Non-Orthogonal Network Slicing for eMBB Service in a Multi-UAV-Aided Network

Xing Xi, Xianbin Cao, Senior Member, IEEE, Peng Yang, Jingxuan Chen, Dapeng Oliver Wu, Fellow, IEEE

Abstract—This paper is concerned with the network slicing problem for enhanced mobile broadband (eMBB) service in a multi-UAV-aided network. Different from most of the existing network slicing approaches, we investigate non-orthogonal network slicing implementation joint with network resource allocation for eMBB service in this paper. Specifically, our objective is to maximize the system energy efficiency while balancing the fairness of service among user equipments (UEs) and the fairness of power consumption among UAs under the constraints on eMBB UEs’ quality of service, UAs’ network capacity and power consumption as well. We formulate the energy efficiency maximization problem as a mixed-integer non-convex programming problem. To alleviate this challenging problem, we develop a solution framework of alternatively optimizing a non-orthogonal slice request acceptance problem and a non-convex network resource allocation problem. Besides, we solve the slice request acceptance problem by designing a two-stage optimization method and tackle the non-convexity of the resource allocation problem by exploiting a successive convex approximation approach. Simulation results show that our proposed framework can achieve the highest energy efficiency compared with other benchmark algorithms.

I. INTRODUCTION

The fifth-generation (5G) wireless networks, which are gradually commercializing, are expected to support three categories of communication services, i.e., enhanced mobile broadband (eMBB), massive machine-type communications (mMTC), ultra-reliable and low latency communications (URLLC) [1]. For various services differing in quality of service (QoS) requirements and device types, the traditional one-size network architecture may not be feasible. A promising solution to provide services tailored for various QoS requirements is network slicing, which separates a common shared physical radio access network into multiple logical (or virtual) slices [2]. Therefore, network slicing has attracted significant attention in wireless academia and industry. Yet, to improve the utilization of network resources in network slicing, efficient network slicing implementation and efficient network resource allocation among slices, which are crucial challenges, should be risen to.

Recently, many works of network slicing have been proposed to solve the challenges mentioned above [3–5]. For example, Rost et al. investigated the basic architecture for network slicing in 5G wireless networks and illustrated the implementation of network slicing in the radio access network and the core network [3]. However, they only considered the implementation of network slicing and ignored the efficient allocation of resources among slices. Some other works studied the share of network resources among slices. For example, Zhang et al. designed a network-slicing-based architecture for 5G wireless networks. Based on this network architecture, they investigated mobility management and virtualized radio resource allocation (including power and subchannel) technologies [4]. Albonda et al. proposed an RAN slicing framework for eMBB and vehicle-to-everything (V2X) services, which included an RAN slicing strategy based on an offline reinforcement learning and a heuristic algorithm of allocating resources to different slices [5].

The works in [3–5] are all based on orthogonal multiple access (OMA) techniques. Besides, to improve the utilization efficiency of network resources, non-orthogonal multiple access (NOMA) techniques have been studied in network slicing. For example, Popovski et al. studied the non-orthogonal share of RAN resources in an uplink communication scenario where a set of eMBB, mMTC, and URLLC devices connect to a common base station [6]. Kassab et al. investigated the performance of OMA and NOMA for the multiplexing of eMBB and URLLC devices in an uplink multi-cell network and analyzed NOMA with different architectures, such as puncturing and successive interference cancellation [7].

However, all of the above network slicing approaches focus on terrestrial base station networks and do not consider the case of network failure or network congestion (e.g., infrastructure malfunction, flash crowd traffic, and remote areas), which may lead to communication service interruption. The guarantee of service provision, in this case, is one of the major challenges in 5G and 6G wireless networks. A promising solution to alleviate the effect of network failure or network congestion is the utilization of unmanned aerial vehicle (UAV) base stations (i.e., low-altitude UAVs equipped with transceivers), which can support fast communication service recovery or even network performance enhancement [8]. Motivated by the advantages of UAV base stations, the extension of network slicing in UAV-aided networks is gradually attracting the attention of researchers. For example, Xilouri et al. studied the integration of UAV-aided networks with network slicing and virtualization in the frame of a 5G wireless network and discussed the architecture and possible applications of flying modes on this basis [8]. Budhiraja et al. investigated the application-specific NOMA-based network

X. Xi, X. Cao, and J. Chen are with the School of Electronic and Information Engineering, Beihang University, Beijing 100083, China, and also with the Key Laboratory of Advanced Technology, Near Space Information System (Beihang University), Ministry of Industry and Information Technology of China, Beijing 100083, China.

P. Yang is with the Information Systems Technology and Design, Singapore University of Technology and Design, 487372 Singapore.

D. Wu is with the Department of Electrical and Computer Engineering, University of Florida, Gainesville FL 32611 USA.
architecture that supports nonorthogonal resource sharing from a set of eMBB, mMTC, and URLLC devices to a common base station. However, the above two works focused on the architecture and applications of network slicing in UAV-aided networks and did not consider the efficient allocation of network resources among network slices.

Considering the efficient utilization of network resources, this paper investigates a joint non-orthogonal network slicing implementation and network resource (UAV transmit power) allocation problem for the eMBB service in a multi-UAV-aided network. Particularly, the main contributions are summarized as follows:

- We build a NOMA-based downlink communication model in a multi-UAV-aided network. Based on this model, we formulate a joint non-orthogonal network slicing and resource allocation optimization problem with a goal of maximizing the system energy efficiency under the constraints on QoS requirements, network capacity, and power consumption. This energy efficiency maximization problem is confirmed to be a mixed-integer non-convex programming (MINCP) problem, which is challenging to mitigate.

- To alleviate the formulated energy efficiency maximization problem, we first decompose it into two separated subproblems, namely the optimization of slice request acceptance for fixed transmit power and the optimization of power allocation for fixed slice request. We design a two-stage optimization method to solve the slice request acceptance subproblem and exploit a successive convex approximation approach to tackle the non-convexity of the power allocation subproblem. Based on the solutions of the above two subproblems, we then proposed an iterative solution algorithm by alternatively optimizing these two subproblems.

- Since the above slice request acceptance optimization includes integer linear programming (ILP) problems, we propose a fast slice request acceptance algorithm that loses some energy efficiency, which may be regarded as an alternative in highly dynamic scenarios.

- Simulation results demonstrate that our proposed algorithms can achieve the performance (energy efficiency) gain compared with other benchmark algorithms.

The rest of this paper is organized as follows: We present the system model and the problem formulation in Section II. Section III and Section IV describe the slice request acceptance optimization and the power allocation optimization, respectively. Based on the results in Section III and Section IV, we develop the problem solution for the formulated problem in Section V. Section VI shows our simulation results and Section VII concludes this paper.

II. SYSTEM MODEL AND PROBLEM FORMULATION

A. System Model

In this paper, we consider a NOMA-based downlink communication scenario in a network slicing system. In this scenario, multiple UAV base stations (UBSs) are deployed to assist a macro base station (MBS) to provide radio access for a collection of congested terrestrial eMBB user equipments (UEs) in a geographical area. All UBSs connect to a centralized network operator by wireless fronthaul links, and the operator decides whether to accept or reject the slice requests, so that network slicing can be implemented smoothly. The set of UBSs and the set of UEs are denoted by $J = \{1, 2, \ldots, J\}$ and $I = \{1, 2, \ldots, J\}$, respectively. We assume that the locations of these UBSs and UEs are fixed and known, and these UBSs are deployed at the same altitude $H$. Meanwhile, this paper considers an OFDMA communication system. The total channel bandwidth is $W$ and is equally divided into $N$ orthogonal subchannels denoted by $N = \{1, 2, \ldots, N\}$. Denote $SC_{jn}$ as the subchannel $n$ of UBS $j$. In the network slicing system, a network slice is allocated with a subchannel $SC_{jn}$. We let $a_{ijn}$ be an binary variable indicating slice request and let $A = \{a_{ijn}, \forall i, j, n\}$ denote the slice request matrix. We set $a_{ijn} = 1$ if a slice request that slice $SC_{jn}$ provides service to UE $i$ is accepted/admitted by the network operator; otherwise, $a_{ijn} = 0$. This paper investigates the optimization of joint slice request acceptance and UBSs’ power allocation.

Denote the horizontal location of UBS $j$ and the location of UE $i$ by $x_j^h$ and $x_i^h$, respectively. This paper leverages the air-to-ground (ATG) propagation model to calculate the channel gain from UBS $j$ to UE $i$ on subchannel $n$, denoted by $h_{ijn}$. For the ATG link, each UE has a line-of-sight (LoS) connection with a UBS with a specific probability. The LoS probability relies on the environment, the locations of both the UBSs and the UE. The LoS probability function can be expressed as

$$P_{LoS}(H, d_{ijn}) = \frac{1}{1 + \alpha_1 \exp(-\alpha_2(\theta_{ijn} - \alpha_1))}$$

where, $\alpha_1$ and $\alpha_2$ are constant values depending on the environment (e.g., rural, suburban, urban, and dense urban), $\theta_{ijn} = \frac{180}{\pi} \times \arctan(h_{ijn}/d_{ijn})$ is the elevation angle of UE $i$ towards UBS $j$, and $d_{ijn}$ is the horizontal distance from UBS $j$ to UE $i$, i.e., $d_{ijn} = \|x_i^h - x_j^h\|_2$. Also, the non-line-of-sight probability is $P_{NLoS}(H, d_{ijn}) = 1 - P_{LoS}(H, d_{ijn})$. Then, the channel gain from UBS $j$ to UE $i$ on subchannel $n$ can be expressed as

$$h_{ijn} = \frac{Rx_{ijn}^{1/2} R_{Rx} N_0}{g_{ijn}^{1/2} g_{Rx} N_0} 10^{-\frac{P_{LoS}(H, d_{ijn})}{10} - \frac{P_{NLoS}(H, d_{ijn})}{10} + \frac{\eta_{BLos}^{dB}}{10} + \frac{\eta_{NLoS}^{dB}}{10}}$$

where, $g_{ijn}$ and $g_{Rx}$ are the transmit and receive antenna gains from UBS $j$ to UE $i$ on subchannel $n$, $\varsigma = c/f_c$ is the carrier wavelength, where $c$ is the speed of light and $f_c$ is the carrier frequency. $d_{ijn} = \sqrt{(d_{ijn})^2 + H^2}$ is the distance from UBS $j$ to UE $i$ and $d_0$ is a far field reference distance. $\eta_{BLos}^{dB}$ and $\eta_{NLoS}^{dB}$ represent propagation losses corresponding to the LoS and NLoS connections, respectively, which depend on environment.

Denote the location of the MBS by $x^M$. This paper leverages the Friis propagation model to calculate the channel gain from MBS to UE $i$ on subchannel $n$, denoted by $h_{im}^{dB}$. 


The channel gain $h_{in}^M$ can be expressed as

$$h_{in}^M = \frac{g_{in}^{MT} \cdot g_{in}^{Rx} \cdot c^2}{16\pi^2 \left( \frac{d_{in}^M}{d_{in}} \right)^\eta}$$

(3)

where, $g_{in}^{MT}$ and $g_{in}^{Rx}$ are the transmit and receive antenna gains from the MBS to UE $i$ on subchannel $n$, $d_{in}^M = ||x_i^n - x_i^M||_2$ is the distance from the MBS $j$ to UE $i$, and $\eta$ is the path-loss exponent ($\eta \in [2, 6]$).

In the downlink NOMA-based system, the successive interference cancellation (SIC) technique is adopted at the receiver to eliminate the interference from other UEs in the same slice $SC_{jn}$ in a certain decoding order [12]. In this paper, we assume that the UE with higher channel gain can decode the signals of the other UEs with worse channel gain in the same slice $SC_{jn}$. Besides, considering the implementation complexity of SIC, we study the simple case where one slice $SC_{jn}$ can provide service to at most two UEs. Thus, we can obtain the following slice request acceptance constraints

C1: $\sum_{i \in A} a_{ijn} \leq 2, \forall j \in J, n \in N,$

(4)

C2: $a_{ijn} \in \{0, 1\}, \forall i \in I, j \in J, n \in N$.

(5)

Considering the number of UEs sharing slice $SC_{jn}$, we calculate the received signal-to-interference-plus-noise ratio (SINR) in the following two cases.

Case 1: When only one UE $i$ shares slice $SC_{jn}$, we name UE $i$ as a primary UE. Then, the received SINR of primary UE $i$ in $SC_{jn}$ is

$$\gamma_{ijn} = \frac{p_{1,jn} h_{ijn}}{\sum_{k \neq j, k \in J} p_{kn} h_{ikn} + p_{jn} h_{in}^M + \sigma_n^2}$$

(6)

Case 2: When two UEs $i_1$ and $i_2$ share the same slice $SC_{jn}$ with $h_{ijn} > h_{i2jn}$, i.e., UE $i_1$ can eliminate the interference from UE $i_2$ in $SC_{jn}$, we name UE $i_1$ and UE $i_2$ as a primary UE and a secondary UE, respectively. Then, the received SINRs of primary UE $i_1$ and secondary UE $i_2$ in $SC_{jn}$ are

$$\gamma_{i_1jn} = \frac{p_{1,jn} h_{i_1jn}}{\sum_{k \neq j_1, k \in J} p_{kn} h_{i_1kn} + p_{jn} h_{in}^M + \sigma_n^2}$$

(7)

$$\gamma_{i_2jn} = \frac{p_{2,jn} h_{i_2jn}}{\sum_{k \neq j_2, k \in J} p_{kn} h_{i_2kn} + p_{jn} h_{in}^M + \sigma_n^2}$$

(8)

where, $p_{1,jn}$ and $p_{2,jn}$ are the transmit powers allocated to the primary UE and the secondary UE in $SC_{jn}$, respectively, $p_{jn} = p_{1,jn} + p_{2,jn}$ is the total transmit power in $SC_{jn}$, $p_{jn}^M$ is the transmit power of the MBS on subchannel $n$, and $\sigma_n^2$ represents the additive white Gaussian noise on subchannel $n$. Let $\mathcal{P} = \{p_{1,jn}, p_{2,jn}, \forall j, n\}$ denote the the transmit power matrix. Note that this paper focuses on the power allocation of UBSs. Thus, we assume that the transmit power $p_{jn}^M$ is fixed and known. According to the Shannon capacity equation, the achievable data rate of UE $i$ in $SC_{jn}$ is

$$r_{ijn} = \frac{W}{N} \log_2 (1 + \gamma_{ijn})$$

(9)

In this paper, we define UEs’ different quality of service (QoS) requirements by their achievable data rate. We denote $R_i$ and $R_i^{\text{min}}$ as the achievable data rate and the minimum required data rate of UE $i$, respectively. Thus, we can obtain the following QoS requirement constraint

$$C3: R_i = \sum_{j \in J} \sum_{n \in N} a_{ijn} r_{ijn} \geq R_i^{\text{min}}, \forall i \in I$$

(10)

Besides, UBS $j$ needs to transmit data received from UEs to a ground station via a downlink with limited capacity. In this paper, we regard it as the maximum capacity of UBS $j$, denoted by $C_j^{\text{max}}$. Thus, we can obtain the following capacity constraint

$$C4: \sum_{i \in I} \sum_{n \in N} a_{ijn} r_{ijn} \leq C_j^{\text{max}}, \forall j \in J$$

(11)

Next, we denote $p_j$, $p_j^{\text{max}}$ and $p_j^{\text{min}}$ as the transmit power, the circuit power and the maximum power consumption limit of UBS $j$. Thus, we can obtain the following power consumption constraints

$$C5: p_j = \sum_{n \in N} (p_{1,jn} + p_{2,jn}) + p_j^{\text{max}}, \forall j \in J$$

(12)

$$C6: p_{1,jn} \geq 0, p_{2,jn} \geq 0, 0 \leq \gamma_{ijn} \leq J, \forall j \in J, n \in N$$

We denote $\eta_{EE}$ as the energy efficiency (EE), which is defined as the ratio of the minimum achievable data rate among all UEs and the maximum power consumption among all UBSs. As such, the objective function can be written as

$$\eta_{EE} = \frac{\min_{i \in I} R_i}{\max_{j \in J} p_j}$$

(14)

B. Problem Formulation

Considering all constraints and the objective function mentioned above, we can formulate the joint slice request acceptance and power allocation problem as

$$\begin{align*}
\max_{A, \mathcal{P}} \quad & \eta_{EE} = \frac{\min_{i \in I} R_i}{\max_{j \in J} p_j} \\
\text{s.t.} \quad & C1, C2, C3, C4, C5, C6
\end{align*}$$

(15)

We define the optimal EE $\eta_{EE}^*$ as

$$\eta_{EE}^* = \frac{\min_{i \in I} R_i(A^*, \mathcal{P}^*)}{\max_{j \in J} p_j(\mathcal{P}^*)}$$

(16)

where, $A^*$ and $\mathcal{P}^*$ denote the optimal slice request matrix and the optimal transmit power matrix when yielding $\eta_{EE}^*$.

Lemma 1. The optimal EE $\eta_{EE}^*$ can be achieved if and only if

$$\max_{A, \mathcal{P}} \left( \min_{i \in I} R_i(A, \mathcal{P}) - \eta_{EE}^* \left( \max_{j \in J} p_j(\mathcal{P}) \right) \right) = \left( \min_{i \in I} R_i(A^*, \mathcal{P}^*) - \eta_{EE}^* \left( \max_{j \in J} p_j(\mathcal{P}^*) \right) \right) = 0$$

(17)

Proof. Due to the space limitation, we omit the proof of Lemma 1. A similar proof can be found in [12].

According to Lemma 1, we can transform the objective function in (15) into a subtractive form, which is more
tractable. Then, the problem (18) can be rewritten as the following form

\[
\max_{A, P} \left( \min_{i \in I} R_i - \eta EE \left( \max_{j \in J} p_j \right) \right) \\
\text{s.t.} \ C1, C2, C3, C4, C5, C6
\]

(18)

In the problem (18), the slice request variables \(a_{ijn}\) are binary, and thus C1-C4 involve integer constraints. Further, even if \(a_{ijn}\) are fixed, C3 and C4 are not convex constraints. Therefore, (18) is a mixed-integer non-convex programming (MINCP) problem, which is indeterminable or NP-hard and challenging to solve. Besides, \(A\) and \(P\) are coupled in the objective function and the constraints C3 and C4, which increases the difficulty of mitigating (18). Fortunately, we observe that the complexity of (18) may be weaken if \(A\) and \(P\) can be decoupled. Based on this crucial observation, we first decompose the problem (18) into two separated subproblems, namely slice request acceptance optimization with fixed transmit power and power allocation optimization with fixed slice request. Based on the solutions of the above two subproblems, we then develop an iterative solution algorithm for (18) by alternatively optimizing these two subproblems. Besides, we propose a fast slice request acceptance algorithm that loses some energy efficiency and may be regarded as an alternative when running time becomes the primary consideration. Additionally, an implementation system of network slicing in (13) is adopted in this paper. The core idea of the system is to separately manage slices and radio resources, i.e., managing the accepted network slices by the network operator and allocating network resources via a slice context manager. Owing to the space limitation, we omit the detailed procedure of the network slicing implementation. The reader can refer to (13) for more details.

In the following sections, we will introduce the slice request acceptance optimization, the power allocation optimization, the iterative solution algorithm, and the fast slice request acceptance algorithm, respectively.

III. SLICE REQUEST ACCEPTANCE

For any given transmit power \(P\), this section considers the subproblem of (18) for optimizing slice request acceptance. Particularly, by introducing auxiliary variables \(\eta_i\) and \(\{a_{ijn}, \forall i \in I\}\), the slice request acceptance subproblem with fixed transmit power \(P\) can be formulated as

\[
\max_{A} \eta_i \\
\text{s.t.} \ C1, C2, C3, C4, C5, C6
\]

(19)

\[
\sum_{i \in I} a_{ijn} \leq 2, \forall j \in J, n \in \mathcal{N} \\
\sum_{j \in J} \sum_{n \in \mathcal{N}} a_{ijn} r_{ijn} \geq \eta_i, \forall i \in I \\
\eta_i \geq R_{i}^{\min}, \forall i \in I \\
\eta_i \geq \eta_{R}, \forall i \in I \\
\sum_{i \in I} \sum_{n \in \mathcal{N}} a_{ijn} r_{ijn} \leq C_j^{\max}, \forall j \in J \\
a_{ijn} \in \{0, 1\}, \forall i \in I, j \in J, n \in \mathcal{N}
\]

However, the above problem (19) is challenging to solve since the achievable data rate \(r_{ijn}\) is not a certain fixed value even with fixed transmit power \(P\). From (6)-(9), it can be observed that \(r_{ijn}\) takes different values according to whether UE \(i\) is a primary UE or a secondary UE in \(\mathcal{SC}_{jn}\). Therefore, the value of \(r_{ijn}\) depends on slice request \(A\) with fixed transmit power \(P\). According to this key observation, we propose a two-stage optimization method to decompose the problem (19) into two integer linear programming (ILP) problems, both of which can be solved efficiently by existing optimization tools such as MOSEK (15) or CVX (16). The detailed procedures are described as follows.

A. Primary Slice Request Acceptance

For the given transmit power \(P\), this subsection considers the primary acceptance of slice requests. Particularly, we assume that each slice \(\mathcal{SC}_{jn}\) can provide service to at most one UE at the primary acceptance stage. Therefore, each UE \(i\) can be regarded as a primary UE in \(\mathcal{SC}_{jn}\) when calculating \(r_{ijn}\) at this stage, and thus we can formulate the primary slice request acceptance subproblem as the following ILP problem

\[
\max_{A} \eta_i \\
\text{s.t.} \ C1, C2, C3, C4, C5, C6
\]

(20)

\[
\sum_{i \in I} a_{ijn} \leq 1, \forall j \in J, n \in \mathcal{N} \\
\sum_{j \in J} \sum_{n \in \mathcal{N}} a_{ijn} p_{ijn} \geq \eta_i, \forall i \in I \\
\eta_i \geq R_{i}^{\min}, \forall i \in I \\
\eta_i \geq \eta_{R}, \forall i \in I \\
\sum_{i \in I} \sum_{n \in \mathcal{N}} a_{ijn} r_{ijn} \leq C_j^{\max}, \forall j \in J \\
a_{ijn} \in \{0, 1\}, \forall i \in I, j \in J, n \in \mathcal{N}
\]

where, \(r_{ijn}^p = \frac{W}{N} \log_2 \left( 1 + \frac{p_{1,j,n} h_{ijn}}{k_{ij} n + p_{n} h_{ijn} + \sigma_n^2} \right)\).

We denote \(\{a_{ijn}^p\}\) as the solution of (20) and Let \(\mathcal{S}_p = \{(i, j, n)|a_{ijn}^p = 1\}\) denote the set of the accepted slice requests determined at the primary acceptance stage. Besides, we let \(R_j^p = \sum_{j \in J} \sum_{n \in \mathcal{N}} a_{ijn}^p r_{ijn}^p\) and\( C_j^p = \sum_{i \in I} \sum_{n \in \mathcal{N}} a_{ijn}^p r_{ijn}^p\) represent the achievable data rate of UE \(i\) and the total data rate of access to UBS \(j\) at the primary acceptance stage, respectively.

B. Secondary Slice Request Acceptance

For the given transmit power \(P\) and the primary slice request \(\{a_{ijn}^p\}\), this subsection considers the secondary acceptance of slice requests. Similarly, we assume that each slice \(\mathcal{SC}_{jn}\) can provide service to at most one UE at the secondary acceptance stage. Particularly, based on the primary acceptance stage, we first calculate the achievable data rate \(r_{ijn}\) of UE \(i\) in \(\mathcal{SC}_{jn}\) at the secondary acceptance stage in two cases.

1) For each slice \(\mathcal{SC}_{jn}\), if the slice \(\mathcal{SC}_{jn}\) does not provide service to any UE at the primary acceptance stage, then at the secondary acceptance stage, \(U_j\) in \(\mathcal{SC}_{jn}\) is a primary UE in \(\mathcal{SC}_{jn}\). Let \(\mathcal{S}_u = \{(i, j, n)|\sum_{i \in I} a_{ijn}^p = 0\}\) represent the index set of slice requests corresponding to this case. Thus, for each \((i, j, n)\in \mathcal{S}_u\), the achievable data rate \(r_{ijn}^u\) at the secondary acceptance stage is

\[
r_{ijn}^u = \frac{W}{N} \log_2 \left( 1 + \sum_{k \neq j, k \in J} \frac{p_{1,j,n} h_{ijn}}{p_{k,n} h_{ijn} + p_{n} h_{ijn}^M + \frac{1}{\sigma_n^2}} \right)
\]

(21)
2) If $SC_{jn}$ provides service to a UE at the primary acceptance stage, then we denote this UE as $i_p$, i.e., $(i_p, j, n) \in S_{ps}$. Note that when the slice $SC_{jn}$ provide service to UE $i_p$ and UE $i$ at the primary and secondary acceptance stages, respectively, one of the two UEs is a primary UE and the other is a secondary UE in $SC_{jn}$, which is determined by the relative relationship of these two UEs' channel gains. If $h_{ijn} > h_{ipjn}$, then UE $i$ and UE $i_p$ are the primary UE and the secondary UE in $SC_{jn}$, respectively, and the achievable data rate $r_{ijn}^p$ of UE $i_p$ at the primary acceptance stage can be calculated by

$$r_{ijn}^p = \frac{W}{N} \log_2 \left( 1 + \sum_{k \not= j, \in J} p_{k,n} h_{ikn} + p_{n} M_n h_{in}^M + \sigma_n^2 \right)$$

(22)

and the change of the achievable data rate $r_{ijn}^p$ of UE $i_p$ at the primary acceptance stage can be calculated by

$$\Delta r_{ijn}^p = \frac{W}{N} \log_2 \left( 1 + \sum_{k \not= j, \in J} p_{k,n} h_{ikn} + p_{n} M_n h_{in}^M + \sigma_n^2 \right) - \frac{W}{N} \log_2 \left( 1 + \sum_{k \not= j, \in J} p_{k,n} h_{ikn} + p_{n} M_n h_{in}^M + \sigma_n^2 \right)$$

(23)

If $h_{ijn} < h_{ipjn}$, then UE $i_p$ and UE $i$ are the primary UE and the secondary UE in $SC_{jn}$, respectively. Let $S_{s2} = \{(i, j, n)| h_{ijn} < h_{ipjn}, (i_p, j, n) \in S_{ps}\}$ represent the index set of slice requests corresponding to this case. Thus, for each $(i, j, n) \in S_{s2}$, the achievable data rate $r_{ijn}^o$ at the secondary acceptance stage is

$$r_{ijn}^o = \frac{W}{N} \log_2 \left( 1 + \sum_{k \not= j, \in J} p_{k,n} h_{ikn} + p_{n} M_n h_{in}^M + \sigma_n^2 \right)$$

(24)

Besides, we let $S_2 = S_u \cup S_{s1} \cup S_{s2}$ represent the index set of feasible slice requests at the secondary acceptance stage, and let $S_{o2}^N(i) = \{(i, j, n)| (i, j, n) \in S_{ps}\}$, $I_{o2}(j, n) = \{(i, j, n)| (i, j, n) \in S_{s2}\}$ and $S_{o2}^N(j) = \{(i, j, n)| (i, j, n) \in S_{s2}\}$. Note that for each $(i, j, n) \in I$, the change of the achievable data rate of UE $i$ at the primary acceptance stage can be calculated by

$$\Delta R_i^p = \sum_{(j,n) \in S_{o2}^N(i)} \sum_{k \in I_{o2}(j,n)} a_{k,n} \Delta r_{ijn}^p$$

(25)

For each $j \in J$, the change of the total data rate of access to UBS $j$ at the primary acceptance stage can be calculated by

$$\Delta C_j^p = \sum_{(i,n) \in S_{o2}^N(j)} a_{ijn} \Delta r_{ijn}^p$$

(26)

Therefore, we can formulate the secondary slice request acceptance subproblem as the following ILP problem

$$\begin{align*}
\text{max} & \quad \eta_R \\
\text{s.t.} & \quad \sum_{j \in J} a_{ijn} \leq 1, \forall j \in J, n \in N \\
& \quad \sum_{j \in J} \sum_{n \in N} a_{ijn} r_{ijn}^2 + \Delta R_i^p + R_i^p \geq \eta_i, \forall i \in I \\
& \quad \eta_i - \eta_R \geq 0, \forall i \in I \\
& \quad \sum_{i \in I} \sum_{n \in N} a_{ijn} r_{ijn}^2 + \Delta C_j^p + C_j^p \leq C_j^{max}, \forall j \in J \\
& \quad a_{ijn} \in \{0, 1\}, \forall (i, j, n) \in S_2 \\
& \quad a_{ijn} = 0, \forall (i, j, n) \notin S_2
\end{align*}$$

(27)

We denote $\{a_{ijn}^*\}$ as the solution of (27), and then the solution of (19) can be approximated as $a_{ijn} = \{a_{ijn}^*, a_{ijn}^o\}$.

IV. POWER ALLOCATION

For any given slice request $A$, this section consider the sub-problem of (18) for optimizing power allocation. Particularly, we first determine the accepted slice requests according to the given slice request $A$, and then let $S_u = \{(i, j, n)| a_{ijn} = 1\}$ and $I_{o2}(j, n) = \{i| a_{ijn} = 1\}$ represent the index set of the accepted slice requests and the index set of UEs in slice $SC_{jn}$, respectively. Then we divide the accepted slice requests into two categories. One category of accepted slice requests are that UE $i$ is a secondary UE in $SC_{jn}$, and we let $S_{ap} = S_u \setminus S_{as}$ represent the index set of such slice requests. Besides, we let $S_{ap2}^N(i) = \{(j, n)| (i, j, n) \in S_{ap}\}$, $S_{as2}^N(i) = \{(j, n)| (i, j, n) \in S_{as}\}$, $S_{ap2}^N(j) = \{(i, j, n)| (i, j, n) \in S_{ap}\}$, and $S_{as2}^N(j) = \{(i, j, n)| (i, j, n) \in S_{as}\}$. Based on the above sets and by introducing auxiliary variables $\eta_R$, $\eta_P$ and $\{\eta_i, \forall i \in I\}$, the power allocation subproblem with fixed slice request $A$ can be formulated as

$$\begin{align*}
\text{max} & \quad \eta_R - \eta_{EE} \eta_P \\
\text{s.t.} & \quad \sum_{(j,n) \in S_{ap2}^N(i)} \sum_{(j,n) \in S_{as2}^N(i)} a_{ijn} \eta_i \geq \eta_i, \forall i \in I \\
& \quad \eta_i - \eta_R \geq 0, \forall i \in I \\
& \quad \sum_{(j,n) \in S_{ap2}^N(j)} + \sum_{(j,n) \in S_{as2}^N(j)} r_{ijn}^p \geq \eta_R, \forall j \in J \\
& \quad \sum_{n \in N} (p_{1,jn} + p_{2,jn}) + p_j^p \leq \eta_p, \forall j \in J \\
& \quad \sum_{n \in N} (p_{1,jn} + p_{2,jn}) + p_j^p \leq \eta_p, \forall j \in J \\
& \quad p_{1,jn} \geq 0, p_{2,jn} \geq 0, \forall j \in J, n \in N
\end{align*}$$

(28a)
where,
\[ r_{ijn}^p(\mathcal{P}) = \frac{W}{N} \log_2 \left( 1 + \frac{p_{1,ijn}h_{ijn} + \sum_{k \neq j, k \in J} (p_{1,ijn} + p_{2,ijn})h_{ikn} + p_n^M h_{in} + \sigma_n^2}{1 + \sum_{k \neq j, k \in J} (p_{1,ijn} + p_{2,ijn})h_{ikn} + p_n^M h_{in} + \sigma_n^2} \right), \]  
(30)
\[ r_{ijn}^s(\mathcal{P}) = \frac{W}{N} \log_2 \left( 1 + \frac{p_{2,ijn}h_{ijn} + \sum_{k \neq j, k \in J} (p_{1,ijn} + p_{2,ijn})h_{ikn} + \sigma_n^2}{1 + \sum_{k \neq j, k \in J} (p_{1,ijn} + p_{2,ijn})h_{ikn} + \sigma_n^2} \right). \]  
(31)

Note that in (29b) and (29g), \( r_{ijn}^p(\mathcal{P}) \) and \( r_{ijn}^s(\mathcal{P}) \) are neither convex nor concave with respect to \( \mathcal{P} \). Thus, (29b) and (29g) are not convex constraints, and the problem (29) is a non-convex optimization problem. To tackle the non-convexity of (29), we explore a successive convex approximate (SCA) approach \([17]\), where in each iteration, the original function is approximated as a more tractable function at a given point. Denote \( \mathcal{P}^{(r)} = \{ p_{1,ijn}^{(r)}, p_{2,ijn}^{(r)} \} \) as the given transmit power point in the \( (r + 1) \)-th iteration \( (r \geq 0) \). Next, we discuss how to transform (29) into a convex optimization problem via the SCA approach in detail. Note that we need to approximate the left-hand side of (29b) as a concave function and the left-hand side of (29g) as a convex function.

First, we study the approximation of \( r_{ijn}^p(\mathcal{P}) \) and \( r_{ijn}^s(\mathcal{P}) \). For \( r_{ijn}^p(\mathcal{P}) \), it can be written as a difference of two concave functions with respect to \( \mathcal{P} \), i.e.,
\[ r_{ijn}^p(\mathcal{P}) = \tilde{r}_{ijn}^p(\mathcal{P}) - \bar{r}_{ijn}(\mathcal{P}) \]  
(32)
where,
\[ \tilde{r}_{ijn}^p(\mathcal{P}) = \frac{W}{N} \log_2 \left( p_{1,ijn}h_{ijn} + \sum_{k \neq j, k \in J} (p_{1,ijn} + p_{2,ijn})h_{ikn} + p_n^M h_{in} + \sigma_n^2 \right) \]  
(33)
\[ \bar{r}_{ijn}(\mathcal{P}) = \frac{W}{N} \log_2 \left( \sum_{k \neq j, k \in J} (p_{1,ijn} + p_{2,ijn})h_{ikn} + p_n^M h_{in} + \sigma_n^2 \right). \]  
(34)

It can prove that any concave function is globally upper-bounded by its first-order Taylor expansion at any point \([17]\). Therefore, we have the following upper bound of \( \tilde{r}_{ijn}^p(\mathcal{P}) \) and \( \bar{r}_{ijn}(\mathcal{P}) \) at the given transmit power point \( \mathcal{P}^{(r)} \)
\[ \tilde{r}_{ijn}^p(\mathcal{P}) \leq B_{ijn}^{(r)} + D_{ijn}^{(r)} \left( h_{ijn}(p_{1,ijn} - p_{1,ijn}^{(r)}) + \sum_{k \neq j, k \in J} h_{ikn}(p_{1,ijn} + p_{2,ijn} - p_{1,ijn}^{(r)} - p_{2,ijn}^{(r)}) \right) \]  
\[ = \tilde{r}_{ijn}(\mathcal{P}) \]  
(35)
\[ \bar{r}_{ijn}(\mathcal{P}) \leq F_{ijn}^{(r)} \left( \sum_{k \neq j, k \in J} h_{ikn}(p_{1,ijn} + p_{2,ijn} - p_{1,ijn}^{(r)} - p_{2,ijn}^{(r)}) \right) \]  
\[ = \bar{r}_{ijn}(\mathcal{P}) \]  
(36)
where,
\[ B_{ijn}^{(r)} = \frac{W}{N} \log_2 \left( p_{1,ijn}h_{ijn} + \sum_{k \neq j, k \in J} (p_{1,ijn} + p_{2,ijn})h_{ikn} + p_n^M h_{in} + \sigma_n^2 \right) \]  
(37)
\[ D_{ijn}^{(r)} = \frac{\log_2(e)W/N}{p_{1,ijn}h_{ijn} + \sum_{k \neq j, k \in J} (p_{1,ijn} + p_{2,ijn})h_{ikn} + p_n^M h_{in} + \sigma_n^2}. \]  
(38)

Similarly, \( r_{ijn}^s(\mathcal{P}) \) can be written as a difference of two concave functions with respect to \( \mathcal{P} \), i.e.,
\[ r_{ijn}^s(\mathcal{P}) = \tilde{r}_{ijn}^s(\mathcal{P}) - \bar{r}_{ijn}(\mathcal{P}) \]  
(41)
Similarly, by substituting \(35\) into \(32\) and substituting \(43\) into \(41\), for all \(j \in J\), we obtain the upper bound of the left-hand side of the constraint \((29c)\) as
\[
\sum_{(i,n) \in S_{I}^{p}(j)} r_{i,jn}^{p}(P) + \sum_{(i,n) \in S_{I}^{s}(j)} r_{i,jn}^{s}(P) \leq \sum_{(i,n) \in S_{I}^{p}(j)} (r_{i,jn}^{p}(P) - r_{i,jn}^{p}(P)) + \sum_{(i,n) \in S_{I}^{s}(j)} (r_{i,jn}^{s}(P) - r_{i,jn}^{s}(P)) \leq \sum_{(i,n) \in S_{I}^{p}(j)} (r_{i,jn}^{p}(P) - r_{i,jn}^{p}(P)) + \sum_{(i,n) \in S_{I}^{s}(j)} (r_{i,jn}^{s}(P) - r_{i,jn}^{s}(P)) \leq C_{j}^{\text{max}}, \forall j \in J
\]

Therefore, with any given transmit power point \(P(r) = \{p_{1,jn}^{(r)}, p_{2,jn}^{(r)}\}\), the problem \((29)\) can be approximated as the following form by referring to \((48)\) and \((49)\)
\[
\begin{align*}
\max_{P} \quad & \eta_{R} - \eta_{\text{EE}} \eta_{P} \\
\text{s.t.} \quad & \sum_{(i,n) \in S_{I}^{p}(j)} (r_{i,jn}^{p}(P) - r_{i,jn}^{p}(P)) + \sum_{(i,n) \in S_{I}^{s}(j)} (r_{i,jn}^{s}(P) - r_{i,jn}^{s}(P)) \geq \eta_{h}, \forall i \in I \\
& \eta_{h} \geq R_{i}^{\text{min}}, \forall i \in I \\
& \eta_{h} \geq \eta_{R}, \forall i \in I \\
& \sum_{(i,n) \in S_{I}^{p}(j)} (r_{i,jn}^{p}(P) - r_{i,jn}^{p}(P)) + \sum_{(i,n) \in S_{I}^{s}(j)} (r_{i,jn}^{s}(P) - r_{i,jn}^{s}(P)) \leq C_{j}^{\text{max}}, \forall j \in J \\
& \sum_{n \in N} (p_{1,jn} + p_{2,jn}) + p_{j}^{c} \leq \eta_{p}, \forall j \in J \\
& \sum_{n \in N} (p_{1,jn} + p_{2,jn}) + p_{j}^{c} \leq p_{j}^{\text{max}}, \forall j \in J \\
& p_{1,jn} \geq 0, p_{2,jn} \geq 0, \forall j \in J, n \in N
\end{align*}
\]

Since the left-hand sides of the constraints \((50b)\) and \((50b)\) are concave and convex with respect to \(P\), respectively, \((50b)\) and \((50b)\) are convex constraints. Therefore, the problem \((50)\) is a convex optimization problem, which can be efficiently solved by existing optimization tools such as MOSEK \(15\) or CVX \(16\).

Note that the inequalities \((48)\) and \((49)\) indicate that any feasible solution of the problem \((50)\) is also feasible for the problem \((29)\), but the reverse is not true in general. Therefore, the optimal objective value obtained by solving \((50)\) is the lower bound of that of \((29)\).

V. PROBLEM SOLUTION

In this section, we introduce two different solutions in the light of different requirements (e.g., energy efficiency and running time) for the problem \((18)\). The details are described in the following subsections.

A. Iterative Acceptance and Power Optimization

Based on the results in Section III and Section IV, the iterative acceptance and power optimization (IAPO) algorithm are proposed for the problem \((18)\) by alternately optimize slice request and transmit power. The main steps of the algorithm are summarized in Algorithm \(1\). For convenience of description, we let \(\eta_{R}(A, P) = \min_{j \in J} R_{i}(A, P), \eta_{P}(P) = \max_{i \in I} p_{i}(P)\), and \(\eta_{\text{EE}}(A, P) = \eta_{R}(A, P) / \eta_{P}(P)\). Note that for the initial power \(P^{(0)}\), there may exist no feasible solutions of \((20)\) and \((27)\). Therefore, the IAPO algorithm does not consider the QoS requirement constraint and the capacity constraint when performing the slice request acceptance optimization in the first iteration (i.e., \(r = 0\)).

It can prove that the convergence of the IAPO algorithm can always be guaranteed. Due to the space limitation, we omit the theoretical analysis of the convergence of the IAPO algorithm; yet, we will verify that the IAPO algorithm is convergent through the simulation. However, the IAPO algorithm cannot guarantee optimality because an approximated acceptance optimization method is developed and the exploration of the SCA approach and the iterative optimization method may lead to converging to a locally optimal solution \(17\).

Algorithm 1: Iterative Acceptance and Power Optimization

1. Initialize \(P^{(0)}\), and let \(r = 0\).
2. repeat
3. Slice request acceptance optimization:
4. For given \(P^{(r)}\), obtain the optimal solution by solving \((20)\) and \((27)\), and denote the optimal solution by \(A^{(r+1)}\).
5. if \(r > 0\) then
6. if \(\eta_{R}(A^{(r+1)}, P^{(r)}) < \eta_{R}(A^{(r)}, P^{(r)})\) then
7. Set \(A^{(r+1)} = A^{(r)}\).
8. end if
9. end if
10. Power allocation optimization:
11. For given \(P^{(r)}\) and \(A^{(r+1)}\), calculate \(\eta_{\text{EE}} = \eta_{\text{EE}}(A^{(r+1)}, P^{(r)})\), obtain the optimal solution by solving \((50)\), and denote the optimal solution by \(P^{(r+1)}\).
12. Update \(r = r + 1\).
13. until Convergence or \(r \geq r_{\text{max}}\).

B. Fast Acceptance and Iterative Power Optimization

In the IAPO algorithm, the slice request acceptance optimization is required for each iteration, which consists of two ILP problems. When running time becomes the primary consideration, it may be necessary to design a fast slice request acceptance algorithm that loses some energy efficiency. Due to this motivation and based on the result in Section IV, we develop a fast acceptance and iterative power optimization (FAIPO) algorithm.

As shown in Algorithm \(2\), we consider that each UE \(i\) is served by the nearest UBS \(j\) to achieve better channel gain. The index set of UEs served by UBS \(j\) is denoted by \(I_{fa}(j)\). Then we can separately implement slice request acceptance for the UEs in each set \(I_{fa}(j)\). Since the goal is to maximize the minimum achievable data rate among all UEs, we consider implementing slice request acceptance preferentially for UEs with farther distance (i.e., worse channel gain) from UBS \(j\).

For each \(j \in J\), we first sort the UEs in the set \(I_{fa}(j)\) in the
descending order of the distance from UBS \( j \), and obtain the corresponding sorted set \( \mathcal{T}^d_{fa}(j) \). Then for each UE \( i \in \mathcal{T}^d_{fa}(j) \), we sequentially calculate the set \( \mathcal{N}_{fs}(i) = \{ n | \sum_{k \in \mathcal{I}_{fa}(i)} a_{k,j,n} < 2, a_{ij,n} = 0 \} \) and judge whether \( \mathcal{N}_{fs}(i) \) is empty or not. If not empty, choose a \( n' \) from the set \( \mathcal{N}_{fs}(i) \) randomly, and let \( a_{ij,n'} = 1 \). When the sets \( \mathcal{N}_{fs}(i) \) are empty for all \( i \in \mathcal{T}^d_{fa}(j) \), the slice request acceptance of UBS \( j \) is terminated. We denote the solution of the fast slice request acceptance as \( \mathcal{A}^f = \{ a_{ij,n}, \forall i, j, n \} \). Next, with the fixed \( \mathcal{A}^f \), we iteratively optimize transmit power by solving (50). Similar to the IAPO algorithm, the convergence of the FAIPO algorithm can be guaranteed, but the optimality cannot be guaranteed. Due to the space limitation, we omit the theoretical analysis of the convergence of the FAIPO algorithm; yet, we will verify that the FAIPO algorithm is convergent through the simulation.

**Algorithm 2** Fast Acceptance and Iterative Power Optimization

1: Fast Slice Request Acceptance:
2: Initialize \( a_{ij,n} = 0 \) for all \( i \in \mathcal{I}, j \in \mathcal{J}, n \in \mathcal{N} \) and \( \mathcal{N}_{fs}(i) = \mathcal{N} \) for all \( i \in \mathcal{I} \).
3: Initialize sets \( \mathcal{T}_{fa}(j) \) for all \( j \in \mathcal{J} \) according to location information.
4: for \( j \in \mathcal{J} \) do
5: Initialize the set \( \mathcal{T}^d_{fa}(j) \) in the descending order of the distance from UBS \( j \).
6: while \( \mathcal{N}_{fs}(i) \neq \phi \) for any \( i \in \mathcal{T}^d_{fa}(j) \) do
7: for \( i \in \mathcal{T}^d_{fa}(j) \) do
8: Calculate \( \mathcal{N}_{fs}(i) = \{ n | \sum_{k \in \mathcal{I}_{fa}(i)} a_{k,j,n} < 2, a_{ij,n} = 0 \} \).
9: if \( \mathcal{N}_{fs}(i) \neq \phi \) then
10: Choose \( n' \in \mathcal{N}_{fs}(i) \) randomly, and set \( a_{ij,n'} = 1 \).
11: end if
12: end for
13: end while
15: Obtain the solution \( \mathcal{A}^f = \{ a_{ij,n} \} \).
16: Power allocation optimization:
17: Initialize \( \mathcal{P}^{(0)} \), and let \( r = 0 \).
18: repeat
19: With the fixed \( \mathcal{A}^m \), for given \( \mathcal{P}^{(r)} \), calculate \( \eta_{EE} = \eta_{EE}(\mathcal{A}^m, \mathcal{P}^{(r)}) \), obtain the optimal solution by solving (50), and denote the optimal solution by \( \mathcal{P}^{(r+1)} \).
20: Update \( r = r + 1 \).
21: until Convergence or \( r \geq r_{max} \).

VI. SIMULATION RESULTS

In this section, we conduct extensive simulations to validate the effectiveness of the proposed IAPO and FAIPO algorithms. Particularly, subsection A presents comparison algorithms and the parameter setting. Subsection B collects and analyzes the simulation results.

A. Comparison Algorithms and Parameter Setting

In this simulation, we compare the proposed IAPO and FAIPO algorithms with two benchmark algorithms: the slice request acceptance optimization-only (SRAAO) algorithm, and the iterative acceptance and power optimization with OMA (IAPO-OMA) algorithm. The brief descriptions of the above two comparison algorithms are as follows:

- SRAAO algorithm: Allocate transmit power according to the initial power \( \mathcal{P}^{(0)} \), and only optimize slice request by solving (20) and (27).
- IAPO-OMA algorithm: This algorithm is similar to the IAPO algorithm, except that it considers an OMA-based communication scenario where each slice can provide service to at most one UE.

The parameter setting of the simulation is summarized as follows: we set the size of the considered geographic area is a circle of radius \( R_u \). The MBS is located at the center (0,0), and the UEs and UBSs are uniformly distributed in the annulus \( (R_l, R_u) \). We set \( R_u = 500 \) m and \( R_l = 250 \) m. For each UE \( i \in \mathcal{I} \), the minimum required data rate \( R_{t_{\min}} \) is subject to a uniform distribution \( U(R_{t_{\min}}^{\min}, R_{t_{\min}}^{\max}) \), and we set \( R_{t_{\min}}^{\min} = 1 \) Mb/s, \( R_{t_{\min}}^{\max} = 2 \) Mb/s. For each UBS \( j \in \mathcal{J} \), we set \( p_{j_{\min}} = 100\) mW (20 dBm), \( p_{j_{\max}} = 251 \) mW (24 dBm), and \( c_{j_{\max}} = 100 \) Mb/s. More simulation parameters are listed in Table I.

| Parameters | Value | Parameters | Value |
|------------|-------|------------|-------|
| \( H \)    | 100 m | \( d_0 \)  | 1     |
| \( N \)    | 4     | \( g_{fa}^{(N)} \) | 1     |
| \( W \)    | 40 MHz| \( g_{fa}^{(N)} \) | 1     |
| \( \alpha_1\) | 0.88  | \( g_{fa}^{(N)} \) | 1     |
| \( \alpha_2\) | 0.43  | \( g_{fa}^{(N)} \) | 1     |
| \( p_{j_{\min}} \) | 251 mW (24 dBm) | \( \eta_{\min}^{S} \) | 0.1   |
| \( \sigma_{\min}^{2} \) | -85 dBm | \( \eta_{\min}^{S} \) | 21    |
| \( f_c \)  | 2.5 GHz| \( \eta \) | 3     |
| \( c \)    | \( 3 \times 10^5 \) m/s | \( r_{\max} \) | 1000  |

B. Performance Evaluation

The above four comparison algorithms all need to initialize the transmit power \( \mathcal{P}^{(0)} \). For the algorithms except the IAPO-OMA algorithm, we initialize \( \mathcal{P}^{(0)} \) to \( p_{j_{\min}}^{(0)} = \frac{p_{j_{\max}}^{(0)} - p_{j_{\min}}^{(0)}}{4N} \) for all \( j \in \mathcal{J}, n \in \mathcal{N} \). For the IAPO-OMA algorithm, we initialize \( \mathcal{P}^{(0)} \) to \( p_{j_{\min}}^{(0)} = \frac{p_{j_{\max}}^{(0)} - p_{j_{\min}}^{(0)}}{2N} \) for all \( j \in \mathcal{J}, n \in \mathcal{N} \).

In the simulation, we perform all comparison algorithms on fifty randomly generated data sets. For each comparison algorithm, it obtains a result on each data set, and the final result is the average of the fifty results. Moreover, we first study the convergence of the proposed IAPO and FAIPO algorithms. Then we consider the effect of the number of UEs \( I \) and the number of UBSs \( J \) on the energy efficiency \( \eta_{EE} \) for all comparison algorithms.

Fig. 1 illustrates the convergence behavior of the energy efficiency of the IAPO and FAIPO algorithms when \( I = 10 \) and \( J = 4 \). We observe that the energy efficiency of both IAPO and FAIPO algorithms increases monotonously with the increase of iteration index and quickly converges to certain
values. This result verifies the monotonic convergence of the proposed algorithms.

Fig. 2 illustrates the energy efficiency vs. the number of UEs with \( J = 4 \). Fig. 3 illustrates the energy efficiency vs. the number of UBSs with \( I = 10 \). From Fig. 2-3, the following observations can be obtained:

- The IAPO algorithm can always achieve the highest energy efficiency compared with the other three algorithms. Particularly, the average energy efficiency of the IAPO algorithm is 1.11, 1.78, 1.21 times of the FAIPO, SRAOO, and IAPO-OMA algorithms, respectively, when \( I = 10 \) and \( J = 4 \). Besides, the average energy efficiency of the FAIPO algorithm is second only to the IAPO algorithm in most cases.

- For all comparison algorithms, the energy efficiency decreases monotonously with the increase of the number of UEs. This is because when more UEs share the limited network resources, the average network resources allocated to each UE will be reduced, which leads to the reduction of the average achievable data rate of each UE.

- For all comparison algorithms, the energy efficiency increases monotonously with the increase of the number of UBSs. This is because more UBSs mean more assignable resources. However, more UAVs also result in higher operating costs. The trade-off between energy efficiency and operating costs is also a factor needed to consider.

**REFERENCES**

[1] M. Shafi, A. F. Molisch, P. J. Smith, T. Haustein, P. Zhu, P. De Silva, F. Tufvesson, A. Benjebbour, and G. Wunder, “5G: A tutorial overview of standards, trials, challenges, deployment, and practice,” IEEE journal on selected areas in communications, vol. 35, no. 6, pp. 1201–1221, 2017.

[2] “Description of network slicing concept version 1.0,” NGMN, Frankfurt, Germany, Tech. Rep., Jan 2016.

[3] P. Rost, C. Mannweiler, D. S. Michalopoulos, C. Sartori, V. Sciancalepore, N. Sastry, O. Holland, S. Tayade, B. Han, D. Bega et al., “Network slicing to enable scalability and flexibility in 5G mobile networks,” IEEE Communications Magazine, vol. 55, no. 5, pp. 72–79, 2017.

[4] H. Zhang, N. Liu, X. Chu, K. Long, A.-H. Aghvami, and V. C. Leung, “Network slicing based 5G and future mobile networks: mobility, resource management, and challenges,” IEEE Communications Magazine, vol. 55, no. 8, pp. 138–145, 2017.

[5] D. R. Albonda and J. Pérez-Romero, “An efficient RAN slicing strategy for a heterogeneous network with eMBB and V2X services,” IEEE Access, vol. 7, pp. 44 771–44 782, 2019.

[6] P. Popovski, K. F. Trillingsgaard, O. Simeone, and G. Durisi, “5G wireless network slicing for eMBB, URLLC, and mMTC: A communication-theoretic view,” IEEE Access, vol. 6, pp. 55 765–55 779, 2018.

[7] R. Kassab, O. Simeone, and P. Popovski, “Coexistence of URLLC and eMBB services in the C-RAN uplink: an information-theoretic study,” in 2018 IEEE Global Communications Conference (GLOBECOM). IEEE, 2018, pp. 1–6.
[8] G. K. Xilouris, M. C. Batistatos, G. E. Athanasiadou, G. Tsoulos, H. B. Pervaiz, and C. C. Zarakovitis, “UAV-assisted 5G network architecture with slicing and virtualization,” in *2018 IEEE Globecom Workshops (GC Wkshps)*. IEEE, 2018, pp. 1–7.

[9] I. Budhiraja, S. Tyagi, S. Tanwar, N. Kumar, and J. J. Rodrigues, “Tactile internet for smart communities in 5G: An insight for NOMA-based solutions,” *IEEE Transactions on Industrial Informatics*, vol. 15, no. 5, pp. 3104–3112, 2019.

[10] A. Al-Hourani, S. Kandeepan, and S. Lardner, “Optimal LAP altitude for maximum coverage,” *IEEE Wireless Communications Letters*, vol. 3, no. 6, pp. 569–572, 2014.

[11] R. Mudumbai, S. Singh, and U. Madhow, “Medium access control for 60 GHz outdoor mesh networks with highly directional links,” in *IEEE INFOCOM 2009*. IEEE, 2009, pp. 2871–2875.

[12] H. Zhang, B. Wang, C. Jiang, K. Long, A. Nallanathan, V. C. Leung, and H. V. Poor, “Energy efficient dynamic resource optimization in NOMA system,” *IEEE Transactions on Wireless Communications*, vol. 17, no. 9, pp. 5671–5683, 2018.

[13] J. Lee and S. Leyffer, *Mixed integer nonlinear programming*. Springer Science & Business Media, 2011, vol. 154.

[14] X. Foukas, M. K. Marina, and K. Kontovasilis, “Orion: RAN slicing for a flexible and cost-effective multi-service mobile network architecture,” in *Proceedings of the 23rd annual international conference on mobile computing and networking*. ACM, 2017, pp. 127–140.

[15] MOSEK ApS, “MOSEK optimization toolbox for MATLAB 8.1.0.81,” https://docs.mosek.com/8.1/toolbox/index.html 2019.

[16] M. Grant and S. Boyd, “The CVX users’ guide release 2.1,” http://cvxr.com/cvx/doc/CVX.pdf 2018.

[17] S. Boyd and L. Vandenberghe, *Convex optimization*. Cambridge university press, 2004.