ALMA MATER STUDIORUM – UNIVERSITY OF BOLOGNA

SCHOOL OF ENGINEERING

Department of
Computer Science - Science and Engineering
DISI

Master's Degree in Computer Engineering

Master's Thesis
in
Mobile Systems

Dynamic Resource Allocation and Energy Optimization in 5G Open Radio Access Network (O-RAN)

Candidate:

Caterina Leonelli

Advisor:

Prof. Paolo Bellavista

Co-Advisors:

Prof. Serge Fdida
Prof. Naceur Malouch
PhD Candidate Dimitrios Kefalas

Academic Year
2023–2024

Contents

Abstract

The energy efficiency of 5G Radio Access Networks (RANs) has become a critical area of research due to the growing energy demands of the 5G disaggregated architectures and the increasing environmental concerns surrounding mobile networks. In this paper, we address the fundamental challenge of optimizing energy consumption in Open RAN (O-RAN) 5G networks by dynamically scaling Central Unit (CU) components based on traffic demands. Leveraging a real-world testbed environment, we empirically analyze the relationship between data volume, architectural configurations, and energy usage. Our study identifies key tuning parameters for energy management and optimization, showcasing the impact of dynamic resource allocation on reducing energy consumption. Experimental results demonstrate that our implementation of a dynamic CU allocation policy can achieve energy savings of up to 60% compared to static configurations, without compromising quality of service (QoS).

Introduction

Recent studies have shown that the sector of Digital Communication Technologies (DCT) is responsible for over 2% of the global greenhouse gas emissions [1]. A significant portion of these emissions comes from cellular networks, driven by the growing demand for connectivity and high traffic loads of mobile devices. As seen in [2], within the mobile network, the Radio Access Network (RAN) is responsible for 73% of the total energy consumption. This highlights the importance of ongoing research and development of 5G RAN architectures and technologies specifically aimed at minimizing energy consumption.

Considering the aforementioned, the energy efficiency of the 5G RAN is currently a major area of research due to the beyond 5G RAN architecture, where the Base Station (BS) is split into multiple components compared to previous monolithic approaches resulting in even greater energy demands than previous generations [3]. In fact, 5G RAN leverages a Network Function Virtualization (NFV) architecture, enabling greater flexibility and scalability for deploying both the 5G Core Network (5G CN) and 5G RAN, but may also generate higher energy usage compared to traditional RAN solutions. Despite that, this architecture presents opportunities to optimize energy consumption, as we demonstrate in our work.

In particular, 5G RAN is essentially a cloud-based version (known as C-RAN) that enables more flexibility in designing the architecture of the infrastructure since there is the possibility of choosing various functional splits, with 3GPP Option 2 being the most used one [4]. According to this split, the gNB has been decomposed to the Central Unit (CU), which hosts network layer functionalities such as the Packet Data Convergence Protocol (PDCP) and Radio Resource Control/Service Data Adaptation Protocol (RRC/SDAP), and the Distributed Unit (DU), which manages data link layer functions including the physical (PHY), medium access control (MAC), and radio link control (RLC) layers. Additionally, the Radio Unit (RU) remains at the edge of the network, directly interfacing with the antenna and focusing on radio frequency (RF) processing. The communication between the CU and DU is established over the F1 interface that is responsible for managing both control plane signaling and user plane data transmission. The F1 Application Protocol (F1AP) supports that a single CU can connect to multiple DUs, by improving the scalability of 5G C-RAN architectures. This can result in high traffic loads that the CU has to handle, resulting in increased processing demands and, consequently, power consumption.

Pushed by the recognized advantages of open ecosystems also in the 5G sector, the Open RAN (O-RAN) architecture adds specifications to the RAN that promote the con-

nection through open interfaces of the different network components, which are managed and optimized by Radio Intelligent Controllers (RIC) and the Session Management Orchestrator (SMO). Based on the response times requirements of the O-RAN system, two distinct types of RICs are introduced: the near Real-Time RIC (NearRT RIC) and the non-Near Real-Time RIC (non-NearRT RIC). In the context of aforementioned disaggregation of the architecture, O-RAN permits to develop different strategies in order to create different topologies of CUs and DUs placement. Moreover, the introduction of the RICs in O-RAN provides a powerful and flexible framework for improving resource management and in particular energy efficiency. This integration paves the way to develop RICs that implement energy management strategies such as dynamic scaling and resource management, the definition of energy metrics and support AI-driven optimization. As a result, RICs play a major role in creating energy-aware networks that can address the growing demands for sustainable and scalable connectivity.

Despite the potential of O-RAN, realistic experimental solutions for energy testing, metric evaluation, and assessment remain limited in the existing literature due to the complexity of such deployments. In this work, we use a real-world testbed environment to analyze the behavior of energy consumption in a 5G network infrastructure by considering different deployment scenarios with different data volume loads and different placements of disaggregated components. Based on those results and using a real-world dataset, we also deployed an execution environment where we prototyped and evaluated a scheduling policy that scales CU instances out or in to minimize the energy consumption of the 5G network. The main contributions of this work can be summarized as follows:

- Collecting energy measurements from real-world 5G experimental setups
- Analysis of the energy consumption behavior of 5G data plane entities
- Propose a policy for scheduling the number of CUs and implementing a proof of concept of this scheduler

Structure and Organization

The rest of the paper is organized as follows: Chapter 1 introduces the background of the topic that is needed to fully appreciate the rest of the document. Chapter 2 provides an overview of the related literature, focusing on existing approaches to minimize the energy consumption in the RAN along with scaling implementations in 5G networks. Chapter 3 provides our overall system model architecture as well as the experimental setup configuration, and Chapter 4 evaluates our implementation by detailing the experimental setup and presenting the results. Finally, in Chapter 4.4 we conclude our work and discuss future directions.

Chapter 1

Background

1.1 A Macro Overview

Mobile communication refers to the wireless transmission of voice, data, and multimedia between devices without the need for fixed physical connections. It enables users to stay connected while moving, using technologies like cellular networks (e.g., 3G, 4G, 5G), Wi-Fi, and satellite communication. Mobile communication relies on a network of base stations, radio frequencies, and protocols to facilitate seamless connectivity, supporting applications from phone calls and messaging to high-speed internet, IoT, and emerging technologies like smart cities and autonomous vehicles.

In the following sections, we will provide a brief overview of the evolution of mobile network standards over time and the key stakeholders involved in their development and maintenance. Additionally, we will explore fundamental concepts such as 5G and the cloud, along with their integration and impact on modern communication.

1.1.1 History of Standards

Mobile communication has evolved through multiple generations, each introducing significant advancements [5]. 1G (1980s) was the first analog cellular system, offering only voice calls with poor security and quality. 1G cellular networks were developed before the establishment of unified global standards and consequently, multiple regional standards emerged independently, such as [6] [7]. 2G (1990s) [8] introduced digital transmission, enabling SMS and basic data services. 3G (2000s) [9] brought mobile internet and multimedia capabilities, significantly improving data speeds. The launch of 4G (2010s) [10] [11] revolutionized mobile broadband with LTE technology, delivering high-speed internet, low latency, and support for HD video streaming. Now, 5G (2020s) [12] takes connectivity further with ultra-fast speeds, minimal latency, massive device connectivity (IoT), and network slicing for optimized performance. Looking ahead, 6G (expected in the 2030s) aims to integrate AI-driven networks, terahertz frequencies, and even holographic communication, pushing the boundaries of wireless technology even further [13].

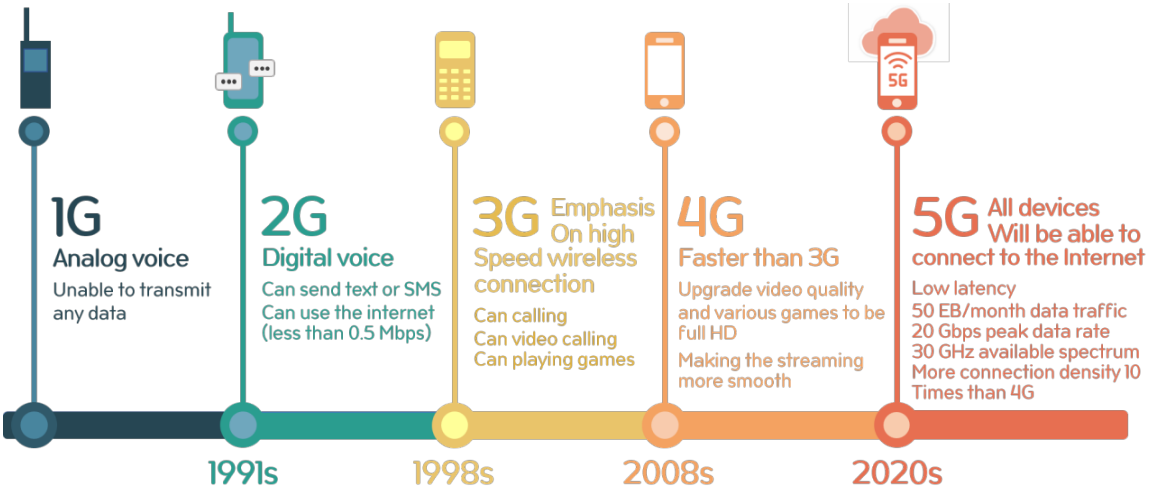


Figure 1.1: Roadmap from 1G to 5G

1.1.2 Players involved

After 2G, the evolution of mobile communication standards has been primarily driven by the 3rd Generation Partnership Project (3GPP). Despite its name still including "3G", the consortium has continuously developed and released new standards beyond third-generation networks. The transition from 2G to 3G marked a significant technological leap, and 3GPP has since defined 4G (LTE) with Releases 8 and 9 in 2010 [14], [15] and later shaped 5G with Release 17, finalized in 2022 [12], [16], [17].

However, 3GPP is not the only entity shaping mobile network technologies. National governments play a crucial role in regulating how the radio spectrum is allocated and used within their respective regions. Governments organize spectrum auctions, where Internet Service Providers (ISPs) and telecom operators bid to acquire exclusive frequency bands. These licensed bands allow operators to deliver controlled Quality of Service (QoS) to subscribers.

Not all frequency bands require licenses, though. Some frequency bands, such as the 2.4 GHz and 5 GHz bands used for Wi-Fi, are designated as unlicensed spectrum through international agreements. The Wi-Fi standard is defined by the IEEE 802.11 family [18]. Although these unlicensed bands offer flexibility for public use, licensed bands provide operators with greater control, ensuring greater reliability and performance, although at the cost of acquiring and maintaining the spectrum.

In addition to 3GPP and national regulatory agencies, a new organization called the Open-RAN Alliance (O-RAN) has emerged in recent years [19]. The O-RAN Alliance aims to "open up" the Radio Access Network (RAN) by promoting interoperability between different vendors' equipment, fostering a more flexible and competitive mobile infrastructure.

These key playersâ3GPP, national regulators, ISPs, and industry alliances like O-RANâall

contribute to the ongoing development and deployment of 5G, shaping the future of wireless communication.

1.1.3 What is 5G

5G [12] is the fifth generation of mobile communication technology, designed to provide faster speeds, lower latency, and greater capacity compared to previous generations. It operates on a mix of low, mid, and high-frequency bands, including millimeter waves (mmWave), to achieve ultra-high data rates. One of its key innovations is network slicing, which allows operators to create customized virtual networks for different applications, such as IoT, autonomous vehicles, and real-time remote surgery. 5G also enhances massive machine-type communications (mMTC), enabling billions of interconnected devices in smart cities and industrial automation. By significantly reducing latency to as low as 1 millisecond, 5G supports real-time applications that were previously impractical over wireless networks. We will deepen the understanding of 5G in Chapter 1.2.

1.1.4 What is the Cloud

The cloud refers to a network of remote servers that store, manage, and process data over the internet, eliminating the need for local storage and computing power [20]. In the context of 5G, cloud computing plays a crucial role in enabling edge computing, where data processing occurs closer to the user to reduce latency and improve performance. Cloud infrastructure supports virtualization and scalability, allowing mobile operators to dynamically allocate resources based on demand. It also facilitates the deployment of cloud-native 5G networks, where network functions are software-defined and can run on general-purpose hardware. The integration of cloud and 5G makes the network more efficient, flexible, and capable of handling vast amounts of data.

1.1.5 Cloud integration into 5G architecture

Unlike legacy systems that relied on vertically integrated architectures, 5G is designed to leverage cloud-native principles, including virtualization, containerization, and software-defined networking (SDN). This shift allows mobile networks to operate more like hyperscale cloud services, enabling dynamic resource allocation, on-demand scalability, and software-driven management of network functions. The adoption of these cloud-native architectures accelerates feature deployment, enhances network automation, and improves overall operational efficiency.

A key innovation in this transition is the emergence of private 5G networks, where enterprises deploy localized cellular networks tailored to specific applications. These deployments often consist of an on-premises data plane, responsible for handling user traffic at the edge, while the control plane remains hosted in the cloud. This architecture allows enterprises to manage their 5G networks similarly to cloud services, leveraging centralized management platforms akin to those used for cloud-based storage and compute resources.

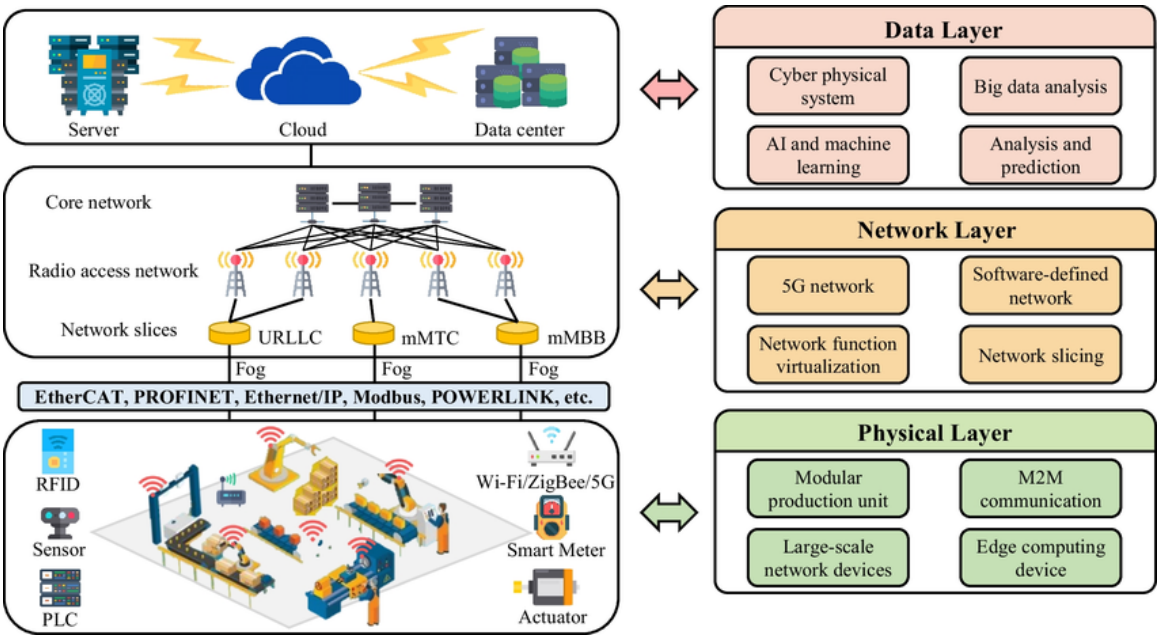


Figure 1.2: An example of logical architecture of a 5g smart factory. Figure from [21].

For example, as depicted in Figure 1.2 taken from [21], in smart manufacturing, a 5G-enabled factory can use cloud-based AI analytics to monitor equipment in real time. Sensors connected via 5G continuously send performance data to the cloud, where AI algorithms analyze it to predict failures before they occur. This allows for proactive maintenance, reducing downtime and increasing efficiency-all while benefiting from the high-speed, low-latency communication enabled by 5G.

1.2 In details Architecture of 5G

The Mobile Cellular Network provides wireless connectivity to User Equipment (UE), encompassing a vast range of devices that may be mobile, such as smartphones, connected cars, drones, industrial robots, agricultural machinery, and medical devices, or stationary, such as smart home appliances and IoT sensors deployed in fixed locations. Regardless of mobility, all these devices rely on the network's ability to establish seamless and reliable wireless connections to support a variety of applications, from basic voice communication to high-bandwidth, low-latency industrial automation.

As depicted in Figure 1.3, a mobile network consists of two fundamental components: the Radio Access Network (RAN) and the Mobile Core. The RAN is responsible for handling the wireless communication between the UE and the network infrastructure, ensuring efficient radio resource management and seamless handover between base stations. The Mobile Core, on the other hand, manages user authentication, mobility tracking, data rout-

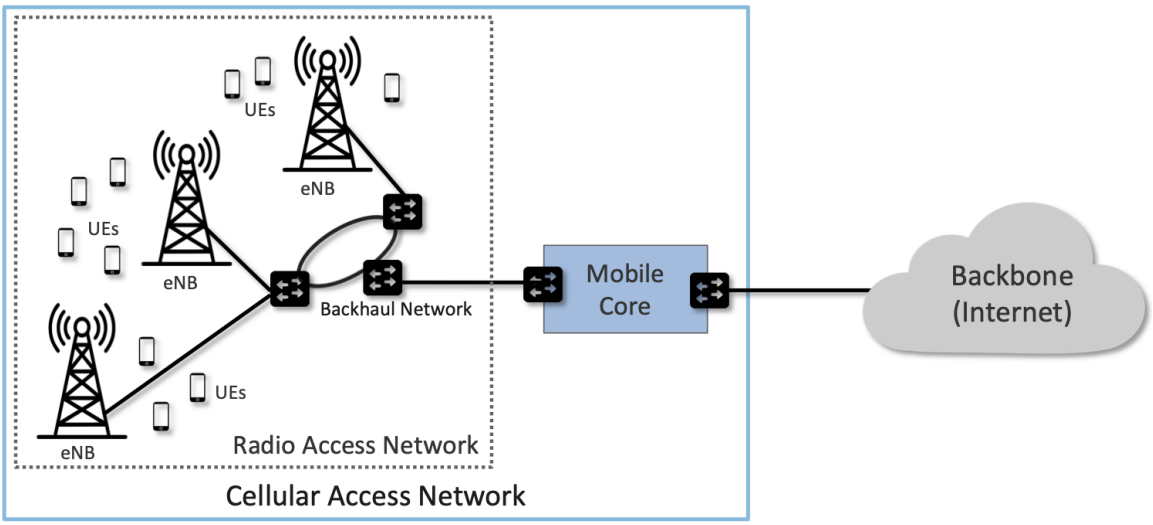


Figure 1.3: The figure illustrates a typical cellular network architecture, comprising the Radio Access Network (RAN) and the Core Network. The RAN consists of User Equipment (UEs) communicating with eNodeBs (eNBs), which are interconnected via the Backhaul Network. The Mobile Core processes and manages network traffic before connecting to the Backbone (Internet) for external communication. Figure from [22].

ing, and service delivery, acting as the backbone of the entire system. These two critical components are interconnected through the Backhaul Network, which typically relies on wired infrastructure. Technologies such as Passive Optical Networks (PON), which facilitate Fiber-To-The-Home (FTTH) connections, and Switched Ethernet, which provides high-speed, low-latency packet switching, are commonly used to ensure efficient and scalable backhaul connectivity. Despite being an essential element of the overall architecture, the backhaul network is not explicitly defined within the 3GPP standards, allowing operators to implement solutions that best suit their deployment scenarios and operational needs.

One of the defining aspects of modern 5G networks is their alignment with the principles of Software-Defined Networking (SDN). Unlike traditional networks, where hardware-based configurations dictate how traffic is managed and forwarded, SDN introduces a centralized control plane that dynamically orchestrates network operations through software-defined policies. This architectural shift enables greater flexibility, allowing operators to optimize resource allocation, implement traffic engineering strategies, and rapidly adapt to changing network demands. By decoupling the control and data planes, SDN facilitates automation, programmability, and the seamless deployment of new network functions without requiring extensive hardware modifications.

In the following sections, we will explore how data is transmitted between user devices via radio communication and examine the advanced techniques introduced in the 5G standard to enhance efficiency and performance. Additionally, we will provide a detailed breakdown of the RAN and Mobile Core, highlighting their key roles and functionalities within the

overall network infrastructure.

1.2.1 Radio Transmission

The main aim of the cellular network service is to deliver a certain quality of service to all the UEs in a determined converage area, while also maintaining this connectivity when the users are moving. Since multiple UEs wants to connect to the same coverage area, the infrastructure wants to minimize the efficiency of the spectrum usage, which is finite and costly. To this end, the infrastructure plays in a higly dynamic and adaptive approach expecially in order to deal with the coding, modulation and scheduling roles. With 5G these approaches are advanced so a higher level of sofistication.

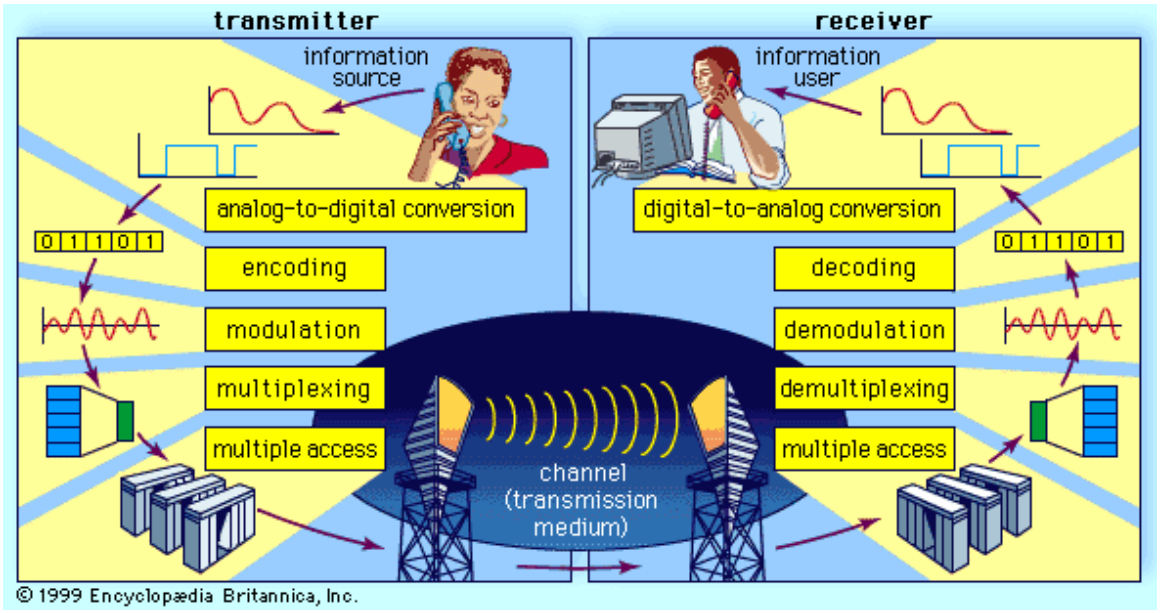


Figure 1.4: This figure illustrates the fundamental processes of a digital communication system. The transmitter converts an analog signal from an information source into a digital format through analog-to-digital conversion, followed by encoding, modulation, multiplexing, and multiple access before transmission via the channel (transmission medium). The receiver performs the reverse operations: multiple access, demultiplexing, demodulation, decoding, and digital-to-analog conversion to reconstruct the original signal for the information user.

Figure 1.4 shows high level detailed pipeline that the information follows in order to be transmitted with a radio signal in a cellular network. In a cellular network, transmitting information starts with converting the original signal, usually an analog voice or data input, into a digital format. This process, called analog-to-digital conversion, involves sampling the continuous signal at discrete intervals and quantizing those samples into binary values, making it suitable for digital processing.

Once digitized, the data undergoes encoding, which adds redundancy and error detection

or correction bits to protect the information from corruption during transmission. Encoding ensures that even if some bits are lost or altered due to interference, the receiver can still reconstruct the original message accurately.

The encoded signal is then prepared for transmission through modulation, where it is mapped onto a high-frequency carrier wave. This step allows the signal to travel over long distances by shifting it to a frequency range suitable for wireless communication. Different modulation techniques, such as amplitude, frequency, or phase modulation, are chosen based on network requirements and interference conditions.

To efficiently use the available bandwidth, multiple signals are combined using multiplexing. This technique allows different data streams to share the same transmission medium by allocating separate frequency bands, time slots, or codes, depending on the multiplexing method employed. It ensures that multiple users can transmit their data without interference.

In a cellular system, multiple users must access the shared communication resources, which is managed through multiple access techniques. These methods, such as FDMA, TDMA, CDMA, or OFDMA, ensure that numerous users can transmit and receive data simultaneously without significant collisions or loss of quality. Later in this section, we talk more on details about the evolution of 5G multiplexing from the 4G ones.

Finally, the prepared signal is sent over the physical medium via channel transmission an an electromagnetic wave over an assigned carrier frequency. The wireless channel introduces noise, interference, and potential signal degradation, requiring robust transmission techniques to maintain data integrity.

On the receiving end, the process occurs in reverse. The received signal is first extracted from the transmission channel, where it undergoes demodulation to recover the baseband data from the carrier wave. Multiple access techniques separate individual signals, and demultiplexing retrieves the original data streams. Decoding removes redundancy and reconstructs any lost or corrupted information, ensuring data accuracy. Finally, digital-to-analog conversion transforms the digital signal back into an analog form, such as sound in a voice call, making it understandable to the end user.

Since we are talking about a radio propagation of a signal, the signal bounces off various stationary and moving objects, following multiple paths from the transmitter to the receiver, who may also be moving. Therefore, the voyage of the radio signal from the emitter to the receiver folows multiple paths and overall the multiple paths of each UE can interfere with each other constructively and destructively. Another complication is that towers where the radio signal is generated/received are now equipped with multiple antennas, each transmitting at a different but overlapping direction. This technology is called Multiple Input Multiple Output (MIMO). One of the most important consequence of these factors is that the transmitter has to receive feedbacks from every receiver to manage the spectrum usage. To this end, in the 3GPP specifications there is a metric called Channel Quality Indicator (CQI), and the receiver send a message with this indicator periodically. Concretely, the CQI communicates the signal to noise ratio, which tells how challenging was for the receiver to recover the data bits information cleaning it from the noise. The

scheduler of the transmitter than uses this indicator to adapt how it allocates the available radio spectrum, which coding and modulation scheme to employ. We talk more in details about the novel properties of the 5G scheduler in a specific paragraph later in this section.

Evolution of multiplexing and the Scheduler from 4G to 5G

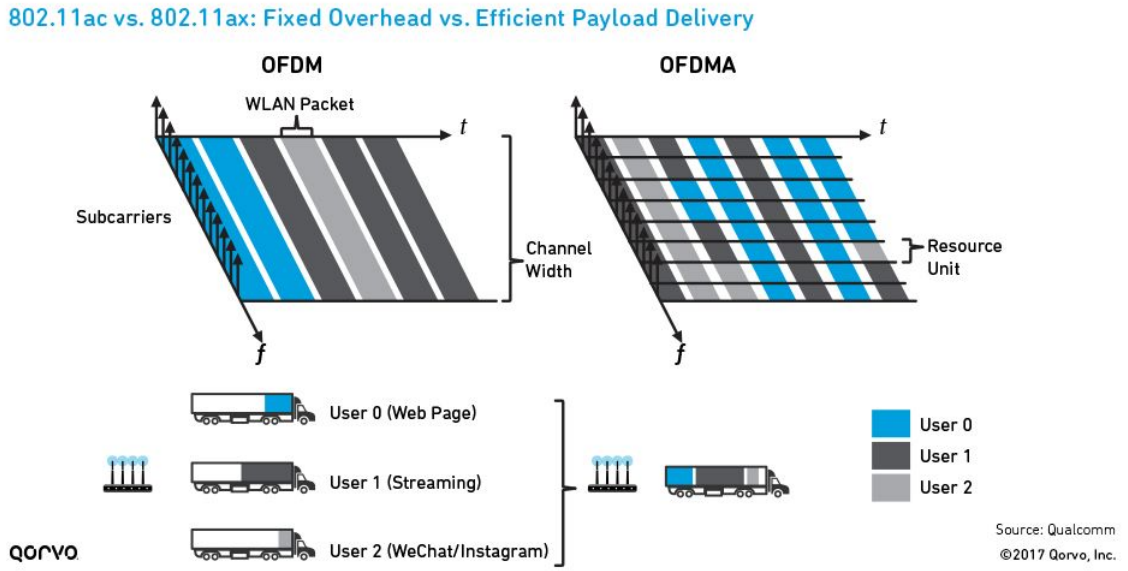


Figure 1.5: OFDM versus OFDMA. Figure from [23]

Understanding how multiplexing works in 4G provides a foundation for grasping the advancements in 5G, as both rely on a technique called Orthogonal Frequency-Division Multiple Access (OFDMA). OFDMA is a specific implementation of Orthogonal Frequency-Division Multiplexing (OFDM) that allows multiple users to share a wireless channel efficiently. In 4G networks, OFDMA distributes data across multiple orthogonal subcarrier frequencies, ensuring they do not interfere with each other. These subcarriers are grouped into sets of 12 subcarriers per user, with each subcarrier being independently modulated.

The "Multiple Access" aspect of OFDMA means that different users can simultaneously transmit data on separate subcarriers and for varying durations, enabling efficient spectral utilization. Each of these subcarriers operates within a narrow frequency band of 15 kHz, and the encoding process ensures minimal data loss due to interference. While 4G employs OFDMA for downlink (base station to user device) transmissions, it uses a different technique for uplink (user device to base station), which is not applicable to 5G and therefore not covered here.

Since OFDMA distributes frequency and time resources, the radio spectrum can be visualized as a two-dimensional grid. The vertical axis represents subcarriers, while the horizontal axis represents time slots for transmitting data symbols. The fundamental unit

in this structure is the Resource Element (RE), which consists of one subcarrier frequency (15 kHz wide) and the duration required to transmit one OFDMA symbol.

The number of bits that can be transmitted in a symbol depends on the modulation scheme used. For instance, if 16-QAM (Quadrature Amplitude Modulation) is used, each symbol carries 4 bits, while 64-QAM increases this to 6 bits per symbol. The choice of modulation depends on the channel conditions, allowing dynamic adaptation: when signal quality is high, the network can use higher-order modulation to transmit more bits per symbol, increasing data rates.

The scheduler in a 4G system is responsible for assigning Resource Elements to users. These assignments occur in 1-millisecond Transmission Time Intervals (TTI), during which the scheduler determines which users receive resources and in what proportion. To ensure efficiency, the smallest unit of allocation is a Physical Resource Block (PRB), which consists of 84 Resource Elements (7 OFDMA symbols per 12 subcarriers). The scheduler continuously assigns PRBs to users based on their data demands, signal quality, and quality-of-service (QoS) requirements.

The Channel Quality Indicator (CQI) plays a key role in this process. Every millisecond, user devices send CQI feedback to the base station, reporting signal strength and interference conditions. The scheduler then uses this feedback to determine the best modulation scheme and coding rate for each user in the next scheduling cycle.

Another critical parameter influencing scheduling is the QoS Class Identifier (QCI), which defines different traffic priorities. There are 26 QCI classes in 4G, divided into Guaranteed Bit Rate (GBR) and non-GBR categories. GBR classes have a minimum data rate guarantee, while non-GBR classes are best-effort services with no strict bandwidth allocation. The scheduler prioritizes traffic based on the QCI level, ensuring mission-critical applications receive the necessary resources while lower-priority data is handled accordingly.

It is important to note that OFDMA itself is not a modulation technique; rather, it provides a framework within which specific modulation and coding schemes are selected dynamically. The scheduler determines which modulation scheme (e.g., QAM) to use, and it also controls coding parameters, such as those used in Turbo coding, to ensure data integrity.

Finally, modern cellular networks employ Multiple Input Multiple Output (MIMO) technology. This means the scheduler also decides which antennas or combination of antennas should transmit to specific users, further optimizing network performance.

The 4G scheduler must make multiple decisions simultaneously, including:

1. Which users to serve at a given moment
2. How many Resource Elements to allocate to each user
3. What coding and modulation levels to apply
4. Which antennas should transmit the data

These decisions form a complex optimization problem that network vendors solve through proprietary scheduling algorithms. This becomes even more critical in 5G, where the sched-

uler has even more variables to consider.
While 4G relies on a fixed waveform structure, 5G introduces greater flexibility in spectrum scheduling, allowing it to support a wider range of devices and applications. Instead of defining a single waveform like LTE, 5G supports multiple waveforms, each optimized for different parts of the radio spectrum. The choice of waveform influences how subcarrier intervals and scheduling are managed.

In 5G, the spectrum is divided into three main frequency ranges:

1. Sub-1 GHz bands: Optimized for long-range mobile broadband and IoT applications.
2. 1-6 GHz bands: Focused on providing a balance between range and capacity, suitable for both broadband and mission-critical services.
3. Above 24 GHz (millimeter-wave bands): Enables extremely high data rates but is limited to short distances and line-of-sight connections.

Each of these bands has different numerology, which refers to the spacing of subcarriers and scheduling intervals.

1. In Frequency Range 1 (410 MHz - 7125 MHz), 5G allows channel bandwidths up to 100 MHz, with subcarrier spacings of 15, 30, or 60 kHz, corresponding to scheduling intervals of 0.5, 0.25, and 0.125 ms, respectively.
2. In Frequency Range 2 (24.25 GHz - 52.6 GHz), channel bandwidths range from 50 MHz to 400 MHz, with subcarrier spacings of 60 or 120 kHz, both using a scheduling interval of 0.125 ms.

Unlike 4G, where Resource Elements (REs) are fixed, 5G dynamically adjusts resource block sizes based on the waveform in use. The scheduler can modify subcarrier spacing and time duration to better suit different traffic types and channel conditions.

The 5G scheduler operates similarly to its 4G counterpart, using CQI feedback from users and a Quality of Service (QoS) indicator called 5QI to determine resource allocation. However, 5QI offers more detailed differentiation than 4G's QCI. Each 5QI value defines attributes such as:

1. Resource type (Guaranteed Bit Rate, Delay-Critical GBR, or Non-GBR)
2. Priority level
3. Packet delay budget
4. Packet error rate
5. Maximum data burst size
6. Averaging window for throughput calculations

Unlike the one-to-one mapping between users and QCI in 4G, 5G allows a single user to send multiple types of traffic simultaneously, each associated with a different 5QI class. This enables better handling of heterogeneous traffic, such as video streaming, voice calls, and IoT data, within the same connection.

Overall, 5G's scheduler has greater flexibility than 4G, as it can:

1. Dynamically adjust subcarrier spacing
2. Modify resource block sizes based on real-time traffic demands
3. Optimize transmission based on CQI feedback and 5QI priorities
4. Allocate radio resources efficiently across different frequency bands and device categories

This enhanced adaptability is crucial for supporting emerging 5G applications, such as ultra-reliable low-latency communication (URLLC), massive IoT deployments, and high-speed broadband services. The increased complexity in 5G scheduling reflects the need for more advanced optimization algorithms to ensure seamless connectivity across diverse use cases.

Slicing

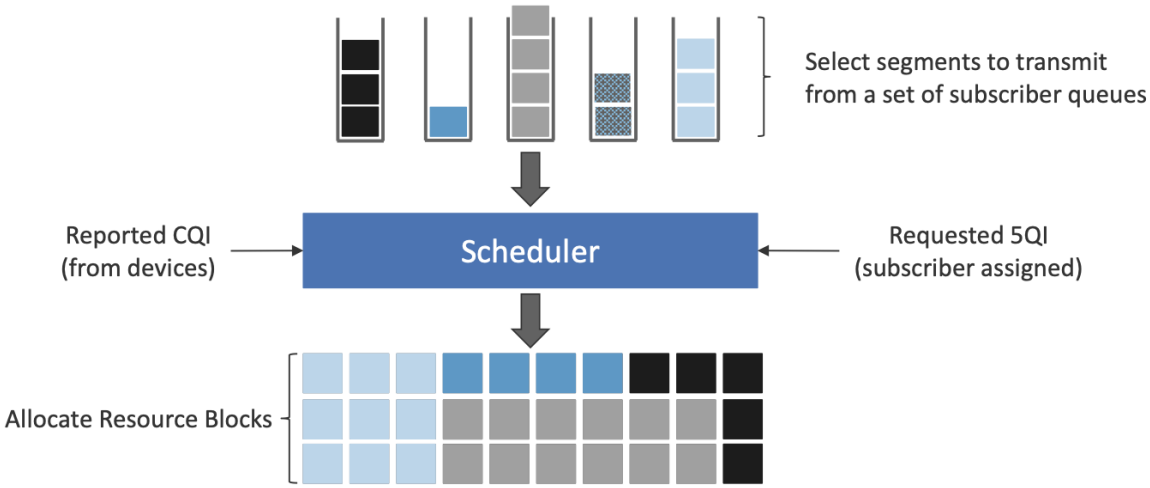


Figure 1.6: This figure illustrates the main inputs that a schduler needs to make its decision. The scheduler selects data segments from subscriber queues based on reported Channel Quality Indicator (CQI) from devices and the Requested 5G QoS Identifier (5QI) assigned to subscribers. The scheduler then allocates resource blocks to optimize network performance and ensure efficient transmission. Figure from [22]

Until now, the discussion has assumed that a single scheduler is sufficient for managing all types of network traffic. However, different applications have varying requirements: some

prioritize low latency, while others focus on maximizing bandwidth. Instead of designing a highly complex scheduler capable of accounting for numerous factors at once, 5G introduces an innovative approach that enables dividing the available radio resources into separate logical units, a technique known as slicing.

At its core, slicing introduces an additional layer of abstraction between the scheduler and the actual physical radio resources. Rather than directly assigning physical resource blocks to users, a virtual layer is introduced where traffic is first mapped onto Virtual Resource Blocks (VRBs), which are then mapped onto Physical Resource Blocks (PRBs). This approach is widely used in computing systems where resource allocation needs to be flexible, similar to how a hypervisor manages virtual machines by abstracting physical hardware resources. In the case of wireless networks, the hypervisor-like component manages the translation between virtual and physical resource blocks without concern for which user or application is affected by each allocation decision.

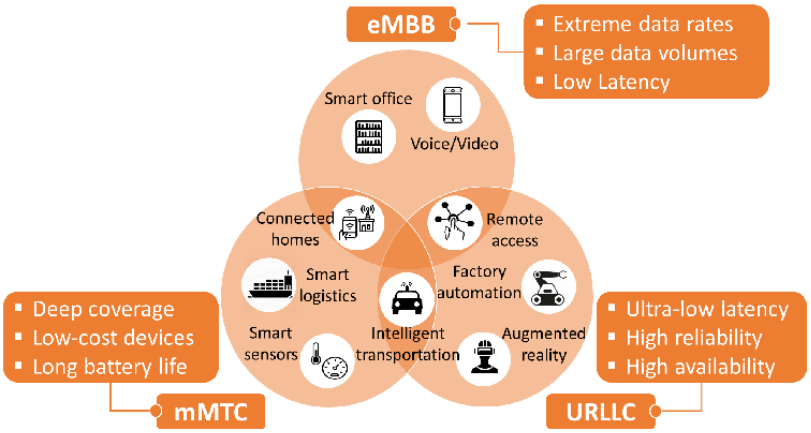


Figure 1.7: This figure illustrates the three main service categories of 5G technology: eMBB (Enhanced Mobile Broadband) supports applications requiring extreme data rates, large data volumes, and low latency, such as smart offices, voice/video streaming. mMTC (Massive Machine-Type Communications) is designed for deep coverage, low-cost devices, and long battery life, enabling applications like connected homes, smart sensors, and smart logistics. URLLC (Ultra-Reliable Low Latency Communications) ensures ultra-low latency, high reliability, and high availability, essential for factory automation, remote access, augmented reality, and intelligent transportation.

By decoupling virtual resources from physical ones, multiple independent scheduling mechanisms can operate within their own designated virtual resource sets. This means that different slices of the network can be assigned to different types of traffic. For example, one slice may be dedicated to high-bandwidth applications such as streaming services, while another may be optimized for ultra-low-latency applications like industrial automation or real-time gaming. Additionally, network operators can reserve a portion of resources for high-priority users, such as public safety communications, while allowing remaining capacity to be dynamically shared among all other users.

This slicing mechanism is highly flexible. While one approach is to allocate fixed portions of the available spectrum to each slice, it is also possible to implement a more dynamic allocation model. In this case, unused resources from one slice can be temporarily reallocated to another, as long as they can be reclaimed when needed. This is similar to work-conserving scheduling techniques used in traditional network queues, where idle capacity in one queue can be used by another traffic flow instead of remaining unused.

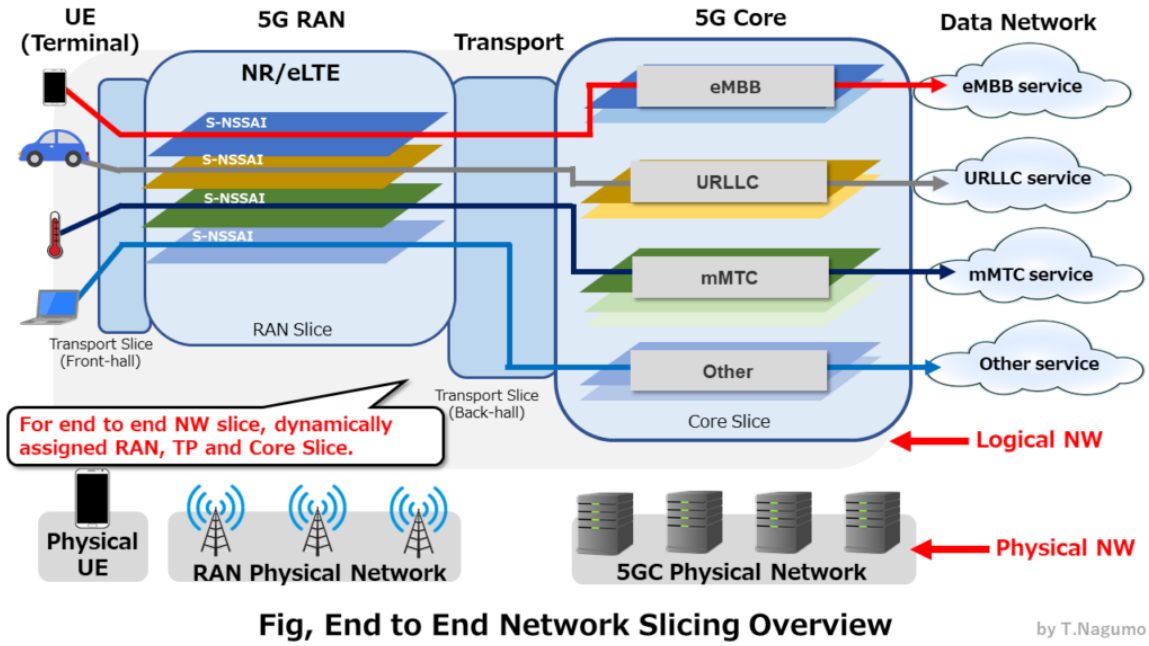


Figure 1.8: 5G end to end network slicing through the logical infrastructure

The increased flexibility in scheduling and resource allocation introduced by 5G does not simply improve efficiency: it fundamentally transforms how wireless networks operate, enabling a range of new applications. The 5G radio interface, often referred to as New Radio (NR), is designed to support use cases that extend beyond just delivering higher bandwidth.

One of the most significant advancements is the ability to modify the waveform dynamically. Unlike previous generations, where resource elements had fixed properties, 5G allows the network to adjust waveform characteristics on the fly, effectively changing the size and number of resource units available. This flexibility is crucial for applications requiring low and predictable latency, as it enables more precise scheduling decisions.

Another key improvement involves how different traffic types share the available spectrum. In 4G, multiplexing techniques allowed multiple users to share the frequency and time domains, but only for downlink transmissions (from the base station to the user). In contrast, 5G NR applies multiplexing in both time and frequency domains for both downlink and uplink transmissions. This added flexibility allows for finer control over scheduling,

ensuring that latency-sensitive applications receive prioritized access to network resources.

The introduction of higher-frequency bands, particularly those in the millimeter-wave (mmWave) spectrum above 24 GHz, plays a crucial role in expanding network capacity. These higher frequencies enable extremely wide bandwidths, which not only support high-throughput applications but also allow for more precise scheduling. The finer granularity of resource blocks in mmWave frequencies, where scheduling intervals can be as short as 0.125 milliseconds, ensures that applications requiring ultra-low latency receive the responsiveness they need.

Beyond increasing bandwidth and reducing latency, 5G also brings improvements in supporting a massive number of connected devices. Unlike traditional mobile broadband users, IoT devices have unique requirements: many transmit small amounts of data sporadically, require long battery life, and must operate with minimal hardware complexity. To accommodate these needs, earlier LTE releases introduced specialized technologies like Massive Machine-Type Communications (mMTC) and Narrowband IoT (NB-IoT), which optimized wireless transmissions for low-power, low-bandwidth devices. In 5G, these technologies continue to evolve, with NR-Light introduced as a simplified version of New Radio designed specifically for massive IoT applications.

5G NR also supports dynamic partitioning of the available bandwidth, allowing different traffic types to operate in their own dedicated portions of the spectrum. This partitioning aligns closely with the concept of slicing, enabling the network to allocate resources in a way that is optimized for each application's needs. Once traffic flows are categorized into slices, the 5G scheduler can tailor resource allocation strategies to match the specific requirements of each slice, whether they prioritize bandwidth, latency, or power efficiency.

By combining these advancements-dynamic waveforms, more flexible multiplexing, high-frequency spectrum access, improved IoT connectivity, and resource slicing-5G provides a level of adaptability far beyond previous wireless technologies. Rather than simply increasing data rates, it fundamentally reshapes how wireless networks operate, ensuring they can efficiently support a diverse range of applications in an increasingly connected world.

1.2.2 RAN

The RAN is a collection of Base Transceiver Stations (BTS), commonly called *base stations*, and it manages the radio resources (the spectrum). Figure 1.9 shows an example of a base station of a RAN, which are called *evolved NodeB* (eNodeB or eNB) in 4G and *next generation NodeB* (gNodeB or gNB) in 5G. A BTS consists of several key components, each playing a vital role in ensuring seamless communication.

At its core, a BTS includes a transceiver station, which manages the transmission and reception of radio signals. Traditionally, this unit was positioned on the ground, but modern deployments often place it closer to the antenna to reduce power loss. Another critical element is the RF signal (radio frequency signal), which carries voice, data, and control information between the BTS and mobile devices. This signal needs to be amplified and processed correctly to maintain a strong connection.

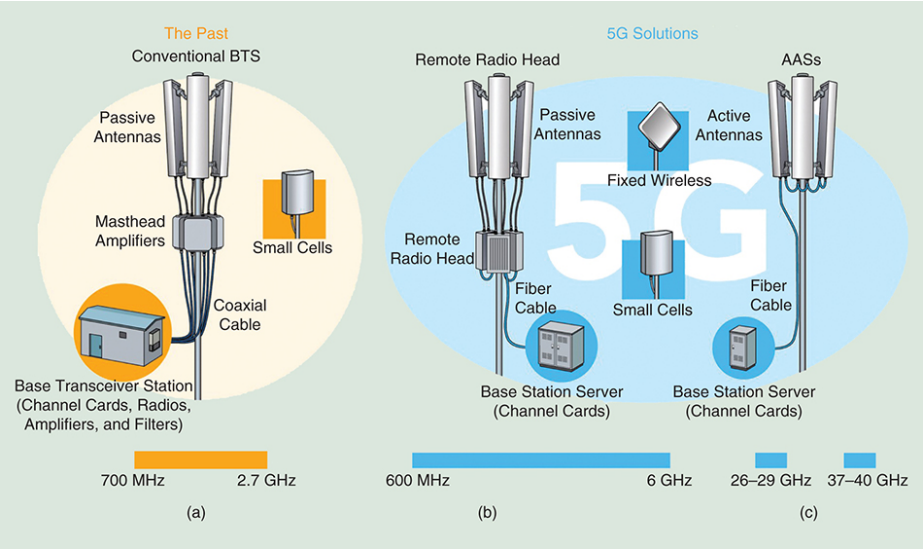


Figure 1.9: Example of a e NodeB or a g NodeB. Figure from [24]

To facilitate this, BTS architectures include power amplifiers, which boost the RF signal strength to ensure it can travel long distances without degrading. In older systems, these amplifiers were located at ground level, requiring the signal to travel through long feeder cables to reach the antenna at the top of the mast. However, this design was inefficient due to energy loss along the cable. To overcome this, modern BTS designs use remote radio heads (RRHs), which move the power amplification process closer to the antenna.

Another crucial element of the BTS is the base station gateway, which handles data processing and connects the BTS to the mobile core network. It contains channel cards, specialized components that manage different frequency bands and communication channels, ensuring that multiple users can connect to the network simultaneously.

As mobile technology has evolved, new BTS designs incorporate Active Antenna Systems (AAS), which integrate all RF components directly into the antenna itself. This integration is essential for enabling massive MIMO (multiple-input, multiple-output) technology, where multiple antennas work together to significantly increase data rates and network capacity.

With the introduction of millimeter-wave (mmWave) frequencies, BTS designs have also adapted to support fixed wireless access (FWA), a technology that provides high-speed internet by replacing traditional wired connections with wireless point-to-point links. These advances allow operators to expand their networks efficiently, making mobile connectivity faster, more reliable, and capable of supporting the growing demand for data-intensive applications.

Ran Protocol Stack

The RAN (Radio Access Network) protocol stack in 5G is a layered architecture responsible for managing communication between the User Equipment (UE) and the 5G Core Network (5GC). It consists of multiple protocol layers that handle different aspects of data transmission, including signaling, encryption, error correction, and transport. In 4G all of the protocol stack was operating inside the Baseband Unit (BBU), but with 5G there is a disaggregation of this stack in multiple devices, as shown in the right part of Figure 1.10. This highlights that in 5G RAN (Radio Access Network), there is a fundamental separation between the User Plane (U-Plane) and the Control Plane (C-Plane) to optimize network efficiency, scalability, and performance. This separation allows the network to independently manage data transmission and signaling, ensuring flexible resource allocation and improved support for diverse applications. Going deeply in details, Figure 1.11 shows the meaning

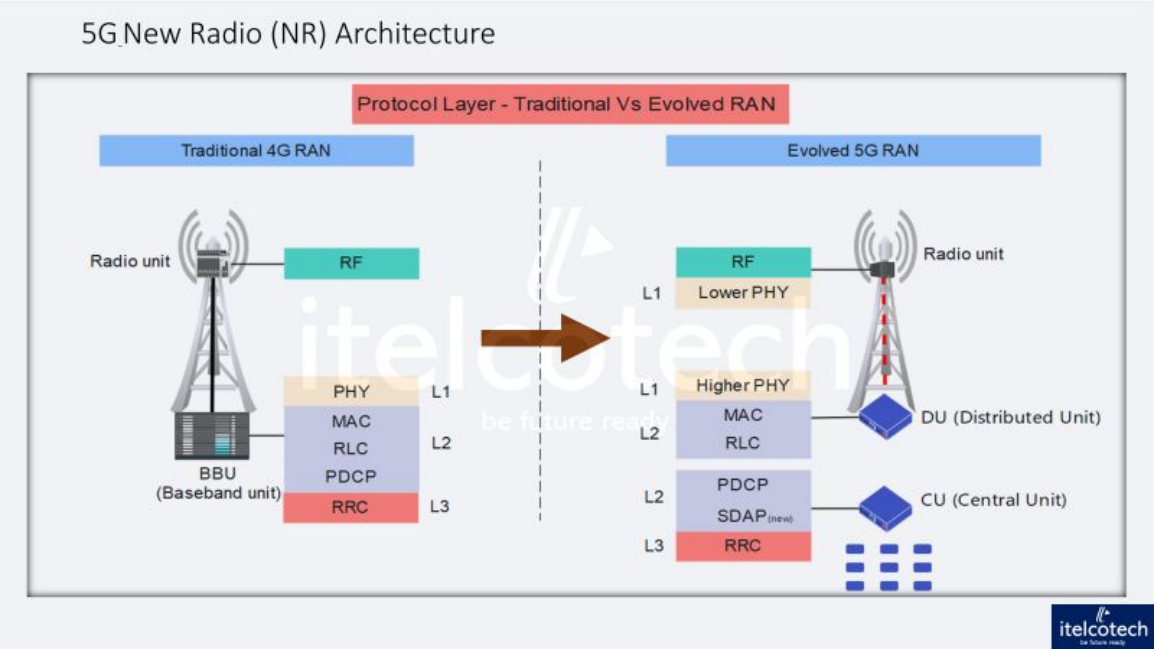


Figure 1.10: The image illustrates the evolution from a traditional 4G RAN architecture to an evolved 5G RAN. In the 4G model, all processing functions, including RF, PHY, MAC, RLC, PDCP, and RRC, are centralized within the Baseband Unit (BBU). In contrast, the 5G architecture introduces a more flexible and efficient approach by distributing functions across the Radio Unit (RU), Distributed Unit (DU), and Central Unit (CU). The RU handles RF and lower PHY processing, while the DU manages higher PHY, MAC, and RLC functions. The CU is responsible for PDCP, SDAP, and RRC, enabling improved scalability, performance, and network efficiency.

of each layer of the 5G protocol stack. In particular, the User Plane (U-Plane) is responsible for carrying the actual user data, such as internet traffic, voice, and video streams. It consists of multiple protocol layers, including SDAP (Service Data Adaptation Protocol),

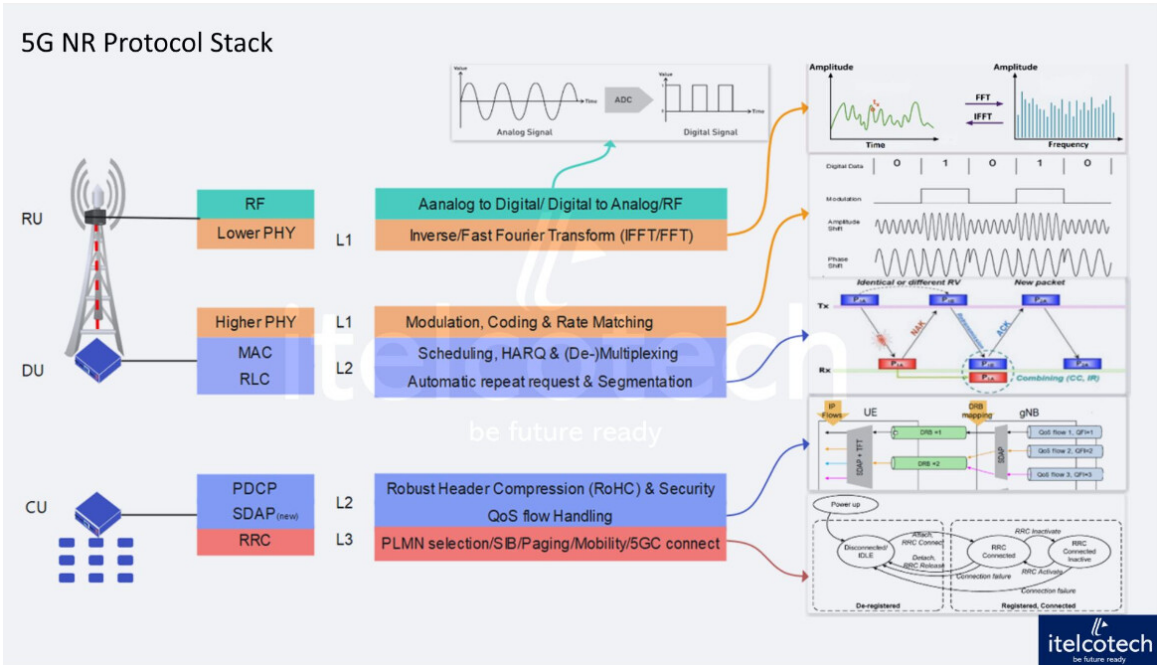


Figure 1.11: Every layer of the 5G protocol stack in details.

PDCP (Packet Data Convergence Protocol), RLC (Radio Link Control), MAC (Medium Access Control), and PHY (Physical Layer).

The Physical Layer (PHY) is the lowest layer in the RAN stack, responsible for the actual transmission and reception of radio signals. It handles modulation, coding, Multiple Input Multiple Output (MIMO), beamforming, and carrier aggregation. The PHY layer operates over OFDM (Orthogonal Frequency Division Multiplexing) waveforms and ensures efficient spectrum usage.

The Medium Access Control (MAC) Layer sits above the PHY layer and manages resource allocation and scheduling. It handles hybrid automatic repeat request (HARQ) mechanisms, ensuring error correction and retransmission of lost packets. The MAC layer also takes care of uplink and downlink scheduling by dynamically allocating radio resources based on network conditions and Quality of Service (QoS) requirements.

The Radio Link Control (RLC) Layer is responsible for segmenting and reassembling data packets before they are transmitted over the *air interface*, which refers to the wireless communication link between the UE and the gNB, where data is sent and received using radio waves. Because wireless communication is inherently less reliable than wired connections (due to interference, fading, congestion, and mobility), the RLC layer provides mechanisms to segment, reassemble, and manage data transmission since different types of data have different reliability and latency requirements. For example, some applications need guaranteed delivery, while others prioritize low latency over reliability. RLC operates in three modes: Transparent Mode (TM), Unacknowledged Mode (UM), and Acknowledged

Mode (AM). TM is used for essential control messages, UM prioritizes speed over reliability, and AM ensures reliable data transfer at the cost of additional delay.

The Packet Data Convergence Protocol (PDCP) Layer ensures security and efficiency in data transmission. Its functions include header compression (ROHC - Robust Header Compression), ciphering, integrity protection, and in-sequence delivery of data packets.

The Service Data Adaptation Protocol (SDAP) Layer is introduced in 5G NR to handle Quality of Service (QoS) flows. It maps different QoS flows to data radio bearers (DRBs), ensuring that network resources are allocated based on application needs (e.g., ultra-low latency for autonomous vehicles or high throughput for streaming).

The Control Plane (C-Plane) is dedicated to managing and controlling the UE's connection to the network. It consists of signaling protocols and procedures that ensure proper mobility management, security, and session establishment. Key components of the Control Plane include RRC (Radio Resource Control), NAS (Non-Access Stratum), and higher-layer signaling protocols.

The RRC (Radio Resource Control) Layer is responsible for managing the overall radio connection between the UE and the gNB (5G base station), ensuring that the radio connection is efficiently established, maintained, and released as needed. One of the primary functions of the RRC layer is connection management, which includes setting up and tearing down radio connections between the UE and the gNB. When a UE needs to access the network, it initiates an RRC connection request, and if the network accepts the request, an RRC connection is established. Once the data transmission is complete or if the UE enters an idle state, the network may release the connection to save resources. As one of

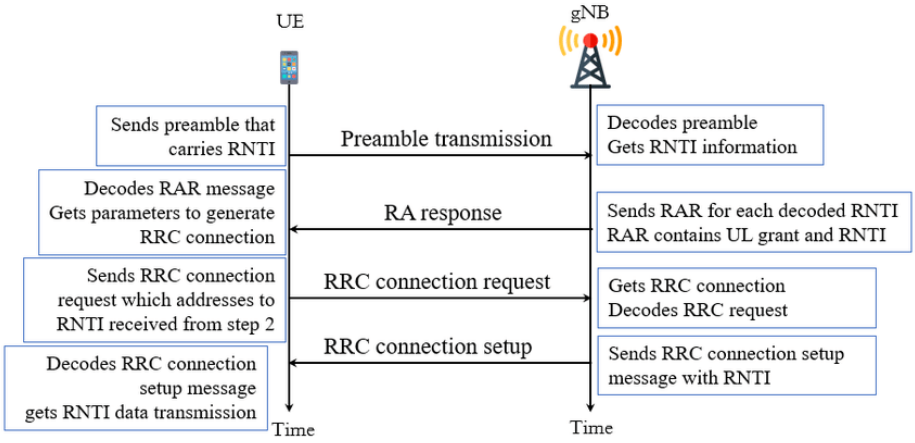


Figure 1.12: Contention based random access procedure, example to show one of the RRC's functionalities.

the policy examples for connection management where RRC layer is involved, Figure 1.12, the messages exchanged by the UE and the gNB in order to establish a connection between the two with a Contention Based RACH procedure. Another critical role of the RRC layer

is mobility management, which ensures seamless connectivity when the UE moves across different gNB coverage areas. When the UE moves from one cell to another, the RRC layer handles handover procedures, allowing for a smooth transition without dropping the connection. This is particularly important for users engaged in real-time applications like video calls, online gaming, or autonomous vehicle communications. The RRC layer is also responsible for paging and system information broadcasting. Paging is a mechanism used by the network to locate and notify UEs about incoming calls, messages, or other events when they are in an idle state. System information broadcasting involves transmitting essential network parameters that UEs need for proper operation, such as network configurations, frequency information, and cell selection criteria. This information is continuously sent over the air so that newly connected UEs or roaming devices can access the network efficiently.

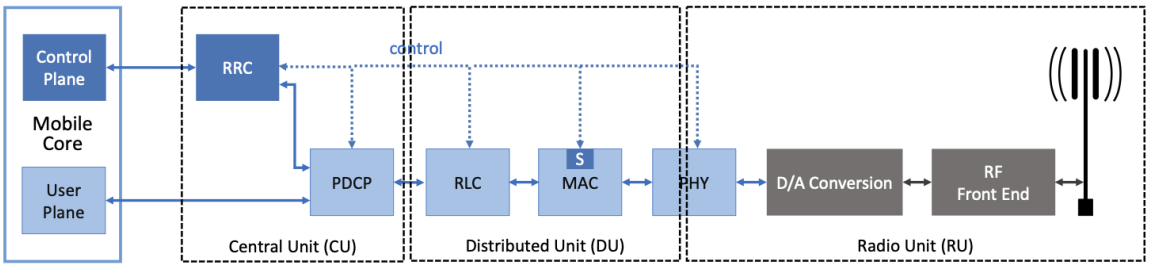


Figure 1.13: Split Option 7-2x of the 5G RAN.

Following the division principle between the User and the Control Plane, the gNB RAN can logically be divided into three parts: the Central Unit (CU), the Distributed Unit (DU), and the Radio Unit (RU). The assigned layers to these parts vary among different split options. One of the most popular ones is the Split Option 7-2x, depicted in Figure 1.13. In this type of split, the CU covers the protocol stack with PDCP, SDAP, and RRC, while the DU handles the RLC, MAC, and the Higher PHY layer. The rest of the physical layer, called the Lower PHY, along with the RF signal, is assigned to the Radio Unit (RU).

Furthermore, as shown in Figure 1.14, the CU itself can be further divided into two logical entities: the CU-Control Plane (CU-CP) and the CU-User Plane (CU-UP). The CU-CP is responsible for handling control signaling, including RRC functions, mobility management, and connection control, while the CU-UP manages user data processing and forwarding at the PDCP layer. Thanks to this layered division of functionalities into modular components, the logical architecture can be mapped onto different physical devices, allowing for flexible deployments and optimized network operations.

One of the main reasons for these separations is independent scaling. In traditional networks where the control and user planes were tightly integrated, increasing data traffic required scaling the entire network, including signaling functions that did not necessarily need expansion. By separating them, operators can scale the User Plane independently to handle large amounts of data traffic without increasing the burden on control functions. This is particularly beneficial in scenarios like video streaming, cloud gaming, and large-scale IoT deployments where data traffic fluctuates.

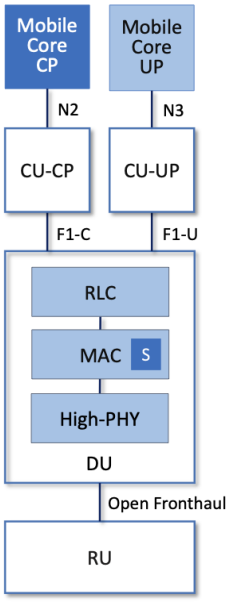


Figure 1.14: Split Option of 5G RAN were also the CU is split in its intrinsic User Plane and Control Plane functionalities.

Another important reason is flexibility in network deployment. With CUPS, operators can distribute network functions across different locations based on efficiency and latency requirements. The Control Plane can be centralized in cloud data centers to simplify network management, while User Plane functions can be placed closer to the user (such as at edge computing nodes) to reduce latency and improve real-time application performance. This is especially critical for applications like autonomous driving, industrial automation, and remote surgery, where even a few milliseconds of delay can have significant consequences.

The separation also enhances reliability and security by allowing independent handling of control signaling and user data. The Control Plane manages authentication, mobility, and security procedures, ensuring that only authorized users can access the network. Meanwhile, the User Plane is responsible for transmitting data efficiently without being affected by control-related operations. This means that even if there are issues with signaling or control messages, data transmission can continue uninterrupted.

Additionally, this separation enables network slicing, a key feature in 5G that allows multiple virtual networks to coexist within the same physical infrastructure. Different slices can be optimized for specific use cases, such as ultra-reliable low-latency communication (URLLC) for mission-critical applications, enhanced mobile broadband (eMBB) for high-speed internet, or massive IoT connectivity. Since each slice may have different control requirements, the ability to manage the Control and User Plane separately ensures that resources are allocated efficiently.

Open RAN

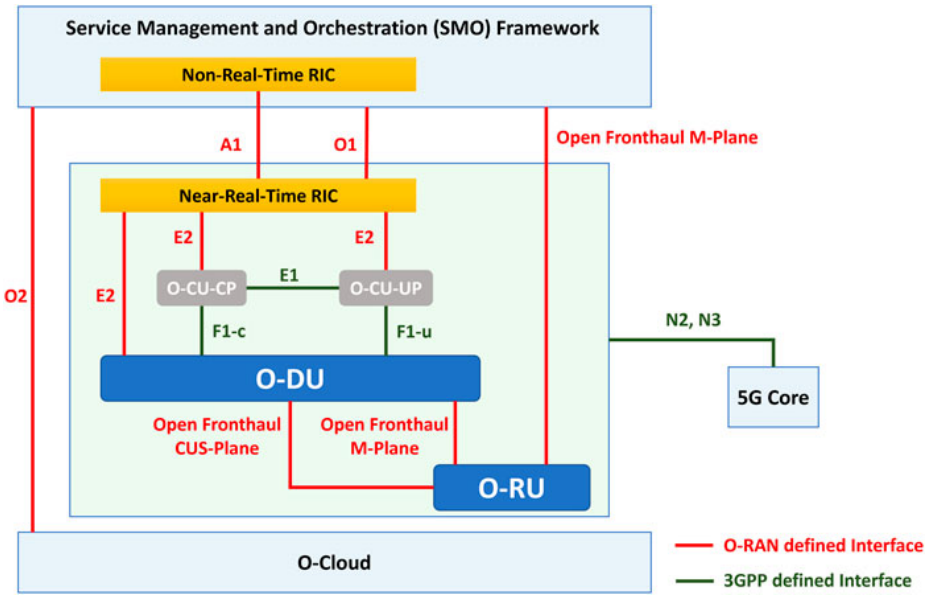


Figure 1.15: Open RAN Interfaces

The logical separation of network functions in Open RAN enables scalability, flexibility, and network slicing while aligning with the industry's broader shift towards open and interoperable architectures. Open RAN extends this modular design by introducing open interfaces and disaggregated components, allowing operators to deploy network elements from multiple vendors. This reduces dependency on proprietary solutions, fosters competition, and accelerates innovation. Standardizing interfaces between the Central Unit (CU), Distributed Unit (DU), and Radio Unit (RU) ensures interoperability, while leveraging cloud-native technologies, virtualization, and automation further enhances network efficiency.

Traditional RAN architectures rely on tightly integrated, vendor-specific solutions, limiting flexibility and making multi-vendor deployments complex. Open RAN addresses these challenges by defining standardized interfaces that facilitate interoperability and modularity. This allows operators to tailor network deployments based on performance, cost, and specific use-case requirements. Beyond these core interfaces, Open RAN introduces advanced control and optimization mechanisms through the RAN Intelligent Controller (RIC), which exists in two key forms: the Near-Real-Time (Near-RT) RIC and the Non-Real-Time (Non-RT) RIC. Figure 1.15 shows the most important interfaces and that connect the different components of the Open RAN architecture.

The Near-RT RIC operates at latencies ranging from 10 ms to 1s and dynamically optimizes radio resource management. It fine-tunes handovers, scheduling, and interference management based on real-time network conditions, ensuring efficient spectrum utilization

and enhanced user experience. The E2 interface connects the Near-RT RIC with the O-DU and O-CU, enabling precise, low-latency control.

The Non-RT RIC, on the other hand, operates over 1s and plays a strategic role in policy-driven optimizations. It aggregates network-wide analytics, predicts congestion, and informs energy-efficient resource allocation strategies. Residing within the Service Management and Orchestration (SMO) framework, the Non-RT RIC leverages AI/ML-driven models to automate network adjustments and optimize long-term performance. The A1 and O1 interfaces facilitate its interaction with the network, enabling high-level policy control and configuration management.

The synergy between these components is at the core of Open RAN's promise of flexibility and innovation. By exposing control loops to third-party developers, Open RAN enables the deployment of specialized applications, which are xApps for the Near-RT RIC and rApps for the Non-RT RIC, allowing operators to implement customized optimizations tailored to specific network scenarios. This programmability fosters an AI-driven, adaptive network where intelligent decision-making continuously refines operations based on evolving conditions.

Figure 1.15 provides a detailed view of the Open RAN architecture, illustrating the key functional components and their interconnections.

At the top, the SMO framework oversees the network, integrating both the Non-RT RIC and the Near-RT RIC. The Non-RT RIC operates at a higher level, using the A1 and O1 interfaces to manage policy-based optimizations and network-wide automation. Below it, the Near-RT RIC communicates with the O-CU and O-DU via the E2 interface, enabling real-time control and resource optimization.

The O-DU is responsible for signal processing and radio resource management, interfacing with both the O-CU and O-RU. The F1 interface connects the O-CU to the O-DU, with F1-c managing control plane traffic and F1-u handling user plane data. The O-DU also interacts with the O-RU through the Open Fronthaul interface, which consists of the Control and User Synchronization (CUS) Plane and the Management (M) Plane, ensuring precise coordination and radio control.

The O-RU serves as the physical radio hardware responsible for transmitting and receiving wireless signals. It connects to the 5G Core via the N2 and N3 interfaces, which are defined by 3GPP standards. Additionally, the entire system is integrated with an O-Cloud environment, enabling cloud-native, virtualized deployments.

The diagram distinguishes between O-RAN-defined interfaces (marked in red) and 3GPP-defined interfaces (marked in green). The O-RAN-defined interfaces highlight the open and disaggregated nature of the architecture, ensuring that equipment from different vendors can seamlessly interoperate within the network. The structured layering of RIC components, distributed processing, and standardized interfaces illustrate how Open RAN supports flexible, scalable, and intelligent network deployments.

By enabling network disaggregation, Open RAN transforms traditional RAN architectures into a more dynamic, multi-vendor ecosystem where innovation is accelerated, costs are optimized, and service delivery is significantly enhanced.

1.2.3 Mobile Core

The Mobile Core is a collection of different functions which are disaggregated in one ore more devices. In 4G it was called Evolved Packet Core (EPC) and in 5G it is also referred as 5G Core (5GC). This large component is usually placed near the edge of the network, acting as a bridge between the RAN and the IP-based Internet. Moreover, since it is a collection of isolated functions, every of its component can be placed in different regions, scaled or turned on and off according to different solutions, bringing much more flexibility to the hole infrastructure. We can divide the all the functional blocks in 3 groups, two of which are considered in the Control Plane and the third in the User Plane. The first group is the aggregation of the followings:

- AMF (Core Access and Mobility Management Function): manages the mobility-related aspects. Responsible for connection and reachability management, mobility management, access authentication and authorization, and location services.
- SMF (Session Management Function): manages each UE session, including IP address allocation, selection of associated UP function, control aspects of QoS, and control aspects of UP routing.
- PCF (Policy Control Function): manages the policy rules that other CP functions then enforce.
- UDM (Unified Data Management): manages user identity, including the generation of authentication credentials.
- AUSF (Authentication Server Function): essentially an authentication server.

These components have a counterpart in the previous 4G structure.

For better understand these functions, Figure 1.16, shows a sequential diagram of the UE registration to the network service. In particular, there are interactions between five key network entities: UE (User Equipment), RAN (Radio Access Network), AMF (Access and Mobility Management Function), AUSF (Authentication Server Function), and UDM (Unified Data Management). The process follows a structured flow, starting with the initial registration phase where the UE sends an InitialUEMessage to the RAN, which then forwards a Registration Request to the AMF. This marks the UE's attempt to connect to the network.

Following the initial registration, the authentication phase begins. The AMF sends an Authentication Request to the AUSF, which then communicates with the UDM to retrieve authentication vectors by sending a Get Auth Vector request. The UDM processes this request and responds with the necessary authentication data, which the AUSF forwards to the AMF. This ensures that the UE is authenticated before proceeding further in the registration process.

Once the authentication is complete, the security setup phase is initiated. The AMF sends a Security Mode Command to the UE, instructing it to establish encryption and

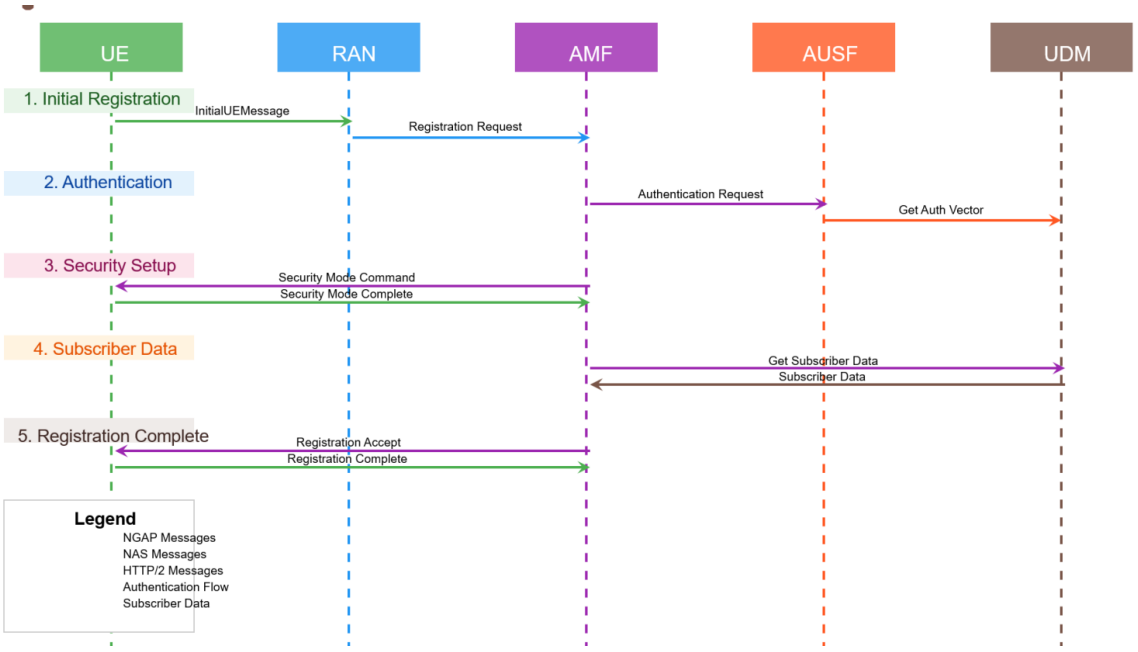


Figure 1.16: 5G UE registration diagram flow.

integrity protection mechanisms for secure communication. Upon receiving this command, the UE processes it and responds with a Security Mode Complete message, confirming that security parameters have been successfully applied.

With security established, the subscriber data phase follows. The AMF requests subscriber-related information from the UDM by sending a Get Subscriber Data request. The UDM processes this request and sends back the required subscriber data to the AMF, ensuring that the network has the necessary user profile and policy information to provide appropriate services.

The final stage in the sequence is the registration completion phase. The AMF sends a Registration Accept message to the UE, signaling that the registration process is nearing completion. The UE, upon receiving this message, sends a Registration Complete response back to the AMF, formally concluding the registration process and confirming that the UE is now successfully registered to the 5G network.

The second group also runs in the Control Plane (CP) but does not have a counterpart in the EPC:

- SDSF (Structured Data Storage Network Function): a "helper" service used to store structured data. Might be implemented by an "SQL Database" in a microservices-based system.
- UDSF (Unstructured Data Storage Network Function): a "helper" service used to store unstructured data. Might be implemented by a "Key/Value Store" in a microservices-

based system.

- NEF (Network Exposure Function): a means to expose select capabilities to third-party services, including translation between internal and external representations for data. Might be implemented by an "API Server" in a microservices-based system.
- NFR (NF Repository Function): a means to discover available services. Might be implemented by a "Discovery Service" in a microservices-based system.
- NSSF (Network Slicing Selector Function): a means to select a Network Slice to serve a given UE. Network slices are essentially a way to differentiate service given to different users. It is a key feature of 5G that we discuss in depth later in this tutorial.

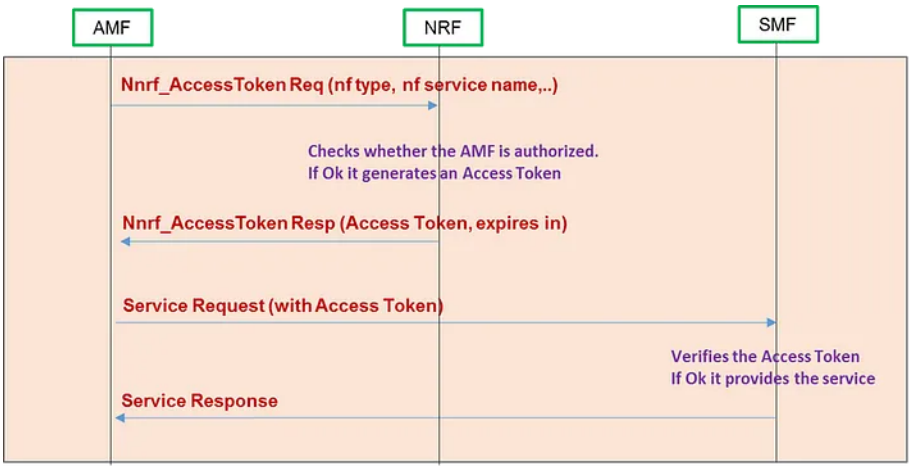


Figure 1.17: 5G UE registration diagram flow.

To show an example, Figure 1.17, depicts the interaction of the NRF with two other NF (the AMF and the SMF).

The process begins with the AMF sending an Nnrf_AccessTokenReq request to the NRF, which includes the network function (NF) type and NF service name. The NRF then checks whether the AMF is authorized to access the requested service. If the authorization is successful, the NRF generates an access token and responds to the AMF with Nnrf_AccessTokenResp, which contains the access token along with its expiration time.

Once the AMF obtains the access token, it proceeds by making a Service Request to the SMF, including the access token for authentication. The SMF then verifies the access token. If the token is valid, the SMF grants access to the requested service and responds to the AMF with a Service Response.

The third group is of the User Plane and it is mainly composed by the UPF component, which forwards traffic between RAN and the Internet. In addition to packet forwarding, it is also responsible for policy enforcement, lawful intercept, traffic usage reporting, and QoS policing. To better understand the use of the UPF, in Figure 1.18 the process begins

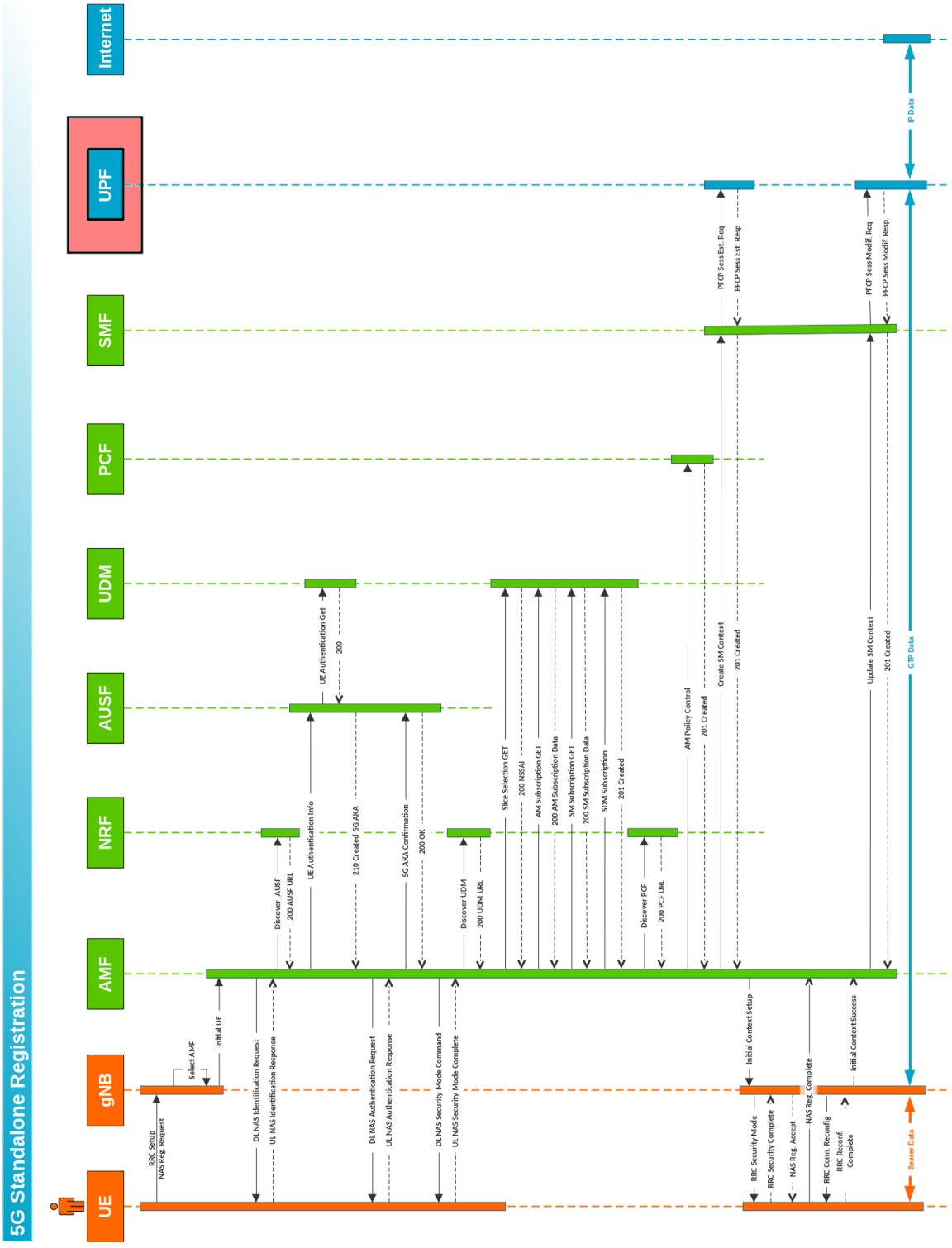


Figure 1.18: 5G SA registration of UE with UPF

with the UE initiating access to the 5G network, performing RRC (Radio Resource Control) setup and sending a NAS Registration Request to the gNB. The gNB forwards the registration request to the AMF, which selects itself as the serving AMF for the UE. The AMF then performs identity verification using NAS Identification Request and Response messages. Authentication follows, where the AMF discovers the AUSF via NRF, sends a UE authentication request to the AUSF, and receives authentication confirmation. The AMF also retrieves subscription information from UDM and network slice-related information.

Once authentication and security procedures are complete, the AMF performs policy control by communicating with the PCF. The SMF context is created, leading to the establishment of session management procedures for data connectivity. The AMF initiates NAS Registration Accept and NAS Security Mode procedures to complete the registration phase.

The final phase focuses on setting up the user plane for data transmission. The SMF updates session information, and the UPF (User Plane Function) is activated to handle bearer data transmission. The PCF facilitates session establishment and modification, ensuring proper policy enforcement for the user session. Once the registration and session setup are complete, the UE can send and receive IP data through the UPF, connecting to external internet services.

Chapter 2

Related Work

2.1 Energy Efficiency in 5G

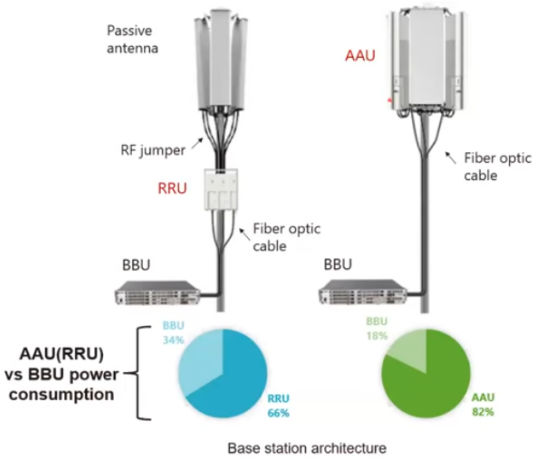


Figure 2.1: Comparison of Traditional RRU-Based and AAU-Based Base Station Architectures: Power Consumption and Design Differences

5G networks are raising concerns regarding energy consumption. Currently, the telecom industry accounts for 3% of global energy consumption, surpassing the aviation sector, which consumes 2%. Beyond sustainability concerns, telecom companies are also motivated to reduce energy costs. On average, electricity expenses make up 23% of an operator's total expenditure.

What's more alarming is that energy consumption in the telecom industry is expected to rise. This increase is driven by the need to deploy additional infrastructure to accommodate the growing number of users and connected devices. Furthermore, emerging services such as 4K video streaming, augmented reality (AR), and virtual reality (VR) demand significantly higher data rates, further exacerbating energy consumption.

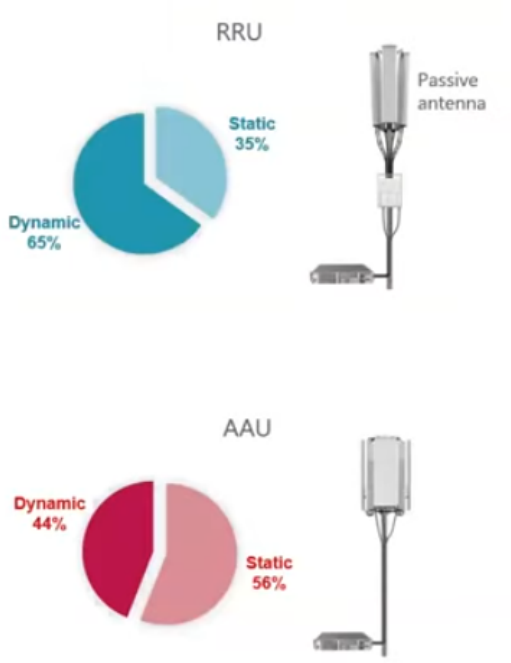


Figure 2.2: 5G Antenna's Static and Dynamic Energy Consumption

However, with the shift towards Open RAN (O-RAN) architectures, the problem of energy efficiency becomes even more complex. O-RAN introduces a highly disaggregated and virtualized architecture where components such as the Centralized Unit (CU) and Distributed Unit (DU) operate on general-purpose hardware. This virtualization increases the flexibility and scalability of network functions but also introduces higher energy consumption due to computational overheads and increased fronthaul/midhaul traffic [25]. Studies such as Mollahasani et al. [26] have demonstrated that the placement and relocation of DU workloads significantly impact energy efficiency in O-RAN environments.

Figure 2.1 illustrates the distribution of energy consumption across different network components. The RAN is responsible for 73% of total energy consumption, while the core network accounts for 13%, data centers consume 9%, and the remaining 5% is attributed to other infrastructure. For this reason, our work mainly focus on optimizing the energy consumption of the RAN part.

$$P_{RAN} = \sum P_{BS} + \sum P_{FH} + \sum P_{VBBU} \tag{2.1}$$

In particular, Equation 2.1, as presented in [27], defines the total power consumption of a Radio Access Network (RAN), which is divided into three main components:

- P_{BS}: The power consumption of the ith base station (gNB)
- P_{FH}: The power consumption of the jth fronthaul

- P_{VBBU}: The power consumption of the kth virtualized baseband unit

Delving deeper into the RAN, base stations (BS) can be categorized into two types:

- **Non-massive MIMO Base Stations (Remote Radio Units, RRU)**: These consume 66% of the total RAN energy.
- **Massive MIMO Base Stations (Active Antenna Units, AAU)**: These are even more energy-intensive, consuming 82% of the RAN's total energy.

This breakdown highlights that the radio unit (RU) is the most power-hungry component of a base station. The energy consumption of the RU can be further divided into:

- **Static energy**: Independent of traffic load, this is the energy required to keep the antenna active so that devices can always connect.
- **Dynamic energy**: Dependent on traffic demand, this fluctuates based on network usage.

Figure 2.2 illustrates this energy division. Most energy-saving techniques in the RU focus on reducing static energy consumption. In the following Section, we present a part of them.

2.2 RU energy saving techniques

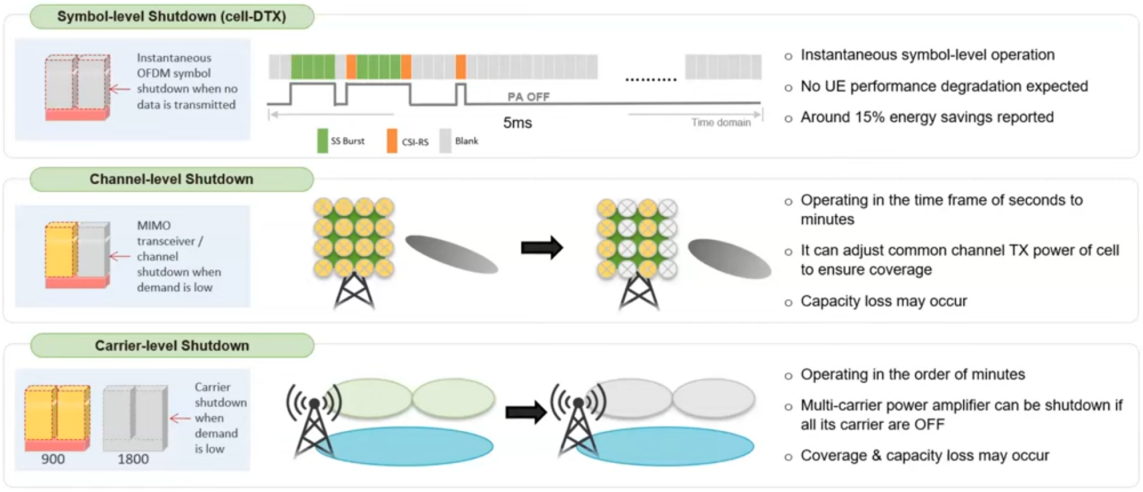


Figure 2.3: Energy-Efficient Shutdown Strategies for the RU part: Symbol-Level, Channel-Level, and Carrier-Level Power Saving Mechanisms

Looking into the methods that are available today to reduce the energy consumption of the RU, there are three most popular energy saving tecnhiques enabled in the industry by the 3GPP 5G NR standard. Figure 2.3 depicts them. They are divided in:

1. Symbol Level Shutdown: the base station automatically detects when a downlink symbol carry no data and the power amplifier (PA) is then switched off. Since the PA is one of the most energivourous component of the RU, it makes perfectly sense that if not used is therefore switched off.

2. Channel Level Shutdown: this works with massive MIMO base stations which can have up to 64 antennas and the idea is that during hours in which the traffic is low we may decide to switch off a part of the number of the antennas. This technique is very effective although by shutting down these pieces of hardware we may incur in capacity and coverage losses.

3. Carrier Level Shutdown: this works in the case that a base station operates at multiple frequencies or carriers and during low traffic hours we can decide to switch off some of these carriers. As the previous tecnique we may experience cover or capacity losses.

All of these methods target shutdown hardware at low traffic and specifically the static energy consumption. In [28] Hoffmann et al. propose an ES-rApp that uses Deep Reinforcement Learning (DRL) to perform RCR (RF Channel Reconfiguration), also known as *antenna selection*. Since RCR selectively deactivates some RF channels (and thus antennas) during low-traffic periods, it is effectively a form of Channel Level Shutdown. However, their work extend the conventional Channel Level Shutdown methods using intelligent, machine-learning-based techniques, making it more dynamic and adaptive. Concretely, instead of

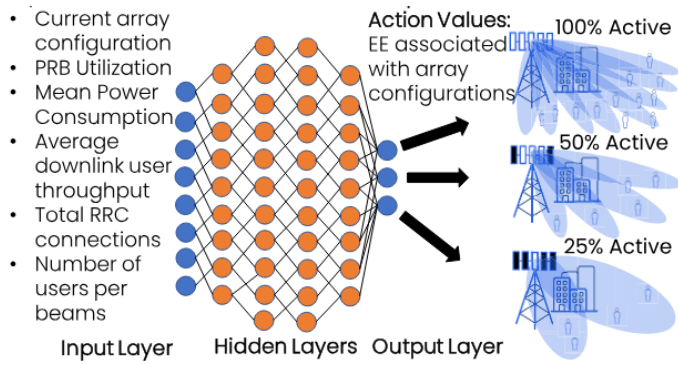


Figure 2.4: Input and output features of energy saving model used in [28]. Figure from [28].

keeping all RF channels (i.e., active antenna elements) powered on at all times, the system dynamically adjusts the number of active RF channels based on real-time network traffic conditions. The system continuously monitors Key Performance Indicators (KPIs) from the base station, such as power consumption, user throughput, traffic load (PRB utilization), distribution of users across beams and these KPIs are collected through Open RAN interfaces (O1, A1, and E2). The ES-rApp (Energy Saving rApp), running in the Non-Real Time RAN Intelligent Controller (Non-RT RIC), uses DRL to decide how many RF channels (antenna elements) to turn off. Once an action is chosen, the system reconfigures the

antenna array via Open RAN interfaces and it then measures the network performance after the change (e.g., checking if user QoS is maintained). In the end, the DRL model updates itself based on the new observations, refining its decision-making process over time.

The survey by Wu et al. [29] provides a comprehensive analysis of sleep mode techniques, highlighting how selectively turning off lightly loaded base stations during low-traffic periods can lead to significant energy savings. The study also emphasizes that traditional network architectures have been designed with peak load provisioning in mind, leading to substantial energy waste during off-peak hours. By dynamically adjusting base station activity, networks can balance coverage and energy efficiency more effectively.

All of these techniques are algorithmically very complex and they base themselves on conditions that must be verified quite frequently and accurately. Such conditions, or KPI, might be the number of UEs connected to the antenna, the traffic load, weather if an handover from a BS to another is possible and many others. Moreover, as shown in [28], there is the need to evaluate or calibrate these energy saving models by measuring the impact on the energy consumption. In the next section we present some of the most famous energy measuring models.

2.2.1 EARTH Model of RU

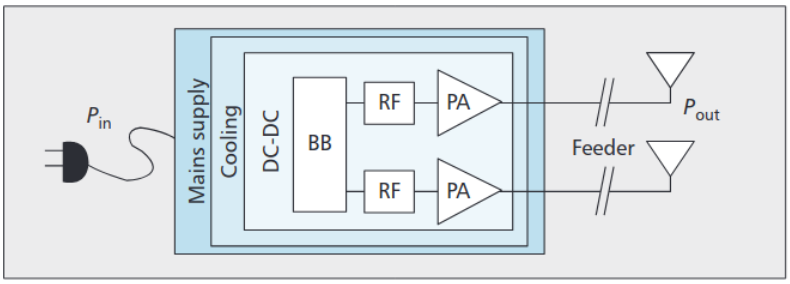


Figure 2.5: Block diagram of a base station transceiver. The power amplifier (PA) is responsible for amplifying the transmitted signal to a power level suitable for wireless transmission, ensuring that the signal can propagate over long distances while maintaining sufficient quality. The RF module encompasses the components necessary for processing radio frequency signals, including mixers, filters, and oscillators, which are crucial for modulating and demodulating signals in wireless communication. The baseband unit (BB) handles signal processing tasks such as encoding, decoding, modulation, and demodulation, as well as network control functions, making it a critical component for data transmission and reception. The power supply ensures that all components of the base station receive the required electrical energy, converting and regulating power as needed for efficient operation. The active cooling system dissipates heat generated by the various electronic components, preventing overheating and ensuring stable performance, especially in high-power and high-density deployments. The connection to the electrical grid provides the base station with a stable and continuous source of electricity, ensuring uninterrupted operation and reliability in delivering communication services. Figure from [30].

The EARTH model, introduced in [30], is a widely used power consumption model for base stations (BS). As depicted in Figure 2.5, a typical BS consists of multiple transceivers, each serving a single antenna element. The total power consumption of a BS is primarily influenced by its power amplifier (PA), RF module, baseband unit (BBU or BB), power supply, active cooling system, and connection to the electrical grid. The model expresses BS power consumption as:

$$P_{BS} = N_{TRX} \times \frac{P_{out}}{\eta_{PA}(1-\sigma_{feed})} + \frac{P_{BB}+P_{RF}}{(1-\sigma_{co})(1-\sigma_{DC})(1-\sigma_{MS})}$$

where N_{TRX} is the number of transceivers, P_{out} is the transmitted power, and η_{PA} is PA efficiency. The terms $\sigma_{feed}, \sigma_{co}, \sigma_{DC}$, and σ_{MS} account for various power losses in the system.

A key observation from [30] is that while PA power consumption scales with the BS load, other components remain relatively constant. This results in a nearly linear relationship between total BS power consumption and transmitted power:

$$P = \begin{cases} N_{TRX} \times P_0 + \xi_P P_{out}, & 0 < P_{out} < P_{max} \\ N_{TRX} \times P_{sleep}, & P_{out} = 0 \end{cases}$$

where P_0 is the fixed power consumption, ξ_P represents the load-dependent power consumption slope, and P_{sleep} is the power used in sleep mode when no traffic is being transmitted.

For O-RAN, the radio unit (RU) is a separate physical node with power consumption mainly associated with RF functionalities and power amplification [31]. The power model of the RU is given by:

$$P_{RU} = \begin{cases} P_{RF} + \frac{P_{out}}{\eta}, & 0 < P_{out} < P_{max} \\ P_{sleep}, & P_{out} = 0 \end{cases}$$

where η represents PA drain efficiency, P_{RF} accounts for RF circuit power consumption, and P_{sleep} is the power used in sleep mode.

2.2.2 Carrier Aggregation Power Consumption Model of RU

Carrier Aggregation (CA) is a feature introduced in LTE-Advanced (LTE-A) that enables the use of multiple Carrier Components (CCs) to increase data rates. Initially, LTE-A supported up to 5 CCs with a maximum of 20 MHz each, while 5G NR extends this to 16 CCs with a total bandwidth of 1 GHz. In CA-enabled RUs, power consumption scales with the number of active CCs, modeled as:

$$P_{RU} = \sum_{j=1}^{N_{CC}} \left(\frac{P_{out,j}}{\eta} + B_j P_{CA,CP_j} \right) + P_{CA,iCP}$$

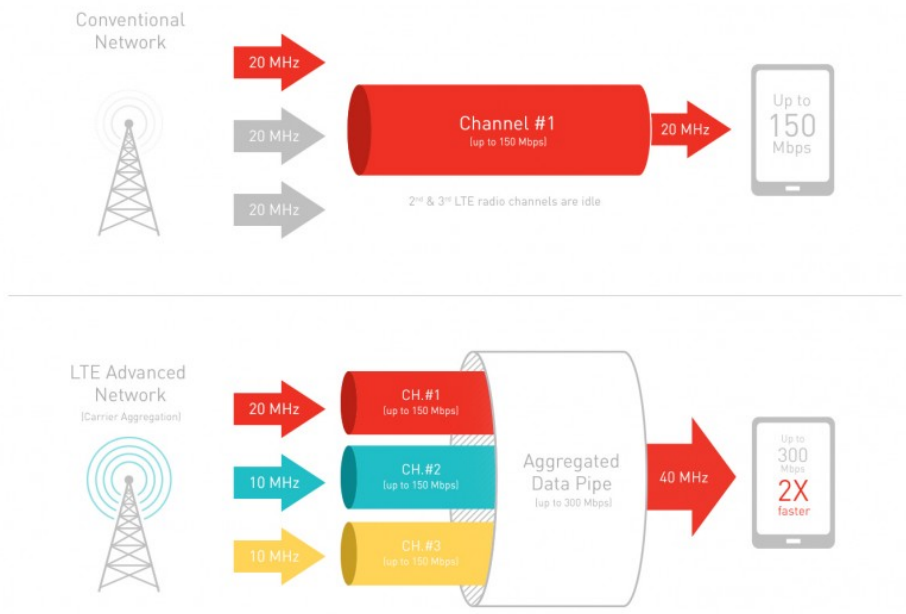


Figure 2.6: Carrier Aggregation; in the LTE-Advanced Network the UE can be allocated DL and UL resources on the aggregated resource consisting of two or more Component Carriers (CC).

where $P_{out,j}$ is the transmitted power for CC j, B_j is its bandwidth, and P_{CA,CP_j} represents the variable power consumption that scales with the number of active CCs. The term $P_{CA,iCP}$ accounts for static circuit power consumption in CA systems.

2.3 DU or CU energy saving techniques

While RU energy-saving techniques focus on hardware-level optimizations, such as shutting down power amplifiers, antennas, or carriers during low traffic periods, CU and DU energy efficiency strategies rely more on intelligent workload management, dynamic scaling, and virtualization.

In disaggregated 5G-RANs, BS ON-OFF switching becomes CU-DU-RU ON-OFF management, with the most common approaches limited to DU-RU sleep mode schemes as seen in [32], since the CU is typically responsible for higher-layer control functions such as ensuring control-plane functionality and transmitting data plane packets. Nowadays, to minimize power consumption at the CU level, the most common approach involves optimizing the placement of CU functionalities across edge and regional clouds, as demonstrated in [33] and [34], leveraging techniques such as Integer Linear Programming (ILP), Mixed Integer Linear Programming (MILP), and heuristic algorithms to reduce active nodes while maintaining latency and QoS requirements.

In particular, in [33], the authors propose a MILP model to determine the optimal allocation of Distributed Units (DUs) and Centralized Units (CUs) within the metro-access

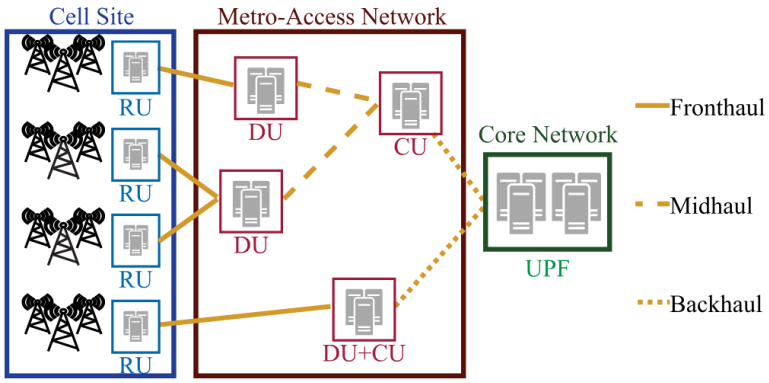


Figure 2.7: Example of placement of DUs and CUs over the network. Figure from [33].

network. Their approach considers latency, network capacity, and functional split constraints.

Since solving MILP is computationally expensive, the authors also develop a heuristic algorithm that provides near-optimal results with significantly reduced complexity. They validate their approach using both small-scale and realistic large-scale network topologies, demonstrating that their method can achieve up to 50% power savings compared to fully distributed (D-RAN) scenarios while ensuring compliance with latency requirements.

In [34], the authors propose an ILP model to optimize the dynamic placement of these functions across edge and regional O-clouds, aiming to maximize user admittance while considering latency and server capacity constraints.

To reduce the computational complexity of solving ILP, the authors introduce a Recurrent Neural Network (RNN)-based heuristic model, which effectively replicates the ILP model's decision-making process with significantly lower execution time. Their approach achieves up to 10% improvement in user admittance ratio compared to baseline placement strategies while also reducing deployment costs and improving overall network throughput. Their findings highlight the advantages of dynamic and flexible O-CU/O-DU placement for enhancing O-RAN efficiency.

Mollahasani et al. [26] propose an Actor-Critic-based approach for dynamically selecting DU workloads to optimize energy efficiency. Their Soft Actor-Critic Energy-Aware Dynamic DU Selection algorithm (SA2C-EADDUS) integrates reinforcement learning to intelligently reallocate network functions while considering latency constraints. Their findings demonstrate that intelligent DU selection can improve energy efficiency by up to 50% compared to conventional heuristic-based approaches.

Our work differentiates from all the previously cited approaches by proposing a simple algorithm that minimizes computational overhead, focusing on identifying the key parameters that optimize energy savings rather than managing and orchestrating them.

It is important to distinguish between DU/CU placement and DU/CU scaling. Placement

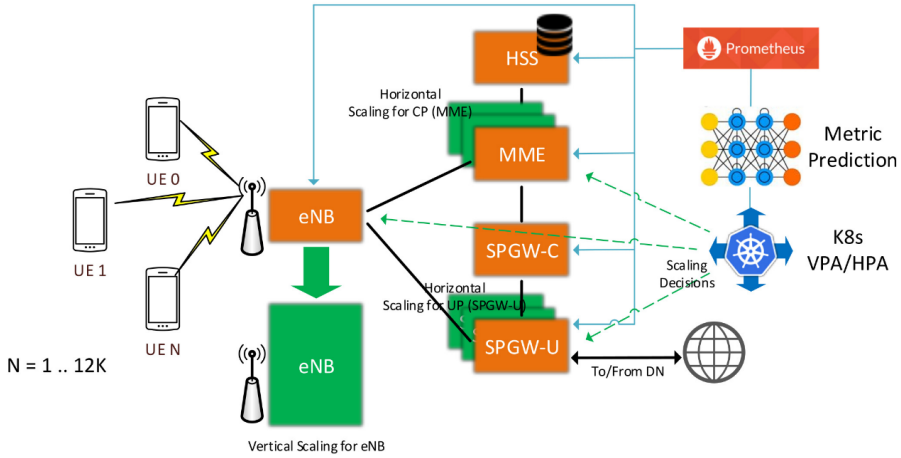


Figure 2.8: System architecture from the work of [35]. This figure shows the joint scaling techniques: vertical scaling for the eNB and horizontal scaling for the virtual components.

refers to the strategic deployment of DU and CU components across edge and regional cloud environments, optimizing network performance, latency, and energy efficiency by determining where these units should be located. Scaling, on the other hand, deals with dynamically adjusting the number of active DU and CU instances based on real-time traffic demand. This means, for example, that in a setup that leverages Kubernetes as an orchestrator, if every DU and CU is mapped into a pod, than the scaling of them follows the scaling of a pod instance.

In our work, we classify our approach as scaling rather than placement, even though it involves increasing or decreasing the number of CU instances, each deployed on a different physical node. This distinction arises because our focus is on adjusting the number of active instances rather than determining their specific placement within the network.

Numerous strategies have been explored to optimize energy consumption in 5G networks, particularly in VNFs using scaling approaches. For example, in [35], the authors examined vertical scaling of the RAN and horizontal scaling of control and user plane functions (CN components). As demonstrated in the paper, the vertical scaling approach was beneficial for accommodating incoming demand, alleviating potential blockages caused by multiple UEs requesting network services concurrently. On the other hand, horizontal scaling of the Core Network components was found to be helpful for achieving significant energy gains. In addition, the authors used machine learning forecasting to develop a scheduler that proactively scales components based on traffic metric predictions.

In stark contrast with them, our work focuses on horizontal scaling of RAN components rather than vertical scaling.

One key difference in VNF scaling strategies is the distinction between proactive and reactive

scaling.

Proactive scaling leverages predictive models to anticipate traffic demand and adjust resources accordingly before congestion occurs, whereas reactive scaling responds to real-time network conditions, adjusting resources dynamically based on observed traffic fluctuations. Proactive scaling techniques often rely on machine learning (ML)-based forecasting models, as explored in [36], where deep learning algorithms such as Transformer-based models are used to predict traffic loads and optimize DU and CU activation patterns. These methods can achieve significant energy savings by preemptively shutting down underutilized units while ensuring that quality of service (QoS) requirements are met.

Reactive scaling, on the other hand, typically operates based on threshold-based policies, where resources are dynamically allocated or deactivated based on network congestion levels and latency constraints. The work in [37] introduces a reinforcement learning (RL)-based approach that optimizes reactive scaling by continuously learning from network state transitions, ensuring optimal CU and DU placement with minimal computational overhead. Compared to traditional rule-based scaling, RL-based techniques can adapt more effectively to dynamic traffic conditions, particularly in ultra-dense 5G deployments.

Our work follows this technique, although we consider proactive scaling a good alternative that we would like to further explore in future works.

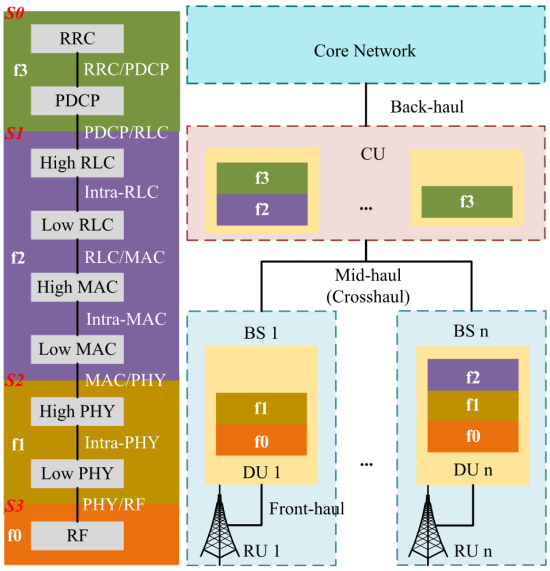


Figure 2.9: Considered functional split in the research work of [38].

Another approach involves dynamic functional split selection in 5G Cloud Radio Access Networks (C-RAN). Researchers in [39] have developed an optimization model that maximizes centralization by adjusting the functional split between Distributed Units (DUs) and Centralized Units (CUs) based on traffic heterogeneity and midhaul bandwidth constraints. This dynamic selection allows for more DUs to operate on higher-layer split options, en-

hancing energy efficiency without compromising performance. The work in [38] further refines this concept by integrating functional split selection with base station (BS) sleeping strategies and user association policies. The authors propose a Mixed Integer Nonlinear Programming (MINLP) framework that jointly optimizes CU-DU assignment, BS activation states, and user association, all while minimizing overall network expenditures. Their results demonstrate that adapting the functional split dynamically, in conjunction with intelligent BS sleeping policies, can lead to significant energy savings without degrading network performance.

A key advantage of this approach is its ability to balance network centralization and midhaul constraints. By allowing more DUs to operate with higher-layer splits when midhaul resources are limited, the system achieves a trade-off between energy efficiency and service quality. Furthermore, the study highlights how user migration due to BS sleeping must be carefully managed, as it affects both functional split decisions and routing paths.

Our work does not incorporate this level of refined analysis; however, we concentrate on adjusting the number of active CU and DU instances based on real-time network demand, ensuring a balance between performance and energy efficiency without the added complexity of cross-layer orchestration.

As explored in [40], agile DU-CU deployment strategies can improve energy efficiency but, if not managed properly, may lead to excessive power consumption due to unnecessary activations of network elements. Frequent activation and deactivation of DUs and CUs introduce overheads related to boot-up sequences, resource provisioning, and software initialization, all of which momentarily spike power consumption before stabilizing. Without proper load balancing, some units may remain underutilized while others operate near peak capacity, leading to fragmented resource usage and wasted energy. In addition, activating a single DU may trigger multiple optical network elements, including lightpaths, transponders, and switching components, resulting in excessive power draw that negates the intended energy savings. The need to balance energy efficiency with service requirements is particularly critical in low-latency applications like URLLC, where aggressive DU-CU deactivation could introduce unacceptable delays. To address these inefficiencies, advanced machine learning models are being explored to predict traffic variations and optimize DU-CU activation patterns, while joint radio-optical network coordination is being investigated to reduce redundant optical component activations.

In particular, paper [41] focuses on network slicing in 5G, specifically its role in enhancing energy efficiency. It explores how slicing, which allows for the virtualization of network resources, can be leveraged to meet the diverse demands of 5G services like eMBB, URLLC, and mMTC. The energy consumption patterns of network slices are intricately tied to the specific service demands they aim to fulfill. For example, eMBB slices, which cater to high-bandwidth applications like streaming, generally consume more energy due to the constant high data throughput required. URLLC slices, designed for ultra-low-latency applications, require energy-intensive resources to ensure minimal delay, often causing more frequent activations of computational and network elements. Meanwhile, mMTC slices, focused

on efficient low-power IoT device connectivity, are optimized for energy savings through reduced transmission frequencies and lower resource demands. Efficient network slicing, therefore, requires careful management to balance energy consumption while meeting the performance needs of each slice. Table 2.10 shows a comparison between the different types of slices

SST ID Type	SST ID Value	Characteristic	Use Case	NS Energy Demand/Capacity/
eMBB	1	Enhanced Mobile Broadband Connectivity (eMBB) slice optimized for managing 5G enhanced mobile broadband services	Entertainment, gaming, virtual and augmented reality, video streaming, fixed wireless access	High
URLLC	2	Slice designed for ensuring ultra-reliable low-latency communication (URLLC) (e.g., 1 ms)	Public safety, remote medicine, emergency response, smart grid	High
MIoT	3	Slice tailored for managing extensive (Massive) Internet of Things (MIoT) applications	Sensor networks, smart telemetry, smart homes, Internet of Everything (IoE)	Low
V2X	4	Slice crafted for handling Vehicle-to-Everything (V2X) services	Autonomous driving, driver and pedestrian safety management, traffic management, road infrastructure management	Very high
HMTC	5	Slice suitable for facilitating high-performance machine-type communications (HMTC)	Industrial IoT, smart factories, smart cities	Low
HDLLC	6	Slice engineered for managing high-data-rate and low-latency communications (HDLLC)	Extended reality and multi-modality services (video, audio, ambient-sensor and haptic data)	Very high

Figure 2.10: Standardized SST Slice Selection Type Identificator (SST ID) values for different use case examples and estimated energy demands. Table from [41].

2.3.1 Power Consumption Models of DU and CU

The power consumption of Distributed Units (DU) and Centralized Units (CU) is influenced by CPU load. The model in [42] defines power consumption based on the utilization of processing resources. For an Edge Processing Module (EPM), which hosts DUs, the energy consumption in a given time slot t is:

$$E_{epm,j}(t) = \left(I(l_{epm,j} > 0) P_{epm} + P'_{epm} \frac{l_{epm,j}}{C_{epm}} \right) T$$

where P_{epm} represents fixed power consumption, P'_{epm} is the load-dependent power, and $l_{epm,j}$ is the CPU load. The same model applies to Central Processing Modules (CPM) hosting CUs:

$$E_{cpm,k}(t) = \left(I(l_{cpm,k} > 0) P_{cpm} + P'_{cpm} \frac{l_{cpm,k}}{C_{cpm}} \right) T$$

Since O-RAN relies on virtualized infrastructure, DU/CU power consumption is primarily dictated by CPU usage rather than traditional BS models. The power model for

DU/CU is:

$$P^t_{DU/CU} = N_c(P_{DU/CU,min} + \Delta P_{DU/CU}\delta_c s^\beta)$$

where N_c is the number of active CPU cores, $P_{DU/CU,min}$ and $P_{DU/CU,max}$ define minimum and maximum power per core, δ_c represents CPU load percentage, s is CPU speed, and β is an exponential coefficient. The CPU load percentage is given by:

$$\delta_c = \frac{Q(r)}{N_{cs}} = c_0 + \frac{kr}{N_{cs}}$$

where $Q(r)$ is the executed instructions per time unit and N_{cs} is the total available instructions per time unit. By combining these equations, the overall DU/CU power consumption model is:

$$P^t_{DU/CU} = N_c P_{DU/CU,min} + \Delta P_{DU/CU}\left(c_0 s^{\beta-1} + krs^{\beta-1}\right)$$

An alternative model in [43] defines energy consumption in terms of activating servers and instantiating O-RAN applications:

$$E_s(x_s, y_s) = x_s E_{base,s} + \sum_{a\in A}\sum_{r\in R} y_{r,a,s} e_{a,s}$$

where x_s represents server activation, $E_{base,s}$ is the fixed energy consumption of an active server, and y_s indicates the load of the server based on the deployed applications. The parameter $e_{a,s}$ captures the energy consumption of an application, which scales with server load.

Energy efficiency in O-RAN is still an emerging research area. Most current studies rely on models originally designed for Cloud RAN (C-RAN) and Virtualized RAN (v-RAN). However, these models do not fully account for the unique characteristics of O-RAN, such as its highly dynamic resource allocation and the varying energy demands of its disaggregated components.

Another debate is concerning metric choice, we highlight that many works [44][45][33][46][47] focus on counting *power* (in Watts) rather than *energy* (in Joules) in their power-saving optimization strategies. Although these works primarily aim to reduce the overall system's power consumption, a recent study [48] highlights energy as a more meaningful metric. In fact, energy takes into account the power and the time span involved in managing dynamically changing BS functions. For those reason, in our paper we align with this assumption, though we also plan to explore the power-versus-energy debate in future work.

This distinction between power and energy metrics underscores the need for precise evaluation methods, as optimization strategies must align with the dynamic and temporal nature of BS operations to achieve meaningful energy savings. There exist some novel work on methodologies for testing and measuring the essential parameters for energy saving in

O-RAN [49]. Our work identifies key parameters offering tuning opportunities for energy management and optimization.

Finally, most of the literature has focused on orchestrating the BS function's resource allocation, validating their results only in simulated environments [42] [50]. For instance, Joda et al. [51] developed a strategy for the placement of CU-DU network functions in regional and O-Cloud nodes, while simultaneously addressing user association with RUs. Their simulations of various user mobility scenarios indicated that their strategy provides a good balance between cost minimization and performance.
However, while these two works used "CPU cycle per second" or "GOPS" as *cost metric*, we empirically measure the energy consumption in Joules from a real-world test-bed by running actual workload experiments.

We believe that our work extends the current state of the art by providing novel, measurable insights that can be verified whose reproducibility is guaranteed by the open-source nature of our framework (see Chapter 4 for more details).

In conclusion, the central focus of our work was to identify the key parameters that could be tuned to optimize energy consumption within the O-RAN 5G framework. We approached this by asking two critical questions: first, "What are the parameters that we can tune for energy optimization?" and second, "How can resources be optimally allocated within the O-RAN 5G framework to minimize energy consumption?" Thus, the novelty of our work lies in its focus on resource allocation based on the key parameters we identified, ensuring that energy optimization is achieved without the complexity of cross-layer management or predictive modeling. Our work provides a fresh perspective on energy-efficient O-RAN management by prioritizing simplicity, real-time adjustments, and tangible, measurable outcomes.

Chapter 3

System Architecture and Experimental Setup

3.1 A Logical System Overview

This section presents the architecture of our proposed 5G system, which includes a nonRT-RIC controller located at the SMO entity and is responsible for horizontally scaling the CU instances in a 5G O-RAN network in order to minimize energy consumption while dynamically adapting to network demands.

The proposed 5G system architecture is based on a foundational 5G CN deployment, having only the necessary key components such as the Network Repository Function (NRF), User Data Repository (UDR), Unified Data Management (UDM), Authentication Server Function (AUSF), Access and Mobility Management Function (AMF), Session Management Function (SMF), and a User Plane Function (UPF). All the components of the 5GCN were placed at the O-CLOUD in order to be in a controlled environment from the SMO. We focused on the basic architecture of the 5GCN as it effectively meets the connectivity and service requirements of the UEs while aligning with our primary objectives. For the 5G-RAN, we follow the typical O-RAN architecture that supports only the existence of the F1 interface that implements the 3GPP Option 2 split, enabling the functional separation of the CU and DU components of a 5G network as mentioned in Section . In our setup the RU is co-located with the DU. Our proposed 5G system is O-RAN compliant since we support the existence of the SMO platform responsible for the RAN and O-Cloud management. The SMO is equipped with an Energy Collector Module (ECM) executed as an rApp, application operating in nonRT-RIC, which aggregates energy data from different components of the disaggregated RAN. Each component of the disaggregated RAN (CUs, DUs) is equipped with Energy Exporter Modules (EEM) that monitor and report its energy consumption to the nonRT-RIC. According to O-RAN, O1 is the interface with which the nonRT-RIC communicates with all O-RAN Managed Elements (MEs). All the aforementioned components discussed are shown in Figure 3.1.

In our real-world testbed environment, we choose to implement a fully controlled scenario

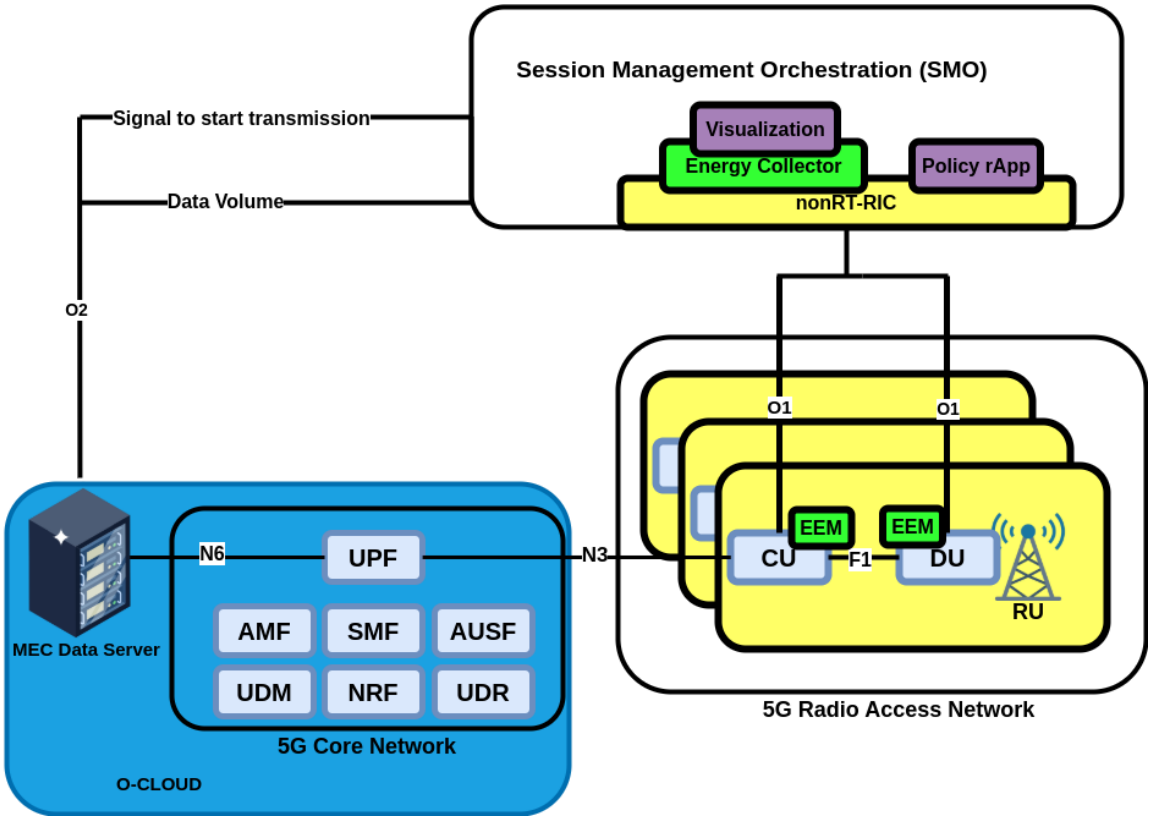


Figure 3.1: Logical Architecture

with respect to the data volume requests of the UEs to validate the proposed scaling policy. In this scenario, the rApp running on the nonRT-RIC is informed by the Multi-access Edge Computing (MEC) server about the demanded traffic volume for the UEs and signals it to start the transmission. The MEC server is located at the O-Cloud, and the communication occurs via the O2 interface. Since the rApp already knows the data volumes that will be sent to the UEs, it can decide whether the CU component should be scaled out or in to minimize the energy consumption of the O-RAN. These predefined conditions ensure the feasibility of the proposed scaling policy, as the focus of this work is on validating the policy itself, under the assumption that the traffic volume and duration are already known.

Once the MEC server intends to transmit data to the UEs, it first advertises the expected data volume to the SMO. The rApp in the SMO processes this information and determines the number of CU instances needed to handle the transmission efficiently. It then allocates the required CU resources and signals the MEC to begin transmission. The downlink data then flows from the MEC to the UE through the standard 5G network path: first passing through the UPF, then the CU, followed by the DU, RU, and finally reaching the UE. Meanwhile, the energy consumption of the CU and DU is directly measured at these components and reported to the SMO via the O1 interface for further monitoring and optimization.

3.2 Experimental Setup

Figure 3.2: Countries that participate to the Slices RI project.

In this section, we describe the use cases and our setup for the experiments that we performed. To conduct our experiments, we utilized the OneLab testbed located at the Sorbonne University in Paris, part of the French node of the SLICES-RI infrastructure [52].

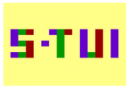
SLICES-RI is a platform that enables researchers worldwide to deploy their experiments on a distributed testbed, with reproducibility being one of its key goals. To promote the reproducibility of our work, aligning with the SLICES-RI objective, the code for all the experiments conducted in this paper is available at the following URL: `https://github.com/RootLeo00/Dynamic-Scaling-Policy-Energy-5G-ORAN`.

To deploy an O-RAN 5G network we utilize the OpenAirInterface (OAI) [53] implementation of the 5GCN and 5G-RAN. OAI is one of the most popular, continuously integrated, and maintained open-source projects for 5G VNF implementation. These VNFs are containerized using Docker and orchestrated with Kubernetes. Specifically, four nodes equipped with Intel(R) Core(TM) i7-10700 CPU @ 2.90GHz CPUs, are used to deploy the different VNFs; in each node we deploy 1 DU-RU and 1 UE and we randomly place from 1 to 4 CU distributed across the nodes. The link among the RUs and UEs is emulated via the RF-simulator from the OAI implementation. We also utilize one extra node with the same CPU model to deploy our 5GCN.

To acquire the power consumption metrics of each VNF, we had to decide between a range of open source solutions. Table 3.3, shows the different tools that have been analyzed

Software based Power Measurement Tools

- s-tui
- kwcollect
- PowerAPI
- Kepler
- **Scaphandre**



Feature	Scaphandre	PowerAPI	kwcollect	s-tui	Kepler
Documentation	Excellent, detailed guides	Moderate	Limited, Grid5000-specific	Basic	Moderate, includes installation guides
K8s Integration	Easy, sidecar container support	Possible, manual setup	Possible, complex setup	Minimal	Strong, Kubernetes-native
Metric Granularity	per-process/ per-pod/ **per VM**/ per-node	Per-process	Node-level only	CPU-only	per-pod/ per-node
Scalability	Great for cloud/edge	Moderate	Best for centralized setups	Local use only	Cloud-native, ML estimation

Figure 3.3: A comparison among various power consumption measuring tools.

for this objective.

In terms of documentation, Scaphandre [54] stands out with excellent and detailed guides, while PowerAPI and Kepler provide moderate documentation, and kwcollect and s-tui offer limited and basic resources, respectively. Kubernetes integration varies across tools, with Kepler offering the strongest support as a Kubernetes-native solution, while Scaphandre provides easy sidecar container integration. PowerAPI and kwcollect require more complex or manual setup, and s-tui has minimal integration.

Metric granularity is another critical differentiator. Scaphandre supports per-process, per-pod, per-VM, and per-baremetal-node measurements, making it the most versatile. Kepler follows closely, offering per-pod and per-node granularity. PowerAPI is limited to per-process, while kwcollect only provides node-level data, and s-tui focuses solely on CPU metrics.

Scalability also varies, with Kepler being optimized for cloud-native environments with machine learning-based power estimation, and Scaphandre being well-suited for cloud and edge computing. PowerAPI offers moderate scalability, while kwcollect is best suited for centralized setups, and s-tui is restricted to local use only.

Scaphandre stands out as the only tool capable of retrieving power consumption data with virtual machine (VM) granularity. This unique capability makes it an essential choice for environments that require fine-grained power measurement at the VM level, particularly in cloud-based and virtualized infrastructures where accurate energy tracking is crucial. Moreover, it is an open-source tool that has also been adopted in several recent research works, such as those in [55] [56]. These are the reasons of why we have picked it.

Scaphandre had to be integrated in the already existing SLICES-RI testbed. It leverages the RAPL (Running Average Power Limit) sensor and the powercap kernel module of the Linux OS to measure the instant power consumption of the node over a specific duration. It then maps this global energy measure to individual processes by scraping information from the /proc directory of the Linux OS. After obtaining the power consumption of a single process, we sum up those belonging to the same container, creating a coarser metric, the "power consumption of a container." These metrics are then available via the Prometheus exporter.

Scaphandre currently provides power consumption measurements at the pod level, allowing users to monitor energy usage within Kubernetes environments. However, while Scaphandre collects power metrics per container, the labeling system is still evolving. At this stage, only the 'container_runtime' label is available, with containerd as the sole possible value. Additionally, Scaphandre retrieves the 'container_id' from the '/proc/PID/cgroup' file, which differs from the "CONTAINER ID" obtained through the standard Docker command ("docker container ls").

Due to this limitation, we had to implement an ETL (Extract, Transform, Load) process to properly associate Scaphandre's metrics with the actual Kubernetes pod names. This method involves periodically retrieving pod metadata using 'kubectl', extracting truncated UIDs from pod information, and mapping them to corresponding power consumption data stored in Prometheus. The system then exposes these refined metrics through a Prometheus-compatible exporter, ensuring accurate power monitoring at the pod level. This approach enables a clearer correlation between energy usage and specific Kubernetes workloads, improving observability and energy efficiency analysis.

Scaphandre offers a variety of exporters to facilitate the collection and dissemination of energy consumption metrics, catering to different monitoring and data analysis needs. Below

is an overview of the available exporters. Among all of them, in order to be compliant with our logical architecture, we rely on the Prometheus Exporter.

The Prometheus exporter exposes power consumption metrics through an HTTP endpoint ('/metrics' by default) in a format compatible with Prometheus. This allows for seamless integration with Prometheus-based monitoring systems. To launch the Prometheus exporter using the default 'powercap_rapl' sensor, the command "scaphandre prometheus" has to be executed.

The data flow from the RAPL sensor to the Prometheus server is shown in Figure 3.4.

RAPL, or Running Average Power Limit, is a power management feature present in modern Intel and AMD x86 CPUs, introduced after 2012. It allows users to monitor and, in some cases, set power consumption limits for different components within a processor. The values provided by RAPL are not always direct measurements but often estimations derived from power models based on hardware behavior.

Figure 3.5 visually represents the power domains that RAPL can track within a multi-package CPU architecture. Each package contains multiple processing cores, a last-level cache, an embedded GPU, and system agent components. The PKG domain encompasses the total power consumption of the package, including cores and uncore elements. The PP0 domain (shown in blue) accounts for the energy used by the CPU cores themselves, while the PP1 domain (marked in red) represents power drawn by the integrated graphics. DRAM domains (in green) track the energy used by memory modules, and PSys, though not well-documented, appears to estimate power usage beyond just the CPU and GPU, possibly including other motherboard-connected components.

The diagram highlights two processor packages, each managing its own power and memory interactions. External components such as PCH (Platform Controller Hub) and eDRAM (embedded DRAM) also play a role in overall system power consumption. The interconnections between the CPU packages and DRAM modules suggest a shared memory architecture, where power management extends beyond just computation cores.

While RAPL provides useful insights into power consumption, its accuracy depends on the CPU model and implementation. Some values, such as DRAM energy consumption, may not always be included in the package domain, varying across different architectures. Understanding RAPL requires careful experimentation and validation, as Intel's documentation does not always specify the exact methodology used in power estimation. Nonetheless, RAPL remains a valuable tool for energy-efficient computing, allowing software to make power-aware decisions based on estimated power usage.

Finally, for further details on Scaphandre and its monitoring capacities, in Appendix we define a tutorial on How To Integrate Scaphandre with Multiple QEMU VMs.

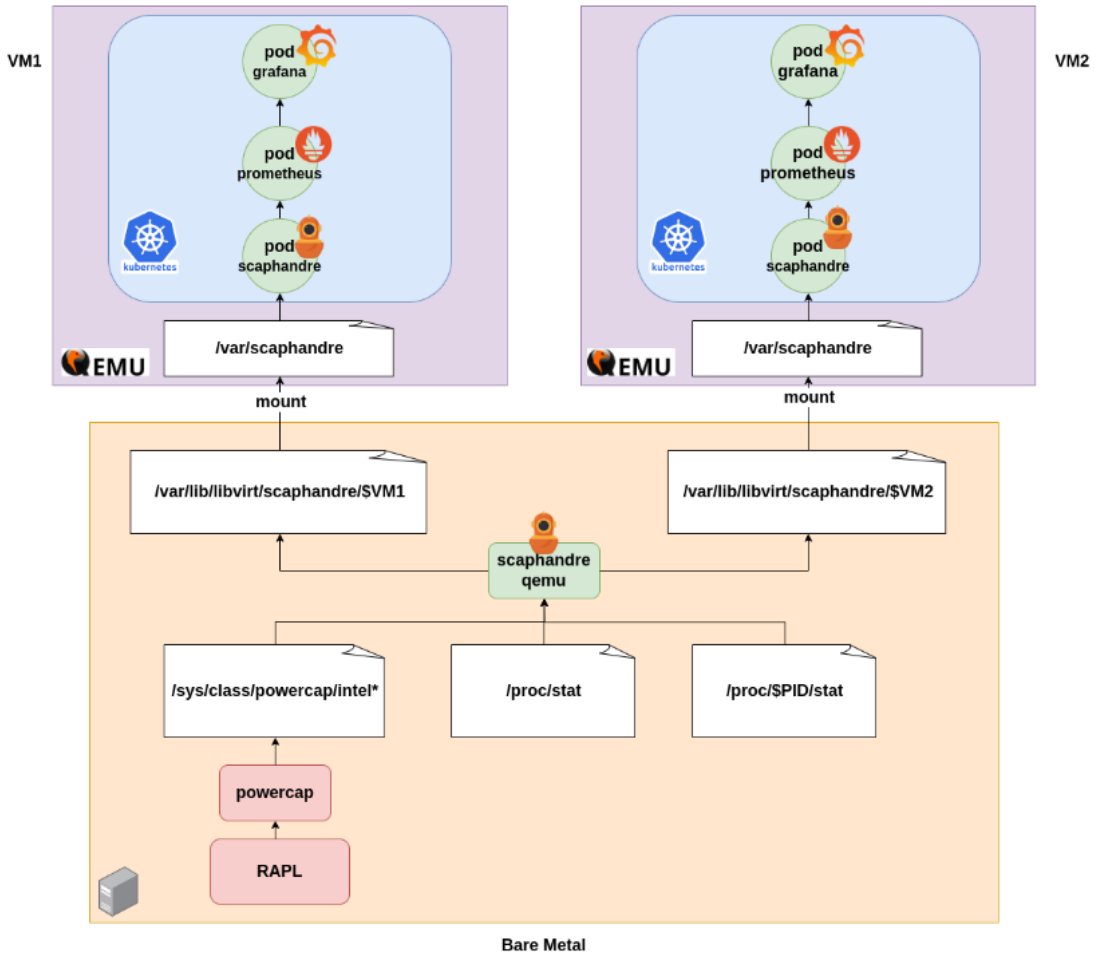


Figure 3.4: This diagram illustrates the architecture for monitoring power consumption in virtualized Kubernetes environments using Scaphandre, an open-source energy monitoring agent. The infrastructure consists of multiple virtual machines (VM1 and VM2), each running Kubernetes pods that include Scaphandre, Prometheus, and Grafana for data collection and visualization. Each VM mounts a directory (/var/scaphandre) that connects to the underlying bare metal host, where Scaphandre QEMU gathers energy consumption metrics from Intel RAPL (Running Average Power Limit) via powercap, as well as CPU usage statistics from /proc/stat and /proc/$PID/stat. The collected data is stored under /var/lib/libvirt/scaphandre for each VM and is used for detailed energy monitoring at both the hypervisor and VM levels.

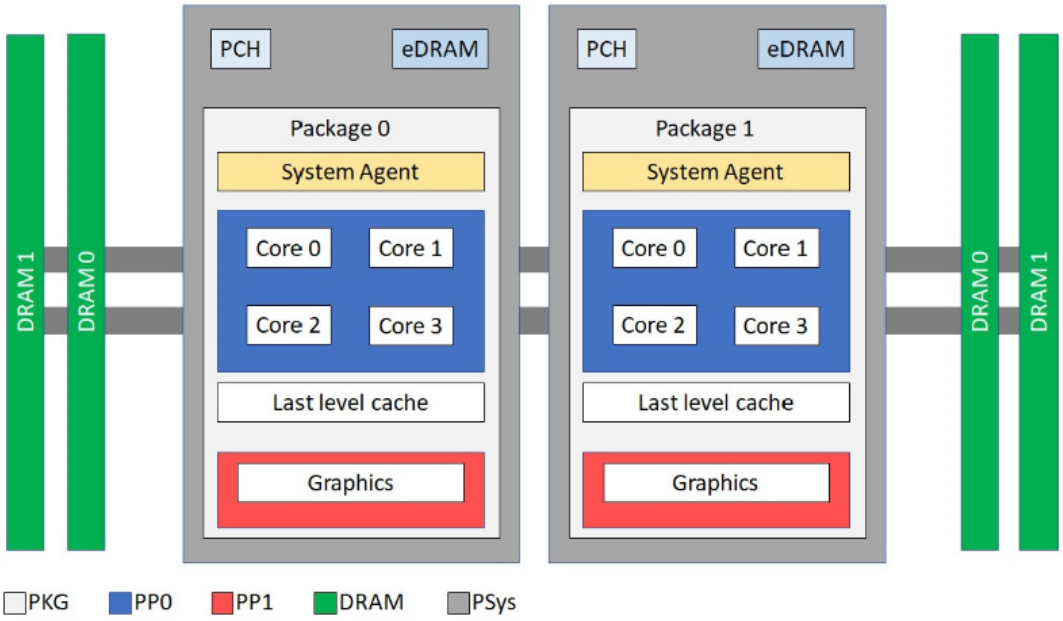


Figure 5. Power domains considered in RAPL interface.

Figure 3.5: This diagram illustrates the power domains monitored by the Running Average Power Limit (RAPL) interface within a multi-package CPU system. Each processor package contains multiple CPU cores (PP0 domain in blue), an integrated GPU (PP1 domain in red), a last-level cache, and a system agent managing interconnections. The PKG domain (outlined in gray) includes both core and uncore power consumption, while DRAM domains (green) represent energy used by memory modules. The PSys domain, though less documented, appears to estimate power consumption beyond the CPU, potentially including motherboard components. This visualization helps in understanding power distribution across CPU packages and memory systems.

Chapter 4

Experimental Evaluation

4.1 Energy Model

Our scenario involves multiple User Equipment (UE) devices downloading substantial data volumes, a common occurrence in the AI era. Use cases include downloading large AI models or their outputs, such as images, documents, 3D objects, or videos. For example, a smartphone may download object detection results from an uploaded image, or an AR device may retrieve 3D objects for real-time interaction.

In our system, the traffic is generated at the MEC server and forwarded to the 5G network. Initially, it passes through the data plane component of the 5GCN, specifically the UPF, before being sent to the 5G RAN. Within the disaggregated RAN, the traffic flows from the CU to the DU and finally to the RU.

We conducted experiments scaling the CU from 1 to 4 nodes, while fixing the number of DUs and UEs to 4, as shown in Figure 4.1. Varying rates of traffic with a fixed packet size was generated using *iperf* between the MEC server and the UEs, to simulate diverse traffic loads. The traffic volume load ranged from 20 MB to 2.5 GB. To ensure simultaneous data transfer across all UEs, Python threads were used to enable parallel execution of the *iperf* sessions. While the *iperf* sessions were running, we collected power metrics (in Watts) from the Prometheus server. Upon completion of the tests, we calculated the energy consumption (in Joules) by integrating the power consumption over the duration of the experiment, using the SciPy library's *trapezoid* function. The SciPy library's *trapezoid* function is a numerical integration method that estimates the integral of a given set of discrete data points using the trapezoidal rule. This rule approximates the area under a curve by dividing it into a series of adjacent trapezoids rather than using higher-order polynomial approximations. Given a set of power measurements collected at discrete time intervals during the experiment, the function computes the total energy consumption by summing the areas of these trapezoids, each of which represents the energy consumed over a small time step. The accuracy of this method depends on the resolution of the collected data, with finer time granularity providing a more precise estimation of the total energy consumption. Since power is measured in Watts and time every 10 seconds, the resulting integral gives energy in Joules, aligning with our

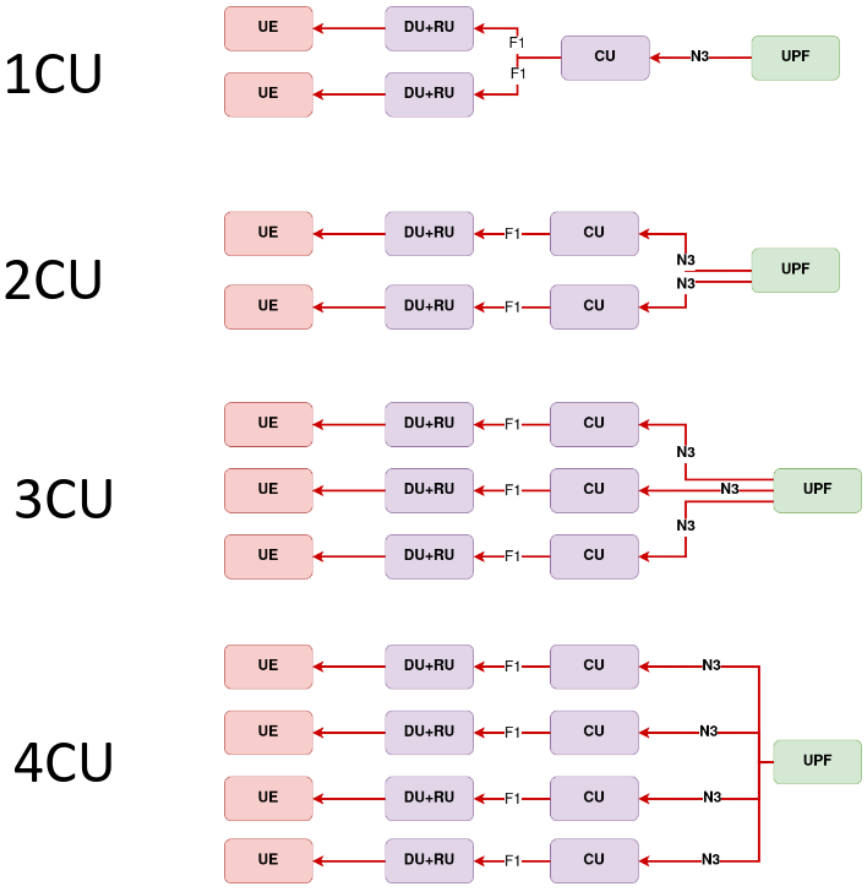


Figure 4.1: experiment setup

objective of quantifying energy usage throughout the experiment. Due to limitations in Prometheus, we were only able to achieve a maximum sampling rate of 10 seconds.

We developed an energy model composed by the following:
The **Host Energy Consumption** (E_{host}), representing the baseline energy for keeping the physical machine operational, is calculated as:

$$E_{\text{host}} = P_{\text{avg}} \cdot T_{\text{service}}$$

where P_{avg} is the average host power consumption, calculated as the mean of the power values sampled over a 10-minute period, and T_{service} is the service duration. For deployments with a single CU, T_{service} is the maximum recorded duration, whereas for multiple CUs, it sums the durations of all involved CUs.

The **Activation Energy Consumption** ($E_{\text{activation}}$) accounts for the energy required to deploy CUs on the node, computed as:

$$E_{\text{activation}} = P_{\text{activation}} \cdot T_{\text{deploy}} \cdot (N_{\text{CU}} - 1)$$

where $P_{\text{activation}}$ is the average activation power rate, T_{deploy} is the fixed deployment duration (averaged over 12 deployments of a single CU), and N_{CU} is the number of CUs being deployed. To guarantee UE connectivity, at least one CU must remain active in the RAN throughout its entire operation. Consequently, $N_{\text{CU}} - 1$ refers to the number of activations corresponding to the additional number of CUs that need to be deployed.

The **Service Energy Consumption** (E_{service}) reflects the energy used by RAN components especially the CU and DU during data processing and transmission, calculated as:

$$E_{\text{service}} = \int_0^{T_{\text{service}}} P_{\text{service}}(t)\, dt$$

where $P_{\text{service}}(t)$ represents the power usage of the service components sampled at one-second intervals.

Finally, the **Total Energy Consumption** (E_{total}) is given by:

$$E_{\text{total}} = E_{\text{host}} + E_{\text{activation}} + E_{\text{service}}$$

giving a comprehensive view of the energy requirements for the network operations.

4.2 In-the-field Experimental Results

Figure 4.2 shows the energy consumption results for each experiment. Specifically, each bar represents the average value of the corresponding experiment with a specified number of MB requested. Each bar is divided into three sections, each corresponding to one of the three energy consumption metrics discussed in Section 4.1. In every bar, the bottom and darkest partition is the host energy, the one in the middle is the activation energy, while the top

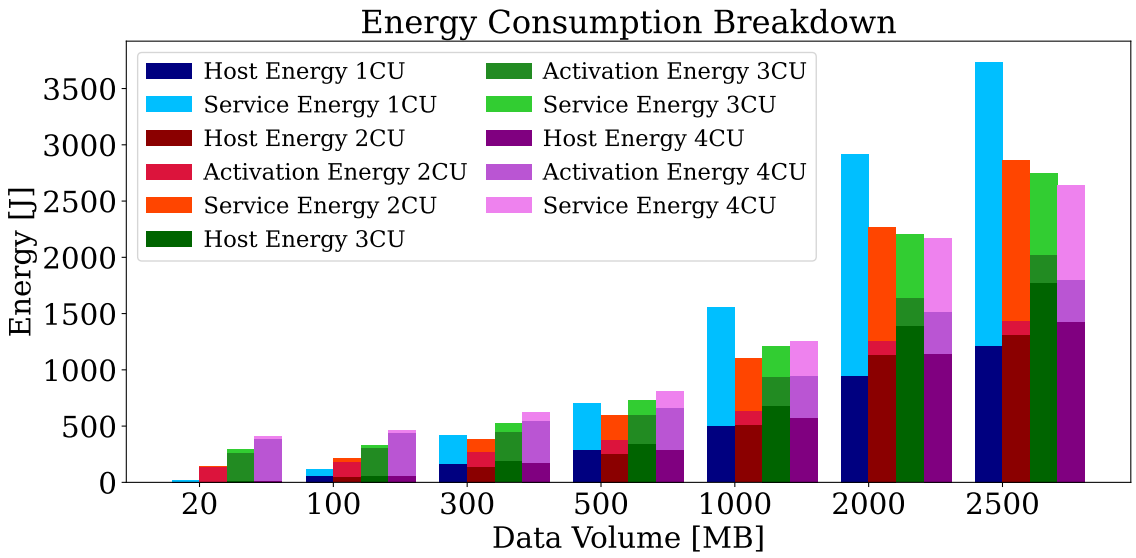


Figure 4.2: This bar chart illustrates the energy consumption across various data volume levels, segmented into host, service, and activation energy for CU configurations ranging from 1 to 4 nodes. The energy consumption increases with data volume, highlighting the impact of scaling CUs on overall energy usage.

and lightest one is the service energy. As expected, the energy consumption of each metric increases linearly with the amount of data requested, a trend confirmed by several other studies [27]. From this plot, we can intuitively observe that for certain ranges of requested data volume, there is an optimal number of CUs that consumes less energy compared to other configurations.

Building on this intuition, Figure 4.3 provides a clearer visual representation of the previous observation. As shown, each line follows a linear slope, indicating that energy consumption is directly proportional to the data volume. Additionally, we observe "interception points" or "switch points" where the energy efficiency of using a particular number of CUs surpasses that of others. These interception points indicate the data volume ranges where a specific configuration minimizes energy consumption. This insight is critical for developing adaptive policies that dynamically select the optimal number of CUs based on the identified data volume ranges. For example, if the requested data volume falls within a range where using 2 CUs is more energy-efficient than using 1 or 3 CUs, the system can adjust in real-time or proactively by predicting future data volume requests. Such policies enable energy savings by operating within the most efficient configuration for the given workload, reducing unnecessary energy consumption while maintaining performance. We will describe our proof-of-concept policy in Section 4.3.

To understand the reasons behind these empirical results, Figure 4.4 provides insights into the time duration changes for each data volume and the number of CU instances scheduled. The time taken by the 5G Network to transmit the total requested data (in

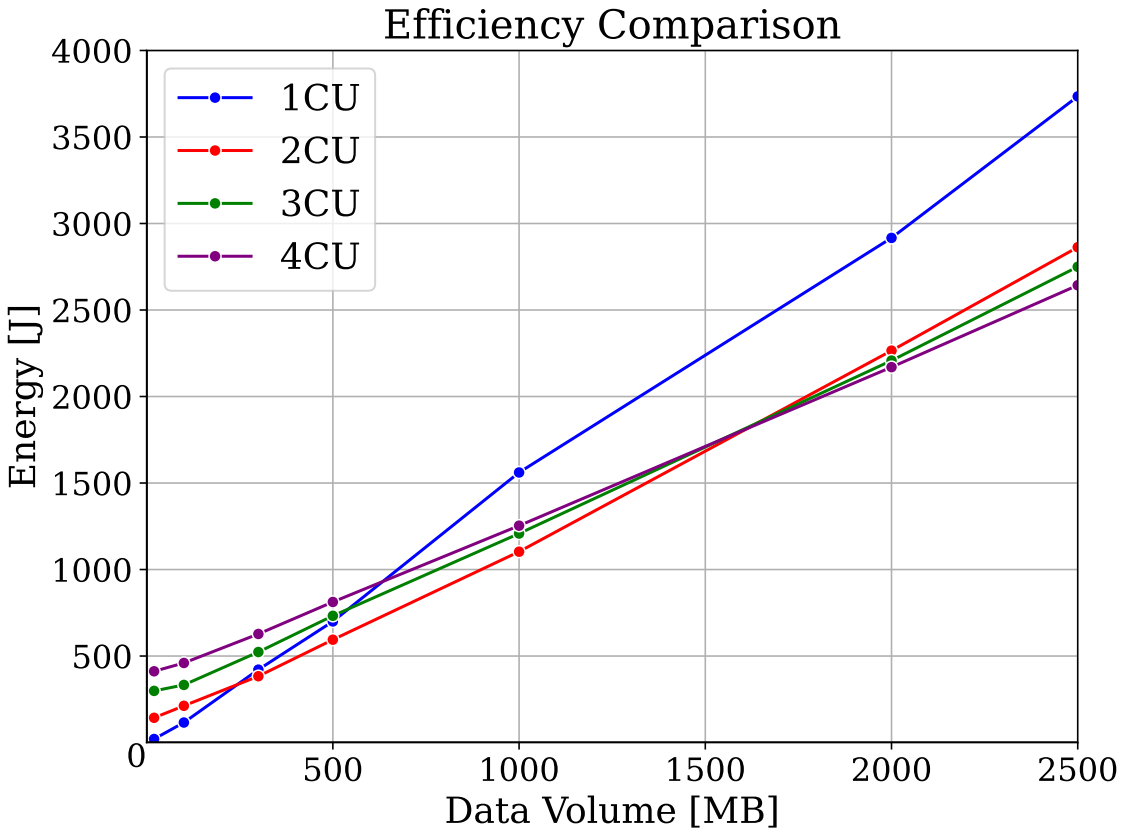


Figure 4.3: This graph compares the energy consumption of different CU configurations (1CU, 2CU, 3CU, and 4CU) as a function of data volume. Initially, for low data volumes (approximately 0-300 MB), the 1CU configuration exhibits the lowest energy consumption, making it the most efficient choice in this range. However, as the data volume increases beyond this threshold, the 1CU configuration becomes less efficient, consuming significantly more energy than higher CU configurations. Around 300 MB, an intercept point emerges where the energy consumption of 2CU, 3CU, and 4CU configurations starts to become lower than that of 1CU. Beyond approximately 1000 MB, the energy consumption of 2CU, 3CU, and 4CU remains relatively close, with 4CU offering a slight advantage at higher loads.

MB) increases as the data volume grows. Additionally, allocating more CUs allows the infrastructure to complete the service in less time, thereby improving performance in any application case where latency is a critical QoS factor. While our work primarily focuses on optimizing the energy consumption of the overall system, this plot highlights the potential for achieving a trade-off between latency and energy consumption or, in other words, between QoS and resource allocation. As better described in Section 4.3, we did not consider this trade-off in our policy manager for the moment, but we plan to explore it in future work.

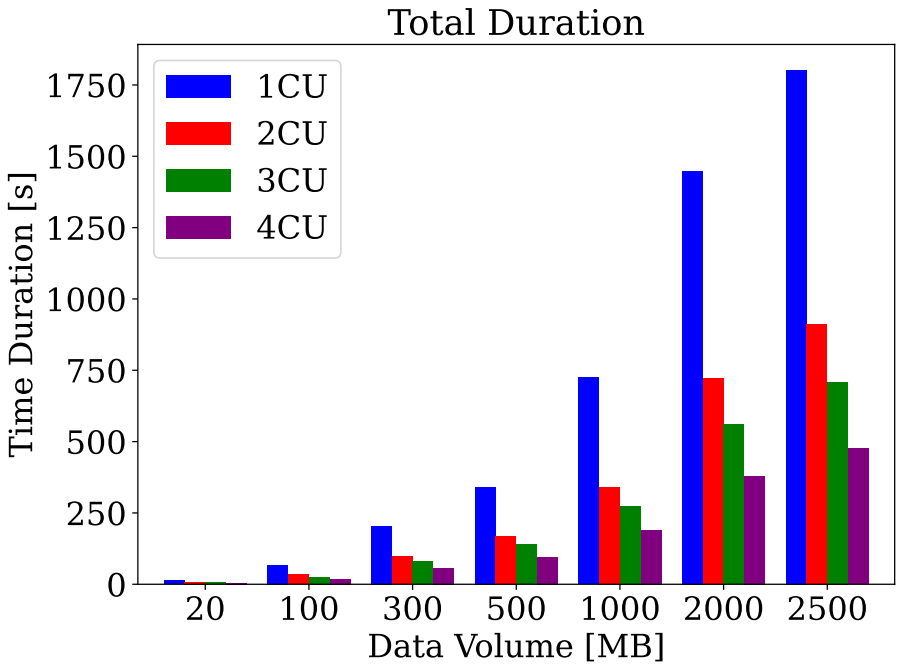


Figure 4.4: This bar chart illustrates the total duration required to transfer different data volumes under varying CU configurations (1CU, 2CU, 3CU, and 4CU). For smaller data volumes (below approximately 300 MB), the differences in transfer time across configurations are minimal. However, as the data volume increases, the performance gap becomes more evident. The 1CU configuration experiences the longest transfer durations, especially beyond 1000 MB, where its time requirements grow significantly. In contrast, higher CU configurations (particularly 3CU and 4CU) demonstrate a substantial reduction in transfer duration.

4.3 Policy Manager

Building on the insights derived from the results, we decided to implement a proof-of-concept scheduler based on our policy. Specifically, the policy dictates that the scheduler must allocate the optimal number of CUs, which is determined between consecutive intercept

points based on the requested real-time data volume. This policy manager is executed on the rApp as illustrated in 3.1. The rApp dynamically adjusts the number of CU instances based on Data Volume advertisements received from the MEC server, as outlined in Algorithm 1. Upon completion of the configuration process, the rApp signals the MEC server to continue data transmission. For instance, if the requested data volume falls within the range of 0 to 300 MB, the scheduler will allocate 1 CU; if the range is from 300 to 576 MB, it will allocate 2 CUs, and so on.

Algorithm 1 Dynamic CU Allocation Policy

Require: Real-time data volume V (in MB)
1: CU allocation thresholds $\{T_0, T_1, \ldots, T_n\}$,
2: corresponding CU allocations $\{CU_1, CU_2, \ldots, CU_{n+1}\}$
Ensure: Allocated number of CUs $CU_{allocated}$
3: Find k such that $T_k \leq V < T_{k+1}$
4: $CU_{allocated} \leftarrow CU_{k+1}$ $CU_{allocated}$

4.4 Benchmarks

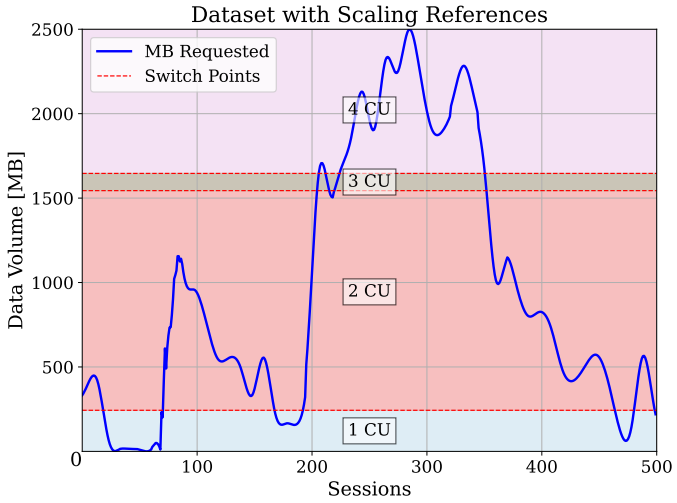


Figure 4.5: This figure illustrates the requested data volume across multiple sessions, overlaid with scaling references for different CU configurations. The blue line represents the MB requested per session, while the red dashed lines indicate the predefined switch points for scaling the number of CUs. The background color segments highlight the different CU regions, where transitions between 1CU, 2CU, 3CU, and 4CU occur based on traffic demand. As data volume increases, the system dynamically scales up the CU allocation to meet demand, and as traffic decreases, it scales down to conserve energy.

To test the scheduler, we selected a real dataset [57], which is a 5G trace of data

collected from a major Irish mobile operator. The dataset includes two mobility patterns (static and car) and two application patterns (video streaming and file download). For our experiment, we choose the file download trace in the static scenario. In Figure 4.5, the data volume over the entire duration of the dataset is shown. Instead of considering each MB requested at every time unit, which would analyze the latency implications of our scheduler–a concern not intended for this work–we map the data volume to a single "session", representing the entire duration of the corresponding data transfer. To make the test feasible, we normalize the MB values between 0 and the maximum data volume that we encounter in our experiments, which is 2500; however, this does not affect the overall behavior of the curve. We conducted the test using the policy described in Algorithm 1 and calculated the total energy consumed to complete all sessions by adding the average energy consumed by the entire infrastructure during each session, based on the experiments outlined in Section 4.1, for each corresponding number of CUs allocated.

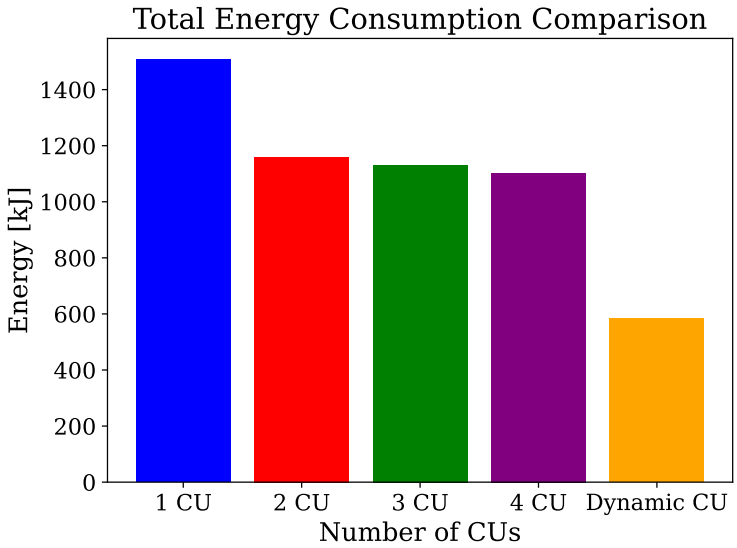


Figure 4.6: This bar chart compares the total energy consumption for different CU configurations, including a dynamic CU allocation approach. The results indicate that a static 1CU setup consumes the highest amount of energy, followed by 2CU, 3CU, and 4CU configurations, which exhibit similar energy usage. The dynamic CU allocation strategy, which adjusts the number of CUs based on real-time traffic demand, achieves the lowest energy consumption.

In Figure 4.6, we present a comparison between our dynamic scaling policy and the static one, where a fixed number of CUs is deployed across all sessions. As shown, our scheduler reduces energy consumption by approximately 53% compared to the static deployment with 4 CUs and up to around 60% compared to the worst-case scenario (the static deployment with 1 CU).

Conclusion and Future Works

In this thesis, we investigated the optimization of resource allocation within the O-RAN 5G framework to minimize energy consumption. We began by examining the existing architecture of 5G networks, focusing on the role of the Centralized Unit (CU) and its impact on power usage. After that, we examined various related works to outline the state of the art in energy efficiency techniques across the entire 5G infrastructure. Our review highlighted a gap in existing research, particularly in the area of dynamic scaling within the 5G RAN, with a specific need for innovation in CU scaling strategies. We developed an emulation of a real 5G infrastructure, incorporating an RF simulator, and conducted a series of experiments to accurately measure energy consumption by adjusting various parameters and operating conditions. Among the many insights gathered, we made an intriguing discovery: under specific data volume demands, there are cases where utilizing fewer resources leads to greater efficiency, while in other scenarios, allocating more resources proves to be the better approach. Building on this key finding, we proposed a dynamic CU scaling policy that intelligently adjusts resource allocation based on real-time traffic demands, optimizing both performance and energy efficiency. A real-world testbed was used to evaluate the energy savings achieved by our approach. Through extensive experimentation, we demonstrated that energy consumption is significantly influenced by CU deployment strategies and network traffic variations. Our results indicate that our proposed dynamic CU allocation policy can reduce energy usage by up to 60% compared to static configurations. These results demonstrate that fine-grained, traffic-aware scaling of 5G CUs can successfully balance energy efficiency and network performance, providing a sustainable approach to managing next-generation mobile networks.

Looking ahead, we plan to extend our system to even more complex scenarios, including real-radio devices such as USRPs and RUs, to better capture the impact of real-world radio operations on energy consumption. By incorporating these elements, we aim to refine our methodology and ensure that our proposed optimization strategies remain effective in more diverse and less controlled environments.

To further enhance energy efficiency, we will integrate AI-driven methods for forecasting traffic patterns and predicting the transmitting duration of UEs. This predictive capability will enable proactive scaling of CU components, allowing the system to dynamically adjust resources before congestion or underutilization occurs. Currently, some information exchange occurs between the MEC server, which is responsible for sending data traffic,

and the SMO, which acts as a centralized control entity managing the RAN. This back-and-forth communication introduces additional overhead is not concretely always possible. By implementing a UE demand prediction mechanism, the system can anticipate traffic fluctuations and optimize CU allocation in advance, reducing the need for constant coordination between the MEC server and SMO. This not only streamlines decision-making but also minimizes unnecessary energy expenditure while maintaining performance, ultimately leading to a more efficient and self-adaptive O-RAN framework.

Additionally, we plan to develop a dedicated framework for monitoring the real-time power consumption of radio devices within the RAN. This will involve retrieving power metrics from USRPs and RUs, providing deeper insights into how radio power consumption contributes to overall energy efficiency. Understanding these dynamics will allow us to refine our energy-saving policies further, ensuring that optimizations extend beyond CU allocation to the entire RAN infrastructure.

By integrating these advancements, we aim to create a more intelligent and adaptive resource management system that not only minimizes energy consumption but also improves the overall efficiency and sustainability of O-RAN 5G networks.

Appendix

A Tutorial on How To Integrate Scaphandre with Multiple QEMU VMs

Scaphandre is a power consumption monitoring tool designed to track energy usage across various environments, including virtual machines (VMs). This tutorial explains how to set up multiple QEMU virtual machines and integrate them with Scaphandre for power monitoring. We will go through two scripts: the first sets up a QEMU VM, and the second configures the VM for Scaphandre integration.

Part 1: Setting Up a QEMU Virtual Machine

Listing 4.1: Step 1: Define Variables

```
VM_NAME="vm0"
USERNAME="ubuntu"
PASSWORD="asd"
TEMPLATE_DIR="/var/lib/libvirt/images/templates"
VM_DIR="/var/lib/libvirt/images/$VM_NAME"
TEMPLATE_IMAGE="ubuntu-22-server.qcow2"
CLOUD_IMAGE_URL="https://cloud-images.ubuntu.com/jammy/current
   /jammy-server-cloudimg-amd64.img"
SSH_KEY=$(cat ~/.ssh/id_rsa.pub)
```

These variables define the VM name, default user credentials, storage locations, and the cloud image URL. The SSH key is read from the local machine to enable passwordless access to the VM.

Listing 4.2: Step 2: Install Required Packages

```
sudo apt-get update
sudo apt-get install -y \
    ninja-build zlib1g zlib1g-dev gcc-11 gcc-11-base libgcc
       -11-dev gcc \
    python3-venv python3-pip libglib2.0-dev git flex bison \
```

```
    libvirt-daemon libvirt-daemon-system cloud-image-utils
       cloud-utils whois \
    qemu qemu-kvm wget
```

This installs essential dependencies, including QEMU, KVM, Libvirt, and cloud utilities required for VM creation and management.

Listing 4.3: Step 3: Enable and Start Libvirt

```
sudo systemctl enable libvirtd
sudo systemctl start libvirtd
```

Ensures that the 'libvirtd' service is enabled and running for VM management.

Listing 4.4: Step 4: Download and Compile QEMU

```
cd /tmp
wget https://download.qemu.org/qemu-9.1.1.tar.xz
tar xvJf qemu-9.1.1.tar.xz
cd qemu-9.1.1
./configure
make -j$(nproc)
sudo make install
```

This downloads and compiles the latest QEMU version for optimal performance and compatibility.

Listing 4.5: Step 5: Prepare the Cloud Image

```
sudo mkdir -p $TEMPLATE_DIR
if [ ! -f "$TEMPLATE_DIR/$TEMPLATE_IMAGE" ]; then
    wget $CLOUD_IMAGE_URL -O /tmp/$TEMPLATE_IMAGE
    sudo mv /tmp/$TEMPLATE_IMAGE $TEMPLATE_DIR/$TEMPLATE_IMAGE
fi
```

This step downloads the Ubuntu cloud image and places it in the template directory.

Listing 4.6: Step 6: Setup VM Storage

```
sudo mkdir -p $VM_DIR
sudo qemu-img convert -f qcow2 -O qcow2 $TEMPLATE_DIR/
   $TEMPLATE_IMAGE $VM_DIR/root-disk.qcow2
sudo qemu-img resize $VM_DIR/root-disk.qcow2 50G
```

Creates a VM directory, converts the base image, and resizes it to 50GB.

Listing 4.7: Step 7: Generate Cloud-Init Configuration

```
cat <<EOF | sudo tee $VM_DIR/cloud-init.cfg
cloud-config
system_info:
```

```
  default_user:
    name: $USERNAME
    home: /home/$USERNAME
password: $PASSWORD
chpasswd: { expire: False }
hostname: $VM_NAME
ssh_pwauth: True
ssh_authorized_keys:
  - $SSH_KEY
EOF
```

Defines the cloud-init configuration for setting up the VM user, hostname, password, and SSH access.

Listing 4.8: Step 8: Create Cloud-Init ISO

```
sudo cloud-localds $VM_DIR/cloud-init.iso $VM_DIR/cloud-init.
   cfg
```

This generates an ISO file for cloud-init.

Listing 4.9: Step 9: Install the VM

```
sudo virt-install \
    --name $VM_NAME \
    --memory 4096 \
    --vcpus 7 \
    --disk $VM_DIR/root-disk.qcow2,device=disk,bus=virtio \
    --disk $VM_DIR/cloud-init.iso,device=cdrom \
    --os-variant ubuntu22.04 \
    --virt-type kvm \
    --graphics none \
    --network network=default,model=virtio \
    --import
```

Creates a VM with defined specifications.

Part 2: Integrating Scaphandre with the VM

Listing 4.10: Step 1: Define Variables and Install Rust

```
sudo apt update
sudo apt install -y curl
curl --proto '=https' --tlsv1.2 -sSf https://sh.rustup.rs | sh
    -s -- -y
source "$HOME/.cargo/env"
```

Installs Rust, required to build Scaphandre.

Listing 4.11: Step 2: Clone and Build Scaphandre

```
git clone https://github.com/RootLeo00/Dynamic-Scaling-Policy-
    Energy-5G-ORAN.git
cd Dynamic-Scaling-Policy-Energy-5G-ORAN/scaphandre-kubernetes
    /scaphandre
sudo apt install -y libssl-dev
cargo build --release
```

Downloads and compiles Scaphandre.

Listing 4.12: Step 3: Setup Shared Filesystem

```
sudo mkdir -p /var/lib/libvirt/scaphandre/$DOMAIN_NAME
sudo mount -t tmpfs tmpfs_$DOMAIN_NAME /var/lib/libvirt/
    scaphandre/$DOMAIN_NAME -o size=10m
```

Creates a shared filesystem for Scaphandre metrics storage.

Listing 4.13: Step 4: Modify VM Configuration for Shared Filesystem

```
sudo apt install -y xmlstarlet
virsh dumpxml $DOMAIN_NAME > ${DOMAIN_NAME}_temp.xml
xmlstarlet ed --inplace \
  -s "//devices" -t elem -n filesystem -v "" \
  -s "//devices/filesystem[last()]" -t attr -n type -v "mount"
      \
  -s "//devices/filesystem[last()]" -t attr -n accessmode -v "
     passthrough" \
  -s "//devices/filesystem[last()]" -t elem -n source -v "" \
  -s "//devices/filesystem[last()]/source" -t attr -n dir -v "
     $SHARED_FS_DIR" \
  -s "//devices/filesystem[last()]" -t elem -n target -v "" \
  -s "//devices/filesystem[last()]/target" -t attr -n dir -v "
     scaphandre" \
  ${DOMAIN_NAME}_temp.xml
```

Modifies the VM XML to mount the shared directory.

Listing 4.14: Step 5: Restart the VM and Start Scaphandre

```
virsh shutdown $DOMAIN_NAME
virsh start $DOMAIN_NAME
cd scaphandre
sudo target/release/scaphandre qemu &
```

Restarts the VM and launches Scaphandre as a background process.

Bibliography

[1] Charlotte Freitag et al., "The real climate and transformative impact of ICT: A critique of estimates, trends, and regulations," 2021.

[2] L. M. P. Larsen, H. L. Christiansen, S. Ruepp, and M. S. Berger, "Toward greener 5g and beyond radio access networksâa survey," *IEEE Open Journal of the Communications Society*, 2023.

[3] A. M. Abdalla, J. Rodriguez, I. Elfergani, and A. Teixeira, "Energy efficiency in the cloud radio access network (câran) for 5g mobile networks," in 2019.

[4] L. M. P. Larsen, A. Checko, and H. L. Christiansen, "A Survey of the Functional Splits Proposed for 5G Mobile Crosshaul Networks," *IEEE Communications Surveys & Tutorials*, 2019.

[5] M. Meraj and S. Kumar, "Evolution of mobile wireless technology from 0 g to 5 g," 2015. [Online]. Available: `https://api.semanticscholar.org/CorpusID:38119273`.

[6] V. H. M. Donald, "Advanced mobile phone service: The cellular concept," *The Bell System Technical Journal*, vol. 58, pp. 15–41, 1979. [Online]. Available: `https://api.semanticscholar.org/CorpusID:26746414`.

[7] J. R. Dumasig, *Total Access Communication System*. Unknown Publisher, 1981.

[8] European Telecommunications Standards Institute (ETSI), "Digital cellular telecommunications system (phase 2+); general packet radio service (gprs); service description; stage 1 (gsm 02.60 version 8.0.0 release 1999)," ETSI, Tech. Rep. TS 100 940 V8.0.0, 2000. [Online]. Available: `https://www.etsi.org/deliver/etsi_ts/100900_100999/100940/08.00.00_60/ts_100940v080000p.pdf`.

[9] International Telecommunication Union, "Recommendation ITU-R M.1457: Detailed specifications of the radio interfaces of IMT-2000," International Telecommunication Union, Tech. Rep., 2000. [Online]. Available: `https://www.itu.int/rec/R-REC-M.1457/en`.

[10] 3rd Generation Partnership Project (3GPP), "Technical specification group radio access network; evolved universal terrestrial radio access (e-utra); physical layer procedures (release 10)," 3GPP, Tech. Rep. TS 36.213 V10.0.0, 2011. [Online]. Available: `https://www.3gpp.org/ftp/Specs/archive/36_series/36.213/36213-a00.zip`.

[11] IEEE Computer Society, *Ieee standard for local and metropolitan area networks part 16: Air interface for broadband wireless access systems amendment 3: Advanced air interface*, 2011. [Online]. Available: https://standards.ieee.org/ieee/802.16m/4098/.

[12] 3rd Generation Partnership Project (3GPP), "Technical specification group radio access network; nr; overall description; stage-2 (release 15)," 3GPP, Tech. Rep. TS 38.300 V15.0.0, 2018. [Online]. Available: https://www.3gpp.org/.

[13] Y. Zhao, J. Zhao, W. Zhai, S. Sun, D. Niyato, and K.-Y. Lam, *A survey of 6g wireless communications: Emerging technologies*, 2020. arXiv: 2004.08549 [eess.SP]. [Online]. Available: https://arxiv.org/abs/2004.08549.

[14] 3rd Generation Partnership Project (3GPP), "Evolved universal terrestrial radio access (e-utra); lte; physical layer procedures (release 8)," 3GPP, Tech. Rep. TS 36.213 V8.0.0, 2008. [Online]. Available: https://www.3gpp.org/.

[15] 3rd Generation Partnership Project (3GPP), "Evolved universal terrestrial radio access (e-utra); lte-advanced; physical layer procedures (release 9)," 3GPP, Tech. Rep. TS 36.213 V9.0.0, 2010. [Online]. Available: https://www.3gpp.org/.

[16] 3rd Generation Partnership Project (3GPP), "Nr; overall description; stage-2 (release 16)," 3GPP, Tech. Rep. TS 38.300 V16.0.0, 2020. [Online]. Available: https://www.3gpp.org/.

[17] 3rd Generation Partnership Project (3GPP), "Nr; overall description; stage-2 (release 17)," 3GPP, Tech. Rep. TS 38.300 V17.0.0, 2022. [Online]. Available: https://www.3gpp.org/.

[18] IEEE, *Ieee 802.11: Wireless lan medium access control (mac) and physical layer (phy) specifications*, 2016. [Online]. Available: https://standards.ieee.org/standard/802_11-2016.html.

[19] O.-R. Alliance, *O-ran: Open radio access network*, Accessed: 2025-03-10, 2020. [Online]. Available: https://www.o-ran.org/.

[20] Q. Zhang, L. Cheng, and R. Boutaba, "Cloud computing: State-of-the-art and research challenges," *Journal of Internet Services and Applications*, vol. 1, no. 1, pp. 7–18, 2019. DOI: 10.1007/s13174-019-0097-1. [Online]. Available: https://link.springer.com/article/10.1007/s13174-019-0097-1.

[21] C.-C. Lin, C.-T. Tsai, Y.-L. Liu, T.-T. Chang, and Y.-S. Chang, "Security and privacy in 5g-iiot smart factories: Novel approaches, trends, and challenges," *Mobile Networks and Applications*, vol. 28, pp. 1–16, Jul. 2023. DOI: 10.1007/s11036-023-02143-5.

[22] L. Peterson, O. Sunay, and B. Davie, *Private 5G: A Systems Approach*. 2023, Licensed under CC BY-NC-ND 4.0. [Online]. Available: https://github.com/SystemsApproach/private5g.

[23] A. F. Rochim, B. Harijadi, Y. P. Purbanugraha, S. Fuad, and K. A. Nugroho, "Performance comparison of wireless protocol ieee 802.11ax vs 802.11ac," in *2020 International Conference on Smart Technology and Applications (ICoSTA)*, 2020, pp. 1–5. DOI: 10.1109/ICoSTA48221.2020.1570609404.

[24] E. Westberg, J. Staudinger, J. Annes, and V. Shilimkar, "5g infrastructure rf solutions: Challenges and opportunities," *IEEE Microwave Magazine*, vol. 20, no. 12, pp. 51–58, 2019. DOI: 10.1109/MMM.2019.2941631.

[25] N. Aryal, E. Bertin, and N. Crespi, "Open radio access network challenges for next generation mobile network," in *2023 26th Conference on Innovation in Clouds, Internet and Networks and Workshops (ICIN)*, 2023, pp. 90–94. DOI: 10.1109/ICIN56760.2023.10073507.

[26] S. Mollahasani, T. Pamuklu, R. Wilson, and M. Erol-Kantarci, "Energy-aware dynamic du selection and nf relocation in o-ran using actorâcritic learning," *Sensors*, vol. 22, no. 13, 2022, ISSN: 1424-8220. DOI: 10.3390/s22135029. [Online]. Available: https://www.mdpi.com/1424-8220/22/13/5029.

[27] D. LÃ³*pez – PÃ©rez*, A. De Domenico, N. Piovesan, *et al.*, "A survey on 5g radio access network energy efficiency: Massive mimo, lean carrier design, sleep modes, and machine learning," *IEEE Communications Surveys Tutorials*, 2022.

[28] M. Hoffmann and M. DryjaÅski, "Energy efficiency in open ran: Rf channel reconfiguration use case," *IEEE Access*, vol. 12, pp. 118 493–118 501, 2024. DOI: 10.1109/ACCESS.2024.3449700.

[29] J. Wu, Y. Zhang, M. Zukerman, and E. K.-N. Yung, "Energy-efficient base-stations sleep-mode techniques in green cellular networks: A survey," *IEEE Communications Surveys Tutorials*, vol. 17, no. 2, pp. 803–826, 2015. DOI: 10.1109/COMST.2015.2403395.

[30] G. Auer, V. Giannini, C. Desset, *et al.*, "How much energy is needed to run a wireless network?" *IEEE Wireless Communications*, vol. 18, no. 5, pp. 40–49, 2011. DOI: 10.1109/MWC.2011.6056691.

[31] A. I. Abubakar, O. Onireti, Y. Sambo, L. Zhang, G. K. Ragesh, and M. Ali Imran, "Energy efficiency of open radio access network: A survey," in *2023 IEEE 97th Vehicular Technology Conference (VTC2023-Spring)*, 2023, pp. 1–7. DOI: 10.1109/VTC2023-Spring57618.2023.10200477.

[32] F. Kooshki, A. G. Armada, M. M. Mowla, A. Flizikowski, and S. Pietrzyk, "Energy-efficient sleep mode schemes for cell-less ran in 5g and beyond 5g networks," *IEEE Access*, 2023.

[33] L. M. Moreira Zorello, M. Sodano, S. Troia, and G. Maier, "Power-efficient baseband-function placement in latency-constrained 5g metro access," *IEEE Transactions on Green Communications and Networking*, 2022.

[34] H. Hojeij, M. Sharara, S. Hoteit, and V. VÃ¨que, "Dynamic placement of o-cu and o-du functionalities in open-ran architecture," in *2023 20th Annual IEEE International Conference on Sensing, Communication, and Networking (SECON)*, 2023.

[35] A. Mudvari, N. Makris, and L. Tassiulas, "Ml-driven scaling of 5g cloud-native rans," in *2021 IEEE Global Communications Conference (GLOBECOM)*, 2021.

[36] M. A. Habib, P. E. I. Rivera, Y. Ozcan, *et al.*, "Transformer-based wireless traffic prediction and network optimization in o-ran," in *2024 IEEE International Conference on Communications Workshops (ICC Workshops)*, 2024, pp. 1–6. DOI: 10.1109/ICCWorkshops59551.2024.10615438.

[37] A. Dalgkitsis, P.-V. Mekikis, A. Antonopoulos, G. Kormentzas, and C. Verikoukis, "Dynamic resource aware vnf placement with deep reinforcement learning for 5g networks," in *GLOBECOM 2020 - 2020 IEEE Global Communications Conference*, 2020, pp. 1–6. DOI: 10.1109/GLOBECOM42002.2020.9322512.

[38] Z. Zhu, H. Li, Y. Chen, Z. Lu, and X. Wen, "Joint optimization of functional split, base station sleeping, and user association in crosshaul-based v-ran," *IEEE Internet of Things Journal*, vol. 11, pp. 32 598–32 616, 2024. [Online]. Available: https://api.semanticscholar.org/CorpusID:270944485.

[39] H. Gupta, A. F. A, M. Kumar, and B. R. Tamma, *Traffic-aware dynamic functional split for 5g cloud radio access networks*, 2022. arXiv: 2202.09256 [cs.NI]. [Online]. Available: https://arxiv.org/abs/2202.09256.

[40] Y. Xiao, J. Zhang, and Y. Ji, "Energy-efficient du-cu deployment and lightpath provisioning for service-oriented 5g metro access/aggregation networks," *Journal of Lightwave Technology*, vol. 39, no. 17, pp. 5347–5361, 2021. DOI: 10.1109/JLT.2021.3069897.

[41] J. Lorincz, A. KukuruzoviÄ, and Z. BlaÅ¾eviÄ, "A comprehensive overview of network slicing for improving the energy efficiency of fifth-generation networks," *Sensors*, vol. 24, no. 10, 2024, ISSN: 1424-8220. DOI: 10.3390/s24103242. [Online]. Available: https://www.mdpi.com/1424-8220/24/10/3242.

[42] R. Singh, C. Hasan, X. Foukas, M. Fiore, M. K. Marina, and Y. Wang, "Energy-efficient orchestration of metro-scale 5g radio access networks," in *IEEE INFOCOM 2021 - IEEE Conference on Computer Communications*, 2021.

[43] S. Maxenti, S. D'Oro, L. Bonati, M. Polese, A. Capone, and T. Melodia, *Scalo-ran: Energy-aware network intelligence scaling in open ran*, 2024. arXiv: 2312.05096 [cs.NI]. [Online]. Available: https://arxiv.org/abs/2312.05096.

[44] H. Li, P. Li, K. D. Assis, *et al.*, "NetMind: Adaptive RAN Baseband Function Placement by GCN Encoding and Maze-solving DRL," in *2024 IEEE Wireless Communications and Networking Conference (WCNC)*, IEEE, 2024. DOI: 10.1109/wcnc57260.2024.10571268.

[45] A. A. A. Rage, N. Wang, and R. Tafazolli, "Nfscaler: Ai-powered 5g-and-beyond network function scaler for qos assurance and energy efficiency," in *2024 IEEE 10th International Conference on Network Softwarization (NetSoft)*, 2024.

[46] S. Urumkar, B. Ramamurthy, and S. Sharma, "Improving energy efficiency in open ran through dynamic cpu scheduling," Dec. 2023. DOI: 10.1109/ANTS59832.2023.10469315.

[47] X. Liang, A. Al-Tahmeesschi, Q. Wang, S. Chetty, C. Sun, and H. Ahmadi, *Enhancing energy efficiency in o-ran through intelligent xapps deployment*, 2024. arXiv: 2405.10116 [eess.SY]. [Online]. Available: https://arxiv.org/abs/2405.10116.

[48] H. Li, A. Emami, K. D. R. Assis, *et al.*, "Drl-based energy-efficient baseband function deployments for service-oriented open ran," *IEEE Transactions on Green Communications and Networking*, 2024.

[49] N. K. Shankaranarayanan, Z. Li, I. Seskar, *et al.*, "Poet: A platform for o-ran energy efficiency testing," in *2024 IEEE 100th Vehicular Technology Conference (VTC2024-Fall)*, 2024.

[50] Z. Zhu, H. Li, Y. Chen, X. Wen, Z. Lu, and L. Wang, "Joint base station sleeping and functional split orchestration in crosshaul-based v-ran," in *2023 IEEE Wireless Communications and Networking Conference (WCNC)*, 2023.

[51] R. Joda, T. Pamuklu, P. E. Iturria-Rivera, and M. Erol-Kantarci, "Deep reinforcement learning-based joint user association and cuâdu placement in o-ran," *IEEE Transactions on Network and Service Management*, 2022.

[52] "Slices, a scientific instrument for the networking community," *Computer Communications*, vol. 193, pp. 189–203, 2022.

[53] N. Nikaein, M. K. Marina, S. Manickam, A. Dawson, R. Knopp, and C. Bonnet, "Openairinterface: A flexible platform for 5g research," 2014.

[54] B. Petit, *Scaphandre*, version v1.0, 2023. [Online]. Available: https://github.com/hubblo-org/scaphandre.

[55] V. Gudepu, B. Chirumamilla, R. R. Tella, *et al.*, "Earnest: Experimental analysis of ran energy with open-source software tools," in *2024 16th International Conference on COMmunication Systems NETworkS (COMSNETS)*, 2024.

[56] M. Jay, V. Ostapenco, L. Lefevre, D. Trystram, A.-C. Orgerie, and B. Fichel, "An experimental comparison of software-based power meters: Focus on cpu and gpu," in *2023 IEEE/ACM 23rd International Symposium on Cluster, Cloud and Internet Computing (CCGrid)*, 2023.

[57] D. Raca, D. Leahy, C. J. Sreenan, and J. J. Quinlan, "Beyond throughput, the next generation: a 5G dataset with channel and context metrics," 2020.