Optimizing Dynamic RAN Slicing in Programmable 5G Networks

Optimizing Dynamic RAN Slicing in Programmable 5G Networks

Arled Papa*, Markus Klügel*, Leonardo Goratti†, Tinku Rasheed†, Wolfgang Kellerer*
*Chair of Communication Networks, Technical University of Munich
†Zodiac Inflight Innovations, Weßling Germany
*arled.papa@tum.de, *markus.klügel@tum.de, †leonardo.goratti@zii.aero, †tinku.rasheed@zii.aero, *wolfgang.kellerer@tum.de

***Abstract*—Network slicing is envisioned as a tool for 5G networks to provide network flexibility and isolation among different logical networks. While network slicing is well investigated in the fixed-network side, in the Radio Access Network (RAN) there remain challenging problems, which originate mainly from the stochastic nature of the wireless channels and complex resource coupling between slices. In this work, we investigate a network slicing problem for the downlink RAN of a cellular network. Our target is the reduction of resource usage while guaranteeing slice isolation and simultaneously accounting for each slice's average rate and delay requirements. We tackle the problem with a Lyapunov optimization approach, leading to a simple resource assignment procedure that we can prove to achieve isolation while satisfying all slice guarantees. The proposed procedure leads to a functional split, where resources are scheduled within each slice by a slice manager, while a Software-Defined RAN (SD-RAN) controller dynamically re-assigns resources to each slice. We verify our approach through extensive simulations and provide insight on how to fine-tune available system parameters.**

***Index Terms*—Network Slicing, Slice Isolation, 5G, RAN, Lyapunov Optimization, QoS**

I. Introduction

In the context of fifth generation (5G) networks, the concepts of network virtualization and programmability are receiving increased attention. By leveraging these techniques, future networks are expected to be more adaptive and cost-efficient. A concept to deliver virtualization and programmability is network slicing [1], which allows sharing the infrastructure and thus enables the coexistence of several tenants as different *slices* of the network. However, coexistence brings the requirement of isolation among the created network slices, which denotes the decoupling of slices from each other's state of operation. One of the main challenges of network slicing, the ability to operate in an isolated manner and stay unaffected by other slices, is a crucial, yet non-trivial task [2], [3].

Network slicing has been targeted both on the Core Network (CN) [4], [5] and Radio Access Network (RAN) [6], [7]. However, while network slicing on CN is well investigated, in RAN it still remains an interesting and challenging problem due to the more complex resource coupling and the stochastic nature of the wireless channel. An intuitive way of providing a network slice, is the allocation of time-frequency resources in a persistent fashion. However, such a static approach does not account for the stochastic nature of the wireless channels and comes at the cost of loosing diversity gains. Thus, dynamic re-assignment of slice resources has been proposed in literature, up to the extreme case that they are re-assigned in each frame. This helps adapting to dynamic changes of the wireless channel and leveraging diversity gains, however, also renders resource assignment more complex.

To the best of our knowledge, we are the first to account both for isolation on the wireless resource level and Quality-of-Service (QoS) performance, while enabling diversity gains. The main contribution of this paper is to formulate and solve an optimization problem for resource usage reduction by dynamically re-assigning resources to slices, while providing each slice a requested average rate and delay in an isolated fashion. The problem is solved by Lyapunov optimization [8], leading to a slim, yet optimal implementation. Extensive simulations are conveyed to show the effectiveness of our model. Considering a representative 5G use-case with high user density, we verify our model in an aircraft in-cabin scenario. Nonetheless no particular assumptions are introduced in the problem formulation, such that the derived approach is applicable to any cellular system. The simulation results show the validity of our approach, where the QoS requirements for each slice are achieved and traffic changes in one slice do not affect the other slices.

The rest of the paper is structured as follows: We provide a brief literature review on RAN resource slicing in Section II. Section III introduces the overall system model and the problem formulation. The Lyapunov technique used to relax and solve the problem is elaborated in Section IV. Section V presents the evaluation of the proposed approach and discusses the results. Finally, the paper is concluded with discussions and an overview of the proposed scheme in Section VI.

II. Related Work

Network slicing in RAN has attracted increased attention in academia. From the architectural perspective, the main solutions rely on softwarization techniques, such as Software Defined Networking (SDN), to facilitate network slicing [9], [10]. As aforementioned, major challenges in wireless network slicing lie in the tasks of efficient wireless resource allocation and slice isolation. To this end, most of the existing contributions focus on the efficient resource allocation among coexisting Mobile Virtual Network Operators (MVNO) [11], or Service Providers (SPs) [12], which share the physical infrastructure. The isolation effect is in general tackled by simply adding constraints regarding the number of resources assigned to each slice. While this enables isolation on the wireless resource level, it does not take into account any QoS performance and hence is prone to loss in diversity gains.

Another interesting aspect of network slicing is investigated in [13]. The authors consider the effect of multiplexing gain and formulate the resource allocation problem as a knapsack problem. Given the shape of a slice request, this approach maximizes the number of accepted slices and therefore increases the resource utilization. However, again isolation is provided only on a wireless resource basis.

Similarly to our approach there exist works which leverage the Lyapunov technique to efficiently re-assign the wireless resources to slices. The authors in [14] consider the Lyapunov technique for enabling network slicing with focus on energy efficiency, however, without guaranteeing slice isolation. In [15], a similar problem is formulated as a throughput maximization problem and relaxed using the Lyapunov approach to efficiently assign resources and power to the slices. Although the authors claim to provide isolation, it is done by the use of a resource ratio parameter for distributing the resources among the slices. Thus, the isolation is again on resource level and no QoS guarantees are given. Finally, the authors in [16] introduce two type of slices, namely delay and capacity critical ones, and define their QoS requirements. Employing the Lyapunov approach they distribute resources to the slices for power minimization. Isolation is investigated in this case, however, the implementation is a bit unclear and therefore the effectiveness of the approach remains open.

III. System Model and Problem Formulation

We consider the downlink scenario of a cellular system with a single small cell Base Station (BS) of a single Infrastructure Provider (InP). The system is time slotted, where the slot duration corresponds to a LTE Transmission Time Interval (TTI) of 1 ms. The BS serves a set $\mathcal{N}$ of N users at a given slot, which are associated to a set $\mathcal{S}$ of S slices. Considering a specific bandwidth, a set $\mathcal{R}$ of R Physical Resource Blocks (PRBs) are available. To support the 5G RAN concepts, the management and orchestration of the system is driven by means of SDN. To this aim, the architecture implies the introduction of a Software-Defined Radio Access Network (SD-RAN) controller, which is responsible for collecting information and requests about the slices using a Northbound Application Programmable Interface (API), as well as the Channel Quality Indicator (CQI) from the BS. The SD-RAN controller creates an instance called *slice manager* for each deployed network slice and dynamically re-assigns resources to each slice by giving its manager the right to allocate it to a user. The slice manager abstracts the slices's users from the controller and coordinates scheduling decisions within the slice. We adopt the principle of FlexRAN [10], and incorporate further functions to enable the proposed slicing approach. The proposed 5G RAN architecture is illustrated in Fig. 1.

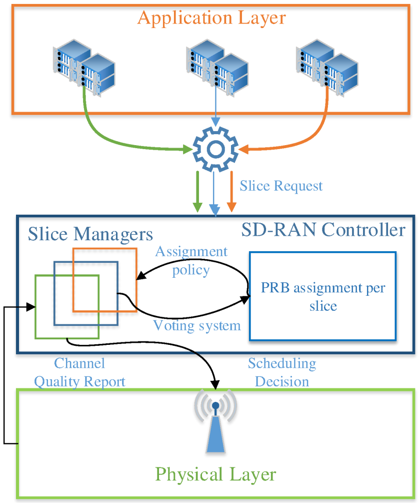

Fig. 1. SD-RAN controller architecture. The controller uses the northbound API to get requests about the slices and also collects CQI information from the BS for the resource allocation. Each slice is controlled by a slice manager which abstracts the users from the controller and coordinates the scheduling decisions.

Let $h_{i,j}^t$ denote the channel gain for user i on PRB j in slot t. We assume that the channel gain remains constant within the slot duration but may vary between slots. The variable $p_{i,j}^t$ is the transmit power between the BS and user i on PRB j in slot t. The variable σ^2 is the thermal noise on each PRB. Moreover, the PRB bandwidth is denoted by B, whereas $w_{i,j}^t$ is a decision variable which indicates whether user i is associated with PRB j in slot t or not. Assuming a uniform power allocation over the PRBs, the user achievable data rate can be calculated as:

$$r_i^t = \sum_{j=1}^{R} w_{i,j}^t r_{i,j}^t \tag{1}$$

where

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

is the rate achievable by user i on PRB j in slot t.

In this work we consider two type of slices. The first type corresponds to capacity-critical slices, which require a minimum expected throughput. Additionally, the second type corresponds to delay-critical slices, where a maximum expected delay must be fulfilled. Each slice sustains a queue for the incoming slice traffic, which has a backlog of U_s^t. The arrival traffic per slot is modeled by a random variable λ_s^t with mean value λ_s, which denotes the amount of traffic in Kb/slot. The arrived traffic is accepted portion-wise after an admission control, with the admitted traffic in each slot denoted by the variable α_s^t. Further, the admitted traffic is then served by a slice rate expressed with the variable r_s^t. Correspondingly, the slice queue backlog evolves from slot to slot by the equation:

$$U_s^{t+1} = \max \left\{ U_s^t + \alpha_s^t - r_s^t, 0 \right\} \quad \forall s \in \mathcal{S} \tag{3}$$

Utilizing the expected queue backlog, the average delay of the queue can be calculated by Little's theorem to be $D_s = U_s/r_s$ [17].

The SD-RAN controller collects the requests from the slices, namely $\underline{C}_s$, which denotes the minimum required throughput, as well as D_s, which is the maximum tolerable delay. Moreover, each slice specifies a rate policy $\gamma_s = [..., \gamma_{s,i}, ...]$ to the controller, that provides information regarding the relative share of slice's minimum throughput among the slice's users. Finally, the SD-RAN controller decides for a maximum throughput allowed per slice, namely $\overline{C}_s$, depending on the wireless channel qualities. It is this maximum throughput that ensures rate isolation among the slices, as it prevents them from overloading the network. An intuitive way of selecting $\overline{C}_s$ is by assigning a portion of the channel capacity in a round robin fashion to each slice. However, in general the selection of $\overline{C}_s$ introduces a new optimization problem that is out of the scope of this work.

A. Problem Formulation

We model the resource allocation problem as a resource use minimization problem, while guaranteeing the QoS objectives for the slices, as well as slice isolation. The optimization problem is formulated as follows:

$$\min_{w_{i,j}^t, \alpha_s^t} \lim_{T\to\infty} \frac{1}{T} \sum_{t=0}^{T-1} \sum_{i=1}^{N^t} \sum_{j=1}^{R} w_{i,j}^t \tag{4a}$$

$$\text{s.t.} \quad \lim_{T\to\infty} \frac{1}{T} \sum_{t=0}^{T-1} \alpha_s^t \leq \lim_{T\to\infty} \frac{1}{T} \sum_{t=0}^{T-1} r_s^t \quad \forall s \in \mathcal{S} \tag{4b}$$

$$\lim_{T\to\infty} \frac{1}{T} \sum_{t=0}^{T-1} U_s^t \leq D_s \lim_{T\to\infty} \frac{1}{T} \sum_{t=0}^{T-1} r_s^t \quad \forall s \in \mathcal{S} \tag{4c}$$

$$\lim_{T\to\infty} \frac{1}{T} \sum_{t=0}^{T-1} r_i^t \geq \gamma_{s,i} \underline{C}_s \quad \forall i \in \mathcal{N}_s, \forall s \in \mathcal{S} \tag{4d}$$

$$\lim_{T\to\infty} \frac{1}{T} \sum_{t=0}^{T-1} r_s^t \leq \overline{C}_s \quad \forall s \in \mathcal{S} \tag{4e}$$

$$\sum_{s=1}^{S} \sum_{i=1}^{\mathcal{N}_s} w_{i,j}^t \leq 1 \quad \forall j \in \mathcal{R}, \forall t \in T \tag{4f}$$

$$0 \leq \alpha_s^t \leq \lambda_s^t \quad \forall s \in \mathcal{S}, \forall t \in T \tag{4g}$$

$$w_{i,j}^t \in \{0, 1\} \quad \forall s, i, j, t \tag{4h}$$

The minimization problem given in (4a) targets the allocation of the minimum number of resources to satisfy the constraints. Constraint (4b) assures that the average throughput of each slice is greater than the average admitted traffic of each slice. Moreover (4c) guarantees that the average delay will not surpass a maximum set delay D_s, where U_s^t represents the queue backlog of slice s in slot t. Furthermore, constraint (4d) guarantees that the average throughput of each slice user is greater than the share of minimum throughput defined by policy γ_s, while constraint (4e) assures that the average throughput of each slice will not exceed a maximum throughput set in order to provide the isolation among the slices. Additionally, constraint (4f) defines the orthogonality constraint for each PRB and finally, (4g) suggests that the total admitted traffic cannot exceed the arrival traffic in any slot.

IV. Resource allocation with Lyapunov Optimization

The aforementioned problem is a stochastic optimization problem, which is hard to solve due to the unknown channel variations and incoming traffic over the slots. Even with full knowledge, the complexity increases when a large T is considered. Therefore, linear programming techniques can not be used to provide a solution to the problem. Lyapunov optimization [8] is used to solve the problem defined in (4a). The constraints are transformed into virtual queues, which evolve over slots according to a process given as:

$$U_s^{t+1} = \max\left\{U_s^t + \alpha_s^t - r_s^t, 0\right\} \quad \forall s \in \mathcal{S} \tag{5}$$

$$G_s^{t+1} = \max\left\{G_s^t + U_s^t - D_s r_s^t, 0\right\} \quad \forall s \in \mathcal{S} \tag{6}$$

$$K_i^{t+1} = \max\left\{K_i^t + \gamma_{s,i} \underline{C}_s - r_i^t, 0\right\} \quad \forall i \in \mathcal{N}_s, \forall s \in \mathcal{S} \tag{7}$$

$$J_s^{t+1} = \max\left\{J_s^t + r_s^t - \overline{C}_s, 0\right\} \quad \forall s \in \mathcal{S} \tag{8}$$

The virtual queue U_s^t expresses (4b) and in our scenario corresponds to the physical queue denoted by (3), whereas virtual queues G_s^t, K_i^t and J_s^t represent the constraints presented in (4c)-(4e) respectively. Defining $K_s^t = \left[..., K_{s,i}^t, ...\right]$, the overall system queue state is denoted by $\Theta^t = \{\Theta_s^t\}$, where $\Theta_s^t = [G_s^t, U_s^t, K_s^t, J_s^t]$ is the state of slice s. We now define the quadratic Lyapunov function as:

$$\begin{aligned} L(\Theta^t) = \frac{1}{2} \sum_{s\in\mathcal{S}} (J_s^t)^2 + \frac{1}{2} \sum_{s\in\mathcal{S}} (U_s^t)^2 + \frac{1}{2} \sum_{s\in\mathcal{S}} (G_s^t)^2 \\ + \frac{1}{2} \sum_{s\in\mathcal{S}} \sum_{i\in\mathcal{N}_s} (K_i^t)^2. \end{aligned} \tag{9}$$

According to Lyapunov optimization, the Lyapunov function is a scalar metric to measure the *queue congestion* state. This function is always non-negative and equals to zero only in the case when Θ^t = 0 (i.e., the queues have reached their stability). When $L(\Theta^t)$ is small, the virtual queues are small, therefore a low queue congestion and a high stability is implied. On the other hand, a large $L(\Theta^t)$ indicates high congestion and low system stability. In any case, all constraints are satisfied if, and only if, the infinite time-horizon limit of $L(\Theta^t)$ is bounded, i.e., $\lim_{T\to\infty} 1/T \sum_{t=0}^{T-1} L(\Theta^t) < \infty$. Thus, in order to cater for stability, the Lyapunov drift [8] is introduced and denoted by:

$$\Delta L(\Theta^t) = \mathbb{E}\left\{L(\Theta^{t+1}) - L(\Theta^t) | \Theta^t\right\} \tag{10}$$

Algorithm 1 Admission Control

```
Collect the status of the admission control queue length $U_s^t \quad \forall s \in S$
if $U_s^t = 0$ then
    Admit $\alpha_s^t = \lambda_s^t$ Kb to minimize (13)
else
    Admit $\alpha_s^t = 0$ Kb to minimize (13)
end if
```

Algorithm 2 Dynamic resource assignment

```
Initialization:
Choose $V > 0$;
Set $G_s^0 = J_s^0 = U_s^0 = K_i^0 := 0 \; \forall i \in \mathcal{N}_s \; \forall s \in S$
for $t := 1, 2, 3, ..., \infty$ do
    Slice Manager:
    Calculate $v_{s,j}^t = \max_{i \in \mathcal{N}_s} \left\{ r_{i,j}^t \left[ G_s^t - J_s^t + U_s^t + K_i^t \right]^+ \right\}$
    Store $n_{s,j}^t = \arg\max_{i \in \mathcal{N}_s} \left\{ r_{i,j}^t \left[ G_s^t - J_s^t + U_s^t + K_i^t \right]^+ \right\}$
    Communicate $v_s^t = \left\{ v_{s,j}^t \right\}$ to SD-RAN Controller
    SD-RAN Controller:
    Assign resource $j$ to slice $s_j^t = \arg\max_{s \in \mathcal{S}} \{ v_s^t \geq V \}$
    Slice Manager:
    If resource $j$ was won, assign it to $n_{s,j}^t$.
    Update $G_s^t, U_s^t, K_i^t, J_s^t$, based on (6), (5), (7), (8).
end for
```

Based on the Lyapunov technique, the objective is to minimize an infinite bound on the Lyapunov drift-plus-penalty in each time slot, where the drift-plus-penalty is expressed as:

$$\Delta L(\Theta^t) + V\mathbb{E}\left\{ \sum_{i=1}^{N^t} \sum_{j=1}^{\mathcal{R}} w_{i,j}^t | \Theta^t \right\}. \tag{11}$$

$V \geq 0$ is a design parameter to determine how much emphasis is given to the resource minimization, compared to the queue stability. The drift-plus-penalty expression given in (11) satisfies the expression stated in:

$$\begin{aligned} &\Delta L(\Theta^t) + V\mathbb{E}\left\{ \sum_{i=1}^{N^t} \sum_{j=1}^{\mathcal{R}} w_{i,j}^t \right\} \leq B + V\mathbb{E}\left\{ \sum_{i=1}^{N^t} \sum_{j=1}^{\mathcal{R}} w_{i,j}^t \right\} \\ &+ \mathbb{E}\left\{ \sum_{s \in \mathcal{S}} \left(r_s^t - \overline{C}_s \right) J_s^t + \sum_{s \in \mathcal{S}} \left(\alpha_s^t - r_s^t \right) U_s^t \right\} \\ &+ \mathbb{E}\left\{ \sum_{s \in \mathcal{S}} \left(U_s^t - D_s r_s^t \right) G_s^t + \sum_{i \in \mathcal{N}_s} \left(\gamma_{s,i} \underline{C}_s - r_i^t \right) K_i^t \right\} \end{aligned} \tag{12}$$

where $B > 0$ is a constant. Due to lack of space we refer the readers to [8, Lemma 4.6] for the full proof of the above expression. In order to solve the stochastic problem given in (4a) we need to minimize the right-hand-side of the expression given in (12) in every time slot, subject to the constraints elaborated in (4f)-(4h).

As we can see from (12), the statement on the right-hand-side is convex and therefore we can find its minimum efficiently. In order to do so, we calculate the derivative with respect to the two decision variables, namely resource allocation $w_{i,j}^t$ and admission control α_s^t, and accordingly distribute the resources:

$$\frac{\partial L}{\partial \alpha_s^t} = U_s^t \geq 0, \tag{13}$$

$$\frac{\partial L}{\partial w_{i,j}^t} = V - r_{i,j}^t \left(G_s^t - J_s^t + U_s^t + K_i^t \right). \tag{14}$$

According to (13), we derive with respect to the variable of the admitted traffic of each slice. As minimization is the target, the result is that $\alpha_s^t = 0$, (i.e., traffic should not be accepted), when $U_s^t > 0$, while it may be accepted arbitrarily if $U_s^t = 0$. This induces an admission control procedure that we state in Alg. 1. In order to apply the dynamic resource allocation, due to the orthogonality constraint (4f), each PRB should optimally be given to the user with maximum value of $r_{i,j}^t \left(G_s^t - J_s^t + U_s^t + K_i^t \right)$, as long as it is larger than V, while it should be not assigned at all if all values are lower than V.

The optimization is implemented by a two-stage bidding system, as shown in Alg. 2: Each slice manager maintains the queue states $\Theta_s^t = [G_s^t, U_s^t, K_s^t, J_s^t]$ of its slice and updates them from slot to slot. Further, it calculates the value

$$v_{s,j}^t = \max_{i \in \mathcal{N}_s} \left\{ r_{i,j}^t \left[G_s^t - J_s^t + U_s^t + K_i^t \right]^+ \right\}, \tag{15}$$

that each PRB is worth to it, stores the associated user $n_{s,j}^t \in \mathcal{N}_s$ and communicates $v_{s,j}^t$ to the SD-RAN Controller. The latter then matches the values and assigns each resource to the slice manager to which it is of most value, or to none of it is smaller than V. The slice manager then schedules the resource internally to the user which produced this value.

V. Evaluation and Analysis

In the evaluation part, the effectiveness of the proposed method is investigated with respect to satisfying the QoS requirements, while also providing isolation guarantees. As stated earlier in the paper, the design parameter V of the Lyapunov technique introduces the trade-off between the resource utilization and the queue stability convergence time, which defines the time it takes for the optimization to satisfy the constraints. Therefore, the first analysis consists of determining the correct value of V for the given scenario. All the simulations are repeated 100 times and the confidence intervals are derived in order to correctly evaluate the performance. Considering in-aircraft communication as a representative use-case of 5G networks, we evaluate an in-cabin scenario, however due to non specific assumptions introduced in our approach the derived model is applicable to any cellular system. The simulation parameters are given in Table I. A Boeing B737-400 is considered according to [18], where 156 passengers are distributed into 26 rows, each of which has 6 seats.

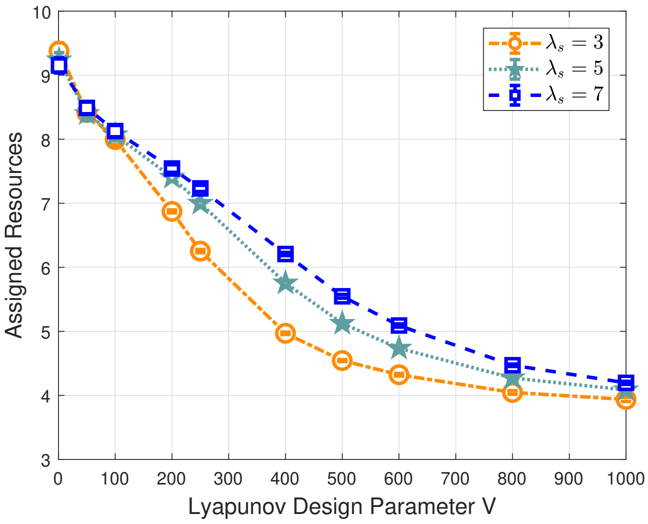

Fig. 2. Number of assigned resources with respect to the design parameter V given a maximum threshold $\overline{C}_s$ of 7 Kb/slot per each slice, evaluated for various traffic arrivals.

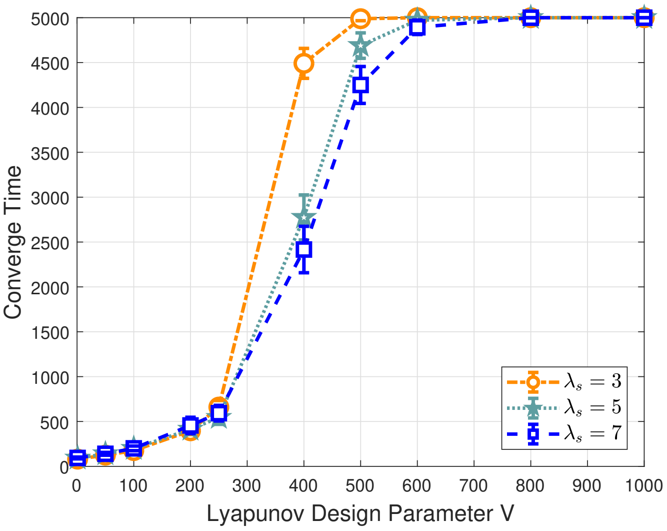

Fig. 3. Convergence time of the optimization with respect to the design parameter V given a maximum threshold $\overline{C}_s$ of 7 Kb/slot per each slice, evaluated for various traffic arrivals.

Furthermore, a single small cell BS is positioned in the middle of the aircraft and it contains a bi-directional antenna. The used channel model corresponds to the work in [19], where real measurements were conducted in a cabin in order to acquire information regarding the path loss exponent, free space loss, slow fading and frequency selective distribution functions.

TABLE I
SIMULATION PARAMETERS

System bandwidth (B)	5 MHz
BS antenna horizontal/vertical	$70°/10°$ beam width
BS antenna down-tilt	$15°$
Number of PRBs	25
Transmission power	-10 dBm
Shadowing	Gaussian zero-mean with 4.8 dB standard deviation (σ_L)
Free space loss at d_r	37.5 dB
Reference d_r	1 m
Pathloss exponent (n)	2.6
Multipath	Rice distribution with -1.4 dB mean and K-factor 8.1 dB
Path loss in dB at distance d	$PL = F_{d_r} + 10n\log(\frac{d}{d_r}) + X(0, \sigma_L)$
Number of Slices	2
Number of users per slice	5
Minimum throughput required per slice	4 Kb/slot
Delay-critical slice maximum tolerable delay	10 ms
Maximum allowed throughput per slice	7 Kb/slot
Number of slots	5000

Fig. 2, illustrates the number of PRBs needed to satisfy the QoS requirements of the slices depending on different design parameter values and traffic demands. As denoted by Fig. 2, the number of the assigned resources increases with the incoming traffic λ_s and decreases with the design parameter value V. In any case, the algorithm manages to satisfy the arrivals, as long as they do not overload the network. However, the larger V is, the closer to the optimal resource utilization the algorithm is, as the number of assigned PRBs to achieve the requirements decreases.

Fig. 3 illustrates the queue stability convergence time with respect to various arrivals and values of V. As depicted, the convergence time increases with the design parameter value V as expected, since the method assigns resources less frequently and therefore a larger time is needed for the queues to reach a stable state. Interestingly, the larger the traffic arrival is, the lower the convergence time. This is due to the fact that the queue state Θ_s increases faster for larger arrivals and therefore the queues reach the required steady-state magnitude faster.

Given the results from Fig. 2 and Fig. 3, a trade-off regarding the queue stability convergence time and the resource utilization must be decided, such that the system can be evaluated correctly. Again, we stress that regardless of the chosen Lyapunov design parameter V, the constraints are not violated, only the time required to satisfy them increases. According to the observations from the aforementioned figures, a suitable design parameter value is $V = 250$, which is therefore considered for all the upcoming evaluations of this paper, if not stated differently.

A. Isolation Performance

Isolation performance in network slicing can be evaluated by the impact that changes happening in the other slices, such as the number of users or traffic demands, have on a given slice. In order to correctly measure and evaluate the isolation performance given by the introduced model, we consider two scenarios, 1) Given a traffic demand for the delay-critical slice, the traffic demand of the throughput-critical slice is varied and the influence on the achieved throughput is evaluated. 2) Similarly, given a traffic demand for the throughput-critical

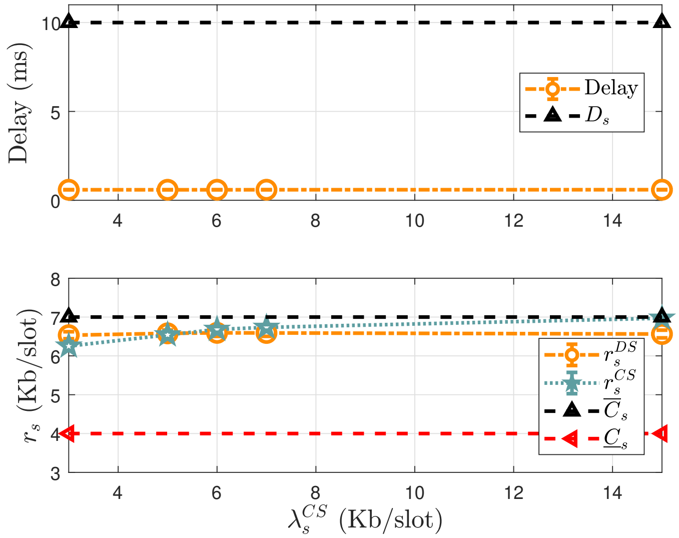

Fig. 4. Average throughput of the delay-critical and throughput-critical slices as well as average delay of the delay-critical slice with respect to various traffic demands of the throughput-critical slice given a 5 Kb/slot arrival of the delay-critical slice and $V = 250$.

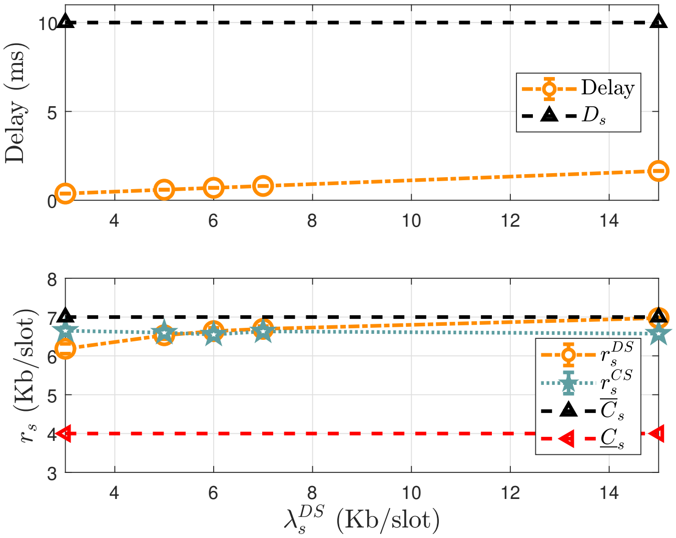

Fig. 5. Average throughput of the throughput-critical and delay-critical slice as well as average delay of the delay-critical slice with respect to various traffic demands of the delay-critical slice given a 5 Kb/slot arrival of the throughput-critical slice and $V = 250$.

slice, the traffic demand of the delay-critical slice is varied and the influence on the maximum delay is evaluated.

Fig. 4 illustrates the delay-critical slice achieved throughput, referred to as r_s^{DS}, for different variations of the throughput-critical slice traffic arrivals, λ_s^{CS}. Given an incoming traffic demand of 5 Kb/slot for the delay-critical slice, the incoming traffic arrival of the throughput-critical slice varies between 3-15 Kb/slot, where $\lambda_s^{CS} = 3$ is lower than $\underline{C}_s$, whereas $\lambda_s^{CS} = 15$ is larger than $\overline{C}_s$. As denoted by Fig. 4 the achieved throughput of the delay-critical slice, namely r_s^{DS} is not affected by the change of the traffic arrival of the other slice. Moreover, the average delay for the delay-critical slice is shown to be unaffected and remains below the requirement D_s. As per the throughput of the throughput-critical slice referred to as r_s^{CS}, it remains larger than the minimum requirement $\underline{C}_s$ and regardless of the traffic demand does not exceed the maximum threshold $\overline{C}_s$. Therefore, the isolation among the two slices can be guaranteed.

Similarly, Fig. 5 illustrates the second scenario, where the traffic arrival of the throughput-critical slice is fixed to 5 Kb/slot, whereas the traffic arrival of the delay-critical slice varies between 3-15 Kb/slot. Even in this case, as shown the traffic variation of the delay-critical slice referred to as λ_s^{DS} does not affect at all the achieved throughput of the throughput-critical slice r_s^{CS}. Moreover, the minimum required throughput is achieved for the delay-critical slice, and further, the maximum threshold is not exceeded regardless of the traffic arrival. Finally, the average delay does not exceed the maximum tolerable delay D_s. These results indicate the effectiveness of our approach in the ability to achieve the QoS requirements, while guaranteeing a smooth functionality and isolation among the slices.

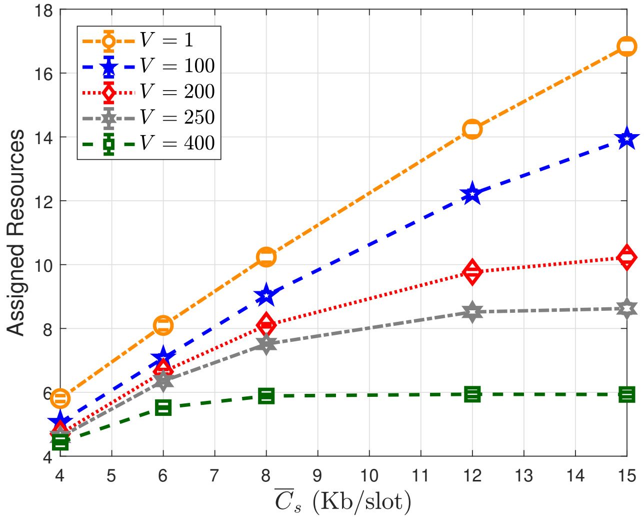

Fig. 6. Number of assigned resources with respect to the maximum throughput threshold $\overline{C}_s$ per slice given a traffic arrival of 5 Kb/slot per each slice.

B. *Maximum Threshold Selection*

Further investigations are conducted to demonstrate the impact of the maximum threshold on the resource utilization. As shown in Fig. 6, the resource assignment is increasing with $\overline{C}_s$. Moreover, by varying the values of the design parameter we can see that the higher the V, the more stable the system gets and the global minimum on the resource utilization is achieved. Therefore, we can conclude that the design parameter V and maximum threshold $\overline{C}_s$ are coupled and a careful selection has to be made depending on the history of the incoming traffic. Furthermore, in general the selection of a maximum threshold $\overline{C}_s$ above the network capabilities can not assure the isolation among the slices, because then traffic may be injected in a magnitude that renders the resulting problem infeasible.

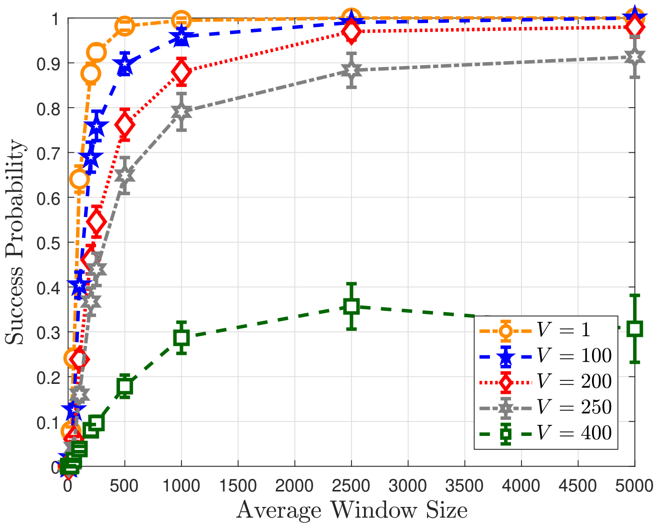

Fig. 7. Probability of achieving the QoS requirements with respect to various average window sizes for various V values given a traffic arrival of 5 Kb/slot and $\overline{C}_s$ of 7 Kb/slot per each slice.

Given the fact that the Lyapunov technique operates on expected values, instantaneous rates and delays are in fact not formally guaranteed with this approach. However, we demonstrate the probability of achieving the QoS requirements in a given, small time window. In Fig. 7, the probability of satisfying all QoS requirements within a finite, moving average window size is depicted for various design parameter values V. As shown, a lower value of V allows the algorithm to converge faster and given a window size of 500 slots it satisfies the requirements with close to 100% probability for $V = 1$. Moreover, a decreasing trend on the probability is noticed when the V parameter becomes larger. Particularly for $V = 400$, even if the average window size corresponds to the entire simulation time of 5000 slots, the probability of fulfilling the results is low. The reason of this behavior relates to the emphasis on the resource utilization, when a larger V is selected. Consequently, we observe the importance of the trade-off between optimality and the convergence time when selecting the Lyapunov parameter V.

VI. Conclusions

In this paper we investigated the network slicing problem in a Software-Defined Radio Access Network (SD-RAN). We made use of the existing state-of-art SD-RAN approaches and proposed a network slicing algorithm that leverages network programmability and dynamically re-assigns resources to the slices, as well as within the slices. We formulated a dynamic re-assignment resource optimization problem to minimize the resource usage for a given set of QoS requirements. The Lyapunov technique was used to solve the problem due to the stochastic nature of the wireless channels and unknown incoming traffic over time. We evaluated the performance of our approach while considering a scenario of two slices with heterogeneous requirements, namely delay-critical and throughput-critical. Moreover, we provided insights on how to select important system parameters that affect the resulting performance. The results show the effectiveness of the proposed technique in terms of guaranteeing the QoS requirements, providing isolation and dynamically re-allocating slice resources.

Acknowledgment

This work is supported in part by a research grant from Zodiac Inflight Innovations (TriaGnoSys GmbH), Germany.

References

[1] N. Alliance, “Description of network slicing concept,” *NGMN 5G P*, vol. 1, 2016.

[2] X. Foukas, G. Patounas, A. Elmokashfi, and M. K. Marina, “Network slicing in 5G: Survey and challenges,” *IEEE Communications Magazine*, vol. 55, no. 5, pp. 94–100, 2017.

[3] H. Zhang, N. Liu, X. Chu, K. Long, A.-H. Aghvami, and V. C. Leung, “Network slicing based 5G and future mobile networks: mobility, resource management, and challenges,” *IEEE Communications Magazine*, vol. 55, no. 8, pp. 138–145, 2017.

[4] A. Blenk, A. Basta, M. Reisslein, and W. Kellerer, “Survey on Network Virtualization Hypervisors for Software Defined Networking,” *IEEE Communications Surveys & Tutorials*, vol. 18, no. 1, pp. 655–685.

[5] Z. A. Qazi, M. Walls, A. Panda, V. Sekar, S. Ratnasamy, and S. Shenker, “A high performance packet core for next generation cellular networks,” in *Proceedings of the Conference of the ACM Special Interest Group on Data Communication*. ACM, 2017, pp. 348–361.

[6] A. Ksentini and N. Nikaein, “Toward enforcing network slicing on RAN: Flexibility and resources abstraction,” *IEEE Communications Magazine*, vol. 55, no. 6, pp. 102–108, 2017.

[7] C. Liang and F. R. Yu, “Wireless virtualization for next generation mobile cellular networks,” *IEEE wireless communications*, vol. 22, no. 1, pp. 61–69, 2015.

[8] M. J. Neely, “Stochastic network optimization with application to communication and queueing systems,” *Synthesis Lectures on Communication Networks*, vol. 3, no. 1, pp. 1–211, 2010.

[9] K. Samdanis, X. Costa-Perez, and V. Sciancalepore, “From network sharing to multi-tenancy: The 5G network slice broker,” *IEEE Communications Magazine*, vol. 54, no. 7, pp. 32–39, 2016.

[10] X. Foukas, N. Nikaein, M. M. Kassem, M. K. Marina, and K. Kontovasilis, “FlexRAN: A flexible and programmable platform for software-defined radio access networks,” in *Proceedings of the 12th International on Conference on emerging Networking EXperiments and Technologies*. ACM, 2016, pp. 427–441.

[11] R. Kokku, R. Mahindra, H. Zhang, and S. Rangarajan, “NVS: A substrate for virtualizing wireless resources in cellular networks,” *IEEE/ACM transactions on networking*, vol. 20, no. 5, pp. 1333–1346, 2012.

[12] M. I. Kamel, L. B. Le, and A. Girard, “LTE wireless network virtualization: Dynamic slicing via flexible scheduling,” in *Vehicular Technology Conference (VTC Fall), 2014 IEEE 80th*. IEEE, 2014, pp. 1–5.

[13] C.-Y. Chang, N. Nikaein, and T. Spyropoulos, “Radio access network resource slicing for flexible service execution,” in *IEEE INFOCOM 2018-IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*. IEEE, 2018, pp. 668–673.

[14] Q. Shi, L. Zhao, Y. Zhang, G. Zheng, F. R. Yu, and H.-H. Chen, “Energy-efficiency versus delay tradeoff in wireless networks virtualization,” *IEEE Transactions on Vehicular Technology*, vol. 67, no. 1, pp. 837–841, 2018.

[15] L. Yin, L. Qiu, and Z. Chen, “Throughput-Maximum Resource Provision in the OFDMA-Based Wireless Virtual Network,” in *Vehicular Technology Conference (VTC Spring), 2017 IEEE 85th*. IEEE, 2017, pp. 1–6.

[16] A. T. Z. Kasgari and W. Saad, “Stochastic optimization and control framework for 5G network slicing with effective isolation,” in *Information Sciences and Systems (CISS), 2018 52nd Annual Conference on*. IEEE, 2018, pp. 1–6.

[17] D. Xue and E. Ekici, “Delay-guaranteed cross-layer scheduling in multihop wireless networks,” *IEEE/ACM Transactions on Networking*, vol. 21, no. 6, pp. 1696–1707, 2013.

[18] The boeing, 2007. [Online]. Available: http://www.boeing.com

[19] N. Moraitis, P. Constantinou, F. P. Fontan, and P. Valtr, “Propagation measurements and comparison with EM techniques for in-cabin wireless networks,” *EURASIP Journal on Wireless Communications and Networking*, vol. 2009, p. 5, 2009.