- We propose an approach named EDRGame for solving the EDR problem based on game theory. Specifically, EDRGame models the EDR problem as a potential game and where a Nash equilibrium is proven achievable.Through its innovative two-phase design, EDRGame can solve the EDR problem effectively and efficiently.
- The performance of EDRGame is theoretically analyzed, and experimentally evaluated against three baseline approaches and two state-of-the-art approaches on a widely-used real-world dataset.

The remainder of the paper is organized as follows. First, the related work is reviewed in Section 2. Then, Section 3 provides an example to motivate the EDR problem. Next, Section 4 formulates the system model. Section 5 introduces EDRGame in detail and analyzes its convergence theoretically. Then, Section 6 evaluates its performance theoretically and experimentally. Section 7 summarizes the conclusions and our future work.

2 Related Work

Energy consumption has been a critical global matter in the past decade as it makes significant contributions to the world's greenhouse gas emissions [1]. Data centers are well-known as one of the world's main energy consumers [15]. Demand response, as a crucial way to save on energy

1. Unfortunately, it is not always possible to serve all the mobile users in the system. For example, when there are excessive mobile users around a base station that are not covered by any other base stations, even running all the physical machines at that base station may not be able to accommodate all those mobile users.

consumption, has been researched extensively to save on data centers' energy consumption. In general, existing demand response approaches for data centers can be partitioned into two groups. The first group of approaches mainly focuses on the supply side and pursues to exploit the differentiated (and sometimes dynamic) electricity supplies and prices across smart grids in different locations [16]. The second group of approaches focuses on the demand side and tries to leverage data centers' dynamic and flexible workloads [4].

The mobile edge computing (MEC) paradigm, widely recognized as an extension of mobile cloud computing, was proposed by Cisco in 2012 [17]. In an MEC system, computing resources are provisioned by edge servers deployed at 5G stations or access points that are geographically close to mobile users [18]. Computation tasks can be offloaded from energy-constrained and resource-limited mobile devices to edge servers [19]. This is referred to as computation offloading and has been investigated intensively in the past several years. In the meantime, app vendors such as YouTube and Uber can serve their users with low latency by deploying and running their applications on edge servers. Recently, from the app vendor's perspective, many new MEC problems have been identified and studied, such as, edge user allocation [10], [20], edge data caching [11], [21], edge application deployment [22], [23], edge data integrity [24], etc.

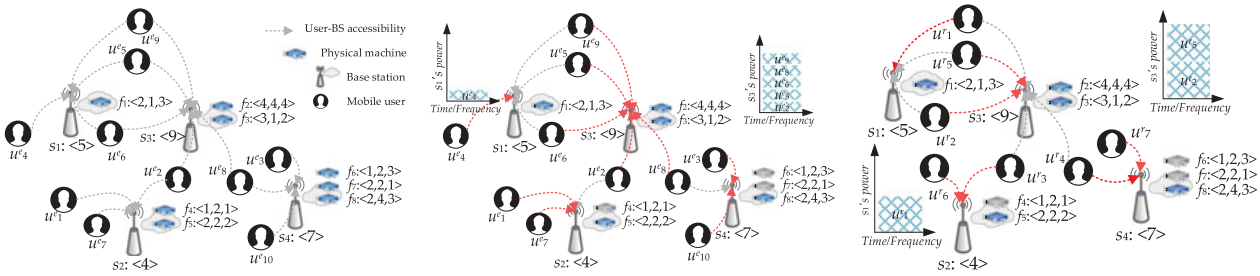
The unique advantages offered by MEC promote the deployment of edge servers around the globe as the rollout of the 5G network. The density of 5G base stations has been increasing rapidly and is expected to reach up to 50 base stations per km^2 in future 5G deployments [7]. Massive edge servers deployed around the globe will contribute significantly to the world's energy consumption and carbon emissions. This critical issue is starting to attract researchers' attention. A straightforward solution is to implement edge demand response (EDR) by powering off some of the physical machines in an MEC system that are not needed to serve mobile users. Very recently, the authors of [8] proposed an online auction mechanism to power on/off entire idle edge servers to reduce their energy consumption. However, an edge server is usually facilitated by a cluster of physical machines [6]. Powering on/off the entire edge servers immediately disconnect mobile users from edge servers if they cannot be served by any other edge servers. In addition, the assumptions made in [8] are not entirely realistic in real-world MEC systems. First, it was assumed that a mobile user can access all the edge servers in the system directly. Second, it was assumed that all the edge servers can communicate with each other directly. These assumptions oversimplify the MEC environment and practically turn it into a cloud-like environment. Thus, the approach proposed in [8] is impractical in real-world MEC systems. This approach is implemented as EDR-Ab and enhanced as EDR-Ae in our experiments. The other main limitation of EDR-Ab and EDR-Ae is the lack of consideration of the NOMA scheme on mobile users' data rates. It is a unique networking resource dimension that differentiates the MEC environment from the cloud computing environment and is starting to attract researchers' attention in studies of MEC in recent years [25], [26]. Some researchers investigated the problem of allocating maximum mobile users to minimum edge servers without considering system energy consumption and the impact of the NOMA scheme on mobile users' data rates [20], pursuing EDR *Objective #1* but not *Objective #2* or *Objective #3*. Their approach is implemented as EDR-H in our experiments.

This paper makes the first attempt to study the novel Edge Demand Response problem in NOMA-based MEC systems, aiming to serve the maximum mobile users with the maximum overall data rate at minimum system energy consumption, while fulfilling the unique constraints in real-world MEC systems, including proximity constraint and capacity constraint [10], [20]. Inspired by wide applications of game theory in studies of mobile cloud computing and mobile edge computing problems, e.g., computation offloading [19], edge user allocation [10], 5G health monitoring [27], edge user association and power allocation [28], etc., a game-theoretic approach named EDRGame is proposed in this paper to solve the EDR problem in a decentralized manner. Aiming to solve two seemingly similar problems based on game theory, our study differs from [28]. First, in [28], it is assumed that a user can be allocated to any of the base stations in the system. However, this is not realistic in most EDR scenarios that involve multiple base stations covering an area collectively. In real-world EDR scenarios, a user can only be allocated to an edge server if it is covered by the corresponding base station to which the edge server is attached. This *proximity constraint* is considered in our study, but not in [28]. Second, the design of EDRGame is different from the approaches proposed in [28], including the utility model and the algorithm for finding the Nash equilibrium in the game. Specifically designed to solve the EDR problem, EDRGame takes into account physical machines' energy consumption while the approaches proposed in [28] focuses only on users' overall data rate. Finally, EDRGame employs an innovative 2-phase design to tackle the EDR problem specifically. In Phase #1, EDRGame assumes maximum intra-cell interference and maximum inter-cell interference for individual users to ensure minimum data rate when actual users arrive in Phase #2. Then, in Phase #2, EDRGame allocates users and their transmit power based on their actual channel conditions and interference. The games proposed in [28] cannot be simply played twice to mimic EDRGame's 2 phases.

3 Motivation Example

Fig. 1 presents an MEC system with 4 edge servers deployed at 4 base stations (BSs), s_1, s_2, s_3 and s_4, facilitated by 1 (f_1), 2 (f_2 and f_3), 2 (f_4 and f_5) and 3 (f_6, f_7 and f_8) physical machines, respectively. Each physical machine has a set of unitized computing resources available, including bandwidth, CPU, memory and storage. For example, f_3 has 3 units of bandwidth, 1 unit of cpu, 2 units of memory and 4 units of storage available, denoted as $\langle 3, 1, 2, 4\rangle$.

In *Phase #1: Preparation*, it is expected that in the next period of time there will be a maximum of 11 mobile users in the system, denoted by $U^e = \{u_1^e, \ldots, u_{11}^e\}$, as shown in Fig. 1a. Each mobile user's resource demand is represented as $\langle 1, 1, 1, 1\rangle$, which is omitted in Fig. 1 for a clear presentation. Similar to [10], [11], [21], unitized computing resources used for a generic model can be easily replaced by specific



(a) Estimated Mobile Users (b) EDR Phase #1: Preparation (c) EDR Phase #2: Operation

Fig. 1. Example EDR Scenario in a NOMA-based MEC system. In this example, each base station is assumed to have one channel for ease of exposition. In real-world NOMA-based MEC systems, each edge server has multiple channels.

real-world resource models. From Fig. 1b, we can see that, to allocate the maximum mobile users at minimum system energy consumption, mobile users are "squeezed" onto the most powerful physical machines to minimize the number of physical machines needed and the system energy consumption.[2] As indicated by the red arrows in the figure, u_4^e is allocated to f_1, u_2^e, u_5^e, u_6^e and u_9^e are allocated to f_2, u_8^e is allocated to f_3, u_1^e and u_7^e are allocated to f_5, u_3^e and u_{10}^e are allocated to f_8. We can see that 5 physical machines, including f_1, f_2, f_3, f_5 and f_8, will suffice to accommodate all the estimated mobile users. The other 3 physical machines, i.e., f_4, f_6 and f_7, are powered off to save on system energy consumption. In this phase, the communication resources in the system, i.e., base stations' channels and transmit power, are allocated based on the NOMA scheme to ensure a minimum data rate for each individual mobile user. The allocation of s_1 and s_3's transmit power is illustrated in the top-left and top-right corners, respectively.

In *Phase #2: Operation*, 7 actual mobile users $U^r = \{u_1^r, \ldots, u_7^r\}$ arrive in the system, as shown in Fig. 1c. They need to be properly allocated to the physical machines running in the system so that their computing resource demands are fulfilled. In the meantime, the actual number of mobile users is no larger than the estimated number, i.e., $|U^e| \geq |U^r|$. Thus, the actual mobile users are given extra transmit power (compared with Phase #1) to maximize their overall data rate with minimum data rate ensured through proper channel allocation and transmit power allocation based on the NOMA scheme. The allocation of s_1 and s_3's transmit power is illustrated in the bottom-left and top-right corners of Fig. 1c, respectively. We can see that the actual mobile users are given more transmit power than the estimated mobile users in Fig. 1b.

The scales of real-world EDR problems can be much larger than this example. Optimal EDR strategies may be pursued in small-scale scenarios due to the NP-hardness of EDR problems. However, in large-scale EDR scenarios, an efficient approach is needed to formulate EDR strategies quickly.

4 System Model

In this study, we model the EDR problem as an EDR game. *Similar to many studies based on game theory [10], [29], players are simulated to make decisions for mobile users on the allocations of edge servers, subchannels and transmit power, to achieve the three EDR objectives introduced in Section 1.* This section first presents the models for formulating the EDR game. Then, it presents the optimization model for EDR and proves the $\mathcal{NP}$-hardness of the EDR problem.

Corresponding to EDR's two phases, the EDR game also goes through two phases, i.e., Phase #1 and Phase #2. The main difference is that in Phase #1, decisions are made for estimated mobile users (U^e) while in Phase #2, decisions are made for actual mobile users (U^r). In both phases, players make decisions following the same models built in this section. Thus, we employ $U = \{u_1, \ldots, u_n\}$ to refer to U^e and U^r in the NOMA-based MEC system in a unified way. Let us assume h base stations in the system, denoted by $S = \{s_1, \ldots, s_h\}$, powered by a total of m physical machines, denoted by $F = \{f_1, \ldots, f_m\}$. Each base station s_l is powered by a set of physical machines, denoted by $F(s_l)$, sharing s_l's K_l subchannels, denoted by $C_l = \{c_l^1, \ldots, c_l^{K_l}\}$. Each channel c_l^k (the kth channel on base station s_l) has transmit power p_l^k. There is $\sum_{l=1}^{h} |F(s_l)| = m$. Each physical machine $f_j \in F$ is equipped with a set of resources denoted by $D = \{\text{cpu}, \text{memory}, \text{storage}, \text{etc.}\}$ with capacities $\tau_j = \{\tau_j^1, \ldots, \tau_j^d, \ldots\}, d \in D$. Similarly, a set of resources is needed to serve each mobile user $u_i \in U$, i.e., $\omega_i = \{\omega_i^1, \ldots, \omega_i^d, \ldots\}, d \in D$. A table that summarizes the notations used in this paper is presented in Appendix A, which can be found on the Computer Society Digital Library at http://doi.ieeecomputersociety.org/10.1109/TMC.2021.3108581.

4.1 Energy Consumption Model

Similar to [8], [30], the energy consumption of a physical machine f_j ($j = 1, \ldots, m$) per unit of time is

$$e_j = e_j^s + e_j^r, \tag{1}$$

where e_j^s is the switching cost, i.e., the start-up energy cost of activating f_j and e_j^r is the running cost, i.e., the energy cost of running f_j.

As discussed in Section 1, an edge server is usually facilitated by a cluster of physical machines [6]. In this study, we assume that the physical machines incur the same switching cost. In addition, EDR does not power on/off physical machines frequently. This indicates $e_j^r \gg e_j^s$. For each mobile user, a dummy physical machine f_0, whose energy consumption is $e_0 \gg e_j, \forall f_j \in F$, is introduced as the default allocation decision. In this way, an EDR approach will

2. Please note that in this example, individual physical machines are assumed to consume the same amount of energy for ease of exposition. The more general cases where physical machines may consume different amounts of energy are studied in the rest of this paper.

pursue to replace f_0 for all the mobile users to lower the overall system energy consumption.

4.2 NOMA Model

In a NOMA-based MEC system, mobile users' overall data rate needs to be maximized (without exceeding the maximum achievable data rate under the Shannon's capacity constraint) through proper channel allocation and transmit power allocation. Over a 4G network, channel allocations are usually orthogonal - each subchannel can be assigned to one and only one mobile user at the same time [31]. Many mobile users may suffer from a significant delay due to the limited number of subchannels available on base stations. The NOMA scheme has been widely acknowledged as the key radio access scheme to facilitate the 5G network [32]. According to the NOMA scheme, a mobile user's data rate relies on its allocated transmit power.

Before we present the NOMA model employed in this study, let us first define the *EDR strategy*, denoted by $\mathbf{a}$.

Definition 1 (EDR Strategy). *Given a set of neighbor physical machines $N(u_i)$ which geographically cover mobile user u_i, i.e., $u_i \in coverage(f_j)$ for $\forall f_j \in N(u_i)$, let $a_i = \{(0,0,0,0)\} \bigcup \{(j,l,k,p_{i,l}^k) | f_j \in N(u_i), f_j \in F(s_l), c_l^k \in C_l\}$ be the allocation decision for u_i. There is $a_i = (j,l,k,p_{i,l}^k)$ if u_i is allocated to physical machine f_j through base station s_l on subchannel c_l^k with transmit power $p_{i,l}^k$; otherwise $a_i = (0,0,0,0)$. The allocation decisions for all the n mobile users in the system constitute the EDR strategy, denoted by $\mathbf{a} = \{a_1, \ldots, a_n\}$.*

Here, $a_i = (j,l,k,p_{i,l}^k)$ is possible only when f_j has adequate computing resources to accommodate u_i and edge server s_l's subchannel[3] c_l^k has adequate transmit power to accommodate u_i, i.e., $\sum_{u_t \in U: a_t = a_i} \omega_t^d \leq \tau_j^d, \quad \forall d \in D \ \& \ \sum_{u_t \in U: a_t = a_i} p_{t,l}^k \leq p_l^k$ (p_l^k is base station s_l's maximum transmit power on subchannel c_l^k), and $coverage(s_l)$ is the geographical coverage of the base station where base station s_l is deployed, and $F(s_l)$ is the set of physical machines attached to s_l. Let $\mathcal{U}_l^k(\mathbf{a}) = \{u_i \in U | a_i = (j,l,k,p_{i,l}^k)\}$ be the set of mobile users served by base station s_l on subchannel c_l^k, and $\mathcal{U}_l(\mathbf{a}) = \{\mathcal{U}_l^k(\mathbf{a})\}$, $c_l^k \in C_l$ be the set of mobile users served by base station s_l.

4.2.1 Signal Model

Based on NOMA, a superposition-coded signal x_l^k is broadcasted by a base station s_l to all the mobile users on its subchannel c_l^k simultaneously [33]. It is calculated as follows:

$$x_l^k = \sum_{u_i \in \mathcal{U}_l^k(\mathbf{a})} \sqrt{p_{i,l}^k} x_{i,l}^k, \tag{2}$$

where $x_{i,l}^k$ is the signal transmitted from s_l to u_i on subchannel c_l^k. NOMA allows simultaneous transmission of multiple users' signals [28]. The total transmit power p_l^k of base station s_l on subchannel c_l^k is shared among all the mobile users allocated to c_l^k

$$p_l^k \geq \sum_{u_i \in \mathcal{U}_l^k(\mathbf{a})} p_{i,l}^k. \tag{3}$$

For each u_i allocated to physical machine f_j at base station s_l on subchannel c_l^k ($a_i = (j,l,k,p_{i,l}^k)$), its received signal, denoted by $y_{i,l}^k$, can be calculated based on the intended signal, intra- and inter-cell interference, as well as other noise [34]

$$y_{i,l}^k = \underbrace{g_{i,l}^k x_l^k}_{\substack{\text{intended signal} \\ + \text{ intra-cell interference}}} + \underbrace{\sum_{s_t \in S \setminus \{s_l\}} g_{i,t}^k x_t^k}_{\text{inter-cell interference}} + \underbrace{z_{i,l}^k}_{\text{noise}}, \tag{4}$$

where $g_{i,l}^k$ denotes the subchannel gain between mobile user u_i and base station s_l on subchannel c_l^k, and $z_{i,l}^k$ is the additive white Gaussian noise with variance σ^2. Similar to [28], the subchannel coefficient is calculated as follows: $|g_{i,l}^k|^2 = |\hat{h}_{i,l}^k|^2 L(d_{i,l})$, where $\hat{h}_{i,l}^k \sim \mathcal{CN}(0,1)$ is the small-scale fading coefficient from u_i to s_l on subchannel c_l^k, $L(d_{i,l}) = \eta d_{i,l}^{-\alpha}$ is the large-scale path loss, η denotes the frequency dependent factor, and α is the path loss exponent.

4.2.2 Successive Interference Cancellation

NOMA implements the Successive Interference Cancellation (SIC) technique so that mobile users $\mathcal{U}_l^k(\mathbf{a})$ can decode the received superposed signal. Let us assume that $\mathcal{U}_l^k(\mathbf{a})$ are allocated to subchannel c_i^k. With SIC, mobile users with better subchannel conditions detect and remove the signals of mobile users with worse subchannel conditions, who treat the signals of mobile users with better subchannel conditions as noise [13]. In this section, without loss of generality, all the mobile users in $\mathcal{U}_l^k(\mathbf{a})$ are ordered by their subchannel conditions: $u_1, u_2, \ldots, u_{|\mathcal{U}_l^k(\mathbf{a})|}$, where u_1 has the worst subchannel condition and $u_{|\mathcal{U}_l^k(\mathbf{a})|}$ has the best subchannel condition. SIC is not required for u_1 since it is the first to decode signal. Similar to [33], u_1 first decodes $x_{1,l}^k$ and subtracts its components from $y_{1,l}^k$. Then, u_2 can decode its received signal. Based on this principle, the signal-to-interference-plus-noise ratio (SINR) for $u_i \in \mathcal{U}_l^k(\mathbf{a})$ is

$$\gamma_{i,l}^k = \frac{|g_{i,l}^k|^2 p_{i,l}^k}{|g_{i,l}^k|^2 \sum_{q=i+1}^{|\mathcal{U}_l^k(\mathbf{a})|} p_{i,q}^k + I_{i,l}^k + \sigma^2}, \tag{5}$$

where $I_{i,l}^k = \sum_{s_t \in S \setminus \{s_l\}} |g_{i,t}^k|^2 P_t^k$ is the inter-cell interference caused by u_i's nearby base stations. Given Eq. (5), the SINR for the last user to decode the received signal, $u_{|\mathcal{U}_l^k(\mathbf{a})|}$, is: $\gamma_{|\mathcal{U}_l^k(\mathbf{a})|,j}^k = (|g_{|\mathcal{U}_l^k(\mathbf{a})|,l}^k|^2 p_{|\mathcal{U}_l^k(\mathbf{a})|,l}^k)/(I_{|\mathcal{U}_j^k(\mathbf{a})|,l}^k + \sigma^2)$.

Given two mobile users $u_i, u_q \in \mathcal{U}_l^k(\mathbf{a})$ that u_q has a better subchannel condition than u_i ($i < q$). User u_q's data rate must not be lower than u_i's data rate [28]: $r_{q \to i,l}^k \geq r_{i \to i,l}^k$. In this way, u_i's data rate $r_{i,l}^k$ on subchannel c_l^k can be expressed by

$$r_{i,l}^k = \min\{r_{q \to i,l}^k | \forall q \geq i\}, \tag{6}$$

where $r_{q \to i,l}^k$ is u_q's data rate for decoding u_i's signal, calculated as follows:

3. Please note that in this paper we often refer to an edge server's communication resources instead of the corresponding base station's. For example, edge server s_l's subchannel c_l^k is in fact the kth subchannel of the bases station where s_l is deployed.

$$r^k_{q\to i,l} = B^k_l \log_2\left(1 + \frac{|g^k_{q,l}|^2 p^k_{i,l}}{|g^k_{q,l}|^2 \sum_{t=i+1}^{|\mathcal{U}^k_l(\mathbf{a})|} p^k_{t,l} + I^k_{q,l} + \sigma^2}\right). \quad (7)$$

It can be easily inferred that u_i's data rate is no more than those of the users after u_i.

SIC Decoding Order. According to the above discussion about SIC, mobile users' decoding order plays an important role in their overall data rate. As can be seen from Eqs. (6) and (7), the data rate is partially determined by the subchannel coefficient and inter-cell interference. Eq. (6) can be transformed into

$$r^k_{i,l} = B^k_l \log_2\left(1 + \frac{p^k_{i,l}}{\sum_{t=i+1}^{|\mathcal{U}^k_l(\mathbf{a})|} p^k_{t,l} + G^k_{i,l}}\right), \quad (8)$$

where

$$G^k_{i,l} = \max\left\{\frac{I^k_{q,l} + \sigma^2}{|g^k_{q,l}|^2}\middle|\forall q \geq i\right\}. \quad (9)$$

Similar to many studies [35], [36], [37] in MEC, under the NOMA scheme, a mobile user's maximum data rate is Shannon's limit for decoding the desired signals after canceling the signals of other users based on the optimal decoding order. Let r_{max} denote this maximum achievable data rate. The range of achievable data rate is $r^k_{i,l} \in (r_{min}, r_{max})$, where r_{min} is the minimum data rate required for serving mobile users.

This study aims to solve the EDR problem in the real-world MEC environment set up based on the NOMA scheme, considering its impact on the EDR problem. NOMA is not a simple extra dimension orthogonal to the computation dimension in the EDR problem. These two dimensions must be considered jointly and systematically. For example, a user cannot be simply allocated to a base station to maximize its data rate without considering the computation dimension because allocating the user to an edge server attached to that base station may not be energy-efficient or feasible due to edge servers' constrained computing resources. In our study, a classic NOMA scheme that has been widely employed in studies of MEC, e.g., [26], [38], is considered in the study to facilitate a generic EDR approach in NOMA-based MEC environments. Please note that NOMA is still an open and active research topic. New features of more sophisticated NOMA schemes, e.g., the power allocation and control techniques proposed in [39], can be integrated into the proposed approach without violating its correctness or performance.

4.3 Decision Benefit Model

To achieve the three EDR objectives introduced in Section 1, in Phase #1 and Phase #2 of the EDR game, decisions are made for individual mobile users, with the aim to maximize their benefits calculated with Eq. (10)

$$B_{\mathbf{a}_{-i}}(a_i) = \begin{cases} \frac{|\mathcal{UF}_j(\mathbf{a})| \cdot r^k_{i,l}}{e_j}, & a_i \neq (0,0,0,0) \\ 0, & a_i = (0,0,0,0) \end{cases}, \quad (10)$$

where $\mathbf{a}_{-i} = (a_1, \ldots, a_{i-1}, a_{i+1}, \ldots, a_n)$ represents the set of decisions for all the mobile users except u_i and $\mathcal{UF}_j(\mathbf{a})$ is the set of mobile users allocated to physical machine f_j by $\mathbf{a}$. Decisions made for individual mobile users to maximize Eq. (10) will allocate them to the most energy-efficient physical machines, i.e., those that can accommodate the most mobile users with the least energy consumption.

As discussed in Section 3, the minimum data rate r_{min} is ensured and pursued for every individual mobile user in Phase #1. Thus, $r^k_{i,l}$ is a constant. The pursuit of maximized $B_{\mathbf{a}_{-i}}(a_i)$ in Phase #1 will allocate mobile users to physical machines that currently serve the maximum mobile users with minimum energy consumption on average. In Phase #2, $r^k_{i,l}$ is a variable because as discussed in Section 3. Extra transmit power will be allocated to mobile users to maximize their overall data rate. Compared with Phase #1, the pursuit of maximized $B_{\mathbf{a}_{-i}}(a_i)$ in Phase #2 will include the overall data rate into consideration.

4.4 Problem Hardness and Optimization Model

The constrained optimization problem (COP) model for an EDR problem involves a vector and five matrices. These matrices include: 1) a matrix of variable $X_{n\times m}$, with a domain $\{0,1\}$ such that $x_{i,j} \in \{0,1\}$ for $x_{i,j} \in X_{n\times m}$, representing the allocation decisions for individual mobile users. $x_{i,j} = 1$ indicates that u_i is served by physical machine f_j, otherwise $x_{i,j} = 0$; 2) a matrix $\mathcal{N}_{n,m}$ with domain $\{0,1\}$ indicating whether physical machine $f_j \in F$ covers each $u_i \in U$, where $\mathcal{N}_{i,j} = 1$ iff $f_j \in N(u_i)$, otherwise $\mathcal{N}_{i,j} = 0$; 3) a matrix of the resource needs for serving mobile users, $\Omega_{n\times d} = (\omega^k_i)_{u_i\in U, k\in D}$; 4) a matrix of physical machines' available computing resources $\tau_{m\times d} = (\tau^k_j)_{f_j\in F, k\in D}$; 5) a vector $E = \{e_1, \ldots, e_m\}$ that represents physical machines' energy consumption per unit of time; and 6) a matrix of transmit power allocation decisions $\mathcal{P}_{m\times n\times K_l} = (p^k_{i,l})_{u_i\in U, f_j\in F, f_j\in F(s_l), c^k_l\in c_l}$.

Now, *Objective #1* can be represented as

$$\max \sum\nolimits_{u_i\in U}\left(\min\left\{\sum\nolimits_{f_j\in F} x_{i,j}, 1\right\}\right), \quad (11)$$

where $\sum_{f_j\in F} x_{i,j}$ is the allocation for mobile user u_i.

Objective #2 can be represented as follows:

$$\min \sum\nolimits_{f_j\in F}\left(e_j \cdot \min\left\{\sum\nolimits_{u_i\in U} x_{i,j}, 1\right\}\right). \quad (12)$$

Objective #3 can be represented as follows:

$$\max \sum_{u_i\in U} r^k_{i,l}, \quad (13)$$

where $r^k_{i,l}$ is u_i's data rate calculated by Eq. (8).

An EDR strategy must also fulfil the constraints below:

$$\sum\nolimits_{f_j\in F} x_{i,j} \leq 1, \text{ for } u_i \in U \quad (14)$$

$$\sum\nolimits_{u_i\in U} \omega^k_i \cdot x_{i,j} \leq \tau^k_j, \text{for } f_j \in F \quad (15)$$

$$x_{i,j} \in \{0\} \bigcup \{\mathcal{N}_{i,j}\}, \text{for } u_i \in U, f_j \in F \quad (16)$$

$$r_{min} < r^k_{i,l}, \text{for } x_{i,j} = 1. \quad (17)$$

In this COP model, (14) is the constraint ensuring that each mobile user can be served by at most one physical machine, (15) is the constraint ensuring that the total resources needed on a physical machine f_j to serve the mobile users allocated to f_j must not exceed its resource capacity, and (16) is the constraint ensuring that a mobile user can only be allocated to one of its neighbor physical machines. In addition, (17) ensures the minimum data rate for all allocated users. The above COP can be solved with a Mixed Integer Programming problem solver, e.g., IBM CPLEX Optimizer[4] and Gurobi.[5] The optimal solution to this COP is the EDR strategy that achieves Objectives (11), (12) and then (13) while fulfilling Constraints (14), (15), (16) and (17). This approach is referred to as EDR-Opt.

Based on this COP model, the hardness of the EDR problem is analyzed in Theorem 1 and the proof if presented in Appendix B, available in the online supplemental material.

Theorem 1. *The EDR problem is $\mathcal{NP}$-hard, since it is reducible from the $\mathcal{NP}$-hard bin packing problem.*

5 Game Formulation and Mechanism Design

When the problem scales up, finding optimal EDR solutions may be intractable due to its $\mathcal{NP}$-hardness. To tackle this challenge, this section presents EDRGame, our game-theoretical approach for solving large-scale EDR problems.

5.1 Game Formulation

Players are simulated in Phase #1 and Phase #2 of the EDR game to make decisions for corresponding mobile users, using benefit function (10), to achieve the three EDR objectives. Specifically, decisions are made on which subchannels of which edge servers mobile users are allocated to, i.e., $a_i \in \{(0,0,0,0)\} \bigcup \{(j,l,k,p_{i,l}^k) | \forall u_i \in cov(s_l), c_j^k \in C_l$.

As discussed in Section 4.3, in Phase #1, given a set of estimated mobile users U^e, the minimum data rate r_{min} is ensured for each mobile user $u_i^e \in U^e$. Based on Eq. (8), the transmit power $p_{i,min}$ for ensuring r_{min} is

$$p_{i,min} = 2^{(\frac{r_{min}}{B_l^k}-1)\cdot(p_l^k+G_{i,l}^k)}, \tag{18}$$

where p_l^k and $G_{i,l}^k$ are used to enforce the maximum intra-cell interference and maximum inter-cell interference, respectively, on user u_i.

In Phase #2, given a set of actual mobile users U^r ($|U^r| < |U^e|$), the base stations in the system usually have extra transmit power to increase their overall data rate with each individual mobile user's data rate maximized within the range of $[p_{i,min}, p_{i,max}]$ without exceeding the total transmit power p_l^k on each subchannel of each base station $s_l \in S$ in the system. Here, $p_{i,max}$ is calculated as follows:

$$p_{i,max} = 2^{(\frac{r_{max}}{B_l^k}-1)\cdot(p_l^k+G_{i,l}^k)}, \tag{19}$$

where r_{max} is the upper bound on mobile users' data rate in the NOMA-based MEC system, as discussed in Section 4.2.

4. https://www.ibm.com/analytics/cplex-optimizer
5. http://www.gurobi.com/

In each iteration of this two-phase EDR game, given $\mathbf{a}_{-i}$, i.e., the allocation decisions for other mobile users, an allocation decision a_i is made for u_i to maximize its benefit

$$\max_{a_i \in \{(0,0,0,0)\} \bigcup \{(j,l,k,p_{i,l}^k) | f_j \in N(u_i), f_j \in F(s_l), c_l^k \in C_j\}} B_{\mathbf{a}_{-i}}(a_i). \tag{20}$$

Base on (20), the EDR game $\chi = (U, \{\mathcal{A}_i\}_{u_i \in U}, \{\mathcal{B}_i\}_{u_i \in U})$ involves a set of mobile users U, with a finite set of channel allocation decisions for each u_i's, denoted as $\mathcal{A}_i$, and the its benefit function $\mathcal{B}_i$. The proximity constraint, capacity constraint and potential interference often cause conflicts among mobile users. To mitigate these conflicts, EDRGame aims to find the Nash equilibrium in the EDR game:

Definition 2 (Nash Equilibrium). *A Nash equilibrium in an EDR game is an EDR strategy, denoted by $\mathbf{a}^* = (a_1^*, a_2^*, \ldots, a_n^*)$, ensuring that no decisions can be updated for individual mobile users unilaterally to increase their own benefits, i.e.,*

$$B_{\mathbf{a}_{-i}^*}(a_i^*) \geq B_{\mathbf{a}_{-i}^*}(a_i), \forall a_i \in \mathcal{A}_i, u_i \in U. \tag{21}$$

Any allocation decision a_i^* in a Nash equilibrium $\mathbf{a}^*$ is the best decision for u_i in $\mathcal{A}_i$ in response to $\mathbf{a}_{-i}$. Thus, any Nash equilibrium in an EDR game is a self-enforcing EDR strategy that allows individual mobile users to achieve their own interests by following the strategy together [10], [29].

5.2 Game Property

As mentioned above, the existence of a Nash equilibrium is critical to the EDR game formulated for an EDR problem. In this section, we analyze the existence of a Nash equilibrium in the EDR game by proving that it is a potential game [29], [40].

Definition 3 (Potential EDR Game). *The EDR game is a potential game if a potential function $\phi(\mathbf{a})$ can be found that fulfills*

$$\begin{aligned} B_{\mathbf{a}_{-i}}(a_i) < B_{\mathbf{a}_{-i}}(a_i') \Rightarrow \phi_{\mathbf{a}_{-i}}(a_i) < \phi_{\mathbf{a}_{-i}}(a_i'), \\ \forall u_i \in U, a_i, a_i' \in \mathcal{A}_i, \mathbf{a}_{-i} \in \prod_{l \neq i} \mathcal{A}_l. \end{aligned} \tag{22}$$

To prove that EDR game is a potential game, Lemma 1 introduces an important property of the EDR game.

Lemma 1 (Allocation Constraint). *For any allocation strategy $\mathbf{a} = \{a_1, \ldots, a_n\}$, u_i can be served by c_l^k if its received interference $\mu_{i,l}^k(\mathbf{a}) \triangleq \sum_{t=i+1}^{|\mathcal{U}_l^k(\mathbf{a})|} p_{t,l}^k \leq T_i$, where*

$$T_i = \frac{p_{i,l}^k}{2^{\frac{r_{min}}{B_l^k}} - 1} - G_{i,l}^k. \tag{23}$$

The proof of Lemma 1 is presented in Appendix C, available in the online supplemental material.

Based on Lemma 1, when u_i's $\mu_{i,l}^k(\mathbf{a})$ is adequately low, u_i can benefit from being allocated to physical machine f_j on base station s_l's subchannel c_l^k. Otherwise, u_i is not allocated to any physical machines. Based on Lemma 1, we can prove that the EDR game is a potential game with Theorem 2.

Theorem 2 (Potential EDR Game). *The EDR game is a potential game with (24) as the potential function*

$$\phi_{\mathbf{a}_{-i}}(a_i) = -\frac{1}{2}\sum_{u_i \in U}\sum_{t=i+1}^{|\mathcal{U}_l^k(\mathbf{a})|} |\mathcal{UF}_j(\mathbf{a})|^2 p_{i,l}^k p_{t,l}^k I_{\{a_i=a_t\}} I_{\{a_i \neq (0,0,0,0)\}} - \sum_{u_i \in U} |\mathcal{UF}_{j,max}|^2 p_{i,l}^k T_i I_{\{a_i=(0,0,0,0)\}}. \quad (24)$$

The proof of Theorem 2 is presented in Appendix D, available in the online supplemental material.

5.3 Algorithm Design

EDRGame employs Algorithm 1 in Phase #1 and Algorithm 2 in Phase #2 to find the Nash equilibrium in the EDR game.

Algorithm 1. Phase #1 in EDRGame

Input: $U^e = \{u_1^e, \ldots, u_{n^e}^e\}$, $F = \{f_1, \ldots, f_m\}$ and $S = \{s_1, \ldots, s_h\}$
Output: $\mathbf{a}^e$

1: Initialize EDR $\mathbf{a}^e = \{(0,0,0,0), \ldots (0,0,0,0)\}$
2: **repeat**
3: **for all** $u_i^e \in U^e$ **do**
4: calculate current benefit $B_{\mathbf{a}_{-i}}(a_i^e)$
5: create $\mathcal{A}_i^e \leftarrow \varnothing$
6: **for all** $f_j \in N(u_i^e), f_j \in F(s_l)$ **do**
7: **if** f_j has adequate resources **or** u_i^e is allocated to f_j **then**
8: **for all** $c_l^k \in c_l$ **do**
9: calculate the remaining transmit power on c_l^k: $\Delta p_i^e = p_l^k - \sum_{u_i^e \in \mathcal{U}_l^k(\mathbf{a}^e)} p_{t,min}$
10: **if** $\Delta p_i^e \geq p_{i,min}$ **then**
11: $\mathcal{A}_i^e = \mathcal{A}_i^e \bigcup \{(j,l,k,p_{i,min})\}$
12: **end if**
13: **end for**
14: **end if**
15: **end for**
16: find $a_i^{e'} \in \mathcal{A}_i$ that produces the highest benefit
17: **if** $B_{\mathbf{a}_{-i}^e}(a_i^e) < B_{\mathbf{a}_{-i}^e}(a_i^{e'})$ **then**
18: send $a_i^{e'}$ to request decision update
19: **if** win the opportunity **then**
20: update a_i^e with $a_i^{e'}$
21: **end if**
22: **end if**
23: **end for**
24: **until** no decision updates needed
25: **return** $\mathbf{a}^e$

In Phase #1, starting with the initialization of $\mathbf{a}^e$ (Line 1), Algorithm 1 determines the physical machines to be running in the next period of time based on the estimated maximum number of mobile users in the system and their distribution. Next, for each mobile user $u_i^e \in U^e$, it calculates the current benefit produced by $\mathbf{a}^e$ with Eq. (10) (Lines 4). After that, it attempts to find out all the possible allocation decisions for u_i^e to be stored in $\mathcal{A}_i^e$ (created on Line 5). To do that, it uses a loop (Lines 6-16) to inspect every physical machine $f_j \in N(u_i^e)$. Then, if any of the subchannels of the physical machines has adequate transmit power to accommodate u_i^e, the corresponding allocation decisions are included in $\mathcal{A}_i^e$ for u_i^e (Lines 10-12). Among all the allocation decisions in $\mathcal{A}_i^e$, the one $a_i^{e'} \in \mathcal{A}_i^e$ that produces the highest benefit is sent to request an update if it produces a higher benefit than the current allocation decision for u_i^e (Lines 17-24).

In each iteration, one submitted decision is randomly selected to be updated in a centralized manner [29] or a decentralized manner [40]. In either way, Algorithm 1 can be executed to make decisions for individual mobile users in parallel. The EDR game iterates until there are no more requests for allocation decision updates. Finally, $\mathbf{a}^e$ is returned by Algorithm 2 (Line 25).

Algorithm 2. Phase #2 in EDRGame

Input: $\mathbf{a}^e$ from Algorithm 1, $U^r = \{u_1^r, \ldots, u_{n^r}^r\}$, $F = \{f_1, \ldots, f_m\}$ and $S = \{s_1, \ldots, s_h\}$
Output: $\mathbf{a}^r$

1: initialize $\mathbf{a}^r \subset \mathbf{a}^e$
2: **repeat**
3: **for all** $u_i^r \in U^r$ **do**
4: calculate current benefit $B_{\mathbf{a}_{-i}^r}(a_i^r)$
5: create $\mathcal{A}_i^r \leftarrow \varnothing$
6: **for all** $f_j \in N(u_i^r), f_j \in F(s_l)$ & f_j is running based on $\mathbf{a}^e$ **do**
7: **if** u_i^r is allocated on f_j **or** f_j has adequate resources **then**
8: **for all** $c_l^k \in c_l$ **do**
9: calculate the remaining transmit power on c_l^k: $\Delta p_i^r = p_l^k - \sum_{u_i^r \in \mathcal{U}_l^k(\mathbf{a}^r)} p_{t,min}$
10: **if** $\Delta p_i^r \geq p_{i,min}$ **then**
11: $p_{i,l}^k = \min\{p_{i,max}, \Delta p_i^r\}$
12: $\mathcal{A}_i^r = \mathcal{A}_i^r \bigcup \{(j,l,k,p_{i,l}^k)\}$
13: **end if**
14: **end for**
15: **end if**
16: **end for**
17: find the decision $a_i^{r'} \in \mathcal{A}_i^r$ that produces the highest benefit
18: **if** $B_{\mathbf{a}_{-i}^r}(a_i^r) < B_{\mathbf{a}_{-i}^r}(a_i^{r'})$ **then**
19: send $a_i^{r'}$ to request decision update
20: **if** wins the opportunity **then**
21: update decision a_i^r with $a_i^{r'}$
22: **end if**
23: **end if**
24: **end for**
25: **until** no more decision updates needed
26: **return** $\mathbf{a}^r$

In Phase #2, given a set of actual mobile users $U^r = \{u_1^r, \ldots, U_{n^r}^r\}$ and EDR strategy $\mathbf{a}^e$ provided by Algorithm 1, Algorithm 2 initializes a new EDR strategy $\mathbf{a}^r \subset \mathbf{a}^e$ by assigning an allocation decision from $\mathbf{a}^e$ to each $u_i^r \in U^r$ based on their location (Line 1). Similar to Algorithm 1, Algorithm 2 is executed for each individual mobile user $u_i^r \in U^r$, starting with calculating the benefit produced by the current allocation decision in $\mathbf{a}^r$ for u_i^r (Line 4). Then, it iterates through u_i^r's running neighbor physical machines f_j and the corresponding base station s_l to find all the subchannels that can accommodate u_i^r with higher transmit power within the range of $[p_{i,min}, \min\{p_{i,max}, \Delta p_i^k\}]$ (Lines 6 - 16), and includes all the possible allocation decisions into $\mathcal{A}_i^r$ (created on Line 5). Next, if the optimal allocation decision $a_i^{r'}$ in $\mathcal{A}_i^r$ (found on Line 17) produces a higher benefit

than the current decision for u_i^r, it will be sent to request an update (Lines 17-23). Phase #2 completes when no more decision updates are needed for any mobile users (Line 25). Finally, the final EDR strategy $\mathbf{a}^r$ is returned as the solution to the EDR problem (Line 26).

5.4 Convergence Analysis

As a potential game, an EDR game can reach a Nash equilibrium after a finite number of iterations [10], [41]. Let V denote the total number of iterations, $Q_i \triangleq |\mathcal{UF}_j(\mathbf{a})|p_{i,l}^k$, $Q_{min} \triangleq \min(Q_i)$, $Q_{max} \triangleq \max(Q_i)$, $T_{min} \triangleq \min(T_i)$, $T_{max} \triangleq \max(T_i)$ ($i = 1, \ldots, n$, $j = 1, \ldots, m$ and $k = 1, \ldots, K_l$), Theorem 3 theoretically analyzes EDRGame's convergence time measured by the maximum number of iterations need to reach a Nash equilibrium.

Theorem 3 (Upper Bound on EDRGame's Convergence Time). *Given two non-negative integers T_i and Q_i, the maximum number of iterations of EDRGame is $n^2Q_{max}^2/2Q_{min} + nQ_{max} \cdot T_{min}/Q_{min}$, that is, $V \leq n^2Q_{max}^2/2Q_{min} + nQ_{max} \cdot T_{min}/Q_{min}$.*

The proof of Theorem 3 is presented in Appendix E, available in the online supplemental material. This theorem shows that EDRGame can reach a Nash equilibrium within a quadratic time for non-negative integers Q_i and T_i. When Q_i and T_i are real numbers, EDRGame's convergence time is evaluated experimentally in Section 6.2.

According to Theorem 3, Algorithms 1 and 2 iterate at most V times. In each iteration, each mobile user can be allocated to one of the K subchannels on one of the (at most) m physical machines, where $K = \max K_l, l = 1, \ldots, h$. Thus, Algorithms 1 and 2 take $O(mVK)$ time to reach a Nash equilibrium in an EDR game.

6 Performance Evaluation

EDRGame is a decentralized approach that finds sub-optimal solutions to EDR problems. Its optimization performance is critical to its practicality. This section evaluates EDRGame's performance theoretically and experimentally.

6.1 Theoretical Analysis

As shown in Algorithms 1 and 2, allocation decisions are randomly updated over the course of the game. This creates the possibility of multiple Nash equilibria in an EDR game. To measure the performance gap between EDRGame and EDR-Opt, this section analyzes EDRGame's Price of Anarchy (POA) [10], [40], measured by the ratios of the number of allocated mobile users, the system energy consumption and users' overall data rate achieved by EDRGame's worst Nash equilibrium and those achieved by EDR-Opt. First, we prove Lemma 2 to facilitate the POA analysis of EDRGame.

Lemma 2 (Number of Allocated Mobile Users). *For any Nash equilibrium $\mathbf{a}$, the number of allocated mobile users $num(\mathbf{a})$ fulfills:*

$$\lfloor T_{min}/Q_{max} \rfloor \leq num(\mathbf{a}) \leq \lfloor T_{max}/Q_{min} \rfloor + 1. \tag{25}$$

The proof of Lemma 2 is presented in Appendix F, available in the online supplemental material.

Let χ repressent the set of different Nash equilibria in the EDR game and $\mathbf{a}^* = (a_1^*, a_2^*, \ldots, a_n^*)$ represent the optimal EDR strategy. Based on Lemma 2, Theorem 4 analyzes $\rho_u(\mathbf{a})$, i.e., EDRGame's POA in terms of number of allocated mobile users.

Theorem 4 (POA in Number of Allocated Mobile Users). *Given any EDR strategy $\mathbf{a} \in \chi$ and $\mathbf{a}^*$, $\rho_u(\mathbf{a})$ fulfills*

$$1 \geq \rho_u(\mathbf{a}) \geq \lfloor T_{min}/Q_{max} \rfloor / (\lfloor T_{max}/Q_{min} \rfloor + 1). \tag{26}$$

The proof of Theorem 4 is presented in Appendix G, available in the online supplemental material.

Theorem 5 analyzes $\rho_e(\mathbf{a})$, i.e., EDRGame's POA in terms of system energy consumption.

Theorem 5 (POA in System Energy Consumption). *Given an EDR strategy $\mathbf{a} \in \chi$ and $\mathbf{a}^*$, $\rho_e(\mathbf{a})$ fulfills*

$$1 \leq \rho_e(\mathbf{a}) \leq (\mathcal{UF}_{max} \lfloor T_{min}/Q_{max} \rfloor) / (\lfloor T_{max}/Q_{min} \rfloor + 1), \tag{27}$$

where $\mathcal{UF}_{max}$ is the maximum number of mobile users that can be served by any physical machine in the system.

The proof of Theorem 5 is presented in Appendix H, available in the online supplemental material.

Theorem 6 analyzes $\rho_d(\mathbf{a})$, i.e., EDRGame's POA in terms of mobile users' overall data rate.

Theorem 6 (POA in Overall Data Rate). *Given an EDR strategy $\mathbf{a} \in \chi$ and $\mathbf{a}^*$, $\rho_d(\mathbf{a})$ fulfills*

$$\frac{r_{min}(\lfloor T_{min}/Q_{max} \rfloor)}{r_{max}(\lfloor T_{max}/Q_{min} \rfloor + 1)} \leq \rho_d(\mathbf{a}) \leq 1. \tag{28}$$

The proof of Theorem 6 is presented in Appendix I, available in the online supplemental material.

6.2 Experimental Evaluation

The performance of EDRGame is evaluated on a set of small-scale experiments (Set #1) and a set of large-scale ones (Set #2), both conducted on a Windows machine equipped with an Intel Core i5-7400T processor (4 CPUs, 2.4GHz) and 8GB RAM.

Dataset. The experiments are conducted on the widely-used real-world EUA dataset.[6] This dataset contains the locations of base stations and mobile users within Metropolitan Melbourne, Australia, covering a total area of over 9,000 km^2.

Experimental Settings. In general, EDR aims to accommodate mobile users' resource demands with minimum system energy consumption. The performance of EDRGame may vary in different EDR scenarios, depending on the difficulty in accommodating mobile users' resource demands. This is impacted by the number of mobile users to allocate, the number of physical machines available for selection and the capacities of these physical machines. Thus, to comprehensively evaluate EDRGame, we simulate various EDR scenarios: 1) the number of mobile users (n); 2) the number of physical machines (m); and 3) the average computing resources available on physical machines (τ). *Please note that*

6. https://github.com/swinedge/eua-dataset

TABLE 1
Experimental Settings

		n	m	τ
Set #1	Set #1.1	$1, 2, \ldots, 8$	4	4
	Set #1.2	4	$1, 2, \ldots, 8$	4
	Set #1.3	4	4	$1, 2, \ldots, 8$
Set #2	Set #2.1	$2^4, 2^5, \ldots, 2^{11}$	200	40
	Set #2.2	2^7	$50, 100, \ldots, 400$	40
	Set #2.3	2^7	200	$10, 20, \ldots, 80$

τ is adequately large to simulate typical EDR scenarios where most, if not always all, mobile users can be served by the physical machines. Each base station is assigned 1-5 physical machines randomly. Thus, the increase in m will increase the number of base stations in general. The experiment settings are summarized in Table 1. Each experiment is repeated 100 times and the results are averaged. Following the same settings as in [8], the idle and peak power of a physical machine are 100W and 300W, respectively. To differentiate physical machines' energy consumption, their running power is set between 100W to 300W, following a normal distribution, also similar to the settings in [8]. To simulate the high data rate offered by the 5G network, base stations are simulated in a way similar to [29], [40], with $|c_l| = 5, \forall s_l \in S$, channel bandwidth $B_j^k = 150$MHz, background noise $\sigma^2 = -100$ dBm, $\alpha = 5$ used to calculate $g_{i,l}^k$ in Eq. (4).

Performance Metrics. The effectiveness of EDRGame is measured by three performance metrics (two for Phase #1 and one for Phase #2): 1) number of allocated mobile users (*Objective #1*); 2) system energy consumption (*Objective #2*); and 3) overall data rate (*Objective #3*). In addition, to evaluate EDRGame's performance in minimizing service latency, which consists of computation latency and communication latency, we also measure allocated users' average service latency under similar task settings as [29]. Specifically, the data size of a task ranges from 500KB to 5,000KB, requiring 100 to 1,000 megacycles to process. The computing capability obtained by each allocated user is 10GHz, i.e., 10,000 megacycles per second.

Comparison. We compare EDRGame with three baseline approach and two state-of-the-art approaches. Since these approaches do not consider the impact of NOMA on users' data rates, in our experiments, they allocate transmit power $p_{mid} = (p_{min} + p_{max})/2$ to each allocated mobile user[7] to achieve *Objective #3*, where p_{min} and p_{max} are calculated as follows:

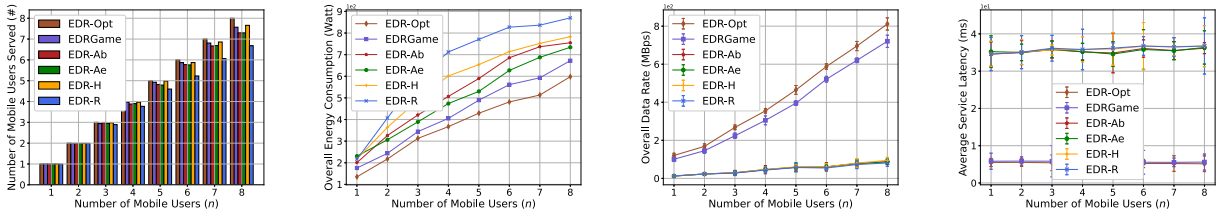
$$\begin{cases} p_{min} = \min_{u_i \in U} p_{i,min} \\ p_{max} = \max_{u_i \in U} p_{i,max} \end{cases}. \tag{29}$$

1) EDR-Opt: This approach employs IBM's CPLEX Optimizer to find the optimal solution to the EDR problem stated in Section 4.4 and allocates mobile users to physical machines based on the optimal solution. This is the first baseline approach which is only applicable to small scale.
2) EDR-R: This approach randomly allocates mobile users to their neighbor physical machines with adequate computing resources without considering system energy consumption. It is the second baseline approach.
3) EDR-H [42]: This is a heuristic approach that attempts to allocate mobile users to their running neighbor physical machines with the most computing resources without considering energy consumption. It is the third baseline approach.
4) EDR-Ab [8]: This is a heuristic approach that powers off edge servers that consume the most energy and allocates mobile users to the remaining edge servers.
5) EDR-Ae [8]: This is an enhanced version of EDR-Ab in the context of this research. Unlike EDR-Ab that powers off edge servers as a whole, EDR-Ae powers off individual physical machines that consume the most energy.

In the experiments, given an EDR strategy, unused physical machines are powered off (except EDR-Ab which powers off entire edge servers) to implement the EDR strategy.

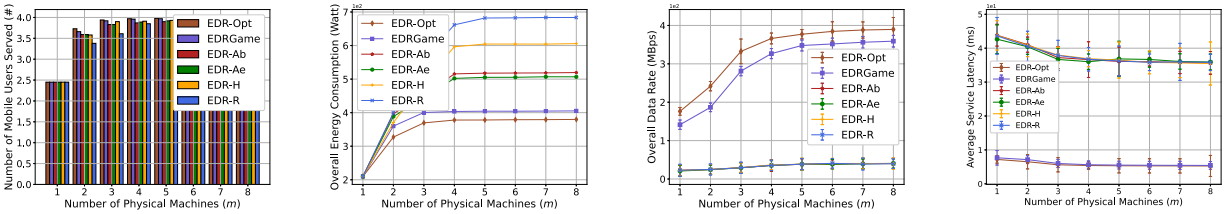
Effectiveness. Figs. 2, 3, and 4 compare the effectiveness of the six approaches in Set #1 and demonstrate the impacts of n, m and τ in the system. Overall, EDR-Opt serves the most mobile users with the lowest overall energy consumption, and achieves the highest overall data rate. EDRGame achieves the second-highest performance of all. Compared with EDR-Opt, EDRGame allocates only 2.13% fewer mobile users, incurs 13.61% more system energy consumption, achieves a 12.79% lower overall data rate and a 5.84% higher service latency on average in Set #1. Fig. 2a shows that the increase in n in Set #1.1 results in a roughly linear increase in the number of users served by all the approaches. This is expected as long as the mobile users' overall resource demand does not exceed the combined capacity of all the physical machines in the system. Similarly, the corresponding system energy consumption incurred by these approaches increases roughly linearly with n, as shown in Fig. 2b. Unsurprisingly, the linear increases in the number of users served lead to similar increases in their overall data rate achieved by these approaches, as shown in (Fig. 2c). Their average data rate remains stable as n increases. As a result, their average service latency remains stable, as shown in Fig. 2d. Fig. 3a shows that, when m increases in Set #1.2, more physical machines in the system can accommodate more mobile users unless all the mobile users are already served, i.e., when $m \geq 4$. To serve an increasing number of mobile users, the corresponding system energy consumption incurred by these approaches increases, quickly at the beginning and slowly afterwards. Similar to Set #1.1, more mobile users served result in higher overall energy consumption and overall data rates, as shown in Figs. 3b and 3c. When m increases from 1 to 4, the increase in the number of allocated users directly increases their overall data rates. In the meantime, users can be allocated to more edge servers, which decreases their interference and increases their average data rate. This leads to a decrease in their average

7. We also try to evenly allocate all the transmit power of each subchannel to the mobile users allocated by the comparing approaches to that subchannel. However, this setting leads to lower overall data rates for these approaches. Thus, the corresponding results are not reported in this paper.



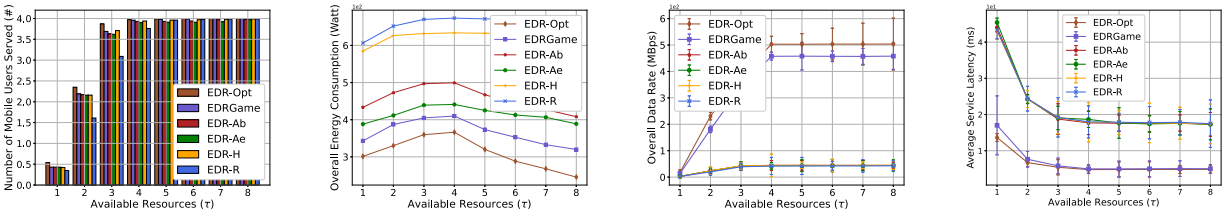
(a) Number of Allocated Users (b) System Energy Consumption (c) Overall Data Rate (d) Average Service Latency

Fig. 2. Effectiveness versus number of mobile users (Set #1.1).



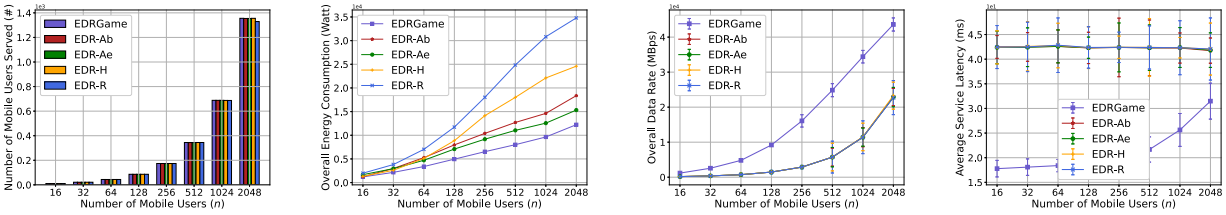
(a) Number of Allocated Users (b) System Energy Consumption (c) Overall Data Rate (d) Average Service Latency

Fig. 3. Effectiveness versus number of physical machines (Set #1.2).



(a) Number of Allocated Users (b) System Energy Consumption (c) Overall Data Rate (d) Average Service Latency

Fig. 4. Effectiveness versus available resources (Set #1.3).

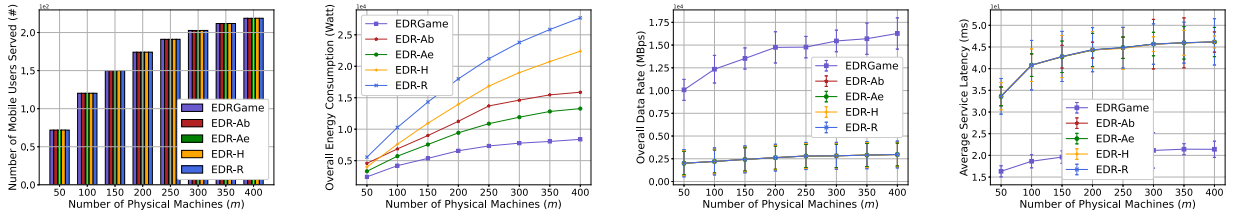


(a) Number of Allocated Users (b) System Energy Consumption (c) Overall Data Rate (d) Average Service Latency

Fig. 5. Effectiveness versus number of mobile users (Set #2.1).

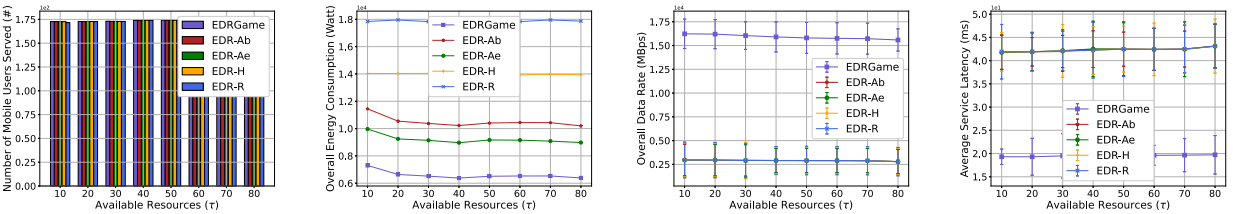
service latency, as shown in Fig. 2d. As m continues to increase, the increase in users' overall data rate slows down because most of them have already been allocated adequate transmit power to maximize their individual data rates. This is also the reason why their average service latency stabilizes as well, as shown in Fig. 3d. Similar to Set #1.2, the increase in τ at the beginning in Set #1.3 allows more mobile users to be served with a higher overall data rate, as shown in Figs. 4a and 2c. This is because more resources on individual physical machines can serve more mobile users in general. The number of physical machines needed to serve all the mobile users increases as well. This incurs more system energy consumption, as shown in Fig. 4b. When τ exceeds 4, all the mobile users in the system can be served and the number of allocated mobile users stabilizes, as shown in Fig. 4a. The increase in τ after $\tau = 4$ allows a fewer number of the most powerful physical machines to accommodate all the mobile users in the system. Accordingly, the corresponding system energy consumption decreases with the increase in τ when τ exceeds 4, which can be observed in Fig. 4b. When all allocated mobile users have reached the maximum data rate, as described in Section 4.2, their overall data rate stabilizes, as shown in Fig. 4c. Thus, $\tau = 4$ is the turning point in Fig. 4c. It is also the turning point in Fig. 4d where users' average service latency starts to stabilize as their average data rate stops increasing.

Figs. 5, 6, and 7 demonstrate the effectiveness results of experiments Set #1 - Set #3. In general, EDRGame outperforms EDR-Ab, EDR-Ae, EDR-H and EDR-R with different margins in different cases. In experiments, to simulate typical EDR scenarios, we set τ to be adequately large so that the total computing resources available on physical machines are more than enough to accommodate all the mobile users in the system. Thus, it is not difficult for most of the approaches to accommodate all the mobile users. Of all the five approaches, EDRGame allocates the most mobile



(a) Number of Allocated Users (b) System Energy Consumption (c) Overall Data Rate (d) Average Service Latency

Fig. 6. Effectiveness versus number of physical machines (Set #2.2).



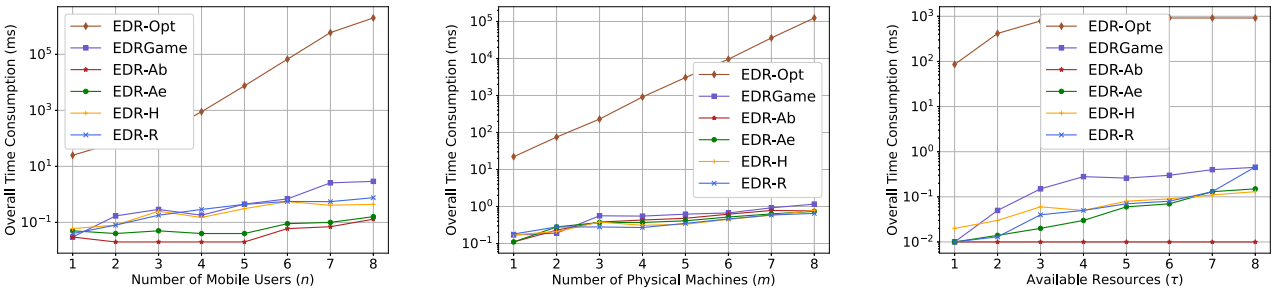
(a) Number of Allocated Users (b) System Energy Consumption (c) Overall Data Rate (d) Average Service Latency

Fig. 7. Effectiveness versus available resources (Set #2.3).

users, outperforming EDR-Ab, EDR-Ae, EDR-H and EDR-R by an average of 1.34%, 0.57%, 0.48% and 0.62%, respectively, as shown in Figs. 5a–7a. In the meantime, as shown in Figs. 5b–7b, its system energy consumption is lower than EDR-Ab, EDR-Ae, EDR-H and EDR-R by 37.51%, 28.91%, 48.77% and 61.46%, respectively. In terms of the overall data rate, EDRGame outperforms EDR-Ab, EDR-Ae, EDR-H and EDR-R by 436.26%, 426.23%, 434.90% and 435.63%, respectively, as shown in Figs. 5c–7c. EDRGame's outstanding performance in maximizing users' overall data rate allows users to enjoy an average service latency much lower compared with the other approaches, as shown in Figs. 5d–7d. Specifically, allocated by EDRGame's, users' average service latency is 52.23%, 52.25%, 52.27% and 52.29% lower than EDR-Ab, EDR-Ae, EDR-H and EDR-R across Sets #2.1-#2.3. These phenomena indicate the importance of considering NOMA in EDR.

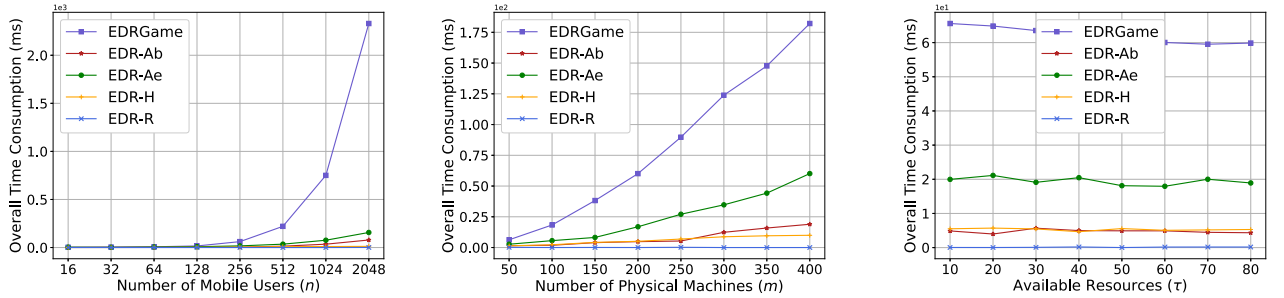
Fig. 5a shows that when n increases in Set #2.1, the number of mobile users and system energy consumption achieved by all the five approaches increase linearly. This is expected as long as the total resources required to serve all the mobile users do not exceed the system's resource capacity, i.e., the total resources available on the physical machines. Managing to serve the most mobile users, EDRGame also requires much less system energy consumption, i.e., 30.80% less than EDR-Ab, 26.60% less than EDR-Ae, 39.98% less than EDR-H and 57.19% than EDR-R, as shown in Fig. 5b. EDRGame's advantage can also be seen in its ability to maximize the overall data rate, as shown in Fig. 5c. Overall, it outperforms EDR-Ab, EDR-Ae, EDR-H and EDR-R by 417.17%, 417.11%, 415.00% and 415.86%, respectively. EDRGame's significant advantages in maximizing the overall data rate indicate the importance of considering interference in NOMA-based MEC systems. By allocating transmit power across mobile users, EDR-Ab, EDR-Ae, EDR-H and EDR-R cannot properly allocate mobile users to the right subchannels on different base stations to reduce the intra-cell and inter-cell interference in the system. This largely impacts the overall data rate received by the mobile users in the system. EDRGame's remarkable advantage in ensuring high data rates for massive users translates to its ability to ensure low service latency. This can be clearly observed in Fig. 5d. Overall, when allocated by EDRGame, users' average service latency is 49.38%, 49.42%, 49.52% and 49.55% lower compared with when they are allocated by EDR-Ab, EDR-Ae, EDR-H and EDR-R. When n increases rapidly from 16 to 2,048 in Set #2.1, EDRGame ensures a gentle increase in users' average service latency from 17.78 milliseconds to 31.48 milliseconds. The experimental results in Set #2 indicate EDRGame's ability to accommodate massive mobile users energy-efficiently based on the NOMA scheme.

In Set #2.2, the number of allocated mobile users increases with the increase in m, as shown in Fig. 6a. This is because when more physical machines are deployed at more base stations, more mobile uses can be covered in the system. As a result, more physical machines running to serve mobile users consume more energy, as demonstrated by the continuously increasing system energy consumption required by all the approaches as shown in Fig. 6b. Being able to aggregate mobile users and powering off idle machines, EDRGame and EDR-Ae excel at minimizing and stabilizing energy consumption when m increases. Comparing with other approaches, EDRGame achieves more energy saving than EDR-Ab, EDR-Ae, EDR-H, and EDR-R by 44.51%, 31.74%, 53.42%, and 64.09%, respectively. In Fig. 6c, an increase in m places more physical machines (and more base stations as a result) in the system, which can accommodate more mobile users with satisfactory computation and communication resources, as demonstrated in Fig. 6a. The mobile users' overall data rate increases accordingly. In Set #2, EDRGame achieves overall data rates much higher than other approaches, outperforming EDR-Ab, EDR-Ae, EDR-H and EDR-R, by 442.94%, 442.91%, 442.35% and 442.62%, respectively. However, users' average data rate decreases slightly because of the accelerating increase in the interference in the system. This is evidenced by Fig. 6d where users' average service latency increases gradually from 16.32 milliseconds to 21.41 milliseconds when m increases.



(a) Number of Mobile Users (Set #1.1) (b) Number of Physical Machines (Set #1.2) (c) Available Server Resources (Set #1.3)

Fig. 8. Time consumption (Set #1).



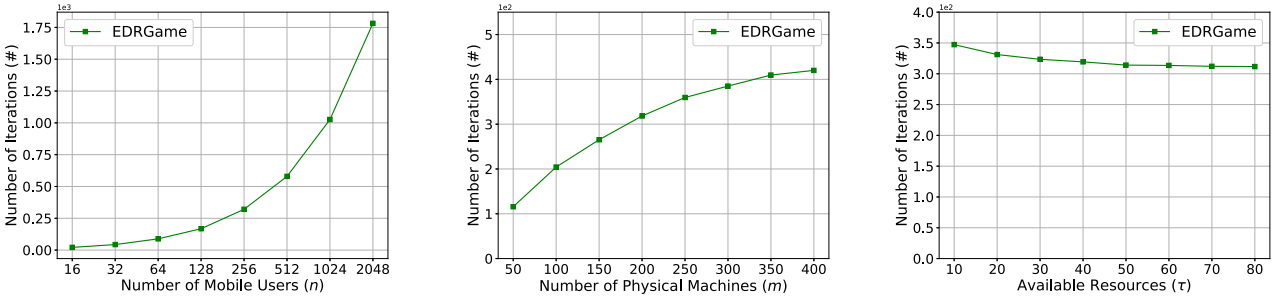
(a) Number of Mobile Users (Set #2.1) (b) Number of Physical Machines (Set #2.2) (c) Available Server Resources (Set #2.3)

Fig. 9. Time consumption (Set #2).

In Set #2.3, with the increase in τ, the number of mobile users allocated by the approaches increases slightly, as shown in Fig. 7a. The reason is that 200 physical machines with $\tau = 10$ can already accommodate most of the mobile users. When τ increases from 10 to 30, each physical machine can serve more mobile users. EDRGame and EDR-Ae leverage that and aggregate mobile users to fewer physical machines with the most computing resources. This reduces their system energy consumption, as shown in Fig. 7b. As τ continues to increase, most physical machines have enough computing resources to accommodate all their nearby mobile users. The number of physical machines needed does not decrease further. As a result, EDRGame's system energy consumption stabilizes after τ exceeds 30. EDRGame is again the clear winner in this set of experiments, accommodating the most mobile users, consuming 37.21% less energy than EDR-Ab, 28.37% less than EDR-Ae, 52.91% less than EDR-H and 63.08% less than EDR-R. As shown in Fig. 7c, EDRGame achieves an overall data rate more than 400% higher than other approaches. With an increase in τ, each individual physical machine in the system can accommodate more mobile users. This increases the number of mobile users allocated to each individual subchannel, incurring slightly higher intra-cell interference and lowering the overall data rate also slightly. As expected, this increases users' average service latency mildly, as illustrated in Fig. 7d.

Efficiency. Fig. 8 demonstrates the time taken by different approaches to find an EDR solution in Set #1. The time consumption of EDR-Opt grows exponentially as n increases from 1 to 8 in Set #1.1, as shown in Fig. 8a, and as m increases from 1 to 8 in Set #1.2, as shown in Fig. 8b. Compared with EDR-Opt, the other five approaches take almost no time to find an EDR solution. Apparently, it is impractical to employ EDR-Opt to solve large-scale EDR problems. This validates Theorem 1. In Set #1.3, when τ increases from 1 to 4, individual physical machines have more computing resources to accommodate more mobile users. This increases the average number of options for allocating individual mobile users, taking EDR-Opt more time to find the optimal EDR solution, as shown in Fig. 8c. As τ continues to increase, more physical machines have adequate computing resources to accommodate all their nearby mobile users. This makes it easier for EDR-Opt to find the optimal EDR solution, and stabilizes its time consumption.

EDR-Opt is excluded from Set #2 because its scale is too large for EDR-Opt to find an EDR solution within a reasonable amount of time. This allows us to evaluate the efficiency of EDRGame properly. Figs. 9a and 9b show that, in Set #2.1 - Set #2.3, the time consumption of EDRGame increases linearly with n and m. The reason for the increase in Fig. 9a is straightforward - more iterations are needed to make allocation decisions for all the mobile users, as shown in Fig. 10a. In Set #2.2, the increase in m slows down EDRGame's convergence, as shown in Fig. 10b. As a result, its time consumption increases when m increases, as shown in Fig. 9b. This correlation between EDRGame's time consumption and convergence time is different from Set #2.1. As m increases from 50 to 400 in Set #2.2, more physical machines are available to cover the mobile users. There are more allocation options for each individual mobile user. Thus, mobile users may be dispersed across different physical machines. EDRGame needs to aggregate these mobile users later, which increases its convergence time, as shown in Fig. 10b. As a result, the extra physical machines available around each individual mobile user complicate the process for finding the right physical machines for them. This



(a) Number of Mobile Users (Set #2.1) (b) Number of Physical Machines (Set #2.2) (c) Available Server Resources (Set #2.3)

Fig. 10. Convergence time (Set #2).

contributes to the increase in EDRGame's time consumption as shown in Fig. 9b. In Set #2.3, $\tau = 10$ is adequately large for the physical machines to accommodate all the mobile users, as shown in Fig. 7a. Thus, an increase in τ in Set #2.3 gradually reduces EDRGame's time consumption, as shown in Fig. 9c, for the same reason for the decrease in its time consumption in Set #2.3. This is consistent with the decrease in EDRGame's convergence time shown in Fig. 10c.

7 Conclusion and Future Work

This paper studied the Edge Demand Response (EDR) problem in a Non-Orthogonal Multiple Access (NOMA) based Mobile Edge Computing environment (MEC). We proved that the EDR problem is $\mathcal{NP}$-hard and proposed a two-phase game-theoretical approach for finding EDR solutions. We theoretically analyzed and experimentally evaluated its performance against three baseline approaches and two state-of-the-art approaches. Extensive experimental results demonstrate its high effectiveness and efficiency.

In our future work, we will consider the possibility that the physical machines facilitating edge servers can sleep and wake up with minimum switching costs. This will allow us to investigate more sophisticated EDR scenarios where mobile users' mobility, dynamic arrivals and departures can be taken into account. The impacts of more sophisticated NOMA schemes on EDR will be studied, e.g., those that support multiple antennas, imperfect channel state information and combination of OMA [43].

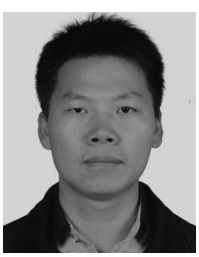
Acknowledgments

This research was supported by the Australian Research Council Discovery Projects under Grants DP180100212 and DP200102491.

References

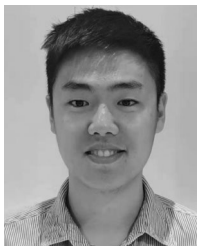
[1] N. Jones, "How to stop data centres from gobbling up the world's electricity," *Nature*, vol. 561, no. 7722, pp. 163–167, 2018.
[2] A. S. G. Andrae and T. Edler, "On global electricity usage of communication technology: Trends to 2030, " *Challenges*, vol. 6, no. 1, pp. 117–157, 2015.
[3] S. Malla and K. Christensen, "A survey on power management techniques for oversubscription of multi-tenant data centers," *ACM Comput. Surv.*, vol. 52, no. 1, pp. 1–31, 2019.
[4] Z. Zhou, F. Liu, S. Chen, and Z. Li, "A truthful and efficient incentive mechanism for demand response in green datacenters," *IEEE Trans. Parallel Distrib. Syst.*, vol. 31, no. 1, pp. 1–15, Jan. 2020.
[5] H. Yin *et al.*, "Edge provisioning with flexible server placement," *IEEE Trans. Parallel Distrib. Syst.*, vol. 28, no. 4, pp. 1031–1045, Apr. 2017.
[6] A. Ceselli, M. Premoli, and S. Secci, "Mobile edge cloud network design optimization," *IEEE/ACM Trans. Netw.*, vol. 25, no. 3, pp. 1818–1831, Jun. 2017.
[7] X. Ge, S. Tu, G. Mao, C.-X. Wang, and T. Han, "5G ultra-dense cellular networks," *IEEE Wirel. Commun.*, vol. 23, no. 1, pp. 72–79, Feb. 2016.
[8] S. Chen, L. Jiao, L. Wang, and F. Liu, "An online market mechanism for edge emergency demand response via cloudlet control," in *Proc. IEEE Int. Conf. Comput. Commun.*, 2019, pp. 2566–2574.
[9] Y. Xiao and M. Krunz, "Distributed optimization for energy-efficient fog computing in the tactile internet," *IEEE J. Sel. Areas Commun.*, vol. 36, no. 11, pp. 2390–2400, Nov. 2018.
[10] Q. He *et al.*, "A game-theoretical approach for user allocation in edge computing environment," *IEEE Trans. Parallel Distrib. Syst.*, vol. 31, no. 3, pp. 515–529, Mar. 2020.
[11] X. Xia, F. Chen, Q. He, J. Grundy, M. Abdelrazek, and H. Jin, "Cost-effective app data distribution in edge computing," *IEEE Trans. Parallel Distrib. Syst.*, vol. 32, no. 1, pp. 31–44, Jan. 2021.
[12] L. Chen, S. Zhou, and J. Xu, "Computation peer offloading for energy-constrained mobile edge computing in small-cell networks," *IEEE/ACM Trans. Netw.*, vol. 26, no. 4, pp. 1619–1632, Aug. 2018.
[13] Y. Saito, Y. Kishiyama, A. Benjebbour, T. Nakamura, A. Li, and K. Higuchi, "Non-orthogonal multiple access (NOMA) for cellular future radio access," in *Proc. IEEE Veh. Technol. Conf.*, 2013, pp. 1–5.
[14] Y. Liu, Z. Qin, M. Elkashlan, Z. Ding, A. Nallanathan, and L. Hanzo, "Nonorthogonal multiple access for 5G and beyond," *Proc. IEEE*, vol. 105, no. 12, pp. 2347–2381, 2017.
[15] N. H. Tran, D. H. Tran, S. Ren, Z. Han, E. Huh, and C. S. Hong, "How geo-distributed data centers do demand response: A game-theoretic approach," *IEEE Trans. Smart Grid*, vol. 7, no. 2, pp. 937–947, Mar. 2016.
[16] J. Chen, D. Ye, Z. Liu, S. Ji, Q. He, and Y. Xiang, "A truthful and near-optimal mechanism for colocation emergency demand response," *IEEE Trans. Mobile Comput.*, vol. 20, no. 9, pp. 2728–2744, Sep. 2021.
[17] F. Bonomi, R. Milito, J. Zhu, and S. Addepalli, "Fog computing and its role in the Internet of Things," in *Proc. 1st Ed. Workshop Mobile Cloud Comput.*, 2012, pp. 13–16.
[18] Y. Mao, C. You, J. Zhang, K. Huang, and K. B. Letaief, "A survey on mobile edge computing: The communication perspective," *IEEE Commun. Surveys Tut.*, vol. 19, no. 4, pp. 2322–2358, 2017.
[19] Y. Mao, J. Zhang, and K. B. Letaief, "Dynamic computation offloading for mobile-edge computing with energy harvesting devices," *IEEE J. Sel. Areas Commun.*, vol. 34, no. 12, pp. 3590–3605, Dec. 2016.
[20] P. Lai *et al.*, "Optimal edge user allocation in edge computing with variable sized vector bin packing," in *Proc. Int. Conf. Serv.-Oriented Comput.*, 2018, pp. 230–245.
[21] X. Xia, F. Chen, Q. He, J. Grundy, M. Abdelrazek, and H. Jin, "Online collaborative data caching in edge computing," *IEEE Trans. Parallel Distrib. Syst.*, vol. 32, no. 2, pp. 281–294, Feb. 2021.
[22] B. Li, Q. He, G. Cui, X. Xia, F. Chen, H. Jin, and Y. Yang, "Read: Robustness-oriented edge application deployment in edge computing environment," *IEEE Trans. Serv. Comput.*, early access, Aug. 10, 2020, doi: 10.1109/TSC.2020.3015316.
[23] F. Chen, J. Zhou, X. Xia, H. Jin, and Q. He, "Optimal application deployment in mobile edge computing environment," in *Proc. IEEE 13th Int. Conf. Cloud Comput.*, 2020, pp. 184–192.
[24] B. Li, Q. He, F. Chen, H. Jin, Y. Xiang, and Y. Yang, "Auditing cache data integrity in the edge computing environment," *IEEE Trans. Parallel Distrib. Syst.*, vol. 32, no. 5, pp. 1210–1223, May 2021.
[25] Z. Ding, P. Fan, and H. V. Poor, "Impact of non-orthogonal multiple access on the offloading of mobile edge computing," *IEEE Trans. Commun.*, vol. 67, no. 1, pp. 375–390, Jan. 2019.

[26] Y. Wu, L. P. Qian, K. Ni, C. Zhang, and X. Shen, "Delay-minimization nonorthogonal multiple access enabled multi-user mobile edge computation offloading," *IEEE J. Sel. Top. Signal Process.*, vol. 13, no. 3, pp. 392–407, Jun. 2019.
[27] Z. Ning *et al.*, "Mobile edge computing enabled 5G health monitoring for internet of medical things: A decentralized game theoretic approach," *IEEE J. Sel. Areas Commun.*, vol. 39, no. 2, pp. 463–478, Feb. 2021,.
[28] K. Wang, Y. Liu, Z. Ding, A. Nallanathan, and M. Peng, "User association and power allocation for multi-cell non-orthogonal multiple access networks," *IEEE Trans. Wireless Commun.*, vol. 18, no. 11, pp. 5284–5298, Nov. 2019.
[29] X. Chen, L. Jiao, W. Li, and X. Fu, "Efficient multi-user computation offloading for mobile-edge cloud computing," *IEEE/ACM Trans. Netw.*, vol. 24, no. 5, pp. 2795–2808, Oct. 2016.
[30] R. Zhou, Z. Li, C. Wu, and M. Chen, "Demand response in smart grids: A randomized auction approach," *IEEE J. Sel. Areas Commun.*, vol. 33, no. 12, pp. 2540–2553, Dec. 2015.
[31] T. Q. Dinh, J. Tang, Q. D. La, and T. Q. Quek, "Offloading in mobile edge computing: Task allocation and computational frequency scaling," *IEEE Trans. Commun.*, vol. 65, no. 8, pp. 3571–3584, Aug. 2017.
[32] Z. Ding, X. Lei, G. K. Karagiannidis, R. Schober, J. Yuan, and V. K. Bhargava, "A survey on non-orthogonal multiple access for 5G networks: Research challenges and future trends," *IEEE J. Sel. Areas Commun*, vol. 35, no. 10, pp. 2181–2195, Oct. 2017.
[33] P. Lai *et al.*, "Cost-effective user allocation in 5G NOMA-based mobile edge computing systems," *IEEE Trans. Mobile Comput.*, early access, May 4, 2021, doi: 10.1109/TMC.2021.3077470.
[34] Z. Zhang, H. Sun, and R. Q. Hu, "Downlink and uplink non-orthogonal multiple access in a dense wireless network," *IEEE J. Sel. Areas Commun.*, vol. 35, no. 12, pp. 2771–2784, Dec. 2017.
[35] G. Cui *et al.*, "Interference-aware SaaS user allocation game for edge computing," *IEEE Trans. Cloud Comput.*, early access, Jul. 10, 2020, doi: 10.1109/TCC.2020.3008448.
[36] G. Cui, Q. He, F. Chen, Y. Zhang, H. Jin, and Y. Yang, "Interference-aware game-theoretic device allocation for mobile edge computing," *IEEE Trans. Mobile Comput.*, early access, Mar. 5, 2021, doi: 10.1109/TMC.2021.3064063.
[37] G. Cui, Q. He, F. Chen, H. Jin, and Y. Yang, "Trading off between multi-tenancy and interference: A service user allocation game," *IEEE Trans. Serv. Comput.*, early access, Aug. 6, 2020, doi: 10.1109/TSC.2020.3028760.
[38] Y. Wu, L. Qian, H. Mao, X. Yang, H. Zhou, and X. Shen, "Optimal power allocation and scheduling for non-orthogonal multiple access relay-assisted networks," *IEEE Trans. Mobile Comput.*, vol. 17, no. 11, pp. 2591–2606, Nov. 2018.
[39] Z. Yang, C. Pan, W. Xu, Y. Pan, M. Chen, and M. Elkashlan, "Power control for multi-cell networks with non-orthogonal multiple access," *IEEE Trans. Wireless Commun.*, vol. 17, no. 2, pp. 927–942, Feb. 2018.
[40] X. Chen, "Decentralized computation offloading game for mobile cloud computing," *IEEE Trans. Parallel Distrib. Syst.*, vol. 26, no. 4, pp. 974–983, Apr. 2015.
[41] Q. He *et al.*, "A game-theoretical approach for mitigating edge DDoS attack," *IEEE Trans. Dependable Secure Comput.*, early access, Jan. 29, 2021, doi: 10.1109/TDSC.2021.3055559.
[42] P. Lai *et al.*, "Cost-effective app user allocation in an edge computing environment," *IEEE Trans. Cloud Comput.*, early access, Jun. 11, 2020, doi: 10.1109/TCC.2020.3001570.
[43] W. Shin, M. Vaezi, B. Lee, D. J. Love, J. Lee, and H. V. Poor, "Non-orthogonal multiple access in multi-cell networks: Theory, performance, and practical challenges," *IEEE Commun. Mag.*, vol. 55, no. 10, pp. 176–183, Oct. 2017.
[44] M. R. Garey and D. S. Johnson, *Computers and Intractability: A guide to the Theory of NP Completeness* (A Series of Books in the Mathematical Sciences). San Francisco, CA: W. H. Freeman and Co., 1979.



Guangming Cui received the master's degree from Anhui University, China, in 2018. He is currently working toward the PhD degree with the Swinburne University of Technology. His research interests include mobile edge computing, service computing, and software engineering.

Qiang He (Senior Member, IEEE) received the first PhD degree from the Swinburne University of Technology, Australia, in 2009, and the second PhD degree in computer science and engineering from the Huazhong University of Science and Technology, China, in 2010. He is currently an associate professor with Swinburne. His research interests include mobile edge computing, service computing, software engineering, and cloud computing.



Xiaoyu Xia received the master's degree from The University of Melbourne, Australia, in 2015. He is currently working toward the PhD degree with Deakin University. His research interests include mobile edge computing, service computing, cloud computing, green computing, and software engineering.

Feifei Chen (Member, IEEE) received the PhD degree from the Swinburne University of Technology, Australia, in 2015. She is currently a lecturer with Deakin University. Her research interests include mobile edge computing, software engineering, cloud computing, and green computing.

Tao Gu (Senior Member, IEEE) received the PhD degree from the National University of Singapore. He is currently a professor with the Department of Computing, Macquarie University, Sydney, Australia. His research interests include mobile computing, ubiquitous computing, wireless sensor networks, sensor data analytics, and Internet of Things. He is an associate editor of the *IEEE Transactions of Mobile Computing*.

Hai Jin (Fellow, IEEE) received the PhD degree in computer engineering from Huazhong University of Science and Technology (HUST) in 1994. He is currently a Cheung Kung Scholars chair professor of computer science and engineering with the HUST, China. His research interests include computer architecture, virtualization technology, cluster computing and cloud computing, peer-to-peer computing, network storage, and network security.

Yun Yang (Senior Member, IEEE) received the PhD degree in computer science from the University of Queensland, Australia, in 1992. He is currently a full professor with the School of Software and Electrical Engineering, Swinburne University of Technology, Melbourne, Australia. His research interests include software technologies, mobile edge computing, cloud computing, workflow systems, and service computing. He is an associate editor of the *IEEE Transactions on Parallel and Distributed Computing Systems*.