Service Multiplexing and Revenue Maximization in Sliced C-RAN Incorporated With URLLC and Multicast eMBB

Jianhua Tang, *Member, IEEE*, Byonghyo Shim, *Senior Member, IEEE*, and Tony Q. S. Quek, *Fellow, IEEE*

***Abstract*— The fifth generation (5G) wireless system aims to differentiate its services based on different application scenarios. Instead of constructing different physical networks to support each application, radio access network (RAN) slicing is deemed as a prospective solution to help operate multiple logical separated wireless networks in a single physical network. In this paper, we incorporate two typical 5G services, i.e., enhanced Mobile BroadBand (eMBB) and ultra-reliable low-latency communications (URLLC), in a cloud RAN (C-RAN), which is suitable for RAN slicing due to its high flexibility. In particular, for eMBB, we make use of multicasting to improve the throughput, and for URLLC, we leverage the finite blocklength capacity to capture the delay accurately. We envision that there will be many slice requests for each of these two services. Accepting a slice request means a certain amount of revenue (consists of long-term revenue and shot-term revenue) is earned by the C-RAN operator. Our objective is to maximize the C-RAN operator's revenue by properly admitting the slice requests, subject to the limited physical resource constraints. We formulate the revenue maximization problem as a mixed-integer nonlinear programming and exploit efficient approaches to solve it, such as successive convex approximation and semidefinite relaxation. Simulation results show that our proposed algorithm significantly saves system power consumption and receives the near-optimal revenue with an acceptable time complexity.**

***Index Terms*— URLLC, eMBB, multicast, C-RAN, network slicing.**

I. Introduction

THE services catered by the incoming fifth generation (5G) wireless systems are expected to fall into three categories [2], [3], i.e., enhanced Mobile BroadBand (eMBB), Ultra-Reliable Low-Latency Communications (URLLC), and massive Machine-Type Communications (mMTC). Specifically, eMBB requires high data rate and reliable broadband access over large areas, URLLC supports ultra-low latency transmission for small payload with a high level of reliability, and mMTC needs wireless connectivity for massive number of sporadically active Internet of Things (IoT) devices. Some works have studied these three services individually [4]–[6]. Nevertheless, how to efficiently and simultaneously support them in a shared physical system is still an unaddressed problem.

Recently, network slicing [7], [8] has been arising as a promising technique to provide flexibility and scalability for a variety of 5G services that attach with manifold technical, service and operation requirements. The main feature of network slicing is to run multiple logically separated networks as independent business operations on top of a common shared physical infrastructure [9]. With network slicing, network resources can be elastically and dynamically allocated to logical network slices according to on-demand tailored service requirements. Thus, network slicing offers the hope to resolve the aforementioned unaddressed problem. To facilitate network slicing, an agile and programmable physical network architecture is a requisite.

Cloud radio access network (C-RAN) has emerged as a prospective architecture for 5G. A typical structure of C-RAN includes three main components: remote radio heads (RRHs), fronthaul links, and baseband unit (BBU) pool. The most significant innovation point of C-RAN is that it decouples baseband processing functionalities from the RRHs and migrates these functionalities to the centralized cloud BBU pool, which consists of many general-purpose servers. With the centralized cloud BBU pool, an agile and programmable software-defined environment in the RAN side can be achieved [10].

In this work, with the merits from network slicing and C-RAN, we attempt to incorporate both eMBB and URLLC services in C-RAN.[1] That is, two different types of network slices are tailored in C-RAN to support these two different services. Particularly, we consider multicast transmission for the eMBB slice, since multicast transmission is envisioned to be a popular transmission scheme in some 5G scenarios [11]. For example, in C-RAN with Caching as a Service [12], popular contents are stored in the centralized BBU pool. This centralized content caching structure is easy to facilitate

Manuscript received June 21, 2018; revised December 6, 2018; accepted January 25, 2019. Date of publication February 11, 2019; date of current version March 15, 2019. This work was supported in part by the Korea Research Fellowship Program through the National Research Foundation of Korea (NRF) through the Ministry of Science and Information and Communications Technology under Grant 2016H1D3A1938245, in part by the NRF grant through the Korean Government (MSIP) under Grant 2014R1A5A1011478, in part by the Singapore University of Technology and Design-Zhejiang University (SUTD-ZJU) Research Collaboration under Grant SUTD-ZJU/RES/01/2016, and in part by the SUTD-ZJU Research Collaboration under Grant SUTD-ZJU/RES/05/2016. Part of this paper [1] will be presented at the 53rd IEEE International Conference on Communications (ICC), Shanghai, China, May 2019. *(Corresponding author: Byonghyo Shim.)*

J. Tang and B. Shim are with the Institute of New Media and Communications, Department of Electrical and Computer Engineering, Seoul National University, Seoul 08826, South Korea (e-mail: jianhua_tang@islab.snu.ac.kr; bshim@islab.snu.ac.kr).

T. Q. S. Quek is with Information Systems Technology and Design Pillar, Singapore University of Technology and Design, Singapore 487372 (e-mail: tonyquek@sutd.edu.sg).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/JSAC.2019.2898745

[1] mMTC service is not considered in this work.

multicast transmission among RRHs to deliver data to a user group with the same interest. Another application example is the live video streaming for some hot events, e.g. FIFA World Cup, where the video stream goes from stream server to UEs via the centralized BBU pool. In these scenarios, multicasting is much more efficient than unicasting (our simulation result in Section VII also verifies this claim).

There have been many previous efforts on network slicing. Most of them focus on the problems in the upper layers of network, such as resource orchestration [13] and service chaining [14]. When compared to the upper layers network slicing, RAN slicing has two additional difficulties:

1) **Inter-slice interference isolation:** Unlike the upper layers network slicing, whose resources in different slices can be isolated by virtualization, different slices share the same physical channel in RAN slicing. How to isolate the interference between different slices is a challenge in RAN slicing.
2) **Two timescales issue:** In order to keep pace with the upper layers network slicing, which is typically executed in the timescale of minutes to hours, a RAN slicing operation should be carried out in the same timescale. However, wireless channel changes in the timescale of millisecond, which is much shorter than the duration of a network slicing operation. Thus, how to harmonize this two timescales issue is another big challenge.

A. Related Works

URLLC is gaining increasingly research attention. In [15], recent theoretical principles in information theory to govern the transmission of short packets in URLLC have been reviewed. Reference [16] provides a high-level discussion about potential techniques to reduce end-to-end latency, such as short analog fountain codes, ultra-fast signal processing, non-orthogonal multiple access and resource reservation via resource block slicing. In [17], reducing the transmission time interval (TTI) length and hybrid automatic repeat request (HARQ) roundtrip time (RTT) is demonstrated as an effective way in the transition from LTE to URLLC. Reference [18] introduces a wireless communication protocol, built on multi-user diversity and cooperative communication, for both uplink and downlink URLLC to reach high levels of reliability. In [19], a two-phase transmission protocol is investigated to fully exploit the device-to-device (D2D) transmission for URLLC, where, in the first phase, messages are transmitted from the base station to group leaders, and then group leaders relay the messages to the other users in the second phase. She *et al.* [20] and [21] make use of queuing model for URLLC to minimize the transmit power [20] and the total bandwidth [21] respectively. However, all these works just focus on URLLC itself.

Some recent works start to explore the coexistence of URLLC and eMBB in the same physical network. [22] studies mobility management mechanisms to guarantee seamless handover in sliced RAN, which contains eMBB, URLLC and IoT slices. Reference [23] provides a communication-theoretic view on orthogonal and non-orthogonal slicing of radio resources respectively, such that eMBB, mMTC and URLLC can be served by the same physical network. Kassab *et al.* [24] incorporate eMBB and URLLC in C-RAN. They analyze the rate for eMBB and the rate, access latency, and reliability for URLLC from information-theoretic aspect respectively. Reference [25] utilizes two different slices to support the multimodal virtual reality, i.e., an eMBB slice for visual perception and a URLLC slice for haptic perception. Unfortunately, these works fail to discuss the revenue maximization issue faced by a RAN operator when receiving multiple network slice requests. This motivates our work.

In this paper, we incorporate multiple eMBB and URLLC slices in C-RAN to maximize the operator's revenue, which consists of long-term revenue and shot-term revenue, and therefore suffices both inter-slice constraints (e.g, total bandwidth of the system) and intra-slice constraints (e.g, quality-of-service (QoS) to each user).

B. Our Contributions

Our main contributions are as follows:

1) We tame the following interesting tussles in our system model:
   - **Multicast vs unicast:** The eMBB slices use multicast transmission, while the URLLC slices still rely on unicast transmission.
   - **High throughput vs low delay:** The eMBB slice aims to have a high throughput, while the URLLC slice desires a low delay per packet.
   - **Shannon's capacity vs finite blocklength capacity:** The achievable rate of eMBB slice can be captured by Shannon's capacity, while the counterpart of URLLC slice depends on finite blocklength capacity due to the small payload.
   - **Long-term vs short-term:** The slice request admission is done at the beginning of each long time slot, while beamforming vectors should be dynamically customized at each short time slot. Moreover, the overall revenue also includes both long-term revenue and shot-term revenue.
2) We propose a generic revenue framework for RAN slicing, which includes
   - **Slice request admission:** Based on the fact that there are many slice requests to the RAN, and the resources of RAN (e.g., power and bandwidth) are limited, a binary integer variable is introduced to indicate whether to accept or reject a slice request.
   - **Revenue modelling:** The objective of request admission is to maximize a RAN operator's revenue, which includes both long-term revenue and shot-term revenue. The long-term revenue is reflected by the parameters in the slice request and stays still over a long time slot span. The short term revenue is determined by the beamformers in each short time slot span, which may change slot-by-slot due to the variation of wireless channel.
3) We formulate the revenue maximization problem as a mixed-integer nonlinear programming (MINLP).

Especially, the intractable finite blocklength capacity constraint is involved in the MINLP. We make use of efficient methods to solve the thorny problem, such as semidefinite relaxation (SDR) and successive convex approximation (SCA). We also prove the tightness of SDR under certain cases. Our simulation results verify the efficiency of our proposed approach and also provide an insight that the long-term revenue and shot-term revenue are indeed intertwined with each other.

Notations: We use calligraphy letters to represent the sets, boldface lower case letters to denote the vectors, and boldface upper case letters to denote the matrices. $\|\mathbf{x}\|_2$ and $\|\mathbf{X}\|_F$ stand for the Euclidean norm and Frobenius norm respectively. $(\cdot)^H$ represents the conjugate transpose. $\mathbf{X} \succeq 0$ means that matrix $\mathbf{X}$ is Hermitian positive semidefinite. $\mathbb{C}$ and $\mathbb{R}^+$ stand for complex numbers and positive real numbers respectively. The notation $\mathcal{A}\backslash\mathcal{B}$ denotes the set $\mathcal{A}$ with its subset $\mathcal{B}$ removed. $|\mathcal{A}|$ represents the cardinality of set $\mathcal{A}$. The $\log(\cdot)$ function is the logarithm function with base 2.

II. System Model and Problem Formulation

We consider the time-slotted RAN slicing framework, which consists of long time slot (LTS) and short time slot (STS). On the one hand, at the beginning of each LTS, the operator has to decide whether to accept or reject the received network slice requests. On the other hand, at the beginning of each STS, the operator generates the beamformers. More details about the two timescales system will be elaborated in the following parts of this section.

In this work, we consider two types of network slice requests, i.e., the multicast eMBB slice request and URLLC slice request. We denote the multicast eMBB slice request set as $\mathcal{S}^e = \{1, \cdots, S^e\}$ and URLLC slice request set as $\mathcal{S}^u = \{S^e + 1, \cdots, S^e + S^u\}$, respectively. Let $\mathcal{S} = \mathcal{S}^e \bigcup \mathcal{S}^u$ and $S = S^e + S^u$. Further, we assume that each LTS contains Q equivalent STSs (in the time order of milliseconds). We call a STS as a *frame*, and denote the set of all STSs in one LTS as $\mathcal{Q} \triangleq \{1, \cdots, Q\}$.

A. Slicing Model

Each network slice request consists of the following two components:

1) **The number of users**. We use I_s^e to denote the number of users in the multicast eMBB slice s, for $s \in \mathcal{S}^e$, and I_s^u to denote the number of users in the URLLC slice s, for $s \in \mathcal{S}^u$.
2) **The QoS requirement**. For different types of network slice, the QoS requirement indicator also differs. In terms of the multicast eMBB slice, we use the minimum throughput as the QoS indicator, which is denoted as R_s, for $s \in \mathcal{S}^e$. For the URLLC slice, we are interested in the maximum packet delay, which is denoted as D_s, for $s \in \mathcal{S}^u$.

On this basis, we use the tuple $\{I_s^e,\ R_s\}$ and $\{I_s^u,\ D_s\}$ to represent a slice request of multicast eMBB slice $s \in \mathcal{S}^e$ and URLLC slice $s \in \mathcal{S}^u$ respectively. We also introduce a binary variable $\lambda_s \in \{0, 1\}$ to indicate whether a slice request $s \in \mathcal{S}$ is accepted/admitted by the operator. In particular, we set $\lambda_s = 1$ if and only if slice request s is accepted/admitted by the operator. Furthermore, we define $\mathcal{S}^{e+} \triangleq \{s : \lambda_s = 1, \forall s \in \mathcal{S}^e\}$ and $\mathcal{S}^{u+} \triangleq \{s : \lambda_s = 1, \forall s \in \mathcal{S}^u\}$ to represent the admitted eMBB slice set and the admitted URLLC slice set, respectively.

To achieve the inter-slice interference isolation and also adaptively guarantee the quality-of-service (QoS) for each slice, we leverage the flexible frequency division duplex (FDD) technique. In this scheme, the frequency resource size assigned to each user can be tailored. This concept is similar to the flexible radio framing [26]–[28], which is the evolution of the celebrated orthogonal frequency-division multiple access (OFDMA) framing. In flexible radio framing, the TTI size can be dynamically adjusted. This provides the flexibility to achieve low latency communication. In this paper, we assign b_s^e bandwidth to the multicast eMBB slice s, for $s \in \mathcal{S}^e$, and b_s^u bandwidth to the URLLC slice s, for $s \in \mathcal{S}^u$. We denote the frame duration as D, and assume each frame has a total bandwidth B.

We consider the data sharing transmission in the downlink C-RAN, this means that each user equipment¡¯s (UE's) desired data can be shared among all the coordinated RRHs. Suppose there are J coordinated RRHs, each with K antennas. We denote the set of all coordinated RRHs as $\mathcal{J} = \{1, \cdots, J\}$, UEs under eMBB slice s as $\mathcal{I}_s^e = \{1, \cdots, I_s^e\}$, and UEs under URLLC slice s as $\mathcal{I}_s^u = \{1, \cdots, I_s^u\}$. We assume that each UE is equipped with single antenna. In each slice $s \in \mathcal{S}$, the channel from RRH j to UE i during the q-th frame is denoted as $\mathbf{h}_{ij,s}(q)$, where $\mathbf{h}_{ij,s}(q) \in \mathbb{C}^K$, for $s \in \mathcal{S}$, $i \in \mathcal{I}_s^e \bigcup \mathcal{I}_s^u$ and $j \in \mathcal{J}$. Suppose that $\mathbf{h}_{ij,s}(q)$ is drawn from a certain random distribution, and this distribution is known in advance by the C-RAN operator. Note that the distribution can be measured off-line in advance by cutting-edge machine learning techniques. The random variables $\mathbf{h}_{ij,s}(q)$, for any $i \in \mathcal{I}_s^e \bigcup \mathcal{I}_s^u$, $j \in \mathcal{J}$, $s \in \mathcal{S}$, and $q \in \mathcal{Q}$, are independent and identically distributed.

It is worth mentioning that the slicing model we just introduced involves two timescales. In the beginning of a LTS, the operator has to decide whether to accept a slice request or not, and also assign the bandwidth to each UE in this LTS (if the request is accepted), based on the known channel distribution and parameters in the slice request. In the beginning of each STS, the operator appropriately generates the beamformers according to the acquired channel information in this STS. In other words, the long-term variables are λ_s, b_s^e and b_s^u, which remain unchanged till the next LTS, and the short-term variables are beamformers, which may change frame by frame.

B. Multicast eMBB Slice

In this work, we only consider the single-group multicasting [29], which means that each eMBB slice serves a group of users that having same interest. For the multicast group in slice $s \in \mathcal{S}^e$, let $u_s^e(q)$ be the data symbol to all UEs in this group during the q-th frame with $\mathbb{E}[|u_s^e(q)|^2] = 1$, and

$\mathbf{v}_{j,s}(q) \in \mathbb{C}^K$ be the transmit beamformer to all UEs in this group from RRH j at the q-th frame.

Note that, in this work, we consider transmit beamforming with perfect channel state information (CSI) to act as a performance upper bound. The case under imperfect CSI can resort to, for instance, the techniques developed in [30] and [31].

With this setup, in multicast slice s, the received signal at UE i during the q-th frame is

$$\hat{u}_{i,s}(q) = \sum_{j\in\mathcal{J}} \mathbf{h}_{ij,s}(q)^H \mathbf{v}_{j,s}(q) u^e_s(q) + \delta_{i,s}(q), \quad \forall i \in \mathcal{I}^e_s,$$

where the first term is the desired signal for UE i and $\delta_{i,s}(q) \sim \mathcal{CN}(0, \sigma^2_{i,s})$ is the additive white Gaussian noise (AWGN) at UE i of slice s. The corresponding signal-to-noise ratio (SNR) at UE i of slice s over the q-th frame is

$$\text{SNR}^e_{i,s}(q) = \frac{|\sum_{j\in\mathcal{J}} \mathbf{h}_{ij,s}(q)^H \mathbf{v}_{j,s}(q)|^2}{\sigma^2_{i,s}}. \tag{1}$$

Then, the achievable rate of multicast group $s \in \mathcal{S}^e$ at frame q is

$$r^e_s(q) \le \min_{i\in\mathcal{I}^e_s} \left\{\log(1 + \text{SNR}^e_{i,s}(q))\right\}, \quad \forall q \in \mathcal{Q}. \tag{2}$$

Note that in (2), the achievable rate of multicast group s is determined by the weakest user in the group (to ensure successful data reception even by the weakest user). To achieve this, adaptive modulation and coding (AMC) can be applied.

For any admitted eMBB slice, per-frame throughput requirement at frame $q \in \mathcal{Q}$ is

$$r^e_s(q) b^e_s \ge R_s, \quad \forall s \in \mathcal{S}^{e+}, \tag{3}$$

where R_s is the minimum per-frame throughput requirement of slice request s. In this paper, we do not consider the effect of frequency diversity on the achievable rate, which means that bandwidth allocation and achievable rate are assumed to be independent.

C. URLLC Slice

For URLLC slice $s \in \mathcal{S}^u$, let $u^u_{i,s}(q)$ be the data symbol for the i-th UE during the q-th frame with $\mathbb{E}[|u^u_{i,s}(q)|^2] = 1$, $\forall q \in \mathcal{Q}$, and $\mathbf{w}_{ij,s}(q) \in \mathbb{C}^K$ be the transmit beamformer to UE i from RRH j during the q-th frame. Suppose that the FDD is applied inside the URLLC slice. Hence, at UE i, there is no interference from other UEs.

On this basis, the received signal at UE $i \in \mathcal{I}^u_s$ in slice s during the q-th frame is

$$\bar{u}_{i,s}(q) = \sum_{j\in\mathcal{J}} \mathbf{h}_{ij,s}(q)^H \mathbf{w}_{ij,s}(q) u^u_{i,s}(q) + \delta_{i,s}(q).$$

The corresponding SNR at UE $i \in \mathcal{I}^u_s$ of slice s in the q-th frame is

$$\text{SNR}^u_{i,s}(q) = \frac{|\sum_{j\in\mathcal{J}} \mathbf{h}_{ij,s}(q)^H \mathbf{w}_{ij,s}(q)|^2}{\sigma^2_{i,s}}.$$

In URLLC, packets are typically very short, so that the achievable rate and the transmission error probability cannot be accurately captured by Shannon's capacity anymore. Instead, the achievable rate in URLLC falls in the finite blocklength channel coding regime, which is derived in [32]. Let $r^u_{i,s}(q)$ be the achievable rate of UE i in the URLLC slice $s \in \mathcal{S}^u$ at frame q and $n_{i,s}$ be the length of codeword block (in symbols). Then we have [32]

$$r^u_{i,s}(q) \le \log(1 + \text{SNR}^u_{i,s}(q)) - \sqrt{\frac{C_{i,s}(q)}{n_{i,s}}} Q^{-1}(\epsilon) \log e, \quad \forall i \in \mathcal{I}^u_s, \quad \forall q \in \mathcal{Q}, \tag{4}$$

where $Q^{-1}(\cdot)$ is the inverse of the Gaussian Q-function, $\epsilon > 0$ is the transmission error probability, and $C_{i,s}(q)$ is the *channel dispersion*[2] of UE i at frame q, given by

$$C_{i,s}(q) = 1 - \frac{1}{(1 + \text{SNR}^u_{i,s}(q))^2}. \tag{5}$$

We assume that, in URLLC, one data packet should be completely transmitted within one frame [33], i.e., $D_s \le D$, $\forall s \in \mathcal{S}^u$. Let $F_{i,s}$ be the packet length to UE i in slice s. Then, at frame q, the delay constraint for UEs in an admitted URLLC slice is

$$\max_{i\in\mathcal{I}^u_s} \frac{F_{i,s}}{r^u_{i,s}(q) b^u_{i,s}} \le D_s, \quad \forall s \in \mathcal{S}^{u+}, \ \forall q \in \mathcal{Q}, \tag{6}$$

where $b^u_{i,s}$ is the bandwidth assigned to UE i, such that $\sum_{i\in\mathcal{I}^u_s} b^u_{i,s} \le b^u_s$.

It is necessary to clarify the difference between blocklength $n_{i,s}$ and packet length $F_{i,s}$. Firstly, packet length is a terminology used in the network layer (measured in bits) and block length is a basic unit to perform channel coding in physical layer (measured in symbols). Secondly, each block consists two parts, i.e., message part and redundancy part. The message part is a fraction of a packet, and the redundancy part is introduced by channel coding. And the delay in this work is per packet delay [34].

D. Inter-Slice Constraints

Since each RRH has its maximum transmitting power E_j constraint, we have

$$\sum_{s\in\mathcal{S}^e} \lambda_s \mathbf{v}_{j,s}(q)^H \mathbf{v}_{j,s}(q) + \sum_{s\in\mathcal{S}^u} \lambda_s \sum_{i\in\mathcal{I}^u_s} \mathbf{w}_{ij,s}(q)^H \mathbf{w}_{ij,s}(q) \le E_j, \quad \forall j \in \mathcal{J}, \ \forall q \in \mathcal{Q}. \tag{7}$$

In addition, the bandwidth resources allocated to these two slices have to satisfy

$$\sum_{s\in\mathcal{S}^e} \lambda_s b^e_s + \sum_{s\in\mathcal{S}^u} \lambda_s \sum_{i\in\mathcal{I}^u_s} b^u_{i,s} \le B. \tag{8}$$

Note that our system model is different from that in [34]. Firstly, in our model, a URLLC slice should be created at the beginning of a LTS. In [34], on the other hand, a URLLC slice can be created at any STS. Secondly, in our model, eMBB traffic and URLLC traffic are isolated by FDD and the frequency bands are reserved for each of them respectively. Whereas in [34], eMBB slices use up the whole band and the URLLC traffic can dynamically superpose/puncture on it. There are both pros and cons from each of these two models:

[2]Other than in [32], here we put $\log e$ in (4) for simplicity.

- In [34], the non-orthogonal slicing approach is adopted, which provides better resource utility and timeliness. The model in [34] is suitable for URLLC applications (with burstiness) which require to set up the slice immediately, for example, vehicle-to-everything (V2X) communications.
- In our model, the orthogonal slicing approach is employed, which achieves better reliability (see Section VI-B for discussions on reliability). Our model is applicable for URLLC applications which can be scheduled, for instance, remote surgery.

Moreover, our model can be readily extended to the case that the fronthaul capacity of C-RAN is limited by utilizing the method proposed in [35].

E. Revenue Model

With the system model we just introduced, the ultimate goal of a RAN operator is to maximizing the revenue, which consists of long-term revenue and short-term revenue.

- The long-term revenue is reflected and mapped by the parameters in the network slice request. We denote $G^e(I_s^e, R_s)$ and $G^u(I_s^u, D_s)$ as the long-term revenue for the multicast eMBB slice and URLLC slice, respectively. We assume that the mapping between the slice request parameters and the long-term revenue is known by the operator. A basic mapping rule is that $G^e(I_s^e, R_s)$ is increasing with I_s^e or R_s, and $G^u(I_s^u, D_s)$ is increasing with I_s^u and decreasing with D_s. Moreover, a long-term revenue is received once a slice is admitted by the operator at the beginning of a LTS.
- The short-term revenue is obtained by saving system power consumption in each frame. Let $\mathbf{v}_s(q) = [\mathbf{v}_{1,s}(q);\ \mathbf{v}_{2,s}(q); \cdots;\ \mathbf{v}_{J,s}(q)] \in \mathbb{C}^{JK\times 1}$ and $\mathbf{w}_{i,s}(q) = [\mathbf{w}_{i1,s}(q);\ \mathbf{w}_{i2,s}(q); \cdots;\ \mathbf{w}_{iJ,s}(q)] \in \mathbb{C}^{JK\times 1}$. The short-term revenue for eMBB slice s at frame q is $\eta \sum_{j\in\mathcal{J}} \left(E_j - \mathbf{v}_{j,s}(q)^H \mathbf{v}_{j,s}(q)\right)$, where η is a constant to strike the tradeoff between long-term revenue and short-term revenue. And similarly, $\eta \sum_{j\in\mathcal{J}} \left(E_j - \sum_{i\in\mathcal{I}_s^u} \mathbf{w}_{ij,s}(q)^H \mathbf{w}_{ij,s}(q)\right)$ is expressed as the the short-term revenue for URLLC slice s at frame q.

Therefore, over one LTS, e.g., one hour, the revenue for multicast eMBB slice s is

$$U_s^e(\mathbf{v}_{j,s}(q); I_s^e, R_s) \triangleq G^e(I_s^e, R_s) + \eta \sum_{q\in\mathcal{Q}} \sum_{j\in\mathcal{J}} \left(E_j - \mathbf{v}_{j,s}(q)^H \mathbf{v}_{j,s}(q)\right),$$

and the revenue for URLLC slice s is

$$U_s^u(\mathbf{w}_{ij,s}(q); I_s^u, D_s) \triangleq G^u(I_s^u, D_s) + \eta \sum_{q\in\mathcal{Q}} \sum_{j\in\mathcal{J}} \left(E_j - \sum_{i\in\mathcal{I}_s^u} \mathbf{w}_{ij,s}(q)^H \mathbf{w}_{ij,s}(q)\right).$$

For simplicity, we drop the constant term E_j in the revenues hereafter. Then the remaining part of short-term revenue can be interpreted as the cost and, as a consequence, $U_s^e(\mathbf{v}_{j,s}(q); I_s^e, R_s)$ and $U_s^u(\mathbf{w}_{ij,s}(q); I_s^u, D_s)$ can be interpreted as the profit.

Remark 1: The long-term and short-term revenue can also be interpreted as the fixed and dynamic revenue respectively. The fixed revenue of one slice is only determined by the slice request itself, i.e., the parameters in the request. However, the dynamic revenue of one slice can be impacted by the other slices and also the channel conditions. This impact is reflected in constraints (7) and (8).

F. Problem Formulation

In the incoming 5G era, the operator will try to accept as many network slice requests as possible to maximize its overall revenue. However, due to the limitation on resources, e.g., transmit power and bandwidth, it is not possible to accept every incoming network slice request. In order to maximizing the revenue, the operator has to properly select and accept the slice requests.

Based on the analysis from the last subsection, the overall revenue gained from the accepted slice requests is

$$\sum_{s\in\mathcal{S}^e} \lambda_s U_s^e(\mathbf{v}_{j,s}(q); I_s^e, R_s) + \sum_{s\in\mathcal{S}^u} \lambda_s U_s^u(\mathbf{w}_{ij,s}(q); I_s^u, D_s).$$

In one LTS, the overall revenue maximization problem can be formulated as

$$\begin{aligned} \text{(P0)} \quad \max_{\substack{\lambda_s,\ b_s^e,\ b_{i,s}^u,\\ \mathbf{v}_s(q),\ \mathbf{w}_{i,s}(q)}} \ & \sum_{s\in\mathcal{S}^e} \lambda_s U_s^e(\mathbf{v}_{j,s}(q); I_s^e, R_s) \\ & + \sum_{s\in\mathcal{S}^u} \rho_s \lambda_s U_s^u(\mathbf{w}_{ij,s}(q); I_s^u, D_s) \\ \text{s.t. } & \lambda_s \in \{0,1\}, \quad \forall s \in \mathcal{S}, \\ & (2),\ (3),\ (4),\ (6),\ (7),\ \text{and } (8), \end{aligned} \tag{9}$$

where ρ_s is a weight to represent the priority. For example, to guarantee the availability for an urgent URLLC slice s, we set $\rho_s > 1$. In the remaining of this paper, we just let $\rho_s = 1$ without loss of generality. Note that other than the constraints listed in problem (P0), there are also some implicit constraints. Specifically, for a certain eMBB slice $s \in \mathcal{S}^e \backslash \mathcal{S}^{e+}$ (which implies slice s is not admitted by the operator), we must have $b_s^e = 0$ and $\|\mathbf{v}_s(q)\|_2 = 0$. And similarity, for a certain URLLC slice $s \in \mathcal{S}^u \backslash \mathcal{S}^{u+}$, we must have $b_{i,s}^u = 0$ and $\|\mathbf{w}_{i,s}(q)\|_2 = 0$.

Problem (P0) needs to be solved at the beginning of a LTS, i.e., at $q = 1$. Whereas, at $q = 1$, the problem is intractable since all actual channels in the future, i.e., $q = 2, 3, \cdots, Q$, are unknown. Besides, the binary variable λ_s makes problem (P0) a mixed-integer nonlinear programming (MINLP), so that the problem becomes more difficult to tackle. Moreover, constraints (2), (3), (4), and (6) are noncovex, which further complicate problem (P0). In the following sections, we propose an approach to obtain an approximate solution for problem (P0).

III. Simplifying the Two Timescales Problem

The two timescales problem (P0) is normally difficult to handle. Fortunately, we can utilize the technique in our

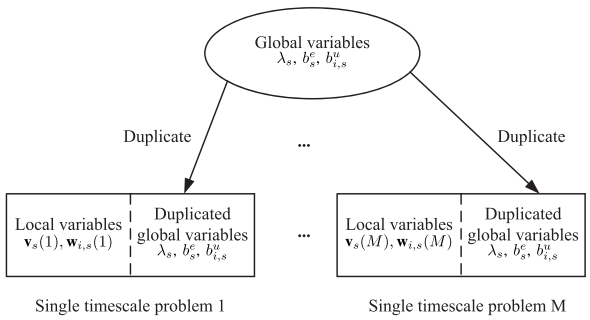


Fig. 1. A sketch of the approaches in [36] and [37].

previous works [36], [37] to decompose the two timescales problem into many single timescale subproblems. The technique in [36] and [37] has two steps:

- Firstly, we make use of the sample average approximation (SAA) to deal with unknown channels' issue. Specifically, the operator generates M samples based on the known channel distribution to approximate the unknown channels in the future.
- Secondly, we transfer the two timescales problem into a global consensus problem. Under this setting, the long-term and short-term variables in the two timescales problem are treated as the global and local variables in the global consensus problem respectively. Then, we utilize alternating direction method of multipliers (ADMM) to solve this global consensus problem. By applying ADMM, one global variable is duplicated as M auxiliary local variables. In each iteration of ADMM, we need to solve M subproblems with local (and auxiliary local) variables. Since all local variables are in the same timescale (i.e., short-time scale), each subproblem is just a single timescale problem. We briefly depict the idea of this step in Fig. 1.

From the approaches in [36] and [37], we understand that two timescales' problem can be solved on top of multiple single timescale problems. Inspired by this, we can simplify the two timescales problem (P0) as the following single timescale problem,

$$\text{(P-S)} \quad \max_{\substack{\lambda_s,\ b_s^e,\ b_{i,s}^u,\\ \mathbf{v}_s,\ \mathbf{w}_{i,s}}} \sum_{s\in\mathcal{S}^e} \lambda_s \left(\frac{1}{M} G^e(I_s^e, R_s) - \eta \sum_{j\in\mathcal{J}} \mathbf{v}_{j,s}^H \mathbf{v}_{j,s} \right) + \sum_{s\in\mathcal{S}^u} \lambda_s \left(\frac{1}{M} G^u(I_s^u, D_s) - \eta \sum_{j\in\mathcal{J}} \sum_{i\in\mathcal{I}_s^u} \mathbf{w}_{ij,s}^H \mathbf{w}_{ij,s} \right)$$

s.t. (8), (9),

$$r_s^e \le \min_{i\in\mathcal{I}_s^e} \left\{ \log(1+\text{SNR}_{i,s}^e) \right\},\ \forall s \in \mathcal{S}^{e+}, \tag{10}$$

$$r_s^e b_s^e \ge R_s,\ \forall s \in \mathcal{S}^{e+}, \tag{11}$$

$$r_{i,s}^u \le \log(1+\text{SNR}_{i,s}^u) - \sqrt{\frac{C_{i,s}}{n_{i,s}}} Q^{-1}(\epsilon) \log e, \quad \forall i \in \mathcal{I}_s^u,\ \forall s \in \mathcal{S}^{u+}, \tag{12}$$

$$\frac{F_{i,s}}{r_{i,s}^u b_{i,s}^u} \le D_s, \quad \forall i \in \mathcal{I}_s^u,\ \forall s \in \mathcal{S}^{u+}, \tag{13}$$

$$\sum_{s\in\mathcal{S}^e} \lambda_s \mathbf{v}_{j,s}^H \mathbf{v}_{j,s} + \sum_{s\in\mathcal{S}^u} \lambda_s \sum_{i\in\mathcal{I}_s^u} \mathbf{w}_{ij,s}^H \mathbf{w}_{ij,s} \le E_j, \quad \forall j \in \mathcal{J}. \tag{14}$$

Note that, in problem (P-S), we dropped all the frame number q, since it just involves one single frame. And the channel samples in problem (P-S) are generated by the operator from the idea of SAA. The variables $\{\lambda_s,\ b_s^e,\ b_{i,s}^u\}$ in problem (P-S) are actually the duplication of those in problem (P-0) (see Fig. 1), where we still keep the same notations for simplicity. In the next two sections of this paper, we aim at resolving this single timescale problem (P-S) efficiently.

Remark 2: Comparing problem (P-S) with problem (P0), we can intuitively understand problem (P-S) as follows:

- The long-term revenue is averaged to every frame, i.e., each frame has $\frac{1}{M}$ of the total long-term revenue.
- We decompose problem (P0) into M subproblems, i.e., problem (P-S), and each subproblem corresponds to one frame, i.e., single timescale.

In addition, if $M = Q = 1$, problems (P0) and (P-S) are exactly the same.

Since λ_s is a binary variable, problem (P-S) is extremely hard to tackle. In the next section, we first assume that the values of λ_s, $\forall s \in \mathcal{S}$ are given, i.e., we assume that we already have the knowledge about each network slice request is accepted or rejected. Then in Section V, we provide the approach to update λ_s, $\forall s \in \mathcal{S}$.

IV. The Single Timescale Problem

In this section, we assume that the value of λ_s is given. Although λ_s is given, problem (P-S) is still a nonconvex problem. In this section, we first solve the nonconvex problem (P-S) under the high SNR regime and then propose the approach to solve the problem under general SNR case.

Lemma 1: In problem (P-S), constraints (11) and (13) are active inequality constraints. That is, an optimal solution $\{b_s^e,\ b_{i,s}^u,\ \mathbf{v}_s,\ \mathbf{w}_{i,s}\}$ to problem (P-S) always satisfies

$$\begin{cases} r_s^e b_s^e = R_s, \\ r_{i,s}^u b_{i,s}^u = \dfrac{F_{i,s}}{D_s}. \end{cases} \tag{15}$$

Proof: Suppose that there is an optimal solution $\{b_s^e,\ b_{i,s}^u,\ \mathbf{v}_s,\ \mathbf{w}_{i,s}\}$ satisfying $r_s^e b_s^e > R_s$ or $r_{i,s}^u b_{i,s}^u > F_{i,s}/D_s$. This implies that we can still scale down $\mathbf{v}_s$ or $\mathbf{w}_{i,s}$ in the feasible region to increase the objective, which contradicts the optimality and establish the lemma. □

Based on Lemma 1, we can use $\frac{R_s}{b_s^e}$ and $\frac{F_{i,s}}{D_s b_{i,s}^u}$ to replace r_s^e and $r_{i,s}^u$ in (11) and (13) respectively. Further, let $\mathbf{V}_s = \mathbf{v}_s \mathbf{v}_s^H \in \mathbb{R}^{JK\times JK}$, $\mathbf{W}_{i,s} = \mathbf{w}_{i,s}\mathbf{w}_{i,s}^H \in \mathbb{R}^{JK\times JK}$ and $\mathbf{H}_{i,s} = \mathbf{h}_{i,s}\mathbf{h}_{i,s}^H \in \mathbb{R}^{JK\times JK}$, where $\mathbf{h}_{i,s} = [\mathbf{h}_{i1,s};\ \mathbf{h}_{i2,s}; \cdots;\ \mathbf{h}_{ij,s}] \in \mathbb{C}^{JK\times 1}$. We can reformulate constraints (11), (13) and (14) as

$$\frac{R_s}{b_s^e} \le \log(1 + \text{tr}\,(\mathbf{H}_{i,s}\mathbf{V}_s)/\sigma_{i,s}^2), \quad \forall i \in \mathcal{I}_s^e,\ \forall s \in \mathcal{S}^{e+}, \tag{16}$$

$$\frac{F_{i,s}}{b_{i,s}^u D_s} \le \log(1+\mathrm{tr}\left(\mathbf{H}_{i,s}\mathbf{W}_{i,s}\right)/\sigma_{i,s}^2) - \sqrt{1-\frac{1}{(1+\mathrm{tr}\left(\mathbf{H}_{i,s}\mathbf{W}_{i,s}\right)/\sigma_{i,s}^2)^2}}\frac{Q^{-1}(\epsilon)\log e}{\sqrt{n_{i,s}}}, \quad \forall i \in \mathcal{I}_s^u, \quad \forall s \in \mathcal{S}^{u+}, \tag{17}$$

$$E_j \ge \sum_{s\in\mathcal{S}^e}\lambda_s \mathrm{tr}\left(\mathbf{G}_j\mathbf{V}_s\right) + \sum_{s\in\mathcal{S}^u}\lambda_s\sum_{i\in\mathcal{I}_s^u}\mathrm{tr}\left(\mathbf{G}_j\mathbf{W}_{i,s}\right), \quad \forall j \in \mathcal{J}, \tag{18}$$

where $\mathbf{G}_j$ is a square matrix with $J\times J$ blocks, and each block in $\mathbf{G}_j$ is a $K\times K$ matrix. In $\mathbf{G}_j$, the block in the j-th row and j-th column is a $K\times K$ identity matrix, and all other blocks are zero matrices. Then, by applying the following property

$$\begin{cases}\mathbf{V}_s = \mathbf{v}_s\mathbf{v}_s^H \Leftrightarrow \mathbf{v}_s \succeq 0, & \mathrm{rank}(\mathbf{V}_s)\le 1,\\ \mathbf{W}_{i,s} = \mathbf{w}_{i,s}\mathbf{w}_{i,s}^H \Leftrightarrow \mathbf{W}_{i,s}\succeq 0, & \mathrm{rank}(\mathbf{W}_{i,s})\le 1,\end{cases}$$

we can obtain an equivalent formulation of problem (P-S) as

(P-S1)

$$\max_{b_s^e,\ b_{i,s}^u,\ \mathbf{V}_s,\ \mathbf{W}_{i,s}} \sum_{s\in\mathcal{S}^e}\lambda_s\left(\frac{1}{M}G^e(I_s^e,R_s)-\eta\mathrm{tr}\left(\mathbf{V}_s\right)\right) + \sum_{s\in\mathcal{S}^u}\lambda_s\left(\frac{1}{M}G^u(I_s^u,D_s)-\eta\sum_{i\in\mathcal{I}_s^u}\mathrm{tr}\left(\mathbf{W}_{i,s}\right)\right)$$

$$\text{s.t. (8), (16), (17), (18)},$$

$$\mathbf{V}_s \succeq 0, \quad \forall s\in\mathcal{S}^e, \tag{19}$$

$$\mathbf{W}_{i,s}\succeq 0, \quad \forall i\in\mathcal{I}_s^u,\ \forall s\in\mathcal{S}^u, \tag{20}$$

$$\mathrm{rank}(\mathbf{V}_s)\le 1, \quad \forall s\in\mathcal{S}^e, \tag{21}$$

$$\mathrm{rank}(\mathbf{W}_{i,s})\le 1, \quad \forall i\in\mathcal{I}_s^u,\ \forall s\in\mathcal{S}^u. \tag{22}$$

Using Lemma 1 and also after the change of variables, problem (P-S1) is simplified from problem (P-S). However, problem (P-S1) is still hard to tackle due to the nonconvexity of constraints (17), (21) and (22). In the following subsections, we leverage the following approaches to resolve the difficulties.

- For the rank constraints, we resort to the semidefinite relaxation (SDR) method. Specifically, we drop the rank constraints first, and then solve the problem without rank constraints. If the resulting $\mathbf{V}_s$ and $\mathbf{W}_{i,s}(v)$ are of rank one or zero, we conclude that the SDR is tight and no more manipulation is needed [38]. On the other hand, if the rank of resulting $\mathbf{V}_s$ or $\mathbf{W}_{i,s}(v)$ is larger than one, we must use a certain method to extract the approximate solution from it, e.g., the randomization method [39].
- For constraint (17), we start from the high SNR regime first. It is verified in [40] that, under high SNR regime, the channel dispersion in (4) approaches to 1. Hence, constraint (17) can be greatly simplified under high SNR regime. After the discussion of high SNR regime, we then propose an approximation approach for the general SNR case.

A. Solution for URLLC Slice Under High SNR Regime

Let τ be the high SNR threshold. Based on the results in [40], for $\mathrm{SNR}_{i,s}^u \ge \tau$, $C_{i,s}\approx 1$. Therefore, constraint (17) is simplified as

$$\frac{F_{i,s}}{b_{i,s}^u D_s} \le \log(1+\mathrm{tr}\left(\mathbf{H}_{i,s}\mathbf{W}_{i,s}\right)/\sigma_{i,s}^2) - \frac{Q^{-1}(\epsilon)\log e}{\sqrt{n_{i,s}}}, \quad \forall i\in\mathcal{I}_s^u, \quad \forall s\in\mathcal{S}^{u+}. \tag{23}$$

Applying SDR to problem (P-S1), we get the following problem

(P-S2)

$$\max_{b_s^e,\ b_{i,s}^u,\ \mathbf{V}_s,\ \mathbf{W}_{i,s}} \sum_{s\in\mathcal{S}^e}\lambda_s\left(\frac{1}{M}G^e(I_s^e,R_s)-\eta\mathrm{tr}\left(\mathbf{V}_s\right)\right) + \sum_{s\in\mathcal{S}^u}\lambda_s\left(\frac{1}{M}G^u(I_s^u,D_s)-\eta\sum_{i\in\mathcal{I}_s^u}\mathrm{tr}\left(\mathbf{W}_{i,s}\right)\right)$$

$$\text{s.t. (8), (16), (18), (19), (20), (23)},$$

$$\mathrm{tr}\left(\mathbf{H}_{i,s}\mathbf{W}_{i,s}\right)/\sigma_{i,s}^2 \ge \tau, \quad \forall i\in\mathcal{I}_s^u,\ \forall s\in\mathcal{S}^{u+}. \tag{24}$$

Problem (P-S2) is a convex optimization problem and can be easily solved by the interior point method, which has been always implemented in standard optimization tool boxes, e.g. CVX [41]. If we denote $\mathbf{V}_s^*$ and $\mathbf{W}_{i,s}^*$ as the optimal solution of $\mathbf{V}_s$ and $\mathbf{W}_{i,s}$ in problem (P-S2) respectively, then the following theorem shows the effectiveness of utilizing SDR in problem (P-S2).

Theorem 1: In problem (P-S2), the SDR for $\mathbf{W}_{i,s}$ is tight. That is,

$$\mathrm{rank}(\mathbf{W}_{i,s}^*)\le 1, \quad \forall i\in\mathcal{I}_s^u,\ \forall s\in\mathcal{S}^u.$$

However, the SDR for $\mathbf{V}_s$ may not be tight.

Proof: See Appendix. □

B. Solution for URLLC Slice Under General SNR

Recall the channel dispersion in (17),

$$C_{i,s} = 1-\frac{1}{\alpha_{i,s}^2}, \quad \forall i\in\mathcal{I}_s^u,\ \forall s\in\mathcal{S}^u, \tag{25}$$

where $\alpha_{i,s} = 1+\mathrm{tr}\left(\mathbf{H}_{i,s}\mathbf{W}_{i,s}\right)/\sigma_{i,s}^2$. It can be verified that $\sqrt{C_{i,s}}$ is concave w.r.t. $\alpha_{i,s} > 1$. On this basis, we can employ successive convex approximation (SCA) to tackle the nonconvex constraint (17). More details are given below.

SCA is an efficient way to solve various types of nonconvex optimization problems [42], [43]. The main idea of SCA is that, a locally tight approximation of the original problem is performed at each iteration to produce a tight convex objective function and constraint sets. In other words, instead of solving a nonconvex optimization problem directly, a series of convex optimization problem is solved iteratively to obtain an approximate solution. In this paper, we utilize the approximation functions to locally approximate the nonconvex functions based on the following assumption [37], [43].

Assumption 1: A function $\tilde{h}(x,y)$ is called as the *approximation function* for the nonconvex function $h(x)$, when the following conditions hold:

- $\tilde{h}(x,y)$ is continuous in (x,y).
- $\tilde{h}(x,y)$ is convex in x.
- The function value of $\tilde{h}(x,x)$ and $h(x)$ are consistent, i.e, $\tilde{h}(x,x)=h(x),\ \forall x$.
- The gradient $\frac{\partial \tilde{h}(x,y)}{\partial x}|_{x=y}$ and $\nabla h(x)|_{x=y}$ are consistent, i.e., $\frac{\partial \tilde{h}(x,y)}{\partial x}|_{x=y}=\nabla h(x)|_{x=y},\ \forall x$.
- $\tilde{h}(x,y)$ is an upper-bound of $h(x)$, i.e., $\tilde{h}(x,y)\geq h(x),\ \forall x,\ y$.

Applying SCA to $\sqrt{C_{i,s}}$, at iteration p, we have

$$\sqrt{C_{i,s}} \leq \sqrt{1-\frac{1}{(\alpha_{i,s}^{(p-1)})^2}}+\beta_{i,s}^{(p-1)} \times\left(1+\text{tr}\left(\mathbf{H}_{i,s}\mathbf{W}_{i,s}\right)/\sigma_{i,s}^2-\alpha_{i,s}^{(p-1)}\right), \quad (26)$$

where $\alpha_{i,s}^{(p-1)}=1+\text{tr}\left(\mathbf{H}_{i,s}\mathbf{W}_{i,s}^{(p-1)}\right)/\sigma_{i,s}^2$ and $\beta_{i,s}^{(p-1)}=(\alpha_{i,s}^{(p-1)})^{-2}((\alpha_{i,s}^{(p-1)})^2-1)^{-0.5}$ are constants obtained from the $(p-1)$-th iteration. It can be verified that the approximation in (26) satisfies Assumption 1. Thus, at iteration p, constraint (17) can be approximated as

$$\begin{aligned}&\frac{F_{i,s}}{b_{i,s}^u D_s}\\ &\leq -\left(\sqrt{1-\frac{1}{(\alpha_{i,s}^{(p-1)})^2}}+\beta_{i,s}^{(p-1)}\right.\\ &\quad\left.\times\left(1+\text{tr}\left(\mathbf{H}_{i,s}\mathbf{W}_{i,s}\right)/\sigma_{i,s}^2-\alpha_{i,s}^{(p-1)}\right)\right)\frac{Q^{-1}(\epsilon)\log e}{\sqrt{n_{i,s}}}\\ &\quad+\log(1+\text{tr}\left(\mathbf{H}_{i,s}\mathbf{W}_{i,s}\right)/\sigma_{i,s}^2),\quad \forall i\in\mathcal{I}_s^u,\ \forall s\in\mathcal{S}^{u+},\end{aligned} \quad (27)$$

which is a convex constraint.

Hence, for the general SNR scenario, if we employ SDR on problem (P-S1), the following problem is required to be solved at iteration p:

$$\begin{aligned}\text{(P-S3)}\quad &\max_{b_s^e,\ b_{i,s}^u,\ \mathbf{V}_s,\ \mathbf{W}_{i,s}}\\ &\sum_{s\in\mathcal{S}^e}\lambda_s\left(\frac{1}{M}G^e(I_s^e,R_s)-\eta\text{tr}\left(\mathbf{V}_s\right)\right)\\ &\quad+\sum_{s\in\mathcal{S}^u}\lambda_s\left(\frac{1}{M}G^u(I_s^u,D_s)-\eta\sum_{i\in\mathcal{I}_s^u}\text{tr}\left(\mathbf{W}_{i,s}\right)\right)\\ &\text{s.t. (8), (16), (18), (19), (20), and (27).}\end{aligned}$$

Note that problem (P-S3) is a convex optimization problem and can be resolved by standard tools as well.

Let $\{b_s^{e(p)},\ b_{i,s}^{u\ (p)},\ \mathbf{V}_s^{(p)},\ \mathbf{W}_{i,s}^{(p)}\}$ be the optimal solution for problem (P-S3) at the p-th iteration. We elaborate the SCA + SDR algorithm for problem (P-S1) under general SNR in Algorithm 1, in which $O^{(p)}$ is the optimal objective function value of problem (P-S1) at iteration p and $\varrho>0$ is a small constant.

The following theorem unravels the convergence of Algorithm 1 and the tightness of SDR for $\mathbf{W}_{i,s}^{(p)}$ under general SNR.

Algorithm 1 SCA + SDR Algorithm to Solve Problem (P-S1) (Under General SNR)

1: Initialization: $\left\{b_s^{e(0)},\ b_{i,s}^{u\ (0)},\ \mathbf{V}_s^{(0)},\ \mathbf{W}_{i,s}^{(0)}\right\}$.
2: Iteration $p\geq 1$: Solving problem (P-S3) with given $\left\{b_s^{e(p-1)},\ b_{i,s}^{u\ (p-1)},\ \mathbf{V}_s^{(p-1)},\ \mathbf{W}_{i,s}^{(p-1)}\right\}$, and obtain $\left\{b_s^{e(p)},\ b_{i,s}^{u\ (p)},\ \mathbf{V}_s^{(p)},\ \mathbf{W}_{i,s}^{(p)}\right\}$.
3: **if** $|O^{(p)}-O^{(p-1)}|<\varrho$ **then**
4: Problem (P-S1) achieves the approximated solution, stop iteration;
5: **else**
6: Let $p=p+1$, go to step 2.
7: **end if**
8: Output: $\left\{b_s^{e(p)},\ b_{i,s}^{u\ (p)},\ \mathbf{V}_s^{(p)},\ \mathbf{W}_{i,s}^{(p)}\right\}$.

Proposition 1: Every limit point $\mathbf{W}_{i,s}^{(\infty)}$ generated by Algorithm 1 is a stationary point of problem (P-S1). That is,

$$\lim_{p\to\infty}\left\|\mathbf{W}_{i,s}^{(p)}-\mathbf{W}_{i,s}^{(p-1)}\right\|_F=0,\quad \forall i\in\mathcal{I}_s^u,\ \forall s\in\mathcal{S}^u.$$

Furthermore, if the Slater condition holds at the limit point $\mathbf{W}_{i,s}^{(\infty)}$, then

1) $\mathbf{W}_{i,s}^{(\infty)}$ is a KKT point;
2) The SDR for $\mathbf{W}_{i,s}^{(p)}$ is asymptotically tight. That is,

$$\lim_{p\to\infty}\text{rank}\left(\mathbf{W}_{i,s}^{(p)}\right)\leq 1,\quad \forall i\in\mathcal{I}_s^u,\ \forall s\in\mathcal{S}^u.$$

Proof: A similar proof can be found in Appendix A of [37], we omit it for brevity. □

V. The Slice Request Admission Problem

In Section IV, we rely on the assumption that λ_s is given. In this section, we identify the approach to properly give the value of λ_s.

At the first glance, since λ_s is a binary variable, we may use an exhaustive search (ES) algorithm to find out the optimal $\lambda_s,\ \forall s\in\mathcal{S}$. However, the time complexity of ES algorithm is rather high, i.e., $\mathcal{O}(2^S)$, so that it is not quite practical in real world implementation. We can only leverage the results produced by the ES algorithm as the performance benchmark in our simulation part.

In what follows, we propose a low-complexity greedy admission (GA) algorithm to handle the binary variable λ_s. Before we proceed, we define two different kind of sets, i.e., temporally accepted set $\mathcal{S}^+$ and temporally rejected set $\mathcal{S}^-$. Temporally accepted set $\mathcal{S}^+$ means that all slice requests in this set has been admitted by the operator, i.e., $\lambda_s=1,\ \forall s\in\mathcal{S}^+$. In contrast, temporally rejected set $\mathcal{S}^-$ implies that all slice requests in this set has not been admitted yet, i.e., $\lambda_s=0,\ \forall s\in\mathcal{S}^-$.

In fact, from problem (P-S1), we are ready to introduce another concept, the *idealized revenue (IR)* of a certain slice request s, which is the maximum revenue can be received by the operator if the operator only accept one single slice request s and reject all other slice requests. In other words,

the IR of slice s, denoted as R_s, can be obtained by solving problem (P-S1) under general SNR case with $\lambda_s = 1$, and $\lambda_{\tilde{s}} = 0,\ \forall \tilde{s} \in \mathcal{S}\backslash s$ (we set $R_s = 0$ if problem (P-S1) is infeasible under this setting).

The main idea of GA algorithm is to iteratively add the slice requests with highest IR into $\mathcal{S}^+$. Specifically, there are two main steps of GA algorithm:

1) **Initialization**: Firstly, figuring out the IR for every slice request. In addition, initializing the temporally accepted set $\mathcal{S}^+$ as an empty set and temporally rejected set $\mathcal{S}^-$ as the full set $\mathcal{S}$.
2) **Updating**: We gradually add slice requests into $\mathcal{S}^+$. That is, at each iteration, we try to add slice request $s = \arg\max_{s\in\mathcal{S}^-} R_s$ into $\mathcal{S}^+$. If problem (P-S1) is feasible, then we update both $\mathcal{S}^+ = \mathcal{S}^+ \bigcup s$ and $\mathcal{S}^- = \mathcal{S}^-\backslash s$, otherwise, we only update $\mathcal{S}^- = \mathcal{S}^-\backslash s$. The iteration terminates until $\mathcal{S}^- = \emptyset$.

We detail the GA algorithm in Algorithm 2, whose time complexity is $\mathcal{O}(S)$.

Algorithm 2 Greedy Admission Algorithm

1: Initialization. Calculate R_s for all $s \in \mathcal{S}$. Set $\mathcal{S}^+ = \emptyset$ and $\mathcal{S}^- = \mathcal{S}$.
2: **while** $|\mathcal{S}^-| \geq 1$ **do**
3: $s^* = \arg\max_{s\in\mathcal{S}^-} R_s$;
4: Check the feasibility of problem (P-S1) if $\mathcal{S}^+ = \mathcal{S}^+ \bigcup s^*$;
5: **if** feasible **then**
6: Update $\mathcal{S}^+ = \mathcal{S}^+ \bigcup s^*$ and $\mathcal{S}^- = \mathcal{S}^-\backslash s^*$;
7: **else**
8: Update $\mathcal{S}^- = \mathcal{S}^-\backslash s^*$;
9: **end if**
10: **end while**
11: Output $\mathcal{S}^+$.

VI. Implementation Issues

In this section, we first explain how to implement our algorithm, and then discuss some possible ways to improve the reliability of URLLC slices.

A. Beamforming in Each Frame

From Section III to Section V, with the system generated channel sample, we have solved problem (P-S) to obtain the optimal values for long-term variables $\{\lambda_s,\ b_s^e,\ b_{i,s}^u\}$. However, in problem (P0), the optimal beamformers should be calculated based on the actual channels (instead of system generated channel samples) at each frame. In this subsection, we present the approach to obtain the optimal beamforming vectors based on given $\{\lambda_s,\ b_s^e,\ b_{i,s}^u\}$ (obtained by resolving problem (P-S)) and actual channels.

In each frame, problem (P-S) is reduced to the following problem with given $\{\lambda_s,\ b_s^e,\ b_{i,s}^u\}$,

$$\text{(P-B1)} \quad \min_{\mathbf{V}_s,\ \mathbf{W}_{i,s}} \sum_{s\in\mathcal{S}^{e+}} \eta \text{tr}\left(\mathbf{V}_s\right) + \sum_{s\in\mathcal{S}^{u+}} \eta \sum_{i\in\mathcal{I}_s^u} \text{tr}\left(\mathbf{W}_{i,s}\right)$$
$$\text{s.t. (16), (17), (18), (19), and (20),}$$

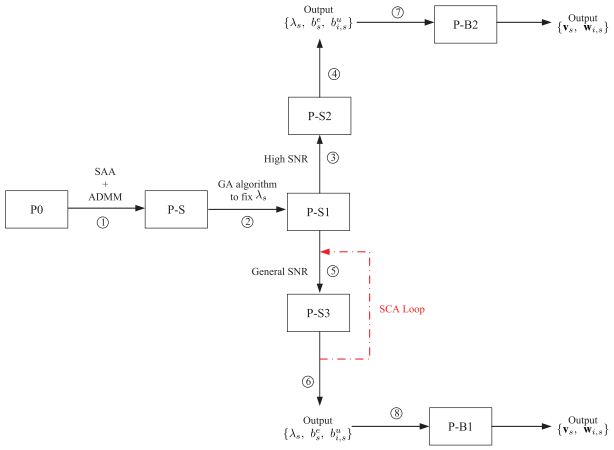


Fig. 2. The logical flow to solve problem (P0).

where channels in above constraints are now **actual channels** at this frame. Correspondingly, when URLLC slices are under high regime, problem (P-B1) is further reduced to

$$\text{(P-B2)} \quad \min_{\mathbf{V}_s,\ \mathbf{W}_{i,s}} \sum_{s\in\mathcal{S}^{e+}} \eta \text{tr}\left(\mathbf{V}_s\right) + \sum_{s\in\mathcal{S}^{u+}} \eta \sum_{i\in\mathcal{I}_s^u} \text{tr}\left(\mathbf{W}_{i,s}\right)$$
$$\text{s.t. (16), (18), (19), (20), (23), and (24).}$$

To obtain optimal $\{\mathbf{V}_s,\ \mathbf{W}_{i,s}\}$ for problem (P-B1) and problem (P-B2), we can still resort to the approaches in Section IV, in which $\{b_s^e,\ b_{i,s}^u\}$ are now constants. That is, using SCA + SDR framework as in Algorithm 1 to solve problem (P-B1) and simply using SDR to solve problem (P-B2). We denote the optimal solution for problem (P-B1) or (P-B2) as $\{\bar{\mathbf{V}}_s,\ \bar{\mathbf{W}}_{i,s}\}$. The remaining problem is how to recover the beamforming vectors $\{\mathbf{v}_s,\ \mathbf{w}_{i,s}\}$ from matrices $\{\bar{\mathbf{V}}_s,\ \bar{\mathbf{W}}_{i,s}\}$.

From Theorem 1 and Proposition 1, we acquire the result that $\bar{\mathbf{W}}_{i,s}$ is of rank one and $\bar{\mathbf{V}}_s$ may not be of rank one. Then,

1) By applying eigen decomposition on $\bar{\mathbf{W}}_{i,s}$, we obtain the beamforming vectors $\mathbf{w}_{i,s}$.
2) If $\bar{\mathbf{V}}_s$ is of rank one, we can still obtain beamforming vectors $\mathbf{v}_s$ by employing eigen decomposition. Otherwise, the randomization/scaling method [44] is utilized to generate a suboptimal solution.

We show the entire logic flow to solve problem (P0) in Fig. 2. In Fig. 2, there is no specific guideline to determine whether a URLLC slice falls in a high SNR regime or not. It depends on the operator's choice. That is, if the operator chooses to use the upper route (lower route), that means the operator forces the UEs' SNR into the high regime (general regime). It leads to a relatively simple (complicated) solution approach, but results in a lower (higher) short-term revenue (the corresponding simulation results are shown in Section VII).

We call the approach to solve problem (P0) under general SNR as the *Idealized Revenue with General SNR (IRGS)*, i.e., logic flow ① → ② → ⑤ → ⑥ → ⑧ in Fig. 2.

B. Improving the Reliability

To ensure the reliability for URLLC, we have made the following efforts so far:

- Coordinated multipoint (CoMP), which is a well-known technique to boost the received SNR. In C-RAN, owing to its special architecture (i.e., one centralized BBU pool connects to multiple RRHs), it is easy to implement CoMP on it. In our system, we employed beamforming on data sharing transmission (see Section II), which is one of the typical CoMP technique, to provide each UE a high received SNR.
- Flexible FDD. Interference is one of the main factors affecting reliability. To avoid both inter-slice and intra-slice interference, in this work, we allocated an individual frequency band for each UE in the URLLC slice.

However, still it may not be enough to achieve the vision of URLLC, whose requirement on successful packet delivery rate is as high as 99.999% [45]. For example, the CSI acquired for problem (P-B1) and (P-B2) may be imperfect/outdated. To further improve the reliability, we can implement the following enabler techniques:

- In the physical layer, adopting low-rate codes which have enough redundancy is an effective way to improve the reliability, especially when channel is in a deep fade. In the presence of CSI imperfection, exploiting the diversity is of great importance. Orthogonal space-time block coding is regarded as an outstanding diversity-seeking approach that can be leveraged to achieve high reliability.
- In the MAC layer, when an outage happens, HARQ can be applied to realize high reliability via sufficient retransmissions. In the mean time, to avoid the violation of the latency requirement when retransmission is conducted, short TTI and short frame structure is also paramount, which need to be carefully redesigned.

In addition, by making use of the *effective bandwidth* theory, our proposed approach can also extend to the case that the URLLC slice with probabilistic delay constraint. That is, instead of (6), we can impose the following constraint

$$\Pr\{\tilde{D}_{i,s}(q) \geq D_s\} \leq \pi_s, \quad \forall s \in \mathcal{S}^{u+}, \ \forall i \in \mathcal{I}_s^u, \ q \in \mathcal{Q}, \tag{28}$$

where $\tilde{D}_{i,s}(q)$ is the packet delay of the i-th UE in URLLC slice s at frame q, π_s is the maximum packet delay violation probability requirement on URLLC slice s. More details can be found in [20] and [46].

From the above discussion, we can also infer that there is a tradeoff between revenue and reliability. In particular, higher reliability may require more coding redundancy or retransmissions, which in turn means that the power consumption in each frame may increase, resulting in the reduction of the (short-term) revenue.

Remark 3: It is worth mentioning that a URLLC slice request is not a packet in the URLLC traffic.[3] A URLLC slice request is generated from the third-party to provide a certain service (e.g., the remote surgery service). And the goal of a URLLC slice request is to set up a (virtual) network slice. The time from sending out the slice request to the reception of the result can be larger than 1 ms. Once the URLLC slice request is accepted, a URLLC network slice is then created. Then the packet going through the URLLC slice should be of very low latency (i.e., < 1 ms). In this paper, Algorithm 2 aims to set up the network slice. Once the slice is created, we just need to solve problem (P-B1) or (P-B2) to obtain the optimal beamforming, which is very fast.

[3]This remark also applies to the eMBB slice.

TABLE I
SIMULATION PARAMETERS

Parameter	Value	Parameter	Value	Parameter	Value
J	3	K	2	η	5
E	1 W	F	500 bytes	$\tilde{a}$	10
σ^2	-83.98 dBm/Hz	M	100	$\tilde{b}$	0.2
n	168	τ	7 dB	B	10 MHz

VII. SIMULATION RESULTS

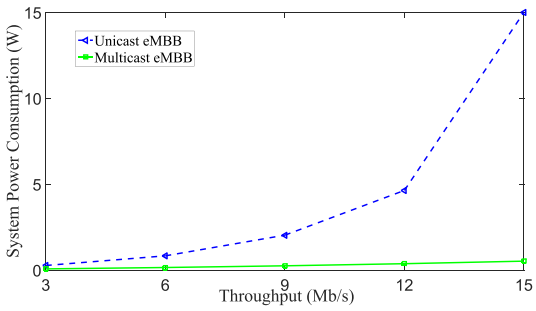
In our simulation, we use $G^e(I_s^e, R_s) = \tilde{a} I_s^e \log_{10}(1 + R_s)$ and $G^u(I_s^u, D_s) = \tilde{b} I_s^u / (1 - e^{-D_s})$ to capture the long-term revenues (R_s is in Mb/s and D_s is in Sec), where $\tilde{a}$ and $\tilde{b}$ are constants. This mapping is in accordance with the commonly used network slicing revenue model [47].

We consider a C-RAN with 3 RRHs, which are located on a circle with radius 0.5 km. The distances between each two RRHs are equal. UEs from different slices are randomly, uniformly and independently distributed within this disk. The received power at a UE located d km away from a RRH is given by p (dB) $= 128.1 + 37.6 \log_{10} d$. The transmit antenna gain at each RRH is 5 dB. The lognormal shadowing parameter is 10 dB. In our simulations, we consider homogeneous RRHs with $E_j = E$, $\forall j$, and homogeneous UEs with $\sigma_{i,s}^2 = \sigma^2$, $F_{i,s} = F$, and $n_{i,s} = n$, $\forall i, \ s$. Our default simulation parameters [48] are summarized in Table I.

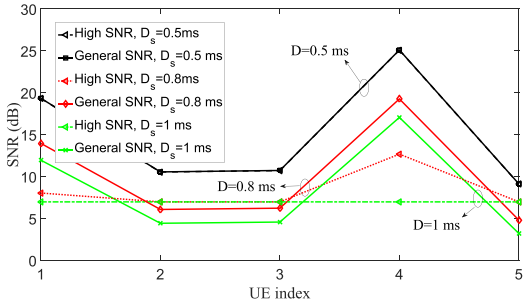
A. Single Slice Results

To comprehensively understand the effectiveness of our proposed model and algorithm, we first show the performance result when the system only supports one slice, i.e., one eMBB slice or one URLLC slice respectively.

Firstly, we assume that the system only supports one eMBB slice, i.e., $S^e = 1$ and $S^u = 0$. We are interested in examining the system power consumption (multiplied by η) when this eMBB slice uses difference schemes, i.e., unicast or multicast respectively. To avoid interference between different UEs in unicast eMBB, we also apply FDD (as we did for URLLC slice). In Fig. 3(a), we show the system power consumption under different throughput requirements R_s (there are 4 UEs in this slice). From Fig. 3(a), we can observe that the system power consumption of unicast eMBB is much higher than multicast eMBB. When the throughput requirement increases, system power consumption of unicast eMBB goes up almost exponentially, while system power consumption of multicast eMBB grows up slightly. This is because, under the multicast scheme, every UE makes full use of the whole bandwidth B. In contrast, under the unicast scheme, every UE just partially use the bandwidth, then the RRH side has to spend more transmit power on each UE to satisfy the throughput requirement.



(a) Power consumption comparison.



(b) SNR values for UEs in URLLC slice.

Fig. 3. Single slice results.

TABLE II
POWER CONSUMPTION FOR URLLC SLICE

	High SNR	**General SNR**
$D_s = 0.5$ ms	4.27 W	4.27 W
$D_s = 0.8$ ms	1.89 W	1.50 W
$D_s = 1$ ms	1.88 W	1.02 W

Secondly, we assume that the system only supports one URLLC slice, i.e., $S^e = 0$ and $S^u = 1$. We are interested in investigating the system power consumption (multiplied by η) and SNR values when apply different SNR regime solution approaches, i.e., high SNR regime (in Section IV-A) or general SNR (in Section IV-B) respectively. We show the result under three different delay requirements in Fig. 3(b), i.e., $D_s = 0.5$ ms, 0.8 ms and 1 ms respectively, under the high SNR threshold $\tau = 7$ dB. We observe from Fig. 3(b) that, for the general SNR solution approach, the optimal SNR value for each UE can be either larger or smaller than τ when delay requirements are not very stringent, i.e., $D_s =$ 0.8 ms and 1 ms. However, when the delay requirement becomes stringent, i.e., $D_s = 0.5$ ms, every UE's SNR should be larger than τ, and the curves of high SNR and general SNR overlap. In addition, we also show that the system power consumption from two different SNR regime solution approaches in Table II. From these, we can conclude that, compared to the high SNR solution approach, the general SNR solution approach saves much power consumption when delay requirements are not very stringent.

B. Multiple Slices Results

To further understand the effectiveness of our proposed algorithm in slice request admission, we then show the performance result when the system supports multiple slices. In our simulation, we consider 3 multicast eMBB slice requests and 2 URLLC slice requests (hence slice request 1, 2, and 3 stand for multicast eMBB slices, and slice request 4 and 5 stand for URLLC slices). Our default slice request parameters are summarized in Table III.

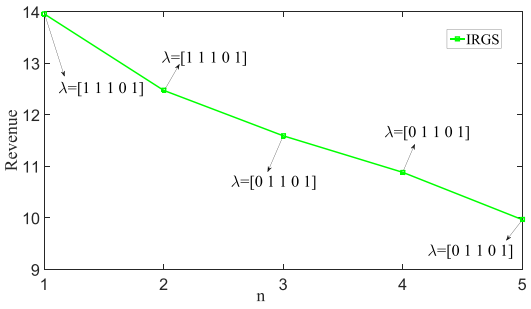
TABLE III
SLICE REQUEST PARAMETERS

eMBB slices			URLLC slices	
$\{I_1^e,\ R_1\}$	$\{I_2^e,\ R_2\}$	$\{I_3^e,\ R_3\}$	$\{I_1^u,\ D_1\}$	$\{I_2^u,\ D_2\}$
{4, 6 Mb/s}	{6, 4 Mb/s}	{8, 2 Mb/s}	{3, 1 ms}	{5, 2 ms}

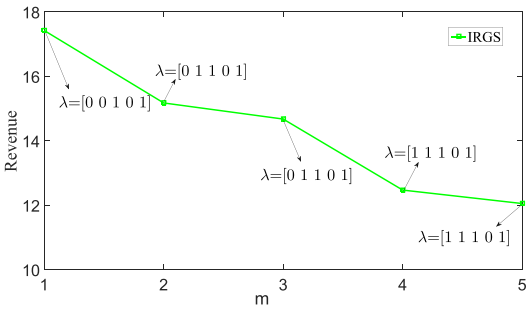
To show the optimality of our proposed IRGS algorithm, we compare our simulation results with the following benchmark algorithms:

- *Exhaustive Search with General SNR (ESGS)*. As mentioned in Section V, we can find out the optimal solution for the binary variable λ_s, $\forall s \in \mathcal{S}$, by exhaustive search, with the searching time complexity $\mathcal{O}(2^S)$. ESGS follows the logic flow ① → ② → ⑤ → ⑥ → ⑧ in Fig. 2, but uses a sophisticated exhaustive search in ②.
- *Long-term Revenue with General SNR (LRGS)*. Different from IRGS, which uses the maximum IR as the admission criteria (see Step 3 in Algorithm 2), LRGS just simply utilizes the long-term revenue as the admission criteria. Specifically, here we use G_s to denote the long-term revenue of slice $s \in \mathcal{S}$, and in each iteration, we try to accept slice request s with the maximum long-term revenue, i.e., $s = \arg\max_{s \in \mathcal{S}^-} G_s$. Then the rest steps just follow Algorithm 2. In other words, comparing to IRGS and ESGS, LRGS also corresponds to the logic flow ① → ② → ⑤ → ⑥ → ⑧ in Fig. 2, but just with a simplification in ②. The time complexity of searching is $\mathcal{O}(S)$.
- *Idealized Revenue with always High SNR (IRHS)*. As discussed in Section IV-A, problem (P-S1) can be significantly simplified to problem (P-S2), when SNR requirements for URLLC slices are high. Thus in IRHS, other than IRGS (which solves problem (P-S1) under general SNR regime), we always force SNRs for URLLC slices to fall in the high regime. IRHS corresponds to the logic flow ① → ② → ③ → ④ → ⑦ in Fig. 2.

We start from studying the request admission behaviors when the QoS requirement changes. Let $\boldsymbol{\lambda} = [\lambda_1\ \lambda_2\ \cdots\ \lambda_5]$. In Fig. 4(a), we change the QoS requirement for eMBB slices as $R_1 = 3\ n$ Mb/s, $R_2 = 2\ n$ Mb/s, and $R_1 = n$ Mb/s, where n is a varying parameter indicated in Fig. 4(a), and keep the URLLC slice requests fixed (as the values in Table III). We can see from Fig. 4(a), when $n = 1$ and 2, the admitted slices are the same, i.e., slice 1, 2, 3 and 5, but the overall revenue decreases from $n = 1$ to $n = 2$. The reason is that the system has to increase the transmit power to satisfy the increased throughput requirements for slices 1, 2, and 3, which leads to a decreased the short-term revenue. Although the long-term revenue increases, the short-term revenue dominates in this case. In addition, slice 1 is removed from the admitted set

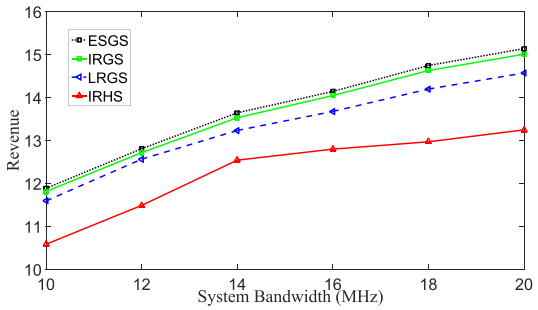


(a) Change R_s for eMBB slices.

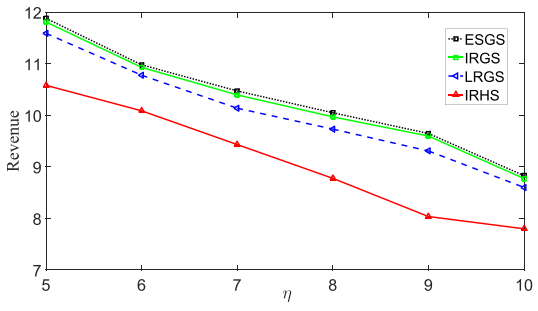


(b) Change D_s for URLLC slices.

Fig. 4. Changing slice requests' QoS requirement.



(a) Varying system bandwidth B.



(b) Varying coefficient η.

Fig. 5. Comparison with benchmark algorithms.

when n changes from 2 to 3, since the system cannot support slice 1, 2, 3 and 5 simultaneously anymore when $n \geq 3$.

Similarly, we change the QoS requirement for URLLC slices as $D_1 = 0.00025\ m$ Sec, and $D_2 = 0.0005\ m$ Sec, where m is a varying parameter indicated in Fig. 4(b), and keep the multicast eMBB slice requests fixed (as the values in Table III). We can see from Fig. 4(b), when $m = 2$, and 3, the admitted slices are the same, i.e., slice 2, 3 and 5, but the overall revenue still decreases from $m = 2$ to $m = 3$. This is because that the long-term revenue of slice 5 decreases. Although the short-term revenue increases, the long-term revenue dominates in this case. In addition, slice 1 is added into the admitted set when m changes from 3 to 4, since the system now can support slice 1, 2, 3 and 5 simultaneously when $m \geq 4$.

From Fig. 4, we obtain an interesting conclusion that the overall revenue depends on the interplay of long-term and short-term revenue. To be more specific, the long-term revenue depends on two parameters/components, i.e., the number of users and QoS requirement. If we change any one of these two parameters, the long-term revenue changes, and interestingly, and the short-term revenue changes as well. For example, if we add one more user to any eMBB slice, the whole system is then updated (i.e., the optimal bandwidth allocation and power consumption should be recalculated based on the new system-wide CSI). Thus, it is nontrivial to conclude that whether long-term or short-term revenue dominates under a specific scenario, other than that *they are intertwined*. That means, either of them can dominate the overall revenue under different scenarios.

In Fig. 5, we compare the revenue obtained by applying different algorithms, in which the slice request parameters are shown in Table III. The values in Fig. 5 are averaged from $M = 100$ random channel realizations. We conclude that IRGS outperforms the benchmark algorithms LRGS and IRHS, and performs closely to the optimal algorithm ESGS, which acts as the upper bound. In Fig. 5(a), the revenues from all algorithms increase with system bandwidth B, since the system power consumption can be reduced (and hence the short-term revenue increases) with B. In Fig. 5(b), the revenues from all algorithms decrease with the coefficient η, since the system power consumption increases with the increasing η (and hence the short-term revenue decreases).

VIII. Conclusion

In this paper, we incorporated multiple URLLC and multicast eMBB slices in C-RAN. We maximized the C-RAN operator's revenue, which includes both long-term and short-term revenues, by properly admitting the slice requests. The revenue maximization problem was formulated as a two timescales MINLP and solved by applying some efficient approaches, such as SCA and SDR. We verified from the simulation that, regarding the slice request admission, both long-term and short-term revenues account. And the performance advances of our proposed algorithm with respect to the power consumption and revenue gain were also examined by simulation.

In the future, we will consider non-orthogonal slicing (which means two different slices may interfere each other), to further improve the resource utilization in our problem.

Appendix
Proof of Theorem 1

On the one hand, for $s \in \mathcal{S}^e \backslash \mathcal{S}^{e+}$, we have $\|\mathbf{V}_s^*\|_2 = 0$, and for $s \in \mathcal{S}^u \backslash \mathcal{S}^{u+}$, we have $\left\|\mathbf{W}_{i,s}^*\right\|_2 = 0$. Thus, the SDR is tight for both $\mathbf{W}_{i,s}$ and $\mathbf{V}_s$ under this case.

On the other hand, for $s \in \mathcal{S}^{e+} \bigcup \mathcal{S}^{u+}$, problem (P-S2) is equivalent to the following problem

(P-S2')

$$\min_{b_s^e,\, b_{i,s}^u,\, \mathbf{V}_s,\, \mathbf{W}_{i,s}} \sum_{s\in\mathcal{S}^{e+}} \eta \text{tr}\left(\mathbf{V}_s\right) + \sum_{s\in\mathcal{S}^{u+}} \eta \sum_{i\in\mathcal{I}_s^u} \text{tr}\left(\mathbf{W}_{i,s}\right)$$
$$\text{s.t. (8), (16), (18), (19), (20), (23) and (24).}$$

The Lagrangian for problem (P-S2') is (we only include the terms that relevant to this proof),

$$\begin{aligned}\mathscr{L} = & \sum_{s\in\mathcal{S}^{e+}} \eta \text{tr}\left(\mathbf{V}_s\right) + \eta \sum_{s\in\mathcal{S}^{u+}} \sum_{i\in\mathcal{I}_s^u} \text{tr}\left(\mathbf{W}_{i,s}\right) \\ & - \sum_{i\in\mathcal{I}_s^e} \sum_{s\in\mathcal{S}^{e+}} \mu_{i,s} \left(\log(1+\text{tr}\left(\mathbf{H}_{i,s}\mathbf{V}_s\right)/\sigma_{i,s}^2)\right) \\ & + \sum_{j\in\mathcal{J}} \xi_j \left(\sum_{s\in\mathcal{S}^{e+}} \text{tr}\left(\mathbf{G}_j\mathbf{V}_s\right) + \sum_{s\in\mathcal{S}^{u+}} \sum_{i\in\mathcal{I}_s^u} \text{tr}\left(\mathbf{G}_j\mathbf{W}_{i,s}\right) \right) \\ & - \sum_{s\in\mathcal{S}^{e+}} \mathbf{\Gamma}_s\mathbf{V}_s - \sum_{i\in\mathcal{I}_s^e} \sum_{s\in\mathcal{S}^{e+}} \mathbf{\Omega}_{i,s}\mathbf{W}_{i,s} \\ & - \sum_{i\in\mathcal{I}_s^e} \sum_{s\in\mathcal{S}^{e+}} \zeta_{i,s} \log(1+\text{tr}\left(\mathbf{H}_{i,s}\mathbf{W}_{i,s}\right)/\sigma_{i,s}^2) \\ & - \sum_{i\in\mathcal{I}_s^e} \sum_{s\in\mathcal{S}^{e+}} \phi_{i,s} \text{tr}\left(\mathbf{H}_{i,s}\mathbf{W}_{i,s}\right)/\sigma_{i,s}^2,\end{aligned}$$

where $\mu_{i,s} \geq 0$, $\xi_j \geq 0$, $\mathbf{\Gamma}_s \succeq \mathbf{0}$, $\mathbf{\Omega}_{i,s} \succeq \mathbf{0}$, $\zeta_{i,s} \geq 0$, $\phi_{i,s} \geq 0$ are Lagrange multipliers for constraints (16), (18), (19), (20), (23), and (24) respectively. $\mathbf{\Gamma}_s$ and $\mathbf{\Omega}_{i,s}$ are both $JK \times JK$ matrices. Then

$$\begin{aligned}\frac{\partial\mathscr{L}}{\partial\mathbf{W}_{i,s}} = & \eta\mathbf{I} - \frac{\zeta_{i,s}}{\sigma_{i,s}^2 \ln 2}(1+\text{tr}\left(\mathbf{H}_{i,s}\mathbf{W}_{i,s}\right)/\sigma_{i,s}^2)^{-1}\mathbf{H}_{i,s} \\ & - \frac{\phi_{i,s}}{\sigma_{i,s}^2 \ln 2}\mathbf{H}_{i,s} + \sum_{j\in\mathcal{J}} \xi_j\mathbf{G}_j - \mathbf{\Omega}_{i,s}, \end{aligned} \tag{29}$$

$$\begin{aligned}\frac{\partial\mathscr{L}}{\partial\mathbf{V}_s} = & \eta\mathbf{I} - \sum_{i\in\mathcal{I}_s^e} \frac{\mu_{i,s}}{\sigma_{i,s}^2 \ln 2}(1+\text{tr}\left(\mathbf{H}_{i,s}\mathbf{V}_s\right)/\sigma_{i,s}^2)^{-1}\mathbf{H}_{i,s} \\ & + \sum_{j\in\mathcal{J}} \xi_j\mathbf{G}_j - \mathbf{\Gamma}_s, \end{aligned} \tag{30}$$

where $\mathbf{I}$ is a $JK \times JK$ identity matrix.

Since $\mathbf{W}_{i,s}^*$ is the optimal solution, we have

$$\frac{\partial\mathscr{L}}{\partial\mathbf{W}_{i,s}^*} = \mathbf{0}, \tag{31}$$

and

$$\mathbf{\Omega}_{i,s}\mathbf{W}_{i,s}^* = \mathbf{0}. \tag{32}$$

Combing (29) and (31), which yields

$$\begin{aligned}\mathbf{\Omega}_{i,s} = & \eta\mathbf{I} + \sum_{j\in\mathcal{J}} \xi_j\mathbf{G}_j - \frac{1}{\sigma_{i,s}^2 \ln 2} \\ & \times \left(\zeta_{i,s}(1+\text{tr}\left(\mathbf{H}_{i,s}\mathbf{W}_{i,s}^*\right)/\sigma_{i,s}^2)^{-1} + \phi_{i,s}\right)\mathbf{H}_{i,s}. \end{aligned} \tag{33}$$

In the right hand side of (33), the first two terms construct a matrix with full rank, i.e., rank $= JK$. In addition, in (33), the coefficient of the last term $\mathbf{H}_{i,s}$ is negative. And recalling that $\mathbf{\Omega}_{i,s} \succeq \mathbf{0}$, and $\text{rank}(\mathbf{H}_{i,s}) \leq 1$, we can conclude

$$\text{rank}(\mathbf{\Omega}_{i,s}) \geq JK - 1. \tag{34}$$

Further, combing (32) and (34), we can obtain $\text{rank}(\mathbf{W}_{i,s}^*) \leq 1$.

Similar to (33), we can also get,

$$\begin{aligned}\mathbf{\Gamma}_s = & \eta\mathbf{I} + \sum_{j\in\mathcal{J}} \xi_j\mathbf{G}_j - \sum_{i\in\mathcal{I}_s^e} \frac{\mu_{i,s}}{\sigma_{i,s}^2 \ln 2} \\ & \times (1+\text{tr}\left(\mathbf{H}_{i,s}\mathbf{V}_s^*\right)/\sigma_{i,s}^2)^{-1}\mathbf{H}_{i,s}. \end{aligned} \tag{35}$$

However, in the right hand side of (35), the third term is the summation of multiple rank one matrices. Therefore, we cannot claim $\text{rank}(\mathbf{\Gamma}_s) \geq JK - 1$, and as a result, we cannot conclude that $\text{rank}(\mathbf{V}_s^*) \leq 1$.

This completes the proof.

REFERENCES

[1] J. Tang, B. Shim, T.-H. Chang, and T. Q. S. Quek, "Incorporating URLLC and Multicast eMBB in sliced cloud radio access network," in *Proc. IEEE ICC*, Shanghai, China, May 2019, pp. 1–7.

[2] *Study on New Radio Access Technology Physical Layer Aspect (Release 14)*, document TR 38.802, 3GPP, Mar. 2017.

[3] M. Shafi *et al.*, "5G: A tutorial overview of standards, trials, challenges, deployment, and practice," *IEEE J. Sel. Areas Commun.*, vol. 35, no. 6, pp. 1201–1221, Jun. 2017.

[4] S. A. Hashemi, C. Condo, F. Ercan, and W. J. Gross, "On the performance of polar codes for 5G eMBB control channel," in *Proc. 51st Asilomar Conf. Signals, Syst., Comput. (ASILOMAR)*, Pacific Grove, CA, USA, Oct./Nov. 2017, pp. 1764–1768.

[5] H. Ji, S. Park, J. Yeo, Y. Kim, J. Lee, and B. Shim, "Ultra-reliable and low-latency communications in 5G downlink: Physical layer aspects," *IEEE Wireless Commun.*, vol. 25, no. 3, pp. 124–130, Jun. 2018.

[6] Z. Dawy, W. Saad, A. Ghosh, J. G. Andrews, and E. Yaacoub, "Toward massive machine type cellular communications," *IEEE Wireless Commun.*, vol. 24, no. 1, pp. 120–128, Feb. 2017.

[7] K. Samdanis, X. C. Perez, and V. Sciancalepore, "From network sharing to multi-tenancy: The 5G network slice broker," *IEEE Commun. Mag.*, vol. 54, no. 7, pp. 32–39, Jul. 2016.

[8] X. Foukas, G. Patounas, A. Elmokashfi, and M. K. Marina, "Network slicing in 5G: Survey and challenges," *IEEE Commun. Mag.*, vol. 55, no. 5, pp. 94–100, May 2017.

[9] *Description of Network Slicing Concept Version 1.0*, NGMN, Frankfurt, Germany, Jan. 2016.

[10] J. Tang, R. Wen, T. Q. S. Quek, and M. Peng, "Fully exploiting cloud computing to achieve a green and flexible C-RAN," *IEEE Commun. Mag.*, vol. 55, no. 11, pp. 40–46, Nov. 2017.

[11] M. Tao, E. Chen, H. Zhou, and W. Yu, "Content-centric sparse multicast beamforming for cache-enabled cloud RAN," *IEEE Trans. Wireless Commun.*, vol. 15, no. 9, pp. 6118–6131, Sep. 2016.

[12] J. Tang and T. Q. S. Quek, "The role of cloud computing in content-centric mobile networking," *IEEE Commun. Mag.*, vol. 54, no. 8, pp. 52–59, Aug. 2016.

[13] R. Wen, J. Tang, T. Q. S. Quek, G. Feng, G. Wang, and W. Tan, "Robust network slicing in software-defined 5G networks," in *Proc. IEEE GLOBECOM*, Singapore, Dec. 2017, pp. 1–6.

[14] N. Zhang, Y.-F. Liu, H. Farmanbar, T.-H. Chang, M. Hong, and Z.-Q. Luo, "Network slicing for service-oriented networks under resource constraints," *IEEE J. Sel. Areas Commun.*, vol. 35, no. 11, pp. 2512–2521, Nov. 2017.

[15] G. Durisi, T. Koch, and P. Popovski, "Toward massive, ultrareliable, and low-latency wireless communication with short packets," *Proc. IEEE*, vol. 104, no. 9, pp. 1711–1726, Aug. 2016.

[16] H. Chen *et al.*, "Ultra-reliable low latency cellular networks: Use cases, challenges and approaches," *IEEE Commun. Mag.*, vol. 56, no. 12, pp. 119–125, Dec. 2018.

[17] G. Pocovi, B. Soret, K. I. Pedersen, and P. Mogensen, "MAC layer enhancements for ultra-reliable low-latency communications in cellular networks," in *Proc. IEEE ICC Workshops*, May 2017, pp. 1005–1010.

[18] V. N. Swamy *et al.*, "Real-time cooperative communication for automation over wireless," *IEEE Trans. Wireless Commun.*, vol. 16, no. 11, pp. 7168–7183, Nov. 2017.
[19] L. Liu and W. Yu, "A D2D-based protocol for ultra-reliable wireless communications for industrial automation," *IEEE Trans. Wireless Commun.*, vol. 17, no. 8, pp. 5045–5058, Aug. 2018.
[20] C. She, C. Yang, and T. Q. S. Quek, "Cross-layer optimization for ultra-reliable and low-latency radio access networks," *IEEE Trans. Wireless Commun.*, vol. 17, no. 1, pp. 127–141, Jan. 2018.
[21] C. She, C. Yang, and T. Q. S. Quek, "Joint uplink and downlink resource configuration for ultra-reliable and low-latency communications," *IEEE Trans. Commun.*, vol. 66, no. 5, pp. 2266–2280, May 2018.
[22] H. Zhang, N. Liu, X. Chu, K. Long, A.-H. Aghvami, and V. C. M. Leung, "Network slicing based 5g and future mobile networks: Mobility, resource management, and challenges," *IEEE Commun. Mag.*, vol. 55, no. 8, pp. 138–145, Aug. 2017.
[23] P. Popovski, K. F. Trillingsgaard, O. Simeone, and G. Durisi, "5G wireless network slicing for eMBB, URLLC, and mMTC: A communication-theoretic view," *IEEE Access*, vol. 6, pp. 55765–55779, 2018.
[24] R. Kassab, O. Simeone, and P. Popovski. (2018). "Coexistence of URLLC and eMBB services in the C-RAN uplink: An information-theoretic study." [Online]. Available: https://arxiv.org/abs/1804.06593
[25] J. Park and M. Bennis. (2018). "URLLC-eMBB slicing to support VR multimodal perceptions over wireless cellular systems." [Online]. Available: https://arxiv.org/abs/1805.00142
[26] K. I. Pedersen, G. Berardinelli, F. Frederiksen, P. Mogensen, and A. Szufarska, "A flexible 5G frame structure design for frequency-division duplex cases," *IEEE Commun. Mag.*, vol. 54, no. 3, pp. 53–59, Mar. 2016.
[27] G. Pocovi, K. I. Pedersen, B. Soret, M. Lauridsen, and P. Mogensen, "On the impact of multi-user traffic dynamics on low latency communications," in *Proc. IEEE ISWCS*, Poznan, Poland, Sep. 2016, pp. 204–208.
[28] R. Peter *et al.*, "Network slicing to enable scalability and flexibility in 5G mobile networks," *IEEE Commun. Mag.*, vol. 55, no. 5, pp. 72–79, May 2017.
[29] C. Lu and Y.-F. Liu, "An efficient global algorithm for single-group multicast beamforming," *IEEE Trans. Signal Process.*, vol. 65, no. 14, pp. 3761–3774, Jul. 2017.
[30] Y. Shi, J. Zhang, and K. B. Letaief, "Robust group sparse beamforming for multicast green Cloud-RAN with imperfect CSI," *IEEE Trans. Signal Process.*, vol. 63, no. 17, pp. 4647–4659, Sep. 2015.
[31] J. Arnau and M. Kountouris, "Delay performance of MISO wireless communications," in *Proc. 16th Int. Symp. Modeling Optim. Mobile, Ad Hoc, Wireless Netw. (WiOpt)*, Shanghai, China, May 2018, pp. 1–8.
[32] Y. Polyanskiy, H. V. Poor, and S. Verdú, "Channel coding rate in the finite blocklength regime," *IEEE Trans. Inf. Theory*, vol. 56, no. 5, pp. 2307–2359, May 2010.
[33] C. She, C. Yang, and T. Q. S. Quek, "Radio resource management for ultra-reliable and low-latency communications," *IEEE Commun. Mag.*, vol. 55, no. 6, pp. 72–78, Jun. 2017.
[34] A. Anand, G. De Veciana, and S. Shakkottai, "Joint scheduling of URLLC and eMBB traffic in 5G wireless networks," in *Proc. IEEE INFOCOM*, Honolulu, HI, USA, Apr. 2018, pp. 1970–1978.
[35] J. Tang, W. P. Tay, T. Q. S. Quek, and B. Liang, "System cost minimization in cloud RAN with limited fronthaul capacity," *IEEE Trans. Wireless Commun.*, vol. 16, no. 5, pp. 3371–3384, May 2017.
[36] J. Tang, L. Teng, T. Q. S. Quek, T.-H. Chang, and B. Shim, "Exploring the interactions of communication, computing and caching in cloud RAN under two timescale," in *Proc. IEEE SPAWC*, Sapporo, Japan, Jul. 2017, pp. 1–6.
[37] J. Tang, T. Q. S. Quek, T.-H. Chang, and B. Shim, "Systematic resource allocation in cloud RAN with caching as a service under two timescales," unpublished. [Online]. Available: https://www.dropbox.com/s/s447g0zd0404wl6/
[38] H.-T. Wai, Q. Li, and W.-K. Ma, "Discrete sum rate maximization for MISO interference broadcast channels: Convex approximations and efficient algorithms," *IEEE Trans. Signal Process.*, vol. 64, no. 16, pp. 4323–4336, Aug. 2016.
[39] Z.-Q. Luo, W.-K. Ma, A. M.-C. So, Y. Ye, and S. Zhang, "Semidefinite relaxation of quadratic optimization problems," *IEEE Signal Process. Mag.*, vol. 27, no. 3, pp. 20–34, May 2010.
[40] S. Schiessl, J. Gross, and H. Al-Zubaidy, "Delay analysis for wireless fading channels with finite blocklength channel coding," in *Proc. ACM MSWiM*, Cancun, Mexico, Nov. 2015, pp. 13–22.
[41] M. Grant and S. Boyd. (Mar. 2014). *CVX: Matlab Software for Disciplined Convex Programming, Version 2.1*. [Online]. Available: http://cvxr.com/cvx
[42] X. Zheng, X. Sun, D. Li, and J. Sun, "Successive convex approximations to cardinality-constrained convex programs: A piecewise-linear DC approach," *Comput. Optim. Appl.*, vol. 59, no. 1, pp. 379–397, 2014.
[43] M. Razaviyayn, H.-W. Tseng, and Z.-Q. Luo. (2015). "Computational intractability of dictionary learning for sparse representation." [Online]. Available: https://arxiv.org/abs/1511.01776
[44] E. Karipidis, N. Sidiropoulos, and Z.-Q. Luo, "Quality of service and max-min fair transmit beamforming to multiple cochannel multicast groups," *IEEE Trans. Signal Process.*, vol. 56, no. 3, pp. 1268–1279, Mar. 2008.
[45] M. Bennis, M. Debbah, and H. V. Poor. (2018). "Ultra-reliable and low-latency wireless communication: Tail, risk and scale." [Online]. Available: https://arxiv.org/abs/1801.01270
[46] C.-S. Chang and J. A. Thomas, "Effective bandwidth in high-speed digital networks," *IEEE J. Sel. Areas Commun.*, vol. 13, no. 6, pp. 1091–1100, Aug. 1995.
[47] G. Wang, G. Feng, W. Tan, S. Qin, R. Wen, and S. Sun, "Resource allocation for network slices in 5G with network resource pricing," in *Proc. IEEE GLOBECOM*, Singapore, Dec. 2017, pp. 1–6.
[48] *LTE; Evolved Universal Terrestrial Radio Access (E-UTRA); Radio Frequency (RF) Requirements for LTE Pico Node B (Release 9)*, document TS 36.931, v9.0.0. 3GPP, May 2011,

Jianhua Tang (S'11–M'15) received the B.E. degree in communication engineering from Northeastern University, China, in 2010, and the Ph.D. degree in electrical and electronic engineering from Nanyang Technological University, Singapore, in 2015. He was a Post-Doctoral Research Fellow with the Singapore University of Technology and Design from 2015 to 2016. He is currently a Research Assistant Professor with the Department of Electrical and Computer Engineering, Seoul National University. His research interests include cloud computing, cloud radio access network, and network slicing.

Byonghyo Shim (S'95–M'97–SM'09) received the B.S. and M.S. degrees in control and instrumentation engineering from Seoul National University, South Korea, in 1995 and 1997, respectively, and the M.S. degree in mathematics and the Ph.D. degree in electrical and computer engineering from the University of Illinois at Urbana–Champaign, IL, USA, in 2004 and 2005, respectively. From 1997 and 2000, he was an Officer (First Lieutenant) and also a full-time Academic Instructor with the Department of Electronics Engineering, Korean Air Force Academy. From 2005 to 2007, he was a Staff Engineer with Qualcomm, Inc., San Diego, CA, USA. From 2007 to 2014, he was an Associate Professor with the School of Information and Communication, Korea University, Seoul. Since 2014, he has been with Seoul National University, where he is currently a Professor with the Department of Electrical and Computer Engineering. His research interests include wireless communications, statistical signal processing, compressed sensing, and machine learning. He is an elected member of the Signal Processing for Communications and Networking Technical Committee of the IEEE Signal Processing Society. He was a recipient of the M. E. Van Valkenburg Research Award from University of Illinois, in 2005, the Hadong Young Engineer Award from the IEIE in 2010, and the Irwin Jacobs Award from Qualcomm and KICS in 2016. He has served as an Associate Editor of the IEEE TRANSACTIONS ON SIGNAL PROCESSING, the IEEE TRANSACTIONS ON COMMUNICATIONS, the IEEE WIRELESS COMMUNICATIONS LETTERS, and the *Journal of Communications and Networks*, and a Guest Editor of the IEEE JOURNAL ON SELECTED AREAS IN COMMUNICATIONS.

Tony Q. S. Quek (S'98–M'08–SM'12–F'18) received the B.E. and M.E. degrees in electrical and electronics engineering from the Tokyo Institute of Technology, Tokyo, Japan, in 1998 and 2000, respectively, and the Ph.D. degree in electrical engineering and computer science from the Massachusetts Institute of Technology, Cambridge, MA, USA, in 2008. He is currently a tenured Associate Professor with the Singapore University of Technology and Design (SUTD). He also serves as the Acting Head of ISTD Pillar and the Deputy Director of the SUTD-ZJU IDEA. His current research topics include wireless communications and networking, Internet-of-Things, network intelligence, wireless security, and big data processing.

He has co-authored the book *Small Cell Networks: Deployment, PHY Techniques, and Resource Allocation* (Cambridge University Press, 2013) and the book *Cloud Radio Access Networks: Principles, Technologies, and Applications* (Cambridge University Press, 2017). He has been actively involved in organizing and chairing sessions, and has served as a member of the Technical Program Committee and symposium chairs in a number of international conferences. He is currently an elected member of the IEEE Signal Processing Society SPCOM Technical Committee. He was an Executive Editorial Committee Member for the IEEE TRANSACTIONS ON WIRELESS COMMUNICATIONS, an Editor for the IEEE TRANSACTIONS ON COMMUNICATIONS, and an Editor for the IEEE WIRELESS COMMUNICATIONS LETTERS.

Dr. Quek was honored with the 2008 Philip Yeo Prize for Outstanding Achievement in Research, the IEEE GLOBECOM 2010 Best Paper Award, the 2012 IEEE William R. Bennett Prize, the 2015 SUTD Outstanding Education Awards–Excellence in Research, the 2016 IEEE Signal Processing Society Young Author Best Paper Award, the 2017 CTTC Early Achievement Award, the 2017 IEEE ComSoc AP Outstanding Paper Award, and the 2016–2018 Clarivate Analytics Highly Cited Researcher. He is a Distinguished Lecturer of the IEEE Communications Society.