Resource Allocation for Radio Access Network Slicing With Delay Guarantees: A Lyapunov Optimization Method

Rong Chai and Shaowei Wang, *Senior Member, IEEE*

***Abstract*—In this paper, we present a delay-guaranteed resource allocation method for radio access network slicing based on Lyapunov optimization theory. The proposed approach can dynamically monitor network traffic and channel conditions, and jointly optimize resource blocks and power allocation so as to satisfy delay constraints of users served by each slice. Numerical results demonstrate that the performance of our proposed algorithm significantly outperforms other ones.**

***Index Terms*—Network slicing, radio access network, radio resource allocation.**

I. Introduction

5G and beyond mobile networks are envisioned to support a wide range of vertical industries, including human-centric multimedia services with high data rates and time-sensitive industrial applications. Conventional networks typically adopt one-size-fits-all architectures, offering homogeneous service treatments that fail to meet the diverse requirements of heterogeneous applications in terms of throughput, delay, and reliability. Network slicing provides a promising solution by enabling multiple isolated subnetworks, i.e., network slices, to operate on shared physical infrastructure, allowing mobile operators to deploy tailored services for specific market scenarios in a flexible and cost-efficient manner [1], [2].

Achieving slice isolation in radio access networks (RAN) remains challenging due to the coupled and non-scalable nature of radio resources, the heterogeneous requirements of RAN slices, and wireless channel variability. Recent studies have explored inter-slice resource allocation with performance guarantees. In [3], a stochastic network calculus approach is proposed to conservatively estimate the resources required to satisfy delay constraints. In [4], an online convex optimization framework is introduced to learn instant resource allocation from empirical data, achieving low packet loss while satisfying multi-slice delay constraints.

Joint optimization of spectrum and power allocation for multiple users across slices entails high complexity, due to variable traffic and the dynamic nature of wireless channels, particularly the user-specific fading across resource blocks (RBs). Adaptive resource management is required to cope with time-varying, frequency-selective channels across users and RBs, as well as the coupling between power and spectrum allocation [5].

In [6], [7], Lyapunov optimization is applied to spectrum and power allocation to improve system utility, while frequency-selective fading is simplified for the sake of analytical tractability. In [8], Q-learning is leveraged for online mode selection and resource management, where the complexity of multi-carrier power allocation is circumvented by restricting each user to a single subcarrier. In [9], a joint spectrum and power allocation algorithm is developed for vehicular networks, which accounts for frequency-selective fading across RBs and proves convergence with respect to the objective function value, while the properties of the convergence point remain uncharacterized.

In this paper, we investigate the online resource allocation for RAN slicing with delay guarantees, where Lyapunov optimization is employed to jointly allocate RBs and power, aiming to satisfy delay requirements of different slices while reducing power consumption and improving spectrum efficiency. Our proposed algorithm can dynamically adjust RB allocation and power configuration in each slot, and converges stably. Numerical results demonstrate that the proposed method ensures slice isolation and significantly reduces resource consumption while satisfying delay constraints.

II. System Model and Problem Formulation

Consider downlink transmissions in a cellular network consisting of a base station and I users from various slices. Let the set of users be denoted by $\mathcal{I} = \{1, 2, \ldots, I\}$. Time is divided into slots, each lasting ϕ seconds and indexed by $t \in \{1, 2, \cdots\}$. The amount of data arriving for user $i \in \mathcal{I}$ in slot t is denoted as $a_{i,t}$. The expected delay from data arrival to transmission completion is constrained by Φ_i seconds.

The available spectrum is divided into J RBs, each with bandwidth B, indexed by $\mathcal{J} = \{1, 2, \ldots, J\}$. Let $h_{i,j,t}$ represent the channel gain for user i on RB $j \in \mathcal{J}$ during slot t. With transmission power $p_{i,j,t}$, the rate achievable by user i on RB j is determined as

$$r_{i,j,t} = B \log_2 \left(1 + \frac{p_{i,j,t} \left|h_{i,j,t}\right|^2}{N_0 B}\right), \tag{1}$$

where N_0 denotes the power spectral density of the Gaussian noise. The achievable throughput for user i is given by

$$b_{i,t} = \phi \sum_{j \in \mathcal{J}} x_{i,j,t} r_{i,j,t}, \tag{2}$$

where $x_{i,j,t} = 1$ indicates that RB j is allocated to user i, and $x_{i,j,t} = 0$ otherwise.

Data arriving in each slot may not be immediately transmitted due to fluctuations in transmission rates, which arise from time-varying channel conditions and competition among users for limited resources. Each user is assigned a buffer to store data awaiting transmission, modeled as a first-in, first-out queue. Assuming the expected transmission time is a constant φ, the average number of slots for which data remains in the buffer should not exceed $D_i = \lfloor \frac{\Phi_i - \varphi}{\phi} \rfloor$. The queue backlog for user i in slot t is denoted by $q_{i,t}$, which is updated as

$$q_{i,t+1} = \max\left(q_{i,t} + a_{i,t} - b_{i,t}, 0\right). \tag{3}$$

RBs and power are allocated to users in each slot, represented by matrices $\boldsymbol{X}_t = [x_{i,j,t}] \in \mathbb{R}^{I \times J}$ and $\boldsymbol{P}_t = [p_{i,j,t}] \in \mathbb{R}^{I \times J}$, respectively. The normalized resource consumption $G(\boldsymbol{X}_t, \boldsymbol{P}_t)$ is defined as a weighted sum of the ratios of allocated RBs and transmission power to the available RBs and maximum allowable power, respectively, given by

$$G(\boldsymbol{X}_t, \boldsymbol{P}_t) = \frac{\alpha}{J} \sum_{i \in \mathcal{I}} \sum_{j \in \mathcal{J}} x_{i,j,t} + \frac{1 - \alpha}{P_{\max}} \sum_{i \in \mathcal{I}} \sum_{j \in \mathcal{J}} p_{i,j,t}, \tag{4}$$

where $P_{\max}$ denotes the maximum allowable power, and $\alpha \in [0, 1]$ is a weighting factor. This formulation ensures unit consistency and normalization, enabling a fair comparison between spectrum and power

Received 17 April 2025; accepted 24 May 2025. Date of publication 19 June 2025; date of current version 20 November 2025. The review of this article was coordinated by Prof. Zhu Han. *(Corresponding author: Shaowei Wang.)*

The authors are with the School of Electronic Science and Engineering, Nanjing University, Nanjing 210023, China (e-mail: rongchai@smail.nju.edu.cn; wangsw@nju.edu.cn).

Digital Object Identifier 10.1109/TVT.2025.3575850

usage while balancing their contributions to the overall system resource consumption.

We aim to design a resource allocation scheme that provides delay guarantees while minimizing average resource consumption, to satisfy slice requirements, enhance resource efficiency, and mitigate over-provisioning. Let the sequences of resource allocation decisions be denoted as $\{\boldsymbol{X}_t\}$ and $\{\boldsymbol{P}_t\}$, we formulate the following problem:

$$\begin{aligned}
\min_{\{\boldsymbol{X}_t\},\{\boldsymbol{P}_t\}} \quad & \lim_{T\to\infty}\frac{1}{T}\sum_{t=1}^{T} G(\boldsymbol{X}_t,\boldsymbol{P}_t) \\
\text{s.t.} \quad & C_1: \lim_{T\to\infty}\frac{1}{T}\sum_{t=1}^{T} b_{i,t} \geq \lim_{T\to\infty}\frac{1}{T}\sum_{t=1}^{T} a_{i,t},\ \forall i, \\
& C_2: \lim_{T\to\infty}\frac{1}{T}\sum_{t=1}^{T} q_{i,t} \leq D_i \lim_{T\to\infty}\frac{1}{T}\sum_{t=1}^{T} a_{i,t},\ \forall i, \\
& C_3: \sum_{i\in\mathcal{I}} x_{i,j,t} \leq 1,\ \forall j,t, \\
& C_4: \sum_{i\in\mathcal{I}}\sum_{j\in\mathcal{J}} p_{i,j,t} \leq P_{\max},\ \forall t, \\
& C_5: x_{i,j,t} \in \{0,1\},\ \forall i,j,t, \\
& C_6: p_{i,j,t} \geq 0,\ \forall i,j,t.
\end{aligned} \tag{5}$$

C_1 ensures queue stability by guaranteeing that the average transmission rate is no less than the arrival rate. *Littles Law* states that the average queue length equals the product of the arrival rate and the mean queuing delay. Accordingly, C_2 satisfies the delay requirement by ensuring that the average queue length does not exceed the product of the arrival rate and the queuing delay constraint. C_3 and C_5 ensure that each RB is allocated to at most one user. C_4 and C_6 ensure that the transmission power does not exceed the constraint.

III. Delay-Guaranteed RAN Resource Allocation

We introduce two virtual queues, $y_{i,t}$ and $z_{i,t}$, for each user to represent the constraints C_1 and C_2 in (5), respectively. These virtual queues are updated as follows

$$\begin{aligned}
y_{i,t+1} &= \max(a_{i,t} - b_{i,t} + y_{i,t}, 0), \\
z_{i,t+1} &= \max(q_{i,t+1} - a_{i,t}D_i + z_{i,t}, 0),
\end{aligned} \quad \forall i. \tag{6}$$

If the virtual queues satisfy $\lim_{t\to\infty}\frac{\mathbb{E}(|y_{i,t}|)}{t} = \lim_{t\to\infty}\frac{\mathbb{E}(|z_{i,t}|)}{t} = 0$, they are considered mean rate stable, ensuring that the associated constraints are satisfied [10]. Let $\mathcal{Y}_t = \{y_{i,t}\}_{i\in\mathcal{I}}$ and $\mathcal{Z}_t = \{z_{i,t}\}_{i\in\mathcal{I}}$ denote the sets of virtual queue backlogs for all users. We define Lyapunov function as $F(\mathcal{Y}_t,\mathcal{Z}_t) = \frac{1}{2}\sum_{i\in\mathcal{I}}(y_{i,t}^2 + z_{i,t}^2)$, with its temporal difference $\Delta F_t = F(\mathcal{Y}_{t+1},\mathcal{Z}_{t+1}) - F(\mathcal{Y}_t,\mathcal{Z}_t)$ referred to as Lyapunov drift. We then define a drift-plus-penalty structure as $\mu G(\boldsymbol{X}_t,\boldsymbol{P}_t) + \Delta F_t$, where $\mu > 0$ is a penalty factor controlling the trade-off between the two components. Using the inequality $\max(x,0)^2 \leq x^2$, we derive an upper bound for the drift-plus-penalty structure as follows

$$\begin{aligned}
&\mu G(\boldsymbol{X}_t,\boldsymbol{P}_t) + \Delta F_t \leq M + \mu G(\boldsymbol{X}_t,\boldsymbol{P}_t) \\
&+ \sum_{i\in\mathcal{I}} y_{i,t}(a_{i,t} - b_{i,t}) + \sum_{i\in\mathcal{I}} z_{i,t}(q_{i,t+1} - a_{i,t}D_i),
\end{aligned} \tag{7}$$

where M is a constant satisfying

$$M \geq \frac{1}{2}\sum_{i\in\mathcal{I}}\left[(a_{i,t} - b_{i,t})^2 + (q_{i,t+1} - a_{i,t}D_i)^2\right].$$

Algorithm 1: Delay-Guaranteed Resource Allocation.

Input: User set $\mathcal{I}$, RB set $\mathcal{J}$, parameter $0 < \epsilon \ll 1$, weighting factor $\alpha \in [0,1]$, penalty factor $\mu > 0$ and maximum number of iterations K.

Find any feasible initial solution $\boldsymbol{X}_t^{(0)}, \boldsymbol{P}_t^{(0)}$;
for $k = 1$ *to* K **do**
  Fix $\boldsymbol{P}_t \leftarrow \boldsymbol{P}_t^{(k-1)}$ and solve (9) to obtain $\boldsymbol{X}_t^{(k)}$;
  Fix $\boldsymbol{X}_t \leftarrow \boldsymbol{X}_t^{(k)}$ and solve (9) to obtain $\boldsymbol{P}_t^{(k)}$;
end
Initialize $\boldsymbol{X}_t^*$ as a zero matrix of size $I \times J$;
for $j = 1$ *to* J **do**
  $i \leftarrow \arg\max_{i_1\in\mathcal{I}} x_{i_1,j,t}^{(K)}$;
  if $x_{i,j,t}^{(K)} > 0.5$ **then**
    $x_{i,j,t}^* \leftarrow 1$;
  end
end
Substitute $\boldsymbol{X}_t^*$ into (9) and solve for $\boldsymbol{P}_t^*$;
return $\boldsymbol{X}_t^*, \boldsymbol{P}_t^*$;

We employ Lyapunov optimization method to solve (5) by minimizing the upper bound of the drift-plus-penalty structure [10]. The resource allocation decision in each slot is determined by solving the following subproblem:

$$\begin{aligned}
\min_{\boldsymbol{X},\boldsymbol{P}} \quad & \mu G(\boldsymbol{X},\boldsymbol{P}) + \sum_{i\in\mathcal{I}} y_i(a_i - b_i) \\
& + \sum_{i\in\mathcal{I}} z_i\left[\max(q_i + a_i - b_i, 0) - a_i D_i\right] \\
\text{s.t.} \quad & C_3, C_4, C_5, C_6 \text{ in (5)}.
\end{aligned} \tag{8}$$

For simplicity, the slot index is omitted here. Equation (8) defines an NP-hard mixed-integer programming problem, which becomes intractable as the problem size increases [5]. To address this, we first solve the continuous relaxation of (8) and subsequently round the non-integer solutions to obtain feasible results. By substituting (1) and (2) into (8) and approximating the function $\max(x,0)$ by $\frac{1}{2}(x + \sqrt{x^2+\epsilon})$, where $0 < \epsilon \ll 1$, the objective function becomes differentiable, as shown in (9) at the bottom of the next page.

Algorithm 1 outlines the delay-guaranteed resource allocation method we propose. Specifically, we employ a two-block coordinate descent method to solve (9), with its global convergence and complexity analyzed later in this section. The RB allocation matrix obtained by K iterations with block coordinate descent method is denoted as $\boldsymbol{X}^{(K)}$. If $\boldsymbol{X}^{(K)}$ is not an integer solution, it must be rounded. For each $x_{i,j}^{(K)} \in \boldsymbol{X}^{(K)}$, if $x_{i,j}^{(K)} > 0.5$ and $i = \arg\max_{i_1\in\mathcal{I}} x_{i_1,j}^{(K)}$, round $x_{i,j}^{(K)}$ to 1; otherwise, round it to 0. The rounded RB allocation matrix is denoted as $\boldsymbol{X}^*$. Finally, substitute $\boldsymbol{X}^*$ into (9) to solve for the transmission power allocation matrix $\boldsymbol{P}^*$.

Lemma 1: Consider two sequences $\{\boldsymbol{c}^{(k)}\}$ and $\{\boldsymbol{d}^{(k)}\}$ within a closed convex set $\mathcal{X} \in \mathbb{R}^n$, and a differentiable function $f: \mathcal{X} \to \mathbb{R}$. For all $k \in \mathbb{N}$, assume the sequence $\{\boldsymbol{d}^{(k)}\}$ is bounded, with $\boldsymbol{c}^{(k)} + \boldsymbol{d}^{(k)} \in \mathcal{X}$ and $\nabla f(\boldsymbol{c}^{(k)})^T\boldsymbol{d}^{(k)} < 0$. There exist $\gamma_1^{(k)}, \gamma_2^{(k)} \in (0,1)$ such that

$$f\left(\boldsymbol{c}^{(k)} + \gamma_1^{(k)}\boldsymbol{d}^{(k)}\right) \leq f(\boldsymbol{c}^{(k)}) + \gamma_2^{(k)}\nabla f(\boldsymbol{c}^{(k)})^T\boldsymbol{d}^{(k)}. \tag{10}$$

As $k \to \infty$, if $\boldsymbol{c}^{(k)} \to \overline{\boldsymbol{c}}$ and $f(\boldsymbol{c}^{(k)}) - f(\boldsymbol{c}^{(k)} + \gamma_1^{(k)} \boldsymbol{d}^{(k)}) \to 0$, then

$$\lim_{k\to\infty} \nabla f\left(\boldsymbol{c}^{(k)}\right)^T \boldsymbol{d}^{(k)} = 0. \tag{11}$$

For brevity, the details of Lemma 1 can be found in [11]. To simplify notation, we vectorize $\boldsymbol{X}$ and $\boldsymbol{P}$ as $\boldsymbol{x}$ and $\boldsymbol{p}$, respectively, with their corresponding feasible regions denoted as $\mathcal{X}$ and $\mathcal{P}$. Let f represent the objective function of (9). The following theorem is presented to analyze the convergence of the two-block coordinate descent method.

Theorem 1: Starting from any feasible solution, the two-block coordinate descent method for problem (9) converges to a critical point where f does not decrease in any feasible direction. Specifically, the sequences $\{\boldsymbol{x}^{(k)}\}$ and $\{\boldsymbol{p}^{(k)}\}$ must have limit points $\overline{\boldsymbol{x}} \in \mathcal{X}$ and $\overline{\boldsymbol{p}} \in \mathcal{P}$, such that

$$\nabla_{\boldsymbol{x}} f\left(\overline{\boldsymbol{x}}, \overline{\boldsymbol{p}}\right)^T \left(\boldsymbol{x}' - \overline{\boldsymbol{x}}\right) \geq 0, \ \forall \boldsymbol{x}' \in \mathcal{X},$$
$$\nabla_{\boldsymbol{p}} f\left(\overline{\boldsymbol{x}}, \overline{\boldsymbol{p}}\right)^T \left(\boldsymbol{p}' - \overline{\boldsymbol{p}}\right) \geq 0, \ \forall \boldsymbol{p}' \in \mathcal{P}. \tag{12}$$

Proof: Please refer to Appendix. ■

Algorithm 1 employs a two-block coordinate descent method, alternately fixing $\boldsymbol{X}$ and $\boldsymbol{P}$ to convert problem (9) into a convex subproblem, which is subsequently solved via the interior-point method. The iteration complexity of the interior-point method is given by $\mathcal{O}(\sqrt{C}\log(\frac{C}{\rho\delta}))$, where C is the number of constraints, satisfying $C \leq IJ + J$, ρ is the accuracy parameter of the initial point, and δ is the stopping criterion [12]. Accordingly, the computational complexity of Algorithm 1 is $\mathcal{O}(K\sqrt{IJ}\log(\frac{IJ}{\rho\delta}))$.

IV. Numerical Results

Consider an urban cell with a radius of 150 m, where the base station is centrally located and operates at a center frequency of 2.6 GHz. The parameters are set as: $\phi = \varphi = 1$ ms, $B = 180$ kHz, $P_{\max} = 40$ dBm, and $J = 25$. Let d_i represent the distance between the base station and user i. Path loss is calculated by $40.74\,\text{dB} + 10n\lg(\frac{d_i}{d_r}) + \xi_\sigma$, where the reference distance $d_r = 1$ m, the path loss exponent $n = 3.19$, and ξ_σ is a Gaussian random variable with zero mean and a standard deviation of $\sigma = 8.2$ dB, modeling shadow fading [13]. Without loss of generality, multipath effects are modeled using a Rayleigh distribution with unit second moment.

The network supports two slices, with arrival traffic modeled as Poisson processes and corresponding delay constraints of 5 ms and 10 ms, respectively. We assume an equal number of users per slice and a uniform spatial distribution of users within the cell. For our proposed method, the parameters are set as $\alpha = 0.5$, $\mu = 1000$, $\epsilon = 10^{-6}$, and $K = 10$. All simulations were conducted over 10,000 slots. Slice isolation is evaluated by assessing the impact of traffic variations in one slice on the other. Specifically, we fix the arrival rate per user in slice 1, vary the arrival rate per user in slice 2, and analyze the resulting changes in the mean delay and average achievable rate for each user in both slices.

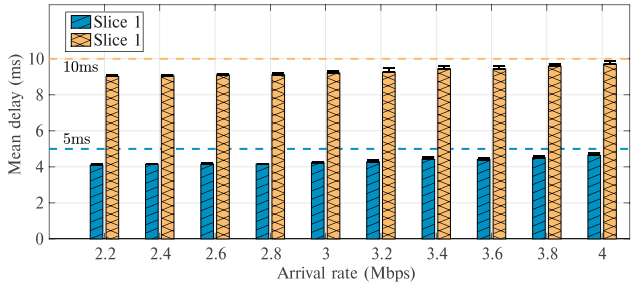

Fig. 1. Mean delay with different arrival rates.

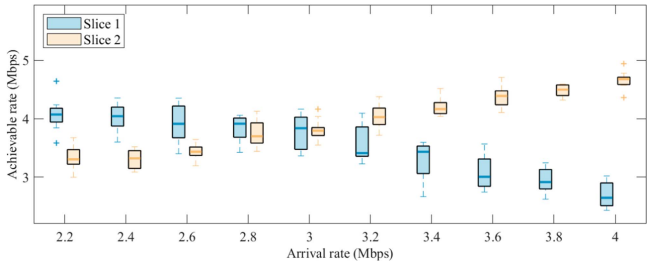

Fig. 2. Average achievable rate with different arrival rates.

Figs. 1 and 2 illustrate the variations in mean delay and average achievable rate of the two slices, respectively. Each slice contains 10 users, with the arrival rate of users in slice 1 fixed at 1 Mbps, while the arrival rate of users in slice 2 increases from 2.2 Mbps to 4 Mbps. As shown in Fig. 1, the mean delay of users in slice 1 exhibits a slight upward trend while consistently satisfying the delay constraint. For slice 2, as the arrival rate increases, the mean delay also rises, yet it remains within the delay constraint.

In Fig. 2, the average achievable rate of users in slice 2 increases with the arrival rate, effectively accommodating the growing traffic demand. Meanwhile, the average achievable rate of users in slice 1 slightly decreases but remains around 3 Mbps, which is significantly higher than their arrival rates, ensuring compliance with both delay and queue stability constraints. These numerical results confirm that the proposed algorithm effectively maintains slice isolation while satisfying the throughput and delay requirements of users in each slice.

We compare the proposed method with three alternatives: equal resource allocation (ERA), Lyapunov optimization with constant power (LOCP), and Q-learning. ERA equally allocates RBs and power among users, with each user applying the water-filling algorithm to adjust power across RBs. LOCP applies Lyapunov optimization to minimize the drift-plus-penalty upper bound in each slot by optimizing RB allocation, with all users transmitting at constant power on their assigned

$$\min_{\boldsymbol{X},\boldsymbol{P}} \ \mu G(\boldsymbol{X},\boldsymbol{P}) + \sum_{i\in\mathcal{I}} y_i \left[a_i - B\phi \sum_{j\in\mathcal{J}} x_{i,j} \log_2\left(1 + \frac{p_{i,j}\left|h_{i,j}\right|^2}{N_0 B}\right)\right] + \sum_{i\in\mathcal{I}} \frac{z_i}{2} \sqrt{\left[q_i + a_i - B\phi \sum_{j\in\mathcal{J}} x_{i,j} \log_2\left(1 + \frac{p_{i,j}\left|h_{i,j}\right|^2}{N_0 B}\right)\right]^2 + \epsilon}$$
$$+ \sum_{i\in\mathcal{I}} \frac{z_i}{2} \left[q_i + a_i - B\phi \sum_{j\in\mathcal{J}} x_{i,j} \log_2\left(1 + \frac{p_{i,j}\left|h_{i,j}\right|^2}{N_0 B}\right)\right] - \sum_{i\in\mathcal{I}} z_i a_i D_i$$
$$\text{s.t.} \quad C_3, C_4, C_6 \text{ in (5)},$$
$$C_7 : 0 \leq x_{i,j} \leq 1, \ \forall i, j. \tag{9}$$

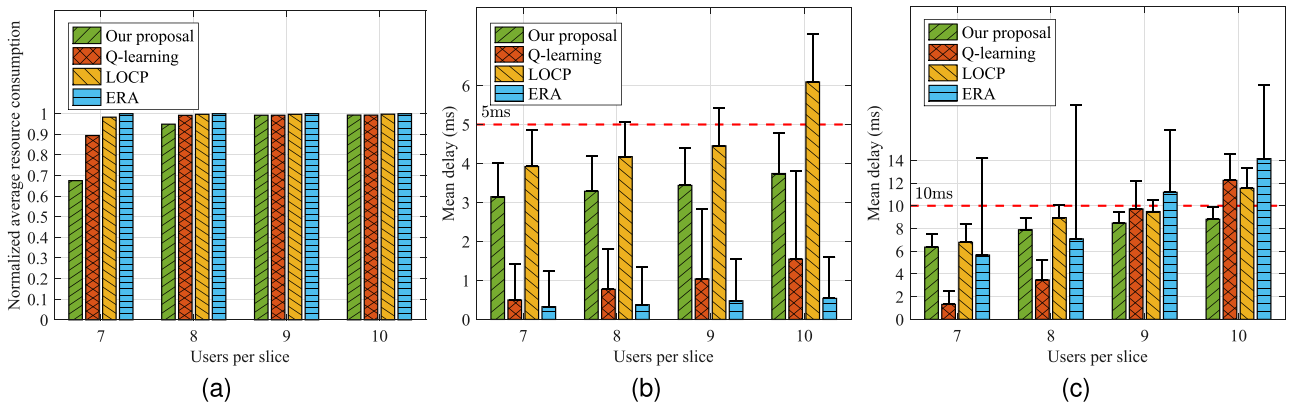

Fig. 3. (a) Resource consumption, (b) mean delay in slice 1, and (c) mean delay in slice 2 with different numbers of users per slice.

RBs [7]. We can employ the Q-learning approach to solve subproblem (8) as discussed in [8].

Fig. 3(a) illustrates the normalized average resource consumption of various methods with different numbers of users per slice. The arrival rates of users in slice 1 and slice 2 are set to 1 Mbps and 4 Mbps, respectively. The resource consumption of the proposed method increases slightly as the number of users grows. Compared to the alternatives, the proposed method dynamically evaluates queue congestion and channel conditions, jointly optimizing both power and RB allocation, which significantly reduces system power consumption and enhances spectral efficiency.

Fig. 3(b) and (c) depict the mean delay of the two slices for different numbers of users per slice. When the number of users reaches 8, LOCP fails to satisfy the stringent delay constraint of slice 1. As the number of users increases to 9, all the compared methods fail to meet the requirements of high data rate applications in slice 2, resulting in mean delays that significantly exceed the constraints. For the proposed method, the mean delay increases slightly with the number of users but consistently remains within the constraint, demonstrating that the proposed algorithm can detect data congestion in each slot and dynamically adjust resource allocation both across slices and among users within slices, ensuring robust performance. Error bar analysis reveals that the proposed method attains the smallest discrepancy between the maximum and mean delays within each slice, highlighting its effectiveness in ensuring user fairness. Compared to LOCP, the proposed method optimizes resource allocation across both the spectrum and power domains, providing more stable delay guarantees as the number of users increases.

V. Conclusion

In this paper, we developed a delay-guaranteed resource allocation scheme for RAN slicing in mobile networks, where we introduce a set of parallel queues to model network congestion and regulates average backlogs, and leverage Lyapunov optimization theory to design a drift-plus-penalty scheme and jointly allocate RBs and power across slices by minimizing the upper bound in each slot. An efficient rounding algorithm with global convergence is proposed to solve this minimization problem and make resource allocation decisions. Numerical results demonstrate that the proposed method outperforms others by consistently meeting delay constraints for heterogeneous slices and significantly reducing resource consumption. These findings provide valuable insights for guiding resource allocation in RAN slicing.

Appendix
Proof of Theorem 1

We first prove that by fixing either $\boldsymbol{x}$ or $\boldsymbol{p}$, problem (9) is convex with respect to the other variable. The feasible region of (9) is convex. To establish the convexity of $f(\boldsymbol{x},\boldsymbol{p})$ with respect to one variable when the other is fixed, we consider the bivariate function $g(x,p)=\Gamma - x\log_2(1+\gamma p)$, where $\Gamma\in\mathbb{R}$ and $\gamma>0$. It is evident that the function $g(x,p)$ is convex with respect to one variable when the other is fixed. Consider the function $h(w)=w+\sqrt{w^2+\epsilon}$. It is straightforward to verify that $h(w)$ is strictly increasing and convex for $w\in\mathbb{R}$ and $\epsilon>0$. The composite function $h(g(x,p))$ is therefore also convex with respect to one variable when the other is fixed. Since the objective function $f(\boldsymbol{x},\boldsymbol{p})$ is a sum of $h(g(x,p))$, $g(x,p)$, and affine functions, it follows that $f(\boldsymbol{x},\boldsymbol{p})$ is convex with respect to one variable when the other is fixed. Fixing $\boldsymbol{x}$ or $\boldsymbol{p}$ transforms (9) into a convex problem that can be efficiently and accurately solved using the interior-point method, yielding

$$\begin{aligned}\boldsymbol{x}^{(k)} &\in \underset{\boldsymbol{x}\in\mathcal{X}}{\operatorname{argmin}} f\left(\boldsymbol{x},\boldsymbol{p}^{(k-1)}\right),\\ \boldsymbol{p}^{(k)} &\in \underset{\boldsymbol{p}\in\mathcal{P}}{\operatorname{argmin}} f\left(\boldsymbol{x}^{(k)},\boldsymbol{p}\right).\end{aligned}\tag{13}$$

From the optimality of $\boldsymbol{x}^{(k)}$, it follows that

$$\nabla_{\boldsymbol{x}} f\left(\boldsymbol{x}^{(k)},\boldsymbol{p}^{(k-1)}\right)^T\left(\boldsymbol{x}'-\boldsymbol{x}^{(k)}\right)\geq 0,\ \forall \boldsymbol{x}'\in\mathcal{X}.\tag{14}$$

According to *Bolzano-Weierstrass theorem*, since $\mathcal{X}$ and $\mathcal{P}$ are both bounded and closed, the sequence $\{(\boldsymbol{x}^{(k)},\boldsymbol{p}^{(k-1)})\}$ has a limit point $(\overline{\boldsymbol{x}},\overline{\boldsymbol{p}})\in\mathcal{X}\times\mathcal{P}$. There exists a subset $\mathcal{K}_1\subseteq\mathbb{N}$ such that the sequence $\{(\boldsymbol{x}^{(k)},\boldsymbol{p}^{(k-1)})\}_{k\in\mathcal{K}_1}$ converges to $(\overline{\boldsymbol{x}},\overline{\boldsymbol{p}})$. Since f is continuous, the sequence $\{f(\boldsymbol{x}^{(k)},\boldsymbol{p}^{(k-1)})\}_{k\in\mathcal{K}_1}$ converges to $f(\overline{\boldsymbol{x}},\overline{\boldsymbol{p}})$. Given that $f(\boldsymbol{x}^{(k+1)},\boldsymbol{p}^{(k)})\leq f(\boldsymbol{x}^{(k)},\boldsymbol{p}^{(k-1)})$, the sequence $\{f(\boldsymbol{x}^{(k)},\boldsymbol{p}^{(k-1)})\}$ is non-increasing and bounded below, and thus converges to $f(\overline{\boldsymbol{x}},\overline{\boldsymbol{p}})$. Noting that $f(\boldsymbol{x}^{(k+1)},\boldsymbol{p}^{(k)})\leq f(\boldsymbol{x}^{(k)},\boldsymbol{p}^{(k)})\leq f(\boldsymbol{x}^{(k)},\boldsymbol{p}^{(k-1)})$, the sequence $\{f(\boldsymbol{x}^{(k)},\boldsymbol{p}^{(k)})\}$ also converges to $f(\overline{\boldsymbol{x}},\overline{\boldsymbol{p}})$. By the continuity of $\nabla_{\boldsymbol{x}} f$ and in conjunction with (14), it follows that

$$\nabla_{\boldsymbol{x}} f\left(\overline{\boldsymbol{x}},\overline{\boldsymbol{p}}\right)^T\left(\boldsymbol{x}'-\overline{\boldsymbol{x}}\right)\geq 0,\ \forall \boldsymbol{x}'\in\mathcal{X}.\tag{15}$$

Reasoning by contradiction, we assume that there exists $\boldsymbol{p}'\in\mathcal{P}$ such that

$$\nabla_{\boldsymbol{p}} f\left(\overline{\boldsymbol{x}},\overline{\boldsymbol{p}}\right)^T\left(\boldsymbol{p}'-\overline{\boldsymbol{p}}\right)<0.\tag{16}$$

Let $\boldsymbol{d}^{(k)}=\boldsymbol{p}'-\boldsymbol{p}^{(k-1)}$, where the sequence $\{\boldsymbol{d}^{(k)}\}$ is bounded. By the

continuity of $\nabla_{\boldsymbol{p}} f$, there exists a subset $\mathcal{K}_2 \subseteq \mathcal{K}_1$ such that

$$\nabla_{\boldsymbol{p}} f\left(\boldsymbol{x}^{(k)}, \boldsymbol{p}^{(k-1)}\right)^T \boldsymbol{d}^{(k)} < 0, \ \forall k \in \mathcal{K}_2. \tag{17}$$

By Lemma 1, for all $k \in \mathcal{K}_2$, there exist $\gamma_1^{(k)}, \gamma_2^{(k)} \in (0,1)$ such that

$$f\left(\boldsymbol{x}^{(k)}, \boldsymbol{p}^{(k-1)} + \gamma_1^{(k)} \boldsymbol{d}^{(k)}\right) \leq f\left(\boldsymbol{x}^{(k)}, \boldsymbol{p}^{(k-1)}\right) + \gamma_2^{(k)} \nabla_{\boldsymbol{p}} f\left(\boldsymbol{x}^{(k)}, \boldsymbol{p}^{(k-1)}\right)^T \boldsymbol{d}^{(k)} \leq f\left(\boldsymbol{x}^{(k)}, \boldsymbol{p}^{(k-1)}\right). \tag{18}$$

Since $\mathcal{P}$ is convex, $\boldsymbol{p}^{(k-1)} \in \mathcal{P}$, $\boldsymbol{p}^{(k-1)} + \boldsymbol{d}^{(k)} \in \mathcal{P}$, and $\gamma_1^{(k)} \in (0,1)$, it follows that $\boldsymbol{p}^{(k-1)} + \gamma_1^{(k)} \boldsymbol{d}^{(k)} \in \mathcal{P}$. From (13), we obtain $f(\boldsymbol{x}^{(k)}, \boldsymbol{p}^{(k)}) \leq f(\boldsymbol{x}^{(k)}, \boldsymbol{p}^{(k-1)} + \gamma_1^{(k)} \boldsymbol{d}^{(k)}) \leq f(\boldsymbol{x}^{(k)}, \boldsymbol{p}^{(k-1)})$. As both $\{f(\boldsymbol{x}^{(k)}, \boldsymbol{p}^{(k)})\}$ and $\{f(\boldsymbol{x}^{(k)}, \boldsymbol{p}^{(k-1)})\}$ converge to $f(\overline{\boldsymbol{x}}, \overline{\boldsymbol{p}})$, it follows that

$$\lim_{k \to \infty, k \in \mathcal{K}_2} f\left(\boldsymbol{x}^{(k)}, \boldsymbol{p}^{(k)}\right) - f\left(\boldsymbol{x}^{(k)}, \boldsymbol{p}^{(k-1)} + \gamma_1^{(k)} \boldsymbol{d}^{(k)}\right) = 0. \tag{19}$$

By Lemma 1, we obtain $\nabla_{\boldsymbol{p}} f(\overline{\boldsymbol{x}}, \overline{\boldsymbol{p}})^T (\boldsymbol{p}' - \overline{\boldsymbol{p}}) = 0$, which contradicts (16). Therefore,

$$\nabla_{\boldsymbol{p}} f\left(\overline{\boldsymbol{x}}, \overline{\boldsymbol{p}}\right)^T \left(\boldsymbol{p}' - \overline{\boldsymbol{p}}\right) \geq 0, \ \forall \boldsymbol{p}' \in \mathcal{P}. \tag{20}$$

REFERENCES

[1] T. Wang and S. Wang, "Inter-slice radio resource allocation: An online convex optimization approach," *IEEE Wireless Commun.*, vol. 28, no. 5, pp. 171–177, Oct. 2021.

[2] A. Banchs, D. M. Gutierrez-Estevez, M. Fuentes, M. Boldi, and S. Provvedi, "A 5G mobile network architecture to support vertical industries," *IEEE Commun. Mag.*, vol. 57, no. 12, pp. 38–44, Dec. 2019.

[3] J. Zhu and S. Wang, "QoS-guaranteed resource allocation in mobile communications: A stochastic network calculus approach," *IEEE/ACM Trans. Netw.*, vol. 32, no. 6, pp. 5159–5171, Dec. 2024.

[4] T. Wang and S. Wang, "Online convex optimization for efficient and robust inter-slice radio resource management," *IEEE Trans. Commun.*, vol. 69, no. 9, pp. 6050–6062, Sep. 2021.

[5] Y. Liu and Y. Dai, "On the complexity of joint subcarrier and power allocation for multi-user OFDMA systems," *IEEE Trans. Signal Process.*, vol. 62, no. 3, pp. 583–596, Feb. 2014.

[6] G. Zhou, L. Zhao, K. Liang, G. Zheng, and L. Hanzo, "Utility analysis of radio access network slicing," *IEEE Trans. Veh. Technol.*, vol. 69, no. 1, pp. 1163–1167, Jan. 2020.

[7] A. Papa, M. Klugel, L. Goratti, T. Rasheed, and W. Kellerer, "Optimizing dynamic RAN slicing in programmable 5G networks," in *Proc. IEEE Int. Conf. Commun.*, Shanghai, China, May 2019, pp. 1–7.

[8] H. Xiang, M. Peng, Y. Sun, and S. Yan, "Mode selection and resource allocation in sliced fog radio access networks: A reinforcement learning approach," *IEEE Trans. Veh. Technol.*, vol. 69, no. 4, pp. 4271–4284, Apr. 2020.

[9] Y. Chen, Y. Wang, M. Liu, J. Zhang, and L. Jiao, "Network slicing enabled resource management for service-oriented ultra-reliable and low-latency vehicular networks," *IEEE Trans. Veh. Technol.*, vol. 69, no. 7, pp. 7847–7862, Jul. 2020.

[10] M. J. Neely, S. T. Rager, and T. F. La Porta, "Max weight learning algorithms for scheduling in unknown environments," *IEEE Trans. Autom. Control*, vol. 57, no. 5, pp. 1179–1191, May 2012.

[11] L. Grippo and M. Sciandrone, "On the convergence of the block nonlinear Gauss–Seidel method under convex constraints," *Oper. Res. Lett.*, vol. 26, no. 3, pp. 127–136, Apr. 2000.

[12] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge, U.K.: Cambridge Univ. Press, 2004.

[13] K. Haneda et al., "5G 3GPP-like channel models for outdoor urban microcellular and macrocellular environments," in *Proc. IEEE 83rd Veh. Technol. Conf.*, Nanjing, China, May 2016, pp. 1–7.