Delphi: Computing the Maximum Achievable Throughput in SD-RAN Environments

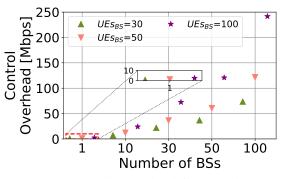Arled Papa, Polina Kutsevol, Fidan Mehmeti, *Member, IEEE,* Wolfgang Kellerer, *Senior Member, IEEE*

Abstract—Software-Defined Radio Access Networks (SD-RANs) foster the concepts of programmability and flexibility, which are vital for next generation cellular networks. However, SD-RANs render network management and orchestration very challenging. Indeed, related works indicate that when thousands of connected devices are spread across the underlying network, SD-RAN approaches with a single controller become deficient and exhibit undesired behavior. Despite this, state-of-the-art research papers lack concrete solutions and evaluations with respect to throughput predictability, where the latter is jeopardized by irregularities in the SD-RAN control plane, specifically in realistic testbeds. In order to overcome the aforementioned issues, in this work, we present `Delphi`: a novel platform that provides both analytical and experimental methods to achieve our goal, which is computing the maximum achievable throughput in SD-RAN environments. Analyzing the results provided by `Delphi`, we can capture the impact of the SD-RAN control plane on throughput. Moreover, we can design important guidelines as to which policy to choose given objectives such as throughput maximization or robustness. Providing a platform for SD-RAN evaluations based on open-source components, `Delphi` enables new avenues for research in the mobile network community. Focusing on FlexRAN SD-RAN controller for our initial results, overall, our findings show that when the number of Base Stations (BSs) and User Equipment (UEs) in the network increases beyond 5000, due to non-timely received control packets for the maximum Channel Quality Indicator (maxCQI) policy the overall throughput decreases by more than 20%.

Index Terms—SD-RAN, RAN Virtualization, 5G, Resource Management, SD-RAN Experimentation.

I. INTRODUCTION

Compared to previous generations of cellular networks, 5G manifests the flexibility to support a wide range of heterogeneous applications, control and data plane separation as well as the introduction of a service-oriented architecture [1]. To cater for the aforementioned requirements, Software-Defined Networking (SDN) has become crucial in the core network [2]–[4]. Similarly, for the Radio Access Network (RAN) counterpart, softwarization [5], [6] and virtualization [7]–[9] are envisioned as the main drivers to fulfill the anticipated 5G design principles.

FlexRAN [5] and 5G-EmPOWER [10] are the first open-source Software-Defined Radio Access Network (SD-RAN) prototypes that provide a separation of the control and data planes of traditional RANs, where the control can be shifted from Base Stations (BSs) towards an entity referred to as the **SD-RAN controller**.

A. Papa (arled.papa@tum.de), P. Kutsevol (polina.kutsevol@tum.de), F. Mehmeti (fidan.mehmeti@tum.de), and W. Kellerer (wolfgang.kellerer@tum.de) are with the Chair of Communication Networks, Technical University of Munich, Germany.

Fig. 1: SD-RAN controller overhead for control messages in Mbps, given the number of UEs and BSs.

In such a setup, the SD-RAN controller maintains a broader network view while updating its knowledge by receiving **control messages** which contain BS and User Equipment (UE) statistics. Based on these messages, **controller decisions**, concerning for instance wireless resource distribution, can be enforced at the BSs. Exploiting the wide network knowledge leads to improved performance as it allows for optimal management decisions.

Yet, the utilization of only one SD-RAN controller can become a single point of failure and a potential bottleneck. This is certainly an issue when the number of RAN elements, i.e., BSs and UEs, increases drastically. Such occurrences result from the high number of control messages traversing the network towards the controller and back to BSs, as shown in Fig. 1 for measurements performed with the FlexRAN controller. Fig. 1 shows the control overhead as a function of the number of UEs and BSs, which increases faster than linearly with the number of network elements. Consequently, for a controller with limited capacity, this can lead to delays in the transmission of the controller decisions toward BSs.

Recently, the trend of RAN softwarization has become realistic as industries such as Rakuten [11], and ORAN [12] are investing in open, intelligent, and programmable systems. ORAN is aiming at standardizing the SD-RAN principle. However, while ORAN paves the way towards a softwarization era and constitutes the basis for ongoing research, it still lacks guarantees when it comes to performance predictability in SD-RAN.

When considering 5G networks with thousands of connected devices [13], [14], there are two main problems that arise. The first lies in the ability of single-controller solutions to provide any performance prediction for networks of large dimensions. The second is concerned with the development of

This article has been accepted for publication in IEEE Transactions on Network and Service Management. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TNSM.2023.3276880

2

architectures and methods that guarantee the predicted level of service.

In order to be able to tackle these issues, initially we need to gain a deeper understanding of the existing SD-RAN solutions in the literature and provide insights on performance predictability, which is a prerequisite to constructing new models and architectures. Whereas works that evaluate state-of-the-art solutions exist [15], still they mainly focus on the control plane aspect of the SD-RAN systems. However, an important part of the network, UEs, have not been considered in depth so far. In that regard, the need for information with respect to performance provisioning on the data plane, where the UEs lie, becomes inevitable.

A serious challenge in maintaining the communication quality in the cellular network stems from the sensitivity of the SD-RAN controller decisions (regarding the allocation of resources to BSs) on the network load increase, as shown in Fig. 1. Indeed, as presented in [15], when the number of data plane devices increases beyond 3000, the controller becomes prone to undesired behavior due to its limited processing capability. Consequently, the controller demonstrates a delay in processing the control packets. The reasonable question which then arises is *how does this affect the UEs in the data plane, and what happens with the overall throughput, especially when the goal of the network operator is throughput maximization?* To the best of our knowledge, this question is left unanswered in the literature so far from the analytical and practical perspectives.

In order to fill this void in the state of the art, in this work we shed light on computing the throughput on the data plane depending on imperfections of the SD-RAN controller in the control plane. To validate our results, we design `Delphi`, a novel platform that is based on open-source code. `Delphi` includes both theoretical models and measurement results for throughput computation in SD-RAN environments, with respect to maximum Channel Quality Indicator (maxCQI) and Round-Robin (RR) resource allocation policies. Considering the case when the goal of network operators is to provide overall throughput maximization, the first policy is studied, whereas the latter represents a static resource allocation solution, and serves as a comparison. Based on the outcomes of this work, we are able to identify performance implications of the SD-RAN controller and provide analytical tractability. One of the main messages of this paper is that for a maxCQI policy, the data plane throughput is not affected considerably up to the point when the number of RAN elements in the data plane increases beyond 5000. Additionally, we verify that the RR policy is oblivious to control packet delays.

Our main contributions are summarized as follows:

- We design `Delphi`, a novel platform which enables experimental evaluation and analytical tractability of throughput in SD-RAN environments. Our approach leverages available open-source code and further extends and unifies separate components into a single system (Section III).
- We provide a mathematical framework to compute the

throughput performance assuming control packet losses[1] on the SD-RAN control plane for various resource allocation policies (Section IV).
- Utilizing `Delphi`, we provide, to the best of our knowledge, the first large-scale measurement campaign based on OpenAirInterface and FlexRAN related to throughput computation that takes into account SD-RAN control plane imperfections (Section V).

II. RELATED WORK

RAN has always played a major role in all the generations of cellular networks. Its importance is even more emphasized in 5G, with its main task lying in the provisioning of heterogeneous services in terms of their requirements and the establishment of programmable and flexible solutions compared to traditionally monolithic RAN infrastructures. All these stringent requirements hinder efficiency and collaboration between different RAN entities. A factor of paramount importance towards this transformation is related to the concepts of SD-RAN and RAN virtualization.

A. SD-RAN Solutions

SD-RAN has triggered vast ongoing research both from theoretical and practical points of view. Conceptual works for SD-RAN are mainly targeted in [16]–[20]. OpenRAN [16] and SoftRAN [18] are among the first studies that envision the introduction of a centralized controller for RAN that enables efficient management. Yet, none of those approaches provides a practical system implementation.

Alternatively, prototypes of SD-RAN are initiated with FlexRAN [5] and 5G-EmPOWER [10], where both platforms provide initial implementations of the SD-RAN concept with a separation of the control plane from the data plane. Further, Orion [21] is a platform similar to FlexRAN. Additionally, [21] introduces new features in RAN, specifically with respect to RAN slicing. Recently, FlexRIC [22], an SD-RAN controller that follows the principles of 5G networks was proposed that provides an ultra-lean design able to be tailored to specific 5G use cases. In turn, the concept of SD-RAN has received significant attention from industrial players, where Rakuten [11] and ORAN [12] embrace softwarization as the main tool. More specifically, ORAN provides two standardized controllers, i.e., the RAN real-time RIC [23] and non-real-time RIC [24]. Further, the E2 interface for communication with the BS and real-time RIC is established [25].

However, while all the above-mentioned approaches provide valuable tools and insights, they only present small-scale measurements for up to 50 RAN elements. FlexRAN [5] also provides initial throughput evaluation given delayed controller decision reception from BSs, but only for 1 UE and 1 BS. Differently from our work, the scenario where the SD-RAN controller is overloaded is not studied. Moreover, there is no theoretical model to capture the throughput degradation analytically. Conversely, the work in [10] demonstrates the

---

[1]In reality, control packets are not lost but received at the BS with delay. In our system, such packets are discarded and for the remainder of this paper will be referred to as *lost*.

CPU consumption of 5G-EmPOWER under a handover use-case for more than 100 RAN emulated elements. Still, these performance evaluations do not target the investigation of imperfections in the deployment. Additionally, in our previous work [15] large-scale measurements for the FlexRAN and 5G-EmPOWER controllers are provided. Yet, only the control plane aspect was in focus, and results for the UEs in the data plane are not shown. Indeed, as most applications in RAN target UE satisfaction, an evaluation of their performance as a result of non-ideal behavior in the control plane becomes more than necessary. An initial evaluation of UE satisfaction in SD-RAN was provided in our previous work [26], however, the solution focuses on simulation only, whereas in this work we focus on the implementation-specific issues.
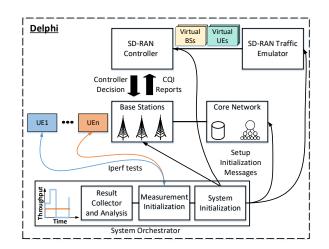
B. RAN Virtualization

In the context of RAN virtualization, OpenAirInterface [27] and srsRAN [?] constitute the pillars of academic research. They both provide a 3GPP-based softwarized platform for 4G/5G networks, where the core, RAN, and UE components are running in pure software. Moreover, they allow for connection to SD-RAN controllers, where srsRAN is compliant with 5G-EmPOWER [10], whereas OpenAirInterface is compliant with both FlexRAN [5] and FlexRIC [22]. However, OpenAirInterface and srsRAN vanilla versions do not provide means for large-scale measurements given SD-RAN controller inconsistencies.

Due to the high traction received in the last years, wireless platform emulators have been developed to allow for large-scale deployment of RAN environments [6], [28]–[31]. Arena [28], POWDER [30], and Colosseum [31] equip the community with powerful wireless platform emulators, with the latter being considered as the largest up-to-date. In turn, Scope [6], which is based on Colosseum [31], provides a framework for open and softwarized prototypes.

Moreover, recent works like CARES [32], vrAIn [33], Concordia [7], and Nuberu [8] deal with resource allocation in such virtualized environments, providing valuable insights on the operation and management aspect. Yet, none of the above-mentioned works consider the large-scale experimentation with respect to SD-RAN environments, especially not with a controller under high load. Furthermore, for the latter scenario of SD-RAN environments under high load, to the best of our knowledge, there exists no other platform that provides both analytical evaluation and measurement tools for throughput evaluation of UEs as a consequence of SD-RAN control plane inconsistencies, which is the main objective of this paper.

III. DELPHI

To measure the throughput of individual UEs as well as the total throughput (over all UEs) in SD-RAN environments, we present a novel platform, which we name Delphi.[2] Our approach is specifically engineered for 4G/5G systems that operate over virtualized RAN environments and are controlled

[2]The name is inspired by the Greek mythology, where Delphi was home of Pythia, the oracle famous for predicting the future.

Fig. 2: Overview of Delphi. The platform is composed of the core network that assigns IPs to UEs, the SD-RAN controller, and physical OpenAirInterface BSs. Additionally, the SD-RAN traffic emulator generates virtual UEs and BSs. The system orchestrator initializes the setup, initiates the measurements, collects the results, and analyzes them.

by an SD-RAN controller with limited processing capacity. Delphi aims at characterizing throughput performance both mathematically and by measurements for the scenarios when the SD-RAN controller is overloaded. This becomes particularly important as these theoretical insights could be used to avoid overloaded scenarios and would allow for maintaining the desired level of service in the network.

Here, we elaborate on the main components of Delphi, and on their mutual interaction. Then, we detail the resource allocation policies utilized in our system and the UE Channel Quality Indicator (CQI) patterns. Finally, we shed light on the control packet losses and the simulation framework.

A. System Design

The high-level architecture of Delphi is presented in Fig. 2. Delphi is based on open-source components and is in line with the principles of programmability and softwarization of 5G.

As can be observed from Fig. 2, our platform is composed of 5 main components: the core network, the BS (which in 5G is known as gNB) and UEs, the SD-RAN controller, the SD-RAN traffic emulator, and the system orchestrator. A detailed explanation of all the components is presented as follows.

1) Core Network: The core network is based on the 4G 3GPP standard compliant OpenAirInteface core [34] and it mainly provides three functionalities: the Home Subscriber Server (HSS), which stores the database of UEs that access the network, the Mobility Management Entity (MME) that keeps track of the UE mobility, and the Service/Packet Gateways (S/P GWs) that are responsible for assigning IPs to UEs in order to maintain a connection. The reason for choosing the 4G core version of OpenAirInterface is related to its stability compared to the 5G counterpart.

Although alternative open source 5G core network components, such as Open5GS [35], could have been used, we
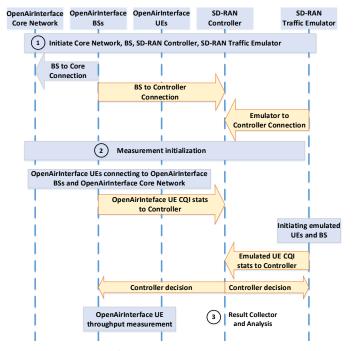
stress that the choice of the core network does not alter the performance of resource allocation in RAN, which is within the scope of this work. The main reasoning stems from the fact that the messages of interest in our work concern only those among the SD-RAN controller and BSs, which do not have any core network impact. The core network is instead used in our work to be able to obtain measurements with respect to UE's throughput.

2) SD-RAN Controller: The SD-RAN controller is based on the FlexRAN principle [5] but utilizes the latest code from Eurecom [36]. In our platform, the control of the network, in particular with respect to wireless resource allocation, involves the SD-RAN controller. Initially, the SD-RAN controller distributes the wireless resources, referred to as Physical Resource Blocks (PRBs), among the BSs in what is called **controller decision**. This logic is aligned with the xApps concept of ORAN [12] that can be used to provide decisions such as radio resource allocation to BSs [37], [38]. In turn, the BSs distribute the assigned PRBs from the SD-RAN controller among the corresponding UEs using the traditional **MAC resource allocation**. `Delphi` utilizes the FlexRAN Application Programmable Interface (API) [39] and FlexRAN protocol [5] for the communication with the BS. Every BS has a unique connection to the FlexRAN controller via its own link, and the correct delivery is ensured by means of Transport Control Protocol (TCP).

The controller decision is sent from the SD-RAN controller towards the BSs every 1 s, which we call **control period** and is based on channel statistics containing information such as the CQI of UEs within BSs.[3] To enable this procedure, `Delphi` extends the available FlexRAN code and implements a set of controller resource allocation policies, as detailed in Section III-C.

3) Base Station: The BS is the main component of any RAN system. Within the BS, several UEs are assigned wireless resources depending on their CQIs and the MAC resource allocation at the BS. In this work, for the BS and UEs, we utilize the emulation mode of OpenAirInterface based on the `mosaic5g-oai-sim` branch [40]. The rationale for this choice is twofold. Firstly, the `mosaic5g-oai-sim` provides a realistic wireless channel model based on 3GPP standardization. Secondly, it allows for a manual configuration and control of CQI values for UEs. That leads to an easier deployment and assessment of results utilizing realistic UE traces, but also fosters reproducibility and analytical tractability.

4) SD-RAN Traffic Emulator: The SD-RAN traffic emulator is based on [15]. This tool generates virtual BSs and virtual UEs that follow the FlexRAN protocol. In our system, we utilize the traffic emulator in order to generate background traffic in the network and stress the SD-RAN controller to its capacity limits in order to track its capabilities. Here we emphasize that for the virtual BSs and UEs generated from [15] we cannot measure their throughput due to the fact that they do not connect to the core network. Hence, they do not receive IPs. This is only available for the OpenAirInterface BSs.

---

[3]The 1 s period is tied to the granularity of *iperf* measurements.

Fig. 3: Overview of `Delphi`'s component interaction. The first step comprises the initialization of the system. Following, the measurement initialization enables sharing of OpenAirInterface and emulated UEs' statistics towards the SD-RAN controller. Once the controller provides the controller decision, the third step enables the measurement of throughput.

5) System Orchestrator: The orchestrator block of `Delphi` is based on scripts that are used to configure network components, such as BSs, core network, and the SD-RAN controller. Moreover, the orchestrator initiates various measurements and finally collects and analyzes the results. In principle, we follow the logic of Mosaic5G [41]. But, we adapt the respective scripts to achieve the automation of: 1) initialization of all system components, 2) adaptation of UE CQI patterns based on a case study and realistic traces, 3) seamless communication with the SD-RAN controller, and 4) throughput measurements, result collection, and analysis.

B. Delphi Component Interaction

While the system design of `Delphi` was provided in the previous subsection, here we elaborate further on the interaction among all components of `Delphi`, as portrayed in Fig. 3. In order to be able to obtain results for analysis with the novel platform, the first step comprises the initialization of the components such as the SD-RAN controller, BSs, core network, and SD-RAN traffic emulator. This in turn consists of the OpenAirInterface BSs' connection to the core network, as well as the connection of the traffic emulator towards the SD-RAN controller.

Once all components have been initialized, the second step, as shown in Fig. 3, consists of the measurement initialization. At this moment, depending on the configuration of choice, the OpenAirInteface UEs for which we want to measure their throughput start to connect to the core network and OpenAir-
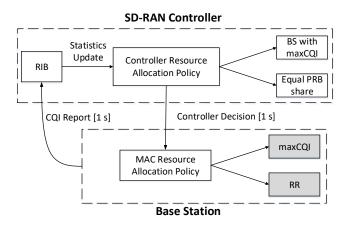
Fig. 4: An overview of the possible system policies. The SD-RAN controller applies the controller decision with respect to BS resource allocation based on CQI statistics. The MAC resource allocation at the BS is performed based on the PRB allocation decided by the controller.

Interface BSs. Similarly, the emulated BSs and emulated UEs are started at the traffic emulator. Once UEs have populated the system, their statistics in terms of CQIs are shared with the SD-RAN controller. The SD-RAN controller provides the appropriate analysis and decides on the resource allocation accordingly, which comprises the distribution of PRBs towards the BSs. Once BSs receive the PRBs, they internally distribute them to UEs. In that case, the third step of Fig. 3 can start, which is the collection and analysis of the obtained results.

C. Controller and MAC Resource Allocation Policies

In `Delphi`, the wireless resource allocation is implemented in two steps. Initially, it is performed at the SD-RAN controller (i.e., controller policy) and then at the BS (i.e., MAC policy). The SD-RAN controller updates the resource distribution to BSs every $1\,$s through the controller decision. These updates are based on CQI reports from the BSs with respect to their UEs, as illustrated in Fig. 4. The goal of `Delphi` is to provide throughput computation in overloaded SD-RAN environments when the network operator's goal is throughput maximization.

Different resource allocation policies running at the SD-RAN controller and within the BS produce different results in terms of throughput. For instance, if the RR policy is used at the SD-RAN controller, then BSs within the controller's operational region share equally the wireless resources among them. In turn, UEs within the BSs will have to share the received wireless resources too. While this may result in a fair distribution of resources among UEs, it will not yield the highest achievable throughput.

On the other hand, if the maxCQI policy is used at the SD-RAN controller, the BS whose UEs achieve the highest CQI during the control period obtains all the resources. In turn, the UEs of the BS with the highest CQI receive all the resources of that BS given that UEs always have data to send.

Covering two very important aspects of a cellular network, which are overall throughput maximization and providing

robust behavior under system imperfections is the reason for choosing the two aforementioned policies (maxCQI and RR) in this work. For the remainder of this paper, to follow the two-step resource allocation logic we refer to these policies as maxCQI-maxCQI and RR-RR, respectively. The first term refers to the controller resource allocation policy, whereas the second to the MAC resource allocation policy at a BS.

D. CQI Patterns

In a cellular system, the UE throughput is impacted by its CQI value. This can be observed in Fig. 6, where the results were obtained with `Delphi`. The x-axis represents the range of UE CQI values, while the y-axis the achieved throughput in the system when all 25 PRBs are assigned to that UE. The minimum with all the resources assigned to the same UE is achieved when $CQI = 1$, and is $0.62\,$Mbps, whereas the maximum is achieved for $CQI = 15$ - in total $18.3\,$Mbps.

Given that CQI values have a direct impact on the achievable UE throughput, in this subsection, we shed light on the selection of the CQI values for our evaluations. To that end, we consider two scenarios:
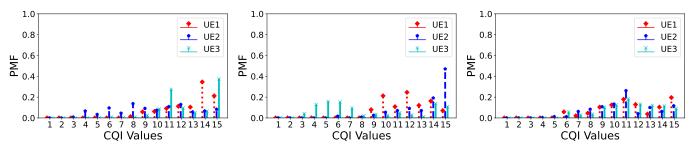
- A **case study**, where UEs have either the CQI of 15 (representing excellent channel conditions) or 7 (representing medium channel conditions). There is an equal probability for a UE to have either one of these CQIs.
- **Real traces** for UEs, as depicted in Fig. 5, which are obtained from [42] and [43], where for 3 BSs we depict CQI patterns of various UEs, all with different Probability Mass Functions (PMF).

E. Control Packet Losses

The first resource allocation step happens at the SD-RAN controller every $1\,$s (control period), and the decision (with the allocated resources) is sent to the BS. Once the BS receives the decision with the respective PRB allocation, it updates the previously stored one. However, when the load of the SD-RAN controller increases, due to the limited processing capacity of the controller, some of the controller decisions sent to the BSs arrive with delay. As demonstrated in Fig. 1, when the number of RAN elements reaches 10000, the aggregated load at the controller reaches $250\,$Mbps, overloading it and making it prone to operating with flaws. We note that these imperfections in controller behavior are not caused by the TCP, since as portrayed in Fig. 1, for 1 BS towards the controller even for 100 UEs the necessary connection capacity does not surpass $5\,$Mbps. In our system, we define two use cases. Initially, we assume that when a control packet arrives at the BS with delays, it is discarded and considered as **lost**. In that case, the controller decision is not updated and the BS keeps the previous one. This can lead to throughput degradation as the channel conditions may have changed drastically from the previous control period.

An example of this scenario is depicted in Fig. 7. Every control period i.e., $1\,$s, the BS receives the controller decision, containing the new resource allocation from the SD-RAN controller, and updates its last stored resource allocation

(a) PMF of CQI values for 3 UEs, BS1   (b) PMF of CQI values for 3 UEs, BS2   (c) PMF of CQI values for 3 UEs, BS3

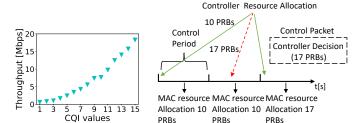Fig. 5: PMFs for 3 BSs and 3 UEs within each BS based on realistic UE traces from a 5G dataset [42].

Fig. 6: CQI to throughput conversion for 25 PRBs from measurements provided by `Delphi`.

Fig. 7: Scenario of control packet losses. MAC resource allocation is updated every control period. Lost packets are in dashed red, received packets are shown in green.

scheme from the previous control period. When the packet is received timely, then the control packet reaches the BS, which is depicted in green. Otherwise, the control packet is considered as lost (shown in the red dashed line in Fig. 7).

While the first scenario is easier in terms of analytical tractability, in reality, the control packet is not lost, but it reaches the BS with a delay. Therefore, in our second use case, we drop this assumption and measure directly the effect of the delayed control packet on the UE throughput with respect to background traffic generated by virtual BSs and UEs, as detailed in Section V.

F. Simulation Framework

In order to verify the measurements and the theoretical model provided by `Delphi`, we also provide a simulation framework that serves as our **baseline**. The simulation contains the actual information with respect to UE CQI values in all scenarios. The conversion of CQI to throughput is applied according to the measurements provided by `Delphi`, as shown in Fig. 6. For every UE in each BS, a list is created, where each entry corresponds to a measurement point that represents a control period. Each list entry contains a CQI value following the distributions depicted in Fig. 5, or the case study (CQIs of 7 and 15 with identical probabilities of 1/2). Having the knowledge of all CQIs at each measurement point

(control period), an assessment occurs to compare the CQIs among all UEs and BSs. Depending on the resource allocation policy (see Fig. 4), the resources are distributed to BSs and consequently to UEs. The achieved throughput then depends on the CQI in the list entry and the policy.

Regarding the control packet losses, at each measurement point, the control packet is either lost or received timely at the BS (see Fig. 7). At the beginning of each simulation, according to the assumed ratio of lost control packets, a list with binary entries is generated, containing 0 if the control packet is received timely and 1 if it is lost. This generation occurs randomly. If the control packet is lost, then the PRB allocation from the previous measurement point is applied, otherwise, the exact resource allocation is carried out.

IV. THEORETICAL MODEL

Similar to 4G, the block resource allocation scheme is used in 5G as well, with *PRB* being the allocation unit [44], but allowing higher flexibility in choosing the subcarrier spacing, and consequently, the PRB bandwidth. This then conditions the duration of the slot. Within a slot, different PRBs are assigned to different UEs. In general, the assignment varies across slots. Consequently, resource allocation is to be performed across two dimensions, *frequency* and *time*. The total number of PRBs in the system is K.

In general, UEs experience different channel conditions in different PRBs even within the same slot, and therefore a different per-block Signal-to-Interference-Plus-Noise-Ratio (SINR). Because of UE mobility and time-varying channel characteristics, per-PRB SINR changes from one slot to another. This changing *per-PRB* SINR translates into a varying per-PRB rate. The value of SINR in a slot determines the CQI (a parameter sent by the UE to the BS), as shown in Table I, which depending on the Modulation and Coding Scheme (MCS) sets the per-PRB rate. There are 15 possible values of CQI [45]. E.g., if at time t the per-PRB SINR lies in the interval $[\gamma_l, \gamma_{l+1}]$, with γ_l and γ_{l+1} being the thresholds of the CQI ($l = 1, \ldots, 15$), the per-PRB rate would be $r_l(t)$ [46], [47]. To maintain analytical tractability and compare the results under the same conditions with OpenAirInterface, in our model the BS splits the transmission power equally among all the K PRBs, and the channel characteristics for a UE remain unchanged across all PRBs (i.e., identical CQI over all

TABLE I: SINR to CQI conversion

SINR threshold (dB)	-6.5	-4	-2.6	-1	1	3	6.6	10	11.4	11.8	13	13.8	15.6	16.8	17.6
CQI value	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15

PRBs), but they change randomly from one slot to another, whereas different UEs experience distinct CQI distributions. This reduces the resource allocation problem to the number of allocated PRBs and not to which PRBs are assigned to a given UE.

Having in mind the previous terms, it follows that in every slot the per-PRB rate of UE j can be modeled as a discrete random variable, R_j, with values in $\{r_1, r_2, \ldots, r_{15}\}$, such that $r_1 < r_2 < \ldots < r_{15}$, with a PMF $p_{R_j}(x)$. The latter is a function of UE's j SINR over time.[4]

In this paper, we focus our attention on two resources allocation policies - maxCQI, where the best UE or BS gets all the PRBs[5] and RR, where resources are split equally among all the entities. Different resource allocation policies can be implemented on the controller and on BSs. In the first case, we present the result when the RR resource allocation is implemented at both entities. Then, we focus on maxCQI.

A. Round-Robin Resource Allocation Policy

When using this resource allocation policy both at the controller and at the BSs, we assume that the PRBs are shared equally among all the BSs, and within the coverage area of a given BS, the PRBs are shared among all its active UEs.

In the general case of M BSs, where the number of UEs in each BS is denoted by $n_i, \forall i \in \{1, \ldots, M\}$, every BS would receive $\frac{K}{M}$ PRBs. The average throughput of UE j, being served by BS i, is[6]

$$\mathbb{E}[Th_{i,j}] = \frac{K}{Mn_i}\mathbb{E}[R_{i,j}]. \tag{1}$$

The throughput in BS i is then

$$\mathbb{E}[Th_i] = \sum_{j=1}^{n_i} \mathbb{E}[Th_{i,j}]. \tag{2}$$

Therefore, we have the following:

Result 1. *The total throughput in the system is*

$$\mathbb{E}[Th] = \sum_{i=1}^{M} \mathbb{E}[Th_i] = \sum_{i}^{M}\sum_{j=1}^{n_i}\mathbb{E}[Th_{i,j}] = \frac{K}{M}\sum_{i=1}^{M}\frac{1}{n_i}\sum_{j=1}^{n_i}\mathbb{E}[R_{i,j}]. \tag{3}$$

Note that the RR resource allocation policy is completely insensitive to the loss of control packets; the assignment of resources is fixed at all times.

B. maxCQI Resource Allocation Policy

To ease the presentation, in this section, we consider the case with two BSs and with a single UE in each of them. Then, we explain how the analysis can be extended to multiple UEs

per BS. Let R_1 denote the per-PRB rate of the UE in BS1. As already mentioned, it can take a value from the discrete set $\{r_1, \ldots, r_{15}\}$, depending on the value of CQI in a slot. Similarly, R_2 denotes the per-PRB rate of the UE in BS2. We denote the PMF of R_1 and R_2 by $p_{R_1}(x)$ and $p_{R_2}(x)$, respectively.

Let q denote the probability of the control packet loss. This can arise as a consequence of a large number of UEs and BSs sending their packets to the controller with finite processing capacity. A packet not processed by the end of the control period at the controller is considered lost at the BS and in that case the resource allocation decisions from the previous control period are enforced (see Fig. 7). First, we solve the scenario when $q = 0$, i.e., in the case when the controller processes and allocates the resources timely.

We also need this result for the case when there are packets that are lost, too.

Conditioning upon the possible outcomes, we obtain the average throughput for this case as

$$\mathbb{E}[Th_{q=0}] = \mathbb{P}(R_1 > R_2)K\mathbb{E}[R_1|R_1 > R_2] + $$
$$\mathbb{P}(R_1 = R_2)\frac{K}{2}\mathbb{E}[R_1 + R_2|R_1 = R_2] + $$
$$\mathbb{P}(R_1 < R_2)K\mathbb{E}[R_2|R_1 < R_2]. \tag{4}$$

Namely, in a slot, either R_1 and R_2 are identical, or one of them is higher than the other. For instance, when $R_1 > R_2$, the average throughput in that slot is $K\mathbb{E}[R_1|R_1 > R_2]$, because the UE in BS1 gets all the resources. A similar allocation is performed when the UE in BS2 has better channel conditions. If both UEs have identical R, they receive the same amount of PRBs, i.e., $\frac{K}{2}$, as can be observed from Eq.(4).

We derive next all the terms from Eq.(4). The first probability term yields

$$\mathbb{P}(R_1 > R_2) = \sum_{x=r_1}^{r_{15}}\mathbb{P}(R_1 > R_2|R_2 = x)p_{R_2}(x) = \sum_{x=r_1}^{r_{15}}\bar{F}_{R_1}(x)p_{R_2}(x), \tag{5}$$

where $\bar{F}_{R_1}(x) = 1 - \mathbb{P}(R_1 \leq x)$ is the Complementary Cumulative Distribution Function (CCDF). Similarly, for the other two probability terms, we have

$$\mathbb{P}(R_1 = R_2) = \sum_{x=r_1}^{r_{15}}p_{R_1}(x)p_{R_2}(x), \text{and} \tag{6}$$

$$\mathbb{P}(R_1 < R_2) = \sum_{x=r_1}^{r_{15}}\bar{F}_{R_2}(x)p_{R_1}(x). \tag{7}$$

For the final result of the first expectation term in Eq.(4), after some algebra, we obtain

$$\mathbb{E}[R_1|R_1 > R_2] = \frac{\sum_{x=r_1}^{r_{15}}\sum_{y=r_1}^{x-\epsilon} x \cdot p_{R_1}(x)p_{R_2}(y)}{\sum_{x=r_1}^{r_{15}}\bar{F}_{R_1}(x)p_{R_2}(x)}. \tag{8}$$

Note that the denominator in Eq.(8) is simply $\mathbb{P}(R_1 > R_2)$ (already derived in Eq.(5)). Also, observe that since R_1 and R_2 can take values from a finite discrete set, and the expectation

---

[4]For notational simplicity, we omit the reference to time from now on.

[5]When two or more entities have identical (and the best) channel conditions in the control period, the resources are split equally among them with the maxCQI resource allocation policy.

[6]We use $R_{i,j}$ to denote the per-PRB rate UE j receives from BS i.

in Eq.(8) is conditioned upon the strict inequality between R_1 and R_2, we use the variable $\epsilon > 0$ to denote a very small positive quantity near 0. This implies that for a given value of R_1, R_2 has to be lower than that.

Similarly, for the third expectation term in Eq.(4) we get

$$\mathbb{E}[R_2|R_1 < R_2] = \frac{\sum_{x=r_1}^{r_{15}} \sum_{y=r_1}^{x-\epsilon} x \cdot p_{R_2}(x) p_{R_1}(y)}{\sum_{x=r_1}^{r_{15}} \bar{F}_{R_2}(x) p_{R_1}(x)}, \quad (9)$$

whereas for the second expectation term in Eq.(4), we obtain

$$\mathbb{E}[R_1 + R_2|R_1 = R_2] = \frac{\sum_{x=r_1}^{r_{15}} 2 \cdot x \cdot p_{R_1}(x) p_{R_2}(x)}{\sum_{x=r_1}^{r_{15}} p_{R_1}(x) p_{R_2}(x)}. \quad (10)$$

1) Packet loss: In this case, control packets are received at the BS with a delay, i.e., $q > 0$. Whenever a packet is delayed, it is dropped and considered lost. In that case, PRBs are allocated in the same way as in the previous control period. We have the following:

Result 2. *The average throughput when there are packets that are lost is expressed as*

$$\mathbb{E}[Th] = (1 - q)\mathbb{E}[Th_{q=0}] + q\mathbb{E}[Th_{q>0}]. \quad (11)$$

The term $\mathbb{E}[Th_{q=0}]$ in Eq.(11) corresponds to the instants when the packet is received timely, which occurs with probability $1-q$. Essentially, it is equal to Eq.(4). The second right-hand side term, $\mathbb{E}[Th_{q>0}]$, captures the throughput when the packet is lost, which occurs with probability q. It is expressed as

$$\mathbb{E}[Th_{q>0}] = \mathbb{P}(R_1 < R_2)\mathbb{E}[A|R_1 < R_2] + \\ \mathbb{P}(R_1 = R_2)\mathbb{E}[B|R_1 = R_2] + \mathbb{P}(R_1 > R_2)\mathbb{E}[C|R_1 > R_2]. \quad (12)$$

It is worth mentioning that in Eq.(12) we condition the expectation on three events in the last control period before the packet was lost: whether the UE in BS1 had a better channel, whether both UEs had identical channels, or whether the channel of the UE in BS2 was better during the control period prior to the control period the packet was lost.

The average system throughput when the UE in BS2 had a better channel in the previous control period when the packet was received timely, provided that in the current control period, the packet is lost, $\mathbb{E}[A|R_1 < R_2]$, is given by

$$\mathbb{E}[A|R_1 < R_2] = \mathbb{P}(R_1 < R_2)\mathbb{E}[KR_2|R_1 < R_2] + \\ \mathbb{P}(R_1 = R_2)\mathbb{E}[KR_2|R_1 = R_2] + \mathbb{P}(R_1 > R_2)\mathbb{E}[KR_2|R_1 > R_2]. \quad (13)$$

Let us look more closely at the individual terms of Eq.(13). If in the current control period (when the packet is lost), $R_2 > R_1$, the average throughput obtained is $\mathbb{E}[KR_2|R_1 < R_2]$. Namely, again the UE of BS2 receives all the resources as it was the best UE in the previous control period. So, in this case, the policy assigns the PRBs correctly despite the fact that the packet was lost. On the other hand, if the UE channels are identical or the UE of BS1 has a better channel in the control period with a lost packet, still the UE of BS2 obtains all the resources, i.e., $\mathbb{E}[KR_2|R_1 = R_2]$ and $\mathbb{E}[KR_2|R_1 > R_2]$, respectively, in spite of not having the highest CQI anymore.

Following a similar reasoning, when both UEs had identical channels in the previous control period, and in the current control period the packet is lost, the average throughput is

$$\mathbb{E}[B|R_1 = R_2] = \mathbb{P}(R_1 < R_2)\mathbb{E}\left[\frac{KR_1}{2} + \frac{KR_2}{2}|R_1 < R_2\right] + \\ \mathbb{P}(R_1 = R_2)\mathbb{E}\left[\frac{KR_1}{2} + \frac{KR_2}{2}|R_1 = R_2\right] + \\ \mathbb{P}(R_1 > R_2)\mathbb{E}\left[\frac{KR_1}{2} + \frac{KR_2}{2}|R_1 > R_2\right]. \quad (14)$$

Finally, if the UE of BS1 had a better channel in the previous control period, the corresponding average throughput is

$$\mathbb{E}[C|R_1 > R_2] = \mathbb{P}(R_1 < R_2)\mathbb{E}[KR_1|R_1 < R_2] + \\ \mathbb{P}(R_1 = R_2)\mathbb{E}[KR_1|R_1 = R_2] + \mathbb{P}(R_1 > R_2)\mathbb{E}[KR_1|R_1 > R_2], \quad (15)$$

meaning that the UE of BS1 will receive all the resources when a packet is lost.

In Eqs.(12)-(15), the expressions for $\mathbb{P}(R_1 < R_2)$, $\mathbb{P}(R_1 > R_2)$, and $\mathbb{P}(R_1 = R_2)$ were already derived in Eqs.(5)-(7). Eqs.(8)-(10) can be used for some of the terms in Eqs.(12)-(15). Finally, for the remaining terms, we have

$$\mathbb{E}[R_1|R_1 = R_2] = \mathbb{E}[R_2|R_1 = R_2] = \frac{\sum_{x=r_1}^{r_{15}} x \cdot p_{R_1}(x) p_{R_2}(x)}{\sum_{x=r_1}^{r_{15}} p_{R_1}(x) p_{R_2}(x)}. \quad (16)$$

We substitute Eqs.(5)-(10), (16) into Eqs.(13)-(15), and the so obtained Eqs.(13)-(15) into Eq.(12). Then, we substitute the latter together with Eq.(4) into Eq.(11) to obtain the total system throughput.

In this section, we have shown the derivation of the overall throughput. The procedure for obtaining the throughput of individual UEs is straightforward. Let us consider the UE in BS1. The differences are as follows. Eq.(4) contains only the first two terms (no allocated resources to that UE when the other BS has a better channel). For the same reason, Eq.(12) contains only the second and third terms. The remainder of the procedure (all the other expressions from Section IV-B, including Eq.(11)) remains unchanged. A similar procedure is followed for the UE of BS2.

C. Scenario: 3 BSs, no Packet Losses

Next, we consider the scenario with 3 BSs, and one UE in each of them for the maxCQI policy. Only the analysis for $q = 0$ is shown. Due to space limitations and because it is of no further technical interest, we omit the analysis for $q > 0$. Following a similar reasoning as with 2 BSs, we have:

Result 3. *The average throughput with three BSs, wherein the area of each one of them there is one UE with per-PRB rates R_1, R_2, and R_3 is*

$$\mathbb{E}[Th_{q=0}] = \mathbb{P}(R_1 > R_2, R_1 > R_3)K\mathbb{E}[R_1|R_1 > R_2, R_1 > R_3]$$
$$+\mathbb{P}(R_1 = R_2, R_1 > R_3)\frac{K}{2}\mathbb{E}[R_1 + R_2|R_1 = R_2, R_1 > R_3]$$
$$+\mathbb{P}(R_1 = R_3, R_1 > R_2)\frac{K}{2}\mathbb{E}[R_1 + R_3|R_1 = R_3, R_1 > R_2]$$
$$+\mathbb{P}(R_1 = R_2, R_1 = R_3)\frac{K}{3}\mathbb{E}[R_1 + R_2 + R_3|R_1 = R_2, R_1 = R_3]$$
$$+\mathbb{P}(R_2 > R_1, R_2 > R_3)K\mathbb{E}[R_2|R_2 > R_1, R_2 > R_3]$$
$$+\mathbb{P}(R_2 = R_3, R_2 > R_1)\frac{K}{2}\mathbb{E}[R_2 + R_3|R_2 = R_3, R_2 > R_1]$$
$$+\mathbb{P}(R_3 > R_1, R_3 > R_2)K\mathbb{E}[R_3|R_3 > R_1, R_3 > R_2]. \quad (17)$$

Note that in Eq.(17), a single BS gets all the K PRBs when its UE has the best channel. When two of the BSs UEs have identical per-PRB rates in a control period and the third BS's UE worse channel, those two BSs split the available K PRBs equally. Finally, if the three UEs have the same per-PRB rate in the control period, each one of them will receive $\frac{K}{3}$ of the PRBs in that control period.

For the probability terms in Eq.(17) we have as follows:

$$\mathbb{P}(R_i > R_j, R_i > R_k) = \sum_{x=r_1}^{r_{15}} \sum_{y=r_1}^{x-\epsilon} \sum_{z=r_1}^{x-\epsilon} p_{R_i}(x)p_{R_j}(y)p_{R_k}(z), \quad (18)$$

$$\mathbb{P}(R_i = R_j, R_i > R_k) = \sum_{x=r_1}^{r_{15}} \sum_{z=r_1}^{x-\epsilon} p_{R_i}(x)p_{R_j}(x)p_{R_k}(z), \quad (19)$$

for $i = \{1, 2, 3\}$, $j = \{1, 2, 3\}$, $k = \{1, 2, 3\}$, and $i \neq j \neq k$. Also,

$$\mathbb{P}(R_1 = R_2, R_1 = R_3) = \sum_{x=r_1}^{r_{15}} p_{R_1}(x)p_{R_2}(x)p_{R_3}(x). \quad (20)$$

For the expectation terms in Eq.(17), following the basic rules from the theory of probability, we have:

$$\mathbb{E}[R_i|R_i > R_j, R_i > R_k] =$$
$$\frac{\sum_{x=r_1}^{r_{15}} \sum_{y=r_1}^{x-\epsilon} \sum_{z=r_1}^{x-\epsilon} xp_{R_i}(x)p_{R_j}(y)p_{R_k}(z)}{\sum_{x=r_1}^{r_{15}} \sum_{y=r_1}^{x-\epsilon} \sum_{z=r_1}^{x-\epsilon} p_{R_i}(x)p_{R_j}(y)p_{R_k}(z)}, \quad (21)$$

$$\mathbb{E}[R_i + R_j|R_i = R_j, R_i > R_k] =$$
$$\frac{\sum_{x=r_1}^{r_{15}} \sum_{z=r_1}^{x-\epsilon} 2xp_{R_i}(x)p_{R_j}(x)p_{R_k}(z)}{\sum_{x=r_1}^{r_{15}} \sum_{z=r_1}^{x-\epsilon} p_{R_i}(x)p_{R_j}(x)p_{R_k}(z)}, \quad (22)$$

for $i = \{1, 2, 3\}$, $j = \{1, 2, 3\}$, $k = \{1, 2, 3\}$, and $i \neq j \neq k$. Similarly,

$$\mathbb{E}[R_1 + R_2 + R_3|R_1 = R_2, R_1 = R_3] =$$
$$\frac{\sum_{x=r_1}^{r_{15}} 3xp_{R_1}(x)p_{R_2}(x)p_{R_3}(x)}{\sum_{x=r_1}^{r_{15}} p_{R_1}(x)p_{R_2}(x)p_{R_3}(x)}. \quad (23)$$

A similar procedure, only with more expectation and probability terms, is followed in the case with more than three BSs. We do not show the derivation here, because it is of no further technical interest.[7]

---

[7]The theoretical results can also be expressed in terms of the distributions of the SINR as well. However, the obtained expressions would then become more cumbersome, with no additional insights gained.

D. maxCQI-maxCQI: Multiple UEs per BS

While the previous analysis was concerned only with one UE per BS, the generalization to any number of UEs where maxCQI resource allocation policy is implemented on both the BS level and UE level is almost straightforward. Namely, in the analysis above, we only need to replace, e.g., R_1, R_2, and R_3 with $\max R_1$, $\max R_2$, and $\max R_3$, respectively, where $\max R_i, i = \{1, 2, 3\}$, is the UE with the highest CQI within BS i. Then, using some algebra, the final result follows easily.

V. PERFORMANCE EVALUATION

In this section, we evaluate the performance of our system with respect to the effect of SD-RAN control plane imperfections in the data plane. We split our results in two parts. In the first part, we assume a certain control packet loss ratio in the control plane between the SD-RAN controller and the BS regarding the controller decision. The second part drops this assumption and evaluates the controller performance given additional background traffic. This is achieved by increasing the number of BSs and UEs that the controller has to manage.

Initially, we portray an example of the control packet loss for 2 BSs considering a maxCQI-maxCQI policy. In that regard, we investigate the drop in the sum throughput that is accompanied by the time instance of packet loss. Further, we demonstrate outcomes for two use cases, namely the case study and the real traces (Section III-D). Results are shown for 2 and 3 OpenAirInterface BSs for maxCQI-maxCQI and RR-RR resource allocation policies. For 2 BSs, we verify our measurements with simulations and theoretical results for all the assumed control packet loss ratios. On the other hand, for 3 BSs, due to space limitations, we only provide the theoretical result for the case where no packets are lost. Results are demonstrated in boxplots. For the measurements, each configuration is repeated $10\times$ for $270\,\text{s}$, by removing the initial and last $15\,\text{s}$ to eliminate the transient phase of the measurements. In turn, the simulations are performed $100\times$.

When the packet loss assumption is dropped, the depicted results for 2 and 3 BSs are generated with OpenAirinterface (Section III-A3), while considering additional emulated UEs and BSs (Section III-A4). Hence, we show how the network load affects the throughput drop in % for maxCQI-maxCQI and achieved throughput for the RR-RR resource allocation policies, as the latter is insensitive to control packet losses.

A. Measurement Setup

The measurement setup follows the system design from Fig. 2, where 1 Desktop PC is running the SD-RAN emulator [15]. The PC is equipped with an Intel(R) Core(TM) i5-3470 CPU, with 4 CPU cores @ 3.2 GHz and 8 GB RAM running Ubuntu 18.04.2 LTS with 4.15.0-58-generic kernel. For the FlexRAN SD-RAN controller, a server running an Intel(R) Xeon(R) CPU E5-2650 v4 @ 2.20 GHz is set. The FlexRAN server contains 12 physical CPU cores and 64 GB of RAM operating under Ubuntu 18.04.3 LTS with 4.15.0-66-generic kernel.

Further, OpenAirInterface BSs and the core network operate in an Intel(R) Core(TM) i7-7700T CPU @ 2.9 GHz. Each

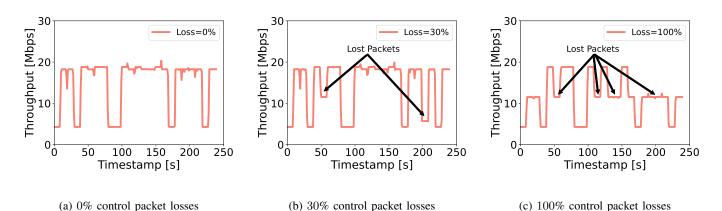(a) 0% control packet losses   (b) 30% control packet losses   (c) 100% control packet losses

Fig. 8: Measured system throughput for 2 BSs over time considering various control packet losses, using maxCQI-maxCQI policy.

BS PC contains 4 physical CPU cores and $16\,GB$ of RAM, whereas the core network, since it is not that demanding, only contains 1 CPU and $4\,GB$ of RAM. The operating system is Ubuntu 16.04.04 LTS with 4.4.0-116-generic kernel.

Unless stated otherwise, a resource allocation application is running on the SD-RAN controller server, which collects CQI values sent by the BSs every $1\,s$. This is a design choice, which can be adjusted according to the channel characteristics (i.e., CQI changes). The CQI values of the UEs within the BSs also change every $1\,s$ [42], [43] and their PMFs for 2 and 3 BSs are illustrated in Fig. 5.

B. Control Packet Loss Ratio Assumption

In this subsection, we assume that a certain amount of packets in the control plane are lost. These messages contain controller decisions with respect to the resource allocation sent from the SD-RAN controller to the BSs. The goal is then to identify what is the influence of these control packet losses on the data plane, specifically the impact on the throughput.

1) Investigation of Sum Throughput Drop: First, we investigate the degradation in the sum throughput over all UEs for the example of 2 BSs and 1 UE in each BS with respect to the control packet loss ratio under the maxCQI-maxCQI resource allocation policy. These results are shown in Fig. 8. The distribution of UEs for the results obtained in Fig. 8 corresponds to the case study. Moreover, they are concerned with only a single measurement with the goal to investigate the sum throughput degradation as a result of control packet loss.

The results for 0% control packet losses, shown in Fig. 8a, reveal the ideal sum throughput given the CQI patterns since there is no throughput deterioration. Alternatively, Fig. 8b shows the results pertaining to the case where the number of lost packets recorded is 30%. Hence, we can observe a slight throughput decrease between $50\,s$ and $60\,s$ from $18.3\,Mbps$ to $11.3\,Mbps$. A further decrease is also observed between $200\,s$ and $210\,s$ to approximately $5.3\,Mbps$. The rationale behind this drop lies in the fact that when a control packet loss occurs, the previous resource allocation policy is applied to each BS. In that case, the BS whose UE has the lowest CQI gets all the

resources and consequently, a lower throughput compared to the ideal case is observed.

The situation degrades even further when the packet losses increase to 100%[8], as demonstrated in Fig. 8c. In such a scenario, the instances where the throughput drops increase drastically. The worst experience is recorded from $230\,s$ till the end of the measurements, where there is a drop to $11.3\,Mbps$ compared to the maximum sum throughput of $18.3\,Mbps$, which represents an overall decrease of 38%.

2) Throughput Evaluation: 2 BSs: So far, we have demonstrated results of the throughput degradation with respect to the % of control plane packet losses. Next, we show results for all the considered patterns not only obtained via measurements, but also analytically in line with the outcomes of Section IV, and simulations (as explained in Section III-F). For all the cases we consider both maxCQI-maxCQI and RR-RR resource allocation policies. The results for 2 BSs up to 3 UEs each for the case study as well as for real UE traces (see Fig. 5) are shown in Fig. 9.

For the results of the maxCQI-maxCQI policy, depicted in Fig. 9a for the CQI case study, the sum throughput for the 2 BSs decreases almost linearly with the increase in control packet loss ratio. The simulation results, theoretical outcomes, and measurements depict the same mean value. In that regard, we can conclude on the accurate prediction feature of our approach to capture correctly the effect on control packet losses in the data plane.

The best-case throughput in Fig. 9a is recorded, as expected when there are no losses in the control plane, and its value is around $14.8\,Mbps$. The obtained result is lower compared to the best-case result obtained in Fig. 8, given the fact that in this scenario the number of UEs per BS is 3 compared to 1 as in the setup of Fig. 8. Consequently, given the distribution of a larger number of UEs as well as the competition for the resources, lower throughput is recorded. When all the control packets are lost, the throughput drops below $12\,Mbps$. This leads to almost 20% in throughput degradation.

---

[8]For a loss ratio of 100%, only the first controller packet is received correctly from BSs and all the others are considered lost.

(a) maxCQI-maxCQI case study   (b) maxCQI-maxCQI real traces   (c) RR-RR policy

Fig. 9: Throughput results for 2 BSs comparing theory, simulation outcomes, and measurements for maxCQI-maxCQI and RR-RR resource allocation policies for the case study and real UE traces

(a) maxCQI-maxCQI case study   (b) maxCQI-maxCQI real traces   (c) RR-RR policy
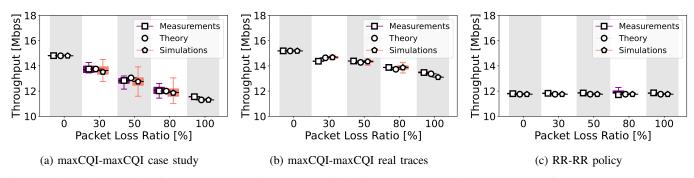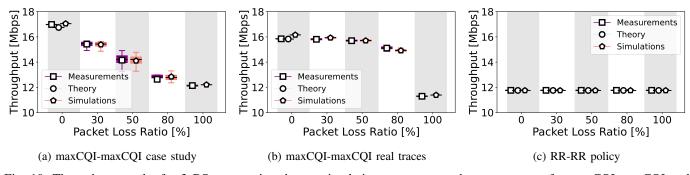
Fig. 10: Throughput results for 3 BSs comparing theory, simulation outcomes, and measurements for maxCQI-maxCQI and RR-RR resource allocation policies for the case study and real UE traces.

Similarly, Fig. 9b shows the throughput achieved for 2 BSs for maxCQI-maxCQI with real UE CQI traces. The maximum throughput is recorded for 0 losses with around 15 Mbps, whereas in the worst case, its value is around 13.8 Mbps for 100% losses. Again, we stress that the results obtained from the measurements match the theoretical results and simulation outcomes, further demonstrating the correctness of our methodology when using real UE CQI traces as well.

Considering the RR-RR policy, as observed in Fig. 9c, the packet loss ratio does not have an impact on the overall system throughput. This is explained by the fact that BSs always receive an equal amount of resources and thus when they do not get updates from the SD-RAN controller, they resume using the previous controller decision. Since that decision is nonetheless the same by design, no throughput degradation occurs. As for both the case study and real traces the RR policy records identical results, we show the results only once. Compared to the maxCQI-maxCQI approach, the achieved throughput is 19% lower, i.e., ≈ 12 Mbps. Therefore, depending on the network conditions, a trade-off between robustness and optimality should be considered.

3) Throughput Evaluations: 3 BSs: Here, we portray results for 3 BSs up to 3 UEs each for the case study as well as for real UE traces for maxCQI-maxCQI and RR-RR policies.

Again, the results of the CQI case study (Fig. 10a) and for real UE CQI traces (Fig. 10b) for the maxCQI-maxCQI resource allocation policy exhibit an almost linear drop in the throughput with the packet loss ratio increase, similar to the 2 BS setup. Compared to the case with 2 BSs shown

in Fig. 9a, the overall achieved throughput is higher for any configuration. The rationale behind this result lies in the higher number of BSs, which increases the chance that at least one of the BSs will experience higher throughput. In that case, even if a packet is lost, the overall system throughput does not decrease considerably. In both scenarios, the theoretical results are presented only for a packet loss ratio of 0 due to space limitations. Nonetheless, simulations are portrayed for all configurations. In any case, theory, simulations, and measurements demonstrate almost equal averages. In terms of throughput degradation, in both scenarios for the case when all the control packets are lost, the throughput drops below 12 Mbps compared to 16 Mbps recorded with no losses. That leads to a throughput degradation of $\approx 29\%$, hardly acceptable in 5G networks.

For the RR-RR policy depicted in Fig. 10c, the throughput is robust to control packet losses and remains constant for all configurations and among BS setups to approximately 12 Mbps, demonstrating again the insensitivity of this policy to control packet losses.

C. SD-RAN Emulated Traffic

While previously we assumed an arbitrary packet loss ratio in the control plane to ease the theoretical evaluation, in this subsection we drop this assumption in order to assess the evaluation of the system. To that end, we generate background traffic in the network with respect to BSs and UEs, using the tool from [15]. For the remainder of this section, results are demonstrated against multiple BSs and UEs, but we only
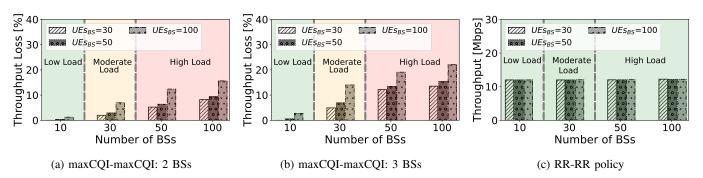
(a) maxCQI-maxCQI: 2 BSs   (b) maxCQI-maxCQI: 3 BSs   (c) RR-RR policy

Fig. 11: Throughput measurements for 2 BSs and 3 BSs while varying the number of emulated BSs and UEs for maxCQI-maxCQI and RR-RR resource allocation policies.

measure the throughput for 2 and 3 BSs OpenAirInterface containing up to 3 UEs each. The additional BSs and UEs generated by the SD-RAN traffic emulator are considered only as background traffic. E.g., if 50 BSs and 30 UEs in each BS are shown, 47 or 48 of those BSs are generated with the SD-RAN traffic emulator, and the additional 2 or 3 ones are generated with OpenAirInterface.
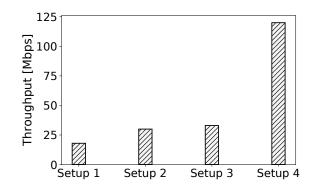
The results for 2 and 3 BSs are presented only for the real UE traces, due to space limitations, and are evaluated with respect to background traffic generated by emulated BSs and UEs. These results are depicted in Fig. 11a and Fig. 11b for the throughput loss in % for the maxCQI-maxCQI resource allocation policy for 2 and 3 BSs. In turn, Fig. 11c shows the achieved throughput for the RR-RR allocation policy.

For both Fig. 11a and Fig.11b, the throughput loss in % increases as the number of BSs and UEs in the network increases. The figure has been split into three areas, namely low load (less than 1000 data plane elements, such as BSs and UEs), moderate load (maximum 3000 data plane elements), and high load with more than 3000 data plane elements. In the regime of low load, in both cases the throughput loss does not exceed 2%, indicating a relatively good operational region. Alternatively, in the moderate load regime, the degradation of the throughput increases up to 7% for 2 BSs and 13% for 3 BSs. This is due to the fact that more losses can occur as the number of BSs increases. Finally, the losses increase beyond 15% and reach almost 23% in the high load scenario for 2 and 3 BSs, accordingly. This indicates that the operation in this region is not satisfying for the network.

On the other hand, as already mentioned, for the RR policy, the throughput remains robust irrespective of the network load, as shown in Fig. 11c.

D. Delphi under 5G core and RAN implementation

While the main results shown in this work are depicted for a 4G core OpenAirInterface (OAI) implementation, in the following we also demonstrate some initial results with Delphi utilizing a 5G core implementation of Free5GC [48] in combination with a 5G OAI RAN implementation [49]. In order to test the difference between the 5G and 4G core we have performed user throughput measurements using both core implementations and varying specific RAN parameters which can differ between a 4G and 5G implementation such as the

Fig. 12: Comparison of user throughput with 4G and 5G core implementations and various RAN scheduling configurations.

overall system bandwidth and the subcarrier spacing. For the measurements provided with the Free5GC core [48] and the 5G OAI RAN [49], a 5G commercial device [50] as a user has been utilized. In total 4 different implementation setups have been used to perform measurements with respect to the user throughput as described below:

- Setup 1: OAI 4G core with 5 MHz system bandwidth and 15 KHz subcarrier spacing, resulting in 25 PRBs.
- Setup 2: OAI 4G core with 10 MHz system bandwidth and 15 KHz subcarrier spacing, resulting in 50 PRBs.
- Setup 3: Free5GC 5G core with 20 MHz system bandwidth and 30 KHz subcarrier spacing, resulting in 51 PRBs.
- Setup 4: Free5GC 5G core with 40 MHz system bandwidth and 30 KHz subcarrier spacing, resulting in 106 PRBs.

Fig. 12 demonstrates the overall user throughput in the specific system. As can be observed while comparing the measurements provided with the 4G core implementation of OAI used in this work with the 5G core implementation with a similar PRB count (i.e., Setup 1 with Setup 2) the overall user throughput is almost identical, around 33 Mbps. This again confirms that for a similar RAN setup for the throughput measurements the core network choice does not impact the results in RAN. However, while taking advantage of the 5G implementation, such as increasing the system bandwidth (i.e., Setup 4) the throughput increases drastically,

achieving approximately $120\,\mathrm{Mbps}$. Thus, highlighting the benefit of utilizing 5G compared to 4G in terms of increasing the possible achievable throughput in a system. Additionally, the advantage of 5G over 4G is mainly observed in the flexibility with respect to the subcarrier spacing choice which can decrease significantly the transmission delay, however, since in this work we only demonstrate results for throughput this becomes obsolete.

VI. DISCUSSION & CONCLUSION

In this paper, we investigated the impact of the imperfections of the SD-RAN control plane on the data plane. More specifically, we demonstrated the throughput degradation of UEs in the data plane as a function of a) assumed control packet losses, and b) SD-RAN emulated traffic on the controller. For the former, we provided mathematical models for 2 and 3 BSs to predict the throughput. We ran extensive simulations and measurements with input parameters from real-life traces to verify our analytical models. While considering two resource allocation policies, namely maxCQI-maxCQI and RR-RR, we showed how much the control plane packet losses influence the throughput of individual UEs, and hence the system throughput.

By and large, the results from the proposed mathematical model match those from simulations and measurements, proving the validity of our work. Moreover, the insights we provide a hint that when a maxCQI-maxCQI allocation policy is used, a high load on the network control significantly deteriorates the throughput (by almost 20%). Contrary to that, the RR-RR policy is robust to such effects. On the other hand, the maxCQI-maxCQI policy achieves higher overall throughput. Therefore, depending on the network conditions, the appropriate resource allocation scheme should be used.

This work provides the first step towards performance prediction in SD-RAN networks while demystifying important aspects of the system with high load. Although the presented results are based on research-related SD-RAN controllers that are not built for high performance (in terms of the number of UEs and BSs being capable to serve), we can use the equations provided by `Delphi` for other controllers too. Moreover, we want to stress that even though a higher performance controller might be used, the main drawbacks of a single SD-RAN controller do not simply occur due to the hardware/software or controller implementation. Instead they are also exhibited due to the additional tasks that an SD-RAN controller has to process as well as the critical RAN scheduling procedure that has to be performed in ms basis. The main benefit lies in a mathematical method for throughput computation that can identify the point of throughput degradation (i.e., the number of UEs and BSs that cause control packet losses). Consequently, a potential solution would be to introduce admission control mechanisms or distributed control planes for SD-RAN controllers. Finally, while our first analysis is based on the maxCQI-maxCQI policy, as the policy that can achieve the highest throughput in an SD-RAN environment, it does not provide the best results in terms of user fairness because it penalizes those users and BSs that do not experience high

CQIs. Therefore, investigating the performance of SD-RAN with respect to alternative policies, for instance, proportional fairness policy, becomes interesting and `Delphi` can be utilized in that regard. In order to provide an initial performance evaluation of the proportional fair theoretical analysis in SD-RAN environments, we refer the interested reader to our work [51], where we demonstrate the achieved throughput among UEs in an SD-RAN environment compared to a traditional non-SD-RAN setup. Compared to the results shown in this work with respect to the maxCQI-maxCQI policy, naturally, the proportional fair policy is expected to achieve less overall throughput among UEs. However, it eliminates the cases where some UEs may starve when no resources are provided at all given their low CQIs. Yet, the concrete results and comparison of different resource allocation policies apart from those presented in this paper are left as future work as this is beyond the scope of our work here.

An alternative avenue of research concerns the development and evaluation of other resource allocation policies which may as well be prone to sensitivity against packet losses using `Delphi`. Given the high importance of energy efficiency and low latency in 5G and emerging 6G networks, it is of utmost importance that resource allocation policies that cater for such objectives are further investigated. In order to support future research in this direction, `Delphi` can help in both analytical and measurement directions. In principle, additional resource allocation policies require adaptation of the theoretical model of `Delphi`. Consequently, an analysis will have to be provided with respect to the policy of choice, similar to the throughput maximization analysis provided in this work. However, with respect to the measurement setup, only the resource allocation policy will have to be installed at the controller and BS without additional changes to the system itself. That means that any state-of-the-art policy with the objective of minimization of latency or maximization of energy efficiency can be utilized and its performance can be verified in a realistic setup with the help of `Delphi`.

Finally, while in this work we have identified the potential bottleneck of SD-RAN environments with a single SD-RAN controller, an alternative interesting research direction comprises the development of detection and recovery policies in the control plane to achieve a minimum percentage of lost or delayed control packets.

In general, with the current means, the SD-RAN controller is not able to detect a control packet being lost or delayed unless there is explicit signaling received by the BSs with respect to the achieved UE throughput. In that case, the SD-RAN controller can analyze the differences between the achieved and estimated throughput for a certain control policy and as such deduce that an issue might have occurred in the control place. In that regard, there are several mechanisms that a controller may use. For instance, the controller might decide to lower the frequency of statistics updates that it receives from the BSs with respect to UEs, or lower the frequency of control decisions that it sends to the BSs. Furthermore, the controller may decide that for some time it delegates the control towards the BSs in order to reduce its load.

Additionally, the introduction of a distributed control plane

with multiple SD-RAN controllers that share the load of the underlying network can be a potential solution. However, there is always a trade-off between optimality and complexity on the control plane, which is out of the scope of this paper.

REFERENCES

[1] 3GPP, "TR 21.915 V1.1.0; Technical Report; Summary of Rel-15 Work Items (Release 15)," 2019.

[2] X. Jin, L. E. Li, L. Vanbever, and J. Rexford, "Softcell: Scalable and flexible cellular core network architecture," in *Proceedings of the ninth ACM conference on Emerging networking experiments and technologies*, 2013, pp. 163–174.

[3] L. Cui, F. R. Yu, and Q. Yan, "When big data meets software-defined networking: SDN for big data and big data for SDN," *IEEE Network*, vol. 30, no. 1, pp. 58–65, 2016.

[4] I. Afolabi, T. Taleb, K. Samdanis, A. Ksentini, and H. Flinck, "Network slicing and softwarization: A survey on principles, enabling technologies, and solutions," *IEEE Communications Surveys & Tutorials*, vol. 20, no. 3, pp. 2429–2453, 2018.

[5] 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*, 2016, pp. 427–441.

[6] L. Bonati, S. D'Oro, S. Basagni, and T. Melodia, "Scope: an open and softwarized prototyping platform for nextg systems," in *Proceedings of the 19th Annual International Conference on Mobile Systems, Applications, and Services*, 2021, pp. 415–426.

[7] X. Foukas and B. Radunovic, "Concordia: teaching the 5G vRAN to share compute," in *Proceedings of the 2021 ACM SIGCOMM 2021 Conference*, 2021, pp. 580–596.

[8] G. Garcia-Aviles, A. Garcia-Saavedra, M. Gramaglia, X. Costa-Perez, P. Serrano, and A. Banchs, "Nuberu: Reliable ran virtualization in shared platforms," in *Proceedings of the 27th Annual International Conference on Mobile Computing and Networking*, 2021, pp. 749–761.

[9] D. Vallis, "5G breaks from proprietary systems, embraces open source RANs," https://www.5gtechnologyworld.com/5g-breaks-from-proprietary-systems-embraces-open-source-rans/, 2020.

[10] E. Coronado, S. N. Khan, and R. Riggio, "5G-EmPOWER: A software-defined networking platform for 5G radio access networks," *IEEE Transactions on Network and Service Management*, vol. 16, no. 2, pp. 715–728, 2019.

[11] Rakuten, "How elegant software can make 5G networks more resilient," https://rakuten.today/blog/5g-network-reliability-lightreading.html, 2020.

[12] O-RAN Alliance e.V., "Operator Defined Open and Intelligent Radio Access Networks," https://www.o-ran.org/, 2019.

[13] G. Liu and D. Jiang, "5G: Vision and requirements for mobile communication system towards year 2020," *Chinese Journal of Engineering*, vol. 2016, no. 2016, p. 8, 2016.

[14] E. J. Oughton, Z. Frias, S. van der Gaast, and R. van der Berg, "Assessing the capacity, coverage and cost of 5G infrastructure strategies: Analysis of the Netherlands," *Telematics and Informatics*, vol. 37, pp. 50–69, 2019.

[15] A. Papa, R. Durner, E. Goshi, L. Goratti, T. Rasheed, A. Blenk, and W. Kellerer, "MARC: On modeling and analysis of Software-Defined Radio Access Network controllers," *IEEE Transactions on Network and Service Management*, vol. 18, no. 4, pp. 4602–4615, 2021.

[16] M. Yang, Y. Li, D. Jin, L. Su, S. Ma, and L. Zeng, "OpenRAN: a software-defined ran architecture via virtualization," *ACM SIGCOMM computer communication review*, vol. 43, no. 4, pp. 549–550, 2013.

[17] A. Mohamed, O. Onireti, M. A. Imran, A. Imran, and R. Tafazolli, "Control-data separation architecture for cellular radio access networks: A survey and outlook," *IEEE Communications Surveys & Tutorials*, vol. 18, no. 1, pp. 446–465, 2015.

[18] A. Gudipati, D. Perry, L. E. Li, and S. Katti, "SoftRAN: Software defined radio access network," in *Proceedings of the second ACM SIGCOMM workshop on Hot topics in software defined networking*, 2013, pp. 25–30.

[19] I. F. Akyildiz, P. Wang, and S.-C. Lin, "SoftAir: A software defined networking architecture for 5G wireless systems," *Computer Networks*, vol. 85, pp. 1–18, 2015.

[20] M. Y. Arslan, K. Sundaresan, and S. Rangarajan, "Software-defined networking in cellular radio access networks: potential and challenges," *IEEE Communications Magazine*, vol. 53, no. 1, pp. 150–156, 2015.

[21] X. Foukas, M. K. Marina, and K. Kontovasilis, "Orion: Ran slicing for a flexible and cost-effective multi-service mobile network architecture," in *Proceedings of the 23rd annual international conference on mobile computing and networking*, 2017, pp. 127–140.

[22] R. Schmidt, M. Irazabal, and N. Nikaein, "FlexRIC: an SDK for next-generation SD-RANs," in *Proceedings of the 17th International Conference on emerging Networking EXperiments and Technologies*, 2021, pp. 411–425.

[23] O-RAN Working Group 3, "O-RAN Near-Real-time RAN Intelligent Controller Near-RT RIC Architecture 2.00, O-RAN.WG3.RICARCH-v02.00, Technical Specification," March, 2021.

[24] O-RAN Working Group 2, "O-RAN Non-RT RIC Architecture 1.0, O-RAN.WG2.Non-RT-RIC-ARCH-TS-v01.00, Technical Specification," July, 2021.

[25] O-RAN Working Group 3, "O-RAN Near-Real-time RAN Intelligent Controller Architecture and E2 General Aspects and Principles 2.00, O-RAN.WG3.E2GAP-v02.01, Technical Specification," July, 2021.

[26] A. Papa, P. Kutsevol, F. Mehmeti, and W. Kellerer, "Effects of SD-RAN control plane design on user Quality of Service," in *8th International Conference on Network Softwarization (NetSoft)*. IEEE, 2022, pp. 312–320.

[27] N. Nikaein, M. K. Marina, S. Manickam, A. Dawson, R. Knopp, and C. Bonnet, "OpenAirInterface: A flexible platform for 5G research," *ACM SIGCOMM Computer Communication Review*, vol. 44, no. 5, pp. 33–38, 2014.

[28] L. Bertizzolo, L. Bonati, E. Demirors, and T. Melodia, "Arena: A 64-antenna SDR-based ceiling grid testbed for sub-6 GHz radio spectrum research," in *Proceedings of the 13th International Workshop on Wireless Network Testbeds, Experimental Evaluation & Characterization*, 2019, pp. 5–12.

[29] K. R. Dandekar, S. Begashaw, M. Jacovic, A. Lackpour, I. Rasheed, X. R. Rey, C. Sahin, S. Shaher, and G. Mainland, "Grid software defined radio network testbed for hybrid measurement and emulation," in *16th Annual IEEE International Conference on Sensing, Communication, and Networking (SECON)*, 2019, pp. 1–9.

[30] J. Breen, A. Buffmire, J. Duerig, K. Dutt, E. Eide, A. Ghosh, M. Hibler, D. Johnson, S. K. Kasera, E. Lewis *et al.*, "Powder: Platform for open wireless data-driven experimental research," *Computer Networks*, vol. 197, p. 108281, 2021.

[31] L. Bonati, P. Johari, M. Polese, S. D'Oro, S. Mohanti, M. Tehrani-Moayyed, D. Villa, S. Shrivastava, C. Tassie, K. Yoder *et al.*, "Colosseum: Large-scale wireless experimentation through hardware-in-the-loop network emulation," in *IEEE International Symposium on Dynamic Spectrum Access Networks (DySPAN)*, 2021, pp. 105–113.

[32] D. Bega, A. Banchs, M. Gramaglia, X. Costa-Pérez, and P. Rost, "CARES: Computation-aware scheduling in virtualized radio access networks," *IEEE Transactions on Wireless Communications*, vol. 17, no. 12, pp. 7993–8006, 2018.

[33] J. A. Ayala-Romero, A. Garcia-Saavedra, M. Gramaglia, X. Costa-Perez, A. Banchs, and J. J. Alcaraz, "vrain: Deep learning based orchestration for computing and radio resources in vRANs," *IEEE Transactions on Mobile Computing*, 2020.

[34] "oai cn," https://gitlab.eurecom.fr/mosaic5g/mosaic5g/-/wikis/tutorials/oai-cn.

[35] "Open5GS," https://https://open5gs.org/.

[36] "flexran-rtc," https://gitlab.eurecom.fr/flexran/flexran-rtc.

[37] M. Polese, L. Bonati, S. D'Oro, S. Basagni, and T. Melodia, "Understanding O-RAN: Architecture, interfaces, algorithms, security, and research challenges," *arXiv preprint arXiv:2202.01032*, 2022.

[38] ——, "ColO-RAN: Developing machine learning-based xApps for open RAN closed-loop control on programmable experimental platforms," *arXiv preprint arXiv:2112.09559*, 2021.

[39] "FlexRAN northbound API documentation," https://mosaic5g.io/apidocs/flexran/.

[40] "mosaic5g-oai-sim," https://gitlab.eurecom.fr/oai/openairinterface5g/-/tree/mosaic5g-oai-sim.

[41] "l2 sim," https://gitlab.eurecom.fr/mosaic5g/mosaic5g/-/wikis/tutorials/l2-sim.

[42] D. Raca, D. Leahy, C. J. Sreenan, and J. J. Quinlan, "Beyond throughput, the next generation: a 5G dataset with channel and context metrics," in *Proceedings of the 11th ACM multimedia systems conference*, 2020, pp. 303–308.

[43] F. Mehmeti and T. L. Porta, "Analyzing a 5G Dataset and Modeling Metrics of Interest," in *Proceedings of IEEE MSN*, 2021.

[44] G. Ku and J. M. Walsh, "Resource allocation and link adaptation in LTE and LTE Advanced: A tutorial," *IEEE Communications Surveys & Tutorials*, vol. 17, no. 3, 2015.

[45] ETSI, "5G NR overall description: 3GPP TS 38.300 version 15.3.1 release 15," www.etsi.org, 2018, technical specification.

[46] F. Mehmeti and C. Rosenberg, "How expensive is consistency? Performance analysis of consistent rate provisioning to mobile users in cellular networks," *IEEE Transactions on Mobile Computing*, vol. 18, no. 5, 2019.

[47] F. Mehmeti and T. F. La Porta, "Reducing the cost of consistency: Performance improvements in next generation cellular networks with optimal resource reallocation," *IEEE Transactions on Mobile Computing*, vol. 21, no. 7, 2022.

[48] "Free5GC," https://github.com/free5gc/free5gc/tree/v3.2.1.

[49] "OAI 5G RAN," https://gitlab.eurecom.fr/oai/openairinterface5g/-/tree/2023.w15.

[50] "Quectel 5G RM50xQ series," https://https://www.quectel.com/product/5g-rm50xq-series.

[51] F. Mehmeti and W. Kellerer, "Proportionally Fair Resource Allocation in SD-RAN," in *Proc. of IEEE Consumer Communications & Networking Conference (CCNC)*, 2023.

Wolfgang Kellerer (M'96, SM'11) is a Full Professor with the Technical University of Munich (TUM), Germany, heading the Chair of Communication Networks at the School of Computation, Information and Technology. He received his Ph.D. degree in Electrical Engineering from the same university in 2002. He was a visiting researcher at the Information Systems Laboratory of Stanford University, CA, US, in 2001. Prior to joining TUM, Wolfgang Kellerer pursued an industrial careerwas being for over ten years with NTT DOCOMO's European Research Laboratories. He was the director of the infrastructure research department, where he led various projects for wireless communication and mobile networking contributing to research and standardization of LTE-A and 5G technologies. In 2015, he has been awarded with an ERC Consolidator Grant from the European Commission for his research on flexibility in communication networks. He currently serves as an associate editor for IEEE Transactions on Network and Service Management and as the area editor for network virtualization for IEEE Communications Surveys and Tutorials.

Arled Papa completed his Bachelor of Science in Electronics Engineering at the Polytechnic University of Tirana, Albania in 2015. He received his Master of Science in Communications Engineering at the Technical University of Munich (TUM) in November 2017 with high distinction. In February 2018 he joined the Chair of Communication Networks at TUM as a research and teaching associate. His research focuses on the design and analysis of QoS, Network Slicing and Programmability of Software-Defined Radio Access Networks.

Polina Kutsevol received the B.Sc. degree in applied mathematics and physics from Moscow Institute of Physics and Technology, Moscow, Russia, in 2019, and the M.Sc. degree in communication engineering from Technical University of Munich, Munich, Germany, in 2021, where she is currently pursuing the Ph.D. degree with the Chair of Communication Networks. Her current research interests include resource management for wireless and mobile communication networks, cyber-physical systems and networked control systems.

Fidan Mehmeti received a graduate degree in Electrical and Computer Engineering from the University of Prishtina, Kosovo, in 2009. He obtained his Ph.D. degree in 2015 at Institute Eurecom/Telecom ParisTech, France. After that, he was a Post-doctoral Scholar at the University of Waterloo, Canada, North Carolina State University and Penn State University, USA. He is now working as a Senior Researcher and Lecturer at the Technical University of Munich, Germany. His research interests lie within the broad area of wireless networks, with an emphasis on performance modeling, analysis, and optimization.