Analytical Modeling and Improvement of Interference-Coupled RAN Slicing

Seyed Ali Hashemian and Farid Ashtiani, Senior Member, IEEE

Abstract—Network slicing, a key component of 5G, enables simultaneously running incompatible service types on a common infrastructure. Inter-slice isolation, as a key requirement of slicing, ensures that slice activity, i.e., containing a flow under transmission, does not affect the activity of other slices. Isolation in radio access network (RAN) slices is challenging due to interaction between slices. In fact, due to direct or indirect overlap among frequency channels in RAN slices, interference is inevitable, leading to interaction. In this paper, we propose an analytical model to analyze interference-coupled multi-cell RAN slicing where the interaction among slices results in dynamic behavior of slices. To this end, we map our scenario onto a suitable state-dependent queueing network, propose an iterative algorithm to obtain approximately the network steady-state probability distribution, and derive conventional QoS metrics (average delay and throughput). To quantify isolation, we define some new key performance indicators (KPIs) that show how changes in interfering slices affect the QoS metrics. Finally, we propose an interference-aware slice channel allocation policy that significantly reduces overlapping frequency channels. Numerical results demonstrate the accuracy of our analysis and the efficacy of the proposed policy in improving isolation-based KPIs compared to some other allocation policies.

Index Terms—Interference-aware channel allocation, interference-coupled RAN, isolation, queueing network, RAN slicing, state-dependent queues.

I. INTRODUCTION

The fifth generation of mobile networks (5G), which is now widely available in major urban areas, is designed to address a wide range of needs of vertical industries. 5G technology offers various service types, including enhanced mobile broadband (eMBB), ultra-reliable low latency communication (URLLC), and massive machine-type communications (mMTC). Each type requires a highly different quality of service (QoS) from the others [1]. It is predicted that, by the end of 2028, the number of worldwide 5G mobile subscriptions will pass five billion [2]. The conventional one-size-fits-all network structure will not be enough to cope with the very different traffic demands and diverse types of services. The desired network architecture needs to be more flexible and adaptive.

Network slicing, in which the network is partitioned into separate slices, is a key component of 5G architecture that helps the flexibility and adaptability of the network. Each slice is configured to serve a specific type of service or users. All these slices are simultaneously running on a common underlying infrastructure. There are different techniques to realize network slicing. Employing radio hypervisors to abstract the existing physical resources as virtual resources or leveraging the 5G NR bandwidth parts (BWP) introduced by 3GPP [3], [4] are two typical techniques.

An end-to-end (E2E) network slice comprises components, including RAN, core network (CN) and transport network (TN) slice subnets. A network slice subnet is a group of network functions (including their corresponding resources) that can be managed independent of the E2E network slice [5]. Compared to other subnets, the RAN slice subnet has higher complexity due to difficulties in the segregation of radio resources [6]. In RAN slicing, spectrum resources are scarce and need to be managed in an elastic, customizable, and efficient way to satisfy all users’ requirements [7]. Similar to 4G systems, frequency resources in 5G RAN will be segmented into physical resource blocks (PRBs), where each PRB corresponds to the minimum allocation unit. However, 5G systems are more flexible, as there can be multiple numerologies, i.e., multiple types of spacing for subcarriers [8]. In a three-layer business model, the infrastructure provider (InP) owns the infrastructure equipment and spectrum resources, and provides infrastructure-as-a-service (IaaS) for tenants. Mobile virtual network operators (MVNOs) lease the network resources, create slices, and then assign the slices to service providers (SPs). Finally, SPs provide services for their end users [6].

Isolation is a required property of network slicing that ensures the performance, e.g., signal and processing quality as well as security, of each tenant do not alter while different tenants use their network slices. Isolation can be done at different levels of network slicing, and depending on the slice definition, inter-slice and intra-slice isolations should be considered [7]. The wireless nature of RAN makes effective and isolated slicing a difficult task, as inter-slice interference may be inevitable. In this respect, two slices at two different cells may have some frequency overlap. Moreover, two slices may have no frequency overlap but, due to partial overlap with a third slice, indirectly interact with each other. Since for a wireless link, the capacity is a function of signal-to-interference-plus-noise ratio (SINR) and allocated...
bandwidth, changes in interference level or location of receivers affect SINR and, consequently, the link capacity [9]. Thus, slice isolation in the RAN domain is highly challenging, especially when slices serve different types of services with heterogeneous and even conflicting QoS requirements, and are managed by different entities (e.g., two MVNOs), and user traffic is dynamic. The modeling and analysis of slicing in dynamic conditions, as well as improving the inter-slice isolation, are the main focus of this paper.

A. Related Works

RAN slicing is a broad area of research, and many problems can be defined there. Depending on the problem, isolation can be a concern at different levels, including physical resource scheduling (to control the SINR value) and admission control (to preserve the QoS of existing users) [10]. Designing a slicing configuration includes determining whether an InP accepts a request for a slice formation and how MVNOs rent resources and form slices. Then, there is the traffic assignment problem, which includes determining which slice should be assigned to serve user traffic from a particular SP and how much traffic should be forwarded to that slice. Finally, the intra-slice resource allocation problem includes how MVNOs allocate the slice resources to the users.

Some works studied general resource allocation in network slicing as an extension to resource allocation in conventional networks, assuming that inter-slice interference has been handled in other layers. Therefore, the slice isolation concept manifests itself in the form of satisfying the slice QoS. In [11], assuming no inter-cell interference, the authors considered a two-level scheduler. The first level periodically allocates PRBs to slices, and the second level allocates power and PRBs to users to satisfy their QoS. Then, they optimized these allocations using machine learning techniques. The authors in [12] considered a single-cell scenario and offered two algorithms that allocate PRBs to slices and their users using proportional fairness while keeping slices isolated by limiting the total number of allocated PRBs. Similarly, in [13], the authors considered a single-cell scenario and designed a two-level PRB allocation optimization problem to satisfy QoS requirements for different types of slices.

Some other works considered the inter-slice interference isolation when allocating PRBs. Usually, the slice PRB allocation, and therefore, the isolation management, is done by the InP, and then MVNOs handle the user PRB allocation. In particular, [14], [15], [16] studied physical resource scheduling for RAN slicing in multi-cell scenarios, taking into account two levels of scheduling. In the first level of scheduling, and to satisfy the service level agreement (SLA), InP allocates PRBs to MVNO slices, maximizing the number of interfering PRBs allocated to the same MVNO. Then, in the second level of scheduling, MVNO can use conventional 5G coherent and coordinated transmission techniques to mitigate the inter-cell interference. In [17], the authors considered a multi-cell RAN slicing scenario with mobile edge computing (MEC). Then, by deriving the downloading delay and embedding the isolation into an optimization problem, optimized the spectrum, power, and computational resource allocation such that slice delay requirements are satisfied. The problem with such studies is that, since the InP does the slice PRB allocation, the role of the InP is prominent, and the control of MVNOs over their network is limited. This contradicts the essence of network slicing, which gives virtual operators enough freedom to maximize their benefits. Moreover, in order to allocate the required resources, the InP needs to predict the users’ activity beforehand. This needs extra computational resources in addition to user data from virtual networks. However, such data is private and the MVNOs may be unwilling to share it because they are in competition and their goal is to maximize their revenue. On the other hand, MVNOs might lease the resources from multiple InPs. Besides, the available resources are usually dependent on factors like the network and business conditions, the price offered by MVNOs, and the available bandwidth. In other words, the slice resource allocation process by the InP has a random nature, and like a physical network operator, it is the MVNO’s job to manage the leased resources to satisfy QoS and the inter-slice isolation. Stochastic approaches, including queueing theory, can provide each MVNO with some performance indicators to estimate other MVNOs state without the need for detailed information. The authors in [18], [19], [20], [21] considered MVNOs to have a more prominent role. In these works, first, the InPs sell their bandwidth resources to MVNOs. Subsequently, MVNOs preside over allocating these resources to serve users. However, in these works, the varying inter-slice interference has not been a concern, as it is either neglected or considered as a constant parameter. This issue matters because, in reality, the amount of interference that a slice experiences depends on the activity period of interfering slices, which changes dynamically.

Queueing theory has been used in a few works to observe slices from a stochastic point of view and obtain values like delay and throughput for slice customers. In particular, the authors in [22] modeled each slice as a queue with batch arrivals and batch departures, where PRBs allocated to each slice are the customers. Then the steady-state probability distribution is obtained using an embedded Markov chain. The authors in [23] achieved an optimal PRB allocation scheme that maximizes the network spectrum utilization. In [24], the authors considered a single-cell uplink scenario with two slices corresponding to two types of services. By considering a discrete-time queue for each user with packets as customers and transmission rate as the service rate, they attempted to optimize the allocated bandwidth for each slice in addition to the power and QoS of slice users such that queues stay stable and user delay doesn’t exceed a maximum value. The authors in [25], [26] studied the slice formation problem with an admission control policy and offered a greedy algorithm that helps the InP decide whether to accept requests to form a new slice or not. For each type of slice, a queue for waiting requests and a queue for accepted requests are considered so that the admission control rate affects the state of the queues. In [27], the authors considered the user admission control problem in a network with two eMBB and URLLC slices with linked arrival rates. They modeled each slice as a queue where frequency channels are the servers. Then, they tried to find an admission control policy that minimized
the eMBB customer blocking probability while preserving the URLLC QoS.

In some RAN slicing scenarios, the spectrum resources are sliced and controlled by one or several operators on multiple cells. In cases where frequency overlap occurs between slices, employing methods analogous to those used to mitigate the well-known inter-cell interference can be beneficial in addressing the challenge of inter-slice isolation. The authors in [28], [29] focused on the coupled nature of a multi-cell network caused by inter-cell interference. That is, the interference value is a function of activity in all cells, therefore, cannot be analyzed separately for a single cell. The flow traffic model has been utilized to simplify the long-term queueing theory analysis of system performance in the time domain as it looks at user activities from a large-timescale perspective (i.e., not at the packet level) and in a continuous-time manner. In [28], the authors modeled each cell as a queue with a round robin scheduler, and therefore, all spectrum resources are either inactive or busy serving data flows. Considering such a model for each cell under investigation simplifies the problem into a two-dimensional Markov chain, as each neighboring cell either has customers and causes interference or is empty and causes no interference. Next, they offered two approximate methods, namely the state aggregation and the time-averaged interference, to solve the Markov chain. Such modeling is not applicable to a network slicing scenario, as it contradicts the fact that MVNOs only have control over their own slices and not the whole spectrum. Besides, in real systems, BSs only allocate the required amount of resources (bandwidth) to each user and keep the rest for other users to increase fairness and multiplexing gain, so the level of interference is dependent on the number of users and does not have only two states.

B. Contributions

In this paper, our goal is to study the dynamic behavior of network slices under the coupled-interference environment in a multi-cell network. To this end, we consider a network slicing scenario including an InP, multiple MVNOs, and multiple SPs and investigate the effect of user activity in slices defined in neighboring cells on the performance of a particular slice assigned to a specific MVNO. We employ an approach based on state-dependent queues to obtain the network steady-state probability distribution, which can then be further used to derive delay and throughput for each slice and the whole network. Rather than randomly assigning slice channels to the customers, we then discuss how MVNOs can cooperate and follow an interference-aware channel allocation policy to control the inter-slice interference and improve the slice isolation and QoS. In summary, our main contributions are

- Formulating the inter-slice isolation problem in a multi-cell RAN slicing using a coupled-interference point of view.
- Modeling the coupled slices as a network of state-dependent queues and offering an iterative algorithm to analyze the queues and extract the network steady-state probability distribution.
- Defining new slice and network KPIs to evaluate QoS and isolation level.

- Proposing and analyzing an interference-aware channel allocation policy that requires cooperation among MVNOs.

The rest of the paper is organized as follows. In Section II, we present the system model as well as our assumptions, describe the slicing configuration, and formulate the isolation problem under coupled interference among slices. In Section III, we model the problem using a state-dependent approach based on queueing theory and devise an iterative algorithm to obtain the network steady-state probabilities. Furthermore, in Section IV, we introduce new KPIs to evaluate the isolation level for slice users. In Section V, we propose an interference-aware channel allocation policy to reduce the interference and increase the isolation by cooperation among MVNOs, and then analyze the proposed policy using the same state-dependent queueing network model. In Section VI, in order to validate our analytical model and investigate our proposed policy, we carry out computer simulations and compare the numerical results. Finally, in Section VII, we provide concluding remarks.

II. SYSTEM MODEL

A. Cellular Network

As shown in Fig. 1, we consider a RAN consisting of a set $B = \{1, \ldots, B\}$ of base stations (BS) where BS $b \in B$ has a transmission power equal to $P_{b,\text{BS}}$ and is located at the center of its corresponding cell in a way that cell $b$ covers a region $L_b \subset \...
with no overlap. Thus, the total network coverage is $\mathcal{L} = \bigcup_{b \in B} \mathcal{L}_b$. Although some users can be served by multiple BSs, as in coordinated multi-point (CoMP) standard, we focus on a scenario in which each user is served by only one BS selected based on some criteria. Here, the definition of a BS can include many radio access technologies, user association policies, and heterogeneous cells, depending on the transmission power and the cell coverage.

We consider a set $\mathcal{U} = \{1, \ldots, U\}$ of SPs where SP $u \in \mathcal{U}$ provides service for a group of users with specific service requirements so that each user only receives its service from a single SP. Although it is possible to consider different simultaneous services provided by some SPs, for a user, we consider such a situation as different independent users with corresponding different services. We also consider a set $\mathcal{V} = \{1, \ldots, V\}$ of MVNOs where MVNO $v \in \mathcal{V}$ rents radio resources from an InP and provides service for SP users based on the corresponding SLA (See Fig. 1). We assume users are stationary or moving slowly and they don’t move between different cell coverages (no handover) for the sake of simplicity. Users are distributed throughout the network coverage area and the spatial two-dimensional density of SP $u$-users at location $l \in \mathcal{L}$ is $\sigma_u(l)$ with $\sum_{u \in \mathcal{U}} \int_{l \in \mathcal{L}} \sigma_u(l) \, dl = 1$.

We consider a downlink scenario where SP $u$-users initiate downloading data flows at random times following a Poisson point process with rate $\lambda_u$. A data flow of SP $u$-users is a group of consecutive data packets that has a random number of bits with a memory-less distribution and an average $\Omega_u$. This is because, in reality, rather than transferring individual data packets within the time-scattered groups of PRBs, users usually start sessions of data downloading at random times when they want to download data files with random sizes. We also assume that a change in the flow transmission rate may happen (e.g., due to change of interference from other cells), but regarding the memory-less assumption on the number of data bits in a flow, it is equivalent to a retransmission with the new rate. Furthermore, since the flow download time is relatively large compared to the usual channel coherence times [30], we may ignore the effect of fast fading as a time-average effect on flow transmission. However, the users are prone to the shadowing effect. Since the users are either stationary or moving slowly, the movement during the flow transmission is small. Therefore, we assume shadowing stays unchanged during the flow transmission. Thus, during the flow transmission, the average signal strength remains constant. The authors in [28] provided a more detailed argument for the validity of this assumption. In addition, since we consider a flow traffic model, we ignore the slotted structure of PRBs in time domain and assume each flow to be a continuous-time stream of a random number of data bits.

### B. Slicing Configuration

Let us consider that frequency spectrum at cell $b$ has been divided into a set $\mathcal{S}_b$ with $\mathcal{S}_b = |\mathcal{S}_b|$ arbitrary, distinctive and sporadic chunks, each one called as a slice. Each slice can be considered for a specific type of service. Frequency range and the number of slices can vary in different cells and the definition of a slice is limited within the corresponding cell, i.e., for $b, b' \in B$ we have $\mathcal{S}_b \cap \mathcal{S}_{b'} = \emptyset$. In other words, it is possible that two slices at different cells have some overlapped frequency bandwidths, but slices at different cells have different indices. In the entire network we have a set $\mathcal{S} = \bigcup_{b \in B} \mathcal{S}_b$ of slices where each slice $s \in \mathcal{S}$ is composed of $Q_s$ frequency channels. Each channel has a bandwidth $\omega_s$ equivalent to a specific number of arbitrary PRBs, allocated to a single user in an OFDMA cellular network as a whole. To elaborate, when a slice channel is active, its corresponding PRBs simultaneously transmit data for the assigned user as long as a flow is being downloaded. Regarding the type of service for which a typical slice has been configured, the channels at different slices may have different frequency bandwidths.

The role of SP is to virtually aggregate its users’ flow traffic and distribute it among multiple MVNOs based on mutual SLAs. Then, if possible, each MVNO assigns each flow (i.e, the corresponding user) to a channel on a proper slice. MVNOs can have different number of slices in different cells and correspond them to flows of different service types. Consequently, we define $\mathcal{S}_{b,v} \subseteq \mathcal{S}_b$ as the set of slices in cell $b$ that has been rented by MVNO $v$ with $\mathcal{S}_{b,v} = |\mathcal{S}_{b,v}|$. We refer to each manner of slice assignment as a slicing configuration. In this paper, we will not investigate the designing problem of slicing configuration. Instead, we will analyze the performance of a typical slicing configuration by analytical modeling of all interactions among slices. Fig. 2 depicts a typical slicing configuration with $\mathcal{S} = \{1, \ldots, 12\}$, $\mathcal{S}_1 = \{1, 2, 3, 4\}$, $\mathcal{S}_{1,1} = \{1, 3\}$ and $\mathcal{S}_{1,2} = \{2, 4\}$. Moreover, slices 1 and 4 at cell 1, respectively, rented by MVNOs 1 and 2, are similar and have three channels, but slices 2 and 3 have only one channel. Regarding the type of service provided by each SP, all corresponding frequency channels have similar bandwidth.

### C. Isolation Among Slices

Isolation is a key feature of effective network slicing which means user activity on one slice should not affect other slices.

---

**Fig. 2.** A typical slicing configuration. Both SP 1 and 2 distribute their traffic among MVNO 1 and 2 and MVNOs exploit their slices in different cells to support traffics.
However, flows assigned to slices in adjacent cells will experience interference if being transmitted on channels with some overlapped frequencies. It is worth mentioning that interference coordination techniques like fractional frequency reuse (FFR), which partition the cell’s bandwidth based on region, are not applicable to our scenario as they require the operator’s full control over bandwidth throughout the network. However, in our network slicing scenario, multiple virtual operators have leased distinct parts of the available spectrum on a common infrastructure to maximize the service rate of their own customers individually, and they do not cooperate per se. $N_{s,q}$ is defined as the set of slice-channel pairs which have frequency overlap with channel $q$ on slice $s \in S$ including $(s,q)$. In other words, for all $s \in S$ and $0 \leq q \leq Q_s$ we have

$$N_{s,q} = \left\{ (s',q') \mid s' \in S, 0 \leq q' \leq Q_{s'}, (s,q) \text{ have frequency overlap} \right\}.$$  

We define interference vector $I_{s,q} \in \{0,1\}^{|N_{s,q}|}$ as a binary vector so that element $I_{s,q}(s',q')$ is 1 or 0 corresponding to whether channel $(s',q') \in N_{s,q}$ is active (i.e., a data flow is allocated to $(s',q')$) or not. With this in mind, SINR for a tagged flow, i.e., a flow under investigation, which is being transmitted by BS $b$ on channel $q$ of slice $s \in S_b$ towards location $l \in L_b$ under interference vector $I_{s,q}$ can be written as

$$\gamma_{s,q}(l,I_{s,q}) = \frac{\sum_{s',q'} I_{s,q}(s',q') P_{s',q'}^\text{IN} l_{s',q'} \gamma_{s',q'}(l) l_{s',q'}^N}{\sum_{s',q'} I_{s,q}(s',q')}$$  

where $P_{s',q'}^\text{IN}(l)$ is the signal power at $l \in L_b$, received on channel $q$ of $s \in S_b$ and $l_{s',q'}^N$ is the interference power affecting $(s,q)$ at $l \in L_b$, received from $b' \in B$ on channel $q'$ of $s' \in S_b$ and finally $N_0$ is the additive noise power. Afterward, using Shannon–Hartley theorem and ignoring the limitation of user processing power, we can obtain wireless link capacity as

$$C_{s,q}(l,I_{s,q}) = \eta_1 \log_2(1 + \eta_2 \gamma_{s,q}(l,I_{s,q})),$$  

where $\eta_1$ and $\eta_2$ are bandwidth and SINR efficiencies, respectively [31].

By harmonic averaging (3) on all possible locations $l \in L_b$ for SP $u$-users [28], we can achieve interference-dependent average wireless link capacity for SP $u$-user flows being transmitted on $(s,q)$ as

$$C^{-1}_{s,q,u}(I_{s,q}) = \int_{l \in L_b} C^{-1}_{s,q,u}(l,I_{s,q}) \sigma_u(l \mid b) dl,$$  

where $\sigma_u(l \mid b) = \frac{\sigma_u(l)}{\int_{l \in L_b} \sigma_u(l) dl}$ is the SP $u$-users density at location $l$ given that $l \in L_b$. Note that, assuming some coverage overlap between BSs, the user density at each BS may vary in overlapping and non-overlapping areas. Knowing the density values, i.e., $\sigma_u(l \mid b)$, one can relax the no overlap assumption indicated in Section II-A. Furthermore, user mobility manifests itself in $\sigma_u$ as it has been studied in other works [32], [33].

Eq. (4) implies that the average wireless link capacity for SP $u$-user flows is a function of interference from slices on adjacent cells. On the other hand, the probability of meeting interference during transmissions in a cellular network is a function of wireless link capacity in different cells since lower link capacity means more transmission time and therefore, a higher probability of overlap in the time domain. It is worth noting that indirect interaction of a slice with our desired slice is included in the temporal behavior of $I_{s,q}$. Thereupon, wireless link capacity is coupled among pairs of network slices and needs to be analyzed considering dynamic environment. To guarantee the inter-slice isolation, this problem needs to be investigated in detail. That is, we need to exploit an analytical model that is able to consider the effect of dynamic coupled interference leading to variable transmission rate (capacity) during downloading a flow. In fact, one can guarantee the required QoS for each slice by knowing the average wireless link capacity under coupled interference and allocating enough resources. Furthermore, by understanding the interference effect on the wireless link capacity and having performance metrics to evaluate isolation, one can control the inevitable inter-slice interference and increase the isolation using a suitable spectrum allocation. For this purpose, we use analysis based on queueing theory to model and study coupled interactions among slices.

### III. QUEUING THEORETICAL ANALYSIS

In this section, we propose a model based on queueing theory to address the interference coupling problem and analyze network performance metrics.

#### A. Flow Traffic and Queue Model

As we mentioned in Section II, users of SPs generate independent and identically distributed (i.i.d.) data flow traffic at random times following a Poisson point process such that the flow rate for SP $u$-users is $\lambda_u$. Then, based on some contracts between SPs and MVNOs, traffic of the SP users needs to be handled by the corresponding MVNOs. As shown in Fig. 1, to control the network performance and ensure QoS for different types of services, SPs and MVNOs are able to manage the distribution of their data flow traffic. For each SP $u$, we define $P_{u,v}$ with $\int_{v \in V} P_{u,v} = 1$ as the probability that SP $u$ delivers its flow traffic to MVNO $v$. Moreover, for $s \in S_{b,u}$, we define $P_{u,v,s}$ with $\sum_{s \in S_{b,u}} P_{u,v,s} = 1$ as the probability that MVNO $v$ assigns a flow of SP $u$ belonging to cell $b$, to slice $s$. Therefore, MVNOs can only distribute the user traffic between slices of cell $b$, which is the cell associated with a tagged user, and not between the cells. However, this can be extended to the case that the MVNO assigns its traffic to any BS (not necessarily to the associated BS), i.e., $\sum_{s \in S} P_{u,v,s} = 1$, and manages the BSs load. The values of $P_{u,v}$ and $P_{u,v,s}$ depend on SPs and MVNOs policies. Afterward, for slice $s \in S_{b,u}$, we can achieve the arrival rate of users’ flows from SP $u$ as

$$\lambda_{s,u} = \lambda_u P_{u,v,s} \sigma_{u,b} P_{u,v,s}$$  

where $\sigma_{u,b} = \int_{l \in L_b} \sigma_u(l) dl$ is the probability that SP $u$-users are in cell $b$.

We consider each slice $s \in S$ as a queue where data flows are customers and channels are servers. Then herewith, we will use the terms slice and queue interchangeably. Flows (customers)
from SP u arrive at queue s with rate \( \lambda_{s,u} \) as a Poisson process and stay in queue until they receive service, i.e., being completely transmitted on one of \( Q_s \) channels of the corresponding slice. The service time for each flow depends on its number of bits and wireless link capacity. Therefore, regarding interference coupling among slices and other assumptions, we consider a negative exponential distribution with variable rates. This means that we have a network of queues with service rate for each queue depending on the state of the other queues (i.e., the status of each channel) in the network. If the number of arrived customers is more than the existing servers, extra customers are buffered until we have a total of \( Q_s^{\max} \) customers in the queue. If the buffer capacity is full, newly arrived customers would be blocked. Accordingly, in the entire system, we would have a network of queues where each one has a standard \( M/M(n)/Q_s/Q_s^{\max} \) model where \( n \) represents the network state. Since the state of each queue depends on the state of the whole network, we propose a state-dependent queueing network to model our system.

### B. Our Proposed State-Dependent Queueing Network Model

We say \( s, s' \in S \) interact with each other, when they either have an overlapped frequency range (direct interaction) or there exists at least one slice \( s'' \in S \) that has some overlapped distinct frequency ranges with both \( s \) and \( s' \) (indirect interaction). For example, in Fig. 2, slices \( s = 7 \) and \( s = 8 \) indirectly interact with each other because they both have overlapped frequency range with \( s = 12 \). Keeping this in mind, one can partition \( S \) into disjoint subsets where queues in each subset interact with each other but not with queues in other subsets. Then, queues in each subset can be analyzed independently. Without loss of generality, we assume that all queues in \( S \) interact with each other.

Now consider a two-part and non-negative vector \( n = (n_{LH}, n_{RH}) \) of the state space \( \mathbb{N}_n \) as the network state vector where \( n_{LH} \) and \( n_{RH} \) are the left-hand and the right-hand parts with equal size \( |S| \). For each queue(s) \( s \in S \), the elements \( n_{RH}(s) \) and \( n_{LH}(s) \), respectively, represent the number of customers and the remaining buffer capacity, therefore we have \( n_{LH}(s) + n_{RH}(s) = Q_s^{\max} \). Also consider a similar two-part and non-negative vector \( a = (a_{LH}, a_{RH}) \) of the state space \( \mathbb{N}_a \) as the movement vector where for each queue \( s \in S \), the elements \( a_{RH}(s) \) and \( a_{LH}(s) \), respectively, represent the departures from and arrivals to \( s \). Regarding the continuous time dynamic in our scenario, only single movements are possible at each time. Therefore, for all vectors \( a \), the first norm is equal to one, i.e., \( ||a||_1 = 1 \). As a numerical example, for slicing configuration shown in Fig. 2 with \( |S| = 12 \), following vectors correspond to arrivals at queue \( s = 1 \) when there are no customers in the network and each queue has a unity-size buffer (i.e., \( Q_s^{\max} = Q_s + 1 \))

\[
\begin{align*}
n = (4, 2, 2, 4, 4, 2, 4, 4, 2, 2, 0, 0, 0, 0, 0, 0, 0, 0) \\
a = (1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0)
\end{align*}
\]

For each possible pair of \((n, a)\) vectors, we define \( \xi(n, a) \) as the rate of leaving the state \( n \) through the movement \( a \). Vector \( a \), then, transforms into another vector \( a' \) with probability \( r(a, a') \) which changes the network state from \( n \) to \( n' \). If \( a \) corresponds to a type of arrival, \( a' \) corresponds to a type of departure, and vice versa. We define \((n, a)\) as the set of all pairs of \((n, a)\) that can transform into each other, i.e., \( n - a = n' - a' \). In the previous example, vector \( a \) transforms into \( a' = (0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0) \) with \( r(a, a') = 1 \) changing the network state from \( n \) to \( n' = (3, 2, 2, 4, 4, 2, 4, 4, 2, 2, 1, 0, 0, 0, 0, 0, 0, 0, 0) \). Obviously, in our model at each \((n, a)\) just two members exist. We also define \( \pi(n) \) as the network steady-state probability distribution, i.e., the normalized network stationary measure.

Now we use the following theorem from [34].

**Theorem 1:** The network stationary measure \( \tilde{\pi} \) satisfies the local balance equations

\[
\tilde{\pi}(n)\xi(n, a) = \sum_{(n', a') \in (n, a)} \tilde{\pi}(n')\xi(n', a')r(a', a) \quad (6)
\]

if and only if there exist a non-negative function \( \Psi \) and a positive function \( \Phi \) such that for all possible pairs of \((n, a)\) the rate \( \xi(n, a) \) takes the form

\[
\xi(n, a) = \frac{\Psi(n - a)}{\Phi(n)}. \quad (7)
\]

When this is true, we have

\[
\tilde{\pi}(n) = \Phi(n) \quad (8)
\]

**Proof:** Please see chapter 9 in [34].

In general, definition of \( P_{MVNO}^{\text{U.S.}} \) implies that each slice can serve customers from multiple SPs. Since slices are configured based on service attributes, and each SP usually provides a special type of service, we assume the value of \( P_{MVNO}^{\text{U.S.}} \) is such that slice \( s \in S \) has only customers from a single SP \( u \) or similar SPs with the same service attributes. Thus, we may use the notation \( u_s \) to remind readers that \( u \) depends on \( s \). If we name the customers from each SP a class of customers, then the last assumption means that we consider queues with a single class of customers. Therefore, for queue \( s \) which has only customers from SP \( u \), we define \( \Lambda_{s,u} \) and \( M_{s,u} \) as the state-dependent arrival and service rates. Due to the coupled interference, \( M_{s,u} \) is dependent on the network state, i.e., the right-hand part \( n_{RH} \), while \( \Lambda_{s,u} \) is only dependent on the state of \( s \), i.e., \( n_{RH}(s) \). Now if we define functions \( \Phi \) and \( \Psi \) in a way that for each \( s \in S \) we have

\[
\xi(n, a) = \frac{\Psi(n - a)}{\Phi(n)} = \begin{cases} \Lambda_{s,u}(n_{RH}(s)) & ; \ a_{LH}(s) = 1 \\
M_{s,u}(n_{RH}) & ; \ a_{RH}(s) = 1 \end{cases}
\]

then as a corollary of Theorem 1, \( \Phi(n) \) is a stationary measure of the network. For this purpose, we present the following proposition.
Proposition 1: Consider a two-part and non-negative variable vector \( \mathbf{x} = (\mathbf{x}_{\text{LH}}, \mathbf{x}_{\text{RH}}) \) similar to \( \mathbf{n} \). The functions

\[
\Phi_1(x) = \prod_{s=1}^{\lvert S \rvert} \prod_{x_{\text{RH}}(s) = 0} \Lambda_{s, x_{\text{RH}}(s)}^{-1}(x) \prod_{x_{\text{RH}}(s) \neq 0} \Lambda_{s, x_{\text{RH}}(s)}^{-1}(x),
\]

\[
\Phi_2(x) = \prod_{s=1}^{\lvert S \rvert} \prod_{x_{\text{RH}}(s) = 0} M_{s, x_{\text{RH}}(s)}^{-1}(x),
\]

\[
\Psi_1(x) = \prod_{s=1}^{\lvert S \rvert} (1 - (1 - \Lambda_{s, x_{\text{RH}}(s)}(Q_{\text{max}}^s)) \delta(1 + \mathbf{x}_{\text{LH}}(s)))
\]

\[
\Psi_2(x) = \prod_{s=1}^{\lvert S \rvert} \prod_{x_{\text{RH}}(s) = 0} (1 + x_{\text{LH}}(s))
\]

satisfy (9) when the rate \( \xi(n, a) \) correspond to a departure, i.e., \( a_{\text{RH}}(s) = 1 \).

**Proof:** Please see Appendix A, available online.

In Proposition 1, \( \delta(\cdot) \) is the Kronecker delta function and \( e(x) \) is a two-part vector-valued function indicating whether there has been a movement in each queue or not and we have

\[
e(x) = (0_{\text{LH}}, e_{\text{RH}}(x)),
\]

where the left-hand part \( 0_{\text{LH}} \) is a vector with size \( |S| \) and the right-hand part \( e_{\text{RH}}(x) \) is a vector with size \( |S| \).

For element \( e_s \) we have

\[
e_s = \begin{cases} 1; & x_{\text{RH}}(s) + x_{\text{LH}}(s) \neq Q_{\text{max}}^s \\ 0; & \text{o.w.} \end{cases}
\]

Finally, notation \( \Phi_2(x)_{\rvert S \rvert \times \rvert S \rvert + e(x)} \) implies that one must determine \( \Phi_2(x) \) first and then substitute \( x \) with \( x + e(x) \). The general idea here is to use the function \( e(x) \) to model perfectly all departure movements (but not the arrival movements) in (11) since a distinction between arrival and departure movements exists in (9). We explain this matter in the sequel.

**C. State-Dependent Service Rate**

To determine \( M_{s, u}(n_{\text{RH}}) \), we need to consider that for each realization of \( n_{\text{RH}} \), multiple channel allocations leading to multiple values for \( \mathbf{I}_{s, q} \) (the interference vector corresponding to \( (s, q) \)) are possible. By averaging \( C_{s, q, u}^{-1}(\mathbf{I}_{s, q}) \) (see (4)) over all possible values of \( \mathbf{I}_{s, q} \) we have

\[
C_{s, q, u}(n_{\text{RH}}) = \sum_{\mathbf{I}_{s, q} \in \{0,1\}^{\lvert S \rvert \times \lvert S \rvert - 1}} \Pr(\mathbf{I}_{s, q}) C_{s, q, u}^{-1}(\mathbf{I}_{s, q}),
\]

where \( \Pr(\mathbf{I}_{s, q}) \) is the probability that the interference vector \( \mathbf{I}_{s, q} \) occurs. MVNOs can assign flows to vacant channels with different policies leading to different \( \Pr(\mathbf{I}_{s, q}) \).

In the case of perfectly random and independent channel allocation policy in all queues, we have

\[
\Pr(\mathbf{I}_{s, q}) = \prod_{s' \in N_{s, q}} \Pr(\mathbf{I}_{s, q, s'}),
\]

where \( \mathcal{N}_{s, q} = \{s'|q', (s', q') \in \mathcal{N}_{s, q}\} \) is the set of slices which have frequency overlap with \( (s, q) \) and \( \mathbf{I}_{s, q, s'} \) is a sub-vector of \( \mathbf{I}_{s, q} \) corresponding to the potentially interfering channels of slice \( s' \), and we have

\[
\Pr(\mathbf{I}_{s, q, s'}) = \begin{cases} 1; & \text{dim}(\mathbf{I}_{s, q, s'}) = \lvert \mathbf{I}_{s, q, s'} \rvert = 1, \text{dim}(\mathbf{I}_{s, q, s'}) \geq \lvert \mathbf{I}_{s, q, s'} \rvert - Q_{\text{max}}^s > 0; \\ 0; & \text{o.w.} \end{cases}
\]

(16)

where \( \text{dim}(\cdot) \) and \( \lVert \cdot \rVert_1 \) respectively, represent the dimension and the first norm of a vector. In justifying (16), it is worth noting that it is possible that more than one channel of a typical slice \( s' \) have frequency overlap with \( (s, q) \). Therefore, the channels of slice \( s' \) can be divided into \( \text{dim}(\mathbf{I}_{s, q, s'}) \) interfering channels and \( Q_{\text{max}}^s - \text{dim}(\mathbf{I}_{s, q, s'}) \) non-interfering channels.

Since, in general, frequency range overlaps between two slices are asymmetric (e.g., slices 1 and 5 in Fig. 2), channels of a single slice may have different average wireless link capacities indicated by \( C_{s, q, u}(n_{\text{RH}}) \) in (14). In terms of queueing theory, this means that we have queues with heterogeneous servers with different service rates, which are complex to analyze. Therefore, for simplicity, we study an equivalent homogeneous queue where the service rate of each server in the equivalent queue is the average of the heterogeneous queue service rates [35]. For this purpose, in each state \( n_{\text{RH}} \), considering a tagged flow that is assigned to an arbitrary channel \( q \) of slice \( s \) and averaging \( C_{s, q, u}(n_{\text{RH}}) \) over all channels of slice \( s \), we have

\[
C_{s, u}(n_{\text{RH}}) = \sum_{q=1}^{Q_s} \Pr(q) C_{s, q, u}(n_{\text{RH}}),
\]

(17)

as the wireless link capacity for each channel in the equivalent homogeneous slice (queue). \( \Pr(q) \) is the probability that channel \( q \) is assigned to a transmitting tagged flow. In the case of uniformly random channel allocation scheme, we have \( \Pr(q) = 1/Q_s \). Afterwards, we can achieve the state-dependent service rate on slice \( s \) (for SP \( u \)-customers) by dividing the obtained wireless link capacity by the average number of bits of the flows as in the following:

\[
M_{s, u}(n_{\text{RH}}) = \sum_{q=1}^{Q_s} \frac{C_{s, u}(n_{\text{RH}})}{\text{min}(n_{\text{RH}}(s), Q_s)} \min(n_{\text{RH}}(s), Q_s) - \frac{Q_{\text{max}}^s}{Q_s} \leq n_{\text{RH}}(s) \leq Q_{\text{max}}^s, 1; \text{o.w.}
\]

(18)

where the term \( \frac{C_{s, u}(n_{\text{RH}})}{\text{min}(n_{\text{RH}}(s), Q_s)} \) represents the service rate of flows at each server of queue \( s \) and the term \( \text{min}(n_{\text{RH}}(s), Q_s) \) implies that the service rate for queue \( s \) increases as the number of customers \( n_{\text{RH}}(s) \) increases until we have \( Q_s \) customers or more in which all servers are busy. When there are zero customers in a queue, the service rate is meaningless. Therefore, in order to neutralize the effect of \( M_{s, u}(n_{\text{RH}}) \) in (10) and (11), we define \( M_{s, u}(n_{\text{RH}}) = 1 \) in all such states.

Authorized licensed use limited to the terms of the applicable license agreement with IEEE. Restrictions apply.
So far and for a known network state vector \( \mathbf{n} \), service rate \( M_{s,u}(\mathbf{n}_{RH}) \) has been obtained by multiple averaging over \( C_{s,q,u}(\mathbf{I}_{s,q}) \), regarding the fact that in each network state, channel allocation for each slice is random and independent of the state of other slices. However, when investigating the steady-state probability distribution for a specific queue, the coupling effect among slices is a burden. In fact, to achieve the steady-state probability distribution, one must know the service rate for all states of queues, but, as indicated in \( M_{s,u}(\mathbf{n}_{RH}) \), the service rate is a function of the state of other network queues. Thus, we need to solve the equations iteratively.

### D. State-Dependent Arrival Rate

We define state-dependent arrival rate as

\[
\Lambda_{s,u}(\mathbf{n}_{RH}(s)) = \begin{cases} \lambda_{s,u}; & 0 \leq \mathbf{n}_{RH}(s) < Q_s^{\max} \\ 0; & \text{o.w.} \end{cases}
\]  

(19)

Obviously, due to the consideration of blocking, when there are \( Q_s^{\max} \) customers in a queue, the arrival rate to the queue is zero.

The defined \( \Phi \) and \( \Psi \) do not satisfy (19) perfectly. In fact, by substituting state-dependent rates defined by (18) and (19) in (10) and (11), one can investigate that (9) will have the following form

\[
\xi(n, a) = \psi(n - a) \Phi(n) = \begin{cases} \Lambda_{s,u}(\mathbf{n}_{RH}(s))\beta_{s,n,a}; & a_{RH}(s) = 1 \\ M_{s,u}(\mathbf{n}_{RH}); & \text{o.w.} \end{cases}
\]

(20)

where \( \beta_{s,n,a} \) is a coefficient shown due to \( \Phi \) and \( \Psi \) imperfection when the pair \((n, a)\) corresponds to a permitted arrival.

For arrivals to slice \( s \in S \) with \( 0 \leq \mathbf{n}_{RH}(s) < Q_s^{\max} \) and from (10) and (11) we obtain

\[
\beta_{s,n,a} = M_{s,u}(\mathbf{n}_{RH}) \prod_{s' \neq s, x > 0} M_{s', u'}(\mathbf{n}_{RH} + \mathbf{1}_s),
\]  

(21)

where \( \mathbf{1}_s \) is a vector with size \( |S| \) that its \( s \)-th element is equal to one and the rest are equal to zero.

To neutralize the effect of \( \beta_{s,n,a} \), a solution is to introduce a modified state-dependent arrival rate \( \Lambda_{s,u}(\mathbf{n}_{RH}(s)) \) such that \( \Lambda_{s,u}(\mathbf{n}_{RH}(s))\beta_{s,n,a} \approx \Lambda_{s,u}(\mathbf{n}_{RH}(s)) \). Therefore, we define

\[
\hat{\Lambda}_{s,u}(\mathbf{n}_{RH}(s)) = \begin{cases} \lambda_{s,u}/\beta_{s,n_{RH}(s)}; & 0 \leq \mathbf{n}_{RH}(s) < Q_s^{\max} \\ 0; & \text{o.w.} \end{cases},
\]

(22)

where \( \beta_{s,n_{RH}(s)} \) is the modification factor. Hence, for all pairs of \((n, a)\) that correspond to an arrival to slice \( s \in S \) when it has \( \mathbf{n}_{RH}(s) \) customers, we approximate (9) with (20) by choosing \( \beta_{s,n_{RH}(s)} \) such that \( \lambda_{s,u}/\beta_{s,n_{RH}(s)} \) has the least weighted squared error with respect to \( \lambda_{s,u} \). In other words, to find the value of \( \beta_{s,n_{RH}(s)} \) for each \( s \in S \) and \( 0 \leq \mathbf{n}_{RH}(s) < Q_s^{\max} \), we have

\[
\min_{\beta_{s,n_{RH}(s)}} \sum_{x \in \mathbf{n}_{n,a \in \mathbf{n}_{RH}(s)} \neq \mathbf{n}_{RH}(s)} \hat{\Phi}_{s,n_{RH}(s)}(x) \left( 1 - \frac{\beta_{s,n_{RH}(s)}}{\beta_{s,n_{RH}(s)}} \right)^2
\]

(23)

### Algorithm 1: Obtaining Network Steady-State Probability

1: Initialize \( \beta_{s,n_{RH}(s)}^{(1)} = 1 \) for all \( s \in S \) and \( 0 \leq \mathbf{n}_{RH}(s) < Q_s^{\max} \)
2: Initialize \( k \leftarrow 1 \) and choose \( \epsilon > 0 \)
3: repeat
4: Calculate for all possible \((n, a)\)
\[
\hat{\Lambda}_{s,u}(\mathbf{n}_{RH}(s)) = \begin{cases} \lambda_{s,u}/\beta_{s,n_{RH}(s)}; & 0 \leq \mathbf{n}_{RH}(s) < Q_s^{\max} \\ 0; & \text{o.w.} \end{cases}
\]
5: Calculate for all possible \((n, a)\)
\[
\Phi_{s,n_{RH}(s)}^{(k)}(n) = \prod_{a \neq 0} (\hat{\Lambda}_{s,u}(\mathbf{n}_{RH}(s)))^{-1} \sum_{x \in S_{\mathbf{n}_{RH}(s)}} \Phi_{s,n_{RH}(s)}^{(k-1)}(n)
\]
6: Update for all \( s \in S \) and \( 0 \leq \mathbf{n}_{RH}(s) < Q_s^{\max} \)
\[
\beta_{s,n_{RH}(s)}^{(k+1)} = \frac{\sum_{x \in S_{\mathbf{n}_{RH}(s)}} \Phi_{s,n_{RH}(s)}^{(k)}(n) \beta_{s,n_{RH}(s)}^{(k)}}{\sum_{x \in S_{\mathbf{n}_{RH}(s)}} \Phi_{s,n_{RH}(s)}^{(k)}(n) \beta_{s,n_{RH}(s)}^{(k)}}
\]
7: \( k \leftarrow k + 1 \)
8: until \( \beta_{s,n_{RH}(s)}^{(k)} - \beta_{s,n_{RH}(s)}^{(k-1)} < \epsilon \)
9: return the latest \( \Phi_{s,n_{RH}(s)}^{(k)}(n) \) for all possible network states

where \( x \) is a variable vector similar to \( n \) and

\[
\hat{\Phi}_{s,n_{RH}(s)}(x) = \sum_{x' \in \mathbf{n}_{n,a \in \mathbf{n}_{RH}(s)} \neq \mathbf{n}_{RH}(s)} \Phi(x')
\]

(24)

is the normalized weight for state \( x \) as \( \Phi(x) \) is a stationary measure of the network according to Theorem 1.

Since \( \Phi(x) \) is dependent on \( \beta_{s,n_{RH}(s)} \) and vice versa, an iterative approach can be used to obtain \( \Phi(x) \) as stated in Algorithm 1. To show its validity please see Appendix B, available online. To make reading easier, for the rest of the paper, we represent the constrained set under sigma in (23) as \( COND(x, a) = \{ x \in \mathbf{n}_{n}, a \in \mathbf{n}, a_{RH}(s) = 1, x_{RH}(s) = \mathbf{n}_{RH}(s) \} \).

For each network state \( n \), the obtained \( \Phi(n) \) using Algorithm 1 is an approximate stationary measure of the network. The accuracy of this approximation depends on the weighted squared error in (23) which is related to three factors: the number of terms in summation, the weights \( \hat{\Phi}_{s,n_{RH}(s)}(x) \) and the values of \( \beta_{s,n_{RH}(s)} \).

An increase in the number of possible network states means more possible pairs of \((x, a)\) in (23), leading to larger error value. The value of \( \hat{\Phi}_{s,n_{RH}(s)}(x) \) can vary depending on the network traffic as it is a normalized stationary measure of the network. In light traffic, the weights for states with a small number of customers are dominant over other states. In heavy traffic, the weights for states with a large number of customers are dominant over other states. In other words, for light and heavy traffic, the
number of terms in summation (23) is usually fewer than the one in moderate traffic and therefore, the error value is less. Our numerical results confirm this matter.

To investigate the third factor, if we call \( \hat{\beta}^s_{n_{RH}(s)} \) as the value that has been obtained from the last iteration of Algorithm 1, we can calculate the minimum cost. For arrivals to slice \( s \in S \) with \( 0 \leq n_{RH}(s) < Q^m_s \), by substituting \( \hat{\beta}^s_{n_{RH}(s)} \) into (23) we have

\[
\sum_{\text{COND}(x,a)} \hat{\Phi}^s_{n_{RH}(s)}(x) \left( 1 - \frac{\beta_{x,a}}{\hat{\beta}^s_{n_{RH}(s)}} \right)^2 = \sum_{\text{COND}(x,a)} \hat{\Phi}^s_{n_{RH}(s)}(x) \times \left( 1 - \frac{\beta_{x,a} \sum_{\text{COND}(x',a')} \hat{\Phi}^s_{n_{RH}(s)}(x') / \beta_{x',a'}^2}{\sum_{\text{COND}(x,a)} \hat{\Phi}^s_{n_{RH}(s)}(x') \beta_{x',a'}^2} \right)^2
\]

\[
= 1 - \frac{\sum_{\text{COND}(x,a)} \hat{\Phi}^s_{n_{RH}(s)}(x) / \beta_{x,a}^2}{\sum_{\text{COND}(x,a)} \hat{\Phi}^s_{n_{RH}(s)}(x)} \right)^2
\]

\[
= 1 - \frac{E^2[\beta_{x,a}]}{E[\beta_{x,a}^2]} = \text{Var}[\beta_{x,a}] / E[\beta_{x,a}^2]
\]

(25)

where the symbols \( E[\beta_{x,a}] = \sum_{\text{COND}(x,a)} \hat{\Phi}^s_{n_{RH}(s)}(x) / \beta_{x,a} \) and \( \text{Var}[\beta_{x,a}] = E[\beta_{x,a}^2] - E^2[\beta_{x,a}] \) respectively represent the expectation and the variance. Consequently, any factor that increases the ratio between \( \text{Var}[\beta_{x,a}] \) and \( E[\beta_{x,a}^2] \), increases the weighted squared error in (23).

IV. NETWORK KPIs

In this section, using the network steady-state probability distribution \( \pi(n) \) obtained in Section III, we extract the network KPIs.

A. Slice Steady-State Probability Distribution

We define \( \pi_s \) as the steady-state probability distribution for slice \( s \) where \( \pi_s(n) \) indicates the steady-state probability distribution of \( n \) customers being present at queue \( s \) and we have

\[
\pi_s(n) = \sum_{n_{RH}(s) = n} \pi(n).
\]

(26)

We also define the blocking probability \( P^B \) as the probability of customers arriving at queue \( s \) being blocked and we have

\[
P^B = \pi_s(Q^m_s).
\]

(27)

B. Slice Throughput and Delay

We define \( T_{s,u} \) as the throughput for SP \( u \)-users in slice \( s \) and we have

\[
T_{s,u} = \lambda_{s,u} (1 - P^B) \Omega_u.
\]

(28)

We also define \( D_{s,u} \) as the average total delay for SP \( u \)-users in slice \( s \). Using the Little’s law we have

\[
D_{s,u} = \frac{\sum_{n_{RH}^m} n \pi_s(n)}{\lambda_{s,u} (1 - P^B)}.
\]

(29)

where the numerator indicates the average number of SP \( u \)-users (flows) in queue \( s \) and the denominator indicates the rate of SP \( u \)-flows admitted in queue \( s \).

For each customer entering queue \( s \), its total delay consists of waiting time in buffer and service time. We define \( D^W_{s,u} \) and \( D^S_{s,u} \), respectively, as the average waiting and service time for SP \( u \)-users in queue \( s \). With an approach similar to (29) we have

\[
D^W_{s,u} = \frac{\sum_{n_{RH}^m} n \pi_s(n - Q_s) \pi_s(n)}{\lambda_{s,u} (1 - P^B)}
\]

(30)

\[
D^S_{s,u} = \frac{\sum_{n_{RH}^m} \min(n, Q_s) \pi_s(n)}{\lambda_{s,u} (1 - P^B)}
\]

(31)

where the numerators, respectively, indicate the average number of SP \( u \)-users in buffer and servers of queue \( s \), and the denominators, respectively, indicate the rate of SP \( u \)-users entering buffer and servers of queue \( s \). One can verify that, for SP \( u \)-users in queue \( s \), the equation \( D_{s,u} = D^W_{s,u} + D^S_{s,u} \) holds as expected.

C. Network Throughput and Delay

Regardless of slice and cell, each MVNO has to serve SP \( u \)-users based on an SLA in the entire network. We define \( T^\text{MVNO}_{u,v} \) as the network throughput for SP \( u \)-users being served by MVNO \( v \) and we have

\[
T^\text{MVNO}_{u,v} = \sum_{b \in B} \sum_{s \in S_{b,v}} T_{s,u}.
\]

(32)

Similar to the slice delay and ignoring the blocking probability, we define \( D^\text{MVNO}_{u,v} \), \( D^\text{MVNO, W}_{u,v} \) and \( D^\text{MVNO, S}_{u,v} \), respectively, as the network total delay, network waiting time and network service time for SP \( u \)-users being served by MVNO \( v \) and we have

\[
D^\text{MVNO}_{u,v} = \sum_{b \in B} \sum_{s \in S_{b,v}} \sigma_{b,v} p^\text{MVNO}_{u,v,s} D_{s,u},
\]

(33)

\[
D^\text{MVNO, W}_{u,v} = \sum_{b \in B} \sum_{s \in S_{b,v}} \sigma_{b,v} p^\text{MVNO}_{u,v,s} D^W_{s,u},
\]

(34)

\[
D^\text{MVNO, S}_{u,v} = \sum_{b \in B} \sum_{s \in S_{b,v}} \sigma_{b,v} p^\text{MVNO}_{u,v,s} D^S_{s,u},
\]

(35)

and the equation \( D^\text{MVNO}_{u,v} = D^\text{MVNO, W}_{u,v} + D^\text{MVNO, S}_{u,v} \) holds.

D. Isolation Metrics

We already stated the importance of isolation among slices. To quantify and compare isolation level in the network, we propose some metrics. Suppose that the isolation for slices \( u \) and \( u' \) that are allocated to SP \( v \)-users is continuously increased from \( \lambda_{u,v} \) to \( \lambda_{u,v} + \Delta \lambda \). Since the network KPIs are functions of SP \( u \)-users traffic, a KPI versus interfering traffic curve can be obtained. Therefore, for network total delay, we can define average of delay deviation (ADD) and variance of delay deviation (VDD) as

\[
\text{ADD}(D^\text{MVNO}_{u,v}, u') = \frac{1}{\Delta \lambda} \int_{z=\lambda_{u,v}}^{\lambda_{u,v} + \Delta \lambda} \frac{D^\text{MVNO}_{u,v}}{D^\text{MVNO}_{u,v}} \, dz.
\]

(36)
propose an interference-aware channel allocation algorithm aiming to decrease interference level and keep the slices more isolated. To show the superiority of the proposed policy, we compute the KPIs introduced in the previous section. To this end, we need to find the average delay and throughput for a typical slice, so we employ the analytical model proposed in Section III. Since (15) and (16), derived in Section III, are based on random independent channel allocation, i.e., interference-blind, we need to adapt them to the interference-aware situation.

First, an allocation lookup table LT is constructed such that interference among slices is at a minimum level. For each pair \((s, q)\) in network state \(n\), the binary value \(LT(n, s, q) \in \{0, 1\}\) indicates whether channel \(q\) of \(s\) can be allocated to flows or not. Next, lookup table LT is distributed among MVNOs and is valid as long as the slicing configuration is not changed. Assuming cooperation among MVNOs to mitigate the interference and know the network state \(n\), an MVNO that owns slice \(s\) is only allowed to allocate flows to channel \(q\) if \(LT(n, s, q) = 1\). After identifying the permitted channels, MVNO randomly assigns flows to those channels.

To construct the lookup table and for each state \(n\), we prioritize slices \(s \in S\) according to the value of \(\text{RH}(s)/Q_s\), i.e., slices with a higher desired ratio of the number of customers to the total number of channels have higher priority. The motivation behind such prioritization is less flexibility in channel allocation in such slices. In the case of some slices having the same value for the desired ratio, we randomly prioritize them over each other.

Afterwards, at each state \(n\), we start by an all zero LT and obtain values of \(LT(n, s, q)\) for all slices \(s \in S\) in the order of their priorities. For this purpose, we calculate the average wireless link capacity for all channels belonging to each slice one at a time to choose the best ones. In other words, for all channels \(1 \leq q \leq Q_s\) in slice \(s\), we calculate \(C_{s, q, u}(1)\) using (4) ignoring the interference from slices with lower priority and assuming that slice \(s\) is only under the interference from slices with higher priority, i.e., all interfering channels that so far have value 1 in LT. Therefore, interference vector \(I_{s, q}\) in (4) can be obtained from the last values in LT.

Next, we rank slice \(s\) channels from highest to lowest according to the calculated \(C_{s, q, u}(1)\) and we choose the first \(\text{RH}(s)\) of them as they are the ones that are under the least interference. We refer to these chosen channels as the set \(Q_s^*\). In case of \(\text{RH}(s) \geq Q_s\), all channels will be chosen \((Q_s^* = \{1, \ldots, Q_s\})\). Then, we update the lookup table such that for \(q \in Q_s^*\), \(LT(n, s, q) = 1\). We continue the aforesaid process, from highest to lowest priority and once for each slice, until LT for all slices is updated. This process has been summarized in Algorithm 2.

Having the LT obtained from Algorithm 2, one can use the state-dependent queuing model mentioned in Section III, with some modifications, to achieve the network steady-state probability for the interference-aware channel allocation policy. For each state \(n\) and slice \(s\), the value \(LT(n, s, q)\) determines whether channel \(q\) can be used or not. In state \(n\), total number of \(\min(\text{RH}(s), Q_s)\) channels are allocated to slice \(s\). We assume
Algorithm 2: Lookup Table Construction.

1: for Each network state $n$ do
2:   Set $\mathrm{LT}(n, s, q) = 0$ for all pairs $(s, q)$
3:   Set slice $s$ priority based on $n_{\text{RH}}(s)/Q_s$
4:   for All $s \in S$ and from highest priority to lowest do
5:     For $1 \leq q \leq Q_s$, calculate $C_{s,q,u}(I_{s,q})$ using (4) considering that all channels that have the value 1 in LT are interfering
6:       Sort channels according to $C_{s,q,u}(I_{s,q})$, choose the first $n_{\text{RH}}(s)$ ones and call it $Q^*_s$
7:   end for
8:   end for
9: end for
10: return LT

that a tagged flow can randomly be assigned to any of the allocated channels, and therefore, $\Pr(q)$ in (17) will be

$$\Pr(q) = \begin{cases} \frac{1}{\min(n_{\text{RH}}(s),Q^*_s)} & \text{LT}(n, s, q) = 1 \\ 0 & \text{o.w.} \end{cases}. \quad (44)$$

To be specific, in the interference-aware channel allocation policy, the channels to be allocated are known for each state. Since each slice supports a single type of service and therefore all flows that are allocated to a slice have the same priority, when calculating the average rate for a tagged flow, the assigned channel to that flow is random. Furthermore, since allocated channels in state $n$ and for all slices are known, the interference on channel $q$ of slice $s$ is also known and accordingly the interference vector is known, i.e., $I^{*}_{s,q} \in \{0,1\}^{\sum \text{slice} - 1}$. Therefore, (15) would be modified as

$$\Pr(I_{s,q}) = \begin{cases} 1 & I_{s,q} = I^{*}_{s,q} \\ 0 & I_{s,q} \neq I^{*}_{s,q} \end{cases}. \quad (45)$$

As we already mentioned, this channel allocation policy requires a cooperation among MVNOs. They also need to always know the number of customers $n_{\text{RH}}(s)$ and lookup table LT. Exchanging the lookup table happens only once at the beginning, but the number of customers has to be exchanged continuously and therefore adds a data overhead to the network. Since $n_{\text{RH}}(s)$ is only a vector of non-negative integers, when its dimension is small, this overhead is negligible.

VI. NUMERICAL RESULTS

In this section, first, we present numerical results for a single-MVNO single-SP scenario with symmetric slicing configuration to show the accuracy of our analytic model and the proposed approximation in Algorithm 1 in comparison to discrete-event simulations (DES). Then, we consider a more comprehensive multi-MVNO multi-SP scenario with an asymmetric slicing configuration and examine the network KPIs and isolation metrics to evaluate the proposed interference-aware channel allocation policy compared to the random allocation policy as well as the exhaustive search-based optimal policy as a benchmark. We use MATLAB environment to achieve both analytic and DES results. A more detailed report on the results of this section can be found in [36].

A. Single-MVNO Single-SP Scenario

In this part, in order to show the validity of our proposed model, we consider a simple network slicing model including a single MVNO $v \in V = \{1\}$ and a single SP $u \in U = \{1\}$ and then compute slice KPIs. Similar to Fig. 1 and without loss of generality, we consider a cellular network with three BSs $b \in B = \{1, 2, 3\}$ located at the center of identical hexagonal cells with cell radius, i.e., hexagon edge, equal to 200 m and equal transmission powers. We consider a shadowing map, i.e., a typical realization of log-normal random variable with 6 dB standard deviation at all different locations, independently. Although different criteria are possible, to mitigate the effect of shadowing, users are associated with BSs based on the received signal strength indicator (RSSI). SP $u$-users that are uniformly distributed throughout the cells, initiate downloading data flows with its average size $\Omega_u = 80$ Mb. At random times following a Poisson process, SP 1 delivers all its flows to MVNO 1 and therefore $I^{\text{SP}}_{u,v} = 1, u = v = 1$.

MVNO $u$ has only one slice in each cell and therefore $s \in S = \{1, 2, 3\}$. We consider a symmetric slicing configuration as depicted in Fig 3 with $Q^\text{max} = 10$ and $Q_s = 5$ channels with bandwidth $w_s = 20$ MHz per channel for all $s \in S$. BS transmission power $P^\text{BS}_b$ in each cell is uniformly distributed throughout the specified bandwidth. MVNO $u$, in each cell, assigns all flows from SP $u$ to its slice and therefore, for all $s \in S$ we have $P^\text{MVNO}_{u,v,s} = 1$. Moreover, as cells are similar, we have $\lambda_{s,u} = \lambda_u/3$. Additional parameters are as in Table I.

To evaluate the accuracy of our state-dependent approach, we consider DES and compare delay and throughput results. We also compare our work with the averaged interference method inspired by [28]. In this method, an iterative approach is used to obtain the network steady-state probability distribution. Starting with an initial value for the distribution in the first iteration, the denominator of SINR in (2), is replaced by an equivalent interfering power obtained by weighted averaging over interfering powers from neighboring cells with network steady-state probabilities as weights. Afterwards, the equivalent service rate for each queue is achieved. Then, the network is decomposed into individual queues and using the obtained equivalent service rate, the steady-state probability distribution is calculated independently, which is used in the next iteration until convergence.

We evaluate the performance of the network for multiple values of SP $u$-users traffic $\lambda_u$ and maximum transmission power $P^\text{BS}_b$. Fig. 4(a) shows $D_{s,u}$ for the typical slice $s \in S$ when
HASHEMIAN AND ASHTIANI: ANALYTICAL MODELING AND IMPROVEMENT OF INTERFERENCE-COUPLED RAN SLICING

Fig. 4. Delay for the typical slice $s \in S$ versus BS power and flow traffic in neighboring cells when considering random and interference-aware channel allocations.

| TABLE I | NUMERICAL PARAMETERS |
|----------|----------------------|
| Common configurations | 128.1 + 37.6log10(d/km) dB |
| Carrier frequency | 2.0 GHz |
| Cell radius | 200 m |
| Thermal noise | $-174$ dBm/Hz |
| SINR efficiency $\eta_1$ | 0.63 |
| SINR efficiency $\eta_2$ | 0.4 |
| Number of packets generated in DES | 300000 |
| Number of iterations for Algorithm 1 | 20 |
| Power bandwidth | 72 MHz |
| Shadowing standard deviation | 6 dB |

| Single MVNO - Single SP configurations | |
|----------------------------------------|----------------|
| Average number of bits $\Omega_u$ | 80 Mbit |
| Slice channel bandwidth $w_s$ | 20 MHz |
| Number of slices $Q_s$ | 5 |
| Queue capacity $Q_s^{\text{max}}$ | 10 |

| Multi MVNO - Multi SP configurations | |
|--------------------------------------|----------------|
| Maximum transmission power $P_{b,\text{max}}$ | 45 dBm |
| SP 1-users channel bandwidth $w_s$ | 18 MHz |
| SP 2-users channel bandwidth $w_s$ | 6 MHz |
| Average number of bits $\Omega_1$ | 8 Mbit |
| Average number of bits $\Omega_2$ | 80 Mbit |

| TABLE II | THROUGHPUT (Mbps) FOR THE TYPICAL SLICE $s \in S$ VERSUS BS POWER AND FLOW TRAFFIC |

Table shows the throughput for the typical slice $s \in S$ using different BS powers and flow traffic scenarios. The throughput values are calculated using the proposed analytical model and compared with the baseline scenario. The throughput values are provided for different interference-aware channel allocation methods, including the proposed state-dependent queueing network model, Algorithm 1, and the averaged interference method.

Using a random channel allocation. When maximum transmission power has the lowest value, i.e., $P_{b,\text{max}} = 33$ dBm, average difference between DES and Algorithm 1 for $D_{s,u}$ is 0.8%. As $P_{b}$ increases from $33$ dBm to $48$ dBm, the average difference increases from 0.8% to 5.9%. Meanwhile, the average difference between DES and the averaged interference method, increases from 0.9% to 15.0%. Furthermore, Table II shows $T_{s,u}$ for the typical slice $s \in S$. It can be seen that increasing $P_{b,\text{max}}$ hardly affects the average difference between DES, Algorithm 1 and the averaged interference method.

Based on the results, one can conclude that the proposed state-dependent queueing network model is more accurate than the averaged interference method. The justification for this accuracy is that, in our proposed method, the service rate changes proportional to the interference changes. But in the averaged interference method, an equivalent averaged service rate represents all interfering states. Hence, when the difference between the full and zero interference states is large, an averaged service rate is not sufficiently accurate in modeling interference-coupled slices. Despite this, the accuracy of our proposed method is limited to the approximation in Algorithm 1 and the cost function (25).

Next, we try to evaluate the efficiency of the interference-aware channel allocation. First, using Algorithm 2, MVNO obtains an interference-aware lookup table LT that dictates which channels must be allocated at each slice. We use the same LT in both DES and state-dependent queueing network model. For the latter, we use LT along with Algorithm 1 as explained in Section V. In DES, each time the network state changes, MVNO assigns channels based on the LT values. To be in accordance with the analysis, once the allocated channels for a slice are known, we assume flows that belong to that slice are randomly assigned to the allocated channels. Therefore, the channel allocation, i.e., specifying the most suitable channel at each slice, is deterministic, but, similar to the analysis, flow assignment to allocated channels within a slice still happens randomly.

Fig. 4(b) shows $D_{s,u}$ for a typical slice $s \in S$. The average total delay $D_{s,u}$ for interference-aware channel allocation is on average 0.7% less than the random channel allocation when $P_{b,\text{max}} = 33$ dBm. As the maximum transmission power increases from $33$ dBm to $48$ dBm, i.e., when the interference is intensified, the average difference increases from 0.7% to 20.0%. Therefore, the interference-aware channel allocation can effectively reduce the interference. Fig. 4(b) also reveals that, when flow traffic is...
Fig. 5. Average and variance of $D^\text{MVNO}_{u,v}$ versus SP $u$-users flow traffic for the multi-MVNO multi-SP scenario.

B. Multi-MVNO Multi-SP Scenario

In this part, in order to investigate the inter-slice isolation, we consider a more comprehensive scenario with asymmetric slicing configuration similar to Fig. 2 with $S = \{1, \ldots, 12\}$, $V = \{1, 2\}$ and $U = \{1, 2\}$. For slices $s \in \{1, 4, 5, 7, 9, 11\}$ channel bandwidth $w_s = 6$ MHz, the number of channels $Q_s = 3$ and there are no buffers, i.e., $Q_{s,\text{max}} = 3$. For slices $s \in \{2, 3, 6, 8, 10, 12\}$ channel bandwidth $w_s = 18$ MHz, the number of channels $Q_s = 1$ and we consider a single buffer, i.e., $Q_{s,\text{max}} = 2$.

We consider the same cellular network described in Section VI-A with maximum transmission power $P^\text{BS}_{b} = 45$ dBm for all cells $b \in B$. SP 1-users and SP 2-users are distributed throughout the network coverage area with a two-dimensional uniform spatial distribution. We consider SP 1 as an example of a delay-sensitive service provider (like URLLC service) with average flow size $\lambda_1 = 8$ Mbit and SP 2 as an example of a throughput-sensitive service provider (like eMBB service) with average flow size $\lambda_2 = 80$ Mbit. MVNO 1 assigns flows of SP 1-users to slices $s \in \{1, 7, 9\}$, i.e., $P^\text{MVNO}_{1,1,s} = 1$ and flows of SP 2-users to slices $s \in \{3, 6, 12\}$, i.e., $P^\text{MVNO}_{2,1,s} = 1$. MVNO 2 assigns flows of SP 1-users to slices $s \in \{4, 5, 11\}$, i.e., $P^\text{MVNO}_{1,2,s} = 1$ and flows of SP 2-users to slices $s \in \{2, 8, 10\}$, i.e., $P^\text{MVNO}_{2,2,s} = 1$. For unmentioned cases, we have $P^\text{MVNO}_{u,v,s} = 0$. Additional parameters are listed in Table I.

Each SP $u \in U$ delivers its flows to MVNOs $v \in V$ with equal probability and therefore $P^\text{SP}_{u,v} = 0.5$. Since SP 2-users are throughput-sensitive, to evaluate the effect of SP 1-users activity on SP 2-users throughput, we keep $\lambda_2 = 0.6$ constant and sweep $\lambda_1$. Accordingly, each MVNO carries 24Mbps traffic for SP 2-users and 8Mbps to 48Mbps for SP 1-users (regarding negligible blocking). Moreover, the increase in SP 1-users traffic increases SP 2-users delay, $D^\text{MVNO}_{2,1,s}$ and $D^\text{MVNO}_{2,2,s}$, as a result of the increase in interference. Since the interference is coupled, SP 1-users delay, $D^\text{MVNO}_{1,1,s}$ and $D^\text{MVNO}_{1,2,s}$, also increase. Fig. 5(a) shows that both ADD and VDD values for delay have been decreased when using the interference-aware channel allocation compared to the random channel allocation. The reason for the difference in KPI improvement between MVNO 1 and MVNO 2 is that in the latter, slices that are allocated to SP 2-users, have less overlap with all slices that are allocated to SP 1-users. Similarly, Fig. 6(a) shows a slight improvement in ATD and VTD values for the network throughput when employing the interference-aware channel allocation.

Next, to evaluate the effect of SP 2-users activity on SP 1-users delay, we keep $\lambda_1 = 6$ and sweep $\lambda_2$. Accordingly, each MVNO carries 24Mbps traffic for SP 1-users and 8Mbps to 48Mbps for SP 2-users. In Figs. 5(b) and 6(b), we observe similar KPI improvement for the interference-aware channel allocation compared to random channel allocation. Additionally, Fig. 7(a) and (b) show the average SINR for the aforementioned experiments. Similarly, it can be seen that slice traffic on interfering SP leads to SINR reduction compared to the zero-interference condition. However, the amount of reduction when using the interference-aware allocation policy is less that means more isolation among slices.

From what mentioned above and by comparing average and variance values in Figs. 5 and 6, it can be concluded that delay is generally more affected by interference than throughput. This is because we did not consider any expiration age for the arrived packets. Moreover, the proposed interference-aware channel allocation can provide more inter-slice isolation. To establish a better understanding, as a benchmark, we also compare the average and variance values with the results of the optimal allocation policy. In optimal policy, the lookup table for channel allocation
is constructed by an exhaustive search. The improvement in results compared to the proposed interference-aware channel allocation policy is obvious. However, if we define complexity as the number of times that \( C_{s,q,u}(I_{s,q}) \) needs to be calculated during the construction of the lookup table, then the complexity of the proposed policy is \( \sum_n \sum_{s=1}^{S} \text{sgn}(n_{RH}(s)) Q_s \), as for each state \( n \) and for all channels of all slices that have at least one customer, the value of \( C_{s,q,u}(I_{s,q}) \) needs to be calculated once. On the other hand, the complexity of the exhaustive search is \( \sum_n \sum_{s=1}^{S} \min(n_{RH}(s), Q_s) \prod_{s=1}^{S} \min(n_{RH}(s), Q_s) \), as for each state \( n \), the value of \( C_{s,q,u}(I_{s,q}) \) needs to be calculated for all permutations of busy channels. For example, in our scenario, the complexity of the proposed policy is \( 5.22 \times 10^7 \) while the complexity of the exhaustive search is \( 2.86 \times 10^{10} \). Although comparison with an exhaustive search does not fully demonstrate the complexity advantage of the proposed policy, it still serves as a useful reference point.

VII. CONCLUSION

In this work, we investigated the isolation between slices in a multi-cell RAN slicing scenario. We first explained that the main bottleneck in the inter-slice isolation problem is the coupled-interference in multi-cell networks. In other words, channel interference due to direct (i.e., frequency overlap) or indirect (i.e., frequency overlap with a channel in a third slice) interaction between slices results in dynamic behavior for slices. We proposed an analytical model comprised of a network of state-dependent queues. Then, to achieve the network steady-state probability distribution, we proposed an iterative algorithm. Further, we defined KPIs that quantify QoS and isolation. At last, we proposed and analyzed an interference-aware channel allocation policy aiming to improve the defined KPIs by reducing the frequency overlap among slices. Our numerical results demonstrated that the proposed analytical model fairly follows the simulation results in terms of slice delay and throughput.
Besides, our proposed interference-aware policy improved our defined isolation metrics and SINR compared to a random allocation policy. In addition, compared to the optimal channel allocation constructed by an exhaustive search, our proposed interference-aware policy offers fair isolation with less complexity.

REFERENCES

[1] 5GPPP, “5GPPP architecture working group: view on 5G architecture,” version 4.0, Oct. 2021. [Online]. Available: https://5g-ppp.eu/wp-content/uploads/2021/11/Architecture-WP-V4.0-final.pdf
[2] Ericsson, “Ericsson mobility report,” Nov. 2022. [Online]. Available: https://www.ericsson.com/4ac28d/assets/local/reports-papers/mobility-report/documents/2022/ericsson-mobility-report-november-2022.pdf
[3] X. Lin, D. Yu, and H. Wiemann, “A primer on bandwidth parts in 5G new radio,” Springer Int. Pub., pp. 357–370, 2021.
[4] J. F. Santos, D. D. S. Brilhante, J. F. De Rezende, N. Marchetti, M. Ruffini, and L. A. DaSilva, “Optimal embedding of heterogeneous RAN slices for secure and technology-agnostic RANaaS,” IEEE Trans. Netw. Service Manag., vol. 19, no. 4, pp. 5284–5298, Dec. 2022.
[5] 3GPP, “5G; management and orchestration; concepts, use cases and requirements,” 3GPP TS 28.530 version 16.2.0 Release 16, Aug. 2020. [Online]. Available: https://www.3gpp.org/ftp/Specs/html-info/28530-162599-28530v16.0.00_60ts_28530v16.020p0.pdf
[6] H. P. Phyu, D. Naboulsi, and R. Stancioiu, “Machine learning in network slicing: A survey,” IEEE Access, vol. 11, pp. 39123–39153, 2023.
[7] I. Ainfoabi, T. Taleb, K. Samadani, A. Ksentini, and H. Finck, “Network slicing and softwarization: A survey on principles, enabling technologies, and solutions,” IEEE Commun. Surveys Tut., vol. 20, no. 3, pp. 2429–2453, Third Quarter, 2018.
[8] 3GPP, “5G: NR; physical layer procedures for data,” 3GPP TS 38.214 version 16.5.0 Release 16, Apr. 2021. [Online]. Available: https://www.3gpp.org/ftp/Specs/html-info/38214-16500p.pdf
[9] M. Richard, J. Balasubramanian, J. Serrat, and J.-L. Gorricho, “Resource slicing in virtual wireless networks: A survey,” IEEE Trans. Netw. Service Manag., vol. 13, no. 3, pp. 462–476, Sep. 2016.
[10] R. Xu et al., “Resource allocation for network slicing in 5G telecommunications network: A survey of principles and models,” IEEE Netw., vol. 33, no. 6, pp. 172–179, Nov./Dec. 2019.
[11] Y. Azimi, S. Yousefi, H. Kalbkhani, and T. Konz, “Energy-efficient deep reinforcement learning assisted resource allocation for 5G-RAN slicing,” IEEE Trans. Veh. Technol., vol. 71, no. 1, pp. 856–871, Jan. 2022.
[12] A. J. Gonzalez et al., “The isolation concept in the 5G network slicing,” in Proc. Eus. Conf. Netw. Commun., 2022, pp. 12–16.
[13] D. Marabissi and R. Fantacci, “Highly flexible RAN slicing approach to manage isolation, priority, efficiency,” IEEE Access, vol. 7, pp. 70190–70142, 2019.
[14] S. D’Oro, L. Bonati, F. Restuccia, and T. Melodia, “Coordinated 5G network slicing: How constructive interference can boost network throughput,” IEEE/ACM Trans. Netw., vol. 29, no. 4, pp. 1881–1894, Aug. 2021.
[15] M. Zambianco and G. Verticale, “Interference minimization in 5G physical-layer network slicing,” IEEE Trans. Commun., vol. 68, no. 7, pp. 4554–4564, Jul. 2020.
[16] S. D’Oro, F. Restuccia, A. Talamonti, and T. Melodia, “The slice is served: Enforcing radio access network slicing in virtualized 5G systems,” in Proc. IEEE Conf. Comput. Commun., 2019, pp. 442–452.
[17] S. Zarandi and H. Tabassum, “Delay minimization in sliced multi-cell mobile edge computing (MEC) systems,” IEEE Commun. Lett., vol. 25, no. 6, pp. 1964–1968, Jun. 2021.
[18] Y. K. Tun, N. H. Tran, D. T. Ngo, S. R. Pandey, Z. Han, and C. S. Hong, “Wireless network slicing: Generalized Kelly mechanism-based resource allocation,” IEEE J. Sel. Areas Commun., vol. 37, no. 8, pp. 1794–1807, Aug. 2019.
[19] N. Raveendran et al., “Cyclic three-sided matching game inspired wireless network virtualization,” IEEE Trans. Mobile Comput., vol. 20, no. 2, pp. 416–426, Feb. 2021.
[20] J. Hu, Z. Zheng, B. Di, and L. Song, “Multi-layer radio network slicing for heterogeneous communication systems,” IEEE Trans. Netw. Sci. Eng., vol. 7, no. 4, pp. 2378–2391, Fourth Quarter, 2020.
[21] S. O. Oladejo, S. O. Ekwe, and L. A. Akinyemi, “Multi-tier multi-domain network slicing: A resource allocation perspective,” in Proc. IEEE AfrICON, 2021, pp. 1–6.

O. Al-Khatib, W. Hardjawa, and B. Vucetic, “Wireless networks virtualisation: Traffic modeling and spectrum sharing,” in Proc. IEEE Int. Conf. Commun., 2015, pp. 5859–5864.

Y. K. Tun, N. H. Tran, D. T. Ngo, S. R. Pandey, Z. Han, and C. S. Hong, “Wireless network slicing: Generalized Kelly mechanism-based resource allocation,” IEEE J. Sel. Areas Commun., 2017, pp. 1794–1807.

S. D’Oro, L. Bonati, F. Restuccia, and T. Melodia, “Coordination of 5G network slicing: How constructive interference can boost network throughput,” IEEE/ACM Trans. Netw., 2020, pp. 1881–1894.

M. Zambianco and G. Verticale, “Interference minimization in 5G physical-layer network slicing,” IEEE Trans. Commun., 2020, pp. 4554–4564.

O. Al-Khatib, W. Hardjawa, and B. Vucetic, “Spectrum sharing in multi-tenant 5G cellular networks: Modeling and planning,” IEEE Access, vol. 7, pp. 1602–1616, 2019.

L. Feng, Y. Zi, W. Li, Z. Zhou, P. Yu, and M. Kadoch, “Dynamic resource allocation with RAN slicing and scheduling for uRLLC and eMBB hybrid services,” IEEE Access, vol. 8, pp. 34538–34551, 2020.

B. Han, V. Sciancalepore, D. Feng, C. Costa-Perez, and H. D. Schotten, “A utility-driven network allocation control solution for network slicing,” in Proc. IEEE Conf. Comput. Commun., 2019, pp. 55–63.

B. Han, V. Sciancalepore, C. Costa-Perez, D. Feng, and H. D. Schotten, “Multiservice-based network slicing orchestration with impatient tenants,” IEEE Trans. Wireless Commun., vol. 19, no. 7, pp. 5010–5024, Jul. 2020.

V. N. Ha, T. T. Nguyen, L. B. Le, and J.-F. Frigon, “Admission control and network slicing for multi-numerology 5G wireless networks,” IEEE Netw. Lett., vol. 2, no. 1, pp. 5–9, Mar. 2020.

H. Klessig, D. Öhmann, A. J. Fehske, and G. P. Fettweis, “A performance evaluation framework for interference-coupled cellular data networks,” IEEE Trans. Wireless Commun., vol. 15, no. 2, pp. 938–950, Feb. 2016.

A. Fehske, H. Klessig, J. Voigt, and G. Fettweis, “Flow-level models for capacity planning and management in interference-coupled wireless data networks,” IEEE Commun. Mag., vol. 52, no. 2, pp. 164–171, Feb. 2014.

T. Rappaport, Wireless Communications: Principles and Practice, 2nd ed. Englewood Cliffs, NJ, USA: Prentice Hall PTR, 2002.

P. Mogensen et al., “LTE capacity compared to the Shannon bound,” in Proc. IEEE 65th Veh. Technol. Conf., 2007, pp. 1234–1238.

J. Harri, F. Filiali, and C. Bonnet, “Mobility models for vehicular ad hoc networks: A survey and taxonomy,” IEEE Commun. Surveys Tuts., vol. 11, no. 4, pp. 19–41, Fourth Quarter, 2009.

F. Ashitani, J. Salehi, and M. Aref, “Mobility modeling and analytical solution for spatial traffic distribution in wireless multimedia networks,” IEEE J. Sel. Areas Commun., vol. 21, no. 10, pp. 1699–1709, Dec. 2003.

X. Chao, M. Miyazawa, and M. Pinedo, Queueing Networks: Customers, Signals and Product Form Solutions, Chichester, U.K.: Wiley, 1999.

G. Fiebig, A. Gavious, and E. Solan, “Averaging principle for second-order approximation of heterogeneous models with homogeneous models,” in Proc. Nat. Acad. Sci. USA, vol. 109, no. 48, pp. 19545–19550, 2012.

S. A. Hashemian and F. Ashitani, “Technical report: Analytical modeling and improvement of interference-coupled RAN slicing,” 2022, arXiv:2208.07045.