

NeutRAN: An Open RAN Neutral Host Architecture for Zero-Touch RAN and Spectrum Sharing

Leonardo Bonati, *Member, IEEE*, Michele Polese, *Member, IEEE*, Salvatore D’Oro, *Member, IEEE*, Stefano Basagni, *Senior Member, IEEE*, Tommaso Melodia, *Fellow, IEEE*



Abstract—Obtaining access to exclusive spectrum, cell sites, Radio Access Network (RAN) equipment, and edge infrastructure imposes major capital expenses to mobile network operators. A neutral host infrastructure, by which a third-party company provides RAN services to mobile operators through network virtualization and slicing techniques, is seen as a promising solution to decrease these costs. Currently, however, neutral host providers lack automated and virtualized pipelines for onboarding new tenants and to provide elastic and on-demand allocation of resources matching operators’ requirements. To address this gap, this paper presents NeutRAN, a zero-touch framework based on the O-RAN architecture to support applications on neutral hosts and automatic operator onboarding. NeutRAN builds upon two key components: (i) an optimization engine to guarantee coverage and to meet quality of service requirements while accounting for the limited amount of shared spectrum and RAN nodes, and (ii) a fully virtualized and automated infrastructure that converts the output of the optimization engine into deployable micro-services to be executed at RAN nodes and cell sites. NeutRAN was prototyped on an OpenShift cluster and on a programmable testbed with 4 base stations and 10 users from 3 different tenants. We evaluate its benefits, comparing it to a traditional license-based RAN where each tenant has dedicated physical and spectrum resources. We show that NeutRAN can deploy a fully operational neutral host-based cellular network in around 10 seconds. Experimental results show that it increases the cumulative network throughput by $2.18\times$ and the per-user average throughput by $1.73\times$ in networks with shared spectrum blocks of 30 MHz. NeutRAN provides a $1.77\times$ cumulative throughput gain even when it can only operate on a shared spectrum block of 10 MHz (one third of the spectrum used in license-based RANs).

Index Terms—O-RAN, Open RAN, Neutral Host, Automation, 5G, 6G.

The authors are with the Institute for the Wireless Internet of Things, Northeastern University, Boston, MA, USA. E-mail: {l.bonati, m.polese, s.doro, s.basagni, melodia}@northeastern.edu.

This article is based upon work partially supported by the U.S. National Science Foundation under grants CNS-1925601, CNS-2112471, and CNS-1923789, by the U.S. Office of Naval Research under grant N00014-20-1-2132, and by OUSD(R&E) through Army Research Laboratory Cooperative Agreement Number W911NF-19-2-0221. The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the Army Research Laboratory or the U.S. Government. The U.S. Government is authorized to reproduce and distribute reprints for Government purposes notwithstanding any copyright notation herein.

1 INTRODUCTION

The need for higher data rates and reduced latency in cellular networks has resulted in unprecedented network densification [1], and in new deployment models where private operators deploy dedicated cellular infrastructure [2]. As a consequence, access to spectrum, cell site facilities (i.e., poles, towers), Radio Access Network (RAN) and edge equipment [3] represents the largest share of capital and operational expenses faced by public and private operators [4].

However, higher costs—which frequently come with lower profits—are barriers to technological innovation for the future of cellular networks. To lower these barriers, renting a *neutral host infrastructure* from a third-party company that leases physical resources (e.g., spectrum, towers, RAN) to multiple operators on a shared-tenant basis is seen as a promising solution as it enables resource sharing and decreases the overall infrastructure costs [5]. Examples of commercial neutral host deployments are reported in several analyses and reports [6, 7] and, according to [8], there is evidence showing that the neutral host market will reach \$9.56 billions by 2028. Similar to what is happening in the infrastructure domain, *spectrum sharing* has also been identified as an effective way to increase the overall spectral utilization [9]. In fact, recent estimates indicate that joint adoption of neutral host models and spectrum sharing techniques can potentially lead to savings of at least 30% on network operational costs in the next five years [10].

RAN and spectrum sharing, however, are not ready for prime time in multi-operator network deployments [11], especially because they still lack mechanisms to enable (i) *fine-grained sharing*, with multiple tenants sharing compute and spectrum slices from the same physical infrastructure, and (ii) *dynamic sharing* that allows infrastructure owners to fully leverage the statistical multiplexing of RAN and spectrum resources, and to tailor infrastructure parameters to tenant requirements that may change in a matter of seconds. For example, spectrum sharing in the CBRS band operates over time scales in the order of minutes [12], limiting the flexibility of the system and spectrum utilization efficiency.

Obstacles to further progress are both technological and strategic in nature, and include the following.

- *Lack of automated and virtualized pipelines for multi-tenant management.* Zero-touch, resilient, fault-tolerant automation frameworks are currently unavailable in RAN environments. These functionalities are necessary to ensure reliability and proper coordination among multiple tenants that dynamically share infrastructure and spectrum, without manual intervention and over-provisioning [13].
- *Lack of timely management of the life cycle of network services.* The dynamic allocation of spectrum and RAN infrastructure resources in a timely fashion is still a challenge, considering that complex software services, such as softwarized 5G Next Generation Node Bases (gNBs), need to be instantiated in a matter of seconds. This is because of the lack of low-latency end-to-end pipelines that interface with, keep track of, and coordinate available RAN elements and resources, and that manage the life cycle of network services [14–16].
- *Operators' perception of resource sharing as a risk.* Due to the absence of reliable sharing solutions that can support Service Level Agreements (SLAs) through dynamic, fine-grained resource allocation provided through optimization engines [17], operators perceive spectrum sharing as a risk [18] and, therefore, prefer exclusive spectrum licensing.

1.1 Novelty and Contribution

This paper addresses these challenges and takes a fundamental step toward enabling *zero-touch dynamic and fine-grained RAN and spectrum sharing* by introducing NeutRAN. NeutRAN is a neutral host framework that automatically manages the deployment of services on shared RAN and spectrum resources, based on high-level intents and requests from multiple tenants. The core innovation of NeutRAN is the design and prototyping of end-to-end pipelines that combine (i) a virtualized and automated RAN infrastructure with (ii) an optimization engine that takes decisions regarding RAN and spectrum sharing policies. The NeutRAN framework, which we develop on top of the O-RAN architecture, shows for the first time how virtualization and automation can be extended to a multi-tenant RAN, providing a fully managed and effective solution for private and public neutral host-based deployments. This allows NeutRAN to break the traditional, isolated spectrum and infrastructure silos and to bring *dynamic, fine-grained statistical multiplexing to the RAN and spectrum*.

The main contributions of this work include the following. 1) *NeutRAN zero-touch framework and automation pipelines.* We define, develop, and prototype NeutRAN over state-of-the-art tools for future cellular network innovation, which include OpenShift, Kubernetes, O-RAN and Software-defined Radios (SDRs). As an open and virtualized framework, NeutRAN enables the deployment of complex, customized core and RAN micro-services in a matter of seconds (e.g., 9.55 s for a gNB) from a centralized Service Management and Orchestration (SMO) entity to edge datacenters and cell sites that are part of the O-RAN O-Cloud. NeutRAN manages 5G disaggregated gNBs as well as the deployment of O-RAN RAN Intelligent Controllers (RANs) and their custom logic units to satisfy tenant requests. These logic units, namely xApps if deployed on the near-real-time (near-RT) RIC and rApps on the non-real-time (non-RT) RIC, get run-time Key Performance Measurements (KPMs) from the RAN nodes,

and can control their functionalities through data-driven techniques acting at different time scales [19].

2) *NeutRAN optimization engine rApp.* We develop an optimization engine that provides guarantees for (i) the execution of latency-critical compute tasks (e.g., a gNB) in shared infrastructures, and for (ii) the Quality of Service (QoS) and SLAs that tenants require for their users. The engine is based on the efficient solution of the *neutral host problem*, which considers tenant requests, available resources, and network analytics to generate an optimal allocation of micro-services and spectrum resources (e.g., spectrum slices). The problem is modeled as a binary Quadratically Constrained Quadratic Programming (QCQP) optimization problem that is solved optimally via reformulation-linearization techniques. The NeutRAN optimization engine is deployed as an rApp in the SMO, and leverages O-RAN interfaces to gather data and analytics on the RAN performance and to deploy the NeutRAN services.

3) *Scalability, efficiency, and experimental evaluation.* We show that NeutRAN computes optimal solutions in less than 2 s for large-scale networks, allocating resources that meet the tenant requirements. We also run experiments on a testbed with 4 softwarized cell sites and 10 commercial User Equipments (UEs) from 3 different tenants. We compare NeutRAN against a license-based, siloed RAN where operators control their own infrastructure and spectrum (10 MHz each; 30 MHz total). Our results show that NeutRAN manages the total (now shared) bandwidth obtaining $2.18 \times$ RAN throughput and $1.73 \times$ average user throughput gains, with a consistent improvement in Signal to Interference plus Noise Ratio (SINR) over the license-based approach. We also show that, when NeutRAN is deployed on a total and shared bandwidth that is one third of the previous configurations (10 MHz), the optimized resource allocation at cell sites offsets the reduction in available spectrum, delivering an improvement in RAN throughput of $1.77 \times$ and unchanged average user throughput. This shows how the combination of virtualization, automation, and optimization to manage RAN and spectrum sharing brings remarkable network and user throughput gains.

The remainder of this paper is organized as follows. Section 2 reviews the state of the art on neutral host applications, RAN and spectrum sharing. Section 3 presents NeutRAN. Section 4 illustrates the neutral host problem and presents NeutRAN optimization engine. Section 5 describes the NeutRAN prototype implementation, and Section 6 profiles NeutRAN performance both via numerical and experimental results. Finally, Section 7 concludes the paper.

2 RELATED WORK

The NeutRAN concept lies at the intersection of solutions for neutral host architectures, spectrum and RAN sharing, and controllers for the virtualized RAN, all fields that have garnered the interest of researchers from industry and academia alike.

Considering neutral host and virtualized architectures for RAN and spectrum sharing, Kibria et al. study the business models enabled by micro-operators sharing neutral host architectures [20], while Di Pascale et al. leverage the blockchain to enforce SLAs for neutral host infrastructure

providers [21]. Incentives for cooperation among operators are analyzed by Vincenzi et al. [22]. Architectural enhancements for neutral hosts are presented by Sarakis et al. [23] (on cost-efficient private networks) and by Paolino et al. [24] (on security for network virtualization). Despite considering both neutral host and virtualization paradigms, the above works are either focused on the business aspects of such architectures, on cooperation incentives between tenants, or on architectural enhancements.

Multi-operator solutions for spectrum and RAN sharing have also been thoroughly investigated. Sharing for the coexistence of Internet of Things (IoT) devices and cellular services is investigated by Xiao et al. [25], and Qian et al. [26]. Lin et al. use RAN proxies to share base stations among core networks [27]. Giannone et al. focus on the deployment of virtualized Distributed Units (DUs) [28]. Samdanis et al. propose a mechanism for tenants to lease resources from infrastructure providers [5]. Kasgari and Saad design an optimization and control framework for isolation of multiple slices and for the reduction of their power consumption [29]. Wang et al. propose a hybrid slice reconfiguration framework to optimize allocated resources with a focus on the profit of each slice [30]. Bega et al. design a Machine Learning (ML) framework for the reallocation of slice resources and revenue maximization based on the forecast network capacity in [31]. Dynamic slice sharing is evaluated by Foukas et al. in [32], by Caballero et al. in [33, 34], and by D’Oro et al. in [35], who also evaluate the sharing of compute and storage resources in [36]. Puligheddu et al. propose an O-RAN-based semantic framework to allocate resources to ML classifiers based on the requirements of the task to perform at the edge [37]; their focus is on computer vision applications. Baldesi et al. propose a mechanism for dynamic adaptation based on incumbents in the shared spectrum [38]. While focusing on sharing network resources, these works propose solutions that are either not tailored to large deployments, do not adapt to network dynamics, or do not consider the automated deployment of micro-services, all functionalities offered by NeutRAN. Similarly, several papers focus on admission control for slicing [39–41], but do not consider the dynamics and ad hoc deployment of RAN base stations as micro-services.

Controllers for virtualized RANs and abstractions among the physical and virtual infrastructure are discussed by Schmidt et al. [42]. Moorthy et al. study the virtualization of control-plane functionalities [43]. Foukas et al. devise a slicing system for the dynamic virtualization of base stations [44] and showcase a neutral-host use case in [45]. Garcia-Aviles et al. focus on synchronization issues among shared DUs and their UEs in case of lack of computational capacity [13]. Even though these solutions provide enhanced and automated network control, they either do not focus on the automated instantiation of RAN functions as micro-services, or do not tackle neutral host architectures specifically, or focus on RAN slicing applications that do not consider the availability of multiple spectrum bands.

Compared to the listed works, NeutRAN proposes an end-to-end solution for the zero-touch automated and rApp-based optimized allocation and deployment of micro-services in a neutral host architecture based on tenant intents. This is achieved by computing the optimized al-

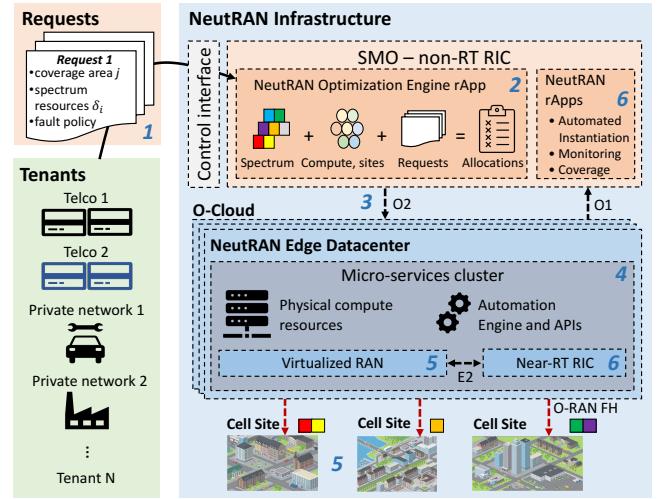


Fig. 1: The NeutRAN framework. The numbering represents the six steps of the NeutRAN workflow.

location of RAN functions starting from intents, and by automatically instantiating such functions as micro-services on edge datacenters managed by enterprise-ready platform-as-a-service frameworks such as OpenShift.

3 THE NEUTRAN FRAMEWORK

NeutRAN combines an O-RAN-based softwarized and automated infrastructure and an optimization engine for practical and efficient RAN and spectrum sharing. A bird’s-eye view of the NeutRAN framework is shown in Figure 1.

The NeutRAN stakeholders are the tenants, who want to provide services to their end users, and the NeutRAN operator, who owns the infrastructure and provides access to the automated RAN and spectrum sharing pipelines. Tenants access a high-level control interface to submit requests to deploy cellular connectivity in certain areas. Based on the available physical resources, these requests are then automatically converted into a set of virtualized networking services and functionalities deployed by NeutRAN on edge datacenters. Tenants have different targets, and upon instantiation of the cellular network, they might also require the instantiation of a near-RT RIC to execute a catalog of xApps tailored to their needs. This section describes NeutRAN in detail, starting from the O-RAN architecture used as supporting infrastructure, then describing the NeutRAN components and the procedures of its automated workflow.

An O-RAN Primer. O-RAN is a disaggregated approach to deploy mobile cellular networks built upon cloud-native principles. It introduces standardized interfaces that facilitate interoperability among disaggregated network elements (e.g., Central Unit (CU)/DU/Radio Unit (RU)), and RICs to oversee and fine-tune the functionalities of the network [46]. RICs operate at different time scales to enable data-driven closed-control loops and network management through custom applications, called rApps (for the *non-RT* RIC) and xApps (in the *near-RT* RIC).¹ These applications

1. The extension of control and inference to *real-time* time scales (i.e., below 10 ms) is possible via *dApps* [47]. These are applications deployed directly on the RAN nodes, e.g., CUs/DUs, and that can access sensitive data, e.g., I/Q samples, that cannot be streamed out of such nodes because of privacy concerns, among others.

receive live KPMs from the RAN, and adapt its configuration to the dynamic channel conditions and traffic demand. O-RAN also enables the deployment of virtualized services for the RAN in a pool of compute resources (the O-Cloud) managed by the SMO (Figure 1), a centralized component deployed in a cloud facility [48]. The SMO provides an abstract view of the network infrastructure and resources (e.g., compute, spectrum, coverage) obtained by using the O-RAN O1 interface. It also triggers new service deployment and updates through the O2 interface, which connects the SMO to virtualization resources in the O-Cloud. The SMO hosts the O-RAN non-RT RIC and its rApps.

NeutRAN components. NeutRAN consists of three main architectural components: (i) an SMO; (ii) edge datacenters, and (iii) cell sites. The SMO in the NeutRAN architecture includes an instance of non-RT RIC with an rApp implementing the optimization engine (see Section 4). Inputs to the engine include tenant requests and analytics from the RAN gathered by a monitoring rApp. Requests from the NeutRAN tenants are then matched into services to be deployed in the O-Cloud through the OpenShift Kubernetes Application Programming Interfaces (APIs). Other NeutRAN rApps automate service instantiation, and infrastructure monitoring for self-healing purposes. Last, a coverage rApp monitors historical and current coverage data to identify areas covered by each cell site (a key step in the optimization process, as described in Section 4).

The NeutRAN *edge datacenters* are illustrated in Figure 2. As part of the O-Cloud resource pool, they are implemented

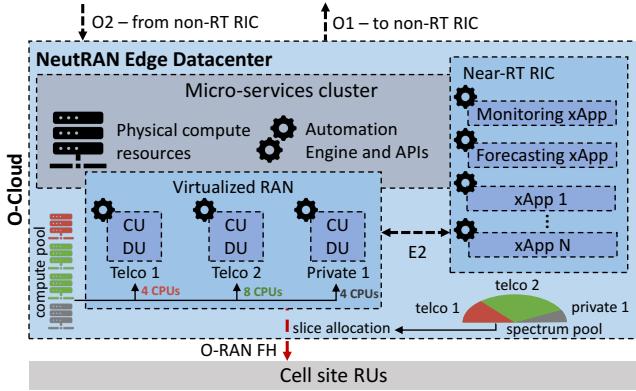


Fig. 2: The NeutRAN edge datacenter, with a pool of shared compute and spectrum resources. The NeutRAN pipelines define how resources are sliced into RAN services and spectrum allocations for the tenants.

through OpenShift, the open-source enterprise-ready hybrid cloud platform-as-a-service framework by Red Hat [49]. OpenShift leverages containerized virtualization technologies managed by Kubernetes [50] to instantiate applications and workloads in the form of containers (or *pods*) on top of white-box compute machines. This framework also offers primitives and APIs to instantiate and manage the life cycle of custom workloads on top of the managed infrastructure. Typical workloads include the virtualized core network and RAN, with the CUs and DUs for different tenants, the near-RT RIC, connected to the virtualized RAN through the E2 interface, and the xApps running on the near-RT RIC. NeutRAN xApps aid and augment the operations of the optimization engine by providing additional monitoring and forecasting of user demand and resource utilization.

The combination of custom OpenShift pipelines and of the directives from the optimization engine rApp enable efficient slicing of the edge datacenter resources, e.g., compute for NeutRAN services and spectrum to be used by the CUs and DUs (Figure 2). As we will discuss in Section 5, automated pipelines in the NeutRAN edge datacenters can start customized network services and workloads in less than 10 s. The edge datacenters also expose Flask REST APIs to let the monitoring rApp query the current resource availability (e.g., compute, spectrum) in the OpenShift clusters.

NeutRAN datacenters are connected to multiple *cell sites* in specific geographic areas via high-speed fiber connections, e.g., the O-RAN Fronthaul (FH) interface. Cell sites host RUs and antennas to provide RAN access over multiple frequency bands.

The NeutRAN Automated Workflow. The NeutRAN end-to-end automated workflow is structured in six steps, shown in Figure 1. In *Step 1*, tenants submit their *requests* to NeutRAN through intents that describe the services required by the tenant (e.g., to cover a specific geographic area in a certain period of time), needed resources (e.g., spectrum), and fault-recovery policies (e.g., whether to reinstantiate services in case of failure). These requests, together with spectrum and infrastructure availability, are the input to the rApp in the SMO that implements the optimization engine (*Step 2*), whose outputs are allocation policies sent to the edge datacenters through the O2 interface in *Step 3*. These policies specify the spectrum (carrier frequency and bandwidth) allocated to each tenant, together with the cell sites and compute resources in the edge datacenters. In *Step 4*, the NeutRAN edge datacenter shown in Figure 2 uses automated pipelines programmed through OpenShift to dispatch services such as the CUs/DUs, core network, and near-RT RICs required by the tenants, together with the xApps in the tenant’s catalog. Upon instantiation, RAN services automatically connect to the core network and near-RT RIC running in the edge datacenter, report run-time KPMs to the RIC through the O-RAN E2 interface, and expose functionalities the xApps can subscribe to. Note that, based on the optimal allocation policy, multiple tenants can share the same base station. In this case, spectrum is shared by means of RAN slicing, where each tenant is assigned a different slice of the network (e.g., a subset of the available bandwidth). NeutRAN allocations are not elastic, i.e., between consecutive iterations of *Step 3*, which happen with a non-real-time periodicity, there is a rigid allocation of spectrum resources for each tenant. However, it is possible to combine NeutRAN with a slicing xApp to dynamically change the allocation for tenants in the same cell site on a near-real-time loop. By *Step 5*, the services required by the tenants on shared RAN and spectrum are fully provisioned. After the instantiation of these micro-services, in *Step 6* the edge datacenters and SMO run monitoring xApps and rApps to perform health checks on the deployed micro-services and resources, and to recover them from potential failures, thus effectively making the RAN self-healing.

4 THE NEUTRAN OPTIMIZATION ENGINE RAPP

This section describes the NeutRAN optimization engine. The engine is implemented as an rApp (Figure 1) that

models the *neutral host problem* with data analytics from the SMO and RAN (Section 4.1). Within the rApp, NeutRAN defines the problem, its constraints (Section 4.2), and solves it (Section 4.3) via reformulation-linearization techniques, discussed in Section 4.4, where we also analyze in detail the problem complexity, and outline strategies for its reduction.

4.1 System Model

We consider the RAN infrastructure in Figure 1 with a set $\mathcal{R} = \{1, 2, \dots, B\}$ of B cell sites and NeutRAN edge datacenters. This infrastructure is offered to a set \mathcal{T} of T tenants according to the neutral host business model: each cell site can be leased to a tenant for a set amount of time to offer network access to their users. The deployment area is partitioned into a set $\mathcal{A} = 1, 2, \dots, A$ of A areas. Each cell site covers one or more areas. For each area $j \in \mathcal{A}$ and cell site $b \in \mathcal{R}$, the indicator variable $c_{b,j} \in \{0, 1\}$ is $c_{b,j} = 1$ if b provides coverage to j , and $c_{b,j} = 0$ otherwise. We do not make assumptions on how indicator variables $c_{b,j}$ are computed, as different deployments may determine coverage with different policies. However, these can be used to enforce minimum QoS requirements to ensure that a base station b is considered to be “covering” a certain area i only if it can deliver a satisfactory performance level to users deployed therein. For example, a realistic approach could set $c_{b,j} = 1$ if and only if a metric $\gamma_{b,j}$ (e.g., SINR, throughput or channel quality) at cell site b for any user in area j exceeds a minimum tolerable value γ^{\min} , where both $\gamma_{b,j}$ and γ^{\min} can be obtained via historical data from the monitoring and coverage rApps (Section 3). Let $\mathcal{A}_b \subseteq \mathcal{A}$ be the set of areas covered by $b \in \mathcal{R}$.

NeutRAN uses a set \mathcal{W} of W 5G frequency bands. Let \mathcal{F}_ω be the set of frequencies in band $\omega \in \mathcal{W}$. 5G systems rely upon an Orthogonal Frequency Division Multiplexing (OFDM) frame structure, which partitions frequencies into subcarriers. These are then organized into blocks of 12 to form the so-called Physical Resource Blocks (PRBs), which are the minimum units that can be scheduled in frequency. As a consequence, we discretize the set \mathcal{F}_ω , which is the set of PRBs in band ω . Therefore, $\bigcup_{\omega=1}^W \mathcal{F}_\omega = \mathcal{F}$.

Each cell site can operate across multiple bands. Thus, we introduce a variable $\beta_{b,f} \in \{0, 1\}$ such that, for any band ω and cell site b , $\beta_{b,f} = 1$ for all $f \in \mathcal{F}_\omega$ if b can operate on band ω . We also introduce the indicator $\xi_b \in \{0, 1\}$, such that $\xi_b = 1$ if cell site b can transmit on a single band (among the bands supported by the cell site) at any given time.

We consider the case where each tenant in \mathcal{T} can submit requests to provide wireless services at different locations. Specifically, each tenant t in \mathcal{T} generates a set \mathcal{I}_t of requests that are then collected into a set $\mathcal{I} = \bigcup_{t \in \mathcal{T}} \mathcal{I}_t$ with a total of I requests. As shown in Figure 1, each request $i \in \mathcal{I}$ specifies the area $j \in \mathcal{A}$ where the service is needed, the required amount of resources δ_i , and the level of fault resiliency. Without loss of generality, we consider a one-to-one mapping between a request i and its associated area j . δ_i represents the amount of PRBs required to accommodate the request. Its value might depend on a variety of factors such as number of users, their type (e.g., best-effort, premium), the type of traffic they generate (e.g., video streaming, browsing) as well as any SLAs in place between tenants

and their customers. We consider two cases. In the first case, δ_i is submitted directly together with request i by the tenants (either by a human or by an intelligent component, e.g., a request forecasting rApp) hosted in the SMO or in the non-RT RIC. This demand might depend on market and business strategies, and its value is usually kept undisclosed out of privacy concerns. As such, we do not make any assumptions on how tenants compute the value of δ_i , and design NeutRAN to let the tenants keep the models used to compute δ_i undisclosed, having only to specify its value. In the second case, we follow an approach similar to that in [40] where operators specify the type of traffic, slice and corresponding SLA levels only, and NeutRAN can predict and compute the amount of resources δ_i that are required to satisfy such request.

Without loss of generality, we consider the case where requests in \mathcal{I} are collected by NeutRAN following a timeslot-based approach. Specifically, tenants are allowed to submit their request at any given time, but NeutRAN will process such requests simultaneously every Δ seconds. As a consequence, the outstanding requests are stored into a buffer (i.e., the outstanding request set \mathcal{I}). Every Δ seconds, NeutRAN computes an optimal solution and removes the satisfied requests from the outstanding set \mathcal{I} . Tenants whose requests have not been selected can decide to either keep their request in the buffer or remove them. As we will show in Section 5, it takes about 9 seconds to activate a base station, while in Section 6.1 we will show how NeutRAN can compute optimal solutions within a few couple of seconds. Therefore, the duration of Δ in most cases is lower-bounded by the time to activate RAN components rather than the time to compute an optimal solution.

As cell sites offer limited coverage, we define two sets \mathcal{R}_i and \mathcal{R}_{-i} to represent the cell sites that offer coverage to the area j specified by request i and those that do not, respectively. These sets are defined as $\mathcal{R}_i = \{b \in \mathcal{R} : r_{i,b} = 1\} \subseteq \mathcal{R}$ and $\mathcal{R}_{-i} = \mathcal{R} \setminus \mathcal{R}_i$, where $r_{i,b} \in \{0, 1\}$ is a variable used to determine whether or not cell site b is a suitable candidate to accommodate request i . Specifically, for any request i and its required area j , we have that $r_{i,b} = 1$ if and only if $j \in \mathcal{A}_b$. Finally, we also introduce a parameter w_i that can be used to model the value of request i . This adds flexibility to NeutRAN, as it can be used by tenants to declare the monetary value of the requests, and by NeutRAN to prioritize profits over infrastructure utilization.

4.2 Problem Definition

NeutRAN is designed to enable neutral host applications for Open RAN cellular systems offering at the same time an automated and optimized platform to (i) instantiate disaggregated 5G gNBs; (ii) allocate spectrum on-demand; (iii) avoid interference; and (iv) satisfy tenant requests.

Let $\mathbf{y} = (y_{i,b})_{i \in \mathcal{I}, b \in \mathcal{R}}$, and $\mathbf{x} = (x_{i,b,f})_{i \in \mathcal{I}, b \in \mathcal{R}, f \in \mathcal{F}}$ be our optimization variables. In our formulation, variable $y_{i,b} \in \{0, 1\}$ is set to 1 if request i is assigned to cell site b ; $y_{i,b} = 0$ otherwise. Variable $x_{i,b,f} \in \{0, 1\}$ indicates which PRBs on cell site b have been allocated to request i , namely, $x_{i,b,f} = 1$ indicates that PRB f in cell site b is assigned to request i . The constraints and objective of the optimization are as follows.

4.2.1 Avoid conflicts and spectrum over-provisioning

First, each request i can be allocated to one cell site b only to ensure that the infrastructure owner does not incur additional costs by instantiating many cell sites to satisfy the same request. For all $i \in \mathcal{I}$, this is formulated as follows:

$$\sum_{b \in \mathcal{R}} y_{i,b} \leq 1 \quad (1)$$

Furthermore, we must also ensure that NeutRAN (i) does not allocate more PRBs than available; (ii) avoids conflicts by not allocating the same PRB to multiple requests (e.g., from different tenants); and (iii) mitigates interference by making sure neighboring cells do not operate over the same spectrum.

For each b and $b' \in \mathcal{R}$, let the interference indicator $I_{b,b'} \in \{0, 1\}$ be such that $I_{b,b'} = 1$ if b and b' have overlapping coverage regions and interfere with each other if using the same spectrum. For all $f \in \mathcal{F}$, and $b, b' \in \mathcal{R}$ with $I_{b,b'} = 1$, the following two inequalities model constraints (i)+(ii), and constraint (iii), respectively.

$$\sum_{i \in \mathcal{I}} x_{i,b,f} \leq \beta_{b,f} \quad (2)$$

$$\sum_{i \in \mathcal{I}} (x_{i,b,f} \beta_{b,f} + x_{i,b',f} \beta_{b',f}) \leq 1 \quad (3)$$

Note that the left-hand side of (2) ensures that only supported bands can be allocated, and that the total number of PRBs allocated to each cell site does not exceed the total number of available PRBs. This is because $F_b = \sum_{f \in \mathcal{F}} \beta_{b,f}$ indicates the number of PRBs available at b across all supported bands. Therefore, since $\beta_{b,f} \leq 1$, (2) enforces the allocation of no more than the available F_b PRBs.

4.2.2 Satisfy locality and spectrum demand

To satisfy request i , NeutRAN can select any cell site b that covers area j requested by i , i.e., $r_{i,b} = 1$, and must satisfy the spectrum demand δ_i by allocating enough PRBs to the request. These constraints can be defined jointly via (4) for all $i \in \mathcal{I}$ and $b \in \mathcal{R}$.

$$\sum_{f \in \mathcal{F}} x_{i,b,f} \beta_{b,f} r_{i,b} = \delta_i y_{i,b} \quad (4)$$

For all $i \in \mathcal{I}$ and $b \in \mathcal{R}$, the following set of constraints (5) forces to zero all variables that would result in an unfeasible solution where a request i is allocated to a cell site b that does not offer coverage to the required area, i.e., $b \in \mathcal{R}_{-i}$.

$$\sum_{b' \in \mathcal{R}_{-i}} y_{i,b'} + \sum_{b' \in \mathcal{R}_{-i}} \sum_{f \in \mathcal{F}} x_{i,b',f} = 0 \quad (5)$$

4.2.3 Enforce contiguous allocation

Allocating contiguous PRBs to requests makes it possible to implement the sharing mechanism through RAN slicing, and decreases the complexity of transceiver architectures, e.g., it eliminates the need for carrier aggregation [44]. For all $i \in \mathcal{I}$ and $b \in \mathcal{R}$, the contiguous allocation of PRBs can be enforced via the following constraint:

$$\sum_{f=1}^{F-1} x_{i,b,f} \cdot x_{i,b,f+1} \alpha_{f,f+1} = (\delta_i - 1) y_{i,b} \quad (6)$$

where $\alpha_{f,f'} \in \{0, 1\}$ is such that $\alpha_{f,f} = 0$ and $\alpha_{f,f'} = 1$ if f and f' belong to the same spectrum band ω , i.e., $(f, f') \in \mathcal{F}_\omega \times \mathcal{F}_\omega$, and they are consecutive, i.e., $f' = f \pm 1$. In this way, together with (4), (6) ensures that $y_{i,b} = 1$ if and only if we allocate exactly δ_i contiguous PRBs.

4.2.4 Support single band cell sites

Smaller cell sites (e.g., micro or picocells) may support multiple spectrum bands, but can transmit over a single spectrum band only at any given time. This must be enforced through equality (7) which, for all $b \in \mathcal{R}$ and $\omega \in \mathcal{W}$ such that $\xi_b = 1$ (i.e., those b that only support single band operations), ensures that only one band is allocated to requests at any given time.

$$\sum_{f \in \mathcal{F}_\omega} \sum_{i \in \mathcal{I}} x_{i,b,f} \cdot \sum_{f' \in \mathcal{F} \setminus \mathcal{F}_\omega} \sum_{i \in \mathcal{I}} x_{i,b,f'} = 0 \quad (7)$$

4.3 Problem Formulation

The neutral host optimization problem is as follows:

$$\begin{aligned} & \underset{\mathbf{x}, \mathbf{y}}{\text{maximize}} & & \sum_{i \in \mathcal{I}} \sum_{b \in \mathcal{R}} y_{i,b} w_i \\ & \text{subject to} & & \text{Constraints (1), (2), (3), (4), (5), (6), (7)} \\ & & & x_{i,b,f} \in \{0, 1\}, \quad \forall i \in \mathcal{I}, b \in \mathcal{R}, f \in \mathcal{F} \\ & & & y_{i,b} \in \{0, 1\}, \quad \forall i \in \mathcal{I}, b \in \mathcal{R} \end{aligned} \quad (8)$$

The objective of (8) is to accommodate as many requests as possible such that their cumulative value is maximized while satisfying the set of constraints discussed in Section 4.2. We notice that a specific instance of (8) is that where $w_i = 1$ for all $i \in \mathcal{I}$. In this case, the formulation would allow infrastructure owners to maximize the number of admitted requests rather than their cumulative value.

4.4 Complexity Analysis and Mitigation

The neutral host problem formalized above is a *binary Quadratically Constrained Quadratic Program* (QCQP), as the optimization variables \mathbf{x} and \mathbf{y} are 0-1 variables and constraints (6) and (7) are quadratic.

In general, even non-binary QCQPs are NP-hard [51], and their binary 0-1 version results in a non-convex optimization problem that inherits the NP-hardness of binary quadratic problems [52], which make binary QCQPs—such as problem (8)—NP-hard. Despite the exponential complexity of such problems, Semidefinite Programming Relaxations (SPR) [53] and Reformulation-Linearization Techniques (RLT) [54] have been shown to be effective tools for solving them optimally [55]. Both SPR and RLT transform the non-linearity from the product of any two variables z_n and z_m (such as those in (6) and (7)) into an auxiliary variable $Z_{n,m} = z_n z_m$. However, the two approaches differ in that each technique adds a different set of constraints to the problem. We refer the reader to [55] for a detailed comparison of the two approaches. In this paper, we solve problem (8) in Gurobi and use RLT, which has been shown to deliver faster computing time and higher accuracy than SPR in large-scale configurations [56].

The primary source of complexity of problem (8) stems from the number of variables N of the problem, i.e.,

$N = N_x + N_y = IBF + IB \in \mathcal{O}(IBF)$. Note that, while the number of requests I and the number B of cell sites might be arbitrarily large, the total number F of PRBs is upper-bounded by $F^{\text{MAX}} = 275 \cdot W$, if all the W spectrum bands in \mathcal{W} support the maximum number of PRBs allowed by 5G NR [57]. For instance, in a scenario with $B = 5$ cell sites, $W = 10$ bands, each with 275 PRBs, and $I = 10$ requests, there are more than 130,000 optimization variables and more than 600,000 constraints from (3) alone.

Therefore, we have designed the following complexity reduction and relaxation techniques:

- *Variable Reduction (VR)*: any variable can be eliminated that: (i) always results in unfeasible solutions, or (ii) is always equal to zero due to the structure of the problem. For example, for any request i , we have that allocating i to any cell site $b \in \mathcal{R}_{-i}$ is unfeasible. Similarly, if cell site $b \in \mathcal{R}_i$ but a specific band $\omega \in \mathcal{W}$ is not supported by the cell site, then any $x_{i,b,f}$ with $f \in \mathcal{F}_\omega$ will always be equal to zero due to (2). For this reason, all of these variables can be removed from the search space, thus reducing the time necessary to compute an optimal solution.
- *PRB grouping (PG)*: PRBs can be bundled together into groups of minimum size K such that the problem is cast into a space with $\tilde{F} = F/K$ PRBs. A preliminary grouping is naturally occurring in any 5G NR systems as PRBs are grouped into Resource Block Groups (RBGs) with varying numbers of PRBs, depending on the specific numerology. We can further extend this concept in at least two ways. The first one assumes that as tenants submit their requests to the infrastructure owner that groups PRBs by computing the greatest common divisor (GCD) among all demands δ_i such that $K = \text{GCD}(\mathcal{I})$. Another approach instead would force tenants to submit requests with a demand whose value is a multiple of a fixed block size K , i.e., $\delta_i = n_i K$, with n_i an integer for all $i \in \mathcal{I}$. While the first approach leaves more room for customizing requests to the tenants (which can potentially generate requests for a single PRB), it is prone to achieve poor complexity reduction as the GCD might be small. On the contrary, the second approach is less flexible for the tenants but results in lower computational complexity as the infrastructure owner can select a large K .

4.4.1 On the optimality gap

The impact of VR and PG on the optimality of the solution is worth discussing. First, VR eliminates only *inactive variables*, i.e., those variables of the problem that would be equal to 0 due to structural properties. For example, VR would eliminate those variables that would force allocating i to any cell site $b \in \mathcal{R}_{-i}$. In this case, it is trivial to prove that VR has no impact on the optimality of the solution as the removed variables do not contribute to the objective function, and do not affect the solution, which remains optimal and with an optimality gap of zero. PG, instead, affects the way requests are being generated and processed by NeutRAN. When using PG NeutRAN forces tenants to either request a number of PRBs that is a multiple of a fixed block size K , or groups resources using their GCD K in an effort to convert requested resources into blocks of requested resources. In both cases, the solution computed by NeutRAN is still optimal with respect to the requests being submitted, as the allocation always satisfies the requests

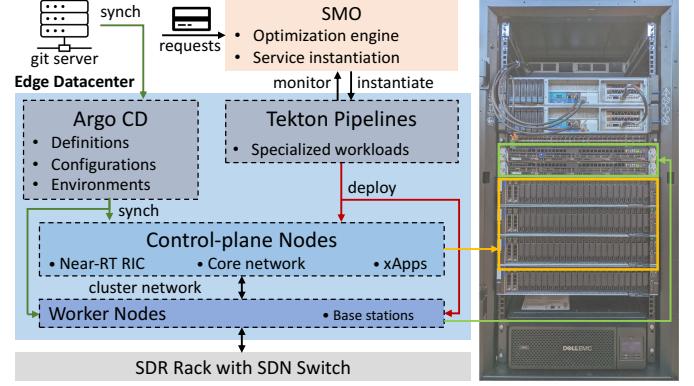


Fig. 3: The OpenShift-based NeutRAN prototype.

and only changes the way they are being allocated (i.e., individually or as a group).

5 THE NEUTRAN PROTOTYPE

The NeutRAN prototype (Figure 3) implements the framework described in Sections 3 and 4. It features unique software and hardware components that enable automated pipelines for end-to-end, optimization-driven spectrum and RAN sharing.

Software—automation. The overall software infrastructure involves more than 330 components executing as either micro-services deployed as OpenShift pods (i.e., containerized applications) or rApps in the SMO. To develop our NeutRAN prototype and enable the seamless transition from tenant requests (specified via a graphical control interface) to deployment of fully operational cellular networks, we have implemented a set of custom automation pipelines on top of the edge datacenter OpenShift infrastructure using open-source, cloud-native continuous integration and delivery frameworks. These make it possible to apply the output of the optimization engine rApp (in the SMO) to *generic application templates* (see Listing 1 for an example) and translate them into *custom services* that are then automatically deployed on the cluster.

Notably, the SMO uses an O2-like interface to control the OpenShift APIs and the various services deployed on the edge datacenter. The optimization engine rApp computes the optimal allocation of RAN services by solving the neutral host problem described in Section 4, and automatically instantiates the resulting services (e.g., the RAN applications) by adapting a set of generic templates to the specific requested services at run time. Templates are deployed on the OpenShift cluster through the Argo Continuous Delivery (CD) framework, which supports declarative application definitions, configurations, and environments synchronized from a version-controlled source (e.g., a git server) [58]. Starting from these templates, the actual workloads and services resulting from the NeutRAN optimization are instantiated as pods from an internal Docker image registry through Tekton pipelines [59].

After their instantiation, applications and services are actively monitored by NeutRAN, which can tune their configuration at run time based on subsequent optimization results, and re-instantiate them if necessary (e.g., in case of conflicts between services, or failure of a certain service). In

```

1 apiVersion: template.openshift.io/v1
2 kind: Template
3 metadata: # template name (e.g., neutran) and annotations
4 parameters: # template parameters (e.g., frequencies;
   core network, RIC, and USRP IP; slice allocations)
5 objects:
6   - kind: Deployment
7     apiVersion: apps/v1
8     metadata: # deployment name (e.g., neutran-cell-1),
   namespace (e.g., neutran), and labels
9     spec:
10       template:
11         metadata: # template labels, and annotations
12         spec:
13           nodeSelector: # to select low-latency nodes
14           containers:
15             - name: # pod name (e.g., neutran-cell-1)
16               image: # Docker image (e.g., neutran-cell)
17               command: # pod-entrypoint (e.g., /run.sh)
18               env: # parameteres of line 4 passed as
19                 environment variables
20                 ports: # exposed ports and protocols
21                 resources: # pod compute resources
22   - kind: Service # exposed services (e.g., Flask APIs)
22   - kind: Route # routes to reach the exposed services

```

Listing 1: Base station deployment template to be specialized at run time.

this way, NeutRAN is resilient to failures, and self-adapts to heterogeneous network deployments and operator requests.

Software—edge services. We deployed the functionalities that support OpenShift on dedicated control-plane nodes (e.g., cluster monitoring services, operators, certificate managers, DNS, etc.), which also host additional edge micro-services. These include (i) an E-release O-RAN near-RT RIC provided by the O-RAN Software Community (OSC), with an E2 termination to the RAN for data collection and performance reporting; (ii) data-driven xApps running on the near-RT RIC; and (iii) a core network implemented through Open5GS [60]. We configured additional compute resources (worker nodes) to only execute low-latency applications, i.e., the base stations, thus providing performance guarantees. The RAN is implemented through the SCOPE software-defined cellular stack, part of the publicly available OpenRAN Gym framework [61]. SCOPE extends srsRAN with network slicing capabilities (leveraged to implement spectrum sharing among different tenants), and the O-RAN-compliant E2 termination to communicate with the near-RT RIC [62, 63]. Every application is containerized and exposes Flask REST APIs for monitoring and re-configuration.

Hardware. We deployed the three main infrastructure components of NeutRAN as follows: (i) the SMO, on an Intel NUC (15 CPU cores, 64 GB RAM); (ii) the edge datacenter, on a bare-metal cluster managed by OpenShift; and (iii) four cell sites, on USRPs X310 part of an infrastructure with SDRs, antenna locations, and computational facilities. The cluster features three control-plane nodes (Dell PowerEdge R740, 32 CPU cores and 192 GB RAM) and two worker nodes (Microway EPYC, 32 CPU cores and 256 GB RAM). To ensure low latency and high performance, each node embeds a 100 Gbps Ethernet card from NVIDIA Mellanox. Workers connect to the SDRs via a Dell 4048T-ON Software-defined Networking (SDN) switch.

An extensive experimental evaluation of NeutRAN and its procedure is presented in Section 6.2, where we demonstrate that NeutRAN can effectively deploy a fully virtualized network in less than 10 seconds while improving a variety of KPMs. We notice that NeutRAN can deploy an

end-to-end cellular network on white-box infrastructure in less than 10 s, demonstrating its feasibility and effectiveness.

6 NEUTRAN PERFORMANCE EVALUATION

This section presents the NeutRAN performance evaluation, including numerical results for scalability and resource utilization, and experimental results about the gain that sharing introduces for multiple KPMs.

6.1 Scalability, Complexity, and Effectiveness

We run large-scale simulations to (i) evaluate the computational complexity and scalability of NeutRAN, and (ii) assess and characterize its performance to determine its applicability to real-world large-scale deployments.

The following results are generated through a custom MATLAB simulator that uses Gurobi to solve problem (8) on a compute node with 32 GB of RAM and a 12-core Intel Core i7-9750H CPU at 2.60 GHz. We consider a deployment with 5G NR cell sites uniformly deployed on a grid with $A = 21 \times 11 = 231$ areas. The RAN uses numerology 4, i.e., 240 kHz subcarrier spacing and 138 PRBs per band [57]. The coverage indicators $c_{b,j}$ are configured such that $c_{b,j} = 1$ if area $j \in \mathcal{A}$ is distant at most $3\sqrt{2}/2$ from the cell site, where the distance is normalized with respect to the width of the area. We consider the case where $w_i = 1$ (i.e., problem (8) aims at maximizing the number of admitted requests). All results are averaged over 100 independent simulation runs.

Complexity Analysis. Here, we leverage and profile the reduction techniques VR and PG presented in Section 4.4. They identify the optimal solution for problem (8) with a reduced complexity, avoiding computations of 60 s or more even in scenarios with few cells or bands.

Figure 4 offers an analysis of the computational complexity of problem (8) aimed at showing the scalability of NeutRAN. We consider scenarios with variable numbers W of spectrum bands per cell site and different values of K . The figure on the left shows that NeutRAN can compute optimal solutions in few hundreds of milliseconds even in the case of large W and grouping coefficients $K \geq 23$ (which corresponds to dividing the available 138 PRBs in 6 groups of size 23). We notice that grouping with higher values of K is indeed effective, as it enables a 97% reduction

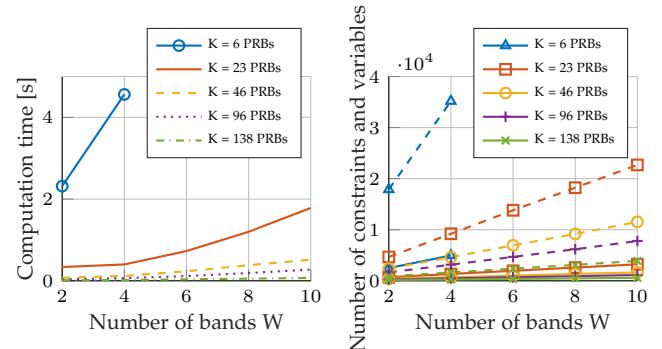


Fig. 4: Computational complexity analysis for different number of spectrum bands (W) and grouping coefficients (K) by using both VR and PG. Left: computation time in seconds; Right: number of variables (solid lines) and constraints (dashed lines). $I = 20$, $B = 50$.

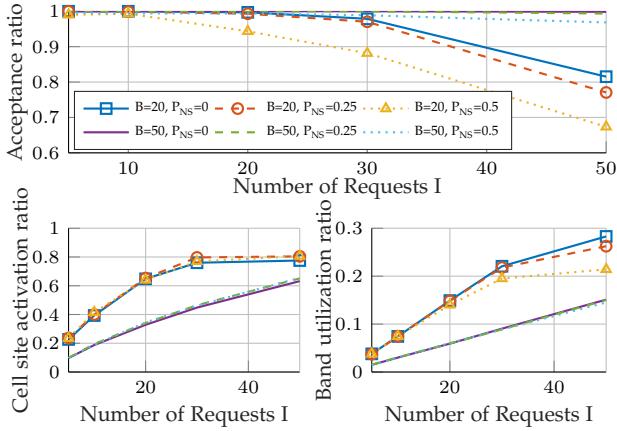


Fig. 5: Acceptance, cell site activation and band utilization ratios as a function of the number of requests (I) for different values of P_{NS} and number of cell sites (B). The figures share the same legend. $W = 5$.

in the computation times of optimal solutions (i.e., from 4.56 s of $K = 6$ to 0.12 s of $K = 23$ for $W = 4$). This is due to the reduction in the number of variables (solid lines) and constraints (dashed lines) required to solve problem (8), which are shown on the right side of Figure 4.

Resource Utilization and Allocation Effectiveness. Another important aspect to investigate is how many requests are admitted by NeutRAN, as well as how many cell sites and spectrum bands are activated for different configurations and deployments. To capture real-world deployment characteristics, the probability P_{NS} models those cases where a cell site b can support a number of bands $W_b \leq W$. In each run and for each $b \in \mathcal{R}$ and band $\omega \in \Omega$, we generate a random variable z from a uniform distribution in $[0, 1]$ and set $\beta_{b,f} = 0$ for each $f \in \omega$ if $z < P_{NS}$. Additionally, the value of the single band indicator variable ξ_b is modeled with a probability P_{SB} : for each $b \in \mathcal{R}$, we randomly draw the variable z again and set $\xi_b = 1$ if $z < P_{SB}$. In the following, we consider $W = 5$.

Figure 5 depicts the acceptance, cell site activation, and band utilization ratios as a function of the number of requests for different values of P_{NS} and number B of cell sites controlled by NeutRAN. The figure on the top shows that the acceptance ratio decreases when a larger number of requests is submitted to NeutRAN. Intuitively, the larger the number of requests, the higher the probability that not all requests can be accommodated due to the limited number of PRBs and spectrum bands. Moreover, when $B = 20$, the acceptance ratio decreases from 82% for $P_{NS} = 0$ (i.e., all cell sites support all $W = 5$ bands) to 67% for $P_{NS} = 0.5$ (i.e., the probability that a cell site does not support a band is 50%). The same applies when $B = 50$, with the acceptance ratio dropping from 99.8% for $P_{NS} = 0$ to 96% for $P_{NS} = 0.5$. As expected, more cell sites also imply a higher acceptance rate as more resources are available to NeutRAN.

Figure 5 (bottom-left) shows that the cell site activation ratio always increases with the number of requests submitted to NeutRAN. The activation ratio is higher for $B = 20$ than $B = 50$, as NeutRAN's optimal policy requires the allocation of as many cell sites as possible in the available pool. Instead, Figure 5 (bottom-right) shows how different configurations affect spectrum utilization. Intuitively, the higher the number of submitted requests, the higher the

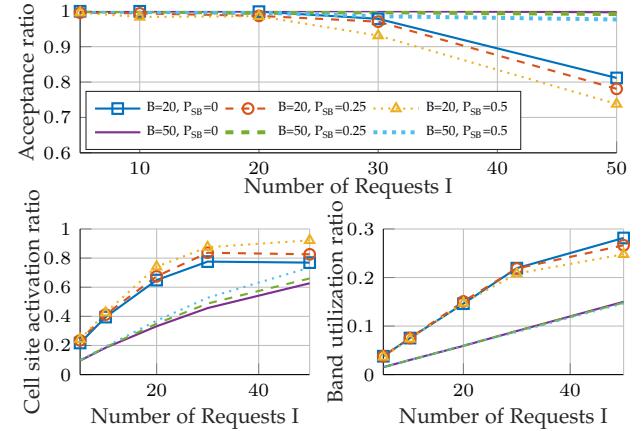


Fig. 6: Acceptance, cell site activation and band utilization ratios as a function of the number of requests (I) for different values of P_{SB} and number of cell sites (B). The figures share the same legend. $W = 5$.

number of spectrum bands to be allocated to support such requests and avoid interference among neighboring cell sites. Indeed, as the cell site activation ratio increases, more bands must be activated to eliminate interference while allocating the necessary PRBs. This happens in particular for the resource constrained case of $B = 20$.

Figure 6 concerns the same metrics for varying values of the probability P_{SB} . Trends for acceptance and band utilization rates are similar to those in Figure 5. However, we notice that varying P_{SB} affects the activation of cell sites to a higher extent (Figure 6, bottom-left). Specifically, we see that when half the cell sites can operate on a single spectrum band at any given time ($P_{SB} = 0.5$), the activation ratio increases by approximately 20% compared to the case when all cell sites support multiple spectrum bands ($P_{SB} = 0$). This illustrates the importance of deploying cell sites with RUs that can transmit simultaneously on multiple bands and that can accommodate more requests with a lower number of active cell sites (lowering operational and energy costs).

6.2 Experimental Evaluation of RAN and Spectrum Sharing

To evaluate the performance of NeutRAN in a real-world scenario, we deployed the prototype in an indoor space with multiple obstacles, heterogeneous equipment and moving humans, creating a wireless environment rich with scattering. The indoor area (Figure 7) covers more than 1100 m^2 . It has been logically divided into $A = 91$ tiles of size

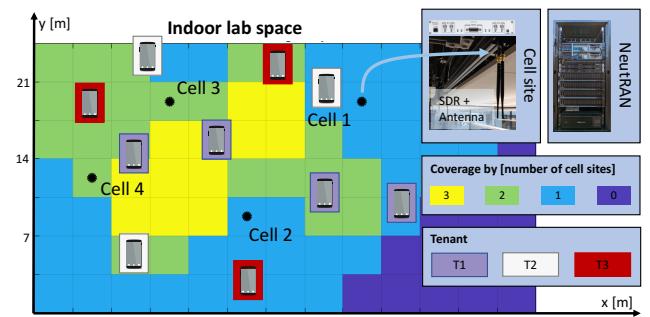


Fig. 7: Experimental setup: NeutRAN on an over-the-air SDR testbed.

sites, each with an USRP X310 frontend, whose antennas are mounted on the ceiling of the testbed. We deployed 10 commercial smartphones as shown in Figure 7. Smartphones represent end users for a set \mathcal{T} with $T = 3$ different tenants (e.g., different mobile network or private network operators). Specifically, 4 users are served by tenant 1, and 3 each by tenants 2 and 3. Connectivity is provided over LTE Band 7 under an experimental license, and $W = 3$ blocks with 10 MHz of spectrum (50 PRBs) each. Blocks are identified by a pair (f_d, f_u) of downlink and uplink carrier frequencies, with $f_d \in \{2.625, 2.645, 2.685\}$ GHz and $f_u \in \{2.505, 2.525, 2.565\}$ GHz. We conservatively define the coverage area of a cell site as the set of tiles where users experience an average throughput higher than 50% of the throughput that they would experience in the tile with the cell site antenna. The result is the coverage map in Figure 7.

To exhaustively evaluate the performance of NeutRAN, we consider the three following scenarios, each involving four independent experiments lasting more than 300 s each.

- *No sharing*: this license-based scenario corresponds to a traditional cellular network deployment where each tenant (i) uses different cell sites without infrastructure sharing, and (ii) owns a dedicated, licensed portion of the spectrum. Each tenant operates independently on one of the three 10 MHz blocks. Since the RAN is not shared, tenant 1 serves users from cell sites 1 and 3, tenant 2 from cell site 2, and tenant 3 from cell site 4.
- *Sharing, 30 MHz*: the NeutRAN optimization engine and automation pipelines are used to allocate tenant requests over different cell sites and spectrum. NeutRAN controls a total of 30 MHz, i.e., the same amount of spectrum available to the *No sharing* configuration. Each cell site, however, only provides service through a single pair of downlink/uplink carrier frequencies and a 10 MHz spectrum block.
- *Sharing, 10 MHz*: this configuration is the same as *Sharing, 30 MHz*, with the difference that the system is deployed on an overall spectrum of 10 MHz, shared among tenants. Since only one band is available in this scenario ($W = 1$), NeutRAN optimization engine does not enforce the interference constraint (3).

The KPMs are collected at the RAN side every 250 ms for each UE. Here, we consider downlink throughput, downlink Modulation and Coding Scheme (MCS), and uplink SINR. We also aggregate the UE downlink throughput over time to compute the average and median UE throughput and the total RAN throughput for each experiment. During

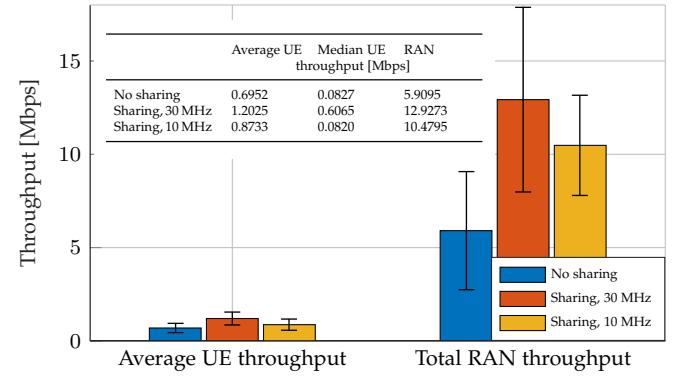


Fig. 8: Average, sum, and median throughput metrics. With *No sharing*, three tenants deploy their RAN on independent spectrum (10 MHz each, for a total of 30 MHz) and cell sites and equipment. With *Sharing, 30 MHz*, the tenants share the RAN infrastructure, cell sites, and 30 MHz of spectrum with the NeutRAN architecture. Finally, *Sharing, 10 MHz* features the same sharing configuration, but only uses 10 MHz of spectrum. We show the 95% confidence intervals.

the experiments, the smartphones stream videos through the YouTube application.

6.2.1 Experimental Results

We investigate whether using the NeutRAN-enabled RAN and spectrum sharing improves network performance. Figure 8 concerns the average throughput.

Both *Sharing* configurations outperform the first one in the total RAN throughput, showing that a network with an optimized approach to resource management and sharing outperforms orthogonal license-based schemes. *Sharing, 30 MHz* achieves a $2.18\times$ gain over *No Sharing*, and the setup with 10 MHz spectrum has a $1.77\times$ gain, as also shown in the table part of Figure 8. In addition, the *Sharing* approach with 30 MHz also outperforms the *No Sharing* scheme in terms of average and median throughput, suggesting that the improvement does not apply only to aggregated performance, but also, on average, to the single UEs. A similar behavior can be observed in Figures 9a-9b, which show the Cumulative Distribution Function (CDF) of the UE throughput and downlink MCS, respectively. Both *Sharing* scenarios outperform the *No Sharing* one.

Considering the *Sharing* setup with 10 MHz, we notice that sharing the available spectrum improves the total throughput when compared to the case *No sharing*, with comparable average and median UE throughput. We would like to point out that, thanks to the capabilities enabled

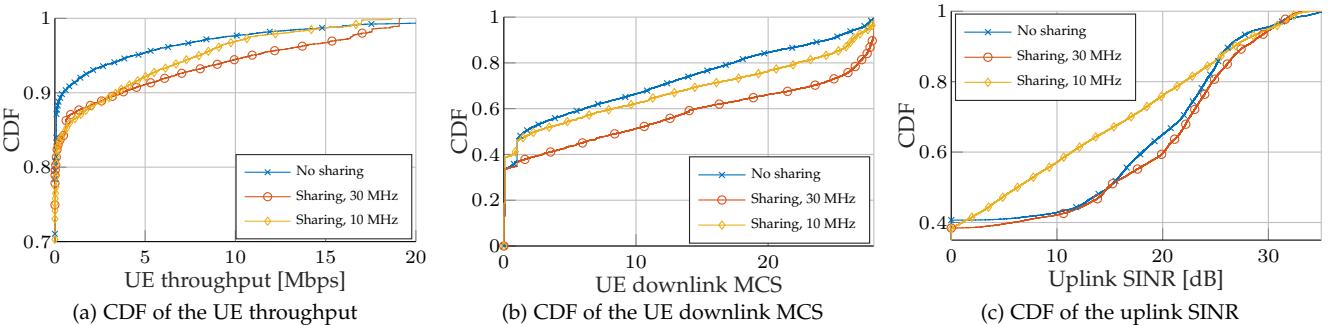


Fig. 9: CDF of UE throughput, downlink MCS, and uplink SINR. With *No sharing*, three tenants deploy their RAN on independent spectrum (10 MHz each, for a total of 30 MHz) and cell sites and equipment. With *Sharing, 30 MHz*, the tenants share the RAN infrastructure, cell sites, and 30 MHz of spectrum with the NeutRAN architecture. Finally, *Sharing, 10 MHz* features the same configuration, but only uses 10 MHz of spectrum.

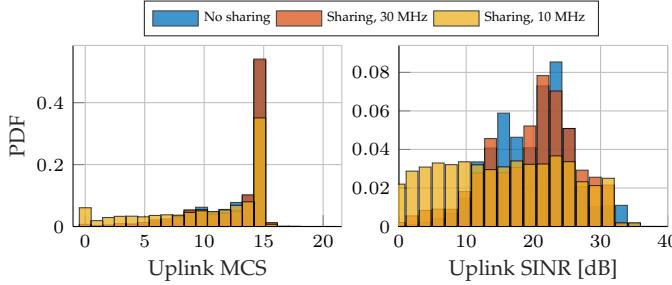


Fig. 10: PDF of UE uplink MCS and SINR in the *No sharing*, *Sharing, 30 MHz*, and *Sharing, 10 MHz* scenarios. The MCS saturates to 15 for values of SINR ≥ 22.7 .

TABLE 1: Average deployment and instantiation time of different applications on the OpenShift-managed NeutRAN infrastructure.

Application	Required Pods	Instantiation Time [s]
Base station	1	9.55
Core network	17	5.90
Near-RT RIC	13	2.93
xApp	1	2.35

by NeutRAN, such comparable behavior is achieved by using 1/3 of the spectrum used in the *No sharing* setup. This is because RAN sharing allows tenants to optimally deploy their resources, e.g., by maximizing the coverage of their users and thus improving the downlink throughput and MCS (Figures 9a-9b), especially for users with the best channel conditions. However, when using a single 10 MHz block, interference is significant for cell edge users, as shown in Figure 9c, which reports the uplink SINR experienced by each UE and shows an average 5 dB gap for 80% of the UEs between the 10 MHz *Sharing* setup and the others. The linear behavior observed in this scenario is due to the higher spectrum utilization of the three tenants sharing a narrower portion of the spectrum. This affects the Probability Density Function (PDF) of MCS and SINR, reported in Figure 10, with values close to those of the uniform distribution in the 10 MHz *Sharing* setup. This causes less fairness among the UEs in uplink.

Another aspect worth considering is related to how fast NeutRAN can deploy a fully working cellular network. Table 1 shows the average instantiation time for the pods of specific micro-services (i.e., base station, core network, near-RT RIC, and xApp pods), as well as the number of pods required to run them.

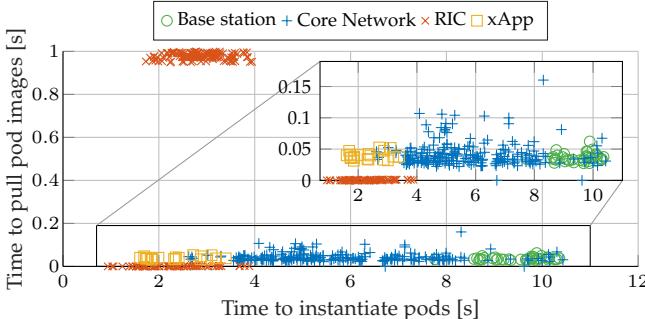


Fig. 11: Time to instantiate pods vs. time to pull pod images for the different NeutRAN components.

Similarly, in Figure 11, we show the time taken by NeutRAN to instantiate each pod, as well as the time needed to download (i.e., pull) the pod image from the image registry to the physical machine where it will be instantiated. The xApps images are the ones that take the least time to be pulled and instantiated, with an average pull time below 0.1 s. A more diverse distribution can be seen in the pods of the core network, whose instantiation times range from 2 s to 11 s. The pods of the base stations have an average pull time of 0.1 s but they take the longest to be instantiated (as also shown in Table 1). Finally, since the pods of the near-RT RIC are the largest ones—and are hosted on the OSC Nexus image registry instead of locally on NeutRAN cluster—they require more time to be pulled (i.e., up to 1 s), but are instantiated within 4 s. These results show that, on average, NeutRAN can instantiate a fully functional cellular network in around 10 s.

7 CONCLUSIONS

We presented NeutRAN, a zero-touch framework to enable and automate RAN and spectrum sharing among multiple tenants. NeutRAN allows tenants to deploy on the fly an end-to-end RAN tailored to their needs. We introduced the NeutRAN framework, and described its components and functionalities, including its optimization engine rApp and virtualized infrastructure. Optimal support of neutral host applications is obtained by modeling it as a complex optimization problem, that we show solvable well within seconds for networks of realistic scale (e.g., 50 cell sites). We prototyped NeutRAN on an OpenShift cluster and an indoor RAN testbed with 4 base stations and 10 users from 3 different tenants, showing that NeutRAN can boost the cumulative throughput of tenants' subscribers by up to 2.18 \times .

REFERENCES

- [1] N. Bhushan, J. Li, D. Malladi, R. Gilmore, D. Brenner, A. Damjanovic, R. T. Sukhavasi, C. Patel, and S. Geirhofer, "Network densification: The dominant theme for wireless evolution into 5G," *IEEE Communications Magazine*, vol. 52, no. 2, pp. 82–89, February 2014.
- [2] M. Wen, Q. Li, K. J. Kim, D. López-Pérez, O. A. Dobre, H. V. Poor, P. Popovski, and T. A. Tsiftsis, "Private 5G networks: Concepts, architectures, and research landscape," *IEEE Journal of Selected Topics in Signal Processing*, vol. 16, no. 1, pp. 7–25, January 2022.
- [3] Federal Communications Commission, "WC Docket No. 18-89," DA 21-355, 2021. [Online]. Available: <https://tinyurl.com/2maycpp3>
- [4] E. J. Oughton and Z. Frias, "The cost, coverage and rollout implications of 5G infrastructure in Britain," *Telecommunications Policy*, vol. 42, no. 8, pp. 636–652, 2018.
- [5] 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, July 2016.
- [6] J. Lahteenmaki, "The Evolution Paths of Neutral Host Businesses: Antecedents, Strategies, and Business Models," *Telecommunications Policy*, vol. 45, no. 10, pp. 1–27, November 2021.
- [7] Alpha Wireless Insights Blog, "Analysis: The Neutral Host Model is Sprouting Wings," 2021. [Online]. Available: <https://alphawireless.com/analysis-the-neutral-host-model-is-sprouting-wings/>
- [8] Market Research, "Neutral Hosting Market Outlook and Forecasts 2021 – 2028," 2021. [Online]. Available: <https://www.marketresearch.com/Mind-Commerce-Publishing-v3122/Neutral-Hosting-Outlook-Forecasts-14883308/>
- [9] L. Zhang, M. Xiao, G. Wu, M. Alam, Y.-C. Liang, and S. Li, "A survey of advanced techniques for spectrum sharing in 5G

networks," *IEEE Wireless Communications*, vol. 24, no. 5, pp. 44–51, October 2017.

[10] C. Gabriel, "The impact of 5G and next-generation networks on mobile OPEX spending," Analysis Mason report, 2018. [Online]. Available: <https://tinyurl.com/ycckndswz>

[11] STL Partners, "Neutral Host: How Open RAN and Neutral Host paves the way for 5G," 2022. [Online]. Available: <https://stlpartners.com/articles/telco-cloud/neutral-host-how-open-ran-and-neutral-host-paves-way-5g/>

[12] M. Palola, M. Höyhtyä, P. Aho, M. Mustonen, T. Kippola, M. Heikkilä, S. Yrjölä, V. Hartikainen, L. Tudose, A. Kivinen, R. Ekman, J. Hallio, J. Paavola, M. Mäkeläinen, and T. Hänninen, "Field trial of the 3.5 GHz citizens broadband radio service governed by a spectrum access system (SAS)," in *Proceedings of IEEE DySPAN*, Baltimore, MD, USA, March 2017.

[13] 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 ACM MobiCom*, New Orleans, LA, USA, October 2021.

[14] M. Liyanage, Q.-V. Pham, K. Dev, S. Bhattacharya, P. K. R. Madikunta, T. R. Gadekallu, and G. Yenduri, "A survey on zero touch network and service management (ZSM) for 5G and beyond networks," *Journal of Network and Computer Applications*, vol. 203, pp. 1–27, July 2022.

[15] C. Benzaid and T. Taleb, "AI-driven zero touch network and service management in 5G and beyond: Challenges and research directions," *IEEE Network*, vol. 34, no. 2, pp. 186–194, March 2020.

[16] S. D'Oro, L. Bonati, M. Polese, and T. Melodia, "OrchestRAN: Network automation through orchestrated intelligence in the open RAN," in *Proceedings of IEEE INFOCOM*, London, United Kingdom, May 2022.

[17] S. K. Sharma, T. E. Bogale, L. B. Le, S. Chatzinotas, X. Wang, and B. Ottersten, "Dynamic spectrum sharing in 5G wireless networks with full-duplex technology: Recent advances and research challenges," *IEEE Communications Surveys & Tutorials*, vol. 20, no. 1, pp. 674–707, February 2017.

[18] GSMA, "Spectrum Sharing," GSMA Public Policy Position, June 2021. [Online]. Available: <https://tinyurl.com/5bju2k44>

[19] L. Bonati, S. D'Oro, M. Polese, S. Basagni, and T. Melodia, "Intelligence and Learning in O-RAN for Data-driven NextG Cellular Networks," *IEEE Communications Magazine*, vol. 59, no. 10, pp. 21–27, October 2021.

[20] M. G. Kibria, G. P. Villardi, K. Nguyen, W.-S. Liao, K. Ishizu, and F. Kojima, "Shared spectrum access communications: A neutral host micro operator approach," *IEEE Journal on Selected Areas in Communications*, vol. 35, no. 8, pp. 1741–1753, August 2017.

[21] E. Di Pascale, H. Ahmadi, L. Doyle, and I. Macaluso, "Toward scalable user-deployed ultra-dense networks: Blockchain-enabled small cells as a service," *IEEE Communications Magazine*, vol. 58, no. 8, pp. 82–88, August 2020.

[22] M. Vincenzi, A. Antonopoulos, E. Kartsakli, J. Vardakas, L. Alonso, and C. Verikoukou, "Cooperation incentives for multi-operator C-RAN energy efficient sharing," in *Proceedings of IEEE ICC*, Paris, France, May 2017.

[23] L. Sarakis, P. Trakadas, J. Martrat, S. Prior, O. Trullols-Cruces, E. Coronado, M. Centenaro, G. Kontopoulos, E. Axtutegi, P. Gkounis *et al.*, "Cost-efficient 5G non-public network roll-out: The Affordable5G approach," in *Proceedings of IEEE MeditCom*, Athens, Greece, September 2021.

[24] M. Paolino, G. Carrozzo, A. Betzler, C. Colman-Meixner, H. Khalili, S. Siddiqui, T. Sechkova, and D. Simeonidou, "Compute and network virtualization at the edge for 5G smart cities neutral host infrastructures," in *Proceedings of IEEE 5GWF*, Dresden, Germany, September 2019.

[25] Y. Xiao, M. Krunk, and T. Shu, "Multi-operator network sharing for massive IoT," *IEEE Communications Magazine*, vol. 57, no. 4, pp. 96–101, April 2019.

[26] B. Qian, H. Zhou, T. Ma, K. Yu, Q. Yu, and X. Shen, "Multi-operator spectrum sharing for massive IoT coexisting in 5G/B5G wireless networks," *IEEE Journal on Selected Areas in Communications*, vol. 39, no. 3, pp. 881–895, August 2020.

[27] Y.-D. Lin, H.-T. Chien, H.-W. Chang, C.-L. Lai, and K.-Y. Lin, "Transparent RAN sharing of 5G small cells and macrocells," *IEEE Wireless Communications*, vol. 24, no. 6, pp. 104–111, December 2017.

[28] F. Giannone, K. Kondepu, H. Gupta, F. Civerchia, P. Castoldi, A. A. Franklin, and L. Valcarenghi, "Impact of virtualization technologies on virtualized RAN midhaul latency budget: A quantitative experimental evaluation," *IEEE Communications Letters*, vol. 23, no. 4, pp. 604–607, February 2019.

[29] A. T. Z. Kasgari and W. Saad, "Stochastic Optimization and Control Framework for 5G Network Slicing with Effective Isolation," in *Proceedings of IEEE CISS*, 2018, pp. 1–6.

[30] G. Wang, G. Feng, T. Q. S. Quek, S. Qin, R. Wen, and W. Tan, "Reconfiguration in Network Slicing—Optimizing the Profit and Performance," *IEEE Transactions on Network and Service Management*, vol. 16, no. 2, pp. 591–605, 2019.

[31] D. Bega, M. Gramaglia, M. Fiore, A. Banchs, and X. Costa-Perez, "DeepCog: Cognitive Network Management in Sliced 5G Networks with Deep Learning," in *Proceedings of IEEE INFOCOM 2019*, Paris, France, April 29–May 2 2019, pp. 280–288.

[32] X. Foukas, M. K. Marina, and K. Kontovasilis, "Iris: Deep reinforcement learning driven shared spectrum access architecture for indoor neutral-host small cells," *IEEE Journal on Selected Areas in Communications*, vol. 37, no. 8, pp. 1820–1837, August 2019.

[33] P. Caballero, A. Banchs, G. De Veciana, and X. Costa-Pérez, "Network Slicing Games: Enabling Customization in Multi-Tenant Mobile Networks," *IEEE/ACM Transactions on Networking*, vol. 27, no. 2, pp. 662–675, April 2019.

[34] P. Caballero, A. Banchs, G. de Veciana, and X. Costa-Pérez, "Multi-Tenant Radio Access Network Slicing: Statistical Multiplexing of Spatial Loads," *IEEE/ACM Transactions on Networking*, vol. 25, no. 5, pp. 3044–3058, October 2017.

[35] S. D'Oro, L. Bonati, F. Restuccia, and T. Melodia, "Coordinated 5G network slicing: How constructive interference can boost network throughput," *IEEE/ACM Transactions on Networking*, vol. 29, no. 4, pp. 1881–1894, 2021.

[36] S. D'Oro, L. Bonati, F. Restuccia, M. Polese, M. Zorzi, and T. Melodia, "Sl-EDGE: Network slicing at the edge," in *Proceedings of ACM MobiHoc*, Virtual Event, October 2020.

[37] C. Puligheddu, J. Ashdown, C. F. Chiasserini, and F. Restuccia, "SEM-O-RAN: Semantic and Flexible O-RAN Slicing for NextG Edge-Assisted Mobile Systems," in *Proceedings of IEEE INFOCOM 2023 (preprint available as arXiv:2212.11853 [cs.NI])*, New York Area, NJ, USA, May 2023.

[38] L. Baldesi, F. Restuccia, and T. Melodia, "ChARM: NextG spectrum sharing through data-driven real-time O-RAN dynamic control," in *Proceedings of IEEE INFOCOM*, London, United Kingdom, May 2022.

[39] D. Bega, M. Gramaglia, A. Banchs, V. Sciancalepore, K. Samdanis, and X. Costa-Perez, "Optimising 5G Infrastructure Markets: The Business of Network Slicing," in *Proceedings of IEEE INFOCOM*, Atlanta, GA, USA, May 2017, pp. 1–9.

[40] V. Sciancalepore, K. Samdanis, X. Costa-Perez, D. Bega, M. Gramaglia, and A. Banchs, "Mobile Traffic Forecasting for Maximizing 5G Network Slicing Resource Utilization," in *Proceedings of IEEE INFOCOM*, Atlanta, GA, USA, May 2017, pp. 1–9.

[41] P. Caballero, A. Banchs, G. de Veciana, X. Costa-Pérez, and A. Azcorra, "Network slicing for guaranteed rate services: Admission control and resource allocation games," *IEEE Transactions on Wireless Communications*, vol. 17, no. 10, pp. 6419–6432, October 2018.

[42] R. Schmidt, C.-Y. Chang, and N. Nikaein, "FlexVRAN: A flexible controller for virtualized RAN over heterogeneous deployments," in *Proceedings of IEEE ICC*, Shanghai, China, May 2019.

[43] S. K. Moorthy, Z. Guan, N. Mastronarde, E. S. Bentley, and M. Medley, "OSWireless: Enhancing automation for optimizing intent-driven software-defined wireless networks," in *Proceedings of IEEE MASS*, Denver, CO, USA, October 2022.

[44] 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 ACM MobiCom*, Snowbird, UT, USA, October 2017.

[45] X. Foukas, F. Sardis, F. Foster, M. K. Marina, M. A. Lema, and M. Dohler, "Experience building a prototype 5g testbed," in *Proceedings of the Workshop on Experimentation and Measurements in 5G*, ser. EM-5G'18. New York, NY, USA: Association for Computing Machinery, 2018, p. 13–18. [Online]. Available: <https://doi.org/10.1145/3286680.3286683>

[46] O-RAN Working Group 1, "O-RAN architecture description 5.00," O-RAN.WG1.O-RAN-Architecture-Description-v05.00 Technical Specification, July 2021.

[47] S. D'Oro, M. Polese, L. Bonati, H. Cheng, and T. Melodia, "dApps: Distributed Applications for Real-Time Inference and Control in O-RAN," *IEEE Communications Magazine*, vol. 60, no. 11, pp. 52–

58, November 2022.

[48] M. Polese, L. Bonati, S. D’Oro, S. Basagni, and T. Melodia, “Understanding O-RAN: Architecture, Interfaces, Algorithms, Security, and Research Challenges,” *IEEE Communications Surveys & Tutorials*, vol. 25, no. 2, pp. 1376–1411, January 2023.

[49] Red Hat. OpenShift. <https://tinyurl.com/382mww6d>.

[50] Linux Foundation. Kubernetes. <https://kubernetes.io>.

[51] K. Huang and N. D. Sidiropoulos, “Consensus-ADMM for general quadratically constrained quadratic programming,” *IEEE Transactions on Signal Processing*, vol. 64, no. 20, pp. 5297–5310, July 2016.

[52] P. Androulakis and P. M. Pardalos, “Quadratic integer programming: Complexity and equivalent forms,” *Encyclopedia of Optimization*. Berlin, Germany: Springer, 2008.

[53] S. Boyd and L. Vandenberghe, “Semidefinite programming relaxations of non-convex problems in control and combinatorial optimization,” in *Communications, Computation, Control, and Signal Processing*. Springer, 1997, pp. 279–287.

[54] H. D. Sherali and W. P. Adams, *A Reformulation-linearization Technique for Solving Discrete and Continuous Nonconvex Problems*. Springer Science & Business Media, 2013, vol. 31.

[55] K. M. Anstreicher, “Semidefinite programming versus the reformulation-linearization technique for nonconvex quadratically constrained quadratic programming,” *Journal of Global Optimization*, vol. 43, no. 2, pp. 471–484, March 2009.

[56] K. Basu, A. Saha, and S. Chatterjee, “Large-scale quadratically constrained quadratic program via low-discrepancy sequences,” *Advances in Neural Information Processing Systems*, vol. 30, 2017.

[57] 3GPP, “NR; physical channels and modulation,” 3rd Generation Partnership Project (3GPP), Technical Specification (TS) 38.211, 06 2022, version 17.2.0. [Online]. Available: <http://www.3gpp.org/DynaReport/38211.htm>

[58] Argo Project. Argo CD. <https://argoproj.github.io/cd>.

[59] The Linux Foundation. Tekton. <https://tekton.dev>.

[60] Open5GS. <https://open5gs.org>.

[61] L. Bonati, M. Polese, S. D’Oro, S. Basagni, and T. Melodia, “Open-RAN Gym: AI/ML Development, Data Collection, and Testing for O-RAN on PAWR Platforms,” *Computer Networks*, pp. 1–12, November 2022.

[62] L. Bonati, S. D’Oro, S. Basagni, and T. Melodia, “SCOPE: An open and softwareized prototyping platform for NextG systems,” in *Proceedings of ACM MobiSys*, Virtual, June 2021.

[63] Software Radio Systems. srsRAN. <https://www.srsran.com>.



Leonardo Bonati is an Associate Research Scientist at the Institute for the Wireless Internet of Things, Northeastern University, Boston, MA. He received the Ph.D. degree in Computer Engineering from Northeastern University in 2022. His main research focuses on softwareized approaches for the Open Radio Access Network (RAN) of the next generation of cellular networks, on O-RAN-managed networks, and on network automation and orchestration. He served on the technical program committee of the ACM Workshop on Wireless Network Testbeds, Experimental evaluation & Characterization, and as guest editor of the special issue of Elsevier’s Computer Networks journal on Advances in Experimental Wireless Platforms and Systems.



Michele Polese is a Principal Research Scientist at the Institute for the Wireless Internet of Things, Northeastern University, Boston, since March 2020. He received his Ph.D. at the Department of Information Engineering of the University of Padova in 2020. He also was an adjunct professor and postdoctoral researcher in 2019/2020 at the University of Padova, and a part-time lecturer in Fall 2020 and 2021 at Northeastern University. His research interests are in the analysis and development of protocols and architectures for future generations of cellular networks (5G and beyond). He has contributed to O-RAN technical specifications and submitted responses to multiple FCC and NTIA notice of inquiry and requests for comments. He received several best paper awards, is serving as TPC co-chair for WNS3 2021-2022, as an Associate Technical Editor for the IEEE Communications Magazine, and has organized the Open 5G Forum in Fall 2021.



Salvatore D’Oro is a Research Assistant Professor at Northeastern University. He received his Ph.D. degree from the University of Catania in 2015. Salvatore is an area editor of Elsevier Computer Communications journal and serves on the Technical Program Committee (TPC) of multiple conferences and workshops such as IEEE INFOCOM, IEEE CCNC, IEEE ICC and IFIP Networking. He is one of the contributors to OpenRAN Gym, the first open-source research platform for AI/ML applications in the Open RAN.

Dr. D’Oro’s research interests include optimization, artificial intelligence, security, network slicing and their applications to 5G networks and beyond, with specific focus on Open RAN systems.



Stefano Basagni is with the Institute for the Wireless Internet of Things and a professor at the ECE Department at Northeastern University, in Boston, MA. He holds a Ph.D. in electrical engineering from the University of Texas at Dallas (2001) and a Ph.D. in computer science from the University of Milano, Italy (1998). Dr. Basagni’s current interests concern research and implementation aspects of mobile networks and wireless communications systems, wireless sensor networking for IoT (underwater, aerial and terrestrial), and definition and performance evaluation of network protocols. Dr. Basagni has published over twelve dozen of highly cited, refereed technical papers and book chapters. His h-index is currently 50 (July 2023). He is also co-editor of three books. Dr. Basagni served as a guest editor of multiple international ACM/IEEE, Wiley and Elsevier journals. He has been the TPC co-chair of international conferences. He is a distinguished scientist of the ACM, a senior member of the IEEE, and a member of CUR (Council for Undergraduate Education).



Tommaso Melodia is the William Lincoln Smith Chair Professor with the Department of Electrical and Computer Engineering at Northeastern University in Boston. He is also the Founding Director of the Institute for the Wireless Internet of Things and the Director of Research for the PAWR Project Office. He received his Ph.D. in Electrical and Computer Engineering from the Georgia Institute of Technology in 2007. He is a recipient of the National Science Foundation CAREER award. Prof. Melodia has served as Associate Editor of IEEE Transactions on Wireless Communications, IEEE Transactions on Mobile Computing, Elsevier Computer Networks, among others. He has served as Technical Program Committee Chair for IEEE INFOCOM 2018, General Chair for IEEE SECON 2019, ACM Nanocom 2019, and ACM WUWnet 2014. Prof. Melodia is the Director of Research for the Platforms for Advanced Wireless Research (PAWR) Project Office, a \$100M public-private partnership to establish 4 city-scale platforms for wireless research to advance the US wireless ecosystem in years to come. Prof. Melodia’s research on modeling, optimization, and experimental evaluation of Internet-of-Things and wireless networked systems has been funded by the National Science Foundation, the Air Force Research Laboratory the Office of Naval Research, DARPA, and the Army Research Laboratory. Prof. Melodia is a Fellow of the IEEE and a Senior Member of the ACM.