Intelligent Traffic Steering in Beyond 5G Open RAN Based on LSTM Traffic Prediction

Fatemeh Kavehmadavani, Van-Dinh Nguyen, *Member, IEEE*, Thang X. Vu, *Senior Member, IEEE*, and Symeon Chatzinotas, *Fellow, IEEE*

***Abstract*— Open radio access network (ORAN) Alliance offers a disaggregated RAN functionality built using open interface specifications between blocks. To efficiently support various competing services, *namely* enhanced mobile broadband (eMBB) and ultra-reliable and low-latency (uRLLC), the ORAN Alliance has introduced a standard approach toward more virtualized, open, and intelligent networks. To realize the benefits of ORAN in optimizing resource utilization, this paper studies an intelligent traffic steering (TS) scheme within the proposed disaggregated ORAN architecture. For this purpose, we propose a joint intelligent traffic prediction, flow-split distribution, dynamic user association, and radio resource management (JIFDR) framework in the presence of unknown dynamic traffic demands. To adapt to dynamic environments on different time scales, we decompose the formulated optimization problem into two long-term and short-term subproblems, where the optimality of the latter is strongly dependent on the optimal dynamic traffic demand. We then apply a long-short-term memory (LSTM) model to effectively solve the long-term subproblem, aiming to predict dynamic traffic demands, RAN slicing, and flow-split decisions. The resulting non-convex short-term subproblem is converted to a more computationally tractable form by exploiting successive convex approximations. Finally, simulation results are provided to demonstrate the effectiveness of the proposed algorithms compared to several well-known benchmark schemes.**

***Index Terms*— Beyond 5G networks, open radio access networks, intelligent resource management, traffic prediction, traffic steering, long short-term memory, network slicing.**

I. Introduction

NEXT-GENERATION ("NextG") mobile communication networks (*e.g.*, beyond fifth-generation (5G) and sixth-generation (6G)) are designed to accommodate a wide range of service types with their own specific demands, such as throughput, reliability, and delay. The mentioned services are basically categorized into three principal cases, enhanced mobile broadband (eMBB), massive machine-type communications (mMTC), and ultra-reliability low-latency communication (uRLLC) [1]. Efficiently supporting the coexistence of these heterogeneous services is challenging in the "NextG" wireless networks due to their competing demands. The existing "one-size-fits-all" 5G architecture makes it very difficult if not impossible to enable the coexistence of heterogeneous services since the present 5G wireless networks are aggregated, closed, and inflexible. Despite the cost-effectiveness of centralized/cloud radio access networks (CRAN) and virtual radio access networks (vRAN), open interfaces, non-proprietary hardware, and software are still lacking in these systems. Open RAN (ORAN) is an emerging solution to enable flexible, virtualized, disaggregated, intelligent, and open "NextG" wireless networks to support the heterogeneity of wireless services [2]. The openness of RAN components not only increases the interoperability between vendors but also speeds up the delivery of new services, which can be dynamically nominated to users. Due to the increasing complexity of "NextG" wireless networks, a self-organizing network's optimization, deployment, and operation are increasingly becoming impossible without intelligence [3], [4].

Accommodating heterogeneous services (uRLLc, eMBB, and mMTC) with competing demands on the identical RAN infrastructure is exceedingly challenging, such that building numerous physical networks to accommodate distinct services is not practical. Hence, it is difficult to efficiently route heterogeneous traffics to enhance user experience and network efficiency [5]. To this end, the concept of RAN slicing has been suggested as a potential remedy to constantly assign the accessible storage, compute, and communication resources across multiple services whilst guaranteeing their isolation [6]. In this study, we concentrate on the RAN slicing mechanism's optimization, which entails the effective allocation of the physical radio resources such as transmit power and the time-frequency unit. Meeting the multi-traffic coexistence to handle nonuniform requirements is not possible only by allocating the transmit power and time-frequency unit. Traffic steering (TS), one of the most efficient approaches, enables network software to steer the traffics in the most proper paths by routing user traffics through the most suitable radio resources. Nevertheless, the available research on TS in 5G is still limited and uncompleted. While most existing works of literature have studied typical TS which treats all users similarly, regardless of users' demands and network conditions, meaning that a network operator may even be wasted its resources if a

Manuscript received 19 August 2022; revised 16 December 2022 and 13 February 2023; accepted 1 March 2023. Date of publication 15 March 2023; date of current version 13 November 2023. This work was supported in part by the European Research Council (ERC) AGNOSTIC Project under Grant H2020/ERC2020POC/957570/DREAM and in part by the Luxembourg National Research Fund through the Project RUTINE under Grant C22/IS/17220888 and the Project ASWELL under Grant C19/IS/13718904. The associate editor coordinating the review of this article and approving it for publication was X. Gong. *(Corresponding author: Fatemeh Kavehmadavani.)*

Fatemeh Kavehmadavani, Thang X. Vu, and Symeon Chatzinotas are with the Interdisciplinary Centre for Security, Reliability and Trust (SnT), University of Luxembourg, 4365 Esch-sur-Alzette, Luxembourg (e-mail: fatemeh.kavehmadavani@uni.lu; thang.vu@uni.lu; symeon.chatzinotas@uni.lu).

Van-Dinh Nguyen is with the College of Engineering and Computer Science, VinUniversity, Gia Lam, Hanoi 100000, Vietnam (e-mail: dinh.nv2@vinuni.edu.vn).

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TWC.2023.3254903.

Digital Object Identifier 10.1109/TWC.2023.3254903

simple strategy is implemented. To address this issue, this paper proposes a novel TS based on the traffic demands to achieve multi-traffic coexistence. To enhance throughput and reliability in wireless networks with limited bandwidth, the multi-connectivity (MC) technique can be used to aggregate multiple links and allow a user to connect to more than two nodes. In practice, MC has the potential to dramatically reduce interference and latency of mobility methods, especially at the cell edge [7]. The multi-link capability makes MC the most practical method for achieving uRLLC and eMBB coexistence, whereas the recent proposals for the 5G air interface in 3GPP Release 15 utilize flexible mixed numerologies [8].

Another great challenge of 5G is achieving low latency in latency-critical applications. To meet this, 5G NR defines a new concept of mini-slot which composes of at most 4 OFDM symbols to support small packet transmission size. Significantly this short slot duration reduces the transmission time. Furthermore, the single numerology which is used in 4G LTE is not suitable for expected multi services in 5G wireless networks. Hence, flexible mixed numerologies have been recently proposed for such wireless networks in 3GPP Release 15 which enhances flexibility. To this end, this paper considers mixed numerologies in a frequency domain such that the assigned services to each slice can select a proper numerology to allocate its data transmission while guaranteeing each service's requirements. It should be mentioned that this new concept introduces new challenges related to RAN slicing that need to be studied. For instance, the dynamic allocation of the mixed numerology-based time-frequency units and transmit power is a vital challenge.

Inspired by [9] and [10], this paper introduces a joint intelligent traffic steering and slice-isolation radio resource allocation framework for allocating the RAN resources with mixed numerologies, taking into account the ORAN architectural requirements, various service requirements, and queue status. To present the role of intelligence in ORAN architecture, this paper benefits from the long short-term memory (LSTM) recurrent neural network (RNN) to learn the network traffic pattern and predict the unknown incoming traffic packets of the network. LSTM has been introduced as an undeniable state-of-the-art method within the deep neural networks to overcome the exploding/vanishing gradient problem, especially in learning long-term dependencies [11]. We outline the compliance of the overall scheme with the ORAN requirements later.

A. Related Works

To improve services for network providers, the work in [12] focused on providing an efficient scheduling scheme to dynamically allocate radio resources in LTE networks. In [13], the authors proposed a joint resource allocation and dynamic link adaptation scheme for multiplexing eMBB and uRLLC on a shared channel, which dynamically tunes the block error probability of URLLC small payload transmissions in each cell. A control channel and packet size aware resource allocation approach was introduced in [14] to enable the packet scheduling and resource allocation for uRLLC and eMBB traffics coexistence in 5G NR networks. Although the heuristic algorithm proposed in [14] meets the uRLLC's requirements by preserving a large number of resources to uRLLC, this method has failed to isolate the slice, resulting in a reduction of the eMBB throughput compared to high uRLLC traffic. Wu et al. [15] developed the puncturing method to eliminate the uRLLC queuing delay for multiplexing of uRLLC and eMBB services. The authors in [16] studied a joint scheduling scheme to maximize the eMBB throughput while minimizing the utility of uRLLC to meet the quality of service (QoS) requirements. Since uRLLC services are prioritized in the puncturing-based schemes and scheduled on the assigned eMBB's resources, the eMBB performance (throughput and reliability) significantly decreases when the uRLLC traffic increases. Moreover, the fixed numerology over frequency-time resources for the scheduling scheme is often considered.

There is significant attention from academia and industry to TS in the literature. In [17], a TS framework was studied in unlicensed bands on the LTE network in order to distribute traffic among radio access technologies, heterogeneous cells, and spectrum bands. To overcome the puncturing difficulties in multiple services, Korrai et al. in [18] proposed a slice-isolated RAN slicing scheme with orthogonal frequency-division multiple access (OFDMA) for the coexistence of uRLLC and eMBB. A joint scheduling and TS scheme based on dynamic MC and RAN slicing in 5G networks were analyzed in [19], in which an effective capacity model to evaluate the frameworks' performance is proposed. To integrate the LTE into 5G networks, Prasad et al. [20] investigated an energy-efficient RAN moderation and dynamic TS based on the connectivity by multiple radio links.

The RAN slicing framework over multiple services networks has been recently developed under frequency-time resources thanks to the flexibility of mixed-numerologies. The authors in [21] studied a resource allocation optimization problem by considering the flexible numerology in both frequency and time domains. The work in [22] analyzed the wireless scheduling optimization problem over the mixed-numerologies to support the heterogeneous services with different QoS requirements, assuming that mapping the radio resources (time and frequency) is decoupled from service scheduling. A joint optimization of RAN slicing, resource block, and power allocation problem for eMBB, mMTC, and uRLLC in 5G wireless networks was considered in [23] under imperfect channel state information (CSI).

However, the aforementioned works have investigated TS with flexible numerology in the "one-size-fits-all" network architecture, which is not adaptable enough to support heterogeneous services. Despite the huge benefit of intelligence of ORAN, there are only a few attempts on the TS in the literature. Niknam et al. in [10] proposed an intelligent traffic prediction and radio resource management framework to control the congested cell based on cell-splitting in ORAN architecture for multiplexing uRLLC and eMBB services. In [24], a systematic analysis for implementing the intelligence in each layer of ORAN architecture for data-driven "NexG" wireless networks was provided by considering the closed-control loops

between ORAN components. Furthermore, in our previous study [9], we have proposed a TS scheme based on MC and RAN slicing technologies to effectively allocate diverse network resources in ORAN architecture by assuming fixed-numerology (*i.e.*, 0.25ms mini-slots) tailored with 5G NR. However, the works mentioned earlier have not designed a traffic steering and RAN resource slicing scheme for heterogeneous traffic applicable to the beyond 5G wireless networks on ORAN architecture. The most of works have investigated the resource allocation scheme for various services with fixed numerology in the monolithic and inflexible architecture of 4G LTE networks. Hence, this is the first attempt to investigate the performance efficiency of mixed numerologies considering the RAN slicing, MC, and mini-slot to achieve multi-traffic coexistence in ORAN architecture, while guaranteeing the users' QoS requirements.

B. Contributions

In this paper, we develop an intelligent TS framework in the presence of unknown dynamic traffic demand to meet the requirements of both uRLLC and eMBB services in beyond 5G based on dynamic MC. Learning an optimal traffic steering policy in dynamic environments is challenging because fluctuations in traffic demand over time are non-stationary and unknown, hindering the computation of cost-efficient associations. This study proposes an intelligent framework by locating rAPPs and xAPP at the non-real-time RAN intelligent controller (non-RT RIC) and near-real-time RIC (near-RT RIC) of the ORAN architecture. The existing rAPPs at non-RT RIC include the traffic prediction, dynamic RAN slicing decision, and flow-split distribution, while the xAPP at near-RT RIC is radio resource management to schedule the joint resource block and transmission power with mixed numerologies based on standardization in 5G NR. To the best of our knowledge, this is the first work to model intelligent TS in ORAN architecture and study TS in-depth detail in ORAN layers considering the mixed-numerology in the presence of unknown traffic demands.

To achieve the maximum throughput for eMBB traffic while guaranteeing the minimum uRLLC latency requirement and vice versa, we propose a joint intelligent traffic prediction, flow-split distribution, dynamic user association, and radio resource management scheme befitting the ORAN architecture. Then, we identify the location of the ML training, AI server, and inference modules to provide a high-level architecture of deployment scenarios and end-to-end flow to prove compatibility with ORAN standards. Our main contributions are summarized as follows:

- We develop a general optimization framework to jointly optimize the intelligent traffic prediction, flow-split distribution, dynamic user association, and radio resource management, called "JIFDR". To maximize the eMBB's throughput while guaranteeing the uRLLC latency requirement, or vice versa, we formulate two optimization problems with different objective designs while satisfying QoS requirements, slice isolation, power budget, and maximum fronthaul (FH) capacity.
- To effectively solve the formulated problems, we divide each problem into long-term and short-term subproblems, which are executed on different time scales. The long-term subproblem is mapped into three dependent rAPPs: traffic prediction, dynamic RAN slicing decision, and flow-split distribution at the non-RT RIC. In contrast, the short-term sub-problem is deployed as the radio resource management xAPP at the near-RT RIC, which is linked to the upper layer through the A1 interface.
- The long-term subproblem benefits from the LSTM RNN to learn and predict traffic patterns and demands. This model is trained offline at the non-RT RIC in the service management and orchestration (SMO) through the long-term collected data from the RAN layer via the O1 interface. RNN is utilized to learn the temporal pattern of the traffic demand from current values in order to forecast future values. Upon the inference result, two heuristic methods are proposed to optimize the RAN slicing and flow-split distribution.
- Next, given rAPPs' outcomes sent from the non-RT RIC via the A1 interface, we propose a successive convex approximation (SCA)-based iterative algorithm to solve the short-term subproblem, which belongs to a class of mixed-integer non-convex programming (MINCP) problem.
- Finally, numerical results are presented to demonstrate the proposed algorithm's quick convergence behavior and to confirm its efficacy in comparison to benchmark schemes. Furthermore, by using a mathematical analysis convergence and complexity analysis are studied. The average mean square error MSE) of the prediction is relatively low at 0.0033.

The rest of this paper is organized as follows. Section II introduces the ORAN architecture and system model. In Section III, we present the problem formulation and overall intelligent TS deployment architecture and algorithm. Section IV first proposes the LSTM model and heuristic methods to solve the long-term subproblem and then develops an SCA-based iterative algorithm to solve the short-term subproblem. Simulation results and discussions are provided in Section V, while Section VI concludes the paper.

II. ORAN Architecture and System Model

A. ORAN Architecture

The ORAN architecture based on the ORAN Alliance is illustrated in Fig. 1, including three main layers (the management, control, and function layers). To further reduce the RAN expenditure, ORAN fosters self-organizing networks by adding two unique modules of near-RT and non-RT RICs to enable a centralized network abstraction which improves efficiency by cost-reducing the human-machine interaction. Following the disaggregation concept, BS functionalities are virtualized as network functions based on the 3GPP functional split and are distributed among various network nodes, *namely* central unit (CU), distributed unit (DU), and radio unit (RU) [10]. Hence, open interfaces (FH, A1, O1, E2, F1) are introduced to enable efficient multi-vendor interoperability, where

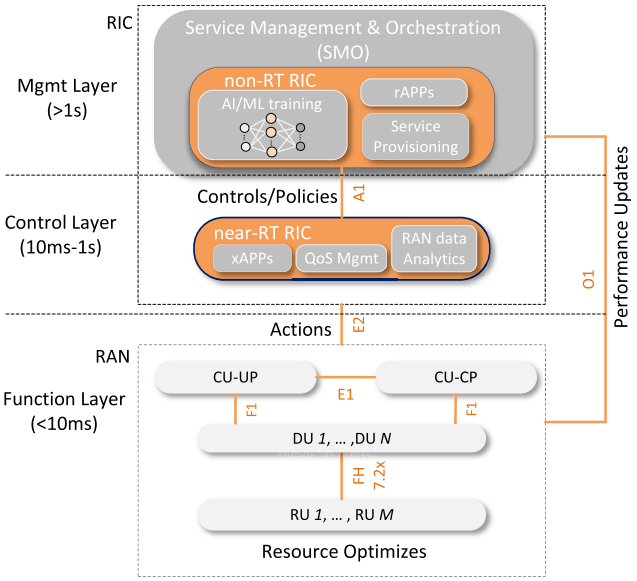

Fig. 1. ORAN architecture based on ORAN Alliance [25].

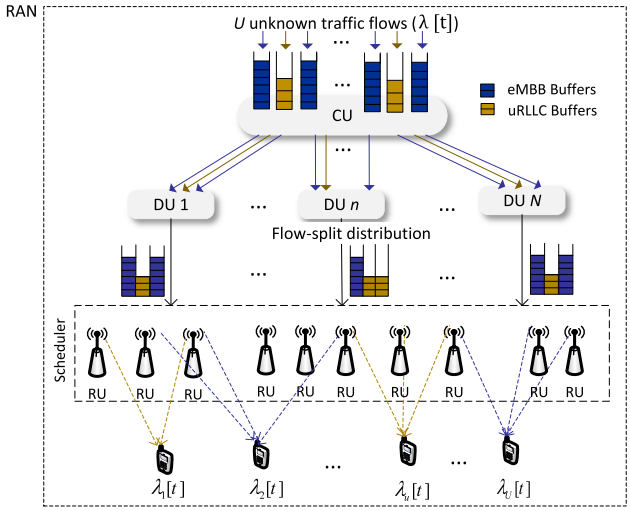

Fig. 2. System model with the traffic-steering scheme.

a network operator can select RAN components from different vendors individually.

The unique feature of RICs is to create closed-control loops (*i.e.*, autonomous action and feedback loops) between RAN components and their controllers. In order to control traffic prediction, network slicing, and hand-over management, ORAN defines three control loops, *namely* non-RT, near-RT, and RT running at different timescales ranging from 1 ms to thousands of ms, enabling real-time control of transmission methods and beamforming. Following ORAN Alliance specifications, each loop that works on a timescale of at least one second is called a non-RT control loop, which involves the coordination between both RICs through the A1 interface. The near-RT control loop operates on a timescale between 10 ms and 1 s while it runs between the near-RT RIC and DU and CU components. The third loop working on sub-10 ms is labeled as the RT control loop, which is largely relevant to interactions between elements of DU and cell site.

In particular, the non-RT RIC carries out tasks with a temporal granularity greater than one second, like service provisioning and training AI/ML models, which rAPPs can be implemented in this controller. On the other hand, the near-RT RIC manages operations with timescales of more than 10 ms, hosts external applications (referred to as xApps), and incorporates intelligence in the RAN by data-driven control loops. RICs may execute applications created by independent third-party specialized software suppliers as a platform for hosting software. These applications are known as "rAPPs" and "xAPPs", and act as key enablers to run on non-RT and near-RT RICs, respectively. rAPPs handle the non-RT functions that require more than 1 second to be executed which may take minutes or hours. While xAPPs are external applications specific to handle radio functions that run between 10 ms and 1 s that interact with RAN elements and the upper layer by open interfaces to reconfigure some exposed functionality. To this end, ORAN Alliance strives to steer the industry toward the development of AI/ML-enabled RICs.

B. Network Model

We consider a downlink OFDMA multi-user multiple-input single-output (MU-MISO) system in the ORAN architecture, consisting of one CU, the set $\mathcal{N} \triangleq \{1, 2, \ldots, N\}$ of N DUs and the set $\mathcal{M} \triangleq \{1, 2, \ldots, M\}$ of M RUs. For cost-effective deployment, each DU serves a cluster of RUs. Let denote by $\mathcal{M}_n \triangleq \{(n, 1), \ldots, (n, M_n)\}$ with $|\mathcal{M}_n| = M_n$ and $\sum_{n\in\mathcal{N}} M_n = M$ the set of RUs served by DU n. The m-th RU served by n-th DU is referred to as RU(n, m), which is equipped with K antennas while users are equipped with a single antenna. Let us denote by $\mathcal{U} \triangleq \{1, \ldots, U\}$ the set of users served by DUs, which can be further divided into two disjoint sets $\mathcal{U}^{\text{ur}}$ of U^{ur} uRLLC users and $\mathcal{U}^{\text{em}}$ of U^{em} eMBB users. The eMBB users generate the traffic with a large packet of size Z^{em} bytes, while uRLLC users generate a sequence of small and identical packets of Z^{ur} bytes. In addition, as shown in Fig. 2, we assume that all data arriving from upper layers are stored in the user-specific transmission buffers of the RUs till it is time to serve it. The RUs serve the users in the cell by allocating the frequency-time radio resource blocks (RBs) and transmission power to each RB. The parameters used in this study are summarized in Table I.

To meet the demands of exigent latency services, we investigate a mini-slot-based framework, where each time slot is broken into two mini-slots. Each mini-slot has a duration of $\delta = 1/2^{\gamma+1}$ ms and comprises 7 OFDM symbols, where $\gamma \in \{0, 1, 2\}$ is the subcarrier spacing (SCS) index. Hereon, we suppose that several RUs operating in MC configuration are simultaneously providing eMBB and uRLLC services. Following [26], numerology with index $i = 1$ (*i.e.* SCS index $\gamma = 1$) is appropriate for eMBB to meet the requirement of high data rate, while numerology with index $i = 2$ (*i.e.* SCS index $\gamma = 2$) is more suitable for the uRLLC service's applications with the latency-critical and small data packet of uRLLC. From the mixed-numerologies point of view, eMBB service sorts the numerology $i = 1$ with RB's bandwidth (BW) of $\beta_i|_{i=1} = 360$ kHz and $\delta_i|_{i=1} = 0.25$ ms of transmission time interval (TTI) duration as the highest priority and uRLLC

TABLE I
SUMMARY OF MAIN NOTATIONS AND VARIABLES

Variable	**Meaning**
$\alpha[t]$	Bandwidth (BW)-split variable per frame t
$\varphi_{m,u}[t]$	Portion of data flow routed to user u via RU m per frame t
$\pi^{\mathsf{ur/em}}_{m,u,f_i}[t_s]$	RB (t_s, f_i) allocated to user u (uRLLC or eMBB) via RU m
$p^{\mathsf{ur/em}}_{m,u,f_i}[t_s]$	Transmit power from RU m to user u (uRLLC or eMBB) via RB (t_s, f_i)
$\lambda_u[t]$	Traffic demand of user u per frame t
Notation	**Meaning**
$\mathcal{N}$, $\mathscr{M}$, $\mathscr{M}_n$	Sets of DUs, RUs, and set of RUs covered by DU n
$\mathcal{U}$, $\mathcal{U}^{\mathsf{ur}}$ $\mathcal{U}^{\mathsf{em}}$	Sets of all users, uRLLC users, and eMBB users, respectively
B, β_i, B_i	Total system BW, RB's BW, and BWP assigned to numerology i
Δ, δ_i	Frame's duration and TTI's duration
S_i, F_i	Numbers of TTI per frame and subcarriers per TTI
$a_u[t]$	Flow-split selection vector for user u
$\boldsymbol{h}_{m,u,f_i}[t_s]$, $g_{m,u,f_i}[t_s]$	Channel vector and effective channel gain between RU m and user u
$\bar{\boldsymbol{h}}_{m,u,f_i}[t]$, $\tilde{\boldsymbol{h}}_{m,u,f_i}[t_s]$	Line-of-sight (LoS) and non-LoS (NLoS) components
$\varrho_{m,u,f_i}[t]$, $\zeta_{m,u,f_i}[t]$	Rician factor and large-scale fading
N_0, N_1	Power of the AWGN
V, P_e, Q^{-1}	Channel dispersion, error probability, and inverse of Q-function
Γ, Γ_0	The received SNR, and minimum received SNR
λ^{max}, Λ	Maximum finite arrival traffic and the total of all traffic demands
Q^{max}, $q_{m,u}$	Maximum queue buffer capacity and queue-length of user u at buffer of RU m
$\mu_{cu/du}$, $f_{cu/du}$, C	Task rate, computation capacities of CU and DU and number of cycles
τ^{ur}_u, $\tau^{\mathsf{pro}}_{cu/du}$	The e2e latency of uRLLC user u and processing latency of all users at CU/DU
$\tau^{\mathsf{tx}}_{cu,du}$, $\tau^{\mathsf{tx}}_{du,ru}$, $\tau^{\mathsf{tx}}_{ru,u}$	Transmission latency under MH and FH links and from RU m to user u
D_{ur}, C^{MH}, C^{FH}	Latency requirement of uRLLC traffic and maximum MH and FH capacity

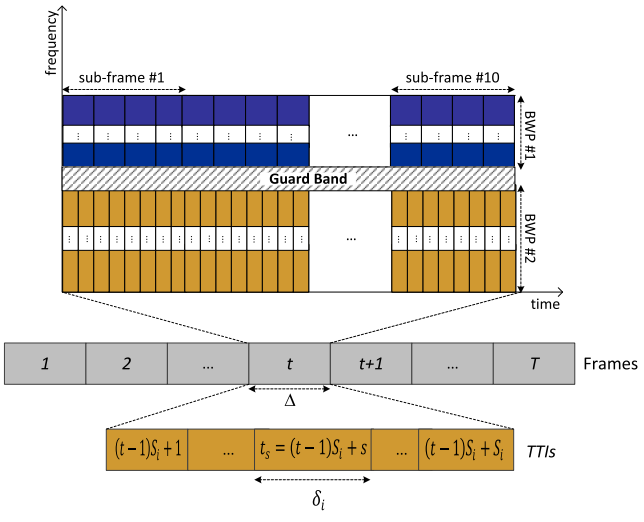

Fig. 3. Time-frequency grid with different numerologies.

service would prioritize numerology $i = 2$ with RB's BW of $\beta_i|_{i=2} = 720$ kHz and $\delta_i|_{i=2} = 0.125$ ms of TTI duration.

The multiplexing of mixed numerologies in the frequency domain is considered in this work, where the carrier BW that is accessible for the downlink transmissions is divided into several bandwidth parts (BWPs). According to this, each user is able to alter its RF bandwidth based on its required data rate by switching between numerous BWPs. As illustrated in Fig. 3, the desirable BWP design to serve two types of services with different requirements is established based on the expected queue length of each service by introducing the BW-split variable $\alpha[t] \in [0, 1]$. Whereas this method does not call for tight time synchronization techniques, using various numerologies in the adjacent sub-bands causes inter-numerology interference (INI). Hence, to reduce INI, a fixed guard band B_G equal to one RB's BW (*i.e.*, 180 kHz) is configured between the two neighbor numerologies (*i.e.*, sub-bands). The scheduled BWP assigned to the uRLLC slice with numerology $i = 2$ is denoted by $B_i[t]|_{i=2} = \alpha[t]B$, to unload the existing packets in the uRLLC slice's queues at frame t, where B is the total carrier BW. In contrast, $B_i[t]|_{i=1} = (1 - \alpha[t])B - B_G$ the scheduled BWP assigned to eMBB slice with numerology $i = 1$.

Assume the proposed system model works in a discrete time-frame indexed by $t \in [1, 2, \ldots, T]$, which corresponds to one large-scale coherence time of $\Delta = 10$ ms duration for each frame, as shown in Fig. 3. Depending on the selected numerology i by each service, each frame in the time domain is divided into S_i TTIs where the duration of each TTI denoted by $t_s = (t-1)S_i + s$ with $s = \{1, \ldots, S_i\}$ is δ_i. Thus, based on the selected numerology i, each BWP is partitioned into F_i number of sub-bands of frequency set $\mathscr{F}_i = \{1, \ldots, f_i, \ldots, F_i\}$ in the frequency-domain and S_i number of TTIs in each frame, indexed by $t_s = \{(t-1)S_i + 1, \ldots, (t-1)S_i + s, \ldots, (t-1)S_i + S_i\}$ in the time-domain. Such that, $F_i[t] = \lfloor B_i[t]/\beta_i \rfloor$ and $S_i = \Delta/\delta_i$. Therefore, a total $F_i[t] \times S_i$ number of RBs are accessible

for the services using the i-th numerology at each frame t via each RU.

As depicted in Fig. 2, the U independent data traffics with different demands at the CU layer are subsequently routed to VNFs in the DUs layer for parallel processing, referred to as data flows. We adopt the $M/M/1$ processing queue model on a first-come-first-serve basis to serve each user's packets. As it is clear from Fig. 2, the maximum number of paths for each user is M. According to the principle of the TS technique, the CU splits the data flow of the u-th user into several sub-flows, which are possibly transmitted via the maximum of M paths and then aggregated at this user. Because of the non-overlapped DUs' coverage, the resource optimization design at one DU is similar to that of other DUs. Thus, for ease of presentation, we drop the subscript index of DUs hereafter. To this end, we define $\boldsymbol{a}_u[t] \triangleq \big[a_{m,u}[t]\big]$ as the flow-split selection vector for the u-th data flow in time-frame t. In particular, if $a_{m,u}[t]=1$, the m-th RU is selected to transmit data of u-th data flow; otherwise, $a_{m,u}[t]=0$. In addition, let us denote by $\boldsymbol{\varphi}[t] \triangleq \{\boldsymbol{\varphi}_u[t],\ \forall u | \sum_m \varphi_{m,u}[t]=1, \varphi_{m,u}[t]\in[0,1]\}$ the global flow-split decision, in which $\boldsymbol{\varphi}_u[t] \triangleq \Big[\varphi_{m,u}[t]\Big]^T$ represents the flow-split portion vector of user u while $\sum_m \varphi_{m,u}[t]=1$, where $\varphi_{m,u}[t]\in[0,1]$ indicates a portion of data flow routed to user u via RU m in time t by selecting action $a_{m,u}[t]$.

1) Achievable Throughput: The channel vector between RU m and the u-th user at the sub-band f_i in TTI t_s is denoted by $\boldsymbol{h}_{m,u,f_i}[t_s] \in \mathbb{C}^{K\times 1}$, which follows the Rician fading model with the Rician factor $\varrho_{m,u,f_i}[t]$. Within each frame, we assume that the channel remains temporally invariant, while it may be different across each short-time scale TTI. We model $\boldsymbol{h}_{m,u,f_i}[t_s]$ as:

$$\boldsymbol{h}_{m,u,f_i}[t_s] = \sqrt{\zeta_{m,u,f_i}[t]}\Big(\sqrt{\varrho_{m,u,f_i}[t]/(\varrho_{m,u,f_i}[t]+1)} \times \bar{\boldsymbol{h}}_{m,u,f_i}[t] + \sqrt{1/(\varrho_{m,u,f_i}[t]+1)}\tilde{\boldsymbol{h}}_{m,u,f_i}[t_s]\Big) \tag{1}$$

where $\zeta_{m,u,f_i}[t]$ is the large-scale fading; $\bar{\boldsymbol{h}}_{m,u,f_i}[t]$ and $\tilde{\boldsymbol{h}}_{m,u,f_i}[t_s]$ are the line-of-sight (LoS) and non-LoS (NLoS) components, which follow a deterministic channel and Rayleigh fading model, respectively. Given the orthogonality constraint, this work considers that each RB of a RU is assigned to only one single user during one TTI, such as $\pi^{\mathsf{em}}_{m,u,f_i}[t_s] \in \{0,1\}$ and $\pi^{\mathsf{ur}}_{m,u,f_i}[t_s] \in \{0,1\}$ for eMBB and uRLLC traffics, respectively. Here, $\pi^{\mathsf{em}}_{m,u,f_i}[t_s]=1$ if the RB(t_s,f_i) associated with sub-band f_i in TTI t_s of RU m assigned to the u-th eMBB user, and $\pi^{\mathsf{em}}_{m,u,f_i}[t_s]=0$, otherwise; a similar definition is given for uRLLC users. Let define$\Pi[t_s] \triangleq \{\pi^{\mathsf{em}}_{m,u,f_i}[t_s], \pi^{\mathsf{ur}}_{m,u,f_i}[t_s] \in \{0,1\} | \sum_{m,u}\big(\pi^{\mathsf{em}}_{m,u,f_i}[t_s]+\pi^{\mathsf{ur}}_{m,u,f_i}[t_s]\big)\le 1\}$ as the RB allocation constraint. This is to ensure the orthogonality constraint and QoS constraint for uRLLC service.

Thanks to the MC technique, the main interference of eMBB is eliminated and the rest of the interference can be supposed as noise, which is also constant [27]. Hence, the instantaneous achievable rate in [bits/s] for a given set of channel realizations at the u-th eMBB user at TTI t_s is given by:

$$r^{\mathsf{em}}_{m,u}(\boldsymbol{p}^{\mathsf{em}}[t_s]) = \sum_{f_i=1}^{F_i} \beta_i \log_2\Big(1+\frac{p^{\mathsf{em}}_{m,u,f_i}[t_s]g_{m,u,f_i}[t_s]}{N_0}\Big) \tag{2}$$

where β_i, N_0 and $p^{\mathsf{em}}_{m,u,f_i}[t_s]$ are the bandwidth of each RB in numerology index i, power of the Additive White Gaussian Noise (AWGN), and transmit power from RU m to user u for eMBB traffic at sub-band f_i at the TTI t_s, respectively; $g_{m,u,f_i}[t_s]$ denotes the effective channel gain, given as $g_{m,u,f_i}[t_s] \triangleq \|\boldsymbol{h}_{m,u,f_i}[t_s]\|_2^2$. Let us define $\boldsymbol{p}^{\mathsf{em}}[t_s] \triangleq [p^{\mathsf{em}}_{m,u,f_i}[t_s]],\ \forall f_i,u,m$. The transmit power must satisfy $p^{\mathsf{em}}_{m,u,f_i}[t_s] \le \pi^{\mathsf{em}}_{m,u,f_i}[t_s]P^{\max}_m$ with $P^{\max}_m$ being the power budget at RU m, which guarantees that RU m allocates power to user u on RB(t_s,f_i) only if $\pi^{\mathsf{em}}_{m,u,f_i}[t_s]=1$; otherwise $\pi^{\mathsf{em}}_{m,u,f_i}[t_s]=0$ and $p^{\mathsf{em}}_{m,u,f_i}[t_s]=0$. As a result, the throughput of eMBB user $u\in\mathcal{U}^{\mathsf{em}}$ in TTI t_s is given as $r^{\mathsf{em}}_u(\boldsymbol{p}^{\mathsf{em}}[t_s]) = \sum_m r^{\mathsf{em}}_{m,u}(\boldsymbol{p}^{\mathsf{em}}[t_s])$. The minimum QoS requirement for eMBB users is guaranteed by the constraint $\sum_{t_s} r^{\mathsf{em}}_u(\boldsymbol{p}^{\mathsf{em}}[t_s]) \ge R_{\mathsf{th}}$, where R_{th} is a given QoS threshold.

In contrast, owing to the finite block-length in uRLLC traffics, the instantaneous achievable rate of u-th uRLLC user from RU m in TTI t_s using the short block-length can be expressed as [28]:

$$r^{\mathsf{ur}}_{m,u}(\boldsymbol{p}^{\mathsf{ur}}[t_s],\boldsymbol{\pi}^{\mathsf{ur}}[t_s]) = \sum_{f_i=1}^{F_i}\beta_i\Big[\log_2\Big(1+p^{\mathsf{ur}}_{m,u,f_i}[t_s]\frac{g_{m,u,f_i}[t_s]}{N_0}\Big) - \frac{\pi^{\mathsf{ur}}_{m,u,f_i}[t_s]\sqrt{V}Q^{-1}(P_e)}{\sqrt{\delta_i\beta_i}}\Big],\ \forall u\in\mathcal{U}^{\mathsf{ur}} \tag{3}$$

where V, P_e and Q^{-1}: $\{0,1\} \to \mathbb{R}$ denote the channel dispersion, error probability, and inverse of the Gaussian Q-function, respectively. Let us define $\boldsymbol{p}^{\mathsf{ur}}[t_s] \triangleq [p^{\mathsf{ur}}_{m,u,f_i}[t_s]]$ and $\boldsymbol{\pi}^{\mathsf{ur}}[t_s] \triangleq [\pi^{\mathsf{ur}}_{m,u,f_i}[t_s]],\ \forall f_i,u,m$. It is observed that $V = 1-\frac{1}{(\Gamma[t_s])^2} \approx 1$ when the received $\Gamma[t_s] = \frac{p^{\mathsf{ur}}_{m,u,f_i}[t_s]g_{m,u,f_i}[t_s]}{N_0} \ge \Gamma_0$ with $\Gamma_0 \ge 5$ dB. This can be easily achieved in cellular networks, by arranging the uRLLC decoding vector into one possible null space of the reference subspace, the scheduler can eliminate inter-user interference of uRLLC [29]. Hence, we consider the constraint $\frac{N_0\Gamma_0}{g_{m,u,f_i}[t_s]}\pi^{\mathsf{ur}}_{m,u,f_i}[t_s] \le p^{\mathsf{ur}}_{m,u,f_i}[t_s] \le \pi^{\mathsf{ur}}_{m,u,f_i}[t_s]P^{\max}_m$ to guarantee the approximation $V\approx 1$ as well as the big-M formulation theory to avoid non-convexity of (2). Similar to the eMBB service, the throughput of uRLLC user $u\in\mathcal{U}^{\mathsf{ur}}$ in TTI t_s is given as $r^{\mathsf{ur}}_u(\boldsymbol{p}^{\mathsf{ur}}[t_s],\boldsymbol{\pi}^{\mathsf{ur}}[t_s]) = \sum_m r^{\mathsf{ur}}_{m,u}(\boldsymbol{p}^{\mathsf{ur}}[t_s],\boldsymbol{\pi}^{\mathsf{ur}}[t_s])$. We have the following power constraint:

$$\mathcal{P}[t_s] = \Big\{0 \le p^{\mathsf{em}}_{m,u,f_i}[t_s] \le \pi^{\mathsf{em}}_{m,u,f_i}[t_s]P^{\max}_m, \times \frac{N_0\Gamma_0\pi^{\mathsf{ur}}_{m,u,f_i}[t_s]}{g_{m,u,f_i}[t_s]} \le p^{\mathsf{ur}}_{m,u,f_i}[t_s] \le \pi^{\mathsf{ur}}_{m,u,f_i}[t_s]P^{\max}_m | \times \sum_i\sum_{f_i,u}(p^{\mathsf{em}}_{m,u,f_i}[t_s]+p^{\mathsf{ur}}_{m,u,f_i}[t_s]) \le P^{\max}_m\Big\}. \tag{4}$$

We denote $\lambda_u[t]$ in [packets/s] as the unknown traffic demand of user u in time-frame t with the length of Z^{x} bytes

with $\mathsf{x} \in \{\mathsf{ur}, \mathsf{em}\}$, which is i.i.d. over time and upper bounded by a finite constant $\lambda^{\max}$, such as $\lambda_u[t] \leq \lambda^{\max} \leq \infty$. We consider that the retained independent queue at each RU for the u-th user, which is denoted by $\{\varphi_{m,u}[t]\lambda_u[t]Z^{\mathsf{x}}\}$ as the arrival processes of sub-flows, is controlled by a congestion scheduler. Thus, the queue-length of data flow u at RU m in TTI (t_{s+1}) is $q_{m,u}[t_{s+1}] = \max\{\big[q_{m,u}[t_s] + \varphi_{m,u}[t]\lambda_u[t]Z^{\mathsf{x}}\Delta - r^{\mathsf{x}}_{m,u}[t_{s+1}]\delta_i\big], 0\}$. In order to avoid the packet loss due to buffer overflow in each RU, the constraint $\sum_u q_{m,u}[t_s] \leq Q^{\max}, \forall m$ is imposed to ensure that the available packets in the buffer of RU shouldn't exceed the maximum queue-length of $Q^{\max}$ for each RU. Let $\boldsymbol{q}[t_s] \triangleq \big[q_{m,u}[t_s]\big]^T,\ \forall m, u$.

2) The E2E Traffic Latency for uRLLC: Denote by f_{cu} and f_{du} the computation capacities of CU and DU [cycles/sec], respectively. Considering the identical packet size, the required computation resource to process one packet of size Z is C (number of cycles). As result, $\mu_{cu} = f_{cu}/C$ and $\mu_{du} = f_{du}/C$ are the task rates [1/sec] at CU and DU, respectively. As a result, $1/\mu_{cu}$ and $1/\mu_{du}$ represent the mean service time of CU and DU layers, respectively. The processing latency of all data flows at the CU layer (τ^{pro}_{cu}) and DU layer (τ^{pro}_{du}) is computed as:

$$\tau^{\mathsf{pro}}_{cu}[t] = \frac{\Lambda[t]}{\mu_{cu}},\ \text{and}\ \tau^{\mathsf{pro}}_{du}[t] = \frac{\Lambda[t]}{\mu_{du}},\ \forall n \in \mathcal{N} \tag{5}$$

where $\Lambda[t] = \sum_u \lambda_u[t]$. Next, the arrival packets $\lambda_u[t]$ for the u-th user is transported to the DU layer via the midhaul (MH) link with the maximum capacity C^{MH} [bits/sec] between CU and DU. By Burke's theorem, the mean arrival data rate of the second layer, which is processed in the first layer, is still the same rate [30]. Hence, the data transmission latency of the traffic flow for user u under the MH limited capacity is:

$$\tau^{\mathsf{tx}}_{cu,du}[t] = \frac{\Lambda[t]Z}{C^{\mathsf{MH}}}. \tag{6}$$

As mentioned previously, the maximum number of paths from DU n to each user is M_n. Since the packets for user u can be transmitted by multiple RUs, the effective response time $\tau^{\mathsf{tx}}_{du,ru}$ to transport all packets in the DUs layer should be computed by the worst average response time among its connected FH links with maximum capacity C^{FH}_m [bits/sec], *i.e.*,

$$\tau^{\mathsf{tx}}_{du,ru}[t] = \max_m \left\{\frac{\sum_{u\in\mathcal{U}^{\mathsf{ur}}} \varphi_{m,u}[t]\lambda_u[t]Z^{\mathsf{ur}}}{C^{\mathsf{FH}}_m}\right\},\ \forall m \in \mathcal{M}_n. \tag{7}$$

The transmission latency from RU m to user u is then calculated as:

$$\tau^{\mathsf{tx}}_{ru,u}[t_s] = \max_m \left\{\frac{\varphi_{m,u}[t]\lambda_u[t]Z^{\mathsf{ur}}}{r^{\mathsf{ur}}_{m,u}[t_s]}\right\},\ \forall u \in \mathcal{U}^{\mathsf{ur}}. \tag{8}$$

Simply put, the e2e latency of each uRLLC user $u \in \mathcal{U}^{\mathsf{ur}}$ per each TTI is computed as:

$$\begin{aligned}\tau^{\mathsf{ur}}_u[t] = {} & \tau^{\mathsf{pro}}_{cu}[t] + \tau^{\mathsf{tx}}_{cu,du}[t] + \tau^{\mathsf{pro}}_{du}[t] + \tau^{\mathsf{tx}}_{du,ru}[t] \\ & + \sum_{t_s} \big(\tau^{\mathsf{tx}}_{ru,u}[t_s] \\ & + \tau^{\mathsf{pro}}_{ru}[t_s]\big),\ \forall u \in \mathcal{U}^{\mathsf{ur}}\end{aligned} \tag{9}$$

where τ^{pro}_{ru} is the process latency at RU m, which is bounded by three OFDM symbols duration that is typically very small. To ensure a minimum latency requirement for uRLLC user u, the e2e latency is bound by a predetermined threshold D^{ur}_u, *i.e.*, $\tau^{\mathsf{ur}}_u[t] \leq D^{\mathsf{ur}}_u$.

III. Problem Formulation and Overall Intelligent Traffic Steering Algorithm

A. Problem Formulation

1) Utility Function: The ultimate goal is to optimize the joint intelligent traffic prediction, flow-split distribution, dynamic user association, and radio resource management in the presence of unknown dynamic traffic demand to serve eMBB and uRLLC users, subject to various resources constraints and diverse QoS requirements. Due to the conflict of objective functions in both services (*i.e.* eMBB and uRLLC), the utility function should capture the eMBB throughput and worst-user e2e uRLLC latency separately such as $\mathcal{R}^{\mathsf{em}} = \sum_{u\in\mathcal{U}^{\mathsf{em}}} r^{\mathsf{em}}_u(\boldsymbol{p}^{\mathsf{em}}[t_s])$ and $\max_{u\in\mathcal{U}^{\mathsf{ur}}}\{\tau^{\mathsf{ur}}_u\}$ on two independent optimization problems. Based on the above definitions and discussions, the JIFDR problem is mathematically formulated as two independent optimization problems with common constraints as follows:

$$\text{P1}: \max_{\boldsymbol{\lambda},\boldsymbol{\varphi},\boldsymbol{\pi},\boldsymbol{p},\alpha} \quad \mathcal{R}^{\mathsf{em}}(\boldsymbol{p}^{\mathsf{em}}[t_s]) \tag{10a}$$

$$\text{s.t.} \quad \boldsymbol{\pi}[t_s] \in \Pi[t_s],\ \forall t_s \tag{10b}$$

$$\boldsymbol{p}[t_s] \in \mathscr{P}[t_s],\ \forall t_s \tag{10c}$$

$$\boldsymbol{\varphi}_u[t] \in \boldsymbol{\varphi}[t],\ \forall t, u \in \mathcal{U} \tag{10d}$$

$$\sum_{t_s} r^{\mathsf{em}}_u(\boldsymbol{p}^{\mathsf{em}}[t_s]) \geq R_{\mathsf{th}},\ \forall u \in \mathcal{U}^{\mathsf{em}} \tag{10e}$$

$$\sum_u \big[r^{\mathsf{em}}_{m,u}(\boldsymbol{p}^{\mathsf{em}}[t_s]) + r^{\mathsf{ur}}_{m,u}(\boldsymbol{p}^{\mathsf{ur}}[t_s], \boldsymbol{\pi}^{\mathsf{ur}}[t_s])\big] \leq C^{\mathsf{FH}}_m, \forall m \in \mathcal{M}_n \tag{10f}$$

$$\sum_{t_s} r^{\mathsf{ur}}_{m,u}(\boldsymbol{p}^{\mathsf{ur}}[t_s], \boldsymbol{\pi}^{\mathsf{ur}}[t_s]) \geq \frac{\varphi_{m,u}[t]\lambda_u[t]Z^{\mathsf{ur}}}{\Delta}, \quad \forall m \in \mathcal{M}_n, u \in \mathcal{U}^{\mathsf{ur}} \tag{10g}$$

$$\tau^{\mathsf{ur}}_u(\boldsymbol{\lambda}[t], \boldsymbol{\varphi}[t], \boldsymbol{\pi}[t_s], \boldsymbol{p}[t_s]) \leq D^{\mathsf{ur}}_u,\ \forall u \in \mathcal{U}^{\mathsf{ur}} \tag{10h}$$

$$\sum_u q_{m,u}[t_s] \leq Q^{\max},\ \forall t_s, m \in \mathcal{M}_n \tag{10i}$$

$$\sum_{f_i=1}^{F_i} \beta_i \leq B_i[t],\ i \in \{1, 2\} \tag{10j}$$

$$0 \leq \alpha[t] \leq 1 \tag{10k}$$

and

$$\text{P2}: \min_{\boldsymbol{\lambda},\boldsymbol{\varphi},\boldsymbol{\pi},\boldsymbol{p},\alpha} \quad \max_{u\in\mathcal{U}^{\mathsf{ur}}}\{\tau^{\mathsf{ur}}_u\} \tag{11a}$$

$$\text{s.t.} \quad (10b)\text{-}(10k) \tag{11b}$$

where $\boldsymbol{\varphi}[t], \boldsymbol{\pi}[t_s]$ and $\boldsymbol{p}[t_s]$ are the vectors encompassing the flow-split portions, sub-band assignments, and power allocation variables at frame t and TTI t_s, respectively. Recall that, for each BWP with the given numerology, $B_i[t]|_{i=2} = \alpha[t]B$ and $B_i[t]|_{i=1} = (1-\alpha[t])B - B_G$. Constraint (10f) expresses

the limited capacity of FH link between DU n and RU m. Constraint (10g) ensures that each RB assigned to the u-th uRLLC user should transmit a complete data packet with the size Z^{ur}.

B. Challenges of Solving JIFDR Problem

The main challenges in solving problems (P1) and (P2) lie in the non-convexity of τ_u^{ur} and constraints (10f), (10g) and (10i) with respect to flow-split portions and transmit power variables. Furthermore, the binary nature of sub-band allocation variables in constraint (10b) makes these problems more difficult to solve directly, which is generally MINCP. Once may employ the MINCP solvers (*e.g.* Gurobi) to directly solve binary $\boldsymbol{\pi}$. However, we argue that the exponential computation complexity of such a MINCP formulation limits its practical feasibility, especially when the number of variables exceeds few thousand in large-scale scenarios. Besides, the traffic demand $\lambda[t]$ for the next time (frame) is unknown in practice. Such that the BW-split $\alpha[t]$ and flow-split vectors $\varphi[t]$ for frame t will be decided based on the previous states updated by the RAN layer and knowledge of the previous traffic demands $\{\lambda[t-1]\}_{\forall t}$. In order to attain high QoE for all users in each TTI, an efficient and adaptable solution to the long-term subproblem of (10) and (11) is required.

C. Sub-Optimization Problems

It is clear, both problems (10) and (11) must be solved on separate time scales, *i.e.* on the long-term scale t and the short-term scale t_s. To reduce the computational complexity and information sharing as well as to provide a stable queuing system, the traffic demand vector $\boldsymbol{\lambda}[t]$, the flow-split decision vector $\boldsymbol{\varphi}[t]$ and BW-splitting variable $\alpha[t]$ are only solved and updated once per time-frame t. In contrast, the power allocation vector $\boldsymbol{p}[t_s]$ and the RB allocation vector $\boldsymbol{\pi}[t_s]$ are optimized in every TTI t_s, adapting to dynamic environments.

Although having different objective functions, we observe that P1 and P2 can share the solution development. In particular, the P2's objective function can be equivalently transformed to the maximization of the worst rate of the uRLLC services. By approximating the channel dispersion V in (2) as 1 for proper SNR ranges, the uRLLC rate has the same concavity as the eMBB rate in (1). Since both problems P1 and P2 have the same set of constraints, hereafter we propose solution development for only P1 to avoid redundancy.

1) Long-Term Subproblem (L-SP): The joint optimization subproblem of the traffic demand, flow-split distribution, and dynamic RAN slicing at time-scale t is re-expressed as:

$$
\begin{aligned}
cl\text{L-SP}: \max_{\boldsymbol{\lambda},\boldsymbol{\varphi},\alpha} \quad & \mathcal{R}^{\mathsf{em}}(\boldsymbol{p}^{\mathsf{em}}[t_s]) && \text{(12a)}\\
\text{s.t.} \quad & \boldsymbol{\varphi}_u[t] \in \boldsymbol{\varphi}[t], \forall t, u && \text{(12b)}\\
& \sum_{t_s} r_{m,u}^{\mathsf{ur}}(\boldsymbol{p}^{\mathsf{ur}}[t_s], \boldsymbol{\pi}^{\mathsf{ur}}[t_s]) \geq \frac{\varphi_{m,u}[t]\lambda_u[t]Z^{\mathsf{ur}}}{\Delta} && \text{(12c)}\\
& \tau_u^{\mathsf{ur}}(\boldsymbol{\lambda}[t], \boldsymbol{\varphi}[t], \boldsymbol{\pi}[t_s], \boldsymbol{p}[t_s]) \leq D_u^{\mathsf{ur}}, \ \forall u && \text{(12d)}\\
& \sum_{f_i=1}^{F_i} \beta_i \leq B_i[t], \ i \in \{1,2\} && \text{(12e)}\\
& 0 \leq \alpha[t] \leq 1. && \text{(12f)}
\end{aligned}
$$

Although the L-SP (12) is non-convex due to the non-convexity of constraints (12c) and (12d), it cannot be solved directly by standard optimization techniques because $\boldsymbol{\lambda}[t]$ is completely unknown at the beginning of each frame. In the next section, three successive methods are proposed for solving this problem, that predict traffic demand, dynamic BW-split distribution, and dynamic flow-split variables as $\boldsymbol{\lambda}^*[t]$, $\boldsymbol{\alpha}^*[t]$ and $\boldsymbol{\varphi}^*[t]$ at the beginning of each frame t, respectively.

2) Short-Term Subproblem (S-SP): Given $\boldsymbol{\lambda}^*[t]$, $\boldsymbol{\alpha}^*[t]$, and $\boldsymbol{\varphi}^*[t]$ forwarded from the non-RT RIC through the A1 interface, the resource allocation problem at time slot t_s in the near-RT RIC is expressed as:

$$
\begin{aligned}
\text{S-SP}: \max_{\boldsymbol{\pi},\boldsymbol{p}} \quad & \mathcal{R}^{\mathsf{em}}(\boldsymbol{p}^{\mathsf{em}}[t_s]) && \text{(13a)}\\
\text{s.t.} \quad & \text{s.t. } \boldsymbol{\pi}[t_s] \in \Pi[t_s], \ \forall t_s && \text{(13b)}\\
& \boldsymbol{p}[t_s] \in \mathscr{P}[t_s], \forall t_s && \text{(13c)}\\
& \sum_{t_s} r_u^{\mathsf{em}}(\boldsymbol{p}^{\mathsf{em}}[t_s]) \geq R_{\mathsf{th}}, \forall u && \text{(13d)}\\
& \sum_{u} \left[r_{m,u}^{\mathsf{em}}(\boldsymbol{p}^{\mathsf{em}}[t_s]) + r_{m,u}^{\mathsf{ur}}(\boldsymbol{p}^{\mathsf{ur}}[t_s], \boldsymbol{\pi}^{\mathsf{ur}}[t_s])\right] \\
& \leq C_m^{\mathsf{FH}}, \forall m && \text{(13e)}\\
& \sum_{t_s} r_{m,u}^{\mathsf{ur}}(\boldsymbol{p}^{\mathsf{ur}}[t_s], \boldsymbol{\pi}^{\mathsf{ur}}[t_s]) \geq \psi, \forall m, u && \text{(13f)}\\
& \tau_u^{\mathsf{ur}}(\boldsymbol{\pi}[t_s], \boldsymbol{p}[t_s]) \leq D_u^{\mathsf{ur}}, \forall u && \text{(13g)}\\
& \sum_{u} q_{m,u}[t_s] \leq Q^{\max}, \forall t_s, m \in \mathscr{M}_n && \text{(13h)}
\end{aligned}
$$

where $\psi = \frac{\varphi_{m,u}^*[t]\lambda_u^*[t]Z^{\mathsf{ur}}}{\Delta}$. The S-SP (13) involves both binary ($\boldsymbol{\pi}$) and continuous ($\boldsymbol{p}$) optimization variables with nonlinear objective function and non-convex constraint (13e) at time slot t_s, which is still remained a MINCP problem. Since MINCP problems incorporate the optimizing challenges under integer variables with managing nonlinear functions, such problems comprise an immense class of difficult optimization problems.

D. Overall Intelligent Traffic Steering Deployment Architecture and Algorithm

In Fig. 4, we show the high-level organization of deployment scenarios and the end-to-end flow of the proposed algorithm within the ORAN architecture. This is inspired by the second set of deployment scenarios listed in the technical report [31] by the ORAN Alliance.

1. The collected data, including performances/observations and resource updates from RAN components and near-RT RIC, are collected into a data collector located at the SMO. This process is done via the O1 interface. Based on these collected data in SMO, three rAPPs for solving L-SP are carried out at non-RT RIC. For $t = 1$, we assume a random traffic demand with a Poisson process and equal flow-split decision for all paths.
2. Utilizing a data bus like Kafka, the collected data at the SMO is routed to non-RT RIC in the SMO.
3. The non-RT RIC queries the relevant ML/AI model, which is hosted in the AI server within the SMO. Once

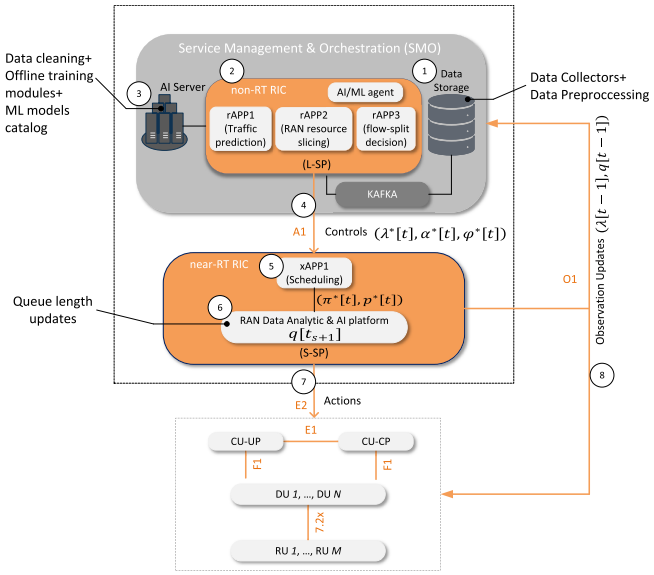

Fig. 4. High-level structure of deploying the proposed intelligent traffic prediction and JIFDR management scheme within the ORAN architecture.

the model has been well-trained on the AI server, non-RT RIC is notified of the inference.

(4) The scheduling xAPP in near-RT RIC is then loaded with inference results and policies via the A1 interface. Applications, that are designed specifically for radio functions or xAPPs, enable RAN components to be programmed.

(5) Given $\boldsymbol{\lambda}^*[t]$, $\boldsymbol{\alpha}^*[t]$ and $\boldsymbol{\varphi}^*[t]$, xAPP1 deployed in near-RT RIC controls congestion through MC technique and optimizes RAN resources and functions in each time-slot t_s by solving S-SP to obtain optimal solutions of RB allocation $\boldsymbol{\pi}^*[t_s]$ and power allocation $\boldsymbol{p}^*[t_s]$.

(6) Subsequently, the RAN Data Analytic component in near-RT RIC updates queue lengths.

(7) Through the E2 interface, the relevant solution is transferred to CU or DU layers.

(8) After S_i TTI (*i.e.* one frame), the performance and observations (*e.g.* $\boldsymbol{q}[t-1]$, $\boldsymbol{\lambda}[t-1]$) are updated to SMO through the O1 interface to re-estimate the traffic demand $\boldsymbol{\lambda}^*[t+1]$ and flow-split decision $\boldsymbol{\varphi}^*[t+1]$.

The overall intelligent TS algorithm to solve the JIFDR problem (10) is summarized in Algorithm 1, where the solutions for subproblems will be detailed in Section IV. It is straightforward to develop a similar procedure to solve problem (11).

IV. Proposed Frameworks for Solving Subproblems

We are now in a position to solve the L-SP and S-SP on different time scales. The optimal solutions for all optimization variables (α $\boldsymbol{\varphi}$, $\boldsymbol{\pi}$ and $\boldsymbol{p}$) strongly depend on the predicted traffic demand vector $\boldsymbol{\lambda}$, which often require prior knowledge of the actual traffic of all services stored at data collector in SMO. Moreover, due to the dynamic environment and data collected from the RAN components being only updated to non-RT RIC on a long-term scale (i.e., frame), the assumption of complete information is unrealistic. In this paper, we aim to leverage observable historical system knowledge gathered over previous time slots via the O1 interface to build a smoother optimal response to maximize the long-term utility.

Algorithm 1 Proposed Intelligent Traffic Steering Algorithm to Solve JIFDR Problem (10)

Initialization: Set $t = 1$, $t_s = 1$, $\varphi_u[1] = \frac{1}{M}[1, \ldots, 1]$ and $\alpha[1] = \frac{1}{2}$; all initial queues are set to be empty $q_{m,u}[1] = 0$ and $\boldsymbol{q}[1] = 0$.

for $t = 1, 2, \ldots, T$ **do**

  Traffic demand prediction: Given $(\boldsymbol{\lambda}[t-1], \boldsymbol{q}[t-1])$, non-RT RIC splits the available of all RUs' BW and traffic flows of all users by (14) and (15) based on the predicted traffic demand (or arrival data rate) $\boldsymbol{\lambda}^*[t]$ by solving the L-SP (12)

  for $t_s = 1, 2, \ldots, S_i$ with $s \in \{1, 2, \ldots, S_i\}$ **do**

    Optimizing scheduling: Given the queue-length vector $\boldsymbol{q}[t_s]$, and all long-term variables such as $(\boldsymbol{\lambda}^*[t], \boldsymbol{\alpha}^*[t]$, and $\boldsymbol{\varphi}^*[t])$, solve the problem (16) by Algorithm 2 to obtain the RB assignment $(\boldsymbol{\pi}^*)$ and power allocation $(\boldsymbol{p}^*)$

    Updating queue-lengths: Queue-lengths are updated as

$$q_{m,u}[t_{s+1}] = \max\{\big[q_{m,u}[t_s] + \varphi_{m,u}[t]\lambda_u[t]Z^{\mathsf{x}}\delta_i - r^{\mathsf{x}}_{m,u}[t_s]\delta_i\big], 0\}$$

    where $\mathsf{x} \in \{\mathsf{ur}, \mathsf{em}\}$.

    Set $s = s + 1$

  end for

  Update $\{\boldsymbol{q}[t], \boldsymbol{\lambda}[t]\} = \{q_{m,u}[t], \lambda_u[t]\}$, $\forall u \in \mathcal{U}$, $m \in \mathcal{M}_n$

  Set $t = t + 1$

end for

A. LSTM for Solving L-SP

As mentioned previously, the L-SP cannot be solved directly by standard optimization techniques since $\boldsymbol{\lambda}[t]$ and $\boldsymbol{q}[t_s]$ are often unknown at the beginning of each frame. Besides, the main challenge in optimizing traffic steering is to predict traffic precisely before the beginning of the next frame. An optimal policy cannot be implemented with an imprecise prediction of future traffic. In this section, utilizing a deep learning approach, we develop a data-driven real-time traffic demand prediction method. We suppose that the queue length of data flows u in the next frame will depend on the traffic demand of data flow u in the current and previous ones. Basically, RNN models utilize the current input as well as the output of one layer as the input for the subsequent layer. In such models, each layer is fed by the very first layer's input. This allows the RNN model to learn from the current and former time steps and then provides more precise predictions for traffic flows. These standard RNN models suffer from short-term memory owing to the vanishing and exploding gradient problems, which appear with longer data sequences. Due to these difficulties, the gradient either entirely disappears or explodes to a very high value, which makes them difficult to learn

some long-period dependencies. To address the long-term dependency issue, the LSTM model has seen extensive use in the field of traffic prediction due to its capabilities in dealing with long time-series flow data. As a result, we utilize the LSTM RNN to learn and predict the traffic pattern of all users in the considered ORAN architecture.

The fact that LSTM includes a memory cell to keep observable data, allows them to handle long-term time series. As shown in Fig. 5, the structure of standard LSTM cells learns through four main gates, *namely* input (i_g), forget (f_g), cell state-update (c_g) and output (o_g), that allows the input data to pass from the previous cells in the learning procedure. The output calculated by the input gate (i_g) and the cell state update (c_g) modify the current cell's state ($c[t]$), while the forget gate enables the current cell to discard or preserve the previous state value. To determine this, we take into account the output of the previous hidden state ($\mathcal{H}[t-1]$) and the actual input data ($\boldsymbol{\lambda}[t-1]$). The new cell state's value is based on the actual input and previous output of the cell. In contrast to other gates that employ the Sigmoid function, the cell state update benefits the hyperbolic tangent as an activation function that yields values between -1 and 1. Eventually, the input, forget, and cell state update gates are combined to create the current cell state. The current cell's output is determined as a function of the previous timestep's output ($\mathcal{H}[t-1]$), the actual input data ($\boldsymbol{\lambda}[t-1]$), and the cell state ($c[t-1]$) through the output gate. Lastly, after crossing through an activation function, the prediction value is calculated. Each LSTM layer comprises a chain of LSTM cells, in which the computed operation of each cell is transmitted to the next cell as an input. As illustrated in Fig. 5, the temporal pattern of the mentioned parameter is learned through the current and a window of previous traffic demands value with the length W $\{\boldsymbol{\lambda[t-W]}, \boldsymbol{\lambda[t-W+1]}, \ldots, \boldsymbol{\lambda[t-1]}\}$ to predict future values.

The LSTM model is trained at non-RT RIC in the ORAN architecture, using long-term data gathered from RAN via O1. The near-RT RIC of the ORAN is then given access via the A1 interface to the trained model for inference. Upon the inference outcome, the intelligent TS is applied through the MC technique to enhance the associated key performance indicators (KPIs). Traffic demand prediction and the corresponding intelligent TS schemes are continually implemented till the desired KPI values, or the required QoS of traffic are met. In the following, the network parameter of data arrival rate $\boldsymbol{\lambda}$ is continuously monitored across all cells of RUs. Upon predicting the data arrival rate per frame, the flow-split distribution, dynamic RAN slicing, and radio resource management with the MC technique can be applied to steer data flows. The weights of the RNN model are eventually updated depending on the actual parameter's value to reflect changes and enhance the performance till the goal KPI criteria are met if the prediction outcome is incorrect.

B. Heuristic Methods for Predicting $\boldsymbol{\alpha}[t]$ and $\boldsymbol{\varphi}[t]$

Upon the inference outcome of the LSTM model, the predicted traffic demands at the next frame $\boldsymbol{\lambda}^*[t]$ are transmitted immediately to two other embedded rAPPs in non-RT RIC for optimizing the dynamic bandwidth separation, $\boldsymbol{\alpha}[t]$ and flow-split decisions, $\boldsymbol{\varphi}[t]$. For efficient deployment, these parameters are designed in a longer time scale, *i.e.*, on the frame basis compared to the time slot basis of power allocation and resource block assignment. Therefore, at the beginning of each frame, $\boldsymbol{\alpha}[t]$ and $\boldsymbol{\varphi}[t]$ should be determined upon getting the predicted traffic demands. Having optimum values of the bandwidth separation and flow split is very difficult if not possible because of the unknown CSI of future time slots in the current frame. Therefore, we propose an efficient heuristic algorithm to determine $\boldsymbol{\alpha}[t]$ and $\boldsymbol{\varphi}[t]$ based on $\boldsymbol{\lambda}^*[t]$. An intuitive way is to allocate the bandwidth to each service proportionally to the corresponding traffic demands. However, since the amount of uRLLC traffics is much smaller than the amount of eMBB traffic, this method is not efficient in meeting the stringent latency requirement of uRLLC applications. To tackle this, we incorporate the maximum tolerable delays of both services and the total traffic demands. Thus, the bandwidth separation between eMBB and URLLC services is computed as follows:

$$\alpha^*[t] = \frac{\sum_{\mathcal{U}^{\mathsf{ur}}} \lambda_u^*[t]}{\sum_{\mathcal{U}^{\mathsf{em}}} \lambda_u^*[t]} \times \frac{\tau_{th}^{\mathsf{em}}}{\tau_{th}^{\mathsf{ur}}} \tag{14}$$

where τ_{th}^{ur} and τ_{th}^{em} represent the maximum allowed latency for uRLLC and eMBB services, respectively. To plan the flow splitting factor $\varphi_u[t]$, we consider each DU's capacity in delivering user traffic demands u. Because we do not know the data rate for the user in the next frame, we take the moving average of the rate in the most recent time slots. For a generic user u (can be uRLLC or eMBB user), let us define $\bar{r}_{m,u}[t] = \frac{1}{W}\sum_{l=t-W+1}^{t} r_{m,u}[l]$, where $r_{m,u}[l]$ is the achievable rate of user u served RU m at time slot l, and W is the window size. The flow split for user u to RU m is computed as follows:

$$\varphi_{m,u}^*[t] = \frac{\bar{r}_{m,u}[t]}{\sum_{m\in\mathcal{M}_n} \bar{r}_{m,u}[t]}, \ \forall m, u. \tag{15}$$

C. SCA-Based Iterative Algorithm for Solving S-SP

To solve the problem (13) as a MINCP, we first relax binary variables to continuous ones (*i.e.* the box constraints between 0 and 1) and transform constraint (13e) into a more traceable form which the SCA-based iterative algorithm can efficiently solve.

1) Penalty Function: We bring forward the following penalty function to accelerate the convergence of the proposed iterative algorithm that will be detailed shortly $\mathcal{P}(\boldsymbol{\pi}) = \sum_{t_s,f_i,m,u}\big[(\pi_{m,u,f_i}^{\mathsf{em}}[t_s])^2 + (\pi_{m,u,f_i}^{\mathsf{ur}}[t_s])^2 - \pi_{m,u,f_i}^{\mathsf{em}}[t_s] - \pi_{m,u,f_i}^{\mathsf{ur}}[t_s]\big]$ which is convex in $\boldsymbol{\pi}[t_s]$. It is clear that $\mathcal{P}(\boldsymbol{\pi}) \leq 0$ for any $\pi_{m,u,f_i}^{\mathsf{x}}[t_s] \in [0,1]$, which is useful to penalize the relaxed variables to obtain near-precise binary solutions at optimum (*i.e.* satisfying (13b)). By incorporating $\mathcal{P}(\boldsymbol{\pi})$ into the objective function of (13b), the parameterized relaxed problem is expressed as:

$$\text{S-SP1}: \max_{\boldsymbol{\pi},\boldsymbol{p}} \quad \mathcal{R}^{\mathsf{em}} + \omega\mathcal{P}(\boldsymbol{\pi}) \tag{16a}$$

$$\text{s.t.} \quad \boldsymbol{\pi}[t_s] \in \tilde{\Pi}[t_s], \quad \forall t_s, \forall u \in \mathcal{U} \tag{16b}$$

$$(13c)\text{-}(13h) \tag{16c}$$

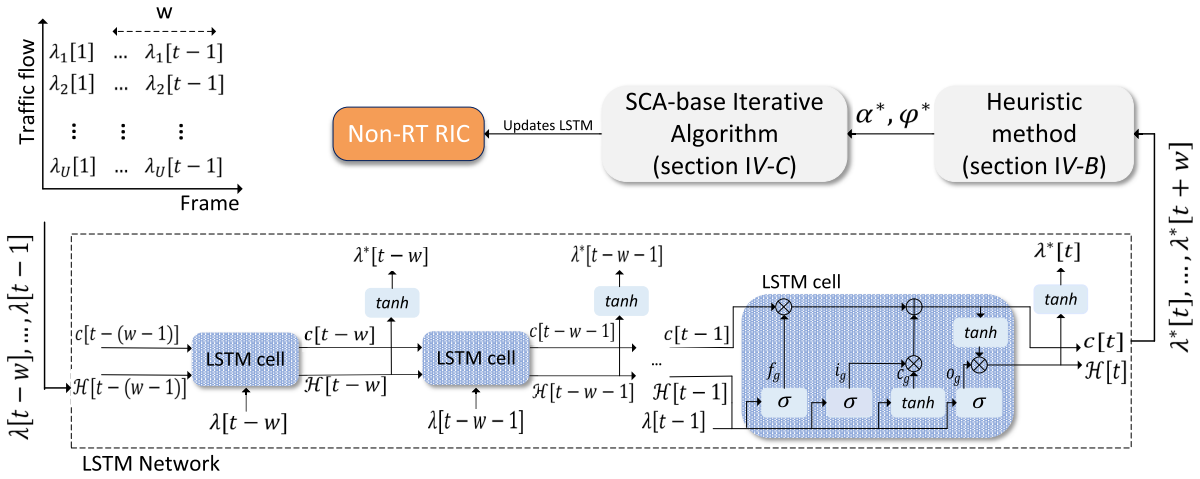

Fig. 5. Implementing the proposed JIFDR management scheme at time-frame t.

Algorithm 2 The Proposed SCA-Based Iterative Algorithm to Solve S-SP (16)

Initialization: Set $j := 0$ and generate initial feasible points for $(\boldsymbol{\pi}^{(0)}[t_s], \boldsymbol{p}^{(0)}[t_s]) := (\boldsymbol{\pi}[t_{s-1}], \boldsymbol{p}[t_{s-1}])$ to constraints in S-SP2 (19)

1: **repeat**
2: Solve (19) to obtain $(\boldsymbol{\pi}^*[t_s], \boldsymbol{p}^*[t_s])$ and $\Xi^*[t_s]$;
3: Update $(\boldsymbol{\pi}^{(j)}[t_s], \boldsymbol{p}^{(j)}[t_s]) := (\boldsymbol{\pi}^*[t_s], \boldsymbol{p}^*[t_s])$ and $\Xi^{(j)}[t_s] := \Xi^*[t_s]$;
4: Set $j := j + 1$;
5: **until** Convergence or $|\Xi^{(j)}[t_s] - \Xi^{(j-1)}[t_s]| \leq \epsilon$ {/*Satisfying a given accuracy level*/}
6: Recover an exact binary by computing $\boldsymbol{\pi}^*[t_s] = \lfloor \boldsymbol{\pi}^{(j)}[t_s] + 0.5 \rfloor$ and repeat step 1 to 5 for given $\boldsymbol{\pi}^*[t_s]$;
7: **Output:** $(\boldsymbol{\pi}^*[t_s], \boldsymbol{p}^*[t_s])$.

where $\tilde{\Pi}[t_s] \triangleq \{\pi^{\mathsf{em}}_{m,u,f_i}[t_s], \pi^{\mathsf{ur}}_{m,u,f_i}[t_s] \in [0,1] | \sum_{m,u}[\pi^{\mathsf{em}}_{m,u,f_i}[t_s] + \pi^{\mathsf{ur}}_{m,u,f_i}[t_s]] \leq 1\}$ and $\omega > 0$ denotes a determined penalty parameter.

Proposition 1: Problems (13) and (16) share the same optimal solution, i.e., $(\boldsymbol{\pi}^*, \boldsymbol{p}^*)$, considering an suitable positive value of ω.

The proof is directly followed [32] by showing the fact that $\mathcal{P}(\boldsymbol{\pi}) = 0$ at optimum in maximizing of the objective function (16). It implies that a constant ω always exists to guarantee that $\boldsymbol{\pi}$ are binary at optimum, and the relaxation is tight. Practically, it is acceptable if $\mathcal{P}(\boldsymbol{\pi}) \leq \varepsilon$ for a tiny ε, which results in a nearly precise optimal solution.

In problem (16), the objective function is non-concave due to $\mathcal{P}(\boldsymbol{\pi})$, while constraints (13e) is non-convex. Based on the SCA method, the first-order Taylor approximation is used to linearize the function $\mathcal{P}(\boldsymbol{\pi})$ at the j-th iteration as follows:

$$\begin{aligned}\mathcal{P}^{(j)}(\boldsymbol{\pi}) &\triangleq \sum_{m,u,f_i} \big[\pi^{\mathsf{em}}_{m,u,f_i}[t_s](2\pi^{\mathsf{em},(j)}_{m,u,f_i}[t_s] - 1) - (\pi^{\mathsf{em},(j)}_{m,u,f_i}[t_s])^2 \\ &\quad + \pi^{\mathsf{ur}}_{m,u,f_i}[t_s](2\pi^{\mathsf{ur},(j)}_{m,u,f_i}[t_s] - 1) - (\pi^{\mathsf{ur},(j)}_{m,u,f_i}[t_s])^2\big] \quad (17)\end{aligned}$$

where $\mathcal{P}(\boldsymbol{\pi}) \geq \mathcal{P}^{(j)}(\boldsymbol{\pi})$ and $\mathcal{P}(\boldsymbol{\pi}^{(j)}) = \mathcal{P}^{(j)}(\boldsymbol{\pi}^{(j)})$.

To address constraint (13e), we indicate its LHS as $r_m(\boldsymbol{p}[t_s]) \triangleq \sum_u \big[r^{\mathsf{em}}_{m,u}(\boldsymbol{p}^{\mathsf{em}}[t_s]) + r^{\mathsf{ur}}_{m,u}(\boldsymbol{p}^{\mathsf{ur}}[t_s], \boldsymbol{\pi}^{\mathsf{ur}}[t_s])\big]$, which is concave in $\boldsymbol{p}[t_s]$. Thus, the function $r_m(\boldsymbol{p}[t_s])$ can be approximated at the feasible point $\boldsymbol{p}^{(j)}[t_s]$ as

$$\begin{aligned}&r^{(j)}_m(\boldsymbol{p}[t_s]) \\ &\quad \triangleq r_m(\boldsymbol{p}^{(j)}[t_s]) - \sum_{u,f_i} \beta_i \frac{\pi^{\mathsf{ur}}_{m,u,f_i}[t_s] Q^{-1}(P_e)}{\sqrt{\delta_i \beta_i}} \\ &\quad + \frac{\beta_i}{\ln 2} \sum_{u,f_i,\mathsf{x}} (p^{\mathsf{x}}_{m,u,f_i}[t_s] - p^{\mathsf{x},(j)}_{m,u,f_i}[t_s]) \\ &\quad \times \left[\frac{g_{m,u,f_i}[t_s]}{N_0 + p^{\mathsf{x},(j)}_{m,u,f_i} g_{m,u,f_i}[t_s]}\right]. \quad (18)\end{aligned}$$

The convex approximate program of (16) solved at iteration j is stated as follows, taking into account all the aforementioned approximations:

$$\begin{aligned}\text{S-SP2}: \max_{\boldsymbol{\pi},\boldsymbol{p}} \quad & \Xi^{(j)} \triangleq \mathcal{R}^{\mathsf{em}} + \omega \mathcal{P}^{(j)}(\boldsymbol{\pi}) && (19a)\\ \text{s.t.} \quad & (13c), (13d), (13f) - (13h), (16b) && (19b)\\ & r^{(j)}_m(\boldsymbol{p}[t_s]) \leq C^{\mathsf{FH}}_m, \forall m \in \mathcal{M}_n. && (19c)\end{aligned}$$

Algorithm 2 provides a summary of the SCA-based iterative algorithm. Step 6 is used to recover an exact binary solution then Steps 1–5 are repeated to refine the final solution in order to ensure a feasible solution to the problem (16). The study gap to the global optimal solution is not considered in this work and is left for future study.

Convergence and complexity analysis: The development of the proposed iterative Algorithm 2 is based on the SCA method [33]. The approximations in (17) and (18) are satisfied the three key inner approximation properties given in [34], while other constraints are already linear and quadratic. In particular, the solution of (19) is always feasible to the parameterized relaxed problem (16) but not vice versa. In addition, Algorithm 2 generates a sequence of the improved solutions $\{\boldsymbol{\pi}^{(j)}, \boldsymbol{p}^{(j)}\}$ in the sense that $\Xi^{(j+1)} \geq \Xi^{(j)}, \forall j$. By [33, Theorem 1], if the number of iterations is sufficiently large, the sequence $\{\boldsymbol{\pi}^{(j)}, \boldsymbol{p}^{(j)}\}$ converges to at least a local

TABLE II
SIMULATION PARAMETERS

Parameter	Value	Parameter	Value
No. of RUs	4	Predetermined uRLLC latency (D_{ur})	0.5 ms
No. of eMBB users	12	Predetermined eMBB throughput (R_{th})	1 Mbps
No. of uRLLC users	8	Maximum FH capacity (C^{FH})	100 Mbps
BW of RU	20 MHz	Maximum MH capacity (C^{MH})	5 Gbps
Error probability (P_e)	10^{-3}	Maximum RU's queue-length (Q^{max})	10 KB
Power of RU	46 dBm	No. of LSTM layer	2
Noise power (N_0)	-110 dBm	No. of LSTM unit	50
uRLLC packet size (Z^{ur})	1 KB	No. of epoch	50
eMBB packet size (Z^{em})	125 KB	Activation function	*tanh*
Length of time-frame	10 ms	Optimizer	*adam*

optimal solution of (16), satisfying the Karush-Kuhn-Tucker (KKT) conditions [33, Theorem 1]. On the other hand, for each numerology i, the convex approximate program (19) has $2MUF_i$ scalar decision variables and $2MUF_i + 4M + 3U$ linear and quadratic constraints. As a result, the worst-case computation complexity of Algorithm 2 in each iteration is estimated as $\mathcal{O}\big(\sqrt{2MUF_i + 4M + 3U}(2MUF_i)^3\big)$, following the interior-point method [35, Chapter 6].

V. PERFORMANCE EVALUATIONS AND NUMERICAL RESULTS

A. Simulation Setup and Parameters

We consider a scenario where all users are uniformly distributed in a circular area with a radius of 500 m, while the locations of RUs are fixed. One RU is located in the central area, serving three sectors, each of which includes one RU. The RU-user channels are generated as Rayleigh fading with the path-loss $\mathrm{PL}_{\mathrm{RU-USER}} = 128.1 + 37.6\log_{10}(d/1000)$ dB. The penalty factor is set to decrease after each TTI as $\omega[t_s] = 20+10/(1+t_s)$ to guarantee the convergence of the short-term subproblem. To estimate the future traffic for the upcoming frames, an RNN model's parameters, which include 2 fully connected hidden layers and 50 LSTM units (neurons), are trained. The operators can configure these parameters based on the provided data and its periodicity. In our setup, the Poisson traffic model has been used to generate traffic for both eMBB and uRLLC services. The RNN training is carried out over the traffic dataset of the cellular network following a Poisson distribution, with the mean arrival rates of 20 and 2.5 for eMBB and uRLLC traffics, respectively [18]. The mean arrival rate is a configurable parameter of the simulator. Incoming traffic packets are sorted in a first-come-first-serve buffer. The dataset contains network measurement in terms of arrival rate collected from M RUs, over a horizon of $T = 10000$ traffic observations over a duration of 100 seconds. The open-source, high-level TensorFlow version 1.13.1 application programming interface, Keras, is used to implement the RNN model. All experiments are done on a Dell desktop computer with an Intel R CPU @ 3.0 GHz. Simulation parameters including the LSTM model are summarized in Table II.

We put into practice the following five benchmark schemes for performance comparison:

1) *Fixed numerology (FIX-NUM)*: In this scheme, the TTI is considered the same for both services as the LTE standard (*i.e.* 0.5 ms) with the SCS of 180 kHz. The resource allocation, flow-split decision, and dynamic BW-split for both traffic follow Algorithm 1 with some slight modifications.
2) *Equal Flow-Split Distribution (EFSD)*: In order to demonstrate the importance of optimizing the flow-split distribution per frame, this scheme considers the equal flow-split for each traffic to RUs, *i.e.* $\varphi_{m,u} = \frac{1}{M}$, $\forall u \in \mathcal{U}$ and follows Algorithm 1.
3) *Equal Power Allocation (EPA)*: The RBs' allocation $\boldsymbol{\pi}$ is optimized by Algorithm 1 for an equal power allocated to all users and subcarriers.
4) *Single Connectivity with uRLLC Priority (SCUP)*: To reveal the performance improvement of MC in heterogeneous wireless networks, this scheme provides the single connectivity (SC) scheme with uRLLC priority in the presence of interference. Due to the stringent requirement of latency, uRLLC will be predominantly guaranteed, and then the remaining resources will be occupied by eMBB users. In this regard, this scheme considers M RUs with disjoint dedicated users while following Algorithm 1.
5) *Proposed Problem in Presence of Known Traffic Demand (PKTD)*: This scheme investigates the performance of both traffics in the presence of known traffic demand λ. In practice, the obtained results of this scheme in the presence of unknown traffic demands show the accuracy of the LSTM model of the proposed method.

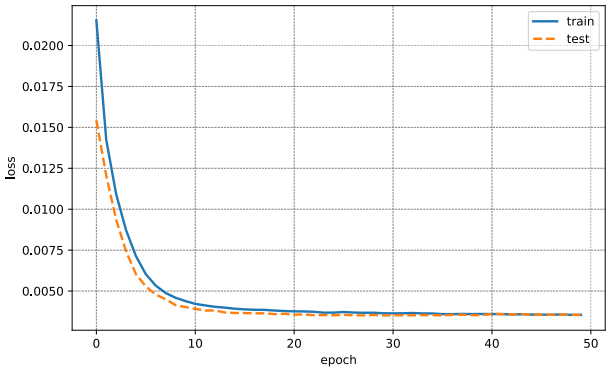

Fig. 6. Training and validation loss for the LSTM RNN model.

B. Numerical Results and Discussions

First, in order to investigate the LSTM's convergence, we monitor the value of the loss function as MSE and keep the training process until the training loss is typically identical to the validation loss after a specific number of epochs. Since the mean arrival rates of both traffics are not in the same range, we normalize traffic demands in the pre-processing phase through the *MinMaxScaler* normalization method from Sklearn. We then divide data into two sets, which are 80% for training and 20% for validation. Fig. 6 plots the training and validation losses for the LSTM model with the most suitable turning hyperparameters, which converge after 50 epochs. It should be mentioned that setting the desirable number of epochs prevents model overfitting. From Table III, we find that the activation function of *tanh* works better than *relu* and *sigmoid*. In the same condition, increasing the number of LSTM layers and decreasing the number of units per layer do

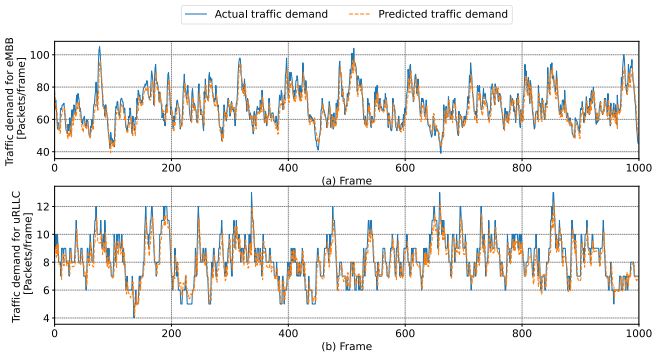

Fig. 7. Traffic demand prediction in ORAN.

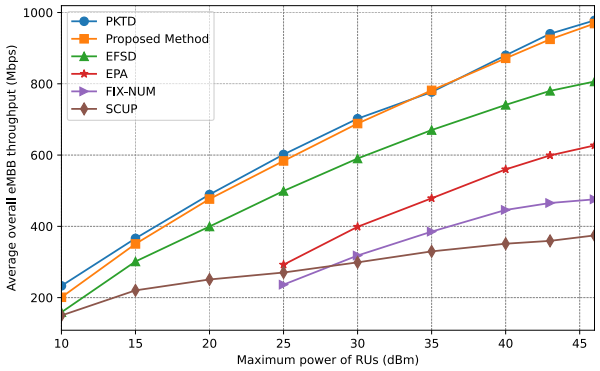

Fig. 8. Average overall eMBB throughput versus $P^{\max}$.

TABLE III
HYPERPARAMETERS FOR THE DIFFERENT PERFORMING LSTM MODELS

No. of LSTM layers	No. of LSTM units in each layer	No. of epochs	Activation function	MSE
2	20	30	*relu*	0.00641
2	50	30	*relu*	0.00382
3	50	100	*relu*	0.00493
3	50	30	*sigmoid*	0.01281
2	50	30	*sigmoid*	0.00782
2	50	100	*tanh*	0.00421
3	20	30	*tanh*	0.00613
2	**50**	**50**	***tanh***	**0.00331**

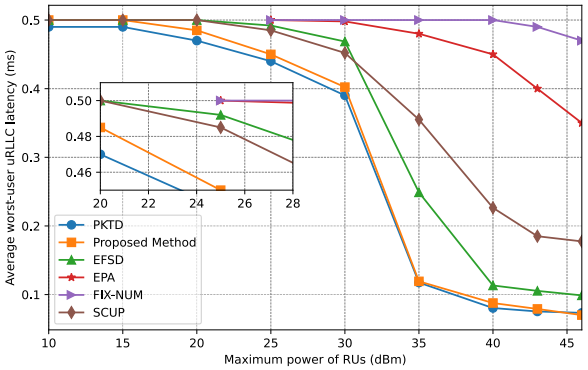

Fig. 9. Average worst-user uRLLC latency versus $P^{\max}$.

not help reduce the MSE value. Based on the search result, the *adam* optimizer converges faster than others, whereas it takes less time for the model's training. In our case, the dropout value is 0.01 for both hidden layers. As a result, Table III shows the search parameters to find the best parameters for the final LSTM-RNN model.

The effectiveness of the LSTM RNN model in both traffic demands is represented in Fig. 7 to illustrate the performance of the ML model prediction. The actual and predicted values for one of the eMBB and uRLLC traffic demands in the proposed system model are shown in Fig. 7 (a) and Fig. 7 (b), respectively. As it is clear from these figures, the trained LSTM-RNN model performs outstandingly in capturing the dynamic traffic demand of services over time. The difference between predicted and actual traffic demands is entirely small. The MSE value has been calculated as a performance measurement to validate the accuracy of the implemented LSTM model. For instance, the measured MSE values of the selected eMBB users in Fig. 7 (a) and uRLLC users in Fig. 7 (b) are 0.00315 and 0.00323, respectively.

To evaluate the eMBB throughput with different resource allocation schemes, Fig. 8 illustrates the sum throughput of eMBB users over different maximum RUs' power budgets from 10 to 46 dBm. Unsurprisingly, the PKTD provides the best performance and acts as the upper bound of all strategies. It can be observed that the gap between our proposed framework and PKTD is less than 2%, which proves the efficiency of the LSTM RNN model in predicting the dynamic traffic demand over time. Whereas the proposed method provides the highest eMBB throughput compared to other benchmark schemes in Fig. 8. Compared to SCUP, FIX-NUM, EPA, and EFSD, the proposed method offers 130.89%, 116.32%, 71.92% and 19.21% gains at the typical power value of $P^{\max} = 30$ dBm, respectively. Furthermore, EPA and FIX-NUM work over $P^{\max} \geq 25$ dBm, while they are infeasible when the maximum RUs' power is less than 25 dBm. Hence, this phenomenon shows the advantage of our proposed method over these schemes, especially at a small $P^{\max}$. Besides, as we mentioned previously, the MC technique plays a vital role in enhancing the eMBB throughput. The gap between the JIFDR framework considering the MC technique and SCUP grows with increasing the maximum power budget of RUs. While the overall eMBB throughput obtained via JIFDR, EFSD, and SCUP are close at $P^{\max} = 10$ dBm, by increasing $P^{\max}$, the MC-based schemes of JIFDR and EFSD significantly exceed that of SCUP.

In order to show the performance of the proposed method on uRLLC latency, Fig. 9 represents the worst-user uRLLC latency under different maximum power of RUs. Similar to the first optimization problem (P1), increasing the maximum power of RUs significantly affects the eMBB throughput improvement, resulting in an efficient reduction of uRLLC latency in the second optimization problem (P2). As we can see from Fig. 9, the uRLLC latency of the proposed method is almost equal to PKTD, which again confirms the accuracy of the LSTM RNN model in predicting the dynamic traffic demand. The performance gain in terms of latency of the proposed method is 181.32% and 49.47% Compared to SCUP and EFSD at $P^{\max} = 40$ dBm. According to the empty region of two benchmark schemes, FIX-NUM and EPA in the range $P^{\max} \leq 25$ dBm, results from Fig. 9 show that

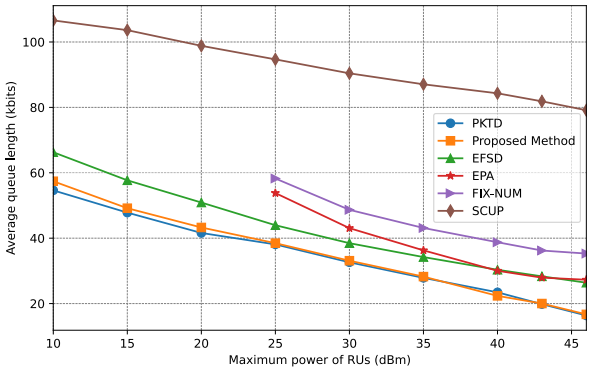

Fig. 10. Average of queue lengths versus $P^{\max}$.

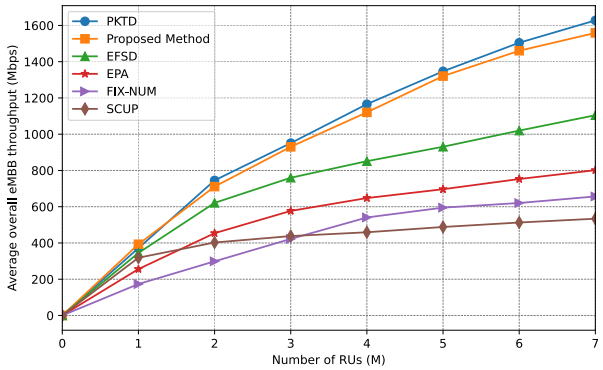

Fig. 11. Average of eMBB throughput versus number of RUs (M) with considering 21 eMBB users and 14 uRLLC users.

these schemes are infeasible over the mentioned range of $P^{\max}$ while having a significant difference in uRLLC latency with the proposed method. Clearly, the EFSD scheme in Fig. 9 greatly outperforms the SCUP scheme. On the one hand, the uRLLC and eMBB traffic are sliced in various virtual slices in SCUP, while the size of the uRLLC traffic packet is considerably smaller than the eMBB packet size. Hence, the assigned slice to uRLLC could meet the uRLLC traffics' requirements alone without waiting in a queue. On the other hand, the SCUP scheme is not able to aggregate multiple links and allow users to connect to more than one RU to achieve the highest throughput.

Fig. 10 depicts the average backlog in the queue under the maximum power budget of RUs with different benchmark schemes. As can be seen, the higher the power budget $P^{\max}$, the lower the average queue length. Similar to the two previous figures, results from the proposed method and PKTD are very close to each other. As expected, the SCUP scheme yields the worst performance in terms of the average queue length, whereas the proposed method yields the best one in Fig. 10. Two FIX-NUM and EPA schemes are infeasible when $P^{\max} < 25$ dBm. Clearly, the EFSD benchmark scheme performs in a better way rather than FIX-NUM, EPA, and SCUP schemes; while EFSD and EPA provide a very close performance to each other for $P^{\max} > 35$ dBm. On the other hand, during the joint scheduling of uRLLC and eMBB traffics, we have numerically observed that uRLLC users always prefer to have only one link in various system setups. This issue indicates that a single connection is generally the best option for traffic with a small data packet size. In contrast, the MC technique is typically a nice option for traffic with high data packet size, *i.e.* eMBB.

As we mentioned before, the MC is an effective technique to improve the data rate for eMBB traffic, especially when the system model faces a limited bandwidth. To demonstrate this, Fig. 11 shows the impact of the increasing number of RUs on overall eMBB throughput. All simulation parameters are assumed unchanged during the simulation of Fig. 11, except the number of eMBB and uRLLC users which are considered 21 and 14, respectively. It is clear from Fig. 11, the eMBB throughput rises with the number of RUs which means an increase in the number of available RBs. As we expected, the PKTD also works as an upper bound for all schemes,

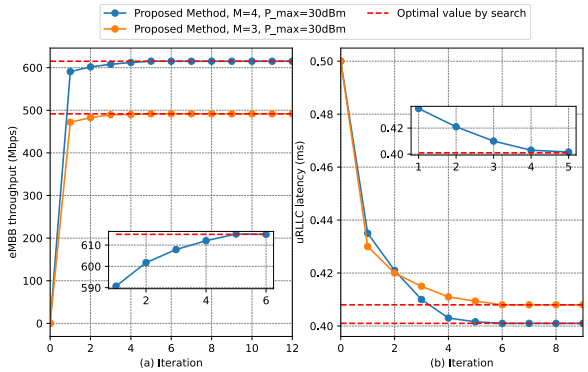

Fig. 12. Convergence behaviour of the proposed Algorithm 2.

regardless of the number of RUs. The small gap between our proposed framework and PKTD (about 2%) shows the high accuracy of traffic prediction by embedded LSTM in the non-RT-RIC component. Compared to other existing benchmark schemes, the proposed method offers the highest throughput. Due to the crucial role of MC in the network, SCUP has the worst performance among all the schemes, with increasing the number of RUs ($M \geq 3$) in the network model. For $M = 5$, the schemes with MC (i.e. the proposed method, EFSD, EPA, and FIX-NUM) have the performance gain of 168.2%, 90.67%, 42.68%, and 19.45% relatively compared to the without MC i.e., SCUP. It should be noted that the SCUP outperforms FIX-NUM at $M \leq 3$, demonstrating the advantage of mixed numerology over fixed numerology. However, as the number of RUs increases to $M \geq 3$, the SCUP is no longer able to offset the weak performance of the single connection scheme. Since all users associate with only one RU, the performance of all schemes is almost the same, while the MC brings a large gap between MC and SCUP by increasing the number of RUs. It is noted that the gap between the proposed method and other schemes also grows with the number of RUs.

Finally, we examine the convergence behavior of the proposed Algorithm 1, comparing the optimal value through the exhaustive search for $P^{\max} = 30$ dBm under the different number of RUs in Fig. 12. It is shown that the proposed algorithm for both problems (P1) and (P2) converges quickly, taking less than 10 iterations to reach the optimal value

within an increment, which is smaller than a given threshold $\epsilon = 10^{-4}$. As expected, based on Fig. 12(a) and Fig. 12(b), as the number of RUs increases in such a network, the eMBB throughput increases, but it does not affect the uRLLC latency remarkably. As we mentioned before that uRLLC users frequently tend to link to only one RU because of their small packet size. There is almost the same convergence speed for both cases with 3 RUs and 4 RUs. Nevertheless, the case with 4 RUs case needs a little more time for CVXPY to solve the MINCP in each step due to additional optimization variables.

VI. Conclusion

In this work, we have developed a novel intelligent TS framework in the presence of unknown dynamic traffic to meet the competing demands of uRLLC and eMBB services in beyond 5G networks based on dynamic MC. To achieve the maximum throughput for eMBB traffic while guaranteeing the minimum uRLLC latency requirement, and vice versa, we have proposed a joint intelligent traffic prediction, flow-split distribution, dynamic RAN slicing, and radio resource management scheme to schedule joint RBs and transmission power with mixed numerologies based on standardization in 5G NR. We have carried out a thorough analysis of E2E uRLLC latency. Due to the execution of the proposed problems in two different timescales, we have divided them into two long-term and short-term subproblems. To solve them, the LSTM method and SCA-based iterative algorithm have been developed to solve the formulated subproblems effectively. Thanks to LSTM, which predicts future traffic with high accuracy, the proposed method based on MC and mixed numerologies greatly improves resource utilization by adapting to dynamic traffic demands compared to benchmark schemes. One of the future works is to deploy more advanced techniques (e.g. deep reinforcement learning) to better estimate $\alpha[t]$ and $\varphi[t]$.

References

[1] 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.

[2] L. Gavrilovska, V. Rakovic, and D. Denkovski, "From cloud RAN to open RAN," *Wireless Pers. Commun.*, vol. 113, no. 3, pp. 1523–1539, Aug. 2020.

[3] C.-X. Wang, M. D. Renzo, S. Stanczak, S. Wang, and E. G. Larsson, "Artificial intelligence enabled wireless networking for 5G and beyond: Recent advances and future challenges," *IEEE Wireless Commun.*, vol. 27, no. 1, pp. 16–23, Feb. 2020.

[4] S. Niknam, H. S. Dhillon, and J. H. Reed, "Federated learning for wireless communications: Motivation, opportunities, and challenges," *IEEE Commun. Mag.*, vol. 58, no. 6, pp. 46–51, Jun. 2020.

[5] M. Dryjanski and M. Szydelko, "A unified traffic steering framework for LTE radio access network coordination," *IEEE Commun. Mag.*, vol. 54, no. 7, pp. 84–92, Jul. 2016.

[6] S. Vassilaras et al., "The algorithmic aspects of network slicing," *IEEE Commun. Mag.*, vol. 55, no. 8, pp. 112–119, Aug. 2017.

[7] M.-T. Suer, C. Thein, H. Tchouankem, and L. Wolf, "Multi-connectivity as an enabler for reliable low latency communications—An overview," *IEEE Commun. Surveys Tuts.*, vol. 22, no. 1, pp. 156–169, 1st Quart., 2020.

[8] H. Arslan, "Flexible multi-numerology systems for 5G new radio," *J. Mobile Multimedia*, vol. 14, no. 4, pp. 367–394, 2018.

[9] F. Kavehmadavani, V.-D. Nguyen, T. X. Vu, and S. Chatzinotas, "Traffic steering for eMBB and uRLLC coexistence in open radio access networks," in *Proc. IEEE Int. Conf. Commun. Workshops (ICC Workshops)*, May 2022, pp. 242–247.

[10] S. Niknam et al., "Intelligent O-RAN for beyond 5G and 6G wireless networks," 2020, *arXiv:2005.08374*.

[11] Y. Yu, X. Si, C. Hu, and J. Zhang, "A review of recurrent neural networks: LSTM cells and network architectures," *Neural Comput.*, vol. 31, no. 7, pp. 1235–1270, 2019.

[12] M. I. Kamel, L. B. Le, and A. Girard, "LTE wireless network virtualization: Dynamic slicing via flexible scheduling," in *Proc. IEEE 80th Veh. Technol. Conf. (VTC-Fall)*, Sep. 2014, pp. 1–5.

[13] G. Pocovi, K. I. Pedersen, and P. Mogensen, "Joint link adaptation and scheduling for 5G ultra-reliable low-latency communications," *IEEE Access*, vol. 6, pp. 28912–28922, 2018.

[14] A. Karimi, K. I. Pedersen, N. H. Mahmood, G. Pocovi, and P. Mogensen, "Efficient low complexity packet scheduling algorithm for mixed URLLC and eMBB traffic in 5G," in *Proc. IEEE 89th Veh. Technol. Conf. (VTC-Spring)*, Apr. 2019, pp. 1–6.

[15] Z. Wu, F. Zhao, and X. Liu, "Signal space diversity aided dynamic multiplexing for eMBB and URLLC traffics," in *Proc. 3rd IEEE Int. Conf. Comput. Commun. (ICCC)*, Dec. 2017, pp. 1396–1400.

[16] A. Anand, G. de Veciana, and S. Shakkottai, "Joint scheduling of URLLC and eMBB traffic in 5G wireless networks," *IEEE/ACM Trans. Netw.*, vol. 28, no. 2, pp. 477–490, Apr. 2020.

[17] N. Zhang, S. Zhang, S. Wu, J. Ren, J. W. Mark, and X. Shen, "Beyond coexistence: Traffic steering in LTE networks with unlicensed bands," *IEEE Wireless Commun.*, vol. 23, no. 6, pp. 40–46, Dec. 2016.

[18] P. Korrai, E. Lagunas, S. K. Sharma, S. Chatzinotas, A. Bandi, and B. Ottersten, "A RAN resource slicing mechanism for multiplexing of eMBB and URLLC services in OFDMA based 5G wireless networks," *IEEE Access*, vol. 8, pp. 45674–45688, 2020.

[19] K. Zhang, X. Xu, J. Zhang, B. Zhang, X. Tao, and Y. Zhang, "Dynamic multiconnectivity based joint scheduling of eMBB and uRLLC in 5G networks," *IEEE Syst. J.*, vol. 15, no. 1, pp. 1333–1343, Mar. 2021.

[20] A. Prasad, F. S. Moya, M. Ericson, R. Fantini, and O. Bulakci, "Enabling RAN moderation and dynamic traffic steering in 5G," in *Proc. IEEE 84th Veh. Technol. Conf. (VTC-Fall)*, Sep. 2016, pp. 1–6.

[21] L. You, Q. Liao, N. Pappas, and D. Yuan, "Resource optimization with flexible numerology and frame structure for heterogeneous services," *IEEE Commun. Lett.*, vol. 22, no. 12, pp. 2579–2582, Dec. 2018.

[22] T. T. Nguyen, V. N. Ha, and L. B. Le, "Wireless scheduling for heterogeneous services with mixed numerology in 5G wireless networks," *IEEE Commun. Lett.*, vol. 24, no. 2, pp. 410–413, Feb. 2020.

[23] P. K. Korrai, E. Lagunas, A. Bandi, S. K. Sharma, and S. Chatzinotas, "Joint power and resource block allocation for mixed-numerology-based 5G downlink under imperfect CSI," *IEEE Open J. Commun. Soc.*, vol. 1, pp. 1583–1601, 2020.

[24] L. Bonati, S. D'Oro, M. Polese, S. Basagni, and T. Melodia, "Intelligence and learning in O-RAN for data-driven NextG cellular networks," *IEEE Commun. Mag.*, vol. 59, no. 10, pp. 21–27, Oct. 2021.

[25] ORAN Alliance. (2018). *O-RAN: Towards an Open and Smart RAN*. [Online]. Available: https://www.o-ran.org/resources

[26] A. B. Kihero, M. S. J. Solaija, and H. Arslan, "Inter-numerology interference for beyond 5G," *IEEE Access*, vol. 7, pp. 146512–146523, 2019.

[27] G. Zheng, I. Krikidis, C. Masouros, S. Timotheou, D.-A. Toumpakaris, and Z. Ding, "Rethinking the role of interference in wireless networks," *IEEE Commun. Mag.*, vol. 52, no. 11, pp. 152–158, Nov. 2014.

[28] Y. Polyanskiy, H. V. Poor, and S. Verdu, "Channel coding rate in the finite blocklength regime," *IEEE Trans. Inf. Theory*, vol. 56, no. 5, pp. 2307–2359, Apr. 2010.

[29] S. Schiessl, J. Gross, and H. Al-Zubaidy, "Delay analysis for wireless fading channels with finite blocklength channel coding," in *Proc. 18th ACM Int. Conf. Modeling, Anal. Simulation Wireless Mobile Syst.*, Nov. 2015, pp. 13–22.

[30] P. J. Burke, "The output of a queuing system," *Oper. Res.*, vol. 4, no. 6, pp. 699–704, Dec. 1956.

[31] O. Alliance, *O-RAN Working Group 2 AI/ML Workflow Description and Requirements*, document ORAN-WG2. AIML. v01, 2019, vol. 1.

[32] E. Che, H. D. Tuan, and H. H. Nguyen, "Joint optimization of cooperative beamforming and relay assignment in multi-user wireless relay networks," *IEEE Trans. Wireless Commun.*, vol. 13, no. 10, pp. 5481–5495, Oct. 2014.

[33] B. R. Marks and G. P. Wright, "A general inner approximation algorithm for nonconvex mathematical programs," *Oper. Res.*, vol. 26, no. 4, pp. 681–683, Aug. 1978.
[34] A. Beck, A. Ben-Tal, and L. Tetruashvili, "A sequential parametric convex approximation method with applications to nonconvex truss topology design problems," *J. Global Optim.*, vol. 47, no. 1, pp. 29–51, May 2010.
[35] A. Ben-Tal and A. Nemirovski, *Lectures on Modern Convex Optimization*. Philadelphia, PA, USA: SIAM, 2001.

Fatemeh Kavehmadavani received the B.S. degree in communication systems from Yazd University, Iran, and the M.Sc. degree in communication systems from the Shahid Bahonar University of Kerman, Iran. She is currently pursuing the Ph.D. degree with the SIGCOM Group, Interdisciplinary Centre for Security, Reliability and Trust (SnT), University of Luxembourg. She joined SIGCOM, headed by Prof. Symeon Chatzinotas. Her research interests include wireless communications towards 6G and the IoT, ORAN, machine learning (ML), and signal processing.

Van-Dinh Nguyen (Member, IEEE) received the B.E. degree in electrical engineering from the Ho Chi Minh City University of Technology, Vietnam, in 2012, and the M.E. and Ph.D. degrees in electronic engineering from Soongsil University, Seoul, South Korea, in 2015 and 2018, respectively.

Since 2022, he has been an Assistant Professor with VinUniversity, Vietnam. He was a Research Associate with the Interdisciplinary Centre for Security, Reliability and Trust (SnT), University of Luxembourg; a Post-Doctoral Researcher and a Lecturer with Soongsil University; a Post-Doctoral Visiting Scholar with the University of Technology Sydney; and a Ph.D. Visiting Scholar with Queen's University Belfast, U.K. He has authored or coauthored in some 60 papers published in international journals and conference proceedings. His current research interests include the mathematical modeling of 5G/6G cellular networks, edge/fog computing, and AI/ML solutions for wireless communications. He received several best conference paper awards, the Exemplary Editor Award of IEEE COMMUNICATIONS LETTERS in 2019, the Exemplary Reviewer Award of IEEE TRANSACTION ON COMMUNICATIONS in 2018, and the IEEE GLOBECOM Student Travel Grant Award in 2017. He has served as a reviewer for many top-tier international journals on wireless communications and has also been a technical program committee member for several flag-ship international conferences in the related fields. He is an Editor of the IEEE SYSTEMS JOURNAL, IEEE OPEN JOURNAL OF THE COMMUNICATIONS SOCIETY, and IEEE COMMUNICATIONS LETTERS.

Thang X. Vu (Senior Member, IEEE) received the B.S. degree in electronics and telecommunications engineering from the University of Engineering and Technology in 2007, the M.Sc. degree in electronics and telecommunications engineering from Vietnam National University, Hanoi, in 2009, and the Ph.D. degree in electrical engineering from Université Paris-Sud, France, in 2014. In 2010, he received the Allocation de Recherche Fellowship to study Ph.D. degree in France. From July 2014 to January 2016, he was a Post-Doctoral Researcher with the Singapore University of Technology and Design (SUTD), Singapore. Currently, he is a Research Scientist with the Interdisciplinary Centre for Security, Reliability and Trust (SnT), University of Luxembourg. His research interests include the field of wireless communications, with particular interests of the applications of machine learning and optimization theory in design, analyze, and optimize multilayer 6G networks. He was a recipient of the 2019 SigTelCom Best Paper Award. He has been served as an Associate Editor for the IEEE COMMUNICATIONS LETTERS since 2022.

Symeon Chatzinotas (Fellow, IEEE) is currently a Full Professor/Chief Scientist I of satellite communications and the Head of the SIGCOM Research Group, Interdisciplinary Centre for Security, Reliability and Trust (SnT), University of Luxembourg, Esch-sur-Alzette, Luxembourg. He coordinates research activities in communications and networking, acting as a PI in more than 20 projects, and is the main representative for 3GPP, ETSI, and DVB. In the past, he was a Visiting Professor with the University of Parma, Parma, Italy, where he is lecturing on 5G wireless networks. He was involved in numerous research and development projects for NCSR Demokritos, CERTH Hellas and CCSR, and the University of Surrey, Guildford, U.K. He has coauthored more than 450 technical papers in refereed international journals, conferences, and scientific books. He was a co-recipient of the 2014 IEEE Distinguished Contributions to Satellite Communications Award and the Best Paper Awards at EURASIP JWCN, CROWNCOM, and ICSSC. He is currently on the editorial board of the IEEE TRANSACTIONS ON COMMUNICATIONS, IEEE OPEN JOURNAL OF VEHICULAR TECHNOLOGY, and the *International Journal of Satellite Communications and Networking*.