IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received August 23, 2017, accepted September 13, 2017, date of publication September 19, 2017,
date of current version October 12, 2017.

Digital Object Identifier 10.1109/ACCESS.2017.2754461

End-to-End Network Slicing in Virtualized OFDMA-Based Cloud Radio Access Networks

VU NGUYEN HA[1,2], (Member, IEEE), AND LONG BAO LE[2], (Senior Member, IEEE)
[1]École Polytechnique de Montréal, Montréal, QC H3T 1J4, Canada
[2]INRS, Université du Québec, Montréal, QC H5A 1K6, Canada

Corresponding author: Long Bao Le (long.le@emt.inrs.ca)

ABSTRACT We consider the resource allocation for the virtualized OFDMA uplink cloud radio access network (C-RAN), where multiple wireless operators (OPs) share the C-RAN infrastructure and resources owned by an infrastructure provider (InP). The resource allocation is designed through studying tightly coupled problems at two different levels. The upper-level problem aims at slicing the fronthaul capacity and cloud computing resources for all OPs to maximize the weighted profits of OPs and InP considering practical constraints on the fronthaul capacity and cloud computation resources. Moreover, the lower-level problems maximize individual OPs' sum rates by optimizing users' transmission rates and quantization bit allocation for the compressed I/Q baseband signals. We develop a two-stage algorithmic framework to address this two-level resource allocation design. In the first stage, we transform both upper-level and lower-level problems into corresponding problems by relaxing underlying discrete variables to the continuous ones. We show that these relaxed problems are convex and we develop fast algorithms to attain their optimal solutions. In the second stage, we propose two methods to round the optimal solution of the relaxed problems and achieve a final feasible solution for the original problem. Numerical studies confirm that the proposed algorithms outperform two greedy resource allocation algorithms and their achieved sum rates are very close to sum rate upper-bound obtained by solving relaxed problems. Moreover, we study the impacts of different parameters on the system sum rate, performance tradeoffs, and illustrate insights on a potential system operating point and resource provisioning issues.

INDEX TERMS Cloud radio access network, resource management, platform virtualization, computational efficiency.

I. INTRODUCTION

Next-generation wireless cellular systems are expected to provide significantly higher capacity in a cost-efficient manner to support the tremendous growth of wireless traffic and services [1], [2]. Some recent studies have indicated that the traditional model of single ownership of network architecture can be inefficient because the average load demand is usually much lower than the designed peak demand [1]–[3], [7], [8]. Therefore, advanced access techniques for C-RAN and wireless virtualization to support multiple OPs (also called *"wireless network virtualization (WNV)"*) have attracted a lot of attention from both industry and academia [1]–[8], [13].

By realizing various communications and processing functions in the cloud, C-RAN enables more efficient utilization of network resources, which results in better network throughput and reduced network deployment and operation costs. With WNV, multiple OPs can efficiently share various network resources such as radio spectrum, computation resources, backhaul/fronthaul capacity; hence, the capital expenditures (CAPEX) and operating expenses (OPEX) can be reduced significantly [8], [7]. To attain the potential benefits of the C-RAN and WNV technologies, one has to address many technical challenges [4], [8]. Resource allocation, which determines the allocation of centralized computation resource [5], [6], fronthaul capacity [9], radio spectrum and power allocation in C-RAN [4], and the slicing/allocation of infrastructure resources for different OPs to optimize desired design objectives in WNV are among the major research challenges [8].

A. RELATED WORKS

Recent literature on C-RAN and WNV has tackled some of these technical problems which are described in the following. In particular, the authors in [10] consider the spectrum sharing problem in a heterogeneous wireless network where small-cell base stations are optimally matched with OPs. In [11], the virtual resource slicing problem, which aims at maximizing the system utility as a function of achieved

cumulative rates and assigned resource slots while meeting the requirement of each slice is studied. The slicing problem for different OPs is also considered in [12] where a fairness-based dynamic resource allocation scheme is proposed. Recently, a novel energy-efficient resource allocation strategy is proposed for C-RAN virtualization with optical fronthaul [13] where the matching problem between cells, users, baseband units (BBUs), and a number of wavelengths in optical links is investigated. Two heuristic methods, namely static and dynamic ones, are presented to solve this problem. However, these works do not study potential strategies to share computation resources among OPs and efficiently utilize the fronthaul transport network.

Resource allocation for efficient utilization of the C-RAN fronthaul capacity has been studied in [14]–[19]. The works [14]–[16] address the precoding problem for remote radio heads (RRHs) to minimize the total power consumption where [15], [16] consider the downlink while [14] addresses both downlink and uplink communications. In [14] and [15], the authors optimize the utilization of fronthaul links accounting for the power consumption of fronthaul links while we consider the downlink joint transmission design for RRHs to minimize the total transmission power under the fronthaul capacity constraint in [16]. Moreover, the data compression issue for reduction of fronthaul capacity utilization has been addressed in [17] and [18] where [17] focuses on minimizing the amount of data transmitted over the fronthaul transport network while [18] jointly designs signal quantization and power control to maximize the system sum rate. The authors in [19] study the joint fronthaul signal compression and signal recovery in the uplink C-RAN; however, this work does not consider transmission design aspects.

A few existing works have considered the cloud computation complexity for processing users' data, which is a major challenge in large-scale C-RAN deployment. The works [20]–[23] study the optimization of C-RAN computational resources. Specifically, [20] models the computation complexity in downlink communication considering computational requirement of electrical circuits for processing the base-band signals where the computation complexity is a non-linear function of different parameters including the number of antennas, modulation bits corresponding to FFT blocks, the coding rate, and the number of data streams. The work [21] then applies this model to quantify the C-RAN energy-efficiency benefits. The authors in [22] propose a different computation complexity model for the uplink C-RAN system, which accounts mainly for the decoding process of turbo-encoded uplink data streams of all users. This work is motivated by the fact that the power utilized in the decoding process is much higher than that in the encoding process. Based on this model, the authors address the rate allocation for uplink transmissions to maximize the system sum rate considering the cloud computational capacity constraint [23]. This work, however, assumes unlimited fronthaul capacity and does not address the resource allocation. All these papers have not studied the problem of end-to-end network slicing of radio, fronthaul, and cloud computational resources for C-RAN, which is studied in our current paper.

B. RESEARCH CONTRIBUTIONS

To the best of our knowledge, uplink C-RAN design considering constraints on limited fronthaul capacity and cloud computation resources has not been studied except our previous work [25], which, however, relies on an over-simplified computational complexity model and does not consider the C-RAN virtualization issue. This paper aims to fill this gap in the existing C-RAN literature where the virtualized OFDMA-based uplink C-RAN design is addressed. In particular, we make the following contributions.

- We consider the virtualized resource allocation design for the uplink OFDMA-based C-RAN where an InP leases its resources to different OPs to support their mobile users. This design boils down to solving two-level coupled problems where the upper-level problem aims to determine the resource slicing solution for the computation resource and fronthaul capacity to maximize the weighted profits of the InP and OPs while the lower-level problems model the resource allocation of individual OPs for their users. Specifically, each lower-level problem must be solved by the corresponding OP to maximize the sum rate of its users by determining the optimal rate and quantization bit allocations for the resource slicing solution given by the upper-level problem.
- We develop a two-stage solution framework to solve the tightly coupled two-level problems. In stage one, we study the relaxed problems of the corresponding upper-level and lower-level problems to deal with the discrete rate and quantization bit allocation variables. We show that the relaxed problems in both levels are convex and we describe how to solve these problems optimally. Specifically, by employing the dual-based approach, we derive the optimal rate and quantization bit allocation solution for a given dual point, which enables us to develop a fast algorithm to solve the relaxed lower-level (RLL) problem optimally. Importantly, the optimal solution of the RLL problem is employed to tackle the relaxed upper-level (RUL) problem. In the second stage, we propose two rounding methods which are applied to the optimal solutions of the relaxed problems to attain a feasible solution for the original problem.
- For performance evaluation of the developed algorithms, we also describe two greedy resource allocation algorithms. Extensive numerical studies are conducted where we examine the convergence and efficiency of the proposed algorithms as well as the impacts of different system parameters on the system sum rate. In addition, we also study different tradeoffs, which characterize the relations of available resources, resource provisioning, and the corresponding benefits achieved by the InP and OPs.

Some preliminary results of this work have been published in [24]. However, the work in [24] only studies the optimization of transmission rate and quantization bit allocation, which corresponds to the lower-level problem investigated in this journal paper. Specifically, the current journal paper studies the more complicated and coupled problems in two different levels which optimize the slicing of fronthaul and computation resources as well as transmission rate and quantization bit allocation. Therefore, the algorithm design and development in this journal version are more extensive compared to those in the conference work [24]. Moreover, numerical results in this journal version are different from those in the conference work [24] since the two-level end-to-end network slicing design in the current paper is more general than the one-level resource allocation engineering in [24].

The remaining of this paper is organized as follows. We describe the system model and formulations of two-level problems in Section II. In Section III, we characterize the convexity of these relaxed problems. Then, we develop the optimal algorithms to solve these relaxed problems and present the proposed rounding methods in Section IV. The greedy resource allocation algorithm is presented in Section V. Numerical results are presented in Section VI followed by conclusion in Section VII. Key notations used in the paper are summarized in Table 2 given at the end of the paper.

II. SYSTEM MODEL

We consider the C-RAN which consists of BBUs in the cloud, K RRHs, and the fronthaul transport network connecting RRHs to the cloud. The C-RAN is owned by an InP which leases network resources to O OPs to serve their own users.[1] The uplink OFDMA transmission based on the 3GPP LTE standard with full frequency reuse (frequency reuse factor of one) is assumed. Specifically, each cell utilizes the whole spectrum comprising S physical resource blocks (PRBs) where each PRB corresponds to 12 sub-carriers (180 kHz) in the frequency domain and a slot duration of $t_s = 0.5\ ms$, which is equivalent to 7 OFDM symbols [6]. We denote the set of all PRBs and OPs as $\mathcal{S}$ and Ω, respectively.

We assume that the PRB allocations to individual OPs in each cell have been predetermined by any existing algorithm.[2] Moreover, let $\mathcal{S}_k^o$ denote the set of PRBs assigned to OP o in cell k. We assume that $\cup_{o\in\Omega}\mathcal{S}_k^o = \mathcal{S}$ and $\mathcal{S}_k^o \cap \mathcal{S}_k^m = \varnothing$ for any two different OPs o and m. Therefore, there is only inter-cell interference on a specific PRB if concurrent transmissions from different cells occur on the underlying PRB. We assume that each RRH upon receiving users' baseband signals of different OPs quantizes these signals and forwards them to the cloud for decoding. In the following, we refer to RRH k and its corresponding coverage area as cell k. We further assume that both RRHs and users are equipped with single antenna.

Let $x_k^{(s)} \in \mathbb{C}$ represent the baseband signal transmitted on PRB s in cell k and we assume that the signal $x_k^{(s)}$ has unit power. Then, the signal received at RRH k on PRB s can be written as

$$y_k^{(s)} = \sum_{j\in\mathcal{K}} h_{k,j}^{(s)}\sqrt{p_j^{(s)}}x_j^{(s)} + \eta_k^{(s)}, \tag{1}$$

where $\mathcal{K}$ denotes the set of cells, $p_j^{(s)}$ represents the transmission power corresponding to $x_j^{(s)}$, $h_{k,j}^{(s)}$ is the channel gain from the user assigned PRB s in cell j to RRH k, and $\eta_k^{(s)} \sim \mathcal{CN}\left(0, \sigma_k^{(s)2}\right)$ denotes the complex Gaussian thermal noise.

A. SIGNAL QUANTIZATION AND PROCESSING

We assume that all baseband signals $y_k^{(s)}$ must be quantized and then forwarded to the cloud for further processing and decoding, which are performed by the BBUs. Moreover, RRH k uses $b_k^{(s)}$ bits to quantize the real and imaginary parts of the received symbol $y_k^{(s)}$. Then, according to the results in [26], the quantization noise power can be approximated as

$$q_k^{(s)}(b_k^{(s)}) \simeq 2Q(y_k^{(s)})/2^{b_k^{(s)}}, \tag{2}$$

where $Q(y) = \left(\int_{-\infty}^{\infty} f(y)^{1/3}dy\right)^3/12$, and $f(y_k^{(s)})$ is the probability density function of both the real and imaginary parts of $y_{k,u}^{(s)}$. Assuming a Gaussian distribution of the signal to be quantized, we have [27]

$$q_k^{(s)}(b_k^{(s)}) \simeq \frac{\sqrt{3}\pi}{2^{2b_k^{(s)}+1}} Y_k^{(s)}, \tag{3}$$

where $Y_k^{(s)}$ is the power of received signal $y_k^{(s)}$, which is equal to

$$Y_k^{(s)} = \sum_{j\in\mathcal{K}} |h_{k,j}^{(s)}|^2 p_j^{(s)} + \sigma_k^{(s)2} = D_k^{(s)} + I_k^{(s)}, \tag{4}$$

where $D_k^{(s)} = |h_{k,k}^{(s)}|^2\ p_k^{(s)}$ and $I_k^{(s)} = \sum_{j\in\mathcal{K}/k} |h_{k,j}^{(s)}|^2\ p_j^{(s)} + \sigma_k^{(s)2}$. Then, the total number of quantization bits required by all users of OP o which are forwarded from RRH k to the cloud in one second (measured in bit-per-second (bps)) can be expressed as[3]

$$B_k^o = 2N_{\mathsf{RE}} \sum_{s\in\mathcal{S}_k^o} b_k^{(s)}, \tag{5}$$

[1] For the more general setting with multiple InPs, one must consider how different InPs compete or cooperate in providing fronthaul and computing resources for different OPs. The joint problem of InPs' competition/cooperation and OPs' resource allocation is certainly interesting but challenging, which is reserved for study in our future work.

[2] The proposed design for optimized fronthaul capacity and computation resource allocation can be realized for a given PRB solution, which can be obtained by any existing PRB allocation algorithm such as that in [12]. In general, the fronthaul capacity and cloud computing resource allocation can be optimized jointly with the PRB allocation; however, this joint allocation problem is more complex, which will be studied in our future work.

[3] The total number of quantization bits B_k^o for each OP o is normalized to one second; therefore, the number of resource elements N_{RE} is defined for each second. Note that the normalization of these quantities to one second is performed just for calculation while the number of quantization bits for individual users and subcarriers $b_k^{(s)}$ must be re-optimized once the wireless channel gains vary.

where N_{RE} is the number of resource elements (REs) in one second and one subchannel of a PRB (corresponding to 12 subcarriers). Then, N_{RE} can be calculated as $N_{\mathsf{RE}} = 12 \times 7/t_s$ [6] since each LTE slot spans an interval $t_s = 0.5\,ms$ with 7 symbols. Let $\mathbf{b}$ denote the vector whose elements represent the numbers of quantization bits allocated to all users in the network. For a given $\mathbf{b}$, the quantized version of $y_k^{(s)}$ can be written as

$$\tilde{y}_k^{(s)} = y_k^{(s)} + e_k^{(s)}, \tag{6}$$

where $e_k^{(s)}$ represents the quantization error for $y_k^{(s)}$, which has zero mean and variance $q_k^{(s)}(b_k^{(s)})$. The SINR of the signal corresponding to PRB s in cell k can be expressed as

$$\gamma_k^{(s)}(b_k^{(s)}) = \frac{D_k^{(s)}}{I_k^{(s)} + q_k^{(s)}(b_k^{(s)})} \simeq \frac{D_k^{(s)}}{I_k^{(s)} + \frac{\sqrt{3}\pi Y_k^{(s)}}{2^{2b_k^{(s)}+1}}}. \tag{7}$$

B. DECODING COMPUTATIONAL COMPLEXITY

We assume that a data rate $r_k^{(s)}$ (in *"bits per channel use (bits pcu)"*) on PRB s is chosen from a given discrete set with M_R different rates (i.e., different predetermined modulation and coding schemes) $\mathcal{R} = \{R_1, R_2, \ldots, R_{M_R}\}$. Moreover, the chosen rate is set smaller than the link capacity to ensure satisfactory communication reliability and manageable decoding complexity, i.e., $r_k^{(s)} \leq \log_2(1 + \gamma_k^{(s)}(b_k^{(s)}))$. We assume that the capacity-achieving turbo code is employed, then the computation effort required to successfully decode information bits depends on the number of turbo-iterations. According to the results [22], the required computation effort expressed in *"bit-iterations (bi)"* for decoding the transmitted signal on PRB s in cell k is a function of $\gamma_k^{(s)}(b_k^{(s)})$ and $r_k^{(s)}$ which can be expressed as[4]

$$\begin{aligned} C_k^{(s)} &= \chi_k^{(s)}(r_k^{(s)}, b_k^{(s)}) \\ &= A r_k^{(s)} \left[B - 2\log_2 \left(\log_2 \left(1 + \gamma_k^{(s)}(b_k^{(s)}) \right) - r_k^{(s)} \right) \right], \end{aligned} \tag{8}$$

where $A = 1/\log_2(\zeta - 1)$, $B = \log_2\left((\zeta - 2)/(\zeta T(\epsilon_{\mathsf{ch}}))\right)$, ζ is a parameter related to the connectivity of the decoder, $T(\epsilon_{\mathsf{ch}}) = -T'/\log_{10}(\epsilon_{\mathsf{ch}})$, T' is another model parameter, and ϵ_{ch} is the target computational outage probability. Note that the computational outage probability occurs when there is not sufficient computational resources to correctly decode the received signal. The set $\{T', \zeta\}$ can be selected by calibrating (8) with an actual turbo-decoder implementation or a message-passing decoder. Let $\mathbf{r}^o$ and $\mathbf{b}^o$ denote the vectors whose elements represent the data rates and numbers of quantization bits selected for all users from OP o, respectively. Then, the total computation effort required by the cloud to successfully decode the signals for all users from OP o (calculated in *"bi per second (bips)"*) can be expressed as [6]

$$C^o(\mathbf{r}^o, \mathbf{b}^o) = N_{\mathsf{RE}} \sum_{k \in \mathcal{K}} \sum_{s \in \mathcal{S}_k^o} C_k^{(s)}. \tag{9}$$

C. BI-LEVEL RESOURCE ALLOCATION FORMULATION

We now present the bi-level problem formulation that models the interactions among the C-RAN InP, the OPs, and mobile users. Specifically, the OPs must pay the InP to rent network resources, which are then utilized to provide services to the mobile users. Moreover, such problem formulation must account for limited cloud computational effort and fronthaul capacity. The upper-level problem aims to maximize the weighted sum profit of the InP and OPs through slicing the fronthaul and computational resources for different OPs while the lower-level problems for individual OPs optimize the rate and quantization bit allocation to achieve their maximum sum rates. By solving these two-level problems, we can obtain an equilibrium, which would be desirable for the InP and all OPs.

In the considered bi-level problem formulation, the performance measure of interest is the profits achieved by the InP and OPs which are modeled as follows. Let C^o and B_k^o denote the computational effort and the total quantization bits per second corresponding to RRH k that OP o requires from the C-RAN InP (i.e., the required amount of fronthaul capacity). Moreover, the prices corresponding to one unit of computational effort and fronthaul capacity are denoted as ψ^o $(¢/bips)$ and β_k^o $(¢/bps)$, respectively. We assume that the profit obtained by the InP is equal to the total payment from all OPs (i.e., we omit the operation cost since it simply adds a constant term to underlying optimization objective), which can be calculated as

$$G^{\mathsf{InP}} = \sum_{o \in \Omega} G_o^{\mathsf{InP}} = \sum_{o \in \Omega} \left(\psi^o C^o + \sum_{k \in \mathcal{K}} \beta_k^o B_k^o \right), \tag{10}$$

where G_o^{InP} is the payment from OP o to the InP for using the amount of cloud computational resource C^o and the amount of fronthaul capacity B_k^o for RRH k, i.e., $G_o^{\mathsf{InP}} = \psi^o C^o + \sum_{k \in \mathcal{K}} \beta_k^o B_k^o$. For convenience, let us define $\mathbf{B}^o = [B_1^o, \ldots, B_K^o]$.

Let us define $R_o(C^o, \mathbf{B}^o) = \sum_{k \in \mathcal{K}} \sum_{s \in \mathcal{S}_k^o} r_k^{(s)}$ as the total rate of all users from OP o, which is achieved by using the amount of computational resource C^o and the amount of fronthaul capacity $\mathbf{B}^o$. Then, the profit achieved by OP o, which is equal to revenue minus cost, can be expressed as

$$G_o^{\mathsf{OP}} = \rho^o N_{\mathsf{RE}} R_o(C^o, \mathbf{B}^o) - G_o^{\mathsf{InP}}, \tag{11}$$

where ρ^o $(¢/bps)$ is the price per rate unit that OP o obtains by providing services to its users. The upper-level problem aims to maximize the weighted sum profit of the InP and OPs,

[4]This required fronthaul capacity for each OP depends on all wireless channel gains.

which can be mathematically stated as[5]

$$\max_{\{C^o,\mathbf{B}^o\}} \upsilon^{\mathsf{InP}} G^{\mathsf{InP}} + \sum_{o\in\Omega} \upsilon^o G_o^{\mathsf{OP}} \tag{12a}$$

$$\text{s. t.} \sum_{o\in\Omega} C^o \le \bar{C}_{\mathsf{cloud}}, \tag{12b}$$

$$\sum_{o\in\Omega} B_k^o \le \bar{B}_k, \quad \forall k \in \mathcal{K}, \tag{12c}$$

where υ^{InP} and υ^o represent the weights corresponding to the InP and OP o, respectively, which can be used to control the desirable profit sharing between the InP and OPs. In fact, only relative values of these weights are important in determining the slicing solution for the cloud computing and fronthaul capacity resources (i.e., $C^o, \mathbf{B}^o$). In some implementation, we can normalize these weights so that their sum is equal to 1, i.e., $\bar{\upsilon}^{\mathsf{InP}} + \sum_{o\in\Omega} \bar{\upsilon}^o = 1$. Moreover, $\bar{C}_{\mathsf{cloud}}$ describes the available computational resource in the cloud and $\bar{B}_k$ denotes the capacity of the fronthaul link connecting RRH k with the cloud.

In the lower level, each OP is interested in maximizing the sum rate through optimizing transmission rates and the numbers of quantization bits allocated to individual users. Because the SINR $\gamma_k^{(s)}(b_k^{(s)})$ is a complicated function of quantization bits $b_k^{(s)}$, we present a tight lower bound of the SINR in the following proposition. This SINR lower bound enables us to focus on a good signal quantization regime, which will be incorporated in the lower-level problems to have tractable design. Moreover, the SINR $\gamma_k^{(s)}(b_k^{(s)})$ can be upper-bounded by $\bar{\gamma}_k^{(s)} = D_k^{(s)}/I_k^{(s)}$, which can be obtained by setting the quantization noise $q_k^{(s)}(b_k^{(s)}) = 0$ in the SINR expression.

Proposition 1: If the number of quantization bits $b_k^{(s)}$ satisfies $q_k^{(s)}(b_k^{(s)}) \le \sqrt{Y_k^{(s)} I_k^{(s)}}$, then we have the following lower bound for the SINR $\gamma_k^{(s)}(b_k^{(s)})$

$$\gamma_k^{(s)}(b_k^{(s)}) \ge \underline{\gamma}_k^{(s)} = \sqrt{\bar{\gamma}_k^{(s)} + 1} - 1. \tag{13}$$

In addition, we have the following relations between the SINR upper and lower bounds

$$\underline{\gamma}_k^{(s)} \simeq \sqrt{\bar{\gamma}_k^{(s)}} \quad \textit{when } \bar{\gamma}_k^{(s)} \gg 1, \tag{14}$$

$$\underline{\gamma}_k^{(s)} \simeq \bar{\gamma}_k^{(s)}/2 \quad \textit{when } \bar{\gamma}_k^{(s)} \ll 1. \tag{15}$$

Proof: The proof is given in Appendix A. □

[5]In general, both InP and OPs would like to maximize to their profits. By maximizing the weighted sum profit of InP and OPs, we aim to reach a Pareto-optimal solution, which can be desirable for both the InP and all OPs. Here, the weights in the objective function can be pre-determined by all stakeholders to reflect their priorities and/or contract agreements.

The requirement $q_k^{(s)}(b_k^{(s)}) \le \sqrt{Y_k^{(s)} I_k^{(s)}}$ in Proposition 1 is indeed equivalent to $b_k^{(s)} \ge \lceil \underline{b}_k^{(s)} \rceil$ where

$$\underline{b}_k^{(s)} = \frac{1}{2}\left(\log_2\left(\sqrt{3}\pi\sqrt{\frac{Y_k^{(s)}}{I_k^{(s)}}}\right) - 1\right), \tag{16}$$

and $\lceil * \rceil$ stands for the ceiling operation.

The lower-level problem for OP o ($\mathcal{P}^o$) can be formally stated as

$$\max_{\mathbf{r}^o,\mathbf{b}^o} \sum_{k\in\mathcal{K}} \sum_{s\in\mathcal{S}_k^o} r_k^{(s)} \tag{17a}$$

$$\text{s. t.} \sum_{k\in\mathcal{K}} \sum_{s\in\mathcal{S}_k^o} C_k^{(s)} \le C^o/N_{\mathsf{RE}}, \tag{17b}$$

$$r_k^{(s)} \le \log_2\left(1 + \gamma_k^{(s)}(b_k^{(s)})\right), \quad \forall k \in \mathcal{K},\ \forall s \in \mathcal{S}_k^o, \tag{17c}$$

$$\sum_{s\in\mathcal{S}_k^o} b_k^{(s)} \le B_k^o/(2N_{\mathsf{RE}}), \quad \forall k \in \mathcal{K}, \tag{17d}$$

$$b_k^{(s)} \ge \lceil \underline{b}_k^{(s)} \rceil, \quad \forall k \in \mathcal{K},\ \forall s \in \mathcal{S}_k^o, \tag{17e}$$

$$b_k^{(s)} \text{ is integer}, \quad \forall k \in \mathcal{K},\ \forall s \in \mathcal{S}_k^o, \tag{17f}$$

$$r_k^{(s)} \in \mathcal{R}, \quad \forall k \in \mathcal{K},\ \forall s \in \mathcal{S}_k^o. \tag{17g}$$

Constraint (17b) requires that the total computation effort required by all users of OP o should not exceed the sliced computational resource for this OP, C^o, which is determined from the upper-level problem. The second constraint (17c) is the standard capacity constraint for PRB s while constraint (17d) ensures that the amount of fronthaul capacity allocated for RRH k of OP o is upper bounded by B_k^o, which is also determined from the upper-level problem. Constraints (17e) capture the good quantization regime with the lower bound of quantization bits $\underline{b}_k^{(s)}$ given in (16).

This two-level resource allocation design is difficult to tackle because we have to optimize the discrete variables related to the rate and quantization bit allocation $\mathbf{r}^o, \mathbf{b}^o$ in the lower-level problem as well as the continuous variables $C^o, \mathbf{B}^o$ in the upper-level problem. Moreover, computational complexity $C_k^{(s)} = \{\chi_k^{(s)}(r_k^{(s)}, b_k^{(s)})\}$ in the lower-level problems is a complex function of the optimization variables $r_k^{(s)}, b_k^{(s)}$. Finally, the lower-level and upper-level problems are tightly coupled since the variables $C^o, \mathbf{B}^o$ in the later are the parameters in the former.

Remark 1: The objective function of the upper-level problem represents the weighted profits of the InP and all OPs. If the weights of InP and OPs are the same (i.e., $\upsilon^{\mathsf{InP}} = \upsilon^o, \forall o$), then the upper-level problem will optimize

$$\max_{\{C^o,\mathbf{B}^o\}} \sum_{o\in\Omega} \rho^o N_{\mathsf{RE}} R_o(C^o, \mathbf{B}^o), \tag{18}$$

where recall that ρ^o (¢/bps) is the price per rate unit that OP o obtains by providing services to its users. The objective function in this case is the weighted sum of the total rates of individual OPs. On the other hand, the lower-level problems

aim to maximize the sum rates of OPs individually. Specifically, the OP o optimizes the sum rate of its users as

$$\max_{\mathbf{r}^o,\mathbf{b}^o} R_o(C^o,\mathbf{B}^o)=\sum_{k\in\mathcal{K}}\sum_{s\in\mathcal{S}_k^o} r_k^{(s)}. \tag{19}$$

Therefore, the upper-level problem and lower-level problems even in this special case still optimize different objectives; therefore, we have to solve all these problems to obtain the final network slicing solution.

III. PROBLEM TRANSFORMATION AND CONVEXITY CHARACTERIZATION

We propose a two-stage solution framework to solve the bi-level resource allocation formulation where we solve the relaxed problems in the first stage and develop rounding methods to find an efficient and feasible solution for the original design problems in the second stage.

A. PROBLEM RELAXATION

The lower-level problem with discrete optimization variables $\mathbf{b}^o$ and $\mathbf{r}^o$ plays an important role in our design. Since optimization discrete variables are highly complex, we adopt the natural relaxation approach to tackle the lower-level problems where the discrete variables are relaxed into the continuous ones. Specifically, the constraint (17g) is relaxed to

$$R_{\min}\le r_k^{(s)}\le R_{\max},\quad \forall k\in\mathcal{K},\ \forall s\in\mathcal{S}, \tag{20}$$

where $R_{\min}=R_1$ and $R_{\max}=R_{M_R}$ represent the lowest and highest rates in the rate set $\mathcal{R}$, respectively. With this relaxation, we study the following relaxed lower-level (RLL) problem[6]

$$\max_{\mathbf{r}^o,\mathbf{b}^o}\ \sum_{k\in\mathcal{K}}\sum_{s\in\mathcal{S}} r_k^{(s)}\quad \text{s. t. (17b)-(17e) and (20).} \tag{21}$$

Let $\bar{R}_o(C^o,\mathbf{B}^o)$ denote the optimal total rate of all users from OP o obtained by solving this RLL problem. It is clear that $\bar{R}_o(C^o,\mathbf{B}^o)$ is the upper bound of $R_o(C^o,\mathbf{B}^o)$. Based on the obtained rates $\bar{R}_o(C^o,\mathbf{B}^o)$, we consider the following relaxed upper-level (RUL) problem

$$\begin{aligned}\max_{\{C^o,\mathbf{B}^o\}}\ &\sum_{o\in\Omega}\Psi^o(C^o,\mathbf{B}^o)\\ &=\upsilon^{\mathsf{InP}}G^{\mathsf{InP}}+\sum_{o\in\Omega}\upsilon^o\left(\rho^o N_{\mathsf{RE}}\bar{R}_o(C^o,\mathbf{B}^o)-G_o^{\mathsf{InP}}\right)\\ &\text{s. t. (12b), (12c),}\end{aligned} \tag{22}$$

where $\Psi^o(C^o,\mathbf{B}^o)=(\upsilon^{\mathsf{InP}}-\upsilon^o)G_o^{\mathsf{InP}}+\upsilon^o\rho^o N_{\mathsf{RE}}\bar{R}_o(C^o,\mathbf{B}^o)$.

Remark 2: Note that the feasible regions of the relaxed problems are larger than those of the corresponding original problems. Hence, the upper-level and lower-level problems are infeasible if the RUL and RLL problems are infeasible, respectively.

[6]Constraints (17f) are not needed after relaxation.

B. CONVEXITY CHARACTERIZATION

1) CONVEXITY OF RLL PROBLEM

To solve the RLL problem, we first characterize the convexity of the computational complexity function $C_k^{(s)}=\{\chi_k^{(s)}(r_k^{(s)},b_k^{(s)})\}$ in the following theorem.

Theorem 1: $\chi_{k,u}^{(s)}(r_k^{(s)},b_k^{(s)})$ is a jointly convex function with respect to variables $(r_k^{(s)},b_k^{(s)})$ if

$$q_k^{(s)}(b_k^{(s)})\le\sqrt{Y_k^{(s)}I_k^{(s)}}. \tag{23}$$

Proof: The proof is given in Appendix B. □

Based on the result in this theorem, we state the convexity of the RLL problem in the following proposition.

Proposition 2: The RLL problem (21) is convex.

Proof: Note that the condition required to have the SINR lower bound (13) in Proposition 1, which is captured in constraint (17e), is exactly the requirement in (23). Hence, the constraint (17b) is convex if $b_k^{(s)}$ satisfies the constraint (17e). In addition, the constraint function in (17c) is convex due to the fact that $\log_2\left(1+\gamma_k^{(s)}(b_k^{(s)})\right)$ is a concave function with respect to $b_k^{(s)}$. Moreover, the objective function and other constraints of the RLL problem (21) are in linear form. Therefore, the RLL problem (21) is convex. □

2) CONVEXITY OF RUL PROBLEM

We characterize the convexity of the RUL problem in the following theorem and proposition.

Theorem 2: $\bar{R}_o(C^o,\mathbf{B}^o)$ is a concave function with respect to C^o and $\mathbf{B}^o$.

Proof: The proof is given in Appendix C. □

Based on the result in this theorem, we have the following proposition.

Proposition 3: The RUL problem (22) is convex.

Proof: Due to the result in Theorem 2, the objective function of the RUL problem is concave with respect to variables C^o and $\mathbf{B}^o$. In addition, all the constraint functions are in linear form. Therefore, the RUL problem (22) is convex. □

IV. RESOURCE ALLOCATION ALGORITHMS

A. PROPOSED ALGORITHM TO SOLVE RLL PROBLEMS

According to the result in Proposition 2, the RLL problem is convex and it can be verified that Slater's conditions hold; hence, it can be solved optimally by tackling the corresponding dual problem. Specifically, the dual function $g(\lambda)$ of the RLL problem can be defined as

$$g^o(\lambda^o)=\max_{\mathbf{r}^o,\mathbf{b}^o}\Phi^o(\lambda^o,\mathbf{r}^o,\mathbf{b}^o)\quad \text{s. t. (17c)-(17e) and (20),} \tag{24}$$

where $\Phi^o(\lambda^o,\mathbf{r}^o,\mathbf{b}^o)$ is the Lagrangian obtained by relaxing the constraint (17b), which can be expressed as

$$\begin{aligned}\Phi^o(\lambda^o,\mathbf{r}^o,\mathbf{b}^o)=&\sum_{k\in\mathcal{K}}\sum_{s\in\mathcal{S}_k^o} r_k^{(s)}\\ &-\lambda^o\left(\sum_{k\in\mathcal{K}}\sum_{s\in\mathcal{S}_k^o} C_k^{(s)}-\frac{C^o}{N_{\mathsf{RE}}}\right),\end{aligned} \tag{25}$$

$$\Phi_k^o(\lambda^o, \mathbf{r}_k^o, \mathbf{b}_k^o) = \sum_{s\in\mathcal{S}_k^o} r_k^{(s)} - \lambda^o \sum_{s\in\mathcal{S}_k^o} C_k^{(s)} = \sum_{s\in\mathcal{S}_k^o} \left[(1-\lambda^o AB) r_k^{(s)} + 2\lambda^o A r_k^{(s)} \log_2\left(\log_2\left(1+\gamma_k^{(s)}(b_k^{(s)})\right) - r_k^{(s)}\right)\right] \quad (29)$$

where λ^o denotes the Lagrange multiplier. Then, the dual problem can be written as

$$\min_{\lambda^o} \; g^o(\lambda^o) \quad \text{s. t. } \lambda^o \geq 0. \quad (26)$$

Since the dual problem is always convex, $g^o(\lambda^o)$ can be minimized by using the standard sub-gradient method where the dual variable λ^o can be iteratively updated as follows:

$$\lambda_{(l+1)}^o = \left[\lambda_{(l)}^o + \delta_{(l)}^o \left(\sum_{k\in\mathcal{K}}\sum_{s\in\mathcal{S}_k^o} C_k^{(s)} - \frac{C^o}{N_{\mathsf{RE}}}\right)\right]^+, \quad (27)$$

where l denotes the iteration index, $\delta_{(l)}^o$ represents the step size, and $[x]^+$ is defined as $\max(0, x)$. This sub-gradient update guarantees to converge to the optimal value of λ^o for given primal point $(\mathbf{r}^o, \mathbf{b}^o)$ if the step-size $\delta_{(l)}^o$ is chosen appropriately so that $\delta_{(l)}^o \to 0$ when $l \to \infty$ such as $\delta_{(l)}^o = 1/\sqrt{l}$ [28].

To solve the RLL problem optimally, one can iteratively solve problem (24) for a given dual point λ_o and employ the sub-gradient method to update λ_o as in (27). Therefore, the remaining step is to solve the optimization problem in the right-hand-side of (24). We will show that this can be accomplished by decoupling this problem into K sub-problems corresponding to K cells and iteratively solving these sub-problems optimally. It can be verified that the Lagrangian function (25) can be rewritten as

$$\Phi^o(\lambda^o, \mathbf{r}^o, \mathbf{b}^o) = \sum_{k\in\mathcal{K}} \Phi_k^o(\lambda^o, \mathbf{r}_k^o, \mathbf{b}_k^o) + \lambda^o \frac{C^o}{N_{\mathsf{RE}}}, \quad (28)$$

where $\Phi_k^o(\lambda^o, \mathbf{r}_k^o, \mathbf{b}_k^o)$ is expressed in (29) at the top of this page and $\mathbf{r}_k^o$, $\mathbf{b}_k^o$ represent the vectors of all rates and numbers of quantization bits corresponding to cell k and OP o. Therefore, we can decouple the RLL problem into K independent sub-problems, $(\mathcal{P}_k^o)$'s, which are given as

$$(\mathcal{P}_k^o) \quad \max_{\mathbf{r}_k^o, \mathbf{b}_k^o} \; \Phi_k^o(\lambda^o, \mathbf{r}_k^o, \mathbf{b}_k^o) \quad (30a)$$

$$\text{s. t. } r_k^{(s)} \leq \log_2\left(1+\gamma_k^{(s)}(b_k^{(s)})\right), \quad \forall s \in \mathcal{S}_k^o, \quad (30b)$$

$$\sum_{s\in\mathcal{S}_k^o} b_k^{(s)} \leq B_k^o/(2N_{\mathsf{RE}}), \quad (30c)$$

$$b_k^{(s)} \geq \lceil \underline{b}_k^{(s)} \rceil, \quad \forall s \in \mathcal{S}_k^o, \quad (30d)$$

$$R_{\min} \leq r_k^{(s)} \leq R_{\max}, \quad \forall s \in \mathcal{S}_k^o. \quad (30e)$$

This problem is still convex due to the result in Proposition 2. In the following, we solve problem $(\mathcal{P}_k^o)$ optimally by alternately optimizing over one variable in $\mathbf{r}_k^o$ and $\mathbf{b}_k^o$ while keeping the other fixed.

1) SOLVING $(\mathcal{P}_k^o)$ FOR GIVEN $\mathbf{b}_k^o$

For a given $\mathbf{b}_k^o$, problem $(\mathcal{P}_k^o)$ becomes

$$\max_{\mathbf{r}_k^o} \sum_{s\in\mathcal{S}_k^o} \left[E_k^{(s)} r_k^{(s)} + 2\lambda^o A r_k^{(s)} \log_2\left(1 - \frac{r_k^{(s)}}{t(b_k^{(s)})}\right)\right] \quad (31a)$$

$$\text{s. t. } R_{\min} \leq r_k^{(s)} \leq \min\left(t(b_k^{(s)}), R_{\max}\right), \forall s \in \mathcal{S}_k^o, \quad (31b)$$

where $E_k^{(s)} = \left(1 - \lambda^o AB + 2\lambda^o A \log_2 t(b_k^{(s)})\right)$ and $t(b_k^{(s)}) = \log_2\left(1+\gamma_k^{(s)}(b_k^{(s)})\right)$.

The optimal solution of this problem is described in the following proposition.

Proposition 4: The optimal solution to problem (31) can be expressed as

$$r_k^{(s)\star} = \max\left[R_{\min}, \min\left(t(b_k^{(s)}), R_{\max}, r|_{\frac{\partial w(r)}{\partial r} = -E_k^{(s)}}\right)\right], \quad (32)$$

where $w(r) = 2\lambda^o A r \log_2\left(1 - r/t(b_k^{(s)})\right)$.

Proof: The proof is given in Appendix D. □

2) SOLVING $(\mathcal{P}_k^o)$ FOR GIVEN $\mathbf{r}_k^o$

For given $\mathbf{r}_k^o$, problem $(\mathcal{P}_k^o)$ becomes equivalent to

$$\max_{\mathbf{b}_k^o} \sum_{s\in\mathcal{S}_k^o} z\left(b_k^{(s)}\right) \quad (33a)$$

$$\text{s. t. } b_k^{(s)} \geq \max\left(\lceil \underline{b}_k^{(s)} \rceil, t^{-1}\left(r_k^{(s)}\right)\right), \quad \forall s \in \mathcal{S}_k^o, \quad (33b)$$

$$\sum_{s\in\mathcal{S}_k^o} b_k^{(s)} \leq B_k^o/(2N_{\mathsf{RE}}), \quad (33c)$$

where $z(b_k^{(s)}) = G_k^{(s)} \log_2\left(\log_2\left(1+\gamma_k^{(s)}(b_k^{(s)})\right) - r_k^{(s)}\right)$, $G_k^{(s)} = 2\lambda A r_k^{(s)}$, and $t^{-1}(r)$ is the inverse function of $t(b)$. The objective function of this problem is concave with respect to $\mathbf{b}_k^o$. Hence, this problem is a convex one whose optimal solution can be obtained by studying the Karush-Kuhn-Tucker optimality conditions. The optimal solution of this problem is summarized in the following proposition.

Proposition 5: The optimal solution of problem (33) can be expressed as

$$b_k^{(s)\star} = \max\left(\lceil \underline{b}_k^{(s)} \rceil, t^{-1}\left(r_k^{(s)}\right), b|_{\frac{\partial z(b)}{\partial b} = \mu}\right), \quad (34)$$

where μ is a constant so that $\sum_{s\in\mathcal{S}_k^o} b_k^{(s)\star} = B_k^o/(2N_{\mathsf{RE}})$.

Proof: The proof is given in Appendix E. □

Remark 3: It can be verified that if $(\mathbf{r}_k^{o\star}, \mathbf{b}_k^{o\star})$ is the optimal solution of the RLL problem, then $(\mathbf{r}_{\min}^o, \mathbf{b}_k^{o\star})$ is a feasible solution where $\mathbf{r}_{\min}^o$ has the same size as $\mathbf{r}_k^o$ and all of its elements are equal $R_{\min}$. Therefore, the feasibility of the

RLL problem can be verified by solving problem (33) *with* $\mathbf{r}_k^o = \mathbf{r}_{\min}^o$, *which is indeed employed in the* ***initialization step*** *of Algorithm 1 presented in the following.*

3) PROPOSED ALGORITHM

We summarize how to solve the RLL problem in Algorithm 1. In this iterative algorithm, we alternatively update one of the two variables $\mathbf{b}_k^o$ and $\mathbf{r}_k^o$ while keeping the other fixed until convergence. Because the optimal solution for each variable can be obtained, the objective value increases over iterations which ensures fast convergence for this algorithm. We then update the dual variable λ^o as in (27) in the outer loop.

Algorithm 1 Algorithm to Solve RLL Problem

1: Initialization: Set $r_k^{(s)} = R_{\min}\ \forall(k,s)$, $\lambda_{(0)}^o = 0$ and $l = 0$. Choose a tolerance parameter ε for convergence.
2: **repeat**
3: **for** $k \in \mathcal{K}$ **do**
4: **repeat**
5: Fix $\mathbf{r}_k^o$ and update $\mathbf{b}_k^o$ as in (34) with $\lambda_{(l)}^o$.
6: Fix $\mathbf{b}_k^o$ and update $\mathbf{r}_k^o$ as in (32) with $\lambda_{(l)}^o$.
7: **until** Convergence.
8: **end for**
9: Calculate all $C_k^{(s)}$ with new $\mathbf{r}_k^o$ and $\mathbf{b}_k^o$.
10: Update $\lambda_{(l+1)}^o$ as in (27).
11: Set $l = l + 1$.
12: **until** $|\lambda_{(l)}^o - \lambda_{(l-1)}^o| < \varepsilon$.

B. PROPOSED ALGORITHM TO SOLVE RUL PROBLEM

Since the RUL problem is convex according to Proposition 3, its optimal solution can be found efficiently by standard convex optimization techniques. It appears non-tractable to derive the closed-form optimal solution of the RUL problem. Therefore, we employ the sub-gradient method to solve this problem based on the sub-gradient $\nabla\bar{R}_o(C^o, \mathbf{B}^o)$ of $\bar{R}_o(C^o, \mathbf{B}^o)$. Specifically, the sub-gradient method to iteratively update $C^o, \mathbf{B}^o$ can be performed as follows:

$$[C^o, \mathbf{B}^o]_{(l+1)} = \mathcal{P}\left[[C^o, \mathbf{B}^o]_{(l)} + \tau_{(l)}^o \nabla\Psi^o(C^o, \mathbf{B}^o)\right], \quad (35)$$

where $[C^o, \mathbf{B}^o]_{(l)}$ denotes the vector formed from the optimization variables C^o and $\mathbf{B}^o$, $\tau_{(l)}^o$ represents step size in the iteration l, $\nabla\Psi^o(C^o, \mathbf{B}^o) = \left[\frac{\partial\Psi^o(C^o,\mathbf{B}^o)}{\partial C^o} \frac{\partial\Psi^o(C^o,\mathbf{B}^o)}{\partial B_1^o} \cdots \frac{\partial\Psi^o(C^o,\mathbf{B}^o)}{\partial B_K^o}\right]^T$. Moreover, $\mathcal{P}[C^o, \mathbf{B}^o]$ represents the projection of $C^o, \mathbf{B}^o$ to the feasible region, which is achieved by solving the following quadratic problem

$$\min_{[C^o,\mathbf{B}^o]} \|[C^o, \mathbf{B}^o] - [\hat{C}^o, \hat{\mathbf{B}}^o]\|^2 \ \text{ s. t. (12b), (12c),} \quad (36)$$

where $\hat{C}^o = C_{(l)}^o + \delta_{(l)}^o \partial\Psi^o(C^o, \mathbf{B}^o)/\partial C^o$ and $\hat{B}_k^o = B_{k,(l)}^o + \delta_{(l)}^o \partial\Psi^o(C^o, \mathbf{B}^o)/\partial B_k^o$, $\forall k \in \mathcal{K}$ are updated $[C^o, \mathbf{B}^o]$ given by (35). The sub-gradient based updates guarantee to converge to the optimal values of $C^o, \mathbf{B}^o$ if the step-size $\tau_{(l)}^o$ is chosen appropriately to satisfy $\tau_{(l)}^o \to 0$ when $l \to \infty$ such as $\tau_{(l)}^o = 1/\sqrt{l}$ [28].

The remaining issue is to determine the value of $\nabla\Psi^o(C^o, \mathbf{B}^o)$, which can be expressed as

$$\nabla\Psi^o(C^o, \mathbf{B}^o) = \nu^o \rho^o N_{\mathsf{RE}} \begin{bmatrix} \partial\bar{R}_o(C^o, \mathbf{B}^o)/\partial C^o \\ \partial\bar{R}_o(C^o, \mathbf{B}^o)/\partial B_1^o \\ \cdots \\ \partial\bar{R}_o(C^o, \mathbf{B}^o)/\partial B_K^o \end{bmatrix} + (\upsilon^{\mathsf{InP}} - \upsilon^o) \begin{bmatrix} \psi^o \\ \beta_1^o \\ \cdots \\ \beta_K^o \end{bmatrix}, \quad (37)$$

where $\partial\bar{R}_o(C^o, \mathbf{B}^o)/\partial C^o$ and $\partial\bar{R}_o(C^o, \mathbf{B}^o)/\partial B_k^o, \forall k \in \mathcal{K}$, can be approximated as

$$\frac{\partial\bar{R}_o(C^o, \mathbf{B}^o)}{\partial C^o} \simeq \frac{\bar{R}_o(C^o + \Delta C^o, \mathbf{B}^o) - \bar{R}_o(C^o, \mathbf{B}^o)}{\Delta C^o}, \quad (38a)$$

$$\frac{\partial\bar{R}_o(C^o, \mathbf{B}^o)}{\partial B_k^o} \simeq \frac{\bar{R}_o(C^o, \mathbf{B}^o + \Delta\mathbf{B}_k^o) - \bar{R}_o(C^o, \mathbf{B}^o)}{\Delta B_k^o}, \quad (38b)$$

where $\Delta\mathbf{B}_k^o$ is the vector of size $K \times 1$ whose elements are zero except that the k^{th} element equals to ΔB_k^o. In (38), the values of ΔC^o and ΔB_k^o, $\forall k \in \mathcal{K}$ are chosen sufficiently small. We summarize the procedure to update $[C^o, \mathbf{B}^o]$'s in Algorithm 2 which is employed to solve the RUL problem.

Algorithm 2 Algorithm to Solve RUL Problem

1: Initialization: Set $C_{(0)}^o = \bar{C}_{\mathsf{cloud}}/O$, and $B_{k,(0)}^o = \bar{B}_k/O$ for all $(o,k) \in \Omega \times \mathcal{K}$, $\nu_{(0)}^o = 0$ for all $o \in \Omega$, and $l = 0$.
2: **repeat**
3: Run Algorithm 1 to obtain $\{\bar{R}_o(C^o, \mathbf{B}^o)\}$ with $\{[C^o, \mathbf{B}^o]_{(l)}\}$ for all $o \in \Omega$.
4: Run Algorithm 1 to obtain $\bar{R}_o(C^o + \Delta C^o, \mathbf{B}^o)$ and $\bar{R}_o(C^o, \mathbf{B}^o + \Delta\mathbf{B}_k^o)$.
5: Calculate $\nabla\Psi^o(C^o, \mathbf{B}^o)$ as in (37) by using $\bar{R}_o(C^o + \Delta C^o, \mathbf{B}^o)$ and $\bar{R}_o(C^o, \mathbf{B}^o + \Delta\mathbf{B}_k^o)$ to determine $\partial\bar{R}_o(C^o, \mathbf{B}^o)/\partial C^o$, and $\partial\bar{R}_o(C^o, \mathbf{B}^o)/\partial B_k^o$ for all $(k,o) \in \mathcal{K} \times \Omega$ as in (38).
6: Update $[C^o, \mathbf{B}^o]_{(l+1)}$ for all $o \in \Omega$ as in (35).
7: Set $l = l + 1$.
8: **until** Convergence.

C. ROUNDING DESIGN

After running Algorithm 2, we obtain a feasible solution $\{C^o, \mathbf{B}^o\}$ and their corresponding $\left\{r_k^{(s)\bigstar}\right\}$ and $\left\{b_k^{(s)\bigstar}\right\}$ of the relaxed problems, which take real values. To obtain a feasible and discrete solution that satisfies the constraints (17e), (17f), (17g), the continuous variables must be appropriately rounded to the corresponding discrete values.[7] This rounding design must be conducted carefully because the resulting discrete results may not satisfy the original cloud computation and fronthaul constraints. Toward this end, we propose two rounding methods which are described in the following.

[7]It is possible that no feasible discrete variables can be found from the relaxed solutions $\left\{r_k^{(s)\bigstar}\right\}$ and $\left\{b_k^{(s)\bigstar}\right\}$ if the constraints of the original problems are very tight.

1) ROUNDING (IR) METHOD

For each OP, we iteratively run Algorithm 2 and perform the following task in each iteration. We choose one value of $r_k^{(s)\bigstar}$ (or $b_k^{(s)\bigstar}$), which is closest to one rate value in $\mathcal{R}$ (or an integer) and fix it to that value in following iterations. This one-by-one rounding process is repeated until convergence.

2) ONE-TIME ROUNDING AND ADJUSTING (RA) METHOD

This method has two phases for each OP. First, we round all $\left\{r_k^{(s)\bigstar}\right\}$ and $\left\{b_k^{(s)\bigstar}\right\}$ to their closest values in $\mathcal{R}$ and the set of integers, respectively in the first phase. Then, if any constraints are violated, we round down the corresponding variables one-by-one where the variable that affects the violated constraints the most is chosen in each rounding-down step in the second phase.

V. GREEDY ALGORITHMS AND COMPLEXITY ANALYSIS

A. GREEDY ALGORITHMS

To the best of our knowledge, there is no existing resource allocation algorithm for the end-to-end network slicing design studied in this paper. For performance evaluation of the developed algorithm, we present two greedy one-level resource allocation algorithms as referencing benchmarks in this section. Hence, the one-level benchmark approaches will be considered by easing one of two challenges of bi-level problem, i.e., network resource slicing among OPs and resource allocation optimization for users of each OP. Specifically, the C-RAN network resources InP are shared directly to all OPs in the first approach, named *"Slicing-Relaxed Greedy Algorithm"* (Algorithm 3) while resources are allocated directly with respect to each user in the second one, named *"Resource-Allocation-Relaxed Greedy Algorithm"* (Algorithm 4).

1) SLICING-RELAXED GREEDY ALGORITHM

This algorithm includes two stages. In stage one, the network resources of InP are fully sliced to OPs based on their upper bound values of achievable rates. Then, each OP will allocate the rate and quantization bits to its users. In particular, we first estimate the upper bound of the rate $r_k^{(s)}$ on every PRB s and cell k as $\log_2\left(1+\bar{\gamma}_k^{(s)}\right)$ where $\bar{\gamma}_k^{(s)} = D_k^{(s)}/I_k^{(s)}$. We then allocate the computational and fronthaul capacity resources for different OPs based on the upper bound of the sum rate of each OP o as follows:

$$C_{\mathsf{ga}}^o = \hat{R}^o \bar{C}_{\mathsf{cloud}} / \sum_{o\in\Omega} \hat{R}^o, \tag{39}$$

$$B_{k,\mathsf{ga}}^o = \hat{R}_k^o \bar{B}_k / \sum_{o\in\Omega} \hat{R}_k^o, \quad \forall k \in \mathcal{K}, \tag{40}$$

where $\hat{R}_k^o = \sum_{s\in\mathcal{S}_k^o} \log_2\left(1+\bar{\gamma}_k^{(s)}\right)$ and $\hat{R}^o = \sum_{k\in\mathcal{K}} \hat{R}_k^o$. In stage two, we propose a simple method to solve the RLL problem for OP o with C_{ga}^o and $\mathbf{B}_{k,\mathsf{ga}}^o$. Specifically, we optimize the quantization bit allocation to maximize the sum rate of all users by relaxing the underlying variables to continuous ones and solving the following problem

$$\max_{\mathbf{b}} \sum_{k\in\mathcal{K}} \sum_{s\in\mathcal{S}} \log_2\left(1+\gamma_k^{(s)}(b_k^{(s)})\right)$$
$$\text{s. t.} \sum_{s\in\mathcal{S}_k^o} b_k^{(s)} \le B_{k,\mathsf{ga}}^o/(2N_{\mathsf{RE}}), \quad \forall k \in \mathcal{K}. \tag{41}$$

Similar to problem (31), it can be shown that this problem is convex because its objective function is concave. By studying the KKT optimality conditions, we can obtain the optimal solution as follows:

$$b_k^{(s)\prime} = \max\left(0, b|_{\frac{\partial t(b)}{\partial b}=\nu}\right), \tag{42}$$

where ν is a constant which must be set to satisfy $\sum_{s\in\mathcal{S}} b_k^{(s)\prime} = B_{k,\mathsf{ga}}^o/(2N_{\mathsf{RE}})$. Then, we can obtain the feasible vector $\mathbf{b}$ by applying the flooring operation to $\mathbf{b}'$.

The remaining task is to determine the users' rates that satisfy constraints (17b) and (17c) which can be accomplished by applying the "Complexity Cut-Off" method [23]. Specifically, we start by setting each $r_k^{(s)}$ to the highest value in the rate set $\mathcal{R}$ which is smaller than $\log_2\left(1+\gamma_k^{(s)}(b_k^{(s)})\right)$. Then, we iteratively reduce the rate variable that requires the highest computation effort if the cloud computation constraint is violated (i.e., the required computation effort of OP o, $C_{\mathsf{tt}}^o(\mathbf{r}^o, \mathbf{b}^o)$, is greater than the assigned value C_{ga}^o). This iterative process is performed until all cloud computation constraints are satisfied.

Algorithm 3 Slicing-Relaxed Greedy Algorithm

1: Define the network resource for each OP as in (39)–(40).
2: **for** OP o **do**
3: Calculate $\{b_k^{(s)\prime}\}_{k\in\mathcal{K},s\in\mathcal{S}_k^o}$ as in (42).
4: Set $b_k^{(s)} = \lfloor b_k^{(s)\prime} \rfloor$, for all $(k,s) \in \mathcal{K}\times\mathcal{S}_k^o$.
5: Set $r_k^{(s)} = \max_{r\in\mathcal{R}} r$ s. t. $r \le t(b_k^{(s)})$, for all (k,s).
6: **while** $C_{\mathsf{tt}}^o(\mathbf{r}^o, \mathbf{b}^o) > C_{\mathsf{ga}}^o$ **do**
7: Find $(k^*, s^*) = \arg\max C_k^{(s)}$.
8: Reduce $r_{k^*}^{(s^*)}$ to the nearest value in $\mathcal{R}$.
9: **end while**
10: **end for**

2) RESOURCE-ALLOCATION-RELAXED GREEDY ALGORITHM

In this approach, we ease the difficulty of resource allocation for each operator by setting the equal rate and quantization bits for all users of each operator in each cell, i.e., $r_k^{(s)} = r_k^o$ and $b_k^{(s)} = b_k^o = B_k^o/2N_{\mathsf{RE}}|\mathcal{S}_k^o|$ if $s \in \mathcal{S}_k^o$. Then, the total CE required by the cloud to successfully decode the signals for all users from OP o in cell k can be given as $C_k^o(r_k^o, b_k^o) = N_{\mathsf{RE}} \sum_{s\in\mathcal{S}_k^o} \chi_k^{(s)}\left(r_k^o, b_k^o\right)$. Considering the constraints (17c) and (17e) of the lower level problem, we can rewrite the bi-level problem for given values of r_k^o's into a

Algorithm 4 Resource-Allocation-Relaxed Greedy Algorithm

1: Choose r_k^o as the largest value in $\mathcal{R}$ so that $r \leq \min_{s\in\mathcal{S}_k^o} \log_2\left(1+\bar{\gamma}_k^{(s)}\right)$, $\forall\,(o,k)$.
2: **repeat**
3: Solve problem (43).
4: **if** Solution of (43) is infeasible **then**
5: Choose r_k^o ($r_k^o > R_{\min}$) corresponding to the highest value of $\sum_{s\in\mathcal{S}_k^o}\chi(r_k^o, \underline{b}_k^{(s)})$ and reduce its value to the lower one in $\mathcal{R}$.
6: **end if**
7: **until** Solution of (43) is feasible.

one-level problem as follows.

$$\max_{\{b_k^o\}} N_{\mathsf{RE}} \sum_{o\in\Omega}\left[\upsilon^o\rho^o\sum_{k\in\mathcal{K}}|\mathcal{S}_k^o|r_k^o + \left(\upsilon^{\mathsf{InP}}-\upsilon^o\right) \times \sum_{k\in\mathcal{K}}\left(2\beta_k^o|\mathcal{S}_k^o|b_k^o + \sum_{s\in\mathcal{S}_k^o}\psi^o\chi_k^{(s)}\left(r_k^o, b_k^o\right)\right)\right] \tag{43a}$$

$$\text{s. t. } N_{\mathsf{RE}}\sum_{(o,k)}\sum_{s\in\mathcal{S}_k^o}\chi_k^{(s)}\left(r_k^o, b_k^o\right) \leq \bar{C}_{\mathsf{cloud}}, \tag{43b}$$

$$\sum_{o\in\Omega} 2N_{\mathsf{RE}}|\mathcal{S}_k^o|b_k^o \leq \bar{B}_k, \quad \forall k\in\mathcal{K}, \tag{43c}$$

$$r_k^o \leq \log_2\left(1+\gamma_k^{(s)}(b_k^o)\right), \quad \forall\,(o,k,s)\in\Omega\times\mathcal{K}\times\mathcal{S}_k^o, \tag{43d}$$

$$b_k^o \geq \lceil\underline{b}_k^{(s)}\rceil, \quad \forall\,(o,k,s)\in\Omega\times\mathcal{K}\times\mathcal{S}_k^o. \tag{43e}$$

Thanks to Theorem 1, one can see that the problem (43) is convex if $\upsilon^{\mathsf{InP}} \leq \upsilon^o$. Hence, this relaxed one-level problem can be solved easily to obtain the network slices for all OPs when r_k^o's are given. When $\upsilon^{\mathsf{InP}} > \upsilon^o$, the problem (43) aims to maximizing a convex function which can be solved by linearising the objective function as shown in Frank-Wolfe Algorithm [29]. Based on these interesting results, we devise a heuristic approach to solve (12) which is summarized in Algorithm 4. Specifically, we will search the values of r_k^o's from high to low to indicate the good ones for which the problem (43) is feasible.

B. COMPLEXITY ANALYSIS

In this section, we investigate the complexities of our proposed solution approach, e.g., Algorithms 2 integrating with Algorithms 1 and two rounding methods, and two greedy algorithms.

1) COMPLEXITY OF OUR PROPOSED ALGORITHMS

a: COMPLEXITY OF ALGORITHM 1

As can be observed, Algorithms 1 applies the *"alternative direction method"* whose convergence rate is $1/m$ where m is the number of iterations [30]. Additionally, based on the Propositions 4 and 5, the complexity of each iteration of Algorithms 1 mainly depends on that due to "water filling" process to calculate $b_k^{(s)}$, i.e., $\mathcal{O}(|\mathcal{S}_k^o|\log(|\mathcal{S}_k^o|))$ [31] where $|\mathcal{A}|$ stands for the cardinal number of set $\mathcal{A}$. Hence, the complexity of Algorithms 1 can be estimated as

$$X_{\mathsf{RLL}} = \mathcal{O}(S\log(S)\mathsf{F}_{\mathsf{fa}}^{-1}(\zeta_{\mathsf{RLL}}^{-1})), \tag{44}$$

where $\mathsf{F}_{\mathsf{fa}}^{-1}(.)$ is the inverse function of factorial function and ζ_{RLL} is the solution accuracy of Algorithm 1.

b: COMPLEXITY OF ALGORITHM 2

It is observed that Algorithms 2 includes employing Algorithms 1 several time to determine the the sub-gradient values. In addition, Algorithm 2 employs the Sub-Gradient Descent method [32]. Hence, the complexity of Algorithm 2 can thus be expressed as

$$X_{\mathsf{RUL}} = 2\,O(K+1)X_{\mathsf{RLL}} \times \mathcal{O}(\zeta_{\mathsf{RUL}}^{-2}), \tag{45}$$

where ζ_{RUL} is the solution accuracy of Algorithm 2.

c: COMPLEXITY OF IR METHOD

For IR method, each OP performs the one-by-one rounding process and iteratively runs Algorithm 2 until convergence. Therefore, the complexity of IR method can be estimated as following.

$$X_{\mathsf{IR}} = \sum_{o\in\Omega}\sum_{s\in\mathcal{S}_k^o}\sum_{k\in\mathcal{K}}\left(2S_o + X_{\mathsf{RLL}}\right), \tag{46}$$

where $S_o = \sum_{k\in\mathcal{K}}|\mathcal{S}_k^o|$.

d: COMPLEXITY OF RA METHOD

For RA process, each OP rounds all its corresponding $r_k^{(s)}$'s and $b_k^{(s)}$'s in the first phase and then one-by-one rounds down the variable affecting the violated constraint the most. Thus, the complexity of RA method can be calculated as

$$X_{\mathsf{RA}} = 2KS + 2R\frac{B}{2N_{\mathsf{RE}}}K^2S^2X_{\mathsf{C}}, \tag{47}$$

where X_{C} is the complexity of calculating $C_k^{(s)}$ for specific values of $r_k^{(s)}$ and $b_k^{(s)}$.

2) COMPLEXITY OF GREEDY ALGORITHMS

a: COMPLEXITY OF ALGORITHM 3

As can be observed, Algorithms 3 employs the "water filling" process to solve the problem (41) and to calculate $b_k^{(s)}$'s after fully slicing the network resource of InP to all OPs. Then, it step-by-step reduces the value of $r_k^{(s)}$ to the next smaller one set $\mathcal{R}$ until the computation constraint is satisfied. Therefore, the complexity of this algorithm can be determined as follows

$$X_{\mathsf{SI-RIxed}} = \mathcal{O}(KS\log(KS)) + RK^2S^2X_{\mathsf{C}}. \tag{48}$$

b: COMPLEXITY OF ALGORITHM 4

In order to analyse the complexity of this algorithm, we first consider how to duel with the problem (43). According to [33], this problem can be solved by employing the duality method. Specifically, the problem is solved by iteratively

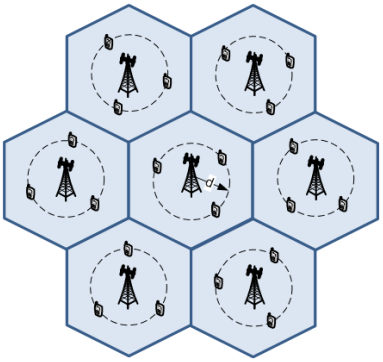

FIGURE 1. Simulation model.

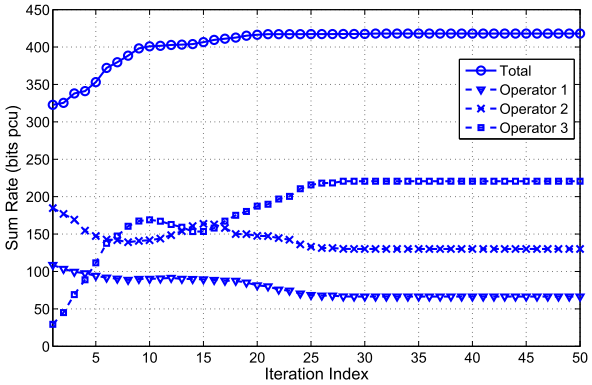

FIGURE 2. Variations of sum rates over iterations.

TABLE 1. Simulation Parameters.

Parameters	Values
K	7
O	3
S, S_1, S_2, S_3	$30, 5, 10, 15$
Path-loss $L_{j,u}^k$	$36.8\log_{10}(d_{j,u}^k) + 43.8 + 20\log_{10}(\frac{f_c}{5})$
f_c	$2.5\,GHz$
σ^2	$10^{-13}\,W$
$p_k^{(s)}$	$0.1\,W$
T'	0.2
ζ	6
ϵ_{ch}	10%

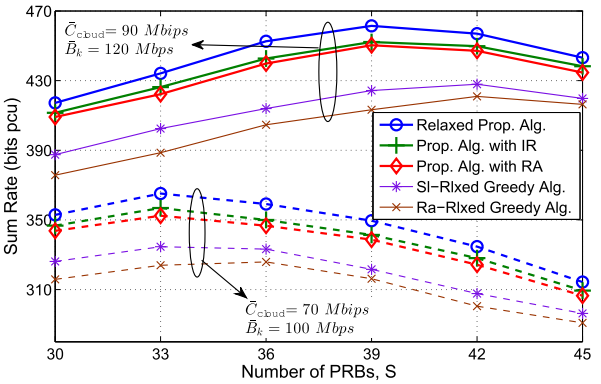

FIGURE 3. Sum rate vs number of PRBs (S).

solving the dual problem and updating Lagrange multiplier. Obtained by moving the constraint (43a) into the objective function, the dual problem is then similar to the problem (33). Hence, the complexity of solving the dual problem in each iteration is $\mathcal{O}(KS\log(KS))$ [31]. Based on the result in [33] the number of iterations is $\mathcal{O}(\zeta_4^{-2})$ where ζ_4 is the solution accuracy of Algorithm 4.

In addition, Algorithm 4 is proposed in the way that the (43) is solved again whenever the value of r_k^o's adjusted as in **Step 4-6**. Therefore, the complexity of this algorithm can be estimated as follows

$$X_{\mathsf{Ra-RIxed}} = ROK\,\mathcal{O}(KS\log(KS)\zeta_4^{-2}). \tag{49}$$

VI. NUMERICAL RESULTS

We consider the 7-cell network for performance evaluation where the distance between the centers of any two nearest RRHs is $400\,m$ as shown in Fig. 1. In each cell, we randomly place users so that the distance from the cell center to every user is d (m) (i.e., all users have the same distance to their corresponding RRHs in this simulation setting). The channel gains are generated by considering both Rayleigh fading and path-loss. The path-loss is modeled as $L_{j,u}^k = 36.8\log_{10}(d_{j,u}^k) + 43.8 + 20\log_{10}(\frac{f_c}{5})$ where $d_{j,u}^k$ denotes the distance from user u in cell j to RRH k and $f_c = 2.5GHz$. This path-loss model is chosen according to the general form of the path-loss formula (4.23) in [34], which is recommended by the WINNER II channel modeling project. We set the noise power $\sigma^2 = 10^{-13}\ W$ and the power $p_k^{(s)} = 0.1\ W$. Moreover, we set $T' = 0.2$, $\zeta = 6$ and $\epsilon_{\mathsf{ch}} = 10\%$ for the computation complexity model. The rate set $\mathcal{R}$ is chosen corresponding to 27 distinct MCSs with turbo coding as in the LTE standard [35]. In fact, the data rate corresponding to each MCS can be calculated as $TBS \times 10^3/N_{\mathsf{RE}}$ where the transport block size (TBS) for each MCS can be determined as in [35, Table 7.1.7.2.1] with $N_{\mathsf{PRB}} = 1$. Key simulation parameters are summarized in Table 1.

We assume that there are three OPs ($O = 3$) to obtain results in all simulations. Except for the results in Fig. 3 and 9, the numbers of PRBs assigned for these three OPs are set equal to 5, 10, 15 in each cell, which means $S = 30$. In each simulation, we allocate the PRBs to the users in each cell randomly so that the assigned number of PRBs for each OP is satisfied. In all simulations, we set the same fronthaul capacity limit for different cells. To obtain the results in Figs. 2-4, we set $\upsilon^{\mathsf{InP}} = \upsilon^o$ for all $o \in \Omega$.

We illustrate the convergence of our proposed algorithms in Fig. 2 where the variations of system sum rate and the rates of individual OPs over iterations by using Algorithm 1 and Algorithm 2 to solve the relaxed problem are shown. Note that the iterations in this figure correspond to the outer loop of these algorithms. To obtain results in this figure, $\bar{B}_k$ is set equal to 120 $Mbps$ for each cell $k \in \mathcal{K}$ and $\bar{C}_{\mathsf{cloud}} = 90$ $Mbips$. As can be seen, the system sum rate increases over the first 25 iterations before settling down at the maximum value. The third OP, who is assigned the largest number of PRBs, achieves low sum rate at the beginning then reaches the higher rate at convergence compared to the rates

TABLE 2. Summary of Key Notations.

Notations	Description
B_k	Capacity of the fronthaul link connecting RRH k with the cloud
B_k^o	Number of quantization bits per second required by OP o for forwarding quantized data from RRH k to the cloud
$\mathbf{b}^o$	Vectors whose elements represent numbers of quantization bits selected for all users from OP o
$b_k^{(s)}$	Number of bits to quantize the real and imaginary parts of $y_k^{(s)}$
$\underline{b}_k^{(s)}$	Lower bound of quantization bits
$C_{\mathsf{tt}}^o(\mathbf{r}^o, \mathbf{b}^o)$	Total computation effort required by the cloud for all users from OP o
$C_k^{(s)}$	Required computation effort for signal on PRB s in cell k
C_{cloud}	Available computational resource in the cloud
$d_{j,u}^k$	Distance from user u in cell j to RRH k
$f(y_k^{(s)})$	Probability density function of both the real and imaginary parts of $y_{k,u}^{(s)}$
$e_k^{(s)}$	Quantization error for $y_k^{(s)}$
G^{InP}	Profit obtained by the InP
G_o^{InP}	Payment from OP o to the InP
G_o^{OP}	Profit achieved by OP o
$h_{k,j}^{(s)}$	Channel gain from the user assigned PRB s in cell j to RRH k
K	Number of remote radio heads
$\mathcal{K}$	Set of cells
M_R	Number of different predetermined modulation and coding schemes
$\mathcal{M}_R$	Set of all data rate values
N_{RE}	Number of resource elements (REs) in one second and one subchannel of a PRB
O	Number of operators
$(\mathcal{P}^o)$	Lower-level problem for OP o
$p_j^{(s)}$	Transmission power corresponding to $x_j^{(s)}$
$q_k^{(s)}(b_k^{(s)})$	Quantization noise power corresponding to $b_k^{(s)}$ and $y_k^{(s)}$
$R_o(C^o, \mathbf{B}^o)$	Total rate of all users from OP o
R_{min} and R_{max}	Lowest and highest rates in the rate set $\mathcal{M}_R$
$\bar{R}_o(C^o, \mathbf{B}^o)$	Optimal total rate of all users from OP o obtained by solving the RLL problem
$\mathbf{r}^o$	Vectors whose elements represent the data rates selected for all users from OP o
$r_k^{(s)}$	Data rate on PRB s in cell k
S	Number of physical resource blocks
$\mathcal{S}$	Set of all PRBs
$\mathcal{S}_k^o$	Set of PRBs assigned to OP o in cell k
T	Model parameter
t_s	Slot duration in a PRB
$t^{-1}(r)$	Inverse function of $t(b)$
$x_k^{(s)}$	Baseband signal transmitted on PRB s in cell k
$Y_k^{(s)}$	Power of received signal $y_k^{(s)}$
$y_k^{(s)}$	Signal received at RRH k on PRB s
$\tilde{y}_k^{(s)}$	Quantized version of $y_k^{(s)}$
$\gamma_k^{(s)}(b_k^{(s)})$	SINR of the signal corresponding to PRB s in cell k
$\bar{\gamma}_k^{(s)}$	Upper-bound of $\gamma_k^{(s)}(b_k^{(s)})$
$\underline{\gamma}_k^{(s)}$	Lower-bound of $\gamma_k^{(s)}(b_k^{(s)})$
β_k^o	Price corresponding to one unit of fronthaul capacity ($¢/bps$)
ϵ_{ch}	Target computational outage probability
$\eta_k^{(s)}$	Gaussian thermal noise
Ω	Set of all OPs
ρ^o	Price per rate unit that OP o obtains by providing services to its users ($¢/bps$)
ζ	Parameter related to the connectivity of the decoder
υ^{InP} and υ^o	Weights corresponding to the InP and OP o
ψ^o	Price corresponding to one unit of computational effort ($¢/bips$)

of other OPs. This is because OP 3 is assigned more network resources than those for OPs 1 and 2.

In Fig. 3, we show the system sum rate obtained by different schemes, namely our proposed algorithms without rounding (Relaxed Prop. Alg.), with IR-rounding and RA-rounding methods (Prop. Alg. with IR and Prop. Alg. with RA), and two greedy algorithms, i.e., Slicing-Relaxed Algorithm (Sl-Rlxed Greedy Alg.) and Resource-Allocation-Relaxed Algorithm (Ra-Rlxed Greedy Alg.), versus the number of PRBs in each cell. To obtain these results, we sequentially add one more PRB to each OP in each cell to obtain different points on each curve. The Relaxed Prop. Alg. gives the upper-bound of the system sum rate of any resource allocation algorithm. The fact that the sum rates achieved by the Prop. Alg. with IR and Prop. Alg. with RA are very close to that achieved by the Relaxed Prop. Alg. confirms the efficiency of our proposed two-stage algorithms. Moreover, our proposed algorithms outperform the greedy algorithms in

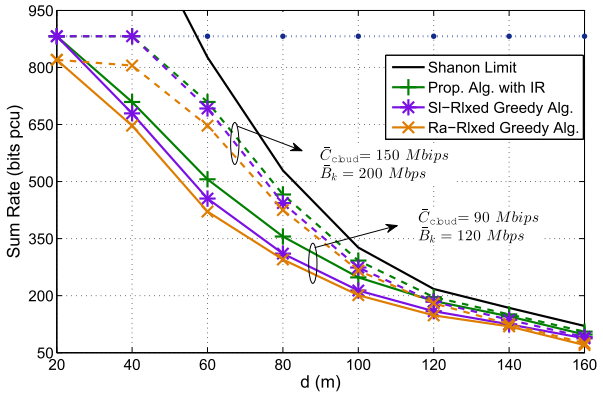

FIGURE 4. Sum rate vs distance from RRHs to their users.

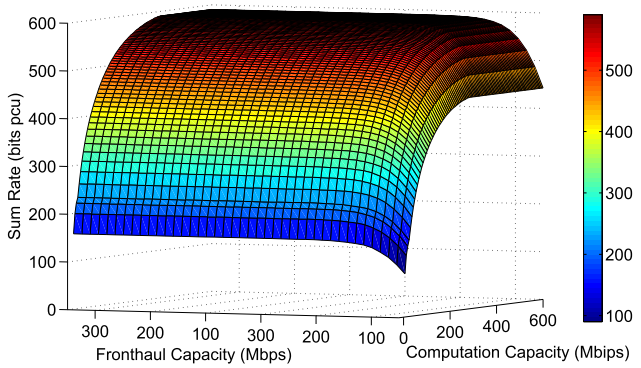

FIGURE 5. Sum rate vs $\bar{C}_{\text{cloud}}$ and $\bar{B}_k$.

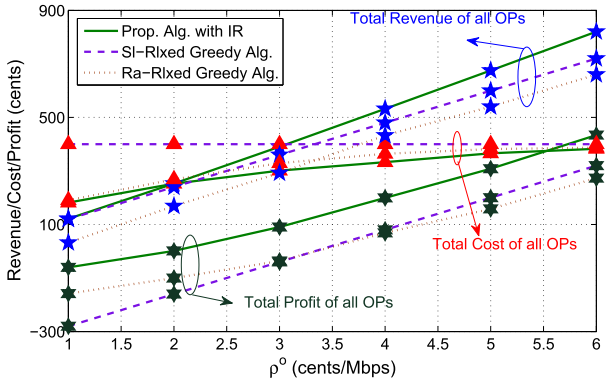

FIGURE 6. OPs' revenue/cost/profit vs service price ρ^o for users where $\upsilon^1 = \upsilon^2 = \upsilon^3 = 1$.

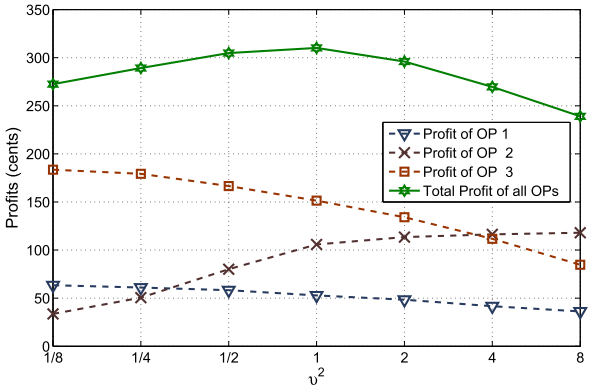

FIGURE 7. OPs' profits vs weighting parameter υ^2 for $\upsilon^1 = \upsilon^3 = 1$.

all studied scenarios and the Sl-Rlxed Greedy Alg. achieves higher total sum rate than the Ra-Rlxed Greedy Alg. does. In addition, the proposed algorithm with IR rounding results in slightly better sum rate than the RA rounding based counterpart. Interestingly, the system sum rate increases and then decreases as number of PRBs in each cell increases. This is because we focus on keeping the quality of the quantized signal so that every user can achieve at least $R_{\min}$ rate on its assigned PRB. This also means that limited computation and fronthaul capacity resources can indeed hurt the system performance if the bandwidth provisioning is not properly provisioned.

Fig. 4 shows the variations of the system sum rate due to the Prop. Alg. with IR and Greedy Alg. versus the user-RRH distance d under two different parameter settings, namely $\bar{C}_{\text{cloud}} = 150$ *Mbips* and $\bar{B}_k = 200$ *Mbps* and $\bar{C}_{\text{cloud}} = 90$ *Mbips* and $\bar{B}_k = 120$ *Mbps*. We also present the upper bound of the system sum rate, which is obtained by using the SINR upper bound $\bar{\gamma}_k^{(s)}$ by setting the quantization noise on all PRB s and cell k to zero. This rate upper bound is equal to $\sum_{\forall (k,s)} \log_2(1+\bar{\gamma}_k^{(s)})$ and it is denoted as "Shannon Limit" in this figure. For smaller d, the received signal becomes stronger in combating the multi-cell interference leading to higher link SINR $\bar{\gamma}_k^{(s)}$, which explains the higher sum rate for smaller d. It can also be observed that the achieved system sum rate tends to the rate upper bound (i.e., the "Shannon Limit") as the cloud computation and fronthaul capacity limits increase. Moreover, the Prop. Alg. with IR outperforms the greedy algorithms in all studied scenarios, which confirms the excellent performance of our proposed design.

To illustrate the impacts of limited network resources, we show the system sum rate upper-bound, which is the outcome of Algorithm 2, versus the computation limit ($\bar{C}_{\text{cloud}}$) and fronthaul capacity from the cloud to each cell $\bar{B}_k$ in Fig. 5. We can see that the higher cloud computation limit and larger fronthaul capacity result in the greater sum rate as expected. In addition, the sum rate becomes saturated as the cloud computation limit or fronthaul capacity become sufficiently large. These results imply that the proposed design framework can be employed for provisioning the cloud computation limit or fronthaul capacity and for analyzing the provisioned network resources and performance tradeoffs. This figure also illustrates the infeasible region of the bi-level problem where this infeasible region can be defined by the boundaries with $\bar{C}_{\text{cloud}}$ less than around 30 *Mbips* and $\bar{B}_k$ less than around 50 *Mbps*.

In Fig. 6 and 7, we study the profits achieved by the OPs by setting $\upsilon^{\text{InP}} = 0$. By setting $\upsilon^o = 1$ for all $o \in \Omega$, the upper-level problem becomes the profit maximization problem for all OPs whose results are illustrated in Fig. 6. In this figure, the OPs' cost (payment from all OPs to the InP), the OPs' revenue (payment of all users to the OPs), and OPs' profit (revenue minus cost) obtained by different schemes, Prop. Alg. with IR, Sl-Rlxed Greedy Alg. and Ra-Rlxed Greedy

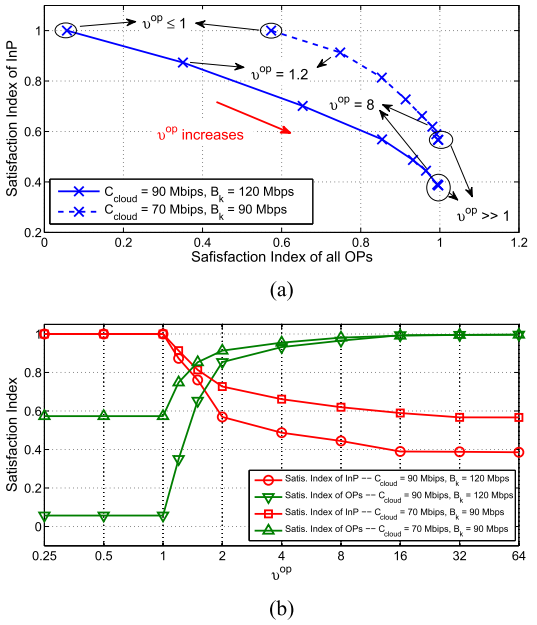

FIGURE 8. Satisfaction indexes of InP and OPs. (a) Tradeoff between satisfaction indexes of InP and OPs. (b) Satisfaction indexes of InP and OPs versus υ^{op}.

Alg., are shown versus ρ^o while ψ^o and $\beta_k^o = \beta^o, \forall k$ are set equal to one for all $o \in \Omega$. As can be seen, the OPs can attain higher profit as the price per data rate unit ρ^o increases. This is because the OPs' cost tends to saturate at sufficiently high ρ^o while the revenue scales linearly with the service price. Once again, the performance of our proposed design is demonstrated when the Prop. Alg. with IR gains higher profits in comparison to the greedy algorithms at all values of ρ^o. Interestingly, the Slicing-Relaxed Algorithm always utilize the total fronthaul capacity and computation resource fully where the corresponding total cost of all OPs are unchanged. Hence, we can see in this simulation, our proposed algorithm and Ra-Rlxed Greedy Alg. require more network resource when ρ^o increase, but they do not utilize the total network resource fully.

In Fig. 7, we study the impact of the weighting parameters on the OPs' profits. Specifically, we fix two weighting parameters as $\upsilon^1 = \upsilon^3 = 1$ while varying the value of υ^2 to obtain the curves in this figure. As expected, the profit of OP 2 increases while those of remaining OPs decrease with increasing υ^2. In addition, the total profit of all OPs is also presented and this figure indicates that the maximum profit can be achieved when $\upsilon^1 = \upsilon^2 = \upsilon^3 = 1$.

We now study a satisfaction index for InP and OPs which is defined as the ratio between the achieved profit and the maximum potential profit in Fig. 8 where ρ^o is set equal to 5 (¢/*Mbps*) and ψ^o and β^o are set equal to 1 (¢/*Mbips* and ¢/*Mbps*) for all OPs. Note that the maximum InP's profit can be calculated as $\psi^o \bar{C}_{\text{cloud}} + \beta^o \sum_{k\in\mathcal{K}} \bar{B}_k$ (¢) while the maximum profit of all OPs can be obtained by the same method employed to obtain results in Fig. 6. Fig. 8a illustrates the trade-off between the satisfaction indices of the InP and all OPs for $\upsilon^{\text{InP}} = 1$ and $\upsilon^1 = \upsilon^2 = \upsilon^3 = \upsilon^{\text{op}}$ as we vary the

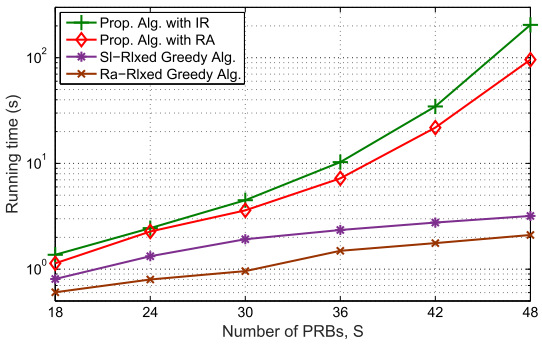

FIGURE 9. Running time vs number of PRBs (S).

value of υ^{op} while Fig. 8b shows these satisfaction indexes versus υ^{op}. As can be observed, when the satisfactory index of InP equals to one, the OPs utilize all network resources. It happens when $\upsilon^{\text{op}} \le 1$ which leads to the minimum OPs' satisfaction. Inversely, when the index corresponding to InP is less than one, i.e. $\upsilon^{\text{op}} > 1$, the total fronthaul capacity and computational resources are not fully used and OPs' satisfaction index increases. The presented tradeoff results indicate that one can determine an operating point of the two-level design framework where both the InP and OPs find it satisfactory.

We study the complexities of the proposed algorithms where we show their average running times versus the number of PRBs in Fig. 9. These running time values are obtained by averaging over 1000 runs. It can be seen that the proposed algorithms, Prop. Alg. with IR and Prop. Alg. with RA, require much longer running time in comparison with the two greedy algorithms, Sl-Rlxed Greedy Alg. and Ra-Rlxed Greedy Alg.. Moreover, the Prop. Alg. with IR has the longest running time while the Ra-Rlxed Greedy Alg. has the shortest one. In addition, the required running time increases as the number of PRBs in each cell increases as expected.

VII. CONCLUSION

We have proposed a novel algorithmic framework for uplink wireless virtualization of the C-RAN supporting multiple OPs via joint rate and quantization bit allocation for users served by each OP. This design aims to maximize the weighted sum profits of the InP and OPs considering practical constraints on the fronthaul capacity and cloud computation limits. Numerical results have illustrated that our proposed algorithms outperform the greedy resource allocation algorithms and achieve the sum rate very close to the sum rate upper-bound obtained by solving relaxed problems. We have also studied the impacts of various parameters on the system sum-rate and relevant performance tradeoffs.

APPENDIX A
PROOF OF PROPOSITION 1

If $b_k^{(s)}$ is selected so that $q_k^{(s)}(b_k^{(s)}) \le \sqrt{Y_k^{(s)} I_k^{(s)}}$, we have

$$\gamma_k^{(s)}(b_k^{(s)}) = \frac{D_k^{(s)}}{I_k^{(s)} + q_k^{(s)}(b_k^{(s)})} \ge \frac{D_k^{(s)}}{I_k^{(s)} + \sqrt{Y_k^{(s)} I_k^{(s)}}}. \quad (50)$$

Substituting $Y_k^{(s)} = D_k^{(s)} + I_k^{(s)}$ into this equality, we attain

$$\gamma_k^{(s)}(b_k^{(s)}) \geq \frac{D_k^{(s)}/I_k^{(s)}}{1+\sqrt{D_k^{(s)}/I_k^{(s)}+1}} = \frac{\bar{\gamma}_k^{(s)}}{\sqrt{\bar{\gamma}_k^{(s)}+1}+1} = \sqrt{\bar{\gamma}_k^{(s)}+1}-1 = \underline{\gamma}_k^{(s)}. \quad (51)$$

In addition, it can be verified that

- When $\bar{\gamma}_k^{(s)} \gg 1$, we have $\sqrt{\bar{\gamma}_k^{(s)}+1} \simeq \sqrt{\bar{\gamma}_k^{(s)}} \gg 1$. Thus, we have $\underline{\gamma}_k^{(s)} \simeq \sqrt{\bar{\gamma}_k^{(s)}}$.
- When $\bar{\gamma}_k^{(s)} \ll 1$, we have $\sqrt{\bar{\gamma}_k^{(s)}+1} \simeq 1+\bar{\gamma}_k^{(s)}/2$. Hence, it can be implied that $\underline{\gamma}_k^{(s)} \simeq \bar{\gamma}_k^{(s)}/2$.

This concludes the proof for Proposition 1.

APPENDIX B
PROOF OF THEOREM 1

To prove that $\chi_k^{(s)}(r,b)$ is a convex function with respect to variables (r,b), we will show that the Hessian matrix of $\chi_k^{(s)}(r,b)$ is positive definite. For simplicity, we omit the superscripts and subscripts in all notations, i.e., $\chi(r,b)$, D, I and Y stand for $\chi_k^{(s)}(r,b)$, $D_k^{(s)}$, $I_k^{(s)}$ and $Y_k^{(s)}$, respectively. First, we derive the Hessian matrix of $\chi(r,b)$. Let $\mathbf{H} = [H_{11}H_{12}, H_{21}H_{22}]$ be the Hessian matrix of $\chi(r,b)$, its elements can be written as

$$H_{11} = \frac{\partial^2\chi(r,b)}{\partial r^2} = \frac{2Ar}{(\ln 2)(Z-r)^2}, \quad (52)$$

$$H_{12} = \frac{\partial^2\chi(r,b)}{\partial r\partial b} = -\frac{2\sqrt{3}\pi ADYr}{(\ln 2)2^{2b}X(X+D)(Z-r)^2}, \quad (53)$$

$$H_{21} = \frac{\partial^2\chi(r,b)}{\partial b\partial r} = -\frac{2\sqrt{3}\pi ADYr}{(\ln 2)2^{2b}X(X+D)(Z-r)^2}, \quad (54)$$

$$H_{22} = \frac{\partial^2\chi(r,b)}{\partial b^2} = \frac{6\pi^2AD^2Y^2r}{(\ln 2)2^{4b}X^2(X+D)^2(Z-r)^2} + \frac{2\sqrt{3}\pi ADYr\left[2^{2b+1}X(X+D)-\sqrt{3}\pi Y(2X+D)\right]}{2^{4b}X^2(X+D)^2(Z-r)}, \quad (55)$$

where $X = I + \frac{\sqrt{3}\pi Y}{2^{2\,b+1}}$ and $Z = \log_2\left(1+\frac{D}{I+\frac{\sqrt{3}\pi Y}{2^{2\,b+1}}}\right)$. Let $q = \frac{\sqrt{3}\pi Y}{2^{2\,b+1}}$, then we have $X = I+q$, $X+D = Y+q$, and $\sqrt{3}\pi Y = 2^{2\,b+1}q$. Substituting these results into (55) yields

$$H_{22} = \frac{6\pi^2AD^2Y^2r}{(\ln 2)2^{4b}X^2(X+D)^2(Z-r)^2} + \frac{4\sqrt{3}\pi ADYr\left(IY-q^2\right)}{2^{2b}X^2(X+D)^2(Z-r)}. \quad (56)$$

Then, if $IY \geq q^2$, we will have $H_{11}, H_{22} > 0$, and

$$\det|\mathbf{H}| = H_{11}H_{22} - H_{12}H_{21} = \frac{8\sqrt{3}\pi A^2DYr^2\left(IY-q^2\right)}{\ln(2)2^{2b}X^2(X+D)^2(Z-r)^3} \geq 0. \quad (57)$$

Hence, the Hessian matrix of $\chi(r,b)$ is positive definite. Thus, we can conclude that $\chi(r,b)$ is jointly convex with respect to (r,b) if $IY \geq q^2$.

APPENDIX C
PROOF OF THEOREM 2

We will prove this theorem by using the definition of a concave function, i.e., $f(\phi x_1 + (1-\phi)x_2) \geq \phi f(x_1) + (1-\phi)f(x_2)$ for all $0 \leq \phi \leq 1$. Let us consider two possible values of the involved variables, $(C_1^o, \mathbf{B}_1^o)$ and $(C_2^o, \mathbf{B}_2^o)$. We assume that there exists the optimum solutions for the lower-level problems (17) corresponding to these two values of variables. Moreover, the optimal sum rates for these cases are denoted as $\bar{R}(C_1^o, \mathbf{B}_1^o)$ and $\bar{R}(C_2^o, \mathbf{B}_2^o)$, respectively with $\{r_{k,1}^{(s)}, b_{k,1}^{(s)}\}$ and $\{r_{k,2}^{(s)}, b_{k,2}^{(s)}\}$ being the optimal rates and number of quantization bits, respectively. Then, $\{r_{k,i}^{(s)}, b_{k,i}^{(s)}\}$ must satisfy all the constraints of (17) corresponding to $(C_i^o, \mathbf{B}_i^o)$ for $i = 1$ or 2. For any value of ϕ such that $0 \leq \phi \leq 1$, we define $\{r_{k,3}^{(s)}, b_{k,3}^{(s)}\}$ as

$$r_{k,3}^{(s)} = \phi r_{k,1}^{(s)} + (1-\phi)r_{k,2}^{(s)}, \quad \forall\,(k,s) \in \mathcal{K}\times\mathcal{S}_k^o, \quad (58)$$

$$b_{k,3}^{(s)} = \phi b_{k,1}^{(s)} + (1-\phi)b_{k,2}^{(s)}, \quad \forall\,(k,s) \in \mathcal{K}\times\mathcal{S}_k^o. \quad (59)$$

Since all constraint functions of problem (17) are convex, it is easy to see that $\{r_{k,3}^{(s)}, b_{k,3}^{(s)}\}$ satisfy all the constraints of (17) corresponding to $\left(C_3^o, \mathbf{B}_3^o\right)$, where $C_3^o = \phi C_1^o + (1-\phi)C_2^o$ and $\mathbf{B}_3^o = \phi\mathbf{B}_1^o + (1-\phi)\mathbf{B}_2^o$. Therefore, $\{r_{k,3}^{(s)}, b_{k,3}^{(s)}\}$ is a feasible solution of problem (17) corresponding to $(C_3^o, \mathbf{B}_3^o)$. Consequently, we have

$$\begin{aligned}\bar{R}(C_3^o, \mathbf{B}_3^o) &\geq \sum_{k\in\mathcal{K}}\sum_{s\in\mathcal{S}_k^o} r_{k,3}^{(s)} \\ &= \sum_{k\in\mathcal{K}}\sum_{s\in\mathcal{S}_k^o}\left(\phi r_{k,1}^{(s)} + (1-\phi)r_{k,2}^{(s)}\right) \\ &= \phi\bar{R}(C_1^o, \mathbf{B}_1^o) + (1-\phi)\bar{R}(C_2^o, \mathbf{B}_2^o), \end{aligned} \quad (60)$$

for any value of ϕ such that $0 \leq \phi \leq 1$. Hence, $\bar{R}(C^o, \mathbf{B}^o)$ must be a concave function with respect to $(C^o, \mathbf{B}^o)$ and we have completed the proof for Theorem 2.

APPENDIX D
PROOF OF PROPOSITION 4

It can be verified that the second derivative of the objective function of (31) is the same as that of $w(r_k^{(s)})$ which can be expressed as

$$\frac{\partial^2 w(r_k^{(s)})}{\partial r_k^{(s)2}} = -\frac{2\lambda A r_k^{(s)} + 4\lambda A\left(t(b_k^{(s)}) - r_k^{(s)}\right)}{(\ln 2)\left(t(b_k^{(s)}) - r_k^{(s)}\right)^2}, \quad (61)$$

which is less than zero if $t(b_k^{(s)}) \geq r_k^{(s)}$. Hence, this function is concave. Therefore, the optimum rate $r_k^{(s)}$ can be obtained by studying the KKT conditions. Taking the first derivative of the objective function and setting it to zero results in

$$\frac{\partial w(r)}{\partial r} = -E_k^{(s)}. \quad (62)$$

Using the constraint (31b), the optimal solution to problem (31) can be written as

$$r_k^{(s)\star} = \max\left[R_{\min}, \min\left(t(b_k^{(s)}), R_{\max}, r|_{\frac{\partial w(r)}{\partial r}=-E_k^{(s)}}\right)\right]. \tag{63}$$

Therefore, we have completed the proof of Proposition 4.

APPENDIX E
PROOF OF PROPOSITION 5

The Lagrangian of problem (33) can be expressed as

$$\mathcal{L}\left(\mathbf{b}_k^o, \mu\right) = \sum_{s\in\mathcal{S}_k^o} z\left(b_k^{(s)}\right) - \mu\left(\sum_{s\in\mathcal{S}_k^o} b_k^{(s)} - B_k^o/(2N_{\mathsf{RE}})\right), \tag{64}$$

where μ is the Lagrangian multiplier associated with the fronthaul capacity constraint of problem (33). In addition, the dual function of problem (33) can be written as

$$g(\mu) = \max_{\mathbf{b}_k^o} \mathcal{L}\left(\mathbf{b}_k^o, \mu\right) \quad \text{s. t. } b_k^{(s)} \geq J_k^{(s)}, \quad \forall s \in \mathcal{S}_k^o, \tag{65}$$

where $J_k^{(s)} = \max\left(\lceil \underline{b}_k^{(s)} \rceil, t^{-1}\left(r_k^{(s)}\right)\right)$. This problem can be decoupled into S parallel sub-problems each of which corresponds to one PRB. In addition, all these sub-problems have the same structure. Since its objective function is concave, each sub-problem can be solved by using the KKT condition $\partial\mathcal{L}\left(\mathbf{b}_k^o, \mu\right)/\partial b_k^{(s)} = 0$, which is equivalent to

$$\partial z(b_k^{(s)})/\partial b_k^{(s)} = \mu. \tag{66}$$

Using the constraint (33b), the optimal solution of $b_k^{(s)}$ must satisfy (34). In addition, the objective function is an increasing function with respect to $\mathbf{b}_k^o$; hence, the fronthaul capacity constraint (33c) must be met with equality. Therefore, μ can be determined to satisfy $\sum_{s\in\mathcal{S}_k^o} b_k^{(s)} = B_k/(2N_{\mathsf{RE}})$. Therefore, we have completed the proof of Proposition 5.

REFERENCES

[1] "Cisco visual networking index: Global mobile data traffic forecast update, 2016–2021," Cisco, White Paper, Feb. 2017.
[2] P. Cerwall *et al.* "Ericsson mobility report," Ericsson, Stockholm, Sweden, Tech. Rep., Jun. 2017.
[3] China Mobile Research Institute, "C-RAN: The road towards green ran," China Mobile, White Paper, 2011.
[4] A. Checko *et al.*, "Cloud RAN for mobile networks—A technology overview," *IEEE Commun. Surveys Tuts.*, vol. 17, no. 1, pp. 405–426, 1st Quart., 2015.
[5] D. Wubben *et al.*, "Benefits and impact of cloud computing on 5G signal processing: Flexible centralization through cloud-RAN," *IEEE Signal Process. Mag.*, vol. 31, no. 6, pp. 35–44, Nov. 2014.
[6] V. Suryaprakash, P. Rost, and G. Fettweis, "Are heterogeneous cloud-based radio access networks cost effective?" *IEEE J. Sel. Areas Commun.*, vol. 33, no. 10, pp. 2239–2251, Oct. 2015.
[7] X. Costa-Perez, J. Swetina, T. Guo, R. Mahindra, and S. Rangarajan, "Radio access network virtualization for future mobile carrier networks," *IEEE Commun. Mag.*, vol. 51, no. 7, pp. 27–35, Jul. 2013.
[8] C. Liang and F. R. Yu, "Wireless network virtualization: A survey, some research issues and challenges," *IEEE Commun. Surveys Tuts.*, vol. 17, no. 1, pp. 358–380, 1st Quart., 2015.
[9] M. Peng, C. Wang, V. Lau, and H. V. Poor, "Fronthaul-constrained cloud radio access networks: Insights and challenges," *IEEE Wireless Commun.*, vol. 22, no. 2, pp. 152–160, Apr. 2015.
[10] P. Luoto, P. Pirinen, M. Bennis, S. Samarakoon, S. Scott, and M. Latva-Aho, "Co-primary multi-operator resource sharing for small cell networks," *IEEE Trans. Wireless Commun.*, vol. 14, no. 6, pp. 3120–3130, Jun. 2015.
[11] R. Kokku, R. Mahindra, H. Zhang, and S. Rangarajan, "NVS: A substrate for virtualizing wireless resources in cellular networks," *IEEE/ACM Trans. Netw.*, vol. 20, no. 5, pp. 1333–1346, Oct. 2012.
[12] M. I. Kamel, L. B. Le, and A. Girard, "LTE wireless network virtualization: Dynamic slicing via flexible scheduling," in *Proc. IEEE VTC Fall*, Sep. 2014, pp. 1–5.
[13] X. Wang *et al.*, "Energy-efficient virtual base station formation in optical-access-enabled cloud-RAN," *IEEE J. Sel. Areas Commun.*, vol. 34, no. 5, pp. 1130–1139, May 2016.
[14] S. Luo, R. Zhang, and T. J. Lim, "Downlink and uplink energy minimization through user association and beamforming in C-RAN," *IEEE Trans. Wireless Commun.*, vol. 14, no. 1, pp. 494–508, Jan. 2015.
[15] Y. Shi, J. Zhang, and K. B. Letaief, "Group sparse beamforming for green cloud-RAN," *IEEE Trans. Wireless Commun.*, vol. 13, no. 5, pp. 2809–2823, May 2014.
[16] V. N. Ha, L. B. Le, and N.-D. Đào, "Coordinated multipoint transmission design for cloud-RANs with limited fronthaul capacity constraints," *IEEE Trans. Veh. Technol.*, vol. 65, no. 9, pp. 7432–7447, Sep. 2015.
[17] S.-H. Park, O. Simeone, O. Sahin, and S. Shamai (Shitz), "Robust layered transmission and compression for distributed uplink reception in cloud radio access networks," *IEEE Trans. Veh. Technol.*, vol. 63, no. 1, pp. 204–216, Jan. 2014.
[18] L. Liu, S. Bi, and R. Zhang, "Joint power control and fronthaul rate allocation for throughput maximization in OFDMA-based cloud radio access network," *IEEE Trans. Commun.*, vol. 63, no. 11, pp. 4097–4110, Nov. 2015.
[19] X. Rao and V. K. N. Lau, "Distributed fronthaul compression and joint signal recovery in cloud-RAN," *IEEE Trans. Signal Process.*, vol. 63, no. 4, pp. 1056–1065, Feb. 2015.
[20] T. Werthmann, H. Grob-Lipski, and M. Proebster, "Multiplexing gains achieved in pools of baseband computation units in 4G cellular networks," in *Proc. IEEE 24th Int. Symp. Pers. Indoor Mobile Radio Commun. (PIMRC)*, Sep. 2013, pp. 3328–3333.
[21] D. Sabella *et al.*, "Energy efficiency benefits of RAN-as-a-service concept for a cloud-based 5G mobile network infrastructure," *IEEE Access*, vol. 2, pp. 1586–1597, 2014.
[22] P. Rost, S. Talarico, and M. C. Valenti, "The complexity–rate tradeoff of centralized radio access networks," *IEEE Trans. Wireless Commun.*, vol. 14, no. 11, pp. 6164–6176, Nov. 2015.
[23] P. Rost, A. Maeder, M. C. Valenti, and S. Talarico, "Computationally aware sum-rate optimal scheduling for centralized radio access networks," in *Proc. IEEE GLOBECOM* Dec. 2015, pp. 1–6.
[24] V. N. Ha and L. B. Le, "Resource allocation for uplink OFDMA C-RANs with limited computation and fronthaul capacity," in *Proc. IEEE ICC*, May 2016, pp. 1–6.
[25] V. N. Ha, L. B. Le, and N.-D. Đào, "Cooperative transmission in cloud RAN considering fronthaul capacity and cloud processing constraints," in *Proc. IEEE WCNC*, Apr. 2014, pp. 1862–1867.
[26] J. Bucklew and N. Gallagher, "Two-dimensional quantization of bivariate circularly symmetric densities," *IEEE Trans. Inf. Theory*, vol. IT-25, no. 6, pp. 667–671, Nov. 1979.
[27] P. Baracca, S. Tomasin, and N. Benvenuto, "Backhaul rate allocation in uplink SC-FDMA systems with multicell processing," *IEEE Trans. Wireless Commun.*, vol. 13, no. 3, pp. 1264–1273, Mar. 2014.
[28] D. Bertsekas, *Nonlinear Programming*. Belmont, MA, USA: Athena Scientific, 1999.
[29] M. Frank and P. Wolfe, "An algorithm for quadratic programming," *Naval Res. Logist. Quar.*, vol. 3, nos. 1–2, pp. 95–110, 1956.
[30] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein, "Distributed optimization and statistical learning via the alternating direction method of multipliers," *Found. Trends Mach. Learn.*, vol. 3, no. 1, pp. 1–122, Jan. 2011.
[31] W. Yu and J. M. Cioffi, "Constant-power waterfilling: Performance bound and low-complexity implementation," *IEEE Trans. Commun.*, vol. 54, no. 1, pp. 23–28, Jan. 2006.
[32] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge, U.K.: Cambridge Univ. Press, 2004.
[33] I. Necoara and A. Patrascu, "Iteration complexity analysis of dual first-order methods for conic convex programming," *Optim. Methods Softw.*, vol. 31, no. 3, pp. 645–678, Mar. 2016.

[34] P. Kyosti *et al.* (Sep. 2007). *WINNER II Channel Models: European Commission, Deliverable IST-WINNER D1.1.2 Ver 1.2*. [Online]. Available: http://projects.celticinitiative.org/winner+/WINNER2-Deliverables/
[35] *LTE; Evolved Universal Terrestrial Radio Access (E-UTRA); Physical Channels and Modulation* document (3GPP TS 36.211 version 12.7.0 release 12), 3GPP Oct. 2015.

VU NGUYEN HA (S'11–M'17) received the B.Eng. degree from the French training program for excellent engineers in Vietnam, Ho Chi Minh City University of Technology, Vietnam, and an addendum degree from de École Nationale Supérieure des Télécommunications de Bretagne-Groupe des École des Télécommunications, Bretagne, France, in 2007, and the Ph.D. degree from the Institut National de la Recherche Scientifique-Énergie, Matériaux et Télécommunications, Université du Québec, Montréal, QC, Canada, in 2017. From 2008 to 2011, he was a Research Assistant with the School of Electrical Engineering, University of Ulsan, Ulsan, South Korea. He is currently a Post-Doctoral Fellow with École Polytechnique de Montréal, Montréal. His research interests include radio resource management and emerging enabling technologies for 5G wireless systems with special emphasis on heterogeneous small-cell networks, cloud RAN, and MIMO communications.

LONG BAO LE (S'04–M'07–SM'12) received the B.Eng. degree in electrical engineering from the Ho Chi Minh City University of Technology, Vietnam, in 1999, the M.Eng. degree in telecommunications from the Asian Institute of Technology, Thailand, in 2002, and the Ph.D. degree in electrical engineering from the University of Manitoba, Canada, in 2007. He was a Post-Doctoral Researcher with the Massachusetts Institute of Technology from 2008 to 2010 and the University of Waterloo from 2007 to 2008. Since 2010, he has been with the Institut National de la Recherche Scientifique, Université du Québec, Montréal, QC, Canada, where he is currently an Associate Professor. He is a coauthor of the books *Radio Resource Management in Multi-Tier Cellular Wireless Networks* (Wiley, 2013) and *Radio Resource Management in Wireless Networks: An Engineering Approach* (Cambridge University Press, 2017). His current research interests include 5G wireless technologies, radio resource management, cognitive radio, and smartgrids. He currently serves on the editorial boards of the IEEE TRANSACTIONS ON WIRELESS COMMUNICATIONS and the IEEE COMMUNICATIONS SURVEYS AND TUTORIALS. He was on the Editorial Board for the IEEE WIRELESS COMMUNICATIONS LETTERS from 2011 to 2016. He has served as technical program committee co-chair of the Medium Access Control track at the IEEE WCNC 2016, the Wireless Access track at the IEEE VTC 2014-Fall, the Wireless Networks track at the IEEE VTC 2011-Fall, and the Cognitive Radio and Spectrum Management track at the IEEE PIMRC 2011.