6G BLUR: The Blurring RAN

SMART

Grant No. TSI-063000-2021-56

E4: Algorithms and results report SMART

Financiado por la Unión Europea NextGenerationEU
Plan de Recuperación, Transformación y Resiliencia
UNICO I+D
Gobierno de España
Centre Tecnològic de Telecomunicacions de Catalunya

Abstract

This report will describe the final mechanisms/algorithms developed and the results obtained within the 6G BLUR SMART project inside the different key concepts that are later included in the defined proof of concepts of the different use cases.

Document properties

Document number	E4
Document title	Algorithms and results report
Document responsible	Jorge Baranda, Albert Bel (CTTC)
Document editor	Jorge Baranda, Albert Bel (CTTC)
Authors	Jorge Baranda, Albert Bel, Sergio Barrachina, Miquel Payaró, Engin Zeydan, Josep Mangues-Bafalluy, Marc Carrascosa, Katerina Koutlia, Ana Larrañaga, Gabriel Carvalho , Biljana Bojovic, Sandra Lagén, Nikolaos Bartzoudis, Antonio Roman, José Rubio Fernández (CTTC), Manuel Lorenzo, Saravanan Kalimuthu, Marc Molla, Jose Luis Jimenez, Inmaculada Rafael, Miguel Angel Lopez Serrano, Alvaro Vlad (Ericsson), Carlos Javier Soleto Ramos, Rubén Cerezo, Diego San Cristobal Epalza, Fernando Beltran Gonzalez, Alejandro Ramiro Muñoz (Ericsson), Mattia Lecci, Haoxin Sun, Frankie Garcia, F. Javier Rivas Tocado, Carles Navarro Manchón (Keysight), David Gregoratti, Ismael Gómez (SRS), Luis Miguel Contreras Murillo, Javier Velázquez Martínez, Óscar Gil, María Teresa Aparicio, Elena Serna Santiago (Telefónica)
Target dissemination level	Public
Status of the document	Final
Version	1.0
Delivery date	31 December 2025
Actual delivery date	31 December 2025

Document history

Revision	Date	Issued by	Description
0.1	17/09/2024	Albert Bel, Jorge Baranda (CTTC)	Initial ToC
0.2	12/11/2024	Jorge Baranda (CTTC)	Joint K1.1, K1.2

Financiado por la Unión Europea NextGenerationEU
Plan de Recuperación, Transformación y Resiliencia
UNICO I+D
GOBIERNO DE ESPAÑA
cttc Centre Tecnològic de Telecomunicacions de Catalunya

0.3	09/12/2024	Saravannan, Jose Luis Jimenez (Ericsson), David Gregoratti (SRS), Jorge Baranda (CTTC)	Ericsson : Joint K1.3, Joint K1.4, Joint K3.2, UC1PoC1, UC2PoC2, UC3PoC1 ; SRS : Smart K1.1, Smart K2.1, UC1PoC2, UC2PoC1; CTTC: Document edition
0.4	10/06/2025	Katerina Koutlia, Ana Larrañaga, Jorge Baranda (CTTC), Javier Velázquez Martínez, Luis M. Contreras, Oscar Gil Lucia (Telefónica), Luis Díez, Fátima Khan, Neco Villegas (UNICAN), Carles Navarro, Mattia Lecci (Keysight)	CTTC: Smart K1.2, Smart K2.3, Smart K2.5, UC1PoC2, UC2PoC1, document edition; Telefonica: Smart K2.2, Smart K2.4, Smart K2.6, Joint K2.1, UC1PoC2, UC2PoC1; UNICAN: Joint K2.2, Joint K2.3, UC2PoC1, Keysight: Joint K3.1, UC2PoC1
0.5	10/11/2025	Albert Bel, Nikolaos Bartzoudis, Jorge Baranda (CTTC)	Smart K1.4, Smart K3.1, Joint K3.1, UC3PoC2, UC3PoC3, document Editions
0.6	25/11/205	Albert Bel, Jorge Baranda (CTTC)	Document division: Smart-Joint
0.7	01/12/2025	Albert Bel, Jorge Baranda (CTTC)	Final Review
1.0	12/12/2025	Jorge Baranda, Albert Bel (CTTC)	Final Edition of SMART E4 document

Disclaimer

This document has been produced in the context of the 6G BLUR Project. The research leading to these results has received funding from the Ministerio de Asuntos Económicos y Transformación Digital (MINECO), under grant TSI-063000-2021-56/-57.

Contents

Financiado por la Unión Europea NextGenerationEU
Plan de Recuperación, Transformación y Resiliencia
UNICO I+D
GOBIERNO DE ESPAÑA
cttc Centre Tecnològic de Telecomunicacions de Catalunya

List of Figures

Financiado por la Unión Europea NextGenerationEU
Plan de Recuperación, Transformación y Resiliencia
UNICO I+D
GOBIERNO DE ESPAÑA
cttc Centre Tecnològic de Telecomunicacions de Catalunya

Financiado por la Unión Europea NextGenerationEU
Plan de Recuperación, Transformación y Resiliencia
UNICO I+D
GOBIERNO DE ESPAÑA
cttc Centre Tecnològic de Telecomunicacions de Catalunya

Financiado por la Unión Europea NextGenerationEU
Plan de Recuperación, Transformación y Resiliencia
UNICO I+D
GOBIERNO DE ESPAÑA
cttc Centre Tecnològic de Telecomunicacions de Catalunya

List of Tables

List of Acronyms

5QI - 5G QoS Identifier

A3C – Asynchronous Advantage Actor-Critic

AD – Anomaly Detection

AF – Application Function

AGV – Automated Guided Vehicle

AI – Artificial Intelligence

API – Application Programming Interface

APN – Access point name

APU – Application Processing Unit

AS – Application Server

ASP – Authorized Service Provider

AXI - Advanced eXtensible Interface

B5G – Beyond 5G

BFP – Block Floating Point

BH – Backhaul

BWP – Bandwidth part

CaaS – Container as a Service

CAPEX – Capital Expenditures

CC – Components Carriers

CG – Cloud Gaming

CLI – Command Line Interface

CP – Control-plane

CR - Control and Scheduling Signaling rate

CSP – Communications Service Provider

CV – Computer Vision

DCI - Downlink Control Indicators

DCI – Downlink Control Indicators

DDP – Drift plus penalty

DL – Downlink

DMA - Direct Memory Access

DNN – Data Network Name

DPDK – Data Plane Development Kit

DRB – Data Radio Bearer

DRL – Deep Reinforcement Learning

DSCP – Differentiated Services Code Point

DT – Digital Twin

EDA – Exploratory Data Analysis

EDA – Exploratory Data Analysis

FFS – Flexible Functional Split

FFT – Fast Fourier Transformation

FH – Fronthaul

FLR – Frame Loss Ratio

gNB – next Generation Node B (5G node B)

GNN – Graph Neural Network

HL – Hierarchical Level

HOL – head-of-line

iFFT – Inverse Fast Fourier Transformation

ILP – Integer Linear Programming

IRU – Indoor Radio Unit

KPI – Key Performance Indicator

KPM - Key Performance Measurement

LC – Logical Channel

LLM – Large Language Model

LSTM – Long Short-Term Memory

MAC – Medium Access Control

MAE – Mean Absolute Error

MANO – Management and orchestration

MCS – Modulation and Coding Scheme

MDP – Markov Decision Process

MH – Midhaul

MILP – Mixed Integer Linear Programming

MIMO – Multiple-Input, Multiple-Output

MKP – Multiple Knapsack Problem

ML – Machine Learning

MLP – Multi-layer Perceptron

MNO – Mobile Network Operator

MPC – model-predictive proactive scheduler

MPSoC – Multiprocessor System-on-Chip

MSE – Mean Square Error

NDT – Network Digital Twin

Near-RT – near Real Time

NEF – Network Exposure Function

NF – Network Function

NIC – Network Interface Card

NPN – Non-Public Network

NRP – Network Resource Partition

NS – Network Service

Financiado por la Unión Europea NextGenerationEU
Plan de Recuperación, Transformación y Resiliencia
UNICO I+D
GOBIERNO DE ESPAÑA
cttc Centre Tecnològic de Telecomunicacions de Catalunya

NSSAI - Network Slice Selection Assistance Information

OAM – Operations, Administration and Maintenance

OFH – Open Fronthaul

OPEX – Operational Expenditures

OS – Operative System

OSM – Open Source Mano

OSS – Operational Support System

OTA – Over the air

OTIC – Open Testing and Integration Centre

OWD – One Way Delay

PDB – Packet Delay Budget

PDU – Packet Data Unit

PF – Proportional Fair

PL – Programmable Logic

PLMN – Public Land Mobile Network

PN – Public Network

PNI-NPN – Public Network Integrated- Non-Public Network

PoC – Proof of Concept

PoP – Point of Presence

PPO – Proximal Policy Optimization

PR – Peak Rate

PRACH – Physical Random Access Channel

PRB – Physical Resource Block

QoS – Quality of Service

R2C – Revenue to Cost Ratio

RAG – Retrieval-Augmented Generation

RAN – Radio Access Network

RDS – Radio Dot System

REG – Resource Element Groups

RFSoC – Radio Frequency System-on-Chip

RIC – Radio Intelligent Controller

RL- Reinforcement Learning

RLC – Radio Link Control

RNN – Recurrent Neural Networks

RR – Round Robin

RRM – Radio Resource Management

RSVP - Resource ReSerVation Protocol

RTT – Round-Trip Time

RU – Radio Unit

RW – Reception Window

SBA – Service Based Architecture

SCS – Subcarrier spacing

SDF – Service Data Flow

SDP – Source Demarcation Point

SDR – Software Defined Radio

SDU – Service Data Units

SIMD – Single Instruction, Multiple Data

SLA – Service Level Agreement

SMF – Session Management Function

SRB – Signalling Radio Bearer

SRS – Software Radio Systems

SST – Slice service type

SVR – Support Vector Regressor

TAE – Time Alignment Errors

TBS - Transport Block Size

TDD – Time Division Duplex

ToR – Top of Rack

TTI - Transmission Time Interval

TTI – Transmission Time Interval

TW – Transmission Window

UE – User Equipment

UL – Uplink

UP – User-plane

UPF – User Plane Function

URLLC – Ultra-reliable Low-Latency Communication

VLiM – Virtual Link Mapping

VM – Virtual Machine

VNE – Virtual Network Embedding

VNF – Virtual Network Function

VNoM – Virtual Node Mapping

VNR – Virtual Network Request

WRR – Weighted Round Robin

XR – eXtended Reality

ZDM – Zero-Defect Manufacturing

ZMQ – Zero Message Queuing

ZSM – Zero-touch Service Management

Financiado por la Unión Europea NextGenerationEU
Plan de Recuperación, Transformación y Resiliencia
UNICO I+D
GOBIERNO DE ESPAÑA
cttc Centre Tecnològic de Telecomunicacions de Catalunya

Executive Summary and Key Contributions

In this deliverable *(E4: Mechanisms and result reports)*, we present the algorithms/procedures and the results obtained within the different key concepts and the derived proof of concepts defined in the 6G-BLUR SMART project.

This project specifically targets: (i) to design and to validate a RAN control architecture adapting to all the required decision timescales, from policy definition to scheduling, and (ii) to design and validate Smart algorithms for efficient end-to-end resource management, including decision-making at all timescales, with real-time and non-real time decisions.

Based on the work developed in previous deliverables, namely E1 and E3, this report focuses on the beyond state of the art work developed within ten Key concepts[1], which explored the disaggregation of E2 node into CU and DU entities and its characterization to achieve more decentralized architectures, strategies to improve capabilities of Open Fronthaul interface or to select the best functional split in O-RAN, optimization of Medium Access Control (MAC) protocols for exploiting multiplexing gains, the design of effective Quality of Service (QoS) algorithms for time-critical immersive applications on for the aggregation of multiple services in the transport network, the design of an orchestration system to perform dynamic offloading/switching of Radio Unit tasks, and the close-loop operation in the RAN segment via a data-driven approach based on AI/ML algorithms with the information exposed by monitoring xApps or a Network Digital Twin (NDT).

All these key concepts, together with other key concepts developed within the other subproject of the 6G-BLUR project (6G-SMART) have been integrated four Proof of concepts (PoCs), showing the complementarity between subprojects. The key contributions of such PoCs are summarised as follows:

- O-Distributed Unit (DU) pooling at centralized locations is indeed feasible with the current O-RAN O-FH specifications, provided that the relevant parameters of the O-DU are appropriately adjusted. It is crucial to adjust the O-DU TX and RX windows in order to reflect the O-RU timing and the packet delay and jitter introduced by the O-FH. The E2E latency achieved by a system with a long O-FH is affected by multiple factors: number of registered UEs, computational capabilities of the servers running the RAN functions, SNR at the air interface, etc. Adjusting the time available for processing at the O-DU can be used to optimize

[1] Key concepts reflected in E3 deliverable K1.3 "Dynamic Configuration of topologies/architectures for MH distribution" and K1.5 "Dynamic Configuration of FH for distributed and centralized architectures" have not been fully developed in favour of more extended development of K2.2 "Characterization of the FH and MH transport" and K2.4 "Radio stack optimization", which are lately integrated into UC2PoC1 development.

Financiado por la Unión Europea NextGenerationEU
Plan de Recuperación, Transformación y Resiliencia
UNICO I+D
GOBIERNO DE ESPAÑA
cttc Centre Tecnològic de Telecomunicacions de Catalunya

the E2E latency, but conservative settings should be prioritized to avoid connectivity problems when the system is under high load conditions.

- Data-driven RAN optimization of Non-Public Networks (NPNs) can be obtained through the exposure of appropriate Application Programming Interfaces (APIs) fed with ML algorithms and exposed by an NDT.
- An adaptive micro-orchestration for FPGA SoC-based RU functions combining machine learning-based PRB usage forecasting with hierarchical reactive control can achieve measurable improvements in energy efficiency (32.6% power reduction), latency performance (15x acceleration), and resource utilization while maintaining safety and reliability constraints.
- The integration of AI-driven orchestration with O-RAN–compliant control loops enable the optimization of radio resource allocation by combining a monitoring xApp, a PRB-allocation xApp, and an LLM-based RAG/agent framework. The proposed system successfully validated the end-to-end cycle of data acquisition, semantic reasoning, and near-real-time actuation.

This summary is a high-level overview, but all the details can be found and are extensively developed within the different sections of this document.

Financiado por la Unión Europea NextGenerationEU
Plan de Recuperación, Transformación y Resiliencia
UNICO I+D
GOBIERNO DE ESPAÑA
cttc Centre Tecnològic de Telecomunicacions de Catalunya

1 Introduction

The aim of this document is to present the algorithms/procedures and the results beyond state-of-the-art (SoA) obtained for each KC and later present the final integrated setups for the different PoC at each UC to obtain further results/procedures (e.g., steps of the zero-touch NDT or steps to perform the e2e orchestration of the o-ran based network with emulated transport topology).

The blurring RAN (6GBLUR) is a project funded by the UNICO MINECO program, structured into two complementary subprojects: 6GBLUR-SMART — focused on intelligent decision-making for efficient end-to-end resource management — and 6GBLUR-JOINT — centred on the joint control and orchestration of RAN and transport networks.

Next-generation RANs face several technical challenges, including the need for high spectral efficiency, reduced energy consumption, resource pooling, scalability, and seamless cross-layer interaction. Current standardization efforts (e.g., 3GPP and O-RAN) address these limitations by promoting virtualized and disaggregated architectures in which traditional base station functions are distributed across logical nodes, interconnected through fronthaul/midhaul transport, and linked to the core network through backhaul. This paradigm enables flexible functional splits and the consolidation of baseband processing across multiple cells, ultimately "blurring" traditional RAN boundaries and adapting processing topology dynamically based on network conditions and operator policies.

Although this virtualized architecture brings advantages such as reduced CAPEX/OPEX and improved energy efficiency, it also introduces five key challenges: (i) fronthaul capacity and latency constraints amplified by 6G features (massive MIMO, wider bandwidths, higher modulation orders, carrier aggregation), (ii) the need for a new hierarchical control architecture supporting real-time and non-real-time decision making, (iii) dynamic and multi-cell functional split selection under shared computing and transport, (iv) KPI-driven variable latency requirements, and (v) trust and security in open and virtualized environments.

The objective of 6GBLUR is therefore to define an end-to-end architecture with efficient resource management procedures and intelligent control mechanisms for adaptive, disaggregated mobile networks. The resources addressed include radio spectrum, transport capacity, computing and storage, energy consumption, and baseband processing elements. AI/ML-driven procedures are adopted to enable real-time and non-real-time control under O-RAN and NPN environments, validating the resulting advances through specific use cases focused on zero-touch disaggregated 6G networks, fronthaul-aware performance optimization, flexible functional splits, QoS management, digital twins for NPN, and coordinated RAN control strategies.

Within this scope, 6GBLUR-SMART targets: (i) the design and validation of a flexible RAN control architecture capable of operating across all decision-making timescales — from long-term policies to real-time scheduling — and (ii) the development of intelligent E2E resource-management algorithms validated through experimental implementations.

To achieve this, a set of Use Cases and Proofs-of-Concept have been developed to experimentally demonstrate the project's Key Concepts (KCs). The definition and assignment of KCs — first consolidated during the November 2023 technical meeting — guide the work of WP2 and WP3 and form the basis for the final architecture and demonstrators.

This deliverable presents the final Key Concepts together with the PoC-validated results that demonstrate their feasibility and impact. It consolidates the final consortium-agreed architecture, the technical advances achieved within each KC, and the outcome of their experimental validation through the selected PoCs.

The document is organized as follows: Section 1 introduces the scope and motivation of the project; Section 2 presents the final key concepts developed; Sections 3 presents the results obtained in 6GBLUR-SMART's PoCs based on the integration of the different KCs.

1.1 SMART Key Concepts Overview

The eleven key concepts in 6G BLUR-SMART are summarized below. We present all the key concepts that are under study in both work packages (WPs), 2 and 3 to be tested in the framework of WP4 within the framework of defined Use Cases and its associated Proofs of concepts (PoC). All these key concepts are the ones that are aligned with the principal objective of 6G BLUR-SMART project, that is to efficiently manage a 6G RAN architecture, by allowing an intelligent orchestration of key components. From the table below, it is worth mentioning that key concepts reflected in E3 deliverable K1.3 "Dynamic Configuration of topologies/architectures for MH distribution" and K1.5 "Dynamic Configuration of FH for distributed and centralized architectures" have not been fully developed in favour of more extended development of K2.2 "Characterization of the FH and MH transport" and K2.4 "Radio stack optimization", which are lately integrated into UC2PoC1 development. For the sake of traceability, table below reflects the same naming for associated key concepts as in E3.

TABLE 1. SMART KEY CONCEPTS AND THEIR MAPPING TO SMART POCS.

Category	KC	*UC1PoC2*[2]	UC2PoC1	UC3PoC1	UC3PoC2	UC3PoC3
K1. RAN control architecture and orchestration	**K1.1** CU and DU implementation	X	X			X
	K1.2 RAN Orchestration		X			
	K1.4 Micro-orchestration of virtual network functions at DU-RU level				X	
K2. Fronthaul and Midhaul optimization	**K2.1** Development and characterization of FH and MH		X			
	K2.2 Characterization of the FH and MH transport		X			
	K2.3 Flexible functional split and physical layer		X			
	K2.4 Radio Stack optimization		X			
	K2.5 QoS Management for time-critical immersive applications		X			
	K2.6 QoS strategies the transport for multi-service aggregation		X			
K3. AI/ML	**K3.1** AI/ML Workflows in O-RAN					X
	***K3.2** AI/ML Platform*[3]			X		

Also, in Table 1, it is shown the relation between Proof of Concepts and Key Concepts. As it can be seen major key concepts are focused on the optimization of RAN and transport components. Following the main objective of the 6G BLUR-SMART project, the different PoCs that have been developed, focus the efforts on optimizing the performance of a 6G network.

[2] Although UC1PoC2 has been developed at 6GBLUR JOINT project, the PoC has partially based on the K1.1 developed in 6GBLUR SMART project. For more information, please refer to 6GBLUR JOINT E4 deliverable.
[3] Although K3.2 has been developed at 6GBLUR JOINT project, the main demonstration has been shown at 6GBLUR SMART project under the UC3PoC1. For more information, please refer to 6GBLUR JOINT E4 deliverable.

Financiado por la Unión Europea NextGenerationEU
Plan de Recuperación, Transformación y Resiliencia
UNICO I+D
Gobierno de España
cttc Centre Tecnològic de Telecomunicacions de Catalunya

The majority of the key concepts will be present in the UC2 that focuses on characterizing and analysing the feasibility of the proposed architecture with different network topologies under different scenarios (e.g., low, medium, high-capacity traffic) to support the selected deployment options explored in UC2PoC4. This characterization and analysis will also help to find the limits of such network deployments with the aim of expanding theoretical limits which will impact in the network's capital and operational expenditures through enhanced hardware/software elements/procedures achieving further levels of centralization and reducing the amount of human intervention to operate the network, while maintaining the user Quality of Service (QoS).

Also, they are summarized the key concepts related with UC3 that will focus on smart methods for resource optimization of the RAN.

[4] Due to the main concepts worked in 6G BLUR-SMART subproject, this document includes only the analysis of the network deployment of UC1 PoC2. For further details about the analysis of UC1 PoC1, please refer to the 6G BLUR-JOINT E1 document.

Financiado por la Unión Europea NextGenerationEU
Plan de Recuperación, Transformación y Resiliencia
UNICO I+D
GOBIERNO DE ESPAÑA
cttc Centre Tecnològic de Telecomunicacions de Catalunya

2 SMART Key Concepts Procedures/Algorithms and Results

2.1 SMART-K1.1: CU and DU implementation

2.1.1 System Design

Conversely to other more research-oriented key-concepts included within the framework of this project, this one is focused on software implementation and on providing a framework for the assessment of the other key-concepts. Specifically, several new features have been implemented in the CU/DU solution provided by SRS that were driven by the necessities of the 6G BLUR project.

For instance, the E2 interface has been extended with a new Radio Link Control (RLC) E2 agent that allows the collection and exposition of metrics about data volume at the RLC Service Data Units (SDU) level (namely *DRB.RlcSduTransmittedVolumeUL_Filter* and *DRB.RlcSduTransmittedVolumeDL_Filter* - the relevant code can be found in https://github.com/srsran/srsRAN_Project/tree/main/lib/e2/e2sm/e2sm_kpm), which is relevant for the work proposed in UC3PoC3 (see Section 3.2.3). Moreover, the gNB software (both the monolithic solution as well as the one with split CU and DU) now has new configuration options for setting different bind addresses for a number of interfaces (e.g., F1-U and N3), a must-have feature for disaggregated, Helm chart-based RAN deployments as the one developed in JOINT KC 1.2 (see 6GBLUR E4 deliverable) and proposed in UC1PoC2 for the whole mobile network entities (see 6GBLUR E4 deliverable). The most relevant contribution of the 6G BLUR project to the development of the srsRAN CU/DU solution is the support for slicing inside the RAN, which is considered mainly in the scenarios proposed in UC1PoC2.

2.1.2 Implementation

In a few words, the idea behind RAN slicing is to have a Medium Access Control (MAC) scheduler that allocates radio resources based on the network slice the UE belongs to, thus effectively enabling the end-to-end customization of the network depending on the service requirements. The design of the slice-aware scheduler follows the RRM (Radio Resource Management) policy described in [3] Section 4.3.36, and keeps into account the guidelines in [4] Section 8.4.3.6 for the control of the slice-level Physical Resource Block (PRB) quota via the E2 interface (although this capability is not fully functional, a basic demonstration has also been implemented in the form of the *simple_xapp* available at https://github.com/srsran/oran-sc-ric).

Financiado por la Unión Europea NextGenerationEU
Plan de Recuperación, Transformación y Resiliencia
UNICO I+D
GOBIERNO DE ESPAÑA
cttc Centre Tecnològic de Telecomunicacions de Catalunya

More specifically, the new slice-aware scheduler included in the srsRAN software (see the relevant code at https://github.com/srsran/srsRAN_Project/tree/main/lib/scheduler/slicing) works on two different levels. At the inter-slice level, and on a slot basis, the scheduler decides how to distribute PRBs across the active RAN slices. This decision is carried out in accordance with the provided minimum and maximum PRB ratios configured by the user for each RAN slice, while ensuring fairness in PRB distribution when the configuration of the RAN slices leads to them contending for resources. For instance, when two RAN slices have maximum PRB ratio set to 100%, the inter-slice scheduler will allocate the two RAN slices in a round-robin fashion over time. Next, at the intra-slice level, the UE bearers inside each given slice are allocated a fraction of the slice resources. It is important to remark that the SRS MAC scheduler can be configured to apply the intra-slice scheduling policy that fits best with the service requirements of each slice. Moreover, the mapping to the slices is done based on the S-NSSAI (Network Slice Selection Assistance Information) indicated by each UE Data Radio Bearer (DRB). Therefore, a UE with multiple DRBs with different S-NSSAI can be associated with several RAN slices at the same time.

The behaviour of the slice-aware scheduler can be controlled via the srsRAN configuration file. In particular, the user can set how many RAN slices the MAC scheduler will need to manage, the S-NSSAI (SST + SD) associated with each slice, the distribution of PRBs between slices, and the scheduling policy used by each slice (e.g. round-robin, proportional fair, guaranteed bitrate). An example configuration is reported below (only the section relevant for RAN slicing). With this configuration, the srsRAN DU will instantiate two RAN slices with SST=1 and SD equal to 1 and 2, respectively. The slice SST=1, SD=1 will use between 40% and 80% of the BWP RBs on each slot, whereas the slice SST=1, SD=2 will be limited to 20% of the BWP RBs. By default, round-robin scheduling of UEs is used when no policy scheduler is configured for the RAN slice. This default can, however, be overridden with any other of the srsRAN DU supported scheduler policies, such as Proportional Fair as shown below.

```
cell_cfg:
  slicing:
      -
        sst: 1
        sd: 1
        sched_cfg:
          min_prb_policy_ratio: 40
          max_prb_policy_ratio: 80
        policy_sched_cfg:
          pf_sched:
            pf_sched_fairness_coeff: 5.0
```

```
-
  sst: 1
  sd: 2
  sched_cfg:
    min_prb_policy_ratio: 0
    max_prb_policy_ratio: 20
   policy_sched_cfg:
    pf_sched:
      pf_sched_fairness_coeff: 5.0
```

The distribution of UE logical channels across different RAN slices works as follows.

- At startup, independently of the srsRAN DU configuration, two RAN slices are instantiated, called "default Signalling Radio Bearer (SRB) slice" and "default DRB slice," which have no S-NSSAI associated with them. The default SRB slice has the highest priority possible.
- If the UE logical channel corresponds to an SRB, the logical channel is placed in the "default SRB slice."
- If the UE logical channel is a DRB and it either does not have an associated S-NSSAI or its S-NSSAI does not match any of the S-NSSAIs specified in the srsRAN DU slicing configuration (example above), the logical channel is placed in the "default DRB slice."
- If the UE logical channel is a DRB and its S-NSSAI matches one of the specified S-NSSAIs in the srsRAN DU slicing configuration, the logical channel is placed in the respective RAN slice.

The inter-slice scheduling in the srsRAN MAC scheduler works as follows.

- On each slot, slices priorities get recomputed based on their min/max PRB ratios and time elapsed since the slice was scheduled.
- From higher to lower priority, the intra-slice scheduler will call each slice's intra-slice scheduler policy.
- Slices with the same priority are scheduled in a round-robin fashion over different slots to ensure fairness.
- On each slot, once the number of PRBs allocated for a given slice reaches the slice's maximum PRB ratio, the intra-scheduler halts, and the inter-slice scheduler moves on to the next slice candidate.
- If a slice has a minimum PRB ratio larger than zero, it starts with higher priority than other slices that have a minimum PRB ratio of zero.

Financiado por la Unión Europea NextGenerationEU
Plan de Recuperación, Transformación y Resiliencia
UNICO I+D
GOBIERNO DE ESPAÑA
cttc Centre Tecnològic de Telecomunicacions de Catalunya

- Once the number of PRBs allocated for a slice reaches the minimum PRB ratio larger than zero, the intra-slice scheduler halts, and the slice priority gets recomputed as if its minimum PRB ratio was zero. Effectively, this means that the slice priority drops when it fulfils its minimum PRB ratio requirement.

2.1.3 Results

The screenshot in Figure 1 shows an excerpt of the metrics trace of the srsRAN gNB running with the slicing configuration above, with two UEs connected to the slice (SST = 1, SD = 1)—RNTIs 4602 and 4603—and one UE connected to the slice (SST = 1, SD = 2)—RNTI 4601. Bidirectional traffic is generated at each UE with the command "*iperf -c XX.YY.ZZ.WW -t 3600 -b 40M -u -d*", where XX.YY.ZZ.WW is the IP address of the core. The trace is updated every second with three new lines, one per UE. Focusing on the downlink (the uplink follows a similar logic), one can readily see that the UE with RNTI 4601 only achieves a throughput of 9.9 Mbps (brate column), while the other two UEs always achieve 20 Mbps each, which is consistent with the slice configuration specified above. As a term of comparison, Figure 2 reports a trace for the same[5] set-up, but without configuring any slice. We can observe that, without the slice PRB allocation, all users achieve the same throughput, according to the proportional-fair policy.

[5] RNTIs are assigned by the gNB, without control from the user.

```
          |--------------------DL---------------------|------------------------UL----------------------------
pci rnti  | cqi  ri  mcs  brate   ok  nok  (%)  dl_bs | pusch  rsrp  mcs  brate  ok  nok  (%)    bsr    ta  phr
  1 4601  |  15 1.0   28   9.9M 1550    0   0%  6.16M |  54.6  -0.6   28   2.2M 300    0   0%   300k   93n   23
  1 4602  |  15 1.0   28    20M 1550    0   0%  6.14M |  54.0  -0.6   28   3.4M 400    0   0%   300k   52n   23
  1 4603  |  15 1.0   28    20M 1550    0   0%  6.14M |  54.0  -0.6   28   3.4M 400    0   0%   300k   52n   23
  1 4601  |  15 1.0   28   9.9M 1550    0   0%  6.16M |  54.7  -0.6   28   2.2M 300    0   0%   300k   61n   23
  1 4602  |  15 1.0   28    20M 1550    0   0%  6.12M |  54.0  -0.6   28   3.4M 400    0   0%   300k   50n   23
  1 4603  |  15 1.0   28    20M 1550    0   0%  6.16M |  54.0  -0.6   28   3.4M 400    0   0%   300k   50n   23
  1 4601  |  15 1.0   28   9.9M 1550    0   0%  6.17M |  55.0  -0.6   28   2.2M 300    0   0%   300k   36n   23
  1 4602  |  15 1.0   28    20M 1550    0   0%  6.13M |  54.5  -0.6   28   3.4M 400    0   0%   300k   47n   23
  1 4603  |  15 1.0   28    20M 1550    0   0%  6.13M |  54.5  -0.6   28   3.4M 400    0   0%   300k   47n   23
  1 4601  |  15 1.0   28   9.9M 1550    0   0%  6.16M |  55.9  -0.6   28   2.2M 300    0   0%   300k   48n   23
  1 4602  |  15 1.0   28    20M 1550    0   0%  6.14M |  55.1  -0.6   28   3.4M 400    0   0%   300k   46n   23
  1 4603  |  15 1.0   28    20M 1550    0   0%  6.14M |  55.1  -0.6   28   3.4M 400    0   0%   300k   46n   23
  1 4601  |  15 1.0   28   9.9M 1550    0   0%  6.16M |  55.3  -0.6   28   2.2M 300    0   0%   300k   79n   23
  1 4602  |  15 1.0   28    20M 1550    0   0%  6.13M |  54.4  -0.6   28   3.4M 400    0   0%   300k   53n   23
  1 4603  |  15 1.0   28    20M 1550    0   0%  6.15M |  54.4  -0.6   28   3.4M 400    0   0%   300k   53n   23
  1 4601  |  15 1.0   28   9.9M 1550    0   0%  6.17M |  54.5  -0.6   28   2.2M 300    0   0%   300k   63n   23
  1 4602  |  15 1.0   28    20M 1550    0   0%  6.13M |  54.3  -0.6   28   3.4M 400    0   0%   300k   52n   23
  1 4603  |  15 1.0   28    20M 1550    0   0%  6.13M |  54.3  -0.6   28   3.4M 400    0   0%   300k   52n   23
  1 4601  |  15 1.0   28   9.9M 1550    0   0%  6.16M |  54.3  -0.6   28   2.2M 300    0   0%   300k   42n   23
  1 4602  |  15 1.0   28    20M 1550    0   0%  6.14M |  54.0  -0.6   28   3.4M 400    0   0%   300k   47n   23
  1 4603  |  15 1.0   28    20M 1550    0   0%  6.14M |  54.0  -0.6   28   3.4M 400    0   0%   300k   47n   23
  1 4601  |  15 1.0   28   9.9M 1550    0   0%  6.16M |  54.7  -0.6   28   2.2M 300    0   0%   300k   84n   23
  1 4602  |  15 1.0   28    20M 1550    0   0%  6.12M |  54.3  -0.6   28   3.4M 400    0   0%   300k   48n   23
  1 4603  |  15 1.0   28    20M 1550    0   0%  6.15M |  54.3  -0.6   28   3.4M 400    0   0%   300k   47n   23
  1 4601  |  15 1.0   28   9.9M 1550    0   0%  6.17M |  54.6  -0.6   28   2.2M 300    0   0%   300k   64n   23
  1 4602  |  15 1.0   28    20M 1550    0   0%  6.13M |  53.9  -0.6   28   3.4M 400    0   0%   300k   49n   23
  1 4603  |  15 1.0   28    20M 1550    0   0%  6.13M |  53.9  -0.6   28   3.4M 400    0   0%   300k   49n   23
  1 4601  |  15 1.0   28   9.9M 1550    0   0%  6.16M |  55.2  -0.6   28   2.2M 300    0   0%   300k   53n   23
  1 4602  |  15 1.0   28    20M 1550    0   0%  6.14M |  54.5  -0.6   28   3.4M 400    0   0%   300k   47n   23
  1 4603  |  15 1.0   28    20M 1550    0   0%  6.14M |  54.5  -0.6   28   3.4M 400    0   0%   300k   48n   23
  1 4601  |  15 1.0   28   9.9M 1550    0   0%  6.16M |  54.9  -0.6   28   2.2M 300    0   0%   300k   87n   23
  1 4602  |  15 1.0   28    20M 1550    0   0%  6.12M |  54.1  -0.6   28   3.4M 400    0   0%   300k   54n   23
  1 4603  |  15 1.0   28    20M 1550    0   0%  6.15M |  54.1  -0.6   28   3.4M 400    0   0%   300k   54n   23
```

FIGURE 1. SRSRAN TRACE FOR THE RAN-SLICING CASE: RNTI 4602 AND RNTI 4603 CONNECTED TO SLICE (SST=1, SD=1), RNTI 4601 CONNECTED TO SLICE (SST=1, SD=2).

Financiado por la Unión Europea NextGenerationEU
Plan de Recuperación, Transformación y Resiliencia
UNICO I+D
GOBIERNO DE ESPAÑA
cttc Centre Tecnològic de Telecomunicacions de Catalunya

```
          |-------------------DL--------------------|------------------------UL----------------------------
pci rnti  | cqi  ri  mcs  brate   ok  nok  (%)  dl_bs | pusch  rsrp  mcs  brate   ok  nok  (%)   bsr    ta  phr
  1 4602 |  15 1.0   28    17M  534    0   0%  6.16M |  54.3  -0.6   28   3.0M  167    0   0%  300k   63n   23
  1 4603 |  15 1.0   28    17M  533    0   0%  6.15M |  54.3  -0.6   28   3.0M  166    0   0%  300k   63n   23
  1 4604 |  15 1.0   28    17M  533    0   0%  6.13M |  54.3  -0.6   28   3.0M  168    0   0%  300k   63n   23
  1 4602 |  15 1.0   28    17M  533    0   0%  6.13M |  54.2  -0.6   28   3.0M  167    0   0%  300k   57n   23
  1 4603 |  15 1.0   28    17M  534    0   0%  6.16M |  54.2  -0.6   28   3.0M  168    0   0%  300k   56n   23
  1 4604 |  15 1.0   28    17M  533    0   0%  6.15M |  54.2  -0.6   28   3.0M  167    0   0%  300k   57n   23
  1 4602 |  15 1.0   28    17M  533    0   0%  6.15M |  54.3  -0.6   28   3.0M  166    0   0%  300k   58n   23
  1 4603 |  15 1.0   28    17M  533    0   0%  6.13M |  54.3  -0.6   28   3.0M  169    0   0%  300k   58n   23
  1 4604 |  15 1.0   28    17M  534    0   0%  6.14M |  54.3  -0.6   28   3.0M  168    0   0%  300k   58n   23
  1 4602 |  15 1.0   28    17M  534    0   0%  6.15M |  54.0  -0.6   28   3.0M  167    0   0%  300k   60n   23
  1 4603 |  15 1.0   28    17M  533    0   0%  6.15M |  54.0  -0.6   28   3.0M  166    0   0%  300k   60n   23
  1 4604 |  15 1.0   28    17M  533    0   0%  6.13M |  54.0  -0.6   28   3.0M  168    0   0%  300k   60n   23
  1 4602 |  15 1.0   28    17M  533    0   0%  6.13M |  54.7  -0.6   28   3.0M  167    0   0%  300k   60n   23
  1 4603 |  15 1.0   28    17M  534    0   0%  6.16M |  54.7  -0.6   28   3.0M  167    0   0%  300k   60n   23
  1 4604 |  15 1.0   28    17M  533    0   0%  6.15M |  54.8  -0.6   28   3.0M  167    0   0%  300k   60n   23
  1 4602 |  15 1.0   28    17M  533    0   0%  6.15M |  54.0  -0.6   28   3.0M  166    0   0%  300k   69n   23
  1 4603 |  15 1.0   28    17M  533    0   0%  6.13M |  54.0  -0.6   28   3.0M  167    0   0%  300k   69n   23
  1 4604 |  15 1.0   28    17M  534    0   0%  6.15M |  54.0  -0.6   28   3.0M  167    0   0%  300k   69n   23
  1 4602 |  15 1.0   28    17M  534    0   0%  6.16M |  53.8  -0.6   28   3.0M  167    0   0%  300k   67n   23
  1 4603 |  15 1.0   28    17M  533    0   0%  6.15M |  53.8  -0.6   28   3.0M  167    0   0%  300k   67n   23
  1 4604 |  15 1.0   28    17M  533    0   0%  6.13M |  53.8  -0.6   28   3.0M  167    0   0%  300k   68n   23
  1 4602 |  15 1.0   28    17M  533    0   0%  6.13M |  54.3  -0.6   28   3.0M  167    0   0%  300k   55n   23
  1 4603 |  15 1.0   28    17M  534    0   0%  6.15M |  54.3  -0.6   28   3.0M  168    0   0%  300k   55n   23
  1 4604 |  15 1.0   28    17M  533    0   0%  6.15M |  54.3  -0.6   28   3.0M  167    0   0%  300k   55n   23
  1 4602 |  15 1.0   28    17M  533    0   0%  6.15M |  54.7  -0.6   28   3.0M  166    0   0%  300k   62n   23
  1 4603 |  15 1.0   28    17M  533    0   0%  6.13M |  54.7  -0.6   28   3.0M  169    0   0%  300k   62n   23
  1 4604 |  15 1.0   28    17M  534    0   0%  6.14M |  54.7  -0.6   28   3.0M  167    0   0%  300k   62n   23
  1 4602 |  15 1.0   28    17M  534    0   0%  6.15M |  53.9  -0.6   28   3.0M  167    0   0%  300k   64n   23
  1 4603 |  15 1.0   28    17M  533    0   0%  6.15M |  53.9  -0.6   28   3.0M  166    0   0%  300k   65n   23
  1 4604 |  15 1.0   28    17M  533    0   0%  6.13M |  53.9  -0.6   28   3.0M  167    0   0%  300k   65n   23
  1 4602 |  15 1.0   28    17M  533    0   0%  6.13M |  54.3  -0.6   28   3.0M  167    0   0%  300k   67n   23
  1 4603 |  15 1.0   28    17M  534    0   0%  6.16M |  54.3  -0.6   28   3.0M  168    0   0%  300k   67n   23
  1 4604 |  15 1.0   28    17M  533    0   0%  6.15M |  54.3  -0.6   28   3.0M  166    0   0%  300k   66n   23
```

FIGURE 2. SRSRAN TRACE FOR THE NO-SLICING CASE: ALL RNTIS GET THE SAME DATA RATE.

2.1.4 Conclusions

During the development of the 6G BLUR project several features have been added into the implementation of the CU/DU split of srsRAN software, which are supporting the work developed in other KCs and PoCs. One of the main features is the inclusion of RAN slicing, thus allowing the possibility of creating E2E slices as those proposed in UC1PoC2 (see 6GBLUR JOINT E4 deliverable). RAN slicing relies on the inclusion of a MAC scheduler working at two levels (i.e., intra-slice and inter-slice). The behaviour of this scheduler is easily configurable via the configuration file used to launch the CU/DU process and performed experiments show the different performance experienced by UEs according to the slice they belong to and the configuration given to this slice. A user can configure the number of slices, and for each slice define the PRB distribution and the policy used for scheduling the different UEs belonging to this slice. The implementation considers two default slices (i.e., the SRB and the DRB slice), so the logical changes of the UE are assigned to these default slices if the configuration does not match any of the slices defined in the configuration file.

Financiado por la Unión Europea NextGenerationEU
Plan de Recuperación, Transformación y Resiliencia
UNICO I+D
GOBIERNO DE ESPAÑA
cttc Centre Tecnològic de Telecomunicacions de Catalunya

2.2 SMART-K1.2: RAN Orchestration

2.2.1 System Design

O-RAN philosophy focuses on promoting innovation and flexibility by building future mobile networks centred around interoperability and intelligence [6], through the adoption of the 3GPP disaggregation paradigm [7] and the introduction of RAN Intelligent Controllers (RICs). 5G and 6G networks are highly envisioned to benefit from the potential of the O-RAN framework to provide improved coverage, reduced Capital Expenditures (CAPEX) and Operational Expenditures (OPEX), better energy efficiency, rapid innovation and network resiliency and support novel use cases and a variety of emerging applications with stringent requirements, such that of eXtended Reality (XR) and Cloud Gaming (CG). These applications require low latency, high data rates and high reliability. However, although O-RAN is considered as an enabler for XR and CG [8], since it is expected to provide improved network performance and meet these requirements, the split of the RAN introduces new challenges related to the links among the different functional blocks, such as the Open Fronthaul that interconnects the DU and the RU. Therefore, the RAN functions and the transport (and particularly the fronthaul) need to be orchestrated jointly. The Open Fronthaul can impact significantly the air-interface performance and is required to meet strict capacity and latency requirements. Moreover, these links can be shared among various RUs to reduce deployment costs and can constitute therefore a bottleneck, especially when bandwidth demanding and time-critical applications, such as the XR and CG, are served.

To tackle the constraints imposed by the fronthaul, in the work presented in [9], we implement and extend, using the ns-3 5G-LENA simulator, various fronthaul-aware scheduling approaches that optimize the fronthaul resource utilization [1],[10], and we study the impact they have on applications generating downlink XR and CG traffic. In particular, we have implemented a Fronthaul Control mechanism following a distributed architecture, as shown in Figure 3, where different methods can be applied, described in the following.

Financiado por la Unión Europea NextGenerationEU
Plan de Recuperación, Transformación y Resiliencia
UNICO I+D
GOBIERNO DE ESPAÑA
cttc Centre Tecnològic de Telecomunicacions de Catalunya

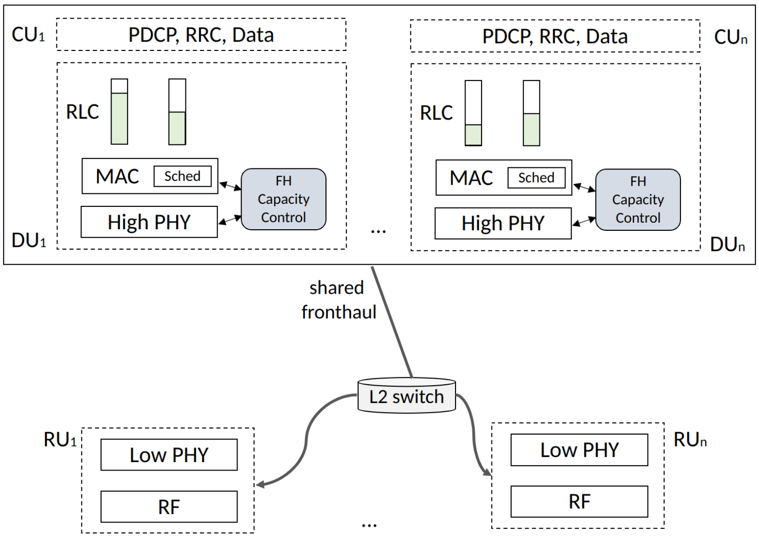


FIGURE 3. FRONTHAUL CONTROL ARCHITECTURE

A. Traditional approaches

The methods described in the following are considered to be the most simplistic and typical methods to handle the data in case that the fronthaul capacity is limited. Such methods present low complexity in their implementation since they do not interfere with the MAC scheduler operation [1].

• **Dropping**: Dropping of the data takes place in the high-PHY layer, after the scheduling process has been completed, i.e. after the MAC layer has performed the allocation of Resource Blocks (RBs) to the active UEs, and the MAC Packet Data Unit (PDU) has been prepared and sent from the MAC to the high-PHY. The latter interacts with the Fronthaul Control to indicate which PDUs (if any) shall be dropped. The Fronthaul Control takes the decision, accounting for the available capacity and the previous allocations that have not been dropped. Let us notice that we employ a random shuffling of the PDU list, in order to promote diversity and to reduce the predictability in the dropped PDU patterns.

• **Postponing**: Postponing of the data is implemented in the MAC layer, once the scheduling process has taken place. The MAC layer consults the Fronthaul Control if the allocation can be served. Similarly to the Dropping approach, the Fronthaul Control takes the decision, considering the available capacity and the previous allocations. In case that the allocation cannot be served, the MAC layer discards the scheduling decision. However, in such case the data remains in the RLC or the HARQ buffer for a later transmission, thus transmission is postponed and consequently, the buffering delay is increased.

B. Optimized approaches

The following two methods (Optimize MCS and Optimize RBs), take into consideration the limitations imposed by the fronthaul capacity and the active UEs of the cell, and perform a dynamic adjustment of the RBs and the Modulation and Coding Scheme (MCS) assigned to each (active) UE [1], respectively. Such methods require some higher complexity, since they interfere with the MAC scheduling operation, however, as it will be demonstrated throughout this report, they allow for improved performances to be achieved. Let us notice that with the current work we have provided a basic framework for Fronthaul Control that can be easily extended to interact with xApps, empowering the decisions with intelligence, as well as to include other possible optimized methods. As such, as an extension of the presented work, we have defined a new Fronthaul Control method named "AdjustRBs" that jointly manages both air and fronthaul resource constraints [11], in order to avoid conflicts between the MAC scheduling schemes and the Fronthaul Control methods.

• **Optimize MCS**: Optimization of the MCS involves limiting the assigned MCS, and consequently the modulation order (modulation compression), in case that the available fronthaul capacity is constraint. Once the scheduling process has finished, the MAC layer interacts with the Fronthaul Control, which provides the maximum MCS that can be assigned for each of the scheduled UEs (UEs that got an allocation). To provide user fairness, a random shuffling of the UEs is also applied in this case. Moreover, this method is combined with the Postponing, in order to ensure that the fronthaul capacity is not exceeded.

• **Optimize RBs**: Optimization of the RBs happens during the scheduling process, through interaction of the MAC scheduler with the Fronthaul Control. The latter indicates to the scheduler the maximum number of RBs that will allow for maximization of the minimum achievable user throughput across all active users, taking into account the capacity limitation. As such, each time that the scheduler assigns 1 RBG to a UE, it asks the Fronthaul Control whether it has reached the maximum number of RBGs. Similarly to the Optimize MCS, this method is also combined with the Postponing.

• **Adjust RBs**: During the resource allocation process, this method verifies whether the total resources assigned to a user fit within the fronthaul capacity. Therefore, AdjustRBs integrates FH awareness directly into the MAC scheduling process, while at the same time respecting the scheduling decisions.

The current Fronthaul Control work has been focused on the downlink data (CU-DU-RU to UE). Differently from the downlink, where the majority of the fronthaul data comes from the PDSCH (downlink data), in the uplink, a significant portion of the fronthaul data comes from the uplink Sounding Reference Signals (SRSs). In this context, the work in [10] has proposed and evaluated various fronthaul-aware handling methods for SRSs. Let us note however that OptimizeMcs, OptimizeRbs, and Postpone methods could also be applied at the gNB MAC during the uplink data

Financiado por la Unión Europea NextGenerationEU
Plan de Recuperación, Transformación y Resiliencia
UNICO I+D
GOBIERNO DE ESPAÑA
cttc Centre Tecnològic de Telecomunicacions de Catalunya

scheduling. Dropping could also be applied for Fronthaul Control, although modifications would be necessary to interface with the low-PHY layer at the gNB.

2.2.2 Implementation

The Fronthaul Control has been implemented in the open-source 5G-LENA simulator and allows the simulation of a limited-capacity fronthaul link based on the available fronthaul capacity in the downlink direction [9]. The C++ class *NrFhControl* implements all the logic of the Fronthaul Control and when it is activated, it creates an instance of the *NrFhControl* per cell. *NrFhControl* gets as inputs the available fronthaul capacity, the fronthaul control method to be applied (i.e., Dropping, Postponing, OptimizeMcs and OptimizeRBs) that will restrict user allocations, and the dynamic overhead to implement modulation compression. The latter accounts for the signalling overhead in bits send through the fronthaul to each DU to indicate the modulation order for each UE. These parameters can be configured through the user simulation. Let us notice that we assume the 7.2x functional split, as adopted in O-RAN.

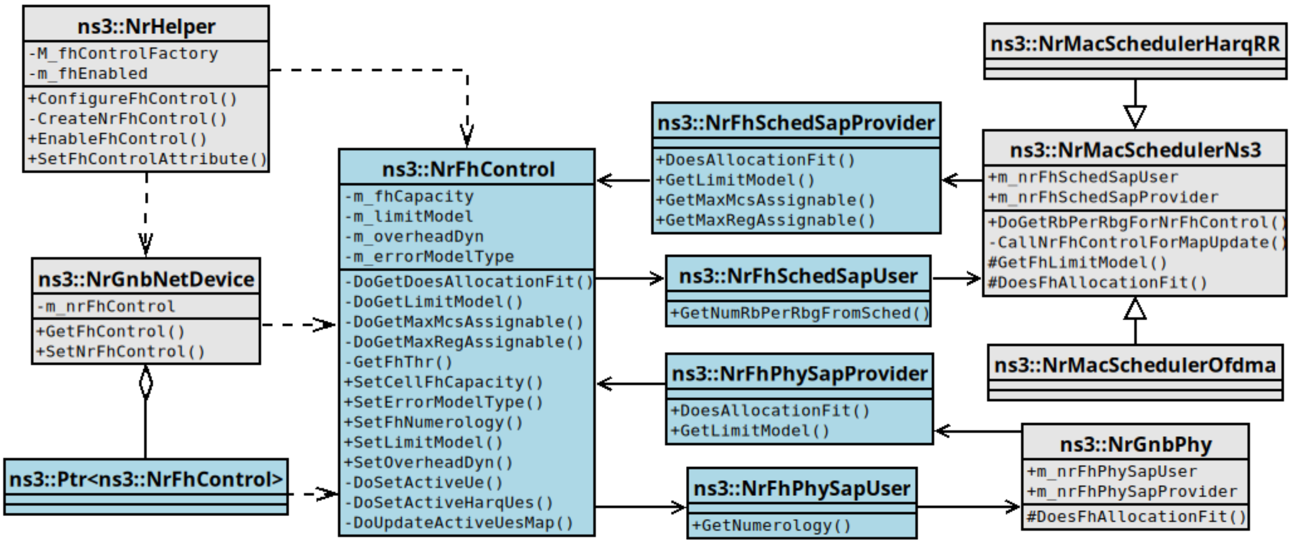


FIGURE 4. UML DIAGRAM OF THE IMPLEMENTED FRONTHAUL CONTROL IN 5G-LENA

Figure 4, shows the simplified UML diagram where the newly implemented classes are depicted in blue colour, while in grey colour are classes already provided by 5G-LENA that have been extended to support the Fronthaul Control. For instance, the NrHelper is a helper provided by 5G-LENA and is designed to simplify the creation of all the necessary components and actions in order to perform a simulation. In this implementation, it is responsible for the creation of the *NrFhControl* instance. The *NrGnbNetDevice* class is extended to hold a pointer to the *NrFhControl* to pass some important configurations of the cell, such as the numerology and the error model. The MAC and high-PHY (classes *NrMacSchedulerHarqRR*, *NrMacSchedulerNs3*, *NrMacSchedulerOfdma* and the *NrGnbPhy*) have been extended to interact with the *NrFhControl* class and limit allocations in case it is instructed by the Fronthaul Control.

For the bidirectional exchange of information between the *NrFhControl* with the MAC and high-PHY layers, a set of SAP interfaces have been implemented, as shown the Figure. Using these interfaces, the *NrFhControl communicates with the MAC to* keep track of the active UEs with new data in their RLC queues and the active UEs with HARQ data. For instance, once a BSR is received, the MAC calls the *NrFhControl*, (through the *NrFhSchedSapProvider* interface), to store the UE (for which the BSR has been received) in a map containing the active UEs along with the amount of bytes of each of the UEs. For the case of active HARQ UEs, once the scheduling process is initiated, the MAC calculates the active HARQ UEs and communicates this list to the *NrFhControl*. Finally, when the scheduling process is finalized, the MAC calls the *NrFhControl* to update the map of the active UEs and the bytes stored based on the performed scheduling decisions.

The limiting process to be applied depends on the method being employed. For the case of Dropping, once the MAC PDUs have been prepared and passed from the MAC layer to the high-PHY layer, the latter calls the *NrFhControl* to indicate which PDUs shall be dropped, taking into account the available fronthaul capacity and the previous allocations that have not been dropped. To address the issue of recurring PDU drops, we employ a random shuffling of the PDU list, thus promoting diversity and reducing predictability in the dropped PDU patterns. It is worth noting that even though the MAC has already inserted the PDU in the HARQ buffer list, the fact that no data is sent will result in no HARQ feedback reception, and consequently, the saved PDU will be discarded once the HARQ timer will expire.

When Postponing is applied, the MAC layer calls the *NrFhControl* to indicate which allocations cannot be served, based on the available fronthaul capacity and the previous allocations. At this point, it is worth highlighting that the 5G-LENA MAC scheduler is designed to schedule first any UEs with HARQ data and afterward UEs with new data (so that retransmissions will always be prioritized). As such, the *NrFhControl* is called in both cases to indicate whether the scheduling decision can be included. In case it cannot be included, the allocation is erased, however the data still stays in the HARQ buffer list (for retransmission data) or RLC buffers (for new data) for later retransmission, as such the transmission of the data is postponed.

For the other two methods, OptimizeMcs and OptimizeRBs, the functionality of the scheduling process is modified to limit the MCS and the number of allocated RBs, respectively. In particular, for the case of OptimizeMcs, once the RBGs have been assigned to the UEs, the scheduler calls the *NrFhControl* to provide the maximum MCS that can be assigned to each of the scheduled UEs. A random shuffling of the UEs is also performed in this case to ensure fairness. For the OptimizeRBs case, each time that the scheduler assigns one RBG to a UE, it checks through the *NrFhControl* if it has reached the maximum number of RBGs. If yes, then this UE is skipped. Both cases are then combined with the Postponing method to reassure that the final allocations will not exceed the available fronthaul capacity.

Finally, let us notice that we have implemented a new user scenario script that allows for simulations of single- or multi-cells (hexagonal deployment) with variable number of UEs with XR traffic per cell and where a variety of parameters can be configured, including enabling Fronthaul Control and selecting the desired method to be applied. The impact of the fronthaul limitations on applications generating XR and CG traffic, and the effectiveness of Fronthaul Control strategies to address these limitations, have been evaluated using this scenario script with a configuration based on 3GPP specifications [12], where we study an Indoor scenario considering a single-cell topology with12 UEs randomly deployed within a disc area with 20 m radius. Out of the 12 UEs, 3 UEs generate AR traffic of 3 streams (Video/Audio/Pose), 3 UEs carry VR traffic (consisting of 1 stream), 3 UEs play CG and 3 UEs generate VoIP traffic with payload size of 40 bytes, all in downlink direction. The detailed parameter configuration used for the evaluation is shown in Table 2.

TABLE 2. EVALUATION SCENARIO PARAMETERS

Scenario Parameters	Value
Carrier Frequency	30 GHz
Bandwidth	400 MHz
Numerology	3
TDD pattern	DDDDU
BS transmit power	30 dBm
UE transmit power	23 dBm
BS antenna height	3 m
UE antenna height	1.5 m
Propagation model	3GPP Indoor Hotspot Open Office TR 38.901
MIMO scheme	SU-MIMO
BS antenna array	2T: 16 × 8 dual-polarized antenna elements (3GPP radiation), 2 ports, (M,N,P,Mg,Ng;Mp,Np) = (16,8,2,1,1;1,1)
BS horizontal/vertical spacing	0.5/0.5 Lambda
UE antenna array	2R: 1 × 4 dual-polarized antenna elements (isotropic radiation), 2 ports, (M,N,P,Mg,Ng;Mp,Np) = (1,4,2,1,1;1,1)

Financiado por la Unión Europea NextGenerationEU
Plan de Recuperación, Transformación y Resiliencia
UNICO I+D
GOBIERNO DE ESPAÑA
cttc Centre Tecnològic de Telecomunicacions de Catalunya

UE horizontal/vertical spacing	0.5/0.5 Lambda
BS noise figure	7dB
UE noise figure	13 dB
Noise power spectral density	−174 dBm/Hz
Scheduler	SU-MIMO PF
Min/Max BS-UE distance	2 m/20 m
Shadowing	Disabled
Fading and channel condition updates	No updates
Channel condition	LoS
Target BLER	10%
HARQ Retx	Enabled
DL AR video data rate/FPS	20 Mbps/60 Fps
DL VR video data rate/FPS	45 Mbps/120 Fps
DL CG video data rate/FPS	30 Mbps/60 Fps
PDCP discarding timer	100 ms for 5QI = 1 (VoIP), 10 ms for 5QI = 80 (XR)
RLC reordering window Timer	10 ms

2.2.3 Results

The analysis of the impact of the fronthaul limitations on applications generating time-critical XR and CG traffic, and the evaluation of the effectiveness of Fronthaul Control strategies to address these limitations, has been carried out using the above presented user script and the configuration of Table 2. Two states for the fronthaul link are considered:

1. **Non-limited fronthaul (100 Gbps)**:
   a. Represents an ideal case where the fronthaul link is not constrained for our system setup, set to 100 Gbps. Such case is equivalent to a scenario where the Fronthaul Control is disabled and therefore, no limitations on the transmissions are applied.
2. **Constrained fronthaul (2 Gbps)**:
   a. The available capacity is set to 2 Gbps. It simulates a stressed scenario where multiple RUs share the fronthaul link and/or a high volume of traffic passes through it, imposing strict capacity limits.

Financiado por la Unión Europea NextGenerationEU
Plan de Recuperación, Transformación y Resiliencia
UNICO I+D
GOBIERNO DE ESPAÑA
cttc Centre Tecnològic de Telecomunicacions de Catalunya

We perform the comparative analysis between the two states by studying the variations in the achieved end-to-end user throughput and end-to-end delay, as well as the average number of RBs used in each slot.

TABLE 3. E2E AVERAGE THROUGHPUT (MBPS) FOR 100 GBPS

	Dropping	Postponing	Opt_MCS	Opt_RBs
AR video	19.99	19.99	19.99	19.99
AR pose	0.26	0.26	0.26	0.26
AR audio	0.81	0.81	0.81	0.81
VR	44.93	44.93	44.93	44.93
CG	30.05	30.05	30.05	30.05
VoIP	0.0081	0.0081	0.0081	0.0081

TABLE 4. E2E AVERAGE THROUGHPUT (MBPS) FOR 2 GBPS

	Dropping	Postponing	Opt_MCS	Opt_RBs
AR video	0.98	19.97	19.99	19.99
AR pose	0.25	0.26	0.26	0.26
AR audio	0.801	0.81	0.81	0.81
VR	1.49	44.93	44.93	44.93
CG	0.06	30.01	30.05	30.05
VoIP	0.0081	0.0081	0.0081	0.0081

Table 3 and Table 4, present the average achieved end-to-end user throughput (in Mbps) for each of the employed Fronthaul Control methods and for each traffic type for fronthaul capacities of 100 Gbps and 2 Gbps, respectively. As it can be observed, the achieved throughout is significantly impacted only for the case of Dropping when considering the highly limited fronthaul capacity (i.e. 2 Gbps).

Figure 5 and Figure 6 present the Cumulative Distribution Function (CDF) of the end-to-end delay (in ms) for the different types of traffic and for each method for the capacity of 100 Gbps and 2 Gbps, respectively. From these figures, it can be seen that for the case of 100 Gbps, all methods perform similarly, validating the correct operation of the of methods since no restrictions are applied in the allocations. On the other hand, when studying the case that the fronthaul capacity is limited, it can be seen that both Dropping and Postponing present increased delays for almost all types of traffic. Such increase can be quite critical for immersive applications, since it can result in service disruption and/or motion sickness. In contrast, the optimized methods (i.e. OptimizeMCS and OptimizeRBs), although they present a small increase in the delay as it is expected in such limited situations, they still allow for delays that fall into the accepted limits for these types of traffic (e.g. under 10 ms for XR), demonstrating the need for such methods when the fronthaul link is constrained.

Financiado por la Unión Europea NextGenerationEU
Plan de Recuperación, Transformación y Resiliencia
UNICO I+D
GOBIERNO DE ESPAÑA
cttc Centre Tecnològic de Telecomunicacions de Catalunya

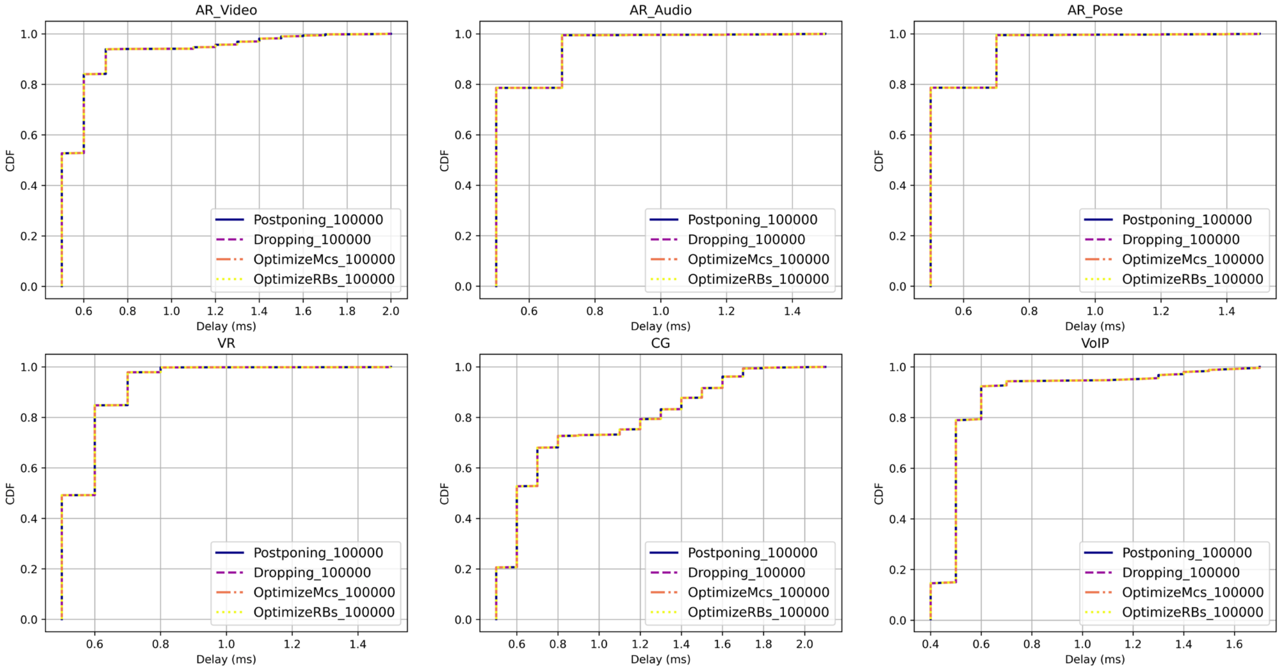


FIGURE 5. E2E DELAY FOR 100 GBPS

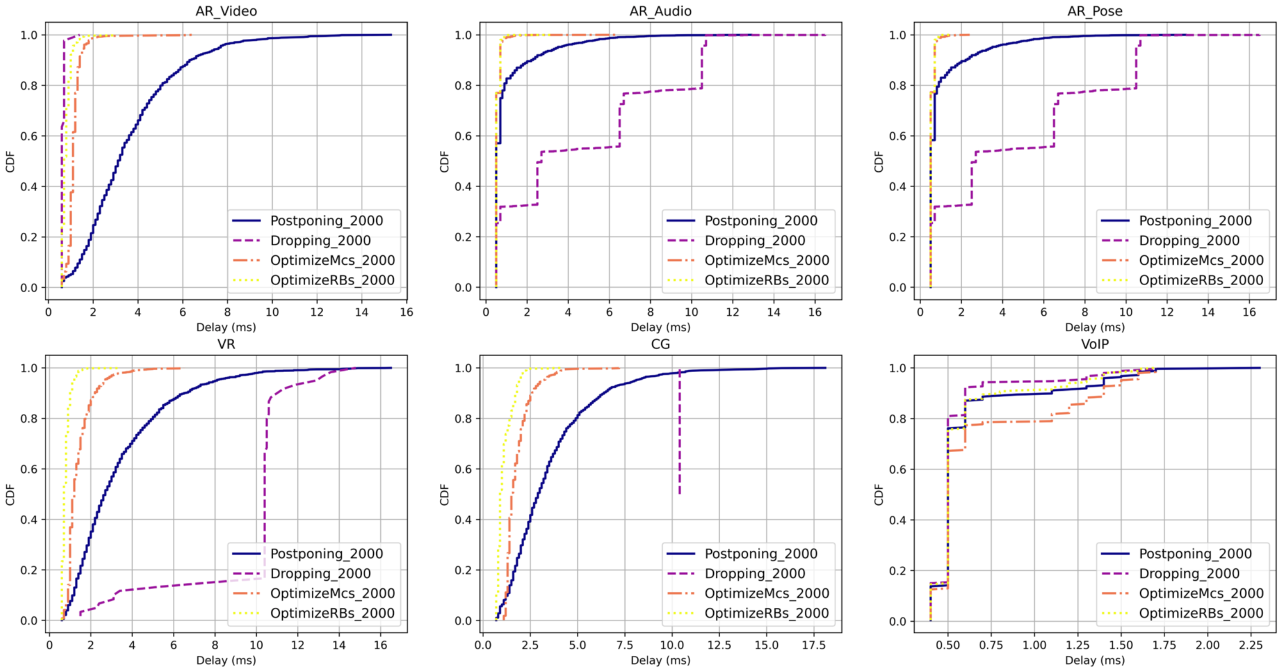


FIGURE 6. E2E DELAY FOR 2 GBPS

Financiado por la Unión Europea NextGenerationEU
Plan de Recuperación, Transformación y Resiliencia
UNICO I+D
GOBIERNO DE ESPAÑA
cttc Centre Tecnològic de Telecomunicacions de Catalunya

Let us notice that for the case of Dropping, CG is the most affected traffic type due to the fact that it generates packets of higher size, resulting in higher chances for the Fronthaul Control to indicate the high-PHY layer to drop these packets.

Finally, Figure 7 and Figure 8 present the average number of RBs assigned in each slot. From the figures it can be seen that when the fronthaul capacity is not limited (100 Mbps), all methods perform a very similar allocation of resources. However, when the capacity is limited to 2 Gbps, it can be notably observed that Dropping allocated the lowest number of resources and OptimizeMCS the highest. Dropping results in dropping all the allocations that exceed the maximum number of RBs that can be assigned as restricted by the fronthaul link, while Postponing presents a denser allocation since it postpones the allocations for a later moment. For the case of Optimize RBs, the allocations are adjusted to fit to the limited fronthaul link. For the case of OptimizeMCS this highest allocation of resources happens due to the fact that initially the scheduler assigns the resources without considering any limitations (this is why the resemblance with the allocation of 100 Gbps), and afterward it interacts with the Fronthaul Control in order to adapt the MCS (if needed). As such, the number of assigned RBs does not change, however what changes is the Transport Block Size (TBS) transmitted, thus the amount of data. Although such behaviour does not bring crucial disadvantages and despite the fact that OptimizeMCS has lower implementation complexity compared to the OptimizeRBs, it can lead to higher power consumption (with the same performance), factor that is critical for devices like VR headsets where battery life is paramount.

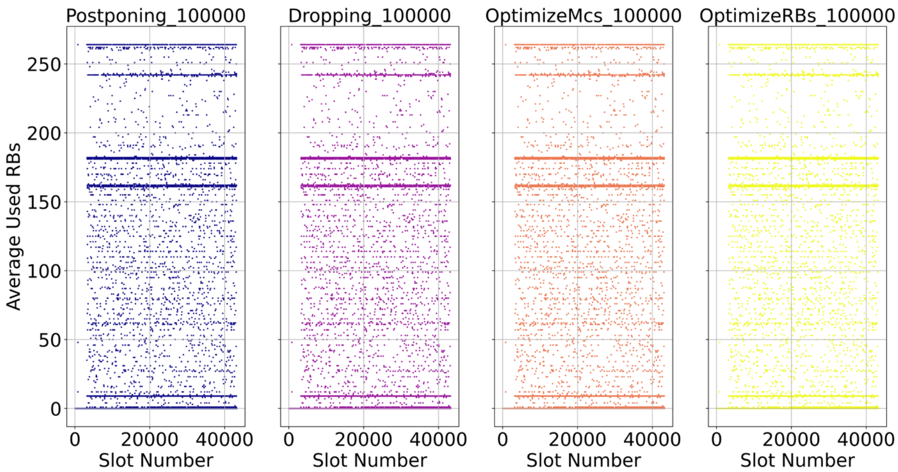


FIGURE 7. RB ALLOCATION FOR 100 GBPS

Financiado por la Unión Europea NextGenerationEU
Plan de Recuperación, Transformación y Resiliencia
UNICO I+D
GOBIERNO DE ESPAÑA
cttc Centre Tecnològic de Telecomunicacions de Catalunya

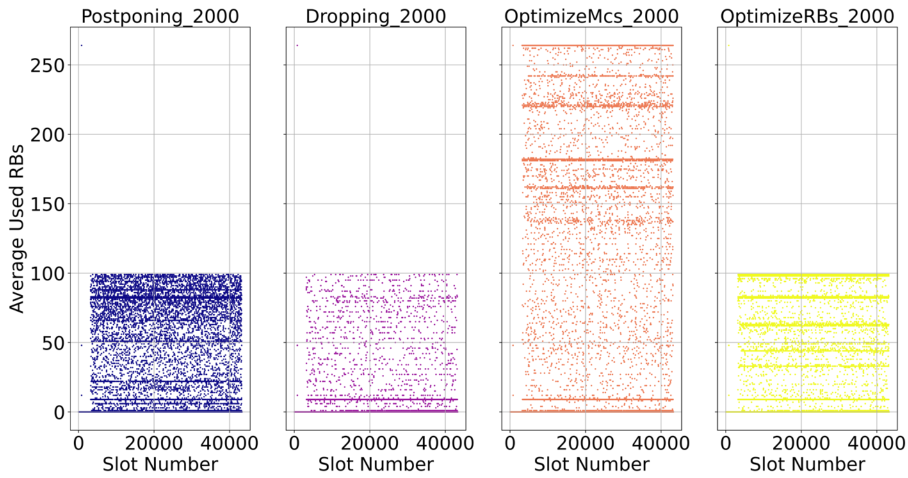


FIGURE 8. RB ALLOCATION FOR 2 GBPS

2.2.4 Conclusions

This work provides a valuable groundwork evaluation of the impact that the Open Fronthaul limitations can have on networks employing XR and CG traffic which are characterized by stringent latency and data rate requirements. Moreover, to address these challenges, we propose a distributed architecture for Fronthaul Control, and we implement and extend various Fronthaul Control methods with different implementation complexities, using the open-source ns-3 5G-LENA simulator. Through simulations we have demonstrated that employing such mechanisms is crucial in order to guarantee an optimal usage of the fronthaul resources. In particular, it has been shown that optimized methods outperform simplistic approaches used in typical networks. Moreover, it is shown that depending on the needs of each application, the choice of mechanism to be applied can vary. Finally, with this work we provide a baseline for future evaluations and extensions to be performed, including further analysis using more advanced scheduling policies and additional optimized methods.

2.3 SMART-K1.4: Micro-orchestration of virtual network functions at DU-RU level

In this section we present a framework for the adaptive offloading of low-PHY Radio Unit (RU) functions (e.g., Fast Fourier Transformation (FFT), Inverse FFT (iFFT), channel estimation, Physical Random Access Channel (PRACH)) between the Programmable Logic (PL) and the Application Processing Unit (APU) in AMD Radio Frequency System-on-Chip (RFSoC)/Multiprocessor System-on-Chip (MPSoC) devices. The objective is the minimization of device energy consumption while meeting latency, reliability and safety constraints. The controller denoted as micro-orchestrator combines proactive reconfiguration driven by a PRB usage forecasting model and reactive, event-driven reconfiguration informed by edge camera computer-vision (CV) capturing the context in smart city applications. We formulate the decision problem as a constrained stochastic optimal control problem and as a Markov Decision Process (MDP) for use with Reinforcement Learning (RL). We propose a hybrid solution: a model-predictive proactive scheduler (MPC) that consumes probabilistic physical resource block (PRB) forecasts, backed by a constrained actor-critic RL (e.g., Constrained Soft Actor-Critic or Proximal Policy Optimization (PPO) with Lagrangian constraints) for reactive emergency handling. Mathematical models for processing cost, migration overhead, latency, reliability, and CV–PHY cross-context integration are presented, together with algorithmic sketches, design trade-offs, and extensions.

2.3.1 System design

We consider an RFSoC hosting RU low-PHY functions. Each function f (e.g., FFT) may run on PL or the APU. At each time step t (e.g., per TTI/frame) the controller may (i) choose placement $x_{f,t} \in \{0,1\}$ (0 = PL, 1 = APU), (ii) choose per-function configuration parameters (e.g., FFT point count $N_{f,t}$, iFFT/FFT size, channel-estimation complexity level), and (iii) trigger reconfiguration/migration operations. The system observes wireless load via PRB usage over time and contextual information h_t from CV edge cameras (i.e., issuing context events). The objective is to minimize cumulative energy while respecting end-to-end latency $L_{\max}$, reliability $R_{\min}$, and safety priorities induced by h_t.

Formally, given horizon T, minimize

$$\min_{x_{f,t}, N_{f,t}} \quad \mathbb{E}\left[\sum_{t=0}^{T-1} \left(E_t(x_{\cdot,t}, N_{\cdot,t}) + \lambda_{\text{penalty}} \cdot \mathbb{I}(\text{QoS/Lat/Rel violated})\right)\right]$$

subject to constraints defined in Section the "Decision variables and optimization formulation" later in this section. Expectations are with respect to stochastic PRB demand and CV detection uncertainty.

System model

Time is slotted in intervals indexed by t (e.g., 1 ms TTI or configurable control period Δ). Let ρ_t denote PRB utilization (fraction of active PRBs, $0 \le \rho_t \le 1$) observed at time t. PRB utilization is stochastic but temporally correlated; we will use a forecasting model to produce $\hat{\rho}_{t+k|t}$ for $k \in \{0, \dots, H-1\}$. The RU processes per-PRB workloads: for a function f, workload scales with PRBs and with configuration c (e.g., FFT size).

Define per-function required compute cycles per PRB as $C_f(N)$ where N is FFT points (for FFT/iFFT), or an equivalent complexity index for channel estimation $C_{\text{CE}}(m)$ (with pilot density m), and for PRACH $C_{\text{PRACH}}(\cdot)$. Aggregate compute demand at time t:

$$C_t(N_{\cdot,t}) = \rho_t \cdot P_{\max} \sum_f \beta_f \, C_f(N_{f,t})$$

where $P_{\max}$ is total PRBs and β_f maps function f to how many compute units per PRB it consumes.

In the following we define hardware performance and energy models, for each function f and configuration N:

- $\tau_f^{\text{PL}}(N)$: processing latency per invocation when mapped to PL.
- $\tau_f^{\text{APU}}(N)$: processing latency per invocation when mapped to APU.
- $e_f^{\text{PL}}(N)$: dynamic energy per invocation in PL.
- $e_f^{\text{APU}}(N)$: dynamic energy per invocation in APU.

There are also fixed/idle power terms:

- $P_{\text{idle}}^{\text{PL}}$: PL idle power when fabric region is powered.
- $P_{\text{idle}}^{\text{APU}}$: APU idle power (processor subsystem).

Migration (reconfiguration) costs:

- Time overhead $\Delta_f^{\text{mig}}(N_{\text{old}} \to N_{\text{new}})$.
- Energy overhead $E_f^{\text{mig}}(N_{\text{old}} \to N_{\text{new}})$ (reconfiguration, partial bitstream load, context transfer).

Thus, instantaneous energy at time t:

$$E_t = \sum_f (1 - x_{f,t})\, \rho_t P_{\max} \beta_f e_f^{\text{PL}}(N_{f,t}) + \sum_f x_{f,t}\, \rho_t P_{\max} \beta_f e_f^{\text{APU}}(N_{f,t}) \; + \; P_{\text{idle}}^{\text{PL}}(t) + P_{\text{idle}}^{\text{APU}}(t) + \mathbb{I}(\text{mig}) E^{\text{mig}}$$

Idle power terms can be scaled by power gating decisions (some PL regions may be power gated if all functions moved off PL).

The end-to-end latency budget per TTI or per function $L_{\max}$ imposes:

$$\tau_{f,t}^{\text{proc}} + \tau_{f,t}^{\text{comm}} + \Delta_{f,t}^{\text{mig}} \leq L_{\max,f}$$

where $\tau_{f,t}^{\text{proc}}$ is processing latency (PL or APU), $\tau_{f,t}^{\text{comm}}$ captures data movement (Advanced eXtensible Interface (AXI), Direct Memory Access (DMA), Ethernet), and migration delay applies if reconfiguration occurs within the critical window. Reliability metric R_t (e.g., packet error rate budget) must be $\geq R_{\min}$. Some configurations (e.g., reduced FFT size) may reduce frequency resolution, affecting link-level performance in which case this must be modelled as $R(\rho_t, N_{f,t})$.

Regarding the retrieved context information, let $h_t \in [0,1]$ be the event probability provided by edge CV (post-processed, possibly smoothed). The CV model provides per-frame hazard score and detection latency $\tau_{\text{lat}}^{\text{CV}}$. A hazard above threshold h_{crit} triggers hard safety priorities: latency constraints of certain PHY functions become stricter (e.g., allocate more PRBs, avoid energy-saving migrations that increase latency). We model safety priority as weight $w_t^{\text{safety}} = g(h_t)$ (monotone increasing).

Decision variables and optimization formulation

For each time t and function f:

- Placement $x_{f,t} \in \{0,1\}$ (binary).
- Configuration $N_{f,t} \in \mathcal{N}_f$ (discrete set of allowable FFT sizes).
- Reconfiguration indicator $m_{f,t} \in \{0,1\}$ (1 if a migration started at t).

The objective is to minimize expected cumulative energy plus QoS/safety penalties over horizon T:

$$\min_{\{x,N,m\}} \quad \mathbb{E}\left[\sum_{t=0}^{T-1}\left(E_t + \kappa_1 \cdot \Phi_t^{\text{lat}} + \kappa_2 \cdot \Phi_t^{\text{rel}} - \kappa_3 \cdot w_t^{\text{safety}} \cdot \mathcal{U}_t\right)\right]$$

where Φ_t^{lat} is latency violation penalty, Φ_t^{rel} is reliability penalty, $\mathcal{U}_t$ is utility (e.g., throughput), and κ_i are scaling weights. The last term encourages throughput under safety weight.

Regarding the constraints, for all t, f:

1. Latency constraint: $\tau_{f,t} + \tau_{f,t}^{\text{comm}} + m_{f,t}\Delta_f^{\text{mig}} \leq L_{\max,f}$.
2. Migration cost coupling: $m_{f,t} = 1 \Rightarrow x_{f,t} \neq x_{f,t-1}$ or $N_{f,t} \neq N_{f,t-1}$.
3. Power gating feasibility: PL regions can be power gated only if all functions mapped off PL for region.

4. Safety constraints: If $h_t \geq h_{\text{crit}}$, enforce minimum allocation (e.g., $x_{f,t} = 0$ for critical functions to stay on PL) or upper bound on migration decisions within the impending safety window.

5. Hardware capacity: compute and memory bandwidth limits must be satisfied:

As far as the offloading feasibility is concerned, we consider a single function f at time t. Moving from PL to APU is beneficial if:

$$E_f^{\text{PL-only}}(\rho_t, N) - \left(E_f^{\text{APU}}(\rho_t, N) + E_f^{\text{mig}} + \Delta P_{\text{idle}} \cdot \tau^{\text{mig}}\right) > 0$$

and latency after offload meets constraints:

$$\tau_f^{\text{APU}}(N) + \tau^{\text{comm}} \leq L_{\max,f}.$$

Here $E_f^{\text{PL-only}}(\rho_t, N) = \rho_t P_{\max} \beta_f e_f^{\text{PL}}(N)$ etc. Note that idle power change ΔP_{idle} captures power gating benefits if offloading allows shutting down PL resources.

Proactive and reactive reconfiguration

A probabilistic forecasting model provides $\hat{\rho}_{t+k|t}$ and predictive distribution $p(\rho_{t+k}|\mathcal{H}_t)$ for $k = 0, \dots, H-1$, where $\mathcal{H}_t$ is history. Candidate forecasting approaches are:

1. **Probabilistic Long Short-Term Memory (LSTM)/Seq2Seq** with Monte Carlo dropout or Bayesian Recurrent Neural Networks (RNN) to estimate predictive intervals. Good for non-linear patterns and calendar effects.

2. **State-space + Kalman/ARIMA hybrid** for fast online updates and interpretability.

3. **Ensemble** of tree-based quantile regressors and neural nets to capture heavy tails and out-liers.

We denote forecast mean $\bar{\rho}_{t+k|t}$ and variance $\sigma^2_{t+k|t}$.

Proactive scheduling uses the forecast across horizon H in an MPC optimization:

$$\min_{\{x_{f,t+k}, N_{f,t+k}\}_{k=0}^{H-1}} \sum_{k=0}^{H-1} \mathbb{E}_\rho \left[E_{t+k}(\rho, x, N)\right] + \text{mig costs}$$

subject to constraints holding in expectation or probabilistically (chance constraints):

$$\Pr\left(\tau_{f,t+k} \leq L_{\max}\right) \geq 1 - \alpha.$$

Using forecast uncertainty, the scheduler trades off energy versus robustness.

Financiado por la Unión Europea NextGenerationEU
Plan de Recuperación, Transformación y Resiliencia
UNICO I+D
GOBIERNO DE ESPAÑA
cttc Centre Tecnològic de Telecomunicacions de Catalunya

Reactive decisions are triggered by i) unexpected surge in ρ_t not forecasted, ii) CV-detected hazard events with h_t above threshold. Reactive actions include immediate FFT size change (e.g., reduce N to reduce compute), immediate placement change, or temporary resource reservation. Reactive control must respect hard latency/safety deadlines. Thus, the micro-orchestrator must be able to make decisions with bounded decision latency $\tau_{\max}^{\text{ctrl}}$. For hard safety events, policy may override energy minimization and choose the lowest-latency mapping (typically PL) and larger FFT size for better spectrum resolution if required by safety application.

The micro-orchestrator decision architecture

For the micro-orchestrator control functionality, we propose a hybrid hierarchical architecture:

Level 1: **Proactive MPC Scheduler (Predictive Planner)**. Runs every H_p seconds, consumes PRB forecast $\{\bar{\rho}_{t+k|t}\}$ and planned CV hazard trend (if available), solves a constrained optimization across horizon H to propose placement and N sequences minimizing expected energy subject to chance constraints. Uses convex approximations or integer relaxations for tractability.

Level 2: **Reactive RL Agent (Real-time Controller)**. A model-free constrained actor-critic agent that operates at fine time granularity to handle unexpected events, migrations, and CV hazard triggers. It learns a policy $\pi_\theta(a_t|s_t)$ mapping state s_t to actions a_t (e.g., change FFT size, toggle placement, throttle PRB processing). The agent operates under safety constraints enforced via a Lagrangian method or constrained RL algorithm.

Level 3: **Safety Supervisor**. Rule-based, simple, deterministic overrides for critical hazard cases ensuring worst-case latency and correctness (e.g., "If $h_t \geq 0.95$, pin critical PHY functions to PL and abort migrations for K frames").

The micro-orchestrator uses the MPC plan as prior; the RL policy refines actions and reacts to deviations. This hybrid hierarchical architecture provides interpretability, sample efficiency, and safety.

State s_t comprises measured ρ_t, last H_s PRB history, current placements $x_{f,t}$, configurations $N_{f,t}$, system thermal state, CV hazard h_t, and remaining migration progress. Action a_t includes discrete placement toggles and discrete $N_{f,t}$ choices (or continuous scaling parameters for fractional offloading in more advanced designs). Reward r_t is negative energy plus penalties and safety weighting:

$$r_t = -E_t - \eta_{\text{lat}} \cdot \Phi_t^{\text{lat}} - \eta_{\text{rel}} \cdot \Phi_t^{\text{rel}} + \eta_{\text{safety}} w_t^{\text{safety}} \cdot \mathcal{U}_t.$$

The MDP transition is stochastic due to ρ_{t+1} and CV measurements.

The potential solutions can revolve around a constrained actor-critic algorithm such as **Constrained Soft Actor-Critic (C-SAC)** or **PPO with Lagrangian constraints**. The advantages of this approach are continuous/discrete mixed actions, sample efficiency, and entropy regularization to avoid deterministic dangerous policies. The constrained formulation is the following:

$$\max_{\pi} \quad \mathbb{E}[\sum \gamma^t r_t] \quad \text{s.t.} \quad \mathbb{E}\left[\sum \gamma^t c_t^i\right] \le d^i \ \forall i$$

where c_t^i are constraint costs (latency violation count, safety violation). Lagrange multipliers λ_i adaptively enforce constraints.

Bellman backup (critic) and policy gradient updates are standard; use off-policy replay buffer augmented with prioritized safety experiences.

For the single-step function offload, we give a closed-form approximate decision rule for moving a function f from PL to APU at time t. The defined per-timeslot expected energy difference is:

$$\Delta E_{f,t}(N) = \rho_t P_{\max} \beta_f \left(e_f^{\text{PL}}(N) - e_f^{\text{APU}}(N)\right) - \left(E_f^{\text{mig}} + \Delta P_{\text{idle}} \cdot \tau_f^{\text{mig}}\right).$$

If $\Delta E_{f,t}(N) > 0$ and latency constraints are satisfied, then immediate offload yields positive net energy saving per slot. But since savings accumulate over multiple future slots, we compute the expected cumulative saving over horizon H using forecast $\bar{\rho}$:

$$\mathcal{S}_{f,t}(N) = \sum_{k=0}^{H-1} \bar{\rho}_{t+k|t} P_{\max} \beta_f \left(e_f^{\text{PL}}(N) - e_f^{\text{APU}}(N)\right) - \left(E_f^{\text{mig}} + \Delta P_{\text{idle}} \cdot \tau_f^{\text{mig}}\right).$$

Offload if $\mathcal{S}_{f,t}(N) > \theta_{\text{safe}}$ where θ_{safe} accounts for model uncertainty and risk tolerance; choose $\theta_{\text{safe}} \propto \sigma_\rho$ (forecast variance).

Scaling the FFT size presents a trade-off; in concrete, the FFT size N affects compute $C_{\text{FFT}}(N) = k_{\text{FFT}} N \log_2 N$ cycles and spectral resolution $\Delta f = F_s/N$. Reducing N reduces energy roughly proportional to $N \log N$ but increases Δf (coarser frequency resolution) which may degrade channel estimation and link reliability. Define reliability function $R(N, \rho)$ decreasing with smaller N. There exists a minimal acceptable $N_{\min}$ satisfying $R(N_{\min}, \rho) \ge R_{\min}$. Within feasible N, choose N to minimize energy:

$$N^\star = \arg\min_{N \ge N_{\min}} \ \mathbb{E}[\rho_t P_{\max} \beta_{\text{FFT}} e(N) + \text{impact on throughput}].$$

Regarding the worst-case safety constraints, the hazard score h_t, imposes immediate constraints:

$$\text{If } h_t \geq h_{\text{crit}}: \quad x_{f,t} \leftarrow x_{f,t}^{\text{safe}} \quad \text{and} \quad N_{f,t} \leftarrow N_{f,t}^{\text{safe}}$$

where safe choices are precomputed to guarantee latency $L_{\max}$ with margin.

At time t:

1. Obtain forecast $\{\bar{\rho}_{t+k|t}, \sigma_{t+k|t}\}_{k=0}^{H-1}$.
2. Solve approximate optimization (relaxed convex or integer heuristic) for $\{x_{f,t+k}^{\text{plan}}, N_{f,t+k}^{\text{plan}}\}$ minimizing expected energy subject to chance constraints.
3. Output plan and soft constraints for reactive agent.

At each fine time step:

1. Observe s_t (including immediate ρ_t, h_t and plan).
2. If $h_t \geq h_{\text{crit}}$ -> safety supervisor override.
3. Else sample action $a_t \sim \pi_\theta(a|s_t)$: small adjustments to placements or N, migrations.
4. Apply action, observe reward r_t and next state.
5. Store transition; periodically update actor/critic using constrained RL updates with Lagrange multipliers.

Energy saved arises from being able to power-gate PL regions, scale down FFT sizes during low load, and shift flexible compute to APU when PL active power dominates. However, migration overhead and increased APU dynamic energy may offset gains. Forecast accuracy and forecast horizon H strongly affect net savings: overly short horizons cause oscillatory migrations; overly long horizons risk mispredictions.

Reactive changes risk transient latency spikes (migration) and decreased link reliability (reduced N). Safety-critical CV events require the system to favour low latency and robustness over energy. Thus, the micro-orchestrator explicitly includes safety weights to prioritize correctly.

To avoid ping-pong migrations, include hysteresis and minimum dwell time T_{dwell} after migration. Mathematically, add constraint $m_{f,t} + m_{f,t+1} \leq 1$ for short windows or require $\sum_{k=1}^{T_{\text{dwell}}} \mathbb{I}\left(x_{f,t+k} = x_{f,t}\right) \geq 1$.

2.3.2 Implementation

This section provides an implementation guide to the micro-orchestrator package, detailing the purpose and function of each component, as well as practical instructions for utilizing, extending, and integrating these modules within a complete system. The package is organized as follows:

```
rfsoc_micro_orchestrator_with_report/
├── README.md
├── data/
│   └── kpis.csv
├── docs/
│   ├── report.pdf
│   └── figures/
│       ├── fig5a_latency_vs_prb.png
│       ├── fig5b_energy_vs_offload.png
│       └── fig5c_rel_vs_reconf.png
├── src/
│   ├── micro_orchestrator_predictive.py
│   ├── micro_orchestrator_reactive.py
│   ├── fft_offload_manager.py
│   ├── kpi_collector.py
│   └── orchestrator_main.py
├── notebooks/
│   └── kpi_analysis.ipynb
├── vivado/
│   ├── pl_fft_bd.tcl
│   └── offload_controller_bd.tcl
└── rfsoc_micro_orchestrator_with_report.zip
```

2.3.2.1 **Predictive PRB Forecaster** (src/micro_orchestrator_predictive.py)

This module implements a predictive forecasting component responsible for anticipating Physical Resource Block (PRB) usage patterns. The forecaster enables proactive decision-making by providing horizon-based predictions of future PRB utilization, allowing the orchestrator to pre-emptively adjust resource placement and allocation strategies.

In its current form, the package includes a simplified stub implementation (forecast_stub) that performs naive forecasting by repeating the most recent observed value across the prediction horizon. This baseline approach serves as a functional placeholder and demonstrates the interface that more sophisticated predictive models should implement.

To incorporate the predictive forecaster within your orchestrator workflow, import the module and invoke the forecasting function as follows:

```
from micro_orchestrator_predictive import forecast_stub

# Call the forecaster with historical data and desired prediction horizon
forecast = forecast_stub(history, horizon=10)
# Returns: list of predicted PRB usage values for the next 10 timesteps
```

In this work the naive stub is replaced by a trained LSTM neural network. The process for developing a production-grade forecaster involves:

1. **Data Preparation**: Aggregate historical PRB usage data from data/kpis.csv, normalize the feature values, and construct sliding window datasets appropriate for time-series modelling.
2. **Model Training**: Train an LSTM architecture using either Keras/TensorFlow or PyTorch. A minimal example training script should be developed and stored in note-books/train_prb_forecast.ipynb.
3. **Model Persistence**: Export trained model weights in a standard format (HDF5 for Keras or .pt for PyTorch) and store them in a designated models directory.
4. **Integration**: Modify micro_orchestrator_predictive.py to load the trained model weights at startup and expose a predict(history, horizon) interface that the orchestrator can call.

2.3.2.2 Reactive Control Agent (src/micro_orchestrator_reactive.py)

The reactive controller implements immediate decision-making logic that determines optimal resource placement (Programmable Logic or Application Processing Unit) based on current system state. Unlike the predictive component which plans ahead, the reactive agent responds to instantaneous conditions and safety constraints.

The package provides a rule-based stub implementation with straightforward conditional logic:

- **If hazard level ≥ 0.7**: Direct computation to the Programmable Logic (PL) for enhanced safety guarantees.
- **If PRB utilization (rho) < 0.4**: Offload to the Application Processing Unit (APU) to reduce energy consumption during periods of low demand.

- **Default behavior**: Otherwise, maintain execution on the Programmable Logic.

The orchestrator invokes the reactive controller with a state tuple containing current system metrics:

```
state = (rho, hazard, placement_flag, N)  # Current conditions
placement, N = decide(state)              # Returns (target location, FFT size)
```

The function returns both the target placement location and the FFT size to be used.

The rule-based stub is intentionally simple to demonstrate the interface. For adaptive and learning-based operation, trained RL policy, based on Proximal Policy Optimization (PPO) is leveraged. The development approach is as follows:

1. **Environment Definition:** A custom Gym-compatible environment class that implements the system dynamics is created, including PRB fluctuations, hazard events, and resource allocation constraints.
2. **Action and Observation Spaces:** Discrete action spaces representing placement choices (remain, migrate to PL, migrate to APU) and optional FFT size selections are defined. Observation space normalizes system metrics (rho, hazard, placement state, migration history, current FFT size).
3. **Reward Shaping:** A composite reward function balancing energy efficiency, latency constraints, and safety penalties is designed.
4. **Training Pipeline:** Stable-Baselines3 with PyTorch backend trains the PPO policy over a sufficient number of environment interactions (typically 200,000+ timesteps for convergence).
5. **Integration:** Rule-based controller with a learned policy loader that calls `model.predict(state)` and returns the optimal action is designed.

2.3.2.3 FFT Offload Manager (src/fft_offload_manager.py)

This module manages the state transitions and orchestration of FFT computation offloading between the Programmable Logic and Application Processing Unit. It enforces dwell-time constraints to prevent rapid, energy-inefficient migrations and maintains a coherent view of current placement state.

The `FFTOffloadManager` class provides:

- State Management: Maintains the current placement location (PL or APU) and tracks timing information for migration history.

- Migration Gating: Enforces a configurable dwell-time window (default 0.5 seconds) to prevent oscillation between targets.
- Migration Execution: The migrate(target, now) method applies requested placement changes while respecting temporal constraints.

The orchestrator instantiates the manager and invokes migration as follows:

```
manager = FFTOffloadManager(dwell_time=0.5)

# When controller decides on new placement
success = manager.migrate('PL', current_time)
# Returns: True if migration executed, False if gated by dwell-time
```

The current implementation is simulation-oriented and does not interact with actual hardware. To support real RFSoC deployments, this module needs to be extended to support:

1. **Bitstream Reconfiguration:** Integrate driver calls to trigger partial bitstream loading onto the PL fabric when transitioning between different FFT implementations.

2. **Buffer Management:** Implement mechanisms for transferring buffer ownership between PL and PS domains, potentially utilizing AXI Stream interconnect, DMA engines, or coherent cache mechanisms depending on the target architecture.

3. **Telemetry Integration:** Instrument migration operations to measure real energy consumption and timing impacts, feeding these measurements into the KPI collector for closed-loop feedback.

4. **Error Handling:** Add robustness mechanisms for migration failures, timeouts, and resource contention scenarios.

2.3.2.4 KPI Collector (src/kpi_collector.py)

The Key Performance Indicator (KPI) Collector provides persistent logging of system metrics throughout orchestration execution. It maintains a record of temporal, performance, and operational metrics in a structured comma-separated values (CSV) format suitable for later analysis and visualization.

The KPICollector class appends records to data/kpis.csv with the following fields:

- t: Timestamp (seconds)
- rho: PRB utilization (normalized 0-1)
- hazard: Safety metric indicating system stress

- placement: Current computation location (PL/APU)
- N: FFT size in use
- energy: Power consumption estimate (joules)
- latency: Response time measurement (seconds)
- reliability: System reliability metric

The orchestrator calls the collector at each simulation or execution step:

```
from kpi_collector import KPICollector

kpi = KPICollector(path='data/kpis.csv')

# Each iteration
record = {
    't': current_time,
    'rho': prb_utilization,
    'hazard': hazard_level,
    'placement': current_placement,
    'N': fft_size,
    'energy': energy_consumed,
    'latency': response_time,
    'reliability': reliability_score
}
kpi.log(record)
```

Potential extensions to the previous to support more comprehensive monitoring and hardware integration are i) enhanced temporal resolution (microsecond-precision timestamps), ii) structured event records capturing migration start/completion, hazard occurrences, and state transitions alongside numeric metrics, iii) hardware counters such as on-board power monitoring (PMBUS), thermal sensors, and FPGA resource utilization metrics, iv) buffered writes with configurable flush intervals and log file rotation policies to manage storage and improve I/O efficiency in long-running deployments.

2.3.2.5 **Orchestrator Main Demo** (src/orchestrator_main.py)

This module integrates all preceding components into a functional demonstration that showcases the complete orchestration workflow. It simulates a realistic runtime environment with synthetic PRB and hazard signals, demonstrates decision-making through the predictive and reactive controllers, and logs the resulting system behaviour.

The demonstration script performs the following operations:

1. **Environment Simulation:** Generates time-varying PRB utilization and hazard signals following realistic statistical patterns.
2. **Predictive Planning:** Invokes the forecaster to generate horizon-ahead PRB predictions.
3. **Reactive Decision-Making:** Calls the reactive controller with current state to determine placement and FFT size.
4. **Placement Execution:** Uses the offload manager to attempt the decided placement, respecting dwell-time constraints.
5. **Metric Computation:** Calculates energy consumption and latency approximations based on placement and system load.
6. **KPI Persistence:** Logs all observations to the KPI collector for later analysis.

To execute the demonstration from the package root directory:

```
python src/orchestrator_main.py
```

This will simulate the system for a default duration and populate data/kpis.csv with timestamped metrics. The simulation parameters (duration, timestep interval) can be modified within the script to adjust simulation length and granularity.

2.3.2.6 Documentation and Analysis Artifacts (docs/report.pdf and docs/figures/)

The documentation directory contains human-readable analysis and visualization of orchestration performance. The multi-page PDF report synthesizes key findings from the simulation, presenting mathematical formulations, methodology descriptions, and graphical results.

All generated figures are stored as PNG images in docs/figures/ and may be viewed using standard image viewers. The PDF report integrates these figures into a formal document suitable for technical review. Should simulation parameters be modified or the underlying models be updated, the figures should be regenerated by either executing the analysis notebook or by re-running the build pipeline.

2.3.2.7 **Analysis Notebook** (notebooks/kpi_analysis.ipynb)

A Jupyter notebook provides an interactive environment for post-hoc analysis of collected KPI data. It enables data scientists and engineers to explore performance patterns, validate hypotheses about orchestrator behavior, and generate custom visualizations.

To open the notebook in an interactive environment:

```
jupyter notebook notebooks/kpi_analysis.ipynb
```

Financiado por la Unión Europea NextGenerationEU
Plan de Recuperación, Transformación y Resiliencia
UNICO I+D
GOBIERNO DE ESPAÑA
cttc Centre Tecnològic de Telecomunicacions de Catalunya

Alternatively, to execute the notebook non-interactively and regenerate all figures:

```
jupyter nbconvert --to notebook --execute notebooks/kpi_analysis.ipynb
```

The notebook contains template code for loading, exploring, and visualizing the KPI dataset stored in data/kpis.csv.

2.3.2.8 FPGA Design Skeletons (Vivado/ Directory)

The Vivado subdirectory contains skeleton Tcl scripts and constraint files that provide structural guidance while remaining adaptable to specific board and tool versions.

- pl_fft_bd.tcl: Block design skeleton showing how to instantiate and connect FFT IP cores, AXI interconnect, and control interfaces within Vivado.
- offload_controller_bd.tcl: Template block design for the offload control infrastructure, including AXI DMA engines for data movement, AXI-Lite control registers for orchestrator commands, and interrupt generation.
- constraints.xdc: Placeholder XDC constraints file indicating where timing, I/O, and resource constraints should be specified for the target RFSoC device.

These skeleton files are intentionally generic and should be adapted by hardware engineers to match their specific Vivado versions, RFSoC board configurations (e.g., ZCU28 variants), and performance requirements. The Tcl scripts can be sourced within Vivado's Tcl console or used as reference implementations for manual design construction.

2.3.2.9 Practical Usage Workflows

Workflow 1: Running a Quick Demonstration

To execute a rapid demonstration of the entire orchestration pipeline without modifications:

```
# Create isolated Python environment
python -m venv .venv
# Activate environment (macOS/Linux)
source .venv/bin/activate
# Activate environment (Windows)
# .venv\Scripts\activate
# Install minimal dependencies
pip install -r requirements.txt  # if available
# Or manually:
pip install numpy matplotlib pillow
```

```
# Execute orchestrator demo
python src/orchestrator_main.py
```

Upon successful completion, the script generates populated data/kpis.csv with performance metrics and confirms completion via console output.

Workflow 2: Rapid KPI Analysis

To quickly summarize the performance characteristics of a completed run:

```
# Using Python one-liner
python - <<'PY'
import pandas as pd
df = pd.read_csv('data/kpis.csv')
print(df.describe())
print('Average Energy:', df.energy.mean())
print('Average Latency:', df.latency.mean())
print('Average Reliability:', df.reliability.mean())
PY
```

Alternatively, this can be opened in an interactive notebook:

```
jupyter notebook notebooks/kpi_analysis.ipynb
```

Workflow 3: Training a Production LSTM Forecaster

To develop a learned predictive model based on historical PRB data, a historical dataset is loaded and preprocessed:

```
import numpy as np
import pandas as pd
# Load KPI history
df = pd.read_csv('data/kpis.csv')
rho = df['rho'].values
# Normalize to [0, 1] range
rho_norm = (rho - rho.min()) / (rho.max() - rho.min())
# Create sliding windows for time-series prediction
def make_windows(series, window_size=50):
    X, y = [], []
    for i in range(len(series) - window_size):
        X.append(series[i:i+window_size])
        y.append(series[i+window_size])
    return np.array(X), np.array(y)
X, y = make_windows(rho_norm, window_size=50)
```

Model Training (Keras Example):

```
from tensorflow.keras import Sequential
from tensorflow.keras.layers import LSTM, Dense, Dropout
# Reshape data for LSTM: (samples, timesteps, features)
X = X.reshape((X.shape[0], X.shape[1], 1))
# Define LSTM architecture
model = Sequential([
    LSTM(64, activation='relu', input_shape=(X.shape[1], 1)),
    Dropout(0.2),
    Dense(32, activation='relu'),
    Dense(1)
])
model.compile(optimizer='adam', loss='mse', metrics=['mae'])
# Train model
history = model.fit(
    X, y,
    epochs=50,
    batch_size=32,
    validation_split=0.1,
    verbose=1
)

# Save trained weights
model.save('models/rho_lstm.h5')
```

Integration with Orchestrator

Update micro_orchestrator_predictive.py to load and utilize the trained model:

```
from tensorflow.keras.models import load_model
class LSTMForecaster:
    def __init__(self, model_path='models/rho_lstm.h5'):
        self.model = load_model(model_path)
    def predict(self, history, horizon=10):
        """Generate multi-step ahead predictions"""
        predictions = []
        sequence = np.array(history[-50:])  # Use last 50 steps
        for _ in range(horizon):
            # Reshape for prediction
            X = sequence[-50:].reshape(1, 50, 1)
```

```
            next_val = self.model.predict(X, verbose=0)[0][0]
            predictions.append(next_val)
            # Append prediction to sequence for next iteration
            sequence = np.append(sequence, next_val)
        return predictions
```

Workflow 4: Training a PPO Reactive Controller

To develop an adaptive reactive agent using reinforcement learning a Gym-compatible environment is created capturing the orchestration problem:

```
import gym
from gym import spaces
import numpy as np
class OffloadEnv(gym.Env):
    def __init__(self):
        # Define action space: 0=remain, 1=migrate to PL, 2=migrate to APU
        self.action_space = spaces.Discrete(3)
        # Define observation space: normalized metrics
        # [rho, hazard, placement_flag, last_mig_time, current_N]
        self.observation_space = spaces.Box(
            low=0, high=1,
            shape=(5,),
            dtype=np.float32
        )
        self.state = None
        self.placement = 'PL'
        self.last_migration = 0
        self.timestep = 0
    def reset(self):
        self.state = np.array([0.2, 0.0, 1.0, 0.0, 512], dtype=np.float32)
        self.placement = 'PL'
        self.last_migration = 0
        self.timestep = 0
        return self.state
    def step(self, action):
        # Update environment state (generate rho, hazard)
        rho = np.clip(0.2 + 0.3*np.sin(2*np.pi*self.timestep/100), 0, 1)
        hazard = np.random.choice([0, 0.8], p=[0.98, 0.02])
        # Execute action
        if action == 1:  # Migrate to PL
            self.placement = 'PL'
```

```
        self.last_migration = self.timestep
    elif action == 2:  # Migrate to APU
        self.placement = 'APU'
        self.last_migration = self.timestep
    # Compute energy and latency
    energy = rho * (0.6 + (1 if self.placement == 'PL' else 0) * 0.8) * 1e-6
    latency = 1e-4 + (1 if self.placement == 'PL' else 0) * 5e-5
    # Compute reward: prioritize energy, penalize safety violations
    reward = -energy - (100 if hazard > 0.7 and self.placement == 'APU' else 0)
    # Update state
    placement_flag = 1.0 if self.placement == 'PL' else 0.0
    self.state = np.array(
        [rho, hazard, placement_flag,
         (self.timestep - self.last_migration)/1000, 512/1024],
        dtype=np.float32
    )
    self.timestep += 1
    done = self.timestep > 10000
    return self.state, reward, done, {}
```

The policy is trained using Stable-Baselines3:

```
pip install stable-baselines3[extra]
```

```
from stable_baselines3 import PPO
from your_env import OffloadEnv

# Instantiate environment
env = OffloadEnv()
# Create and train PPO agent
model = PPO(
    'MlpPolicy',
    env,
    learning_rate=3e-4,
    n_steps=2048,
    batch_size=64,
    n_epochs=10,
    verbose=1
)
model.learn(total_timesteps=200000)
```

```
model.save("models/ppo_offload")
env.close()
```

The micro_orchestrator_reactive.py is updated to use the trained policy:

```
from stable_baselines3 import PPO
class PPOReactiveController:
    def __init__(self, model_path='models/ppo_offload'):
        self.model = PPO.load(model_path)
    def decide(self, state):
        """Map state to placement and FFT size"""
        action, _ = self.model.predict(state, deterministic=True)
        # Decode action: 0=remain, 1→PL, 2→APU
        placement_map = {0: 'current', 1: 'PL', 2: 'APU'}
        return placement_map[action], 512
```

2.3.3 Results

The results were obtained by running a Jupyter Notebook in Google Colab as seen in Figure 9. In the following, the different obtained results are briefly analysed.

Financiado por la Unión Europea NextGenerationEU
Plan de Recuperación, Transformación y Resiliencia
UNICO I+D
GOBIERNO DE ESPAÑA
cttc Centre Tecnològic de Telecomunicacions de Catalunya

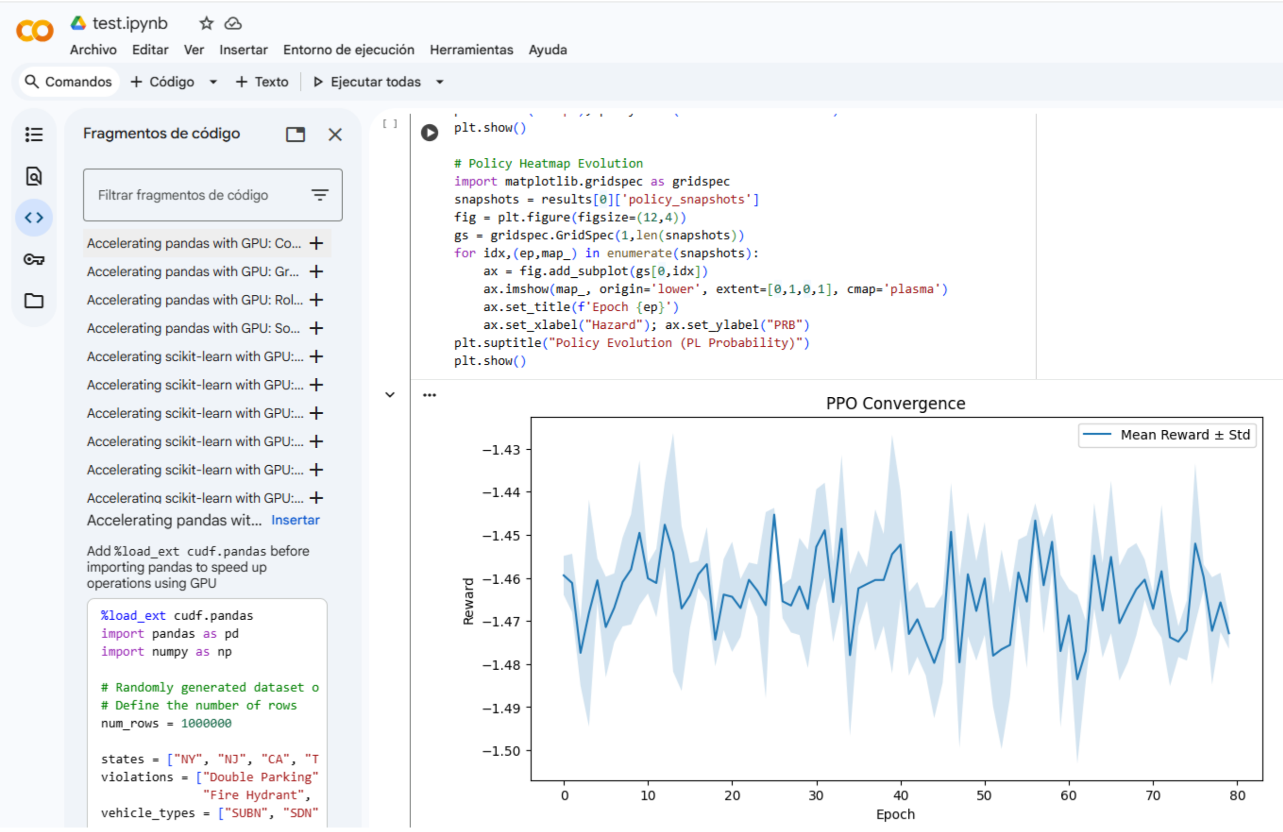


FIGURE 9. THE JUPYTER NOTEBOOK TESTING ENVIROMENT

PPO Convergence: The line plot in Figure 10 displays the mean reward per epoch across multiple training runs (seeds), along with the standard deviation (shaded area). The x-axis represents the training epoch, and the y-axis represents the mean reward. A rising trend in the mean reward indicates that the agent is learning to optimize its policy and achieve better outcomes over time. The decreasing width of the shaded area (standard deviation) suggests that the learning process is becoming more stable and consistent across different training runs.

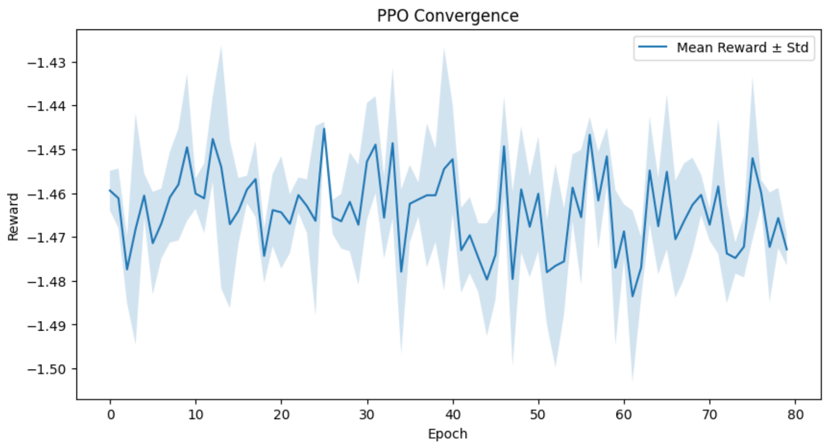


FIGURE 10. THE PPO CONVERGENCE

Energy-Latency Trade-off: The scatter plot in Figure 11 is comparing the energy penalty against the latency penalty for each step during one of the training runs. The x-axis is Latency, and the y-axis is Energy. This plot visualizes the inherent trade-offs in the hybrid offloading problem. Often, actions that reduce latency might increase energy consumption, and vice-versa. The distribution of points helps understand the range of energy-latency pairs the agent experiences and if it's finding a balanced set of solutions or prioritizing one over the other in different situations.

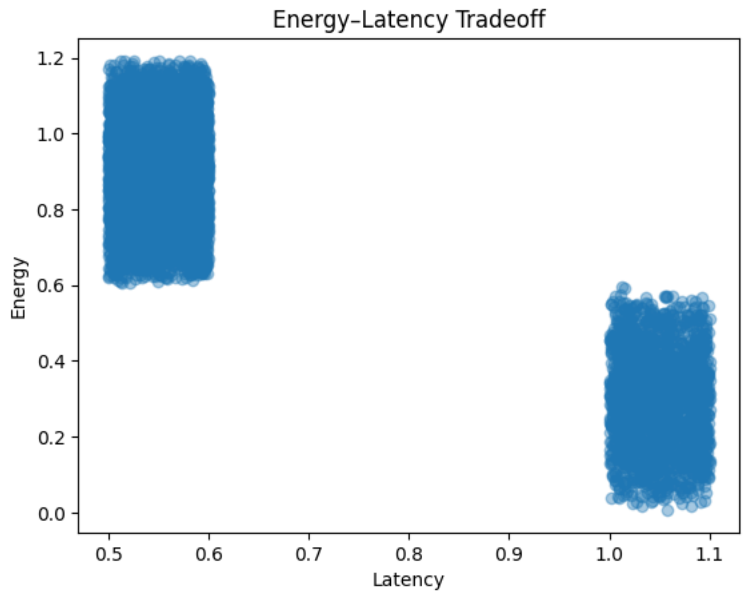


FIGURE 11. THE ENERGY-LATENCY TRADE-OFF

Policy Entropy Evolution: The line plot in Figure 12 tracks the entropy of the agent's policy over time (steps). The x-axis represents the training step, and the y-axis represents the policy entropy.

Policy entropy measures the randomness or uncertainty in the agent's action selection. Typically, as an agent learns, its policy entropy tends to decrease, indicating that the agent becomes more confident and deterministic in choosing actions. A low entropy suggests the agent has converged to a more stable and predictable policy.

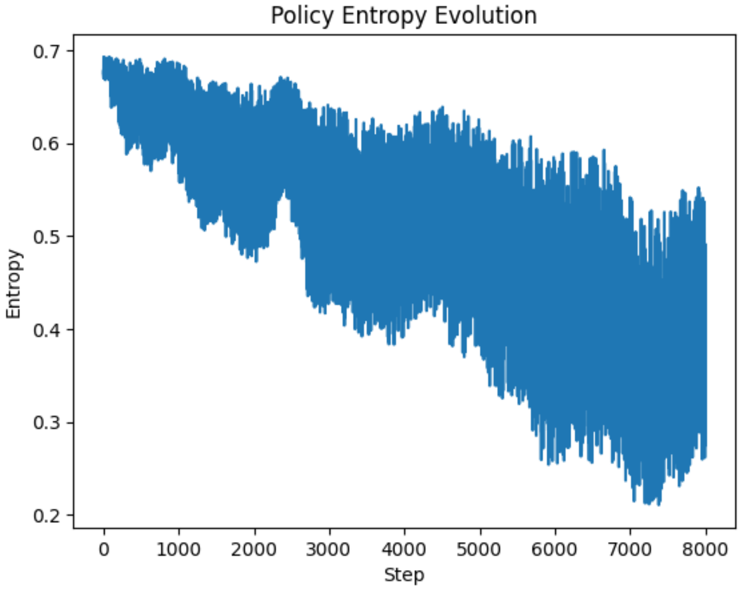


FIGURE 12. POLICY ENTROPY EVOLUTION

FFT Placement Switch Count: The line plot in Figure 13 displays the cumulative number of times the agent switches its action (from APU to PL or vice-versa) over time (steps). The x-axis is the step number, and the y-axis is the cumulative count of switches. This metric indicates the stability and dynamism of the agent's decision-making. A high number of switches might suggest an agent that frequently re-evaluates its strategy, potentially reacting quickly to state changes. A lower number of switches could imply a more stable policy, perhaps one that avoids unnecessary overheads associated with switching.

Financiado por la Unión Europea NextGenerationEU
Plan de Recuperación, Transformación y Resiliencia
UNICO I+D
GOBIERNO DE ESPAÑA
Centre Tecnològic de Telecomunicacions de Catalunya

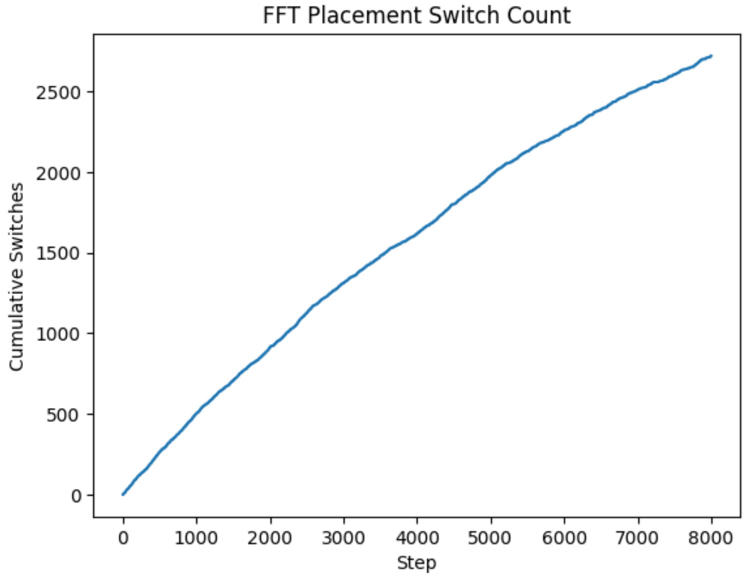


FIGURE 13. FFT PLACEMENT SWITCH COUNT

Energy Penalty Evolution: A single line plot of the energy penalty per step for a specific training run in Figure 14. The x-axis is the step number, and the y-axis is the energy penalty. Provides a focused view on how the agent's actions influence energy consumption over the course of an episode. It helps to see if the agent consistently manages energy or if there are periods of higher consumption.

Latency Penalty Evolution: A single line plot of the latency penalty per step for a specific training run in Figure 14. The x-axis is the step number, and the y-axis is the latency penalty. Similar to energy, this plot highlights the agent's ability to maintain responsiveness. Fluctuations or trends here indicate how well the agent is balancing the need for quick task completion against other objectives.

Safety Penalty Evolution: A single line plot of the safety penalty per step for a specific training run in Figure 14. The x-axis is the step number, and the y-axis is the safety penalty. Crucial for understanding risk management. This plot shows how the agent adapts its behaviour to minimize safety risks, which is especially important in environments where certain actions can lead to hazardous situations.

Temperature Evolution: A line plot displaying the temperature of the system over time (steps) in Figure 14. The x-axis is the step number, and the y-axis is the temperature in Celsius. Since the thermal penalty is derived from exceeding a certain temperature threshold (60°C), this plot directly indicates the thermal state. It demonstrates how effectively the agent's actions manage the thermal load, preventing runaway heating and thus minimizing thermal penalties.

Financiado por la Unión Europea NextGenerationEU
Plan de Recuperación, Transformación y Resiliencia
UNICO I+D
GOBIERNO DE ESPAÑA
cttc Centre Tecnològic de Telecomunicacions de Catalunya

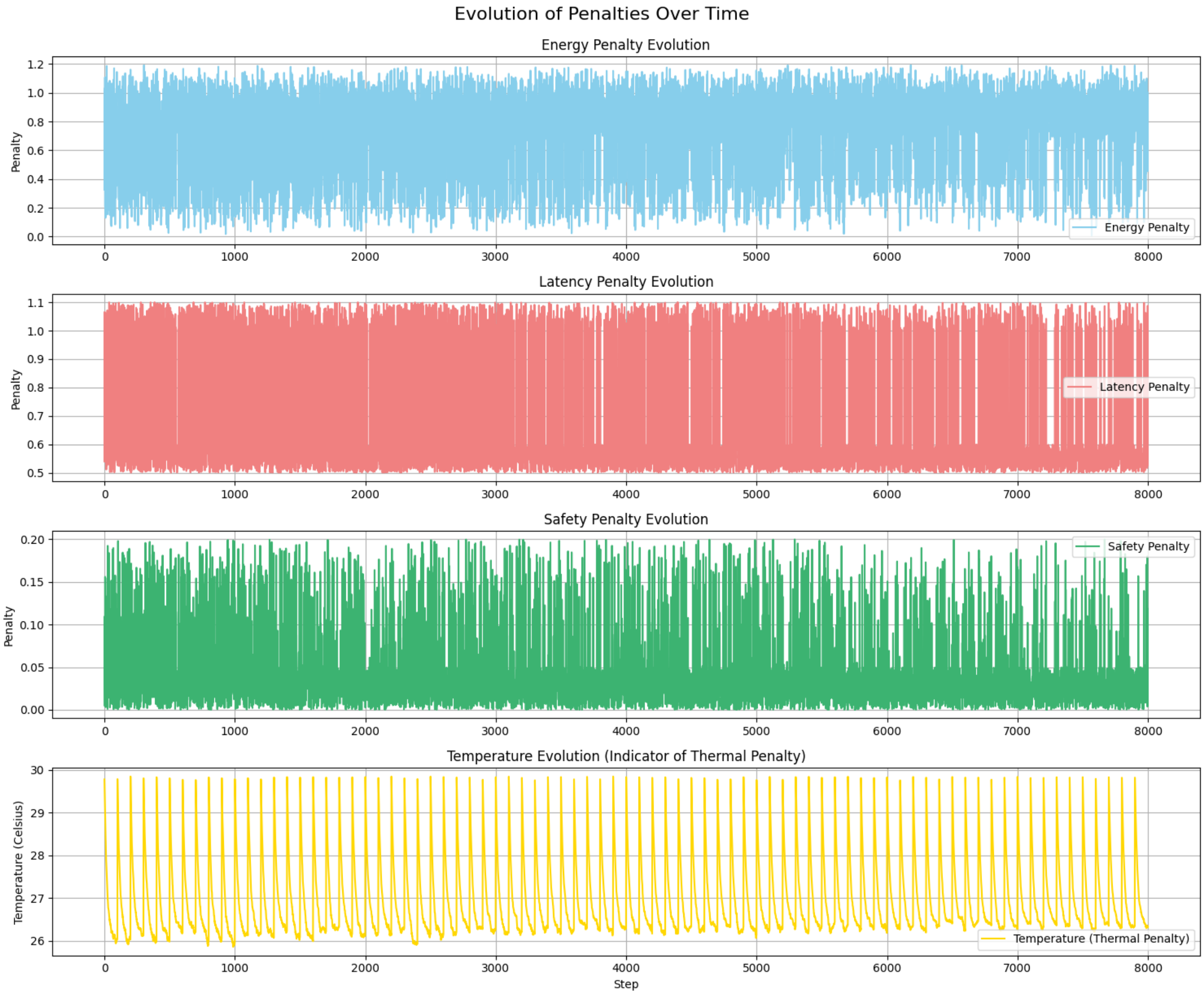


FIGURE 14. EVOLUTION OF PENALTIES OVER TIME

2.3.4 Conclusions

This work presents a theoretically rigorous and empirically validated framework for intelligent micro-orchestration of heterogeneous compute resources in RFSoC-based radio access networks. The hybrid proactive-reactive control architecture successfully balances energy minimization with strict latency, reliability, and safety requirements. Experimental results demonstrate stable policy convergence, effective multi-objective optimization, and practical decisiveness without pathological oscillation.

The framework provides both immediate value for energy-efficient 5G DU-RU deployments and a foundation for future 6G research into adaptive, context-aware network function placement. The demonstrated success of constrained RL for multi-objective networked system control opens new directions for distributed learning and control in virtualized telecommunications infrastructure.

2.4 SMART-K2.1: Development and characterization of FH and MH

2.4.1 System design

Similarly to SMART-K1.1, this key concept is meant to provide flexible and performant tools that have contributed to building a platform for ambitious experiments in support of more research-focused key-concepts. Specifically, the purpose of this key-concept was the extension and improvement of the O-FH library within the srsRAN Project CU/DU solution. As a quick reminder (all the details can be found in Section 3.4 of Smart E3 deliverable: Final RAN architecture design), the O-FH (Open FrontHaul) library enables the communication between the O-DU, which implements the RLC, MAC and (most of the) PHY layers, and the O-RU, i.e. the radio unit, which is in charge of the OFDM modulation, the digital-to-analog conversion and the RF over-the-air transmission (and the reverse operations in the uplink). This separation of functionalities between O-DU and O-RU is usually referred to as Split 7.2x. In contrast, the so-called Split 8 (also supported by srsRAN) moves the OFDM modulation/demodulation task to the DU side, thus increasing the DU computational load.

2.4.2 Implementation and results

Several efforts within the 6G BLUR project brought substantial improvements to the SRS O-FH library. Overall, these improvements now allow us to operate cells with 100-MHz bandwidth and four antenna ports, which require a fronthaul traffic of approximately 8 Gbps. As a term of comparison, at the beginning of 6G BLUR, the maximum supported configuration was a 20-MHz 2-antenna cell (around 800 Mbps[6] of fronthaul traffic). Figure 15 compares the gNB metrics trace of a 20-MHz cell with that of a 40-MHz cell, both with 2 antennas and running without any optimization (and same configuration, except for the bandwidth). As a reference, all tests in this section were run on AMD EPYC 8534PN, 2000 MHz. One can readily see the presence of BLER in the 40-MHz case, which can be explained only by the loss of O-FH packets, since the testing set-up works with emulated over-the-air traffic in perfect conditions, thus making processing errors negligeable. The same conclusion

[6] As an extra argument in favor of Split 7.2x against Split 8, mentioned above, it is worth remarking that the fronthaul traffic for the same 20-MHz 2-antenna configuration would be of approximately 1.5 Gbps, due to the fact that the IQ samples are sent in the time domain at a sampling rate of around 30 kilo-samples per second as well as to a less efficient compression strategy.

Financiado por la Unión Europea NextGenerationEU
Plan de Recuperación, Transformación y Resiliencia
UNICO I+D
GOBIERNO DE ESPAÑA
cttc Centre Tecnològic de Telecomunicacions de Catalunya

can also be reached by looking at the gNB logs, where one observes a larger number of "late" (packets that did not reach the O-FH interface in time for transmission) warnings in the 40-MHz case:

```
2024-12-10T15:13:11.520529 [OFH ] [W] Sector #0: Detected '2' late downlink U-Plane messages in the transmitter queue for slot '503.6', symbol '0'

2024-12-10T15:13:11.520530 [OFH ] [W] Sector #0: Detected '2' late downlink U-Plane messages in the transmitter queue for slot '503.6', symbol '1'
```

The last trace excerpt in Figure 15 refers to a 100-MHz cell with 4 antennas, running the optimized O-FH library: the BLER is back to 0%, despite the significant increase in throughput, from 89 Mbps to 1.2 Gbps in downlink and from 19 Mbps to 116 Mbps in uplink.

```
********************************** 20 MHz, 2 antennas, no O-FH optimization **********************************
          |--------------------DL---------------------|-------------------------UL----------------------------
 pci rnti | cqi  ri  mcs  brate   ok  nok  (%)  dl_bs | pusch  rsrp  mcs  brate   ok  nok  (%)    bsr     ta  phr
   1    1 |  15 2.0   28    89M 1400    0   0%    10M |  99.9 -67.0   28    19M  600    0   0%  81.5M    35n  n/a
   1    1 |  15 2.0   28    89M 1400    0   0%    10M |  99.9 -67.1   28    19M  600    0   0%  81.5M     6n  n/a
   1    1 |  15 2.0   28    89M 1400    0   0%    10M |  99.9 -67.1   28    19M  600    0   0%  81.5M   -38n  n/a
   1    1 |  15 2.0   28    89M 1400    0   0%    10M |  99.9 -67.1   28    19M  600    0   0%  81.5M    16n  n/a

********************************** 40 MHz, 2 antennas, no O-FH optimization **********************************
          |--------------------DL---------------------|-------------------------UL----------------------------
 pci rnti | cqi  ri  mcs  brate   ok  nok  (%)  dl_bs | pusch  rsrp  mcs  brate   ok  nok  (%)    bsr     ta  phr
   1    1 |  15 2.0   27   129M  736   62   7%    10M |  99.9 -68.1   28    33M  462   51   9%  81.5M    -7n  n/a
   1    1 |  15 2.0   27   162M  924   44   4%    10M |  99.9 -68.0   28    38M  535   40   6%  81.5M   -16n  n/a
   1    1 |  15 2.0   27   132M  757   71   8%    10M |  99.9 -68.1   28    34M  483   56  10%  81.5M   -25n  n/a
   1    1 |  15 2.0   27   164M  934   53   5%    10M |  99.9 -68.1   28    37M  513   39   7%  81.5M    32n  n/a

************************************ 100 MHz, 4 antennas, optimized O-FH ************************************
          |--------------------DL---------------------|-------------------------UL----------------------------
 pci rnti | cqi  ri  mcs  brate   ok  nok  (%)  dl_bs | pusch  rsrp  mcs  brate   ok  nok  (%)    bsr     ta  phr
   1    1 |  15 4.0   27   1.2G 1400    0   0%    10M |  99.9 -66.5   28   116M  600    0   0%  81.5M   -21n  n/a
   1    1 |  15 4.0   27   1.2G 1400    0   0%    10M |  99.9 -66.4   28   116M  600    0   0%  81.5M    10n  n/a
   1    1 |  15 4.0   27   1.2G 1400    0   0%    10M |  99.9 -66.5   28   116M  600    0   0%  81.5M    -9n  n/a
   1    1 |  15 4.0   27   1.2G 1400    0   0%    10M |  99.9 -66.5   28   116M  600    0   0%  81.5M     9n  n/a
```

FIGURE 15. SRSRAN TRACES FOR DIFFERENT CELL CONFIGURATIONS: 20 MHZ, 2 ANTENNAS, NO O-FH OPTIMIZATION (TOP), 40 MHZ, 2 ANTENNAS, NO O-FH OPTIMIZATION (CENTER), 100 MHZ, 4 ANTENNAS, OPTIMIZED O-FH (BOTTOM).

Although there have been several improvements in the O-FH library, three points are worth highlighting. The first is the adoption of the Data Plane Development Kit (DPDK), a bundle of libraries and controllers that allows off-loading the process of network data packets from the operative system (OS) kernel to the user space, thus achieving higher throughputs over the Ethernet interface (see code at https://github.com/srsran/srsRAN_Project/tree/main/lib/ofh/ethernet/dpdk). In other words, O-FH packets from the DU to the RU can be written directly to the DPDK buffers (*mbufs*) and, from there, to the Ethernet port without forcing a context switch between user and kernel space. Similarly, when receiving packets from the RU, the O-FH deserializer reads the packets directly from the DPDK *mbufs* and no intermediate copies are needed.

A second relevant contribution to the optimization of the O-FH library is the SIMD-based implementation of the compression/decompression algorithms (code available at https://github.com/srsran/srsRAN_Project/tree/main/lib/ofh/compression). Indeed, given the huge amount of data that needs to be sent in both directions over the fronthaul, the O-RAN specifications define a number of compression mechanisms to relieve the throughput requirements. To reduce the computational costs of compression, the SRS O-FH library relies on SIMD (Single Instruction, Multiple Data) instructions. These are a set of CPU-specific instructions that allow the parallelization of operations performed on arrays of data. Specifically, the compression algorithms have been SIMD-optimized for x86 processors (both AVX2 and AVX512 SIMD instruction sets) and ARM Neon. As an example of the benefits of parallelization for the compression/decompression step, Figure 16 compares the compressing/decompressing rate, in megasamples per second, of one of the most commonly used O-RAN compression schemes, namely *BFP-9b*. Focusing on the 50-th percentile (the median value), we measure a 20-time gain for the encoder and a 25-time gain for the decoder, approximately.

```
************************************** generic ***************************************
 Percentiles:           |  50th  |  75th  |  90th  |  99th  | 99.9th |99.99th | Worst  |
 BFP-9b compression     |   164.1|   164.0|   163.9|   163.0|   161.7|   160.4|   160.4|
 BFP-9b decompression   |   198.3|   198.2|   198.2|   197.4|   195.7|   195.1|   195.1|


************************************** AVX512 ****************************************
 Percentiles:           |  50th  |  75th  |  90th  |  99th  | 99.9th |99.99th | Worst  |
 BFP-9b compression     |  3333.5|  3320.8|  3308.2|  3283.3|  3078.5|  2585.8|  2585.8|
 BFP-9b decompression   |  4891.3|  4846.1|  4810.5|  4740.9|  4262.1|  2259.8|  2259.8|
```

FIGURE 16. BFP-9B COMPRESSION ALGORITHM BENCHMARK: SAMPLE RATE IN MEGASAMPLES PER SECOND WITHOUT (TOP) AND WITH (BOTTOM) AVX512 SIMD INSTRUCTIONS.

From the network management point of view, now the O-FH library offers the possibility to configure different vLANs for the user-plane and the control-plane traffic between DU and RU. In other words, even though both DU and RU are still equipped with a single network interface, it is now possible to partition the fronthaul transport network into two distinct segments (via the tags vlan_tag_cp and vlan_tag_up in the ru_ofh configuration section), each one optimized according to the specific characteristics of either one of the user and control plane, therefore contributing in terms of flexibility and scalability.

Even though it does not have a direct effect on the performance of the O-FH library, it is worth mentioning here that the user interface has also been noticeably improved, exposing several O-FH related parameters in the srsRAN Project configuration file. For instance, it is extremely simple to fine tune the timings of packet transmission and reception, so as to match the joint requirements of the

RU and the fronthaul transport network. This fact has turned out to be extremely helpful for fine-tuning the set-up in UC2PoC1, one of whose goals is the characterization of deployments with a challenging fronthaul network (see Section 3.1.1). As well, statistics are collected for end-to-end debugging that show the fraction of packets received before (early) or after (late) the configured time windows, as the ones showed at the beginning of this section. The relevant section of the configuration file is reported below:

```
ru_ofh:
  gps_alpha: 0
  gps_beta: 0
  ru_bandwidth_MHz: 20
  t1a_max_cp_dl: 500
  t1a_min_cp_dl: 258
  t1a_max_cp_ul: 500
  t1a_min_cp_ul: 258
  t1a_max_up: 300
  t1a_min_up: 85
  ta4_max: 300
  ta4_min: 85
  is_prach_cp_enabled: true
  is_dl_broadcast_enabled: false
  ignore_ecpri_seq_id: false
  ignore_ecpri_payload_size: false
  warn_unreceived_ru_frames: true
  compr_method_ul: bfp
  compr_bitwidth_ul: 9
  compr_method_dl: bfp
  compr_bitwidth_dl: 9
  compr_method_prach: bfp
  compr_bitwidth_prach: 9
  enable_ul_static_compr_hdr: true
  enable_dl_static_compr_hdr: true
  iq_scaling: 0.35
  cells:
       -
       network_interface: enp1s0f0
       ru_mac_address: 70:b3:d5:e1:5b:06
       du_mac_address: 00:11:22:33:00:77
       vlan_tag_cp: 1
       vlan_tag_up: 1
       ru_prach_port_id: [4]
       ru_dl_port_id: [0, 1]
       ru_ul_port_id: [0]
log:
```

```
ofh_level: warning
```

More examples of configuration files and O-FH logs, as well as a full performance analysis of the improvements described in this section, are available online at 6G BLUR repository [7].

2.4.3 Conclusions

The use of advance networking (e.g., DPDK) and processing (e.g., SIMD instructions) features has allowed the boosting of the performance of the O-FH library provided by srsRAN software. This is allowing the configuration of a wide range of configurations for the RU-DU. Before this improvement, the maximum supported configuration was a 20-MHz 2-antennas. After 6G BLUR, the system supports configuration of 10MHz 4-antennas, if appropriate underlying HW for the DU is available. One additional feature developed in the O-FH is the capability of easily configuring the timings of packet transmission and reception, which is relevant to match the joint requirements of the RU and the fronthaul transport network.

2.5 SMART-K2.2: Characterization of the FH and MH transport

2.5.1 System design

Unlike conventional RAN networks, Open RAN architecture introduces a functional split within the baseband unit, dividing it into distinct processing entities: the Centralized Unit (O-CU) and the Distributed Unit (O-DU). Each of these components handles specific tasks within the RAN protocol stack, with their separation determined by various functional split options, being the 7.2x split the one standardized by the O-RAN Alliance.

In most Open RAN deployments, a distributed model is typically adopted, where the O-RU and O-DU are co-located at the cell site. Meanwhile, the O-CU can either be positioned at the same site or deployed in a more centralized location, depending on network requirements and design considerations.

[7] Available online at: https://gitlab.cttc.es/mdi-6gblur/smart/-/blob/main/KeyConcepts/SMART-K2.1%20Development%20and%20Characterization%20of%20FH%20and%20MH/O_FH_measurements.md.

As networks continue to evolve to accommodate more demanding services and enhance operational efficiency, there is an increasing trend toward adopting a more centralized architecture, based on the specific requirements of the services being provided.

Centralizing the Open RAN architecture brings several advantages but also poses some challenges. For that reason, it is needed to gain insights from a theoretical and practical perspective on the extent to which network can be centralized without degrading the user performance, and that means analysing timing and capacity under those network conditions. The goal in this key concept is to characterize specially the FH with the aim of study its behaviour under a centralized topology. This characterization allows to know and understand the properties of the links between the involved equipment (O-RU, O-DU and O-CU) by measuring some qualitative and quantitative capabilities. This process is fundamental to properly design and optimize a network, its topology, its quality, its stability in time, etc.

Thus, several parameters and attributes of these links have been defined and described, as well as different transport conditions with several patterns, priorities, impairments, and other configurations. In this sense, several KPIs have been studied, like throughput, latency, and possible impairments that may arise in certain situations in the network, like jitter, PDV (Packet Delay Variation), or others, as well as possible issues that may occur related with the capacity, reliability, and scalability of the links [20].

Although the objective should be to characterize both, FH and MH, this study focuses on FH characterization due to its complexity and sensitivity to all the described factors above. MH links are less stringent in terms of time and capacity, so it is easily feasible to support over any existing transport network.

Hence, the workplan for this topic is done in two main stages:

(1) An analytical phase, to analyse the timing relationships in the FH link to yield meaningful conclusions such as what time windows should be applied at the O-DU and O-RU side so that the constraints imposed by the transport network in terms of latency and traffic (capacity) does not affect to the feasibility of the centralization. This serves as an input for the network dimensioning and optimization and also O-DU SW stack enhancements.

(2) An experimental phase that has been performed in 6G-SANDBOX laboratory premises [21] with an E2E (end-to-end) testbed composed by the network entities involved in the analytical phase (O-RU, O-DU, O-CU), in which also a network emulator has been installed in the FH to emulate different FH profiles with a specific parametrization in the links to be representative from the scenarios analyzed in phase (1). Longer delays, impairments, QoS/traffic prioritization were added allowing to emulate in a fair way a real transport network.

In this testbed, it has set-up an O-RU - O-DU connection to analyse empirically the time windows resulted from the analytical phase. The ultimate goal of this practical phase is to confirm and validate the theoretical analysis, so that it can be empirically demonstrated by laboratory testing that the centralization of the architecture can be realized under a non-ideal FH network. An extensive detail of this analysis can be found in the section developing UC2PoC1 activity.

For the time windows analysis, it has been developed a software tool in MATLAB that allows to automize the analysis of the feasibility of centralization scenarios attending to the time delay imposed by different transport networks and O-RU characteristics. This MATLAB tool can be found in the 6G BLUR repository[8] and it is also exploited within the development of K2.4 related with Radio Stack optimization. The aim is to model several circumstances where transport network conditions are different and to extract the minimum and maximum allowable delays at FH and the O-DU timing in order to support those centralization distances. This outcome could be considered later for other activities related to the optimization of transport network and O-DU SW stack enhancements.

In constructing the scenario, the dimensioning of the network elements was carefully considered, factoring in the maximum aggregate capacity under the most challenging conditions. This was based on the highest capacity supported by the network links, aggregation routers, and the virtual O-DU pool capacity, explained later.

Thus, the outcome of this analysis is twofold:

(1) To determine the feasibility of centralization under different scenarios concluding on timing relationships need to be applied to both transport and O-DU sides to achieve such centralization level.
(2) To optimize network dimensioning considering the maximum capacity needed to be supported by such centralization (number of cells, sites aggregated per centralized O-DU, capacity of links, etc.).

2.5.2 Implementation

The SW tool has three main inputs for the time delay analysis: O-RU fixed minimum and maximum processing times; O-DU timing; and transport NW characteristics.

The network considered is Telefónica de España´s IP/Fusion transport network, an optical IP-based network that allows all the traffic from different access technologies to be incorporated into a single

[8] Available online at: https://gitlab.cttc.es/mdi-6gblur/smart/-/tree/main/KeyConcepts/SMART-K2.2%20Characterization%20of%20the%20FH%20and%20MH%20transport

network infrastructure, independent of access technology, thus combining traffic from different services such as mobile access, fixed access, business and others in Spain. This transport network features hierarchical structure with five hierarchical levels (HL) formed by HLx routers in a star topology. These levels are:

- HL5: the lowest level, consisting of the Cell Site Router (CSR) located at cell site.
- HL4: is composed of routers located at the lower layer of the IP network and performs functions such as traffic classification and data routing to the destination, among others.
- HL3: performs the traffic aggregation / distribution function. It collects traffic from the different geographical areas of the network.
- HL2: it does not aggregate traffic, but it is responsible for forwarding traffic to a node of equal hierarchy or to an HL1 node.
- HL1: the highest level, corresponding to the nodes of the backbone network at the national level. This level is also the interface between the IP network and the international Internet provider.

A high-level scheme of a typical aggregation network with the different aggregation levels can be seen in Figure 17.

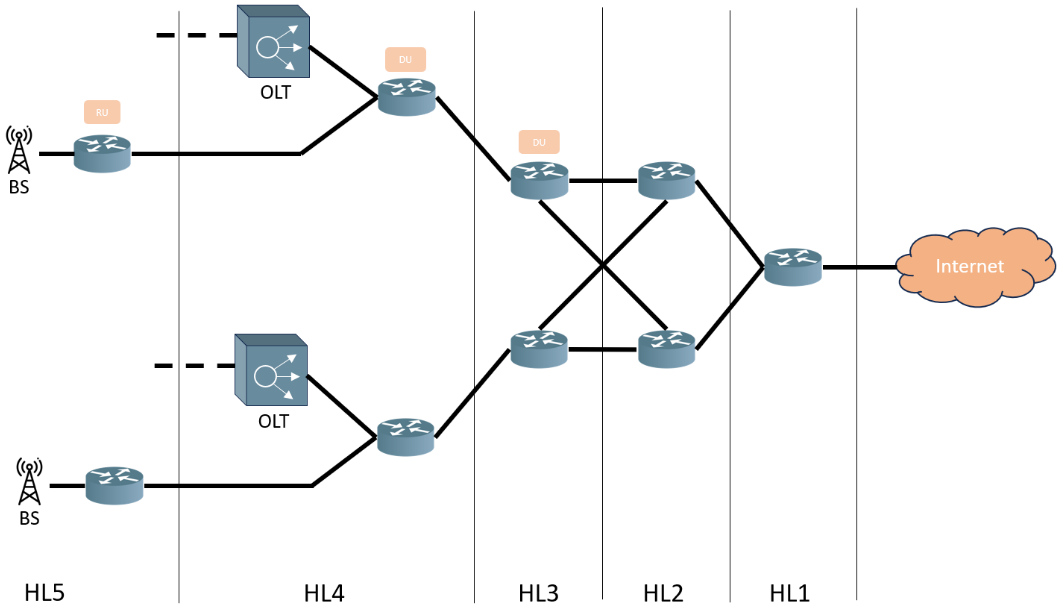


FIGURE 17. HIGH-LEVEL ARCHITECTURE OF AN AGGREGATION NETWORK

More specifically, our research focuses on the HL3-HL4-HL5 levels. A simpler architecture is depicted on Figure 18. O-RU will be located in HL5 at cell site as usual, and depending on the centralization distance achieved, O-DU will be located either on HL4 or HL3. In order to guarantee maximum service availability, the links between levels are redundant, with each HL5 router having two connections to HL4 routers, and each HL4 router having also two connections to HL3 routers.

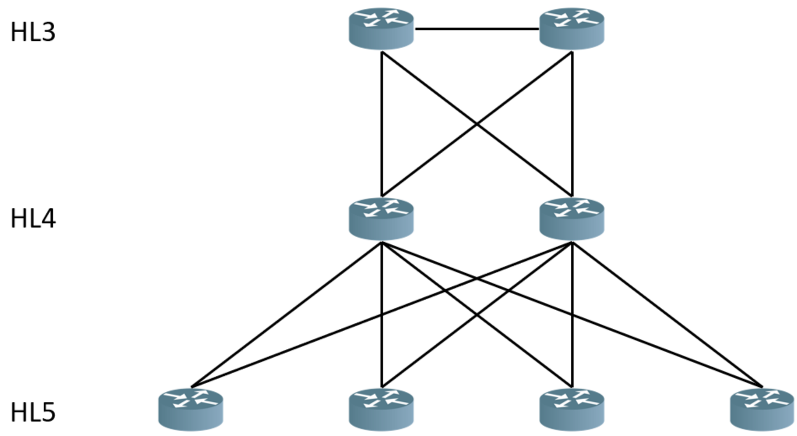


FIGURE 18. STAR TOPOLOGY BETWEEN HL5-HL3 LEVELS IN THE AGGREGATION NETWORK

The dimensioning is based on the existing ratio between the different levels of the aggregation network in Spain, being on average 9 HL5 (cell sites) for each HL4 and 23 HL5 for each pair of HL3. From this, the capacity scenario has been defined through estimations of the maximum possible aggregated traffic in the case of higher congestion, with the objective of knowing the aggregated traffic by a single O-DU and the requirements of the network routers and network links. This allows to know not only if the centralization distances obtained are achievable, but also if the aggregate capacity in the network is bearable, serving as input to analyse which aspects should be modified or optimized in the transport network in order to make the scenario feasible in terms of time and capacity.

To this end, the traffic generated at each site (tri-sectorial) has been calculated for the C-band (3500 MHz), considering pooling gains with one cell in peak capacity and two on average, with the configuration shown in Table 5.

TABLE 5. CELL PARAMETERS

NR panel configuration	DL	UL
Number of Tx/Rx antennas	32	32
Number of bands	1	1
Number of carriers	1	1
Channel bandwidth (MHz)	100	100
Modulation (QAM): number of symbols	256	256

MIMO: number of layers	16	8
Sub-carrier spacing (kHz)	30	30

Applying Block Floating Point (BFP) compression with 9-bit Mantissa per symbol and 8-bit exponent results in around 23 Gbps per sector/cell in U-Plane, that being approximately 70 Gbps in DL and 28 Gbps in UL generated per site/HL5.

That leads to two capacity scenarios: on the one hand, when the O-DU is located at HL4 it aggregates 9 sites and up to 630 Gbps; and on the other hand, when the O-DU is located at HL3 it aggregates up to 23 sites and up to 1610 Gbps. As the link's capacity has to be dimensioned to the upper hundred, that translates to have a capacity of 100 Gbps in HL5-HL4 links, 700 Gbps in HL4-HL3 links and 1700 Gbps in HL3-HL3 links. Both scenarios are depicted in Figure 19.

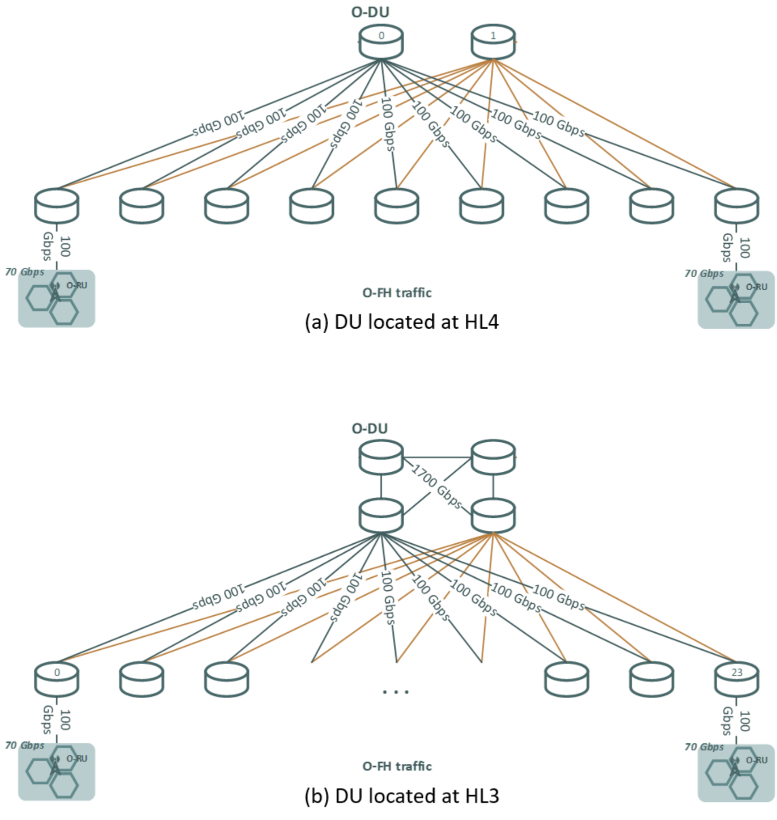


FIGURE 19. CAPACITY SCENARIOS

In the time domain, there are several timings to consider in this network to model the transmission. The parameters that will affect the transmission and, in turn, will affect to the centralization, are:

- Physical distance between HLx routers, i.e., propagation time, considering 5 μs/km.
- Routers delays: composed of several sub-timings:
    - Switching delay: based on the capacity of the router commutation matrix. For the different levels of the network, we have considered the following timing (calculated for a single packet of 273 PRB * 12 subcarriers * 18 Mantissa-bits, that is, 60 kb):
        - HL3:
            - Commutation matrix: 4 Tbps → Switching time: 0.015 μs.
        - HL4:
            - Commutation matrix: 2 Tbps → Switching time: 0.03 μs
        - HL5:
            - Commutation matrix: 300 Gbps → Switching time: 0.2 μs.
    - Transmission time: the time that takes for a packet to be sent through the network card. For a packet of 60 kb in a 100 Gbps port, that means 0.6 μs.
    - Input/Output queueing delay: the jitter originated in the routers input and output queues. To extract its values, an HL4 setup was implemented in Telefónica Technology and Automation Lab using a commercial router with a capacity of 4.8 Tbps, nine 100 Gbps links to HL5, and seven 100 Gbps links to HL3 (making link bonding), along with a traffic generator injecting the DL and UL traffic commented above. By analysing scenarios with injected capacities of 630, 700, and 701 Gbps to reach saturation, and using packet sizes of 128, 512, and 1024 bytes, the results showed an average delay of 10 μs in all cases, with jitter increasing with the frame sizes but below 0.5 μs. It was concluded that these results could be extrapolated to other HLx levels, as variations in aggregated capacity are accompanied by corresponding adjustments in the switching matrix capacity. So, for HL5/HL4/HL3 routers, in each of them an approximately constant delay of 10 μs is considered, given that, despiteO-FH traffic can be expected to be carried over a multi-service network together with other sources of traffic, such as fixed services, O-FH packets are considered to be assigned with the highest priority due to their stringent timing requirements. This would ensure that this traffic remains unaffected by the rest, resulting in an almost constant delay and jitter.

- ToR switching times: commutation time of the ToR (Top Of Rack) switches located in each hierarchical level of the network. It has been considered 3 switches per level, each one introducing 2 μs of delay.

So, all of these parameters have been considered in the MATLAB SW tool to model a transport network similar to the Telefónica´s network topologies as depicted above in Figure 17. Nevertheless, for the sake of having nation-wide input of the typical distances between HL5 and HL3, it was considered the linear distance between these two routers instead of the physical distance. We are aware this would result in loss of accuracy for the scenarios, but it would, in turn, allow to model a wider network as getting the physical distance between these two hops nation-wide would result impractical and out of scope for this project.

Linear distances between HL3 – HL5 routers were modelled based on its typical distribution in Telefonica's Spain transport network, thus the software developed allows to select three different distance ranges among their histogram distribution for the two O-DU locations (O-DU at HL4 and O-DU at HL3), comprising the most common cases, that are shown in Table 6.

TABLE 6. CENTRALIZATION DEGREES

Centralization degree	Properties
O-DU at HL4	Number of aggregated cell sites per vDU pool: ~ 9
	Shortest case: 10 – 15 km
	Average case: 15 – 20 km
	Longest case: 20 – 25 km
O-DU at HL3	Number of aggregated cell sites per vDU pool: ~ 23
	Shortest case: 35 – 45 km
	Average case: 45 – 55 km
	Longest case: 55 – 65 km

Therefore, having explained in general terms the tool developed, its operation will now be explained in more detail. To do so, we will begin by explaining the time windows analysis, which, as mentioned above, it is based on the O-RAN Alliance standards:

- CUS-planes specification [22]:
    - Section 4.4: latency requirements.
    - Section 4.6: transmission windows.
    - Section 4.7: O-RU external antenna delay handling.
    - Section 4.8: UL frame timing to DL frame.
    - Annex B: delay management use cases.

Financiado por la Unión Europea NextGenerationEU
Plan de Recuperación, Transformación y Resiliencia
UNICO I+D
GOBIERNO DE ESPAÑA
cttc Centre Tecnològic de Telecomunicacions de Catalunya

- M-plane specification [23]:
    - Section 7.7: C/U-plane delay management.
    - Section 7.8: O-RU adaptative delay capability.

The latency boundaries in DL and UL between the O-DU and O-RU are based on eCPRI reference points as specified in eCPRI specification [24]. These points, shown in Figure 20, are:

- O-DU: transmit/receive interface at O-DU.
- O-RU: transmit/receive interface at O-RU.
- Ra: antenna interface at O-RU.

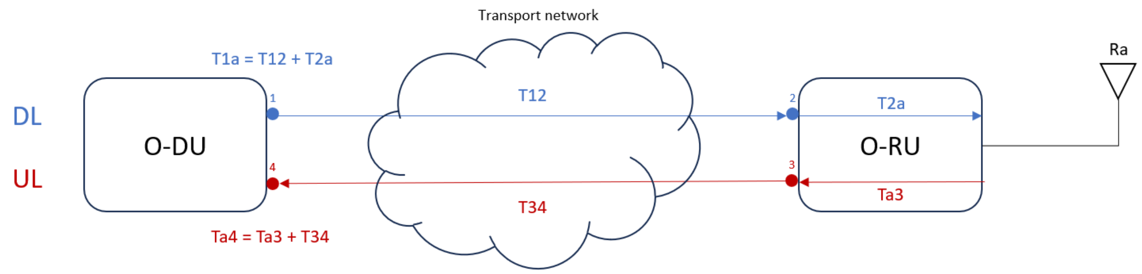


FIGURE 20. REFERENCE POINTS FOR DELAY MANAGEMENT

Transmission delays between O-DU and O-RU due to FH network are specified as *T12* in downlink and *T34* in uplink. As in real Ethernet networks, there are impairments that modify these delays causing some variation, so transport delay shall be considered as a range defined by an upper and lower bound. The minimum delay corresponds to the combination of the propagation time (function of centralization distance) and the routers switching time (HL3-HL4-HL5, ToR and others). The maximum delay is the minimum delay with the addition of the network impairments, changing depending on the conditions of the network and its level of optimization. The mathematical expressions of the minimum and maximum delay found in the FH can be seen in Equation *(1)* and Equation *(2)*, respectively.

$$t_{FH\_min} = t_{prop} + t_{switching} \tag{1}$$

$$t_{FH\_max} = t_{FH\_min} + t_{NW\_impairments} \tag{2}$$

This can be seen visually in Figure 21 (numbers are not representative).

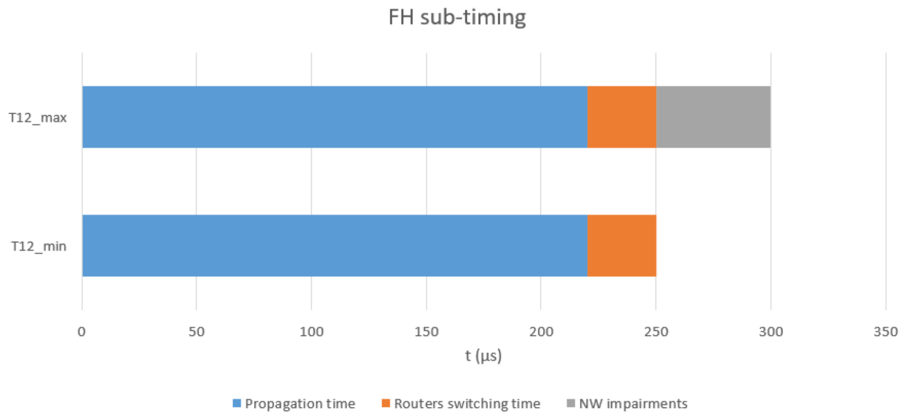


FIGURE 21. DIFFERENCES IN SUB-TIMING BETWEEN FH MAXIMUM AND MINIMUM DELAYS

The propagation time corresponds to the time it takes to the signal to propagate to one point to another through a fibre with a refraction index of 1.5, that is, 5 μs/km. The routers switching time includes the HL3-HL4-HL5 switching time and the ToR switches switching. Additionally, there exists other impairments that may be added to the routing part (transport segment) which are result of the network variability, queueing traffic, etc., This has been object of study as part of the network optimization with the goal of gaining extra time for further centralization.

Regarding O-RU processing times, defined by *T2a* in downlink and *Ta3* in uplink as depicted in Figure 20, shall be considered to be a factor in the equation to derive O-DU timing windows. In this case, these values minimum and maximum delays will be considered fix, as there is mainly resulting from O-RU HW design and there is not intention to change them. The accuracy of the reported O-RU delay times values shall be within a maximum range of 200 ns, according to typical values found in consulted O-RU specifications. Fixed timing at Ra is also required, serving as a reference point for delay management in the eCPRI protocol with transmission and reception points measured relative to Ra. The transmission from the O-RU to the antenna delay is considered negligible.

In this study, both C-Plane and U-Plane are considered related through the *Tcp_adv_dl* variable which defines the timing advance of the control plane over user plane to support coordination.

With all these parameters it is possible to define now the transmission window (TW) and reception window (RW) of O-DU and O-RU considering the imposed FH timing from the network characteristics and the O-RU fixed values. They represent the time range in which both equipment can send and receive data. It is only a representation of the mathematical boundaries imposed by O-RU (as it is a fixed processing time) and transport constraints (scenarios under analysis). So, the TW is then defined as the maximum amount of time allowed for the transmitter to send all the data based on O-RU timings and transport conditions. Likewise, RW is defined as the maximum amount of time allowed for the receiver to collect all the data on time. These windows allow to guarantee that all the samples for a specific symbol in a certain TTI are transmitted at point *Ra* at the required time.

Nonetheless, the objective of this study is not only to derive the TW and RW windows but also to adjust them so that FH can be extended achieving larger centralization distances. Figure 22 allows to explain the concept on how these windows can be adjusted to gain centralization distance, where the timing for O-RU, O-DU and FH for both UL and DL directions are presented. As O-RU transmission and reception windows are fixed, in order to achieve longer delays it is needed to make some modifications on transport network latencies or O-DU windows to accommodate this extra time but still fulfilling the timing relationships (Figure 20). Since in the O-DU it is possible to adapt its windows as desired to match them with the imposed O-RU timing and transport delays, their start and end points can be displaced in time as desired to fulfill the requirements. In this sense, in downlike, the start of the O-DU TW can be increased to gain more time (distance) in FH interface, advancing in time up to the maximum value supported by the O-DU capacity. Also, in uplink, the end of the O-DU RW can be prolonged, translating in more time (and distance) for FH.

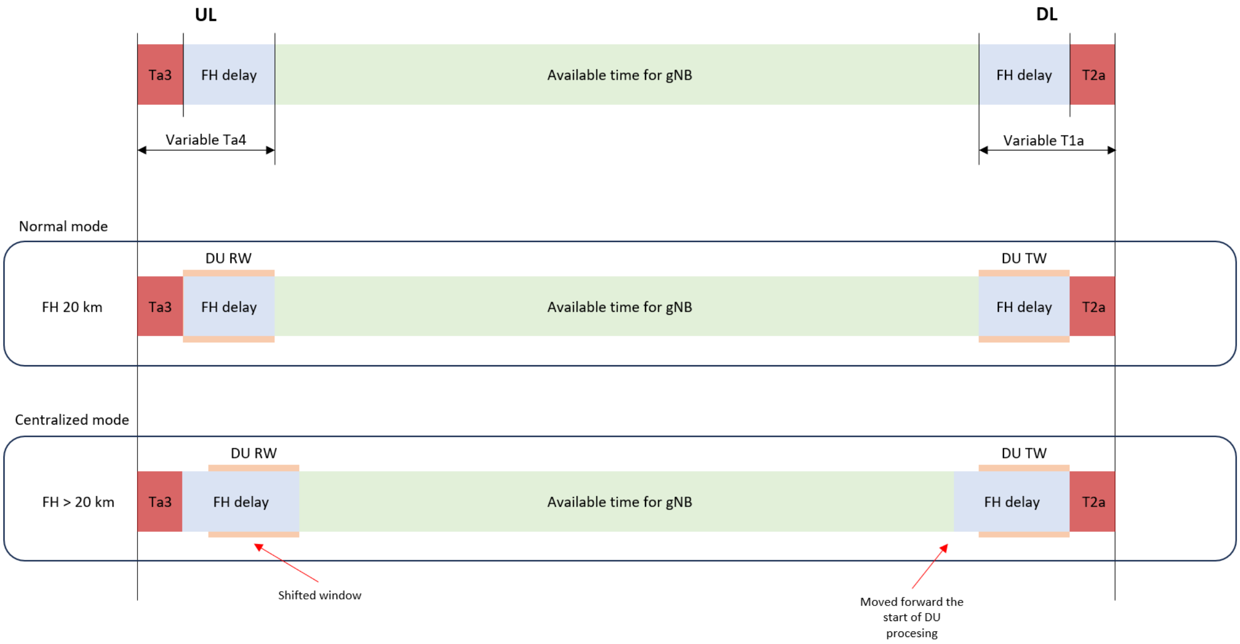


FIGURE 22. CONCEPTUAL REPRESENTATION OF O-DU TX AND RX WINDOWS ADJUSTMENTS TO ACHIEVE LARGER CENTRALIZATION

Within the O-DU TW, it is defined the maximum O-DU transmission time, *TXmax*, that is, the minimum value the O-DU needs to transmit a symbol in the worst case and can be shifted within the window (*T1a_max* – *T1a_min*). All the concepts commented above are listed in the following tables, with their description and their calculations or relations they shall meet.

In downlink direction, Table 7 shows FH concepts involving U-Plane in downlink, while Table 8 shows the same concepts but in C-Plane.

TABLE 7. U-PLANE DL DELAY PARAMETERS

U-Plane DL parameter	Description	Condition
T1a_max_up_O-DU	Maximum *T1a_max_up supported by O-DU capacity*	*T1a_max_up_O-DU* ⩾ *T1a_max_up*
T1a_max_up	Earliest time the O-DU can transmit	≤ (*T12_min* + *T2a_max_up*) OR ≥ (*T1a_min_up* + *TXmax*)
T1a_min_up	Latest time the O-DU can transmit	⩾ (T12_max + T2a_min_up)
T2a_max_up	Earliest time the O-RU can receive data	Fixed value depending on HW design provided by vendor
T2a_min_up	Latest time the O-RU can receive data	Fixed value depending on HW design provided by vendor
T12_max	FH maximum delay in DL	It depends on transport network
T12_min	FH minimum delay in DL	It depends on transport network
TXmax	Maximum O-DU transmission time	It depends on the radio carrier and technology
O-DU TW	Interval in which the O-DU can transmit	≥ *TXmax*
O-RU RW	Interval in which the O-RU can receive	*T2a_max_up* – *T2a_min_up*

TABLE 8. C-PLANE DL DELAY PARAMETERS

C-Plane DL parameter	Description	Condition
Tcp_adv_dl	Time advance of C-Plane over U-Plane	Fixed value
T1a_max_cp_dl	Earliest time the O-DU can transmit	≥ (*T1a_max_up* + *Tcp_adv_dl*)

Financiado por la Unión Europea NextGenerationEU
Plan de Recuperación, Transformación y Resiliencia
UNICO I+D
GOBIERNO DE ESPAÑA
cttc Centre Tecnològic de Telecomunicacions de Catalunya

T1a_min_cp_dl	Latest time the O-DU can transmit	(*T1a_min_up* + *Tcp_adv_dl*)
T2a_max_cp_dl	Earliest time the O-RU can receive	(*T2a_max_up* + *Tcp_adv_dl*)
T2a_min_cp_dl	Latest time the O-RU can receive	(*T2a_min_up* + *Tcp_adv_dl*)
T12_max	DL FH maximum delay in DL	Same as U-Plane DL
T12_min	DL FH minimum delay in DL	Same as U-Plane DL
TXmax	Maximum O-DU transmission time	Same as U-Plane DL
O-DU TW	Interval in which the O-DU can transmit	Same as U-Plane DL
O-RU RW	Interval in which the O-RU can receive	Same as U-Plane DL

In uplink direction, Table 9 shows concepts involving U-Plane in uplink, while Table 10 shows the same but in C-Plane, this latter being equal to C-Plane in downlink.

TABLE 9. U-PLANE UL DELAY PARAMETERS

U-Plane UL parameter	Description	Condition
Ta3_max	Latest time the O-RU can transmit	O-RU fixed value provided by vendor
Ta3_min	Earliest time the O-RU can transmit	O-RU fixed value provided by vendor
Ta4_max	Latest time the O-DU can receive	≥ (*Ta3_max* + *T34_max*)
Ta4_min	Earliest time the O-DU can receive	≤ (*Ta3_min* + *T34_min*)
T34_max	DL FH maximum delay in UL	It depends on transport network
T34_min	DL FH minimum delay in UL	It depends on transport network
O-RU TW	Interval in which the O-RU can transmit	*Ta3_max* – *Ta3_min*
O-DU RW	Interval in which the O-DU can receive	≥ (*Ta4_max* – *Ta4_min*)

TABLE 10. C-PLANE UL DELAY PARAMETERS

C-Plane UL parameter	Description	Condition
T1a_max_cp_ul	Earliest time the O-DU can transmit	*T1a_max_cp_dl*

Financiado por la Unión Europea NextGenerationEU
Plan de Recuperación, Transformación y Resiliencia
UNICO I+D
GOBIERNO DE ESPAÑA
cttc Centre Tecnològic de Telecomunicacions de Catalunya

T1a_min_cp_ul	Latest time the O-DU can transmit	*T1a_min_cp_dl*
T2a_max_cp_ul	Earliest time the O-RU can receive	*T2a_max_cp_dl*
T2a_min_cp_ul	Latest time the O-RU can receive	*T2a_min_cp_dl*
T12_max	DL FH maximum delay in DL	Same as DL
T12_min	DL FH minimum delay in DL	Same as DL
O-DU TW	Interval in which the O-DU can transmit	Same as C-Plane DL
O-RU RW	Interval in which the O-RU can receive	Same as C-Plane DL

ForO-RU parameters, as they are confidential and depend on each radio vendor, they have been defined following the ranges defined in the Fronthaul Test Specification [25] by O-RAN Alliance, being *T2a_min_up* = 134 μs and *T2a_max_up* = *375* μs for DL; and Ta3_min = 50 μs and *Ta3_max* = *171* μs for *UL*.

For the O-DU in U-Plane in DL, there are several critical timing considerations that must be addressed to ensure proper data handling. Before the start of the O-DU TW, the O-DU executes the entire processing chain to process the data slots from L2 prior to sending the symbols through the FH. This processing consists of:

- MAC scheduler.
- FAPI processing.
- High-PHY processing.
- FH processing.
- O-DU Buffer.

Once the data has completed the processing pipeline, the O-DU sends the symbols through the fronthaul with strict adherence to time windows, which are essential for maintaining temporal relations with the O-RU. Any deviation in this timing could lead to issues such as symbol misalignment, increased latency or data packet loss. Therefore, the O-DU must manage and optimize these timings carefully to support high-throughput and low-latency communication.

Within the FH, the O-DU has other parameters to take into account:

- Optimum O-DU TW: it is a trade-off, and its size is usually multiplot of multiple of several symbol times. We considered having a 6-symbol time window.
- Transmission time (*TXmax_DU*), which is the time the network interface card (NIC) of the O-DU server takes to send a symbol within the O-DU TW.
- Offset, the time between the start of the O-DU TW and the actual sending of a symbol.

To understand better all these timing, Figure 23 is exposed. Vertical lines symbolize the symbol time, horizontal lines represent O-DU TW and the black squares represent *TXmax_DU*. So, after the O-DU has processed all the slot, it accumulates the symbols in the O-DU buffer ready to be sent. In each symbol time, the O-DU sees if it has symbols to be sent and if the O-DU TW is open to actually send them. For example, in this case, it can send up to 3 symbols.

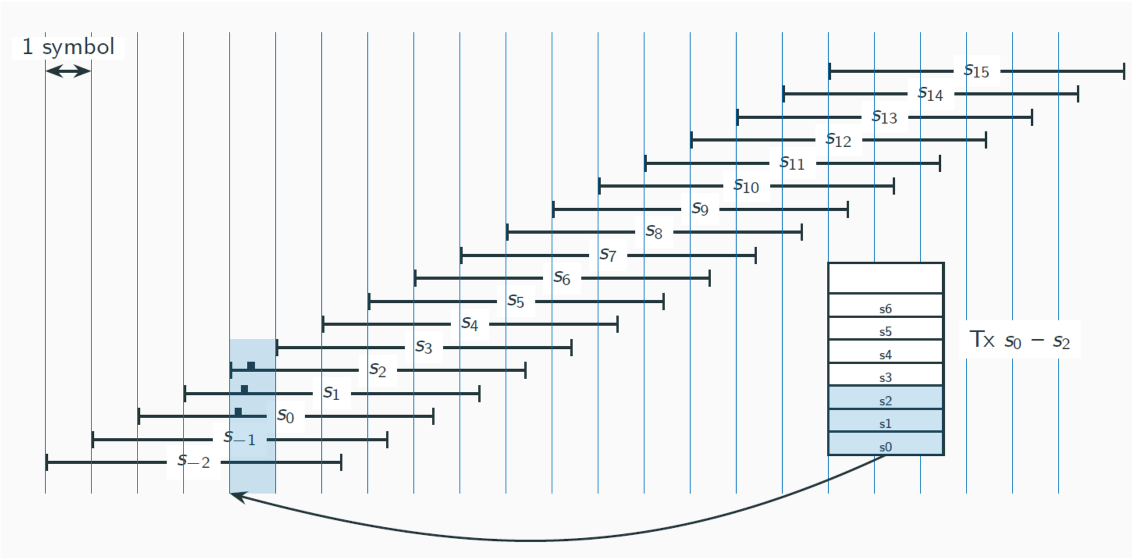


FIGURE 23. HOW O-DU TW WORKS

Therefore, in DL the O-DU transmits data within the time range defined by its transmission window so that the O-RU could receive it within the timeframe defined by its reception window as follows (Equations *(3)* and *(4)*):

$$O-DU\ TW = T1a\_\max - T1a\_\min = (T12\_\min + T2a\_max) - (T12\_\max + T2a\_min) \quad \textbf{(3)}$$

$$O-RU\ RW = T2a\_\max - T2a\_min \quad \textbf{(4)}$$

in which the O-RU RW will be larger than the O-DU TW, so that RW ≥ (TW + *DL_variation*).

DL_variation establishes the allowable gap between *T12_min* and *T34_min* with respect to T12_max and T34_max, respectively, for a given O-DU and O-RU equipment. This comes from the necessity that the O-DU TW shall be greater than its transmission time, that is, (T1a_max_up – T1a_min_up) ≥ TXmax. All of this derives in the following constraint that needs to be satisfied during the establishment of the timing windows (Equation 5 and depicted in Figure 24). It happens similarly in UL.

$$DL_{variation} = T12\_\max - T12\_\min \leq (T2a_{max} - T2a_{min}) - TXmax \quad \textbf{(5)}$$

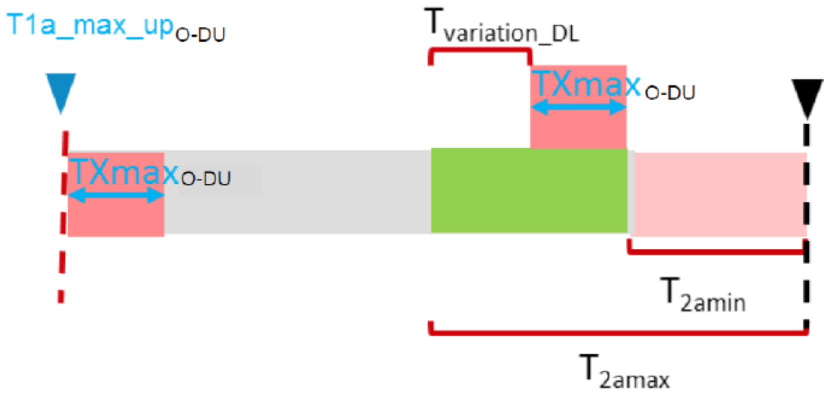


FIGURE 24. DL TRANSPORT VARIATION [26]

For this reason, in order to guarantee that the O-DU TW is large enough, that is, *T1a_max_up* >= (*T1a_min_up* + *TXmax*), the O-RU maximum processing time shall be greater than a certain value, following the relation of the Equation *(6)*:

$$T2a_{max} \geq T2a_{min} + (T12_{max} - T12_{min}) + TXmax \quad \textbf{(6)}$$

All these relations can be seen more visually in the next figures to ease the understanding. Figure 25 shows the DL with the U-Plane diagram, where it can be seen that there are two limitations given the above conditions:

- *T12_min*: a symbol sent at the start of O-DU TW shall arrive later than the start of O-RU RW.
- *T12_max*: a symbol sent at the end of O-DU TW shall arrive earlier than the end of O-RU RW.

Parameters are always referenced to t_{DL}, that is the transmission at antenna port of the earliest IQ sample in time domain within a symbol including cyclic prefix. Moreover, it is possible to observe the optimum O-DU TW (dotted square), the variable offset from the start of the O-DU TW to the real start of the symbol transmission and the maximum O-DU transmission time (*TXmax*). In the case of C-Plane, the diagram is the same but *Tcp_adv_dl* units advanced in time.

The sum of timing of O-RU, FH and the timing of the O-DU processing has to be less than *T1a_max_up_O-DU*, the maximum time in advance the O-DU is able to begin processing DL packets. This parameter depends on the processing capacity of the O-DU, with its value being defined as a multiplot of the slot time. A visual representation of all the timing chain from L2 to OTA (over-the-air) is represented in Figure 25.

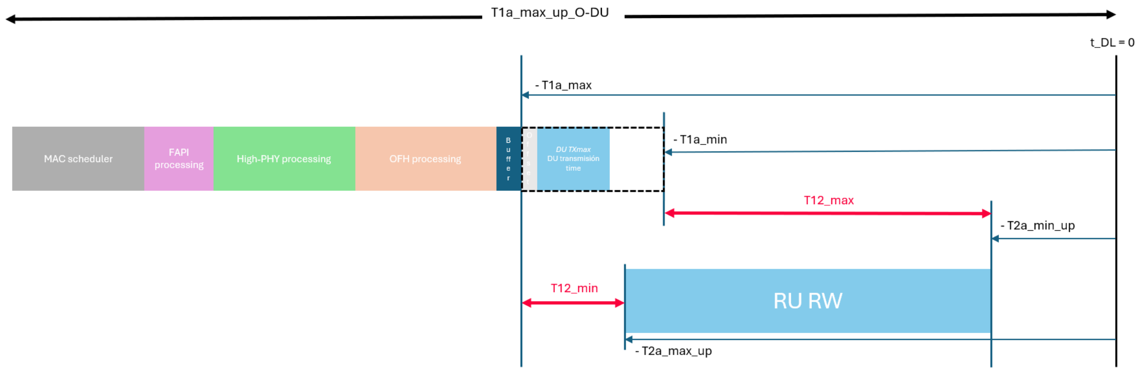


FIGURE 25. U-PLANE DL DIAGRAM

Figure 26 shows the U-Plane diagram in UL, where it can be seen that there are two limitations:

- *T34_min*: a symbol sent at the start of O-RU TW shall arrive later than the start of O-DU RW.
- *T34_max*: a symbol sent at the end of O-RU TW shall arrive earlier than the end of O-DU RW.

Parameters are always referenced to t_{UL}, that is the reception at antenna port of the earliest IQ sample in time domain.

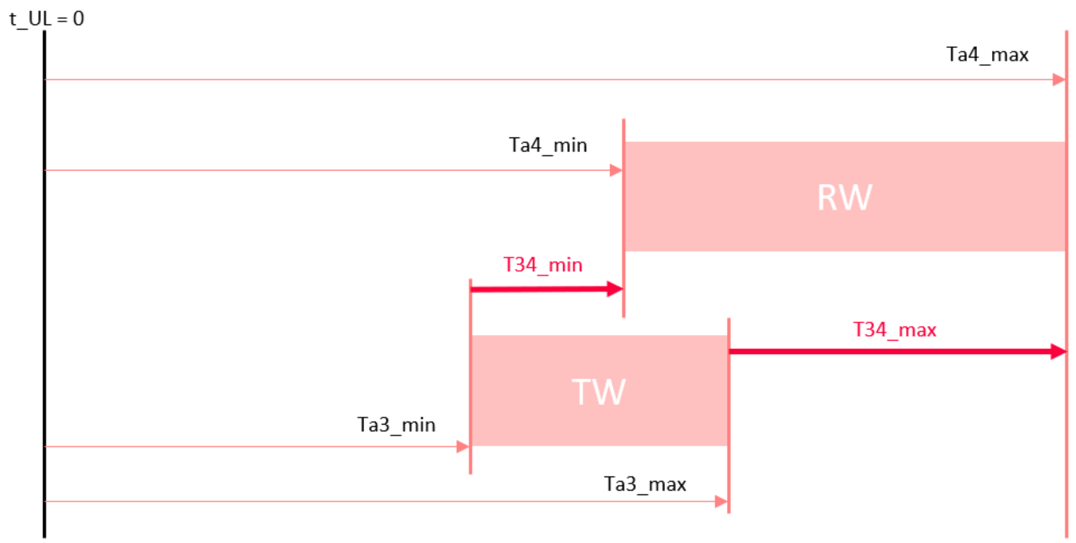


FIGURE 26. U-PLANE UL DIAGRAM

As a summary, the flowchart of the whole developed tool is depicted in Figure 27 to represent in a high level way how this tool works. Once the input data of the O-RU and network characteristics are available, and the user chooses the number of simulations to be carried out, the location of the O-

DU and the distance range, the timing of the resulting scenario is calculated. The SW then shows preliminary results and makes one first pre-analysis: to check if the timing complies with the *T1a_max_up_O-DU* limit or not (in the latter being reduced the propagation time).

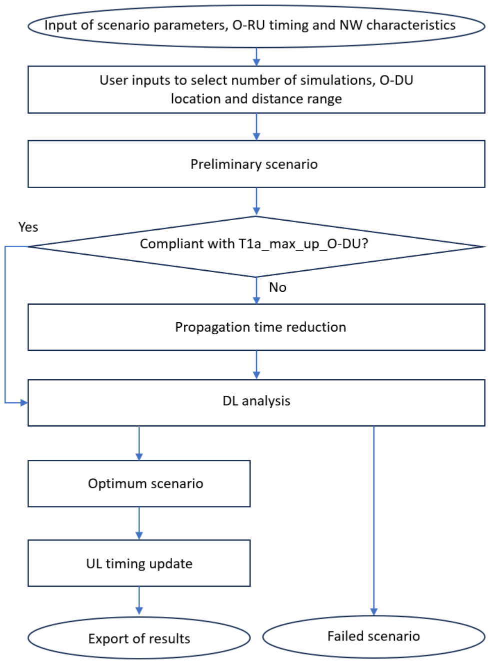


FIGURE 27. FLOWCHART OF THE FH TIMING SIMULATOR

Then, it is performed the DL analysis, whose own low-level flowchart is exposed in Figure 28. If the optimization allows to correct the scenario, timing changes are made in DL and consequently those involved also in UL (FH timing in both directions are the same) and given the scenario is optimum, final results are presented. If not, the SW warns of a failed scenario impossible to adapt to get the desired centralization with those O-RU and network conditions.

Hence, in the timing analysis, once we know the input parameters that will drive the analysis (O-RU timing and transport network characteristics), it is needed to perform an analysis of the feasibility of the scenario to determine whether the centralization is feasible under those assumptions. For that, it is made an analysis only in downlink. The reason is that in downlink the relationships result from the cross-sums of minimum and maximum boundaries, what can cause some issues need to be verified, as for example, the minimum timing (*T1a_min_up*) to be larger than the maximum timing (*T1a_max_up*) if the time conditions are not met. In uplink, this issue does not happen given that all the maximum and minimum timings are defined by only maximum and minimum boundaries, respectively, so the maximum timing (*Ta4_max*) will always be greater than the minimum (*Ta4_min*).

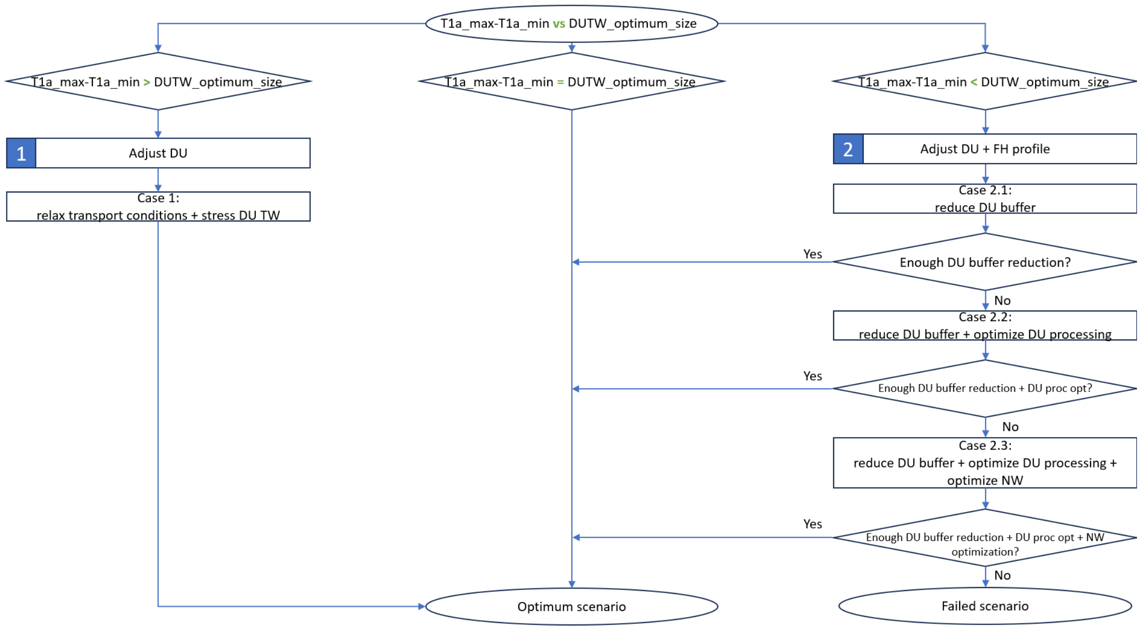


FIGURE 28. FLOWCHART OF THE DL ANALYSIS IN THE FH TIMING SIMULATOR

There may exist two major types of case studies that do not meet initially the timing relationships but could be adjusted to translate into several resolving methods, depending on the E2E segment needs to be adjusted. So, if the scenario is initially failed or not optimum, it can be solved through:

1) **Adjusting O-DU (if (*T1a_max – T1a_min*) > DUTW_optimum_size)**
   In this case, the O-DU transmission window is unnecessarily longer than the optimum O-DU TW, thus, reducing the centralization capabilities by limiting the maximum achievable distance. The idea here is to increase the FH delay so the end of the O-DU TW fits exactly with the optimum window size, so that the centralization distance could be increased as much as possible. Therefore, this case consists of adding some extra time for the transport network. This is:

$$T12\_max' > T12\_max.$$

   One way of increasing *T12_max*:

   - Relaxing transport network conditions but stressing O-DU: implies increasing only *T12_max* without affecting other timing.

1.1) Relaxing transport conditions, i.e., relaxing NW routers conditions; but stressing O-DU

In this case, in order to increase the FH maximum delay, the timing budget reserved initially for the variability of the network, such as impairments, jitter, queueing, etc., is increased. This sub-timing is only defined in FH maximum delay, not in the minimum so it will not affect then to the $T12\_min$ timing. Thus, there will be an extra time to accommodate longer delays allowing to relax transport network constraints. So, this is:

$$t'_{NW\_impairments} > t_{NW\_impairments}$$

This implies:

- *T12_max* increases → *T1a_min* increases
- No changes in *T12_min* → No changes in *T1a_max*

The result is shown in Equations *(7)* and *(8)*:

$$T12_{max}' = t_{prop} + t_{switching} + t'_{NW_{impairments}} \quad \textbf{(7)}$$

$$T1a\_min' = T12\_max' + T2a\_min \quad \textbf{(8)}$$

These changes are shown in Figure 29. *T1a_min* is shifted the quantity that it is given as extra time for the NW variable conditions and impairments that may exist, thus affecting only to *T12_max*. In this way, the network will have more flexibility to operate, since it tolerates longer delays that may occur due to different impairments or queues management in the routers.

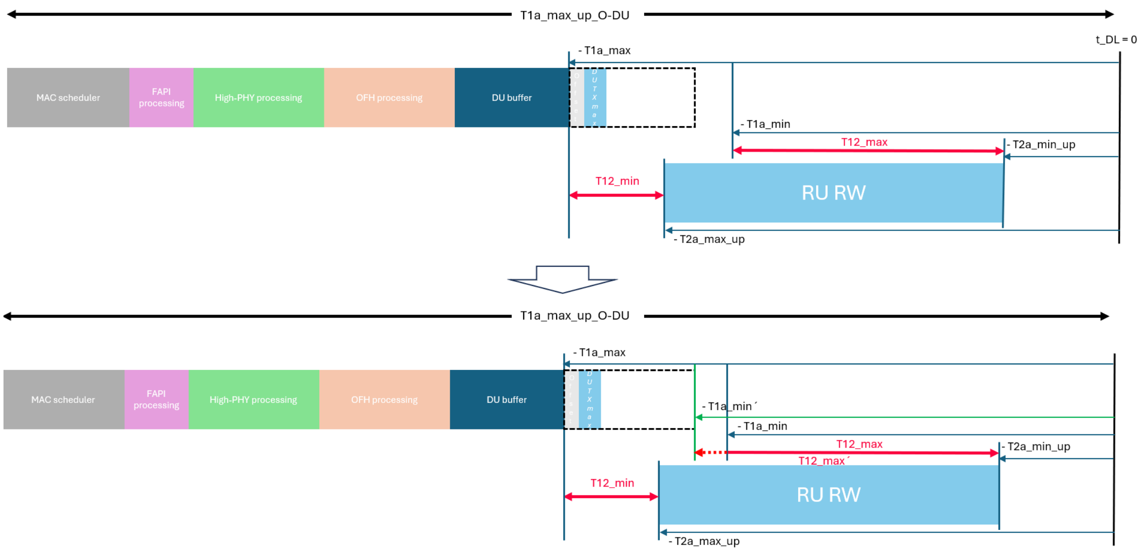


FIGURE 29. TIME MODIFICATIONS IN DL DIAGRAM AFTER RELAXING TRANSPORT CONDITIONS

2) **Adjusting O-DU + FH profile (if (*T1a_max* – *T1a_min*) < DUTW_optimum_size)**
The O-DU transmission window is shorter than the optimum one. This means that the O-DU has less opportunities to send symbols, what can affect to the system capacity. To solve this, both the DU and FH profile are adjusted.
There are three ways of solving this, ordered by priority:
- Reducing DU buffer.
- Reducing DU buffer + Optimizing DU processing.
- Reducing DU buffer + Optimizing DU processing + Optimizing NW.

2.1) Reducing O-DU buffer

The O-DU uses a buffer to accumulate the symbols ready to be sent through the FH. This buffer can be reduced if needed in order to move forward *T1a_max_up* and give more time for the O-DU TW to fit perfectly with the optimum size. This is:

{O-DU buffer}´ < {O-DU buffer}

This implies:

- *T2a_max* virtually increases by adding a buffer to O-RU RW.
- *T1a_max* increases. No changes in *T12_min* by adding a buffer to O-RU RW. This represents an extra time in which the O-RU can store packets before sending them to the antenna.

The result is exposed in Equation *(9)* and Equation *(10)*:

$$T2a_{max}{}' = T2a\_\max + Buffer \quad \textbf{(9)}$$

$$T1a_{max}{}' = T12\_\min + T2a_{max}{}' \quad \textbf{(10)}$$

The changes can be seen visually in Figure 30.

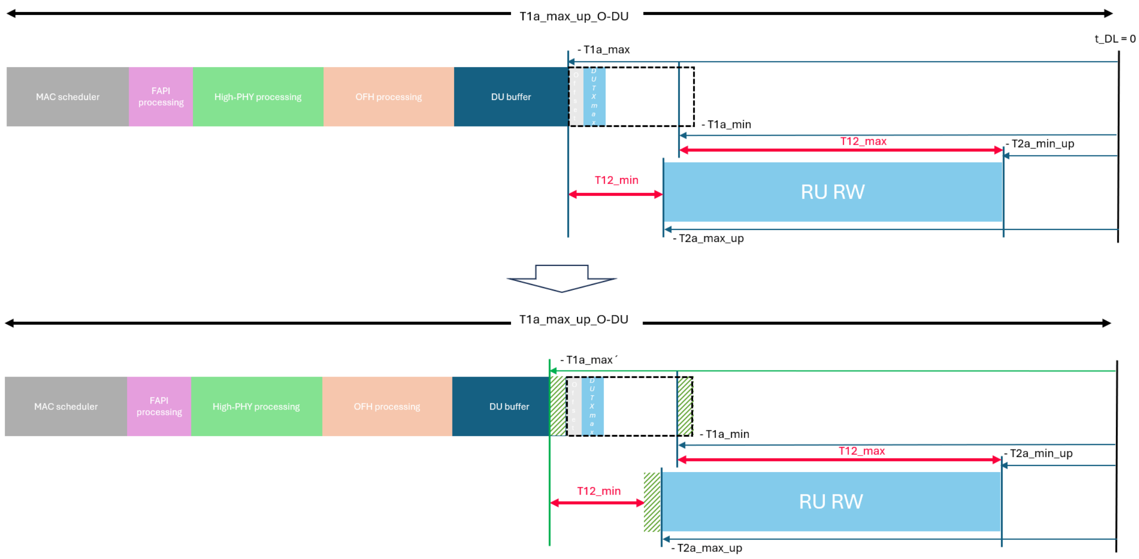


FIGURE 30. TIME MODIFICATIONS IN DL DIAGRAM AFTER REDUCING O-DU BUFFER

2.2) Reducing DU buffer + Optimizing DU processing

When the DU buffer is not enough to achieve the optimum scenario, it becomes necessary to optimize also some of the sub-timings of the DU processing chain, in order to give more time for *T1a_max*. This is:

{O-DU proc.}´ < {O-DU proc.}

This involves the same fronthaul timing changes as in the previous case, and the diagram is exposed in Figure 31.

Financiado por la Unión Europea NextGenerationEU
Plan de Recuperación, Transformación y Resiliencia
UNICO I+D
GOBIERNO DE ESPAÑA
cttc Centre Tecnològic de Telecomunicacions de Catalunya

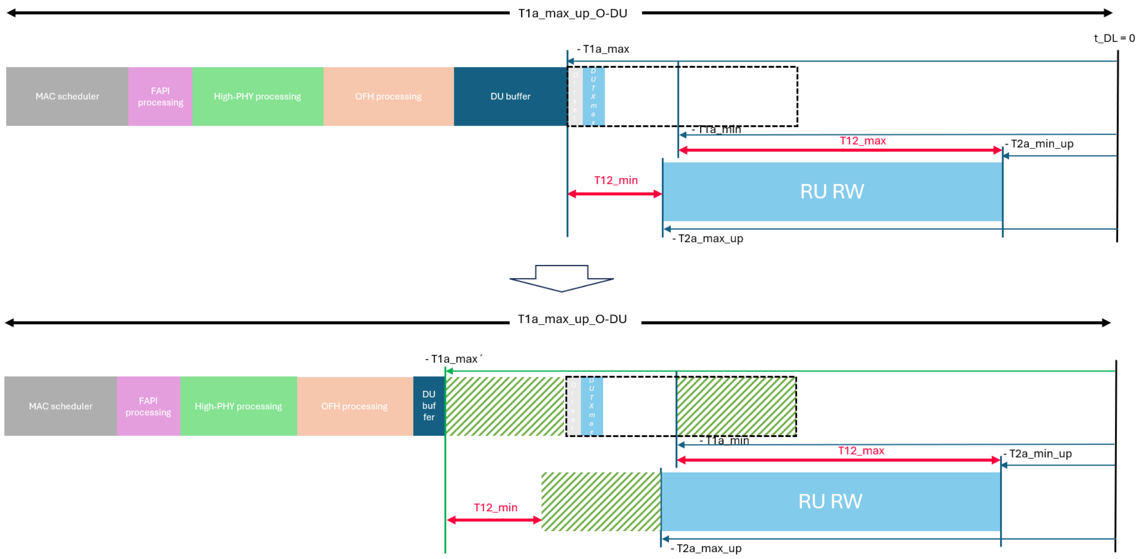


FIGURE 31. TIME MODIFICATIONS IN DL DIAGRAM AFTER REDUCING DU BUFFER AND OPTIMIZING DU PROCESSING

2.3) Reducing DU buffer + Optimizing DU processing + Optimizing NW

Furthermore, when the above is still not enough to achieve an optimal scenario, it is necessary to optimize the network as a last step. As described above in previous pages, FH maximum delay (*T12_max*) is the sum of the propagation time, the routers switching time and the network impairments. If this latter is reduced will lead to higher stress for the network elements, but without affecting the propagation distance, since *T12_min* remains unchanged. This case is based on:

{NW routers variable times}´ < {NW routers variable times}

This implies:

- *T12_max* decreases → *T1a_min* decreases
- No changes in *T12_min* → No changes in *T1a_max*

The result is exposed in Equation *(11)* and Equation *(12)*:

$$T12'_{max} = t_{prop} + t_{switching} + t'_{NW\_impairments} \quad \textbf{(11)}$$

$$T1a\_min' = T12\_max' + T2a\_\text{min} \quad \textbf{(12)}$$

It is important to take into account that *T12_max* can be reduced ideally as maximum to be equal to *T12_min* at the cost of optimizing the NW routers impairments, as could be seen in Figure 21.

The timing changes in U-Plane DL are shown in Figure 32.

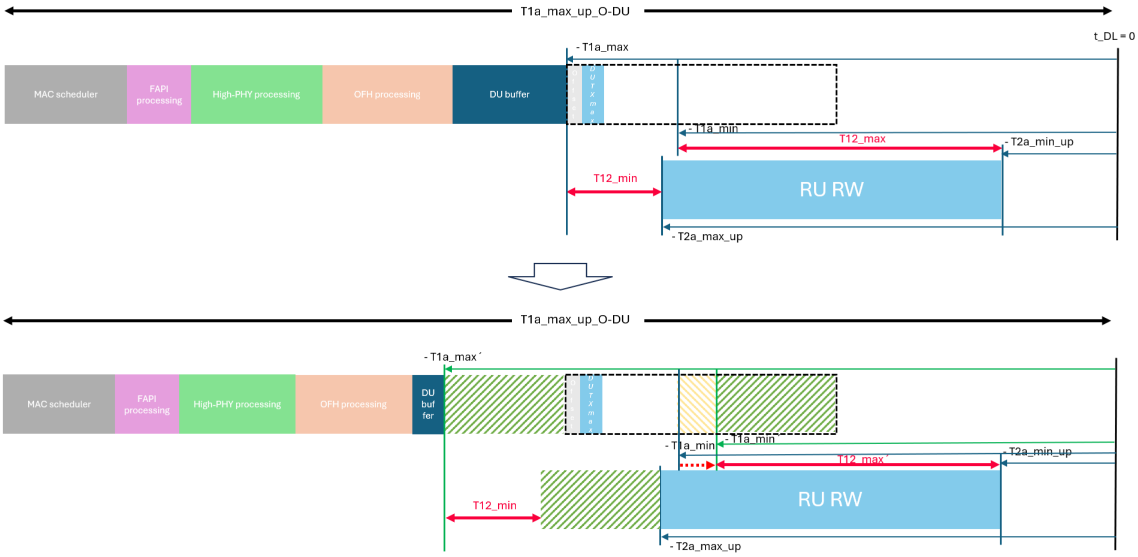


FIGURE 32. TIME MODIFICATIONS IN DL DIAGRAM AFTER REDUCING DU BUFFER, OPTIMIZING DU PROCESSING AND OPTIMIZING NW

Finally, if those changes are not enough to achieve optimum scenario, the output is considered failed and that centralization cannot be obtained.

2.5.3 Results

After executing the corresponding scenario, the developed software outputs the following results:

- In MATLAB command window, runtime comments are automatically generated detailing the configured scenario and the results obtained. These comments provide relevant information about the changes made in the simulation, along with the justification for each setting, allowing a better understanding of the impact of these on the model. The output in the command window also includes the centralization distance achieved and the maximum and minimum values of the FH delay.
- Multiple figures, with the architecture topology, the DL and UL diagrams of the initial scenario, the DL and UL diagrams of the optimal scenario (if achievable) and the DL and UL diagrams with the temporal changes made to see easily the differences.
- An Excel file, which exports several parameters with their values before and after the corrections applied in DL analysis (if needed): the case of study, the centralization distance, the start of the O-DU TW, the end of the O-DU TW, FH maximum and minimum delays, and the O-DU processing span, among others.

Financiado por la Unión Europea NextGenerationEU
Plan de Recuperación, Transformación y Resiliencia
UNICO I+D
GOBIERNO DE ESPAÑA
cttc Centre Tecnològic de Telecomunicacions de Catalunya

Thus, results were extracted for each of the test cases shown in Table 6. Since centralization distances and network conditions in each simulation are not exactly the same because of impairments or different centering distances, there is some variability in the results between simulations. Therefore, 10 simulations have been run for each test case to see the ranges of results.

In general terms, all the extracted scenarios have been improved using the above mentioned case study 1 (see Figure 27), i.e., the scenario allowed relaxing the transport conditions in order to have a higher centralization.

As commented, MATLAB developed tool allows user to select the scenario and displays in command window the changes made and the final results. An example is shown in Figure 33.

```
Command Window
************************************************************
***************OPEN RAN FH TIMING SIMULATOR***************
************************************************************
Number of simulations: 1
DU located at HL3 (Option 1) or HL4 (Option 2)? (1/2): 1
Distance case: shortest (Option 1), average (Option 2), longest (Option 3): 3

***************Feasibility analysis of centralization scenario by time delay***************
Analysis ----------------------------------------
DL:
DU TW is bigger than needed. It is possible to adjust the DU TW
Time to correct: 40.028 us
Case 1
T12_max has been extended, then relaxing the NW routers conditions but stressing the DU
Previous T12_max = 345.972 us -> New T12_max = 386 us
***************Feasibility analysis of centralization scenario by time delay***************
Analysis ----------------------------------------
DL:
DU TW is bigger than needed. It is possible to adjust the DU TW
Time to correct: 40.028 us
Case 1
T12_max has been extended, then relaxing the NW routers conditions but stressing the DU
Previous T12_max = 345.972 us -> New T12_max = 386 us

UL:
  OK
Final results ----------------------------------------
  Centralization distance: 59.4 km

  - DL:
    FH max delay ≤ 386 us
    FH min delay ≥ 345 us

  - UL:
    FH max delay ≤ 386 us
    FH min delay ≥ 345 us

*******************************************************
*****************Successful scenario*****************
*******************************************************
Exporting results ----------------------------------------
```

FIGURE 33. COMMAND WINDOW COMMENTS

Regarding the visual outputs, the developed tool extracts several diagrams. One of them is to show the initial result, taking into account the O-RU timing and the introduced network conditions. An example of these diagrams both in DL and UL can be seen in Figure 34 (green squares represent control plane, blue ones represent user plane).

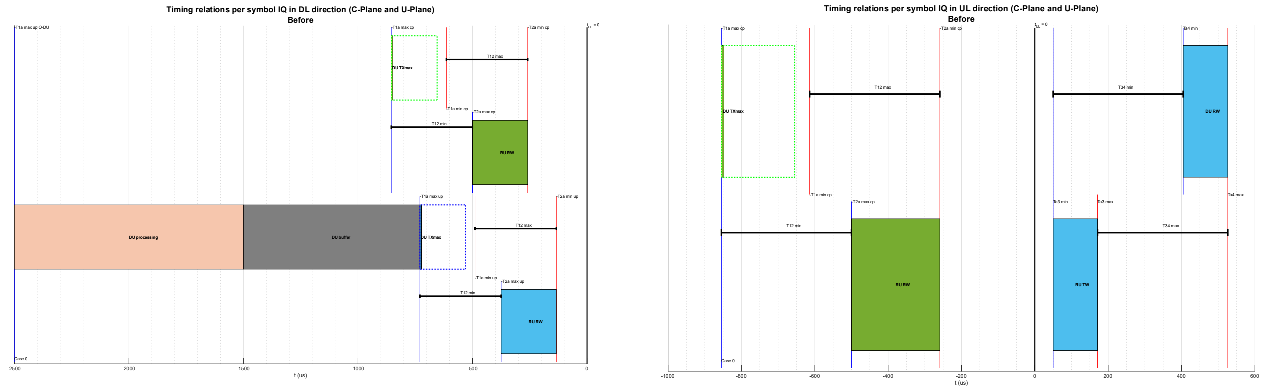


FIGURE 34. INITIAL VISUAL RESULTS IN DL (LEFT) AND UL (RIGHT)

As previously mentioned, the program, if possible, adjusts that scenario to improve centralization. Consequently, it displays the visual results once the modifications have been applied. This can be seen in Figure 35.

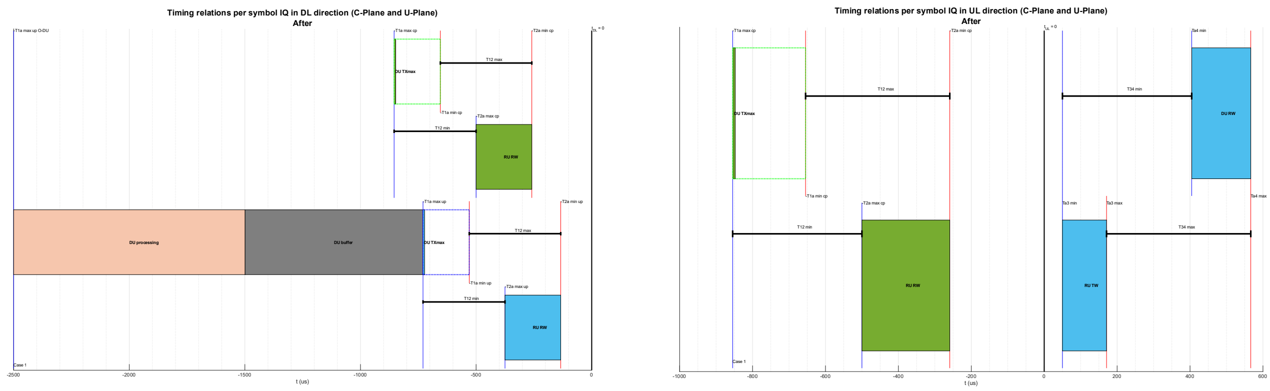


FIGURE 35. VISUAL RESULTS IN DL (LEFT) AND UL (RIGHT) AFTER APPLYING CORRECTIONS

Additionally, to better visualize the applied changes, the software overlays the initial and final scenarios, clearly highlighting the timing that has been modified. This is shown in Figure 36. It can be seen that the maximum FH delay (*T12_max*) has been extended to fit perfectly with the O-DU TW, and consequently in the UL the O-RU RW has also been increased.

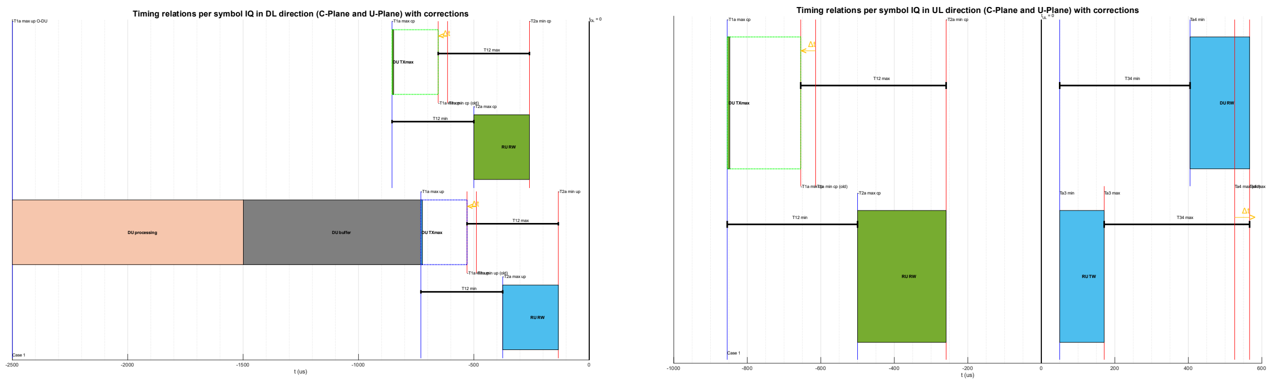


FIGURE 36. OVERLAID VISUAL RESULTS IN DL (LEFT) AND UL (RIGHT) OF THE INITIAL AND FINAL

2.5.4 Conclusions

As observed across all scenarios, the transport conditions (case study 1) were relaxed in order to explore the feasibility of further centralization. Specifically, the maximum FH delay was extended to align precisely with the end of the optimal O-DU TW.

Six scenarios were validated using an E2E testbed to ensure proper system functionality under realistic network conditions, which are shown in the development of UC2PoC1 activities.

The results clearly show that when the fronthaul delay exceeds the limits specified by the O-RAN Alliance, the transmission and reception windows between the O-DU and O-RU become misaligned. This progressive misalignment leads to a gradual degradation in system performance. However, when the time windows are properly adjusted, system performance remains stable and consistent. From this, it can be concluded that further centralization is technically feasible beyond the current state-of-the-art fronthaul specifications. The experimental results, obtained from a real operator network, demonstrate that distances up to 65 kilometres can be supported without compromising performance, provided that careful timing alignment is maintained.

Financiado por la Unión Europea NextGenerationEU
Plan de Recuperación, Transformación y Resiliencia
UNICO I+D
GOBIERNO DE ESPAÑA
cttc Centre Tecnològic de Telecomunicacions de Catalunya

2.6 SMART-K2.3: Flexible functional split

2.6.1 System Design

The 6G wireless network is currently in the early stages of defining its targeted use cases and capabilities. Early research suggests that 6G will enable advancements in several use-cases and applications, including ultra-high data rate transmissions for holographic images, ultra-low latency communications for autonomous driving and industrial production, and ultra-dense connectivity for crowded environments like shopping malls. To support these emerging applications, 6G will require innovative technologies, such as Flexible Functional Splits (FFS). As the 6G standard evolves from 5G, it is essential to establish novel transport network requirements tailored to the unique demands of 6G, which will facilitate the operation of FFS and fully exploit the potential of the new network.

Within this key concept, 6G BLUR project makes two contributions. First, we calculate the fronthaul requirements that would be needed to support a 6G network. Second, we evaluate various functional splits with the ns-3 5G-LENA simulator as a first step towards exploring the potential of FFS. Specifically, we extend the current implementation in the 5G-LENA simulator, which already considers capacity constraints for the 7.2x functional split. Our contribution adds new constraints by introducing the 6, 7.1 and 7.3 functional splits with specific capacity requirements, which will limit packet transmission through the FH, as well as we include functional split option 7.2 without modulation compression technique. This work is a first step toward developing algorithms in the ns-3 5G-LENA simulator that can dynamically select the most suitable functional split based on the traffic demand, network load and scenario conditions.

To preview the transport capacity requirements for 6G, we define four 6G RAN scenarios: basic and advanced scenarios, both with and without multi-band (or carrier aggregation) capabilities. In the basic scenario, we use the values from 5G massive MIMO use cases. For the 6G advanced scenario, we extrapolate the values by applying scaling factors derived from previous generations (3G, 4G, 5G). Additionally, we examine both single-band and multi-band configurations with multiple Components Carriers (CCs) for each scenario, as adopted in 3GPP, which significantly increases the aggregated bandwidth available on the air interface, thus raising the required transport capacity. Table 11 presents the considered reference values for 4G, 5G, 6G basic and advanced single-band scenarios, taking into account downlink (DL) configurations.

Financiado por la Unión Europea NextGenerationEU
Plan de Recuperación, Transformación y Resiliencia
UNICO I+D
GOBIERNO DE ESPAÑA
cttc Centre Tecnològic de Telecomunicacions de Catalunya

TABLE 11. ASSUMED 6G NETWORK CONFIGURATION REFERENCE VALUES FOR DL SCENARIOS.

	Reference 4G	Reference 5G	6G Basic	6G Advanced
Peak Rate (PR)	150 Mbps	-	-	-
Channel Bandwidth (BW)	20 MHz	100 MHz	400 MHz	1 GHz
Modulation (M)	64QAM (M=6)	256QAM (M=8)	256QAM (M=8)	1024QAM (M=10)
Number of MIMO streams (N_S)	2	8	8	16
Number of antenna ports (N_{AP})	8	32	64	256
Sample Rate (SR)	-	30.72 Msamples/s	4*30.72 Msamples/s	10*30.72 Msamples/s

The different functional split options in 5G and 6G networks impact the required transport network capacity. In Split 1, only RRC signaling and user plane data are transmitted, with the transport capacity mainly determined by the maximum user data peak rate (PR). For 4G, the PR is 150 Mbps for DL, and for 6G, this PR is scaled based on bandwidth, modulation, and MIMO streams. Split 2 introduces extra PDCP signaling, increasing the required capacity.

Split 6 centralizes not only the RRC, PDCP, and RLC layers but also the MAC layer, requiring the transport network to handle control and scheduling signaling rate (CR). This signaling is crucial for radio resource allocation and HARQ processes. The Split 7.x options divide the PHY layer. Split 7.3 moves the DL channel coding to the DU. Split 7.2 further increases capacity requirements due to the centralization of modulation and spatial streams mapping in the DU. Meanwhile, Split 7.1 centralizes antenna ports, scaling transport capacity according to the number of antenna ports.

Finally, Split 8 requires the most restrictive fronthaul capacity, as the transport network must transmit IQ samples to the DU after the RF signal is processed and the analog-to-digital conversion is performed in the RU. This increases the capacity needed for the fronthaul and midhaul networks, especially in multi-band scenarios where the bandwidth and sampling rate are multiplied by the number of component carriers. Table 12 shows the equations used to calculate the capacity requirements for 6G Basic and Advanced scenarios, along with the corresponding capacity values.

Financiado por la Unión Europea NextGenerationEU
Plan de Recuperación, Transformación y Resiliencia
UNICO I+D
GOBIERNO DE ESPAÑA
cttc Centre Tecnològic de Telecomunicacions de Catalunya

TABLE 12. EQUATIONS TO CALCULATE THE TRANSPORT CAPACITY FOR DIFFERENT FUNCTIONAL SPLITS AND CORRESPONDING CAPACITY VALUES FOR 6G NETWORK SCENARIOS FOR DL.

Split option	Capacity	6G Basic	6G Advanced
1	$PR \cdot \frac{BW}{BW_{ref}} \cdot \frac{N_S}{N_{S_{ref}}} \cdot \frac{M}{M_{ref}}$	16 Gbps	100 Gbps
2	$PR \cdot \frac{BW}{BW_{ref}} \cdot \frac{N_S}{N_{S_{ref}}} \cdot \frac{M}{M_{ref}} + s_{PDCP}$	16.01 Gbps	100.02 Gbps
4	$PR \cdot \frac{BW}{BW_{ref}} \cdot \frac{N_S}{N_{S_{ref}}} \cdot \frac{M}{M_{ref}}$	16 Gbps	100 Gbps
6	$(PR + CR) \cdot \frac{BW}{BW_{ref}} \cdot \frac{N_S}{N_{S_{ref}}} \cdot \frac{M}{M_{ref}}$	16.53 Gbps	103.33 Gbps
7.3	$(PR + CR) \cdot \frac{BW}{BW_{ref}} \cdot \frac{N_S}{N_{S_{ref}}} \cdot \frac{M}{M_{ref}} \cdot \frac{1}{R} + MAC_{info}$	50.4 Gbps	310.80 Gbps
7.2	$N_{SC} \cdot N_{sym} \cdot W \cdot N_s \cdot 1000 + MAC_{info}$	86.71 Gbps	430.78 Gbps
7.1	$N_{SC} \cdot N_{sym} \cdot W \cdot N_{AP} \cdot 1000 + MAC_{info}$	688.25 Gbps	6.88 Tbps
8	$SR \cdot W \cdot N_{AP} \cdot 5$	1.25 Tbps	12.58 Tbps

The transport latency requirement (T_{trans}), which is equal to (T_{FH}) for fronthaul and (T_{MH}) for midhaul, is determined by the interactions between protocol layers in the CU, DU, or RU. The latency depends on the functional split and is calculated based on the most restrictive processes that require signal exchange between different units. In 5G, latency is divided into three groups: Split 1 and Split 2, where only high protocol layers are centralized and latency is less restrictive; Split 4, which divides the RLC and MAC layers; and low-layer splits (6, 7.x, and 8), where more complex processes centralize at the MAC layer, leading to higher latency.

For Split 4, the latency is constrained by the interaction between the RLC and MAC layers, particularly due to the process of scheduling radio resources for each packet. Within 1ms Transmission Time Interval (TTI), the RLC sends information about the stored amount of data to the MAC layer in order to schedule resources. The MAC scheduling process takes approximately 500 µs. After scheduling, the MAC layer informs the RLC layer about the transmission opportunity (TxOpp), this information is sent through the transport network. Then, RLC creates Packet Data Units, involving a segmentation or concatenation process that takes around 200 µs. Since these signals are transmitted via the fronthaul and must be completed within 1ms, the transport network must support a maximum delay of 100 µs.

For splits 6, 7.x, and 8 (see Figure 37), the MAC layer is centralized, and the most latency-restrictive process is the HARQ, which in 4G is fixed at 4ms. The parameter *k1* represents the time required from the moment the packet is transmitted until it is processed in the UE and received by the gNB. Assuming *k1* is equal to 4ms (as in 4G), negligible propagation delay, and a 500 µs decoding and ACK/NACK preparation time, the transport latency must be equal to 250 µs as shown in Figure 37. This latency can vary depending on the numerology or subcarrier spacing (SCS) used in the system. In 5G, multiple TTI durations are supported, such as 1 ms, 500 µs, 250 µs, and 125 µs. For 6G, these durations will likely continue, therefore, the transport latency would be equal to 25 µs for Split 4 at SCS = 60 kHz and 62.5 µs for Split 6 and beyond.

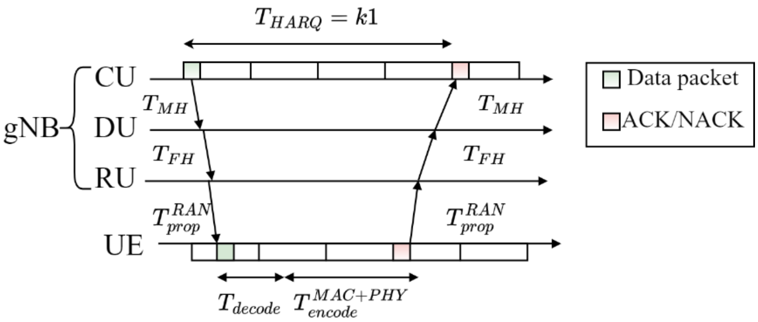


FIGURE 37. TRANSPORT NETWORK LATENCY REQUIREMENT CONSIDERING SPLIT 6, 7.X OR 8 OPTIONS

Furthermore, 6G will consider new technologies such as repeaters or relays. These technologies introduce additional latency due to multi-hop signal transmission and processing. As a result, the 6, 7.x, and 8 split options will experience an increase in latency due to the longer HARQ process time.

2.6.2 Implementation

The 5G-LENA simulator can simulate a FH between the DU and RU considering different functional splits. Before transmitting data, the expected FH throughput is measured to verify that it can be supported by the FH link capacity C_{FH}. This is done by comparing the total data to be transmitted with the FH's maximum capacity. As shown in Figure 3 of Section 2.2, in 5G-LENA, the transmitted data over the FH is controlled by a class called *NrFhControl* (more details about this class can be found in section 2.2.2 of this document). This class includes the *GetFhThr* algorithm, which calculates the throughput to be transmitted depending on the selected functional split.

Financiado por la Unión Europea NextGenerationEU
Plan de Recuperación, Transformación y Resiliencia
UNICO I+D
GOBIERNO DE ESPAÑA
cttc Centre Tecnològic de Telecomunicacions de Catalunya

Initially, the 5G-LENA simulator implements only the split option 7.2 with modulation compression technique. In contrast, in this work we have extended the *NrFhControl*, specifically the *GetFhThr* function, in order to consider functional split option 6, 7.1 and 7.3, as well as 7.2 without modulation compression technique. This extension is based on the equations presented in the Table 12. To incorporate the capacity requirements for each split we have modified *GetFhThr* algorithm. To do this, we defined an attribute named *FunctionalSplit*. This attribute can be configured in the 5G-LENA example to specify the split option to be used in the simulation.

When the simulation starts, the *GetFhThr* function is called in each slot to calculate the amount of data that will be transmitted through the fronthaul. *GetFhThr* is used to calculate the FH throughput for each UE, and these values are aggregated per bandwidth part (BWP) to obtain the total throughput. However, as we have seen in the equations of Table 12, what is transmitted through the fronthaul is not only the data from the UEs but also control signals that are necessary to process that data at the RU. These control signals are represented by a single numerical value for the total throughput in the theoretical equations. However, in the simulator, we will incrementally add a smaller value for each UE.

Algorithm 1: Calculate the Fronthaul Throughput

Input: Scheduled UEs, Functional Split
Output: thr_{bwp}
For from i = 1 to the total number of scheduled UEs
  thr_{UE_i} = Calculate the throughput generated to transmit the UE_i data and control information using *GetFhThr*
  For search among every BWP
    if map(thr_{UE_i}, *BWP*) = true
      $thr_{bwp} = thr_{bwp} + thr_{UE_i}$
    else if every BWP has been evaluated
      $thr_{bwp} = thr_{UE_i}$
    else
      go to next BWP
    end
  end
end

For split options 6 the total capacity is calculated as: $(PR + CR) \cdot \frac{BW}{BW_{ref}} \cdot \frac{N_S}{N_{S_{ref}}} \cdot \frac{M}{M_{ref}}$. When calculating the throughput transmitted through the fronthaul for each UE in the *GetFhThr* function, we need to determine the actual BW used by the UE for data transmission. In case of using OFDMA scheme to access radio resources, the BW used by this UE is given by:

Financiado por la Unión Europea NextGenerationEU
Plan de Recuperación, Transformación y Resiliencia
UNICO I+D
GOBIERNO DE ESPAÑA
cttc Centre Tecnològic de Telecomunicacions de Catalunya

$$bw_{UE_i} = 12 \cdot \left\lceil \frac{n_{REG}}{n_{sym}} \right\rceil \cdot 2^{\mu} \cdot 15 \cdot 10^3 \quad \textbf{(13)}$$

where n_{REG} is the total number of Resource Element Groups (REG) assigned to a UE across n_{sym} symbols. To determine the number of Resource Blocks (RBs) assigned, we divide n_{REG} by n_{sym}. Each RB contains 12 subcarriers, and the subcarrier spacing varies depending on the numerology (μ) used. In 5G, there are 7 different numerologies from 0 to 6. The subcarrier spacing is calculated as $2^{\mu} \cdot 15 \cdot 10^3$, this results in subcarrier spacing ranging from 15 to 960kHz.

Therefore, to determine bw_{UE_i}, we need to consider the number of assigned RBs to the UE and the subcarrier spacing. Additionally, instead of using the maximum possible values for N_s and M, we should use the actual number of streams and modulation being employed for that specific user UE_i. Therefore, the reference values $(BW_{ref}, N_{S_{ref}}, M_{ref})$ as well as PR and CR are fixed, but the other parameters $(bw_{UE_i}, N_{S_i}, M_i)$ can vary on a per-slot and per-user basis.

2.6.3 Results

2.6.3.1 Scenario parameters

In order to evaluate the different functional split options, we have simulated with 5G-LENA a 5G network deployed in an indoor scenario. The network uses a 400MHz of BW, with one gNB configured with a transmission power of 30dBm. The 5G frames are configured with 120 kHz of sub-carrier spacing and they follow a TDD pattern. The network uses an adaptive modulation and coding scheme (MCS) following MCS Table 2 in TS 38.214 which supports up to 256QAM.

A total of 14 UEs are randomly distributed within the scenario, seven UEs running the Virtual Reality (VR) application and the other seven the Cloud Gaming (CG) application. The gNB transmits VR and CG packets in DL with probabilistic sizes that follow a truncated Gaussian distribution. The 5QI for VR is 89, DGBR_VISUAL_CONTENT_89 (high-priority), while for CG equals 3, GBR_GAMING (medium-priority). The average data bit rate offered for VR and CG is 45Mbps and 30Mbps, respectively.

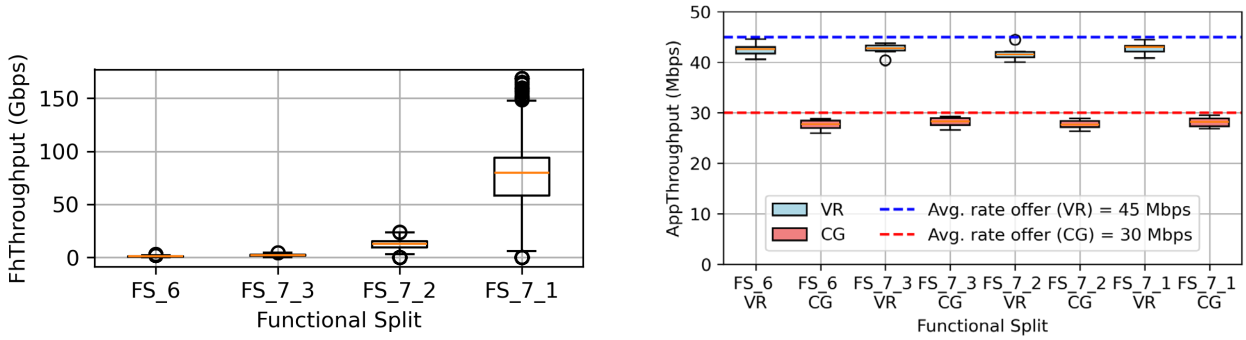
We adopt uniform planar arrays. For the antenna configuration, we have assumed 4 rows and 8 columns of dual-polarized antenna elements for the gNB, organized into 4 horizontal antenna ports and 2 vertical antenna ports. For the UE, 2 rows and 4 columns of dual-polarized antenna elements have been considered, spread in 2 horizontal antenna ports and 1 vertical antenna port. A dual-polarized configuration has been assumed. This leads to a total of 16 and 4 antenna ports at gNB and UE, respectively. We use the Proportional Fair (PF) MAC scheduler along with OFDMA access mode in order to allocate radio resources to transmit the packets.

Financiado por la Unión Europea NextGenerationEU
Plan de Recuperación, Transformación y Resiliencia
UNICO I+D
GOBIERNO DE ESPAÑA
cttc Centre Tecnològic de Telecomunicacions de Catalunya

The 5G network simulates a virtualized RAN architecture, implementing different functional split options 6, 7.3, 7.2 and 7.3 between the RU and DU in each simulation. We have evaluated the same scenario with varying capacities in the FH, from no limiting to highly limiting capacity: 1Tbps, 100Gbps and 1Gbps. In order to control the throughput transmitted through the FH, the Postponing FH control method (see section 2.2) is configured, which postpones the scheduled data that does not fit within the FH and tries to schedule it again in the next slots.

2.6.3.2 Evaluation results

To evaluate the different FS options, we first compared the throughput transmitted over the FH when capacity and bandwidth are not limiting factors, choosing a FH capacity equal to 1Tbps and a BW of 400MHz. Figure 38.A) represents the comparison of different FS versus the experienced FH throughput. The lower and upper edges of the box correspond to the 25th and 75th percentiles, respectively. Besides, the whiskers indicate 1.5 times the Inter Quartile Range, which represent approximately the 2 and 98 percentiles. Outliers are represented by circles corresponding to values that exceed the mentioned percentiles. We can observe that as the protocol stack is more centralized (e.g. FS option 7.1) the transmitted FH throughput increases. As can be seen, in none of these cases the transmitted throughput reaches 1Tbps, meaning that the capacity of the FH is not limiting the data transmission. This is what is being represented in Figure 38.B), where both applications receive the generated average bit rate.



A) B)

FIGURE 38. DIFFERENT FUNCTIONAL SPLIT OPTIONS COMPARISON OF A) FH THROUGHPUT AND B) APPLICATION THROUGHPUT.

After evaluating the different FS options, the next step was to analyse the scenario when the FH capacity becomes a limiting factor. Therefore, the following figures show the throughput obtained by the CG and VR applications when the capacity is set to 100Gbps and 1Gbps. As seen in the Figure 39.A), when using 100Gbps capacity and FS option 7.1, the users are unable to receive generated average data rate since the capacity limits the data transmitted over the FH. In contrast, the end-to-end performance is not limited by this capacity when using other FS options such as 6, 7.3 or 7.2. This becomes even more evident when the FH capacity is reduced to 1Gbps, where all FS options are affected, and even with FS 6, the throughput achieved for both applications is less than what is generated.

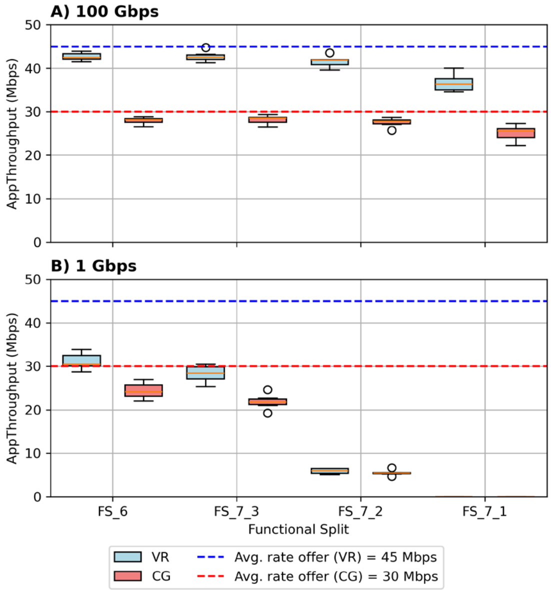


FIGURE 39. VR AND CG APPLICATIONS WHEN FH CAPACITY IS A) 100 GBPS AND B) 1GBPS.

Finally, we evaluated the impact of the modulation compression technique on FS option 7.2 (

Figure 40) when FH capacity is equal to 1Gbps. The results show that when modulation compression technique is considered, FS option 7.2 can achieve similar results to FS 7.3. However, when this technique is not used, the throughput achieved with split option 7.2 is below 10Gbps. With the split 7.1, the throughput obtained is equal to 0.

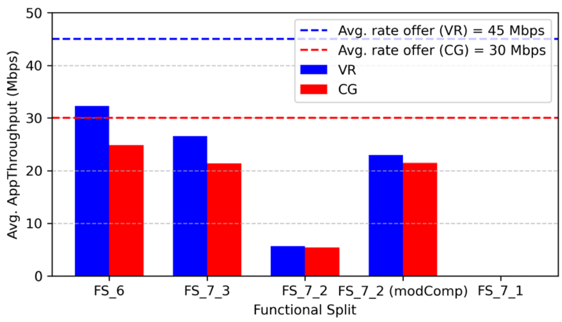


FIGURE 40. VR AND CG APPLICATIONS THROUGHPUT WITH DIFFERENT FS OPTIONS AND CONSIDERING MODULATION COMPRESSION TECHNIQUE FOR FS 7.2

2.6.4 Conclusions

As part of this key concept, we have extended the understanding of capacity and latency requirements that transport networks will need to meet in future 6G deployments [19]. In parallel, we have analyzed the impact of different functional split options using the 5G-LENA simulator, and this will be contributed to the research community. The work presented here represents the first step towards implementing dynamic Flexible Functional Splits in 5G-LENA, which is left for future work.

2.7 SMART-K2.4: Radio Stack optimization

2.7.1 System design

Nowadays, as outlined by the O-RAN Alliance, the maximum distance the fronthaul link can sustain is approximately 20 km (equivalent to approximately 100 µs in propagation time, one-way) [26], which correspond to the requirements of eCPRI protocol involved between O-DU and the O-RU and the type of service. The utilization of Split 7.2x involves splitting the physical layer, with the low-PHY residing in the O-RU and the high-PHY along with the MAC layer located in the O-DU. Implementing these layers separately and at a great distance between them results in a reduction of the time budget available for the O-RU and O-DU to process L1 and L2 layers that are also highly sensitive to timing because part of the time budget has to be spent in propagation time.

O-DU centralization overstresses some of the functions of RAN protocol stack, particularly those associated with the MAC layer. This basically introduces the need for apply a series of adjustments and considerations aimed at accommodating the longer distances derived from that level of centralization and optimizing processing techniques in the new architecture.

In the context of a centralized architecture, it becomes realistic to pursue the introduction of certain adjustments in the protocols and/or parameterization at the O-DU stack to counterbalance the increased delays within the FH deployed over non-dedicated networks. This is essential to ensure the seamless operation of all protocols and E2E timing (L2 + L1 + FH + O-RU) within a network topology where certain functions are placed far away from the central site.

This basically introduces a series of adjustments and considerations aimed at accommodating the longer distances derived from that level of centralization and optimizing processing techniques in

Financiado por la Unión Europea NextGenerationEU
Plan de Recuperación, Transformación y Resiliencia
UNICO I+D
GOBIERNO DE ESPAÑA
cttc Centre Tecnològic de Telecomunicacions de Catalunya

the new architecture. As a consequence, the protocols operating within the MAC layer will need to operate under longer latencies, which this arises a need to assess and optimize the performance of these protocols under these conditions to ensure efficient operation in these centralized scenarios as it occurs also in distributed topologies.

2.7.2 Implementation

One of the main changes to be considered is the adjustment of the timing parameterization in the O-DU processing and the fronthaul time windows for both C-Plane and U-Plane in both DL and UL directions. These timing determines the dedicated space to process one U-Plane slot in the E2E DL chain:

- MAC processing: performs various functions, such as data multiplexing/demultiplexing, HARQ retransmission management and scheduling, among others.
- FAPI: processing corresponding to the interface between L2 and L1. Negligible value (order of ns).
- High-PHY layers processing: it is the most processing-intensive and therefore gives the most room for optimization. It is responsible for functions such as modulation and channel coding.
- FH processing: it involves obtaining the IQ samples, compressing them (usually with the BFP-9 algorithm), and assembling the eCPRI packages.
- O-DU buffer: used to store the IQ symbols until the next transmission opportunity.
- FH time windows: time ranges when the O-DU and O-RU can transmit or receive over the FH interface.

On the O-DU side, the main adjustable parameters in SRS SW for adapting the processing times are listed in the Table 13.

TABLE 13. CONFIGURABLE O-DU PROCESSING PARAMETERS IN SRS SW

Parameter in SRS SW	Description	Default value
max_proc_delay	Known as *T1a_max_up_O-DU* in O-RAN Alliance specifications. Measured in slots, it is the time in advance that a downlink slot is processed throughout the processing chain with respect to when the slot is transmitted over the air.	5 slots

Financiado por la Unión Europea NextGenerationEU
Plan de Recuperación, Transformación y Resiliencia
UNICO I+D
Gobierno de España
cttc Centre Tecnològic de Telecomunicacions de Catalunya

dl_processing_time	FH processing time for one slot.	400 us
l1_dl_cpus	Number of cores assigned to L1 downlink tasks.	-
nof_dl_threads	Number of PHY threads to process downlink. Each thread process 2 FH ports.	-
enable_dl_parallelization	Flag for the parallelization of tasks in FH.	true

On the FH side, the configurable parameters are those related with the O-DU transmission and reception windows in downlink and uplink respectively, shown in Table 14.

TABLE 14. CONFIGURABLE O-DU FH PARAMETERS IN SRS SW

Parameter in SRS SW	Description
t1a_max_cp_dl	Sets T1a maximum value for downlink control-plane
t1a_min_cp_dl	Sets T1a minimum value for downlink control-plane
t1a_max_cp_ul	Sets T1a maximum value for uplink control-plane
t1a_min_cp_ul	Sets T1a minimum value for uplink control-plane
t1a_max_up	Sets T1a maximum value for user-plane
t1a_min_up	Sets T1a minimum value for user-plane
ta4_max	Sets the Ta4 maximum value for User-Plane
ta4_min	Sets the Ta4 maximum value for User-Plane

All the E2E timing (L2 to over the air (OTA)) in DL can be seen in Figure 41.

Financiado por la Unión Europea NextGenerationEU
Plan de Recuperación, Transformación y Resiliencia
UNICO I+D
GOBIERNO DE ESPAÑA
cttc Centre Tecnològic de Telecomunicacions de Catalunya

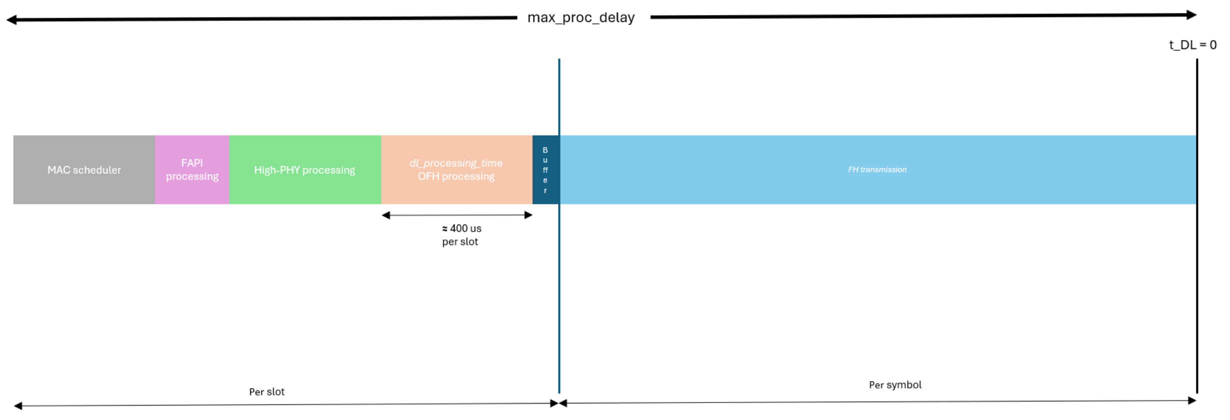


FIGURE 41. E2E DL TIMING BLOCKS

The timings of the FH windows are not arbitrary but depend directly on the specific characteristics of the network elements involved, in particular the O-RU as well as the capacity and characteristics of the aggregation network carrying the fronthaul traffic. These parameters were previously obtained by performing a comprehensive characterization of the fronthaul link, explained in detail in Section 2.5 (SMART-K2.2). Based on this characterization, reference values were obtained in order to adjust the O-DU timing to ensure that the network meets timing requirements in high demand scenarios or under changing conditions, thus maximizing efficiency and quality of service in the network deployment. In this way, the O-DU window will adapt to the best scenario, achieving a greater degree of centralization by bringing forward the start of its transmission window.

2.7.3 Results

In the context of this project, it has been analysed that the O-DU can be centralized to two different levels, following the nature of the Telefónica´s IP/Fusion network in Spain:

- HL4: it is an intermediate scenario, where the benefits are limited but is more feasible in the short-term.
- HL3: the desirable scenario, with larger distances and capacity aggregated (i.e. larger benefits) but more complex.

For each of the levels and based on the distance-distribution the routers have in the real network, the O-DU is deployed in three cases: a shortest one, an average one and a longest one.

The resulting scenarios from the FH characterization are presented in Table 15. They are extracted from the different locations the O-DU can be centralized (HL4/HL3) and their different distance ranges in order to have multiple cases (see Table 6).

Financiado por la Unión Europea NextGenerationEU
Plan de Recuperación, Transformación y Resiliencia
UNICO I+D
GOBIERNO DE ESPAÑA
cttc Centre Tecnològic de Telecomunicacions de Catalunya

TABLE 15. CENTRALIZATION TEST CASES

Test case	DU location	Centralization distance [km]	*T1a_max_up* [µs]	*T1a_min_up* [µs]	FH maximum delay [µs]	FH minimum delay [µs]	Real FH delay in tests
HL4-shortest	HL4	14,8	473,2	273,2	139,2	98,2	[min, mid, max]
HL4-average	HL4	19,8	498,2	298,2	164,2	123,2	[min, mid, max]
HL4-longest	HL4	22,12	509,8	309,8	175,8	134,8	[min, mid, max]
HL3-shortest	HL3	41,56	619	419	285	244	[min, mid, max]
HL3-average	HL3	47,78	650,1	450,1	316,1	275,1	[min, mid, max]
HL3-longest	HL3	61,96	721	521	387	346	[min, mid, max]

Just as a comment, in the above table are shown the DL timing, but of course they have relation with those of UL. In this sense, FH maximum and minimum delays are the same in both directions (*T12_max* = *T34_max*; *T12_min* = *T34_min*), and the UL O-DU window boundaries can be calculated easily with the equations exposed in Table 9, included in the description of the key concept SMART-2.2 above (Section 2.5).

Then, the time span obtained for the different windows of O-DU and O-RU in both directions are displayed in Table 16.

TABLE 16. DURATION OF DL AND UL TIME WINDOWS

		TW	RW
O-RU	DL		T2a_max_up – T2a_min_up = 375 -134 = 241 µs
	UL	Ta3_max – Ta3_min =171-50 = 121 µs	
O-DU	DL	T1a_max_up - T1a_min_up	
	UL		Ta4_max – Ta4_min = (171 + T34_max) – (50 + T34_min)

The FH delay values to be tested will initially consist of constant delay scenarios, where minimum, mid-range, and maximum values within the defined delay ranges will be evaluated. This phased approach ensures a comprehensive understanding of how static FH delays impact system performance under predictable and controlled conditions.

Subsequently, the testing will advance to include dynamic delay scenarios to test the conditions of a worst-case scenario, with the network experiencing impairments such as jitter existing in intermediate routers, traffic congestion, and varying latency caused by fluctuating network conditions.

Through this two-step testing methodology, insights are gained not only into the system's ability to meet these windows relationships but also its resilience and adaptability to more complex, time-varying delay patterns typical of operational networks.

As shown in Table 15, in the longest-case scenario, the maximum time advance required by the O-DU to initiate its transmission windows is 721 µs. This timing requirement indicates that in principle there is no immediate need to modify the foundational L1 (physical layer) and L2 (MAC layer) protocols to accommodate the longer delays introduced, as for example the HARQ protocol. Instead, the adjustment may be addressed by reconfiguring the parameterization of the O-DU time windows to align with the updated timing constraints. This approach simplifies implementation and minimizes the complexity of adapting existing protocol frameworks.

However, as a next step, the possibility of adapting and introducing changes within the O-DU stack remains as an open research question. While current findings suggest that reconfiguring timing parameters may suffice in the short term, a more in-depth exploration into modifying the O-DU's protocol stack (particularly at the MAC level) could potentially yield greater benefits in terms of system performance and centralization efficiency. Such adaptations might involve the redesign or optimization of scheduling algorithms, buffer management strategies, and HARQ timing configurations to better align with increased fronthaul delays and more centralized network architectures. Therefore, further investigation into how protocol-level enhancements within the O-DU can contribute to improved synchronization and reduced latency impacts.

In this case, to better understand the practical implications of the theoretical results, they have been subjected to in-depth analysis within the context of experimentation done within UC2PoC1 in the 6G-SANDBOX testbed. This analysis forms a key part of the planned activities for the project, providing an opportunity to validate the theoretical conclusions under controlled yet realistic testbed conditions and will allow to assess how the revised timing configurations impact overall system performance, including factors such as latency, synchronization, and throughput, ultimately allowing us to reach higher degrees of TRL (Technology Readiness Level). These tests also explore potential challenges or limitations that might arise when implementing these adjustments in practical network scenarios, offering valuable insights to refine the proposed solutions further. The experimental testbed consists of the following elements (further details can be found in the UC2PoC1 description):

Financiado por la Unión Europea NextGenerationEU
Plan de Recuperación, Transformación y Resiliencia
UNICO I+D
GOBIERNO DE ESPAÑA
cttc Centre Tecnològic de Telecomunicacions de Catalunya

- A traffic generator and an O-RU emulator, that allows to inject traffic and emulate one or several O-RU instances.
- A network emulator, installed on the FH, to be able to introduce in the link the long propagation times and impairments of the scenario.
- An O-DU and an O-CU on a COTS server.
- A PTP switch, in order to distribute the synchronization signal to the O-DU and O-RU emulator.
- A Core emulator, to emulate the other side of the network.
- An orchestration and automation platform.
- A PC for metrics collection and analysis.

The configuration of the air interface and FH parameters is presented in Table 17. Since this work focuses on the FH interface, we assume ideal conditions for the emulation of the air interface. Furthermore, the system's capacity is constrained by the limitations imposed by the HW equipment and the SW. Although these constraints reduce the overall scope of the analysis and prevent a thorough evaluation of the impact of aggregated traffic on the FH interface, the testbed is enough to verify the timing relationships and ultimately the FH centralization results. Eventually, the system could be scaled to be able to emulate conditions closer to the theoretical scenario in order to see the real impacts in a practical deployment.

TABLE 17. 6G-SANDBOX FH PARAMETERS

Parameter	Value
Radio access technology	5G NR TDD
TDD pattern	DDDDDDDSUU
Subcarrier spacing	30 kHz
Transmission bandwidth	100 MHz
Number of spatial/antenna streams	4
O-RU category	Category A
IQ compression	BFP9

Financiado por la Unión Europea NextGenerationEU
Plan de Recuperación, Transformación y Resiliencia
UNICO I+D
GOBIERNO DE ESPAÑA
cttc Centre Tecnològic de Telecomunicacions de Catalunya

2.7.4 Conclusions

The theoretical framework presented in this section provides valuable insights that support the validation and interpretation of the experimental results presented in Section 3.1.1.3.3 related with UC2PoC1. It establishes a solid analytical basis for understanding the timing behavior of the O-DU in relation to FH delays and transmission scheduling. According to the analysis, there is no fundamental need to modify the underlying software or protocol stacks. Instead, the observed timing requirements can be addressed through the adjustment of O-DU time window parameters. This suggests a practical and low-complexity approach to adapting the system, highlighting the efficiency of parameter reconfiguration over more invasive changes to the protocol implementation.

2.8 SMART-K2.5: QoS Management for time-critical immersive applications

2.8.1 System Design

Recent advancements in mobile networks have enabled immersive applications, such as XR and CG, to revolutionize fields like entertainment, education, healthcare, and manufacturing. These technologies are rapidly gaining a lot of momentum in both industry and consumer markets. However, XR and CG require low latency and high bit rates to deliver a seamless and immersive user experience, with most XR use cases classified as time-critical. Moreover, XR applications generate multiple traffic flows with different QoS requirements, as such, managing properly such types of traffic is quite challenging.

5G New Radio (NR) has presented a new framework for QoS provisioning, known as QoS model, that that is particularly suited for XR and CG applications, as it shifts the control to the flow level [13]. This framework enables the support and effective management of time-critical and multi-flow traffic. However, although the architectural design is defined by the standard, the scheduling algorithms are considered as implementation-specific functions. Therefore, defining and designing proper scheduling algorithms is crucial to optimize resource utilization and meet the stringent latency, reliability, and throughput requirements demanded by next-generation applications.

2.8.1.1 QoS scheduler

In this work we propose and implement a new generalized QoS MAC scheduler, presented in [14] and [1], capable of managing properly various traffic types, while guaranteeing the requirements of time-critical traffic. The proposed design, presented in the following, is based on the 5G QoS model. Let us notice that the 5G QoS model maps IP packets to QoS flows in the 5G core, and QoS flows to Data Radio Bearers (DRBs) in the 5G RAN, allowing for mapping of one or multiple Service Data Flows (SDFs) to one QoS flow and one or more QoS flows to one DRB. Moreover, differently from LTE, the 5G QoS model extends the support of non-Guaranteed Bit Rate (GBR) and GBR to include also delay-critical (DC) GBR. The differentiation among the various QoS flows is carried out based on a QoS flow identifier known as QFI that is included in an encapsulation header [15]. A QoS flow profile is used to define how the QoS flow should be treated and includes the 5G QoS Identifier (5QI), among other parameters. The 5QI includes the characteristics of the flow, such as the resource type (GBR/non-GBR/DC-GBR), the Priority Level, the Packet Delay Budget (PDB), the Packet Error Rate (PER), the Averaging Window and Maximum Data Burst Volume [15].

Based on the above, the proposed QoS MAC scheduler considers the 5G QoS profile characteristics and combines them with real-time MAC measurements to calculate scheduling weights based on which the user prioritization will be performed. In particular, it considers the 5QI information, such as the resource type, the priority level and the PDB, along with the head-of-line (HOL) delay and the Proportional Fair (PF) metric as provided at the MAC layer. The proposed calculation of the scheduling weight based on these metrics is shown in the following equation:

$$w = \begin{cases} (100-P)\dfrac{r^{\gamma}}{R_{(\tau)}} + F \quad for\ non-GBR\ and\ GBR \\ (100-P)\dfrac{r^{\gamma}}{R_{(\tau)}} D + F \quad for\ delay-critical\ GBR \end{cases}$$

where P is the default Priority Level of the QoS flow mapped to the DRB (lower P indicates higher priority for scheduling), r is the instantaneous achievable data rate calculated by the spectrum efficiency and the channel bandwidth, R(τ) is the past average data rate updated within the updated window size τ, F = 100 when the DRB has retransmission data and F = 10 otherwise, and γ is a configurable parameter. Moreover, we include the newly introduced delay budget factor D, that is the delay-aware weight related to the HOL packet delay and the PDB, and is calculated as:

$$D = \frac{PDB}{PDB - HOL}$$

Notice that when γ = 1, and assuming the same priority for all users and ignoring D, the scheduler corresponds to a typical PF scheduler. The past average data rate is calculated as:

$$R_{(\tau)} = (1-\alpha)R_{(\tau-1)} + \alpha A_{(\tau)}$$

In this equation, A(τ) is the current data rate over the updated window size τ computed as the ratio of all successfully delivered bits (including those bits still in retransmission) in the past updated window size, and α balances between the current data rate (A(τ)) and the past average data rate in the previous window (R(τ − 1)).

The active users are then classified in descending order in each Transmission Time Interval (TTI) based on the sum of the calculated scheduling weights for all their active flows:

$$W = \sum_{n=1}^{N} w$$

where N is the number of active logical channels for a given user. This classification results in scheduling first the users that have higher W.

Based on the above, it is worth highlighting that the proposed QoS MAC scheduler can meet the low latency requirements of delay-critical traffic, by combining the resource type, the Priority Level and the PDB of each flow, with the HOL delay and the PF metric. In this way, flows with tighter latency requirements are promoted and packet discards are avoided for these flows (since the weight increases as the HOL and consequently the D increase), while at the same time the Priority Level of each flow is considered to provide adequate QoS management, and includes the PF metric to achieve higher data rates and ensure user fairness.

In addition to the proposed QoS MAC scheduler, we present an algorithm applied in the downlink direction, known as QoS LC assignment, that allocates the assigned resources among the various flows or Logical Channels (LCs) of an active user, by considering the guaranteed bit rate (e_rabGuaranteedBitRate) and consequently the resource type of a flow. The logic behind the algorithm is to find initially all the active GBR and DC-GBR flows with its e_rabGuaranteedBitRate requirements set. In case there are more than one and their total requirements exceed the assigned bytes (Transport Block Size (TBS)), then the algorithm assigns equally the assigned bytes (as in a RR fashion) to these GBR/DC-GBR flows. In case their total requirements are less than the assigned bytes, the algorithm assigns to each flow the minimum among the e_rabGuaranteedBitRate and the RLC buffer size. The rest of the bytes, if any, are assigned in the rest of the LCs in RR fashion.

Although in this section we focus on the QoS MAC scheduler, it is worth highlighting that in the framework of the 6G BLUR project we have also developed XR loopback mechanisms [16], which adjust the XR video application payload as a function of the actual 5G RAN performance, using ns-3 5G-LENA simulator. Also, a holistic view on the multi-layer QoS management for XR traffic and end-to-end evaluation considering QoS MAC scheduling and XR loopback is presented in [17].

Financiado por la Unión Europea NextGenerationEU
Plan de Recuperación, Transformación y Resiliencia
UNICO I+D
GOBIERNO DE ESPAÑA
cttc Centre Tecnològic de Telecomunicacions de Catalunya

2.8.1.2 Lyapunov-based (DPP) scheduler

In [18], we present a MAC scheduler based on Lyapunov optimization. The scheduler is designed to pursue multiple objectives whose importance can be configured: traffic flow stability, guaranteed throughput (e_rabGuaranteedBitRate) and minimization of radio resources allocated by the base station. The scheduler design considers a number of traffic queues, each having a 5QI and potentially requiring a guaranteed bit rate, as is the case of GBR/DC-GBR flows. Simultaneously, the stabilization of the RLC queues is considered for all the UEs. On top of that, the scheduler seeks the minimization of allocated radio resources, according to a configuration parameter V. In order to tackle the resulting stochastic optimization problem, the Lyapunov theoretical framework is applied, which allows to define a throughput virtual queue that enforces e_rabGuaranteedBitRate as a time-average goal, evaluated over a time window whose duration is large enough, typically of 2 seconds according to the standardized 5QI values. In this context, a virtual queue pushes the scheduler to deliver sufficient data within each 2-second window to meet the e_rabGuaranteedBitRate. By applying the minimum drift-plus-penalty (DPP) algorithm, the scheduling problem is reduced to solving the following optimization at every TTI:

$$\min_{\alpha} \quad V\sum_{n}^{N}\alpha_n + \sum_{n}^{N}[Q_n(a_n - b_n)] \ + \ \sum_{n}^{N}[G_n(\Gamma_n - \rho_n)]$$

$$s.t. \ \sum_{n}^{N}\alpha_n \leq \Lambda$$

$$b_n \leq Q_n \ \forall n \ \in \{1, \ \dots, \ N\}$$

where Q refers to the RLC traffic queues, G to the virtual queues, a to the incoming traffic, b the outgoing traffic, Γ to the e_rabGuaranteedBitRate, and ρ to the current throughput. The first term of the objective function corresponds to the minimization of allocated resources, denoted by α. Then, the second and third term are related to the stabilization of the traffic queues, and the virtual queues, that address the throughput requirements. As it can be observed, the first term is multiplied by the design parameter V which permits giving more relevance to the objective function compared to the queues' backlog. In each slot the scheduler follows the following steps:

(1) Observe the states of $Q_n(t)$, $G_n(t)$ $\forall n \in N$.

(2) Take the decision that minimizes the objective function.

(3) Update queues.

$$Q_n(t+1) = \max[Q_n(t) - b_n(t), 0] + a_n(t)$$

$$G_n(t+1) = \max[G_n(t) + \Gamma_n - \rho_n(t), 0]$$

Additionally, we propose a refinement to the virtual queue update mechanism that leverages the concept of PDU set introduced in 3GPP Release 18 for the XR traffic. The definition of PDU sets for the XR traffic allows the grouping of multiple PDUs into a single logical unit, enabling more efficient handling of latency-sensitive and high-throughput flows. By incorporating the PDU set information provided by the XR applications into the virtual queue update process, the scheduler can better align resource allocation decisions with the unique requirements of XR services, such as tight latency bounds and bursty traffic patterns. This allows for finer-grained control over XR QoS metrics, while preserving RLC queue stability and reducing radio resource consumption. The virtual queues associated with flows carrying PDU set information are modified to incorporate XR-specific deadline constraints. Specifically, these virtual queues are updated per PDU set (τ) instead of per TTI (t), taking into account the PDU Set Size (PSSize) and the PDU Set Delay Budget (PSDB). In essence, the virtual queue exerts a control force that drives the scheduler to maintain a sufficient throughput to satisfy the latency requirements imposed by each PDU set. This approach ensures that delay-sensitive XR traffic is delivered within its target bounds, increasing the number of XR UEs satisfied (i.e. with more than 99% of PDU sets delivered on time) and, consequently, the XR capacity (defined as the maximum number of XR UEs for which at least 90% are satisfied) performed by the scheduler. The throughput target, Γ, is updated every time a PDU set arrives at time τ, as follows:

$$\Gamma_n(\tau) = \frac{PSSize_n(\tau)}{PSDB_n(\tau) - HOL_n(\tau)}$$

2.8.2 Implementation

QoS MAC Scheduler

The proposed QoS MAC scheduler has been implemented in the 5G-LENA simulator as depicted in Figure 42. Let us notice that we have followed the design approached used in 5G-LENA, where the core class of the scheduler is the *NrMacSchedulerNs3*, while the TDMA and ODFMA schedulers are child classes of the *NrMacSchedulerNs3* and implement the allocation of resources (symbols and RBGs) and the creation of the corresponding Downlink Control Indicators (DCIs). Moreover, the specialized schedulers, such as the Round Robin (RR) and PF, are implemented as child classes of the TDMA and ODFMA classes.

Financiado por la Unión Europea NextGenerationEU
Plan de Recuperación, Transformación y Resiliencia
UNICO I+D
GOBIERNO DE ESPAÑA
cttc Centre Tecnològic de Telecomunicacions de Catalunya

Based on the above, we have implemented the proposed scheduler as a new specialization in the above described framework and therefore we have created two new child classes known as *NrMacSchedulerTdmaQos* and *NrMacSchedulerOfdmaQos*, as shown in the UML Diagram in Figure 42 in blue colour. These classes are used to set the scheduler type and the access mode type through the user example. In addition, they implement the virtual functions that update the downlink and uplink metrics for each UE and the potential throughput based on the available resources. These metrics are then used by the newly implemented class *NrMacSchedulerUeInfoQos* that performs the sorting of the active users based on the calculation of the weights.

In order to have all the necessary QoS profile information in the NrMacSchedulerUeInfoQos, it has been necessary to extend the *NrMacSchedulerLC* class (denoted with grey colour in Figure 42) to include the 5QI of a flow, the resource type, the priority and the e_rabGuaranteedBitRate. Finally, although it is not depicted in Figure 42, we have extended the *BwpManagerAlgorithm* class to support Release 18 5QIs.

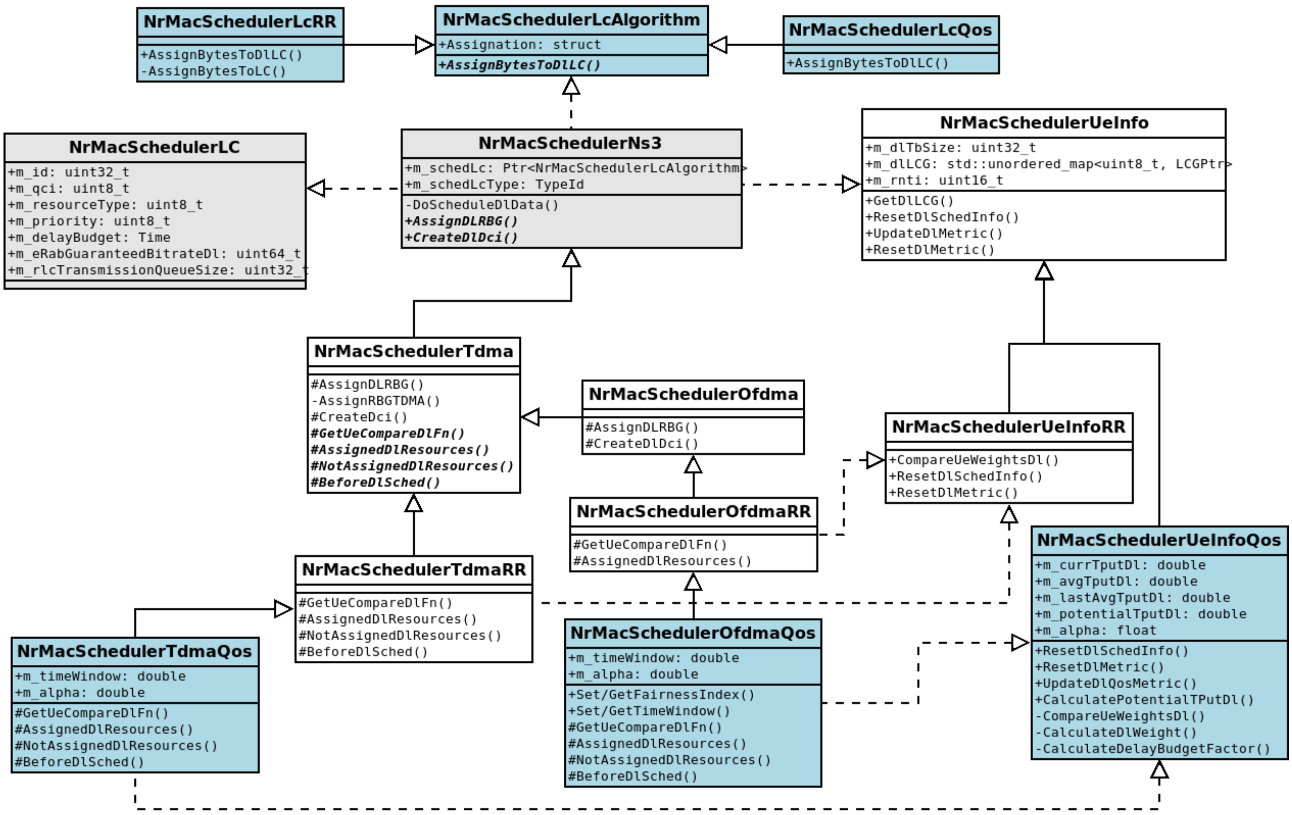


FIGURE 42. UML DIAGRAM OF THE PROPOSED QOS MAC SCHEDULER. BLUE: NEWLY IMPLEMENTED CLASSES, GREY: MODIFIED CLASSES, WHITE: ORIGINAL CLASSES.

QoS LC Assignment

5G-LENA in its original implementation was considering a distribution of bytes among the various LCs of a user based on a RR fashion. However, this approach may result inadequate since it does not take under consideration the load of each LC and its requirements. In addition, the implementation was in the form of a method in the *NrMacSchedulerNs3* class, limiting the flexibility to include alternative designs/algorithms. To address these limitations, we introduced the base class *NrMacSchedulerLcAlgorithm* and two child classes, the *NrMacSchedulerLcRR* and the *NrMacSchedulerLcQoS*, as illustrated in blue in Figure 42 in blue colour. The *NrMacSchedulerLcRR* class follows the same approach as the original implementation, assigning bytes to the LCs of the user in RR fashion, while the *NrMacSchedulerLcQoS* implements QoS LC assignment as presented above.

Moreover, in order to support the QoS provisioning for delay-critical and multi-flow traffic, we have extended the LTE module of ns-3 and the 5G-LENA simulator to include DC-GBR support, along with Release 18 5QIs and PDCP packet discarding based on the PDB of a packet. For more details the interested reader can refer to [1]. Finally, let us notice that we have implemented a test for non-GBR flows that validates the correct operation of the proposed QoS MAC scheduler and a user script that allows simulations to be performed.

Lyapunov-based MAC scheduler

Similarly to the QoS MAC scheduler, the Lyapunov-based scheduler has been integrated into the existing 5G-LENA framework as a new specialization, which required the creation of two new classes: NrMacSchedulerOfdmaDPP and NrMacSchedulerUeInfoDPP. The first class, NrMacSchedulerOfdmaDPP, is used to configure the scheduler via the user example and to implement the virtual functions responsible for updating UE-specific metrics, computing the potential throughput based on available resources, and calculating the weights that govern the scheduler's behaviour. This information is then leveraged by the NrMacSchedulerUeInfoDPP class, which sorts the active users according to the DPP strategy.

2.8.3 Results

For the evaluation of the proposed QoS MAC scheduler we have used the developed a user script, where a single-cell topology is considered, with two users. One of the users generates downlink non-GBR traffic (5QI=80), while the second user generates two flows in the downlink direction, one non-GBR (5QI=80) and one DC-GBR (5QI=87). For the DC-GBR flow we set the e_rabGuaranteedBitRate to a fixed value as shown in Table 18, along with rest of parameters.

Financiado por la Unión Europea NextGenerationEU
Plan de Recuperación, Transformación y Resiliencia
UNICO I+D
GOBIERNO DE ESPAÑA
cttc Centre Tecnològic de Telecomunicacions de Catalunya

TABLE 18. EVALUATION SCENARIO PARAMETERS

Scenario Parameter	Value
Carrier Frequency	4 GHz
Bandwidth	5/10/50 MHz
Numerology	1
BS/UE transmit power	43/23 dBm
BS/UE antenna height	10/1/5 m
Propagation model	3GPP UMa TR 38.901
BS antenna array	1 TXRU: 1x1 (3GPP elements)
UE antenna array	1 TXRU: 1x1 (isotropic element)
Shadowing/Fading	Disabled/Disabled
Channel condition	LoS, No updates
PDCP Discarding Timer	10ms for 5QI=80 (non-GBR), 5ms for 5QI=87 (DC-GBR)
e_rabGuaranteedBitRate (DC-GBR)	5 Mbps

We consider two network states by varying the configured bandwidth, non-saturation and saturation. This choice lies on the fact that although the performance of a scheduler is usually evaluated considering a non-saturated scenario, user prioritization cannot be studied in terms of throughput, since the available resources are sufficient to serve all the generated traffic. As such, our intention is to provide clear information related to the average end-to-end delay and throughput. Moreover, the operation and performance of a scheduler are more critical in a saturated network. For the case of the non-saturated state, we set the bandwidth to 50MHz, while for the saturated state, we study two different bandwidth configurations, 10MHz and 5MHz. We compare the proposed QoS MAC scheduler, in conjunction with the QoS LC assignment algorithm (denoted as QoS-QoS in the figures), against the traditional RR and PF schedulers and the same QoS MAC scheduler, but considering the already implemented RR LC assignment (denoted as QoS-RR in the figures).

Figure 43 presents the end-to-end average delay and throughput for the non-saturated state of 50 MHz. Notice, that in such state, the available resources are sufficient to serve all the user traffic, as also confirmed by the end-to-end average throughput. Moreover, from the figure it can be seen that all the schedulers present very similar performance. Focusing however on the end-to-end average delay, it can be clearly seen that the proposed QoS MAC scheduler used in conjunction with the QoS LC assignment, not only effectively reduces the delays of the DC-GBR flow and non-GBR of UE 2, but also it presents a reduction in the flow of UE 1, with respect to the rest of the solutions.

Figure 44 and Figure 45 depict the results for 10 MHz and 5 MHz, respectively. Setting the bandwidth in such values corresponds to a low saturation and high saturation conditions. Focusing on Figure X2, it can be observed how the bandwidth reduction impacts the performance of the schedulers. RR and PF achieve similar throughput for the 2 UEs of the network and they do not perform any distinction among the two flows of the same UE (the 2 flows of the same UE get half of the throughput irrespective of their QoS needs). The proposed QoS MAC scheduler, however, performs the allocation of resources considering their QoS requirements and provides therefore higher throughput for UE 2. In addition, comparing the QoS-QoS and the QoS-RR, it can be observed how the QoS LC assignment allows distinguishing among the various LCs of the same UE, something that was not possible with the original RR LC assignment implementation in 5G-LENA. On the other hand, the QoS MAC scheduler combined with the QoS LC assignment (QoS-QoS) reduces the delay of the DC-GBR flow and slightly increases its throughput, while it guarantees its e_rabGuaranteedBitRate requirements. Finally, Figure 45, corresponds to the most critical network condition, since the more we reduce the available bandwidth, and consequently the available resources, the more difficult is to meet user needs in terms of latency and throughput. As such, in such cases proper QoS management and flow treatment becomes crucial. From the figure it can be seen that the proposed QoS MAC scheduler is able to correctly prioritize the delay-critical traffic (DC-GBR) when compared to the RR and PF schedulers, increasing significantly the throughput and reducing the delay, at the cost of decreasing the throughput and increasing the delay of the non-GBR flow of UE 1, as expected. Additionally, it can be observed that the QoS-QoS outperforms the QoS-RR solution in terms of delay reduction and throughput increment for the DC-GBR flow, but also it can be seen that the QoS-QoS guarantees its QoS requirements.

Financiado por la Unión Europea NextGenerationEU
Plan de Recuperación, Transformación y Resiliencia
UNICO I+D
GOBIERNO DE ESPAÑA
cttc Centre Tecnològic de Telecomunicacions de Catalunya

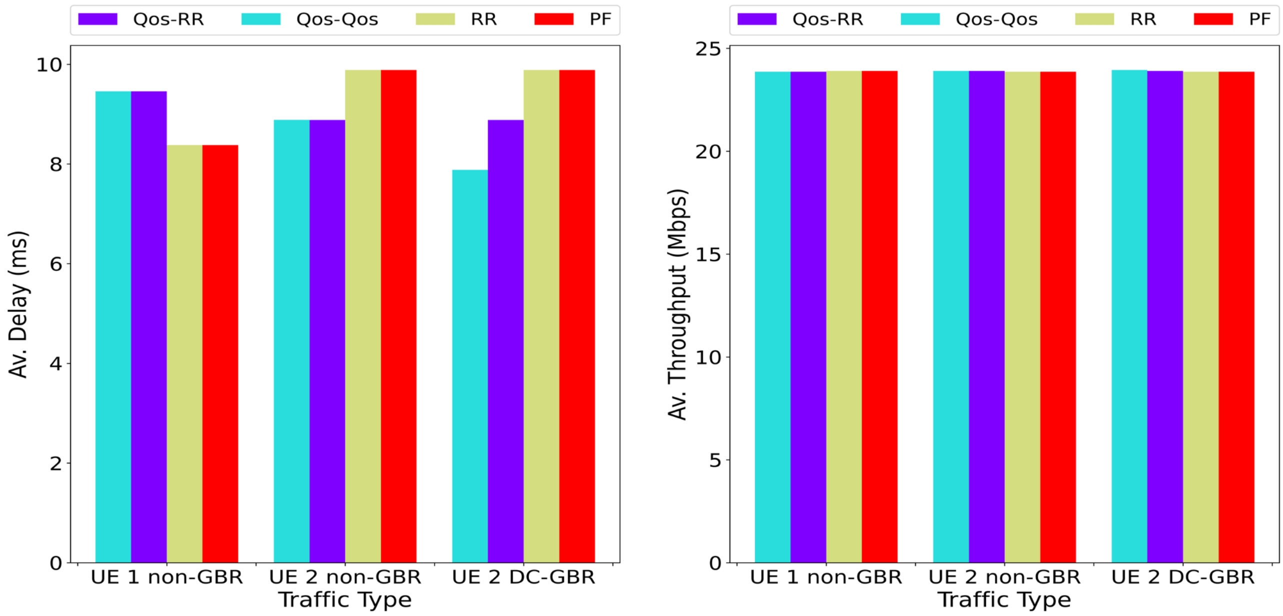


FIGURE 43. AVERAGE DELAY AND AVERAGE THROUGHPUT FOR 50 MHZ BANDWIDTH.

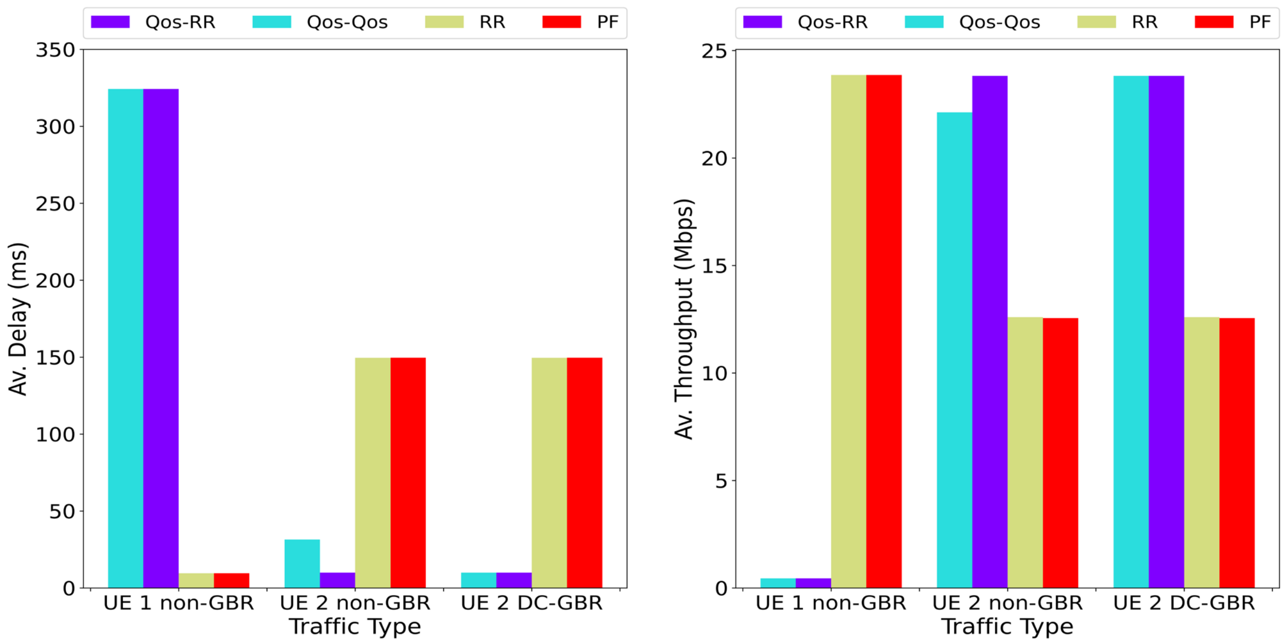


FIGURE 44. AVERAGE DELAY AND AVERAGE THROUGHPUT FOR 10 MHZ BANDWIDTH.

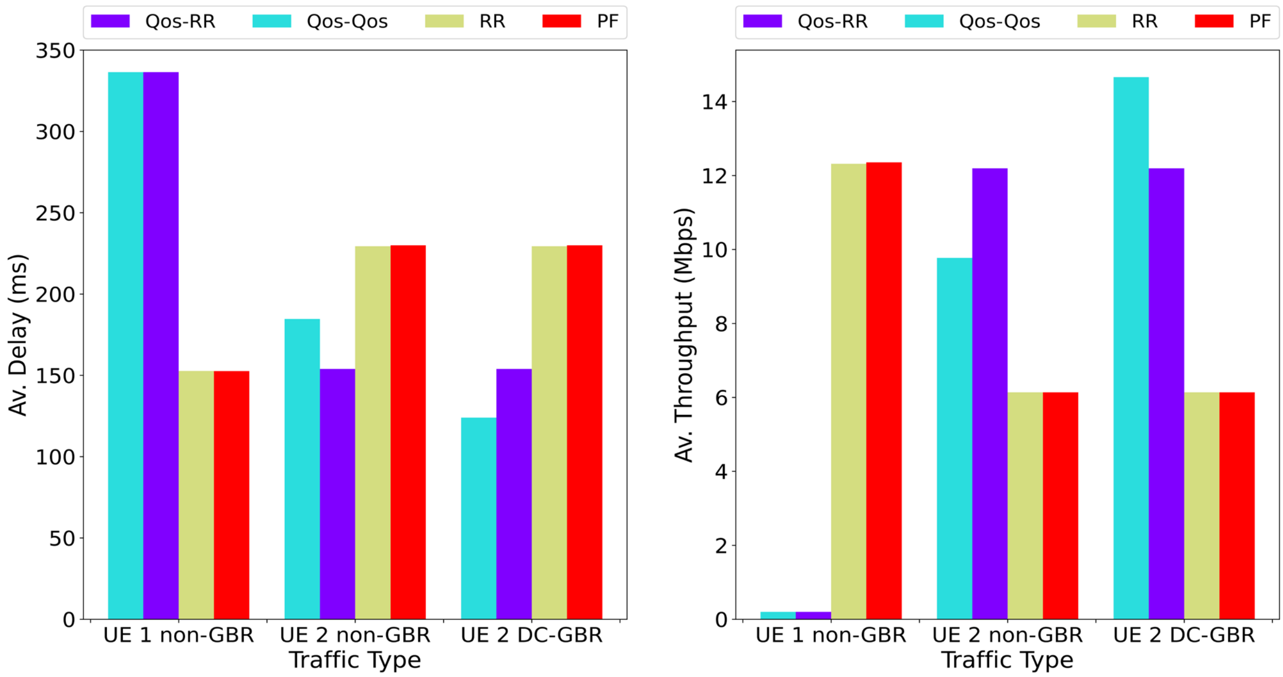


FIGURE 45. AVERAGE DELAY AND AVERAGE THROUGHPUT FOR 5 MHZ BANDWIDTH.

In addition, all the schedulers available in 5GLENA were tested under various network scenarios, and the simulation results show their effectiveness in meeting QoS requirements for XR traffic. Figure 46 shows some boxplots of the throughput achieved by every UE in a scenario with VR and CG UEs coexisting. The lower and upper edges of the box correspond to the 25th and 75th percentiles, respectively. Besides, the whiskers indicate 1.5 times the Inter Quartile Range (IQR), which represent approximately the 2 and 98 percentiles. A circle marks the average values. A range of bandwidths has been also added, starting from 40 MHz, which corresponds to a configuration where the capacity is not enough to meet traffic and QoS requirements, and goes up to 100 MHz. The corresponding throughput requirements (e_rabGuaranteedBitRate) are 45 Mbps and 30 Mbps for VR and CG UEs, respectively. Among all the evaluated algorithms, only the QoS-based schedulers (QoS and DPPH) explicitly prioritize VR traffic, reflecting its more stringent latency and throughput requirements.

Financiado por la Unión Europea NextGenerationEU
Plan de Recuperación, Transformación y Resiliencia
UNICO I+D
GOBIERNO DE ESPAÑA
cttc Centre Tecnològic de Telecomunicacions de Catalunya

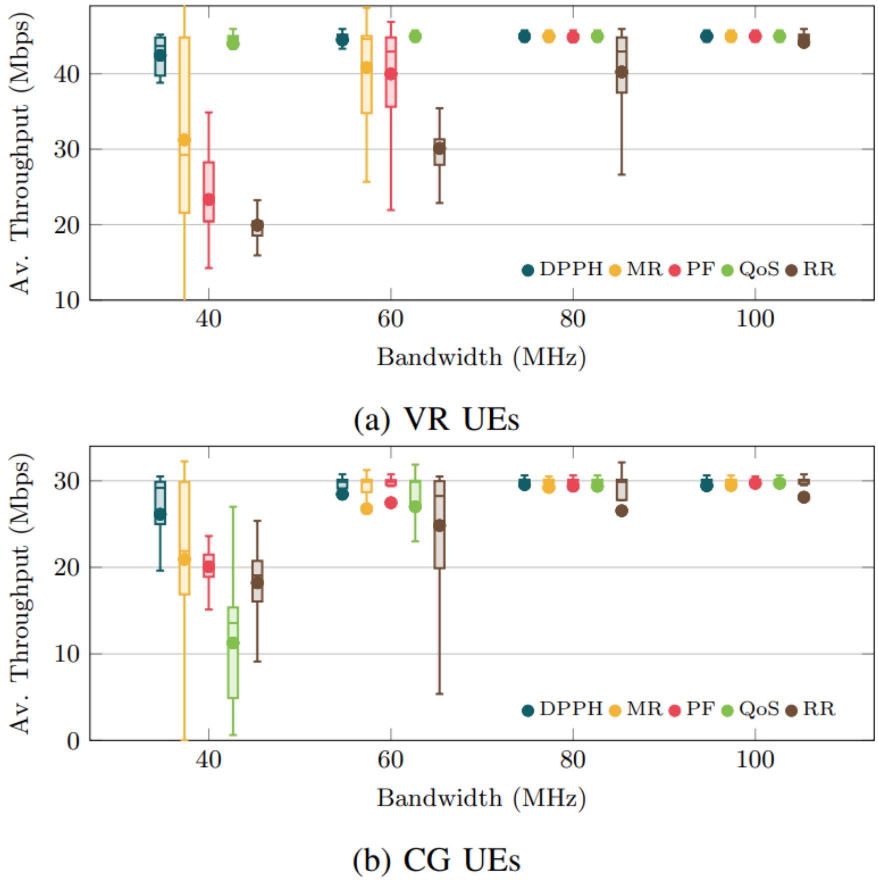


FIGURE 46. AVERAGE THROUGHPUT FOR DIFFERENT BANDWIDTH OPTIONS AND TRAFFIC APPLICATIONS.

2.8.4 Conclusions

This work presented a novel QoS MAC scheduler for multi-flow QoS provisioning, that accounts for the QoS characteristics of a flow by combining QoS indicators with real-time MAC measurements. To enforce QoS provisioning at the flow level, we additionally proposed and implemented a QoS LC assignment algorithm, that enables efficient handling of multi-flow traffic based on specific QoS requirements. The implementation included various extensions, both in the ns-3 LTE module and the 5G-LENA simulator, such as support for delay-critical GBR resource type flows, Release 18 5QIs and PDCP discard timers. The evaluation of the proposed solution has been carried out through end-to-end system-level simulations under non-saturation and saturation network conditions. It has been demonstrated that the proposed QoS MAC scheduler is able to provide QoS provisioning based on the QoS characteristics of each flow and correctly prioritize delay-critical traffic through the combination of QoS metrics and real-time measurements.

Furthermore, the proposed Lyapunov-based scheduler effectively addresses the stringent QoS requirements of XR traffic in 5G networks. Its ability to dynamically allocate resources based on real-time backlog information ensures system stability and high performance under varying network conditions. The scheduler achieves a balance between queues stabilization, throughput and delay guarantees, and resource efficiency, making it suitable for demanding applications like XR.

2.9 SMART-K2.6: QoS strategies at the transport for multi-service aggregation

2.9.1 System Design

2.9.1.1 Fronthaul, Midhaul and Backhaul in the context of Transport Networks

In the case of the functional split of disaggregated radio networks, the transport infrastructure supports connectivity between various elements of the radio access and the 5G core network. It comprises three segments: Fronthaul (FH, between O-RU and O-DU), Midhaul (MH, between O-DU and O-CU), and Backhaul (BH, between O-CU and the UPF component of the 5G Core), each with specific roles, interfaces, and transport requirements. Figure 47 illustrates the aforementioned network segments.

FIGURE 47. FRONTHAUL, MIDHAUL AND BACKHAUL NETWORK SEGMENTS

The main requirements associated to the transport of FH, MH, and BH are described by O-RAN Alliance in [36]. The following tables from [36] summarize the main requirements of FH and MH.

TABLE 19. TRANSPORT REQUIREMENTS FOR FH AND MH: (A) ECPRI ONE WAY DELAY; (B) ECPRI FRAME LOSS RATIO; (C) FH/MH CAPACITY DIMENSIONING

(a)

Latency Class	Max. One-way Frame Delay Performance	Use case
High25	25 µs	Ultra-low latency performance
High75[1]	75 µs	For full NR performance with fiber lengths in 10km range
High100	100 µs	For standard NR performance with fiber lengths in 10km range
High200	200 µs	For installations with fiber lengths in 30km range
High500	500 µs	Large latency installations > 30 km
Medium	User Plane (slow) & C&M Plane (fast)	1 ms
Low	C&M Plane	100 ms

1. New requirement category added based on deployment needs.

Financiado por la Unión Europea NextGenerationEU
Plan de Recuperación, Transformación y Resiliencia
UNICO I+D
GOBIERNO DE ESPAÑA
cttc Centre Tecnològic de Telecomunicacions de Catalunya

(b)

CoS Name	Example use	Maxim One-way Frame Loss Ratio Performance
High	User Plane (fast)	10^{-7}
Medium	User Plane (slow), C&M Plane (fast)	10^{-7}
Low	C&M Plane	10^{-6}

(c)

	Fronthaul (O-RU <-> O-DU)		Midhaul (O-DU <-> O-CU)	
	Peak Capacity (Gbps)	Interfaces	Peak Capacity (Gbps)	Interfaces
Small site FR1 only	7,5	1x 10GE	2	1x 10GE
Small site FR2 only	14,57	1x 25GE	3,7	1x 10GE
Medium site FR1+FR2	110	6x 25GE 3X 10GE	28,4	1x 100GE
Large site FR1+FR2	264,8	18x 25GE	68,9	1x 100GE

2.9.1.1.1 Fronthaul Transport Requirements

The FH focuses primarily on the functional split 7-2x (lower PHY split) supported between O-RU and O-DU, as defined by O-RAN. From the point of view of protocols used for forwarding such traffic, both Ethernet VLAN-based or IP/UDP transport, supporting eCPRI and CPRI protocols can be used in principle. The traffic volume generated at the FH depends on the number of sectors, carriers, MIMO layers, and frequency bands (Sub-6 GHz and mmWave).

The One-way transport delay requirements vary by use case, ranging from ultra-low latency of 25 µs for fast user plane traffic to up to 500 µs for large latency installations (e.g., >30 km fiber length).

Regarding Frame Loss Ratio (FLR), stringent FLR requirements for high-performance classes are set at 10^$^{-7}$ for high user plane traffic, and 10^$^{-6}$ for control and management planes.

With respect to the dimensioning of the bandwidth needed for the FH, the fronthaul peak bandwidth is calculated based on the number of MIMO layers, resource blocks, numerology, and overhead control plane traffic. Typical site configurations (small, medium, large) have peak bandwidths ranging from a few Gbps (small sites) to tens of Gbps (large sites with massive MIMO). Importantly, statistical multiplexing and oversubscription are possible due to the user data-dependent nature of traffic.

Finally, tight synchronization and timing requirements, with relative time errors between O-DU and O-RU limited to ±1.5 µs to support TDD and radio coordination functions.

Financiado por la Unión Europea NextGenerationEU
Plan de Recuperación, Transformación y Resiliencia
UNICO I+D
GOBIERNO DE ESPAÑA
cttc Centre Tecnològic de Telecomunicacions de Catalunya

2.9.1.1.2 Midhaul Transport Requirements

The MH segment connects O-DU and O-CU components, supporting the 3GPP F1 and E1 interfaces. It also includes inter O-CU communication over the Xn interface. It carries split 2 traffic from the perspective of radio functional split.

From a protocol perspective, the MH is IP-based for all interfaces, with the control plane using IP/SCTP encapsulation and the user plane using IP/UDP/GTPv2. Multi-point logical connectivity at the IP layer is required due to multiple O-DUs and O-CUs interconnections.

In terms of performance requirements, the transport delay for MH is driven mainly by service latency targets, ranging approximately from 1.5 ms to 10 ms one-way delay. The flexibility in placement of Midhaul and Backhaul components (i.e., the specific deployment location of O-DU, O-CU and UPF) is essential to meet latency requirements, sometimes necessitating co-location of components.

From the bandwidth provisioning perspective, the MH considers statistical multiplexing gains similar to Backhaul, with the traffic dimensioning depending on the number of O-DUs per O-CU and the number of aggregated flows.

2.9.1.1.3 Backhaul Transport Requirements

The final segment to consider is the BH which connects the O-CU or combined RAN components to the 5G core network. In conventional RAN architectures, Backhaul connects the integrated O-RU+O-DU+O-CU at the cell site to the core, while in O-RAN like architectures the BH only covers the connection of the centralized O-CU to the core network components.

The BH supports the 3GPP control plane interfaces N1, N2, N4, and user plane interfaces N3, N9. As encapsulation protocol the BH uses IPv4/IPv6 with control plane over SCTP and user plane over UDP/GTPv2. Again, multi-point IP connectivity is essential in this case.

Latency varies in the case of BH depending on service type and can typically range from 1 ms to 50 ms one-way delay. The BH must support very large-scale networks, accommodating a huge number of O-RU locations. In any case, this will be highly dependent on the particular telecom operator deployment strategy.

For the BH, transport network must handle high data rates; example peak site bandwidths range from ~2 Gbps (small sites) to nearly 70 Gbps (large sites).

From a provisioning perspective, it can be considered the statistical multiplexing across sectors and carriers with different provisioning strategies (peak rate, lower bound, conservative bound).

Financiado por la Unión Europea NextGenerationEU
Plan de Recuperación, Transformación y Resiliencia
UNICO I+D
GOBIERNO DE ESPAÑA
cttc Centre Tecnològic de Telecomunicacions de Catalunya

2.9.1.2 Relevance of QoS for multi-service aggregation

Defining a robust and effective Quality of Service (QoS) strategy at the transport layer is critical in modern telecommunications networks, especially those supporting multi-service aggregation, which can be assumed to be the common scenario when telecom operators decide to deploy radio functional split disaggregated networks. It can be assumed that such O-RAN like deployments co-exist with the aggregation of other services such conventional (distributed) radio, fixed residential, or corporate / enterprise services, all of them using the same underlying transport network. Thus, transport networks often carry diverse traffic types, including delay-sensitive services such as voice and video, ultra-reliable low latency communications (URLLC) for industrial and 5G applications, as well as best-effort data like bulk file transfers and internet browsing. The coexistence of these heterogeneous traffic streams with varying performance requirements presents significant challenges that can only be addressed through carefully designed QoS mechanisms. Figure 48 represents a potential network aggregation topology aggregating conventional (distributed) RAN for 4G services in conjunction with the aggregation of 5G services supported by an O-RAN like functional split, based on [37].

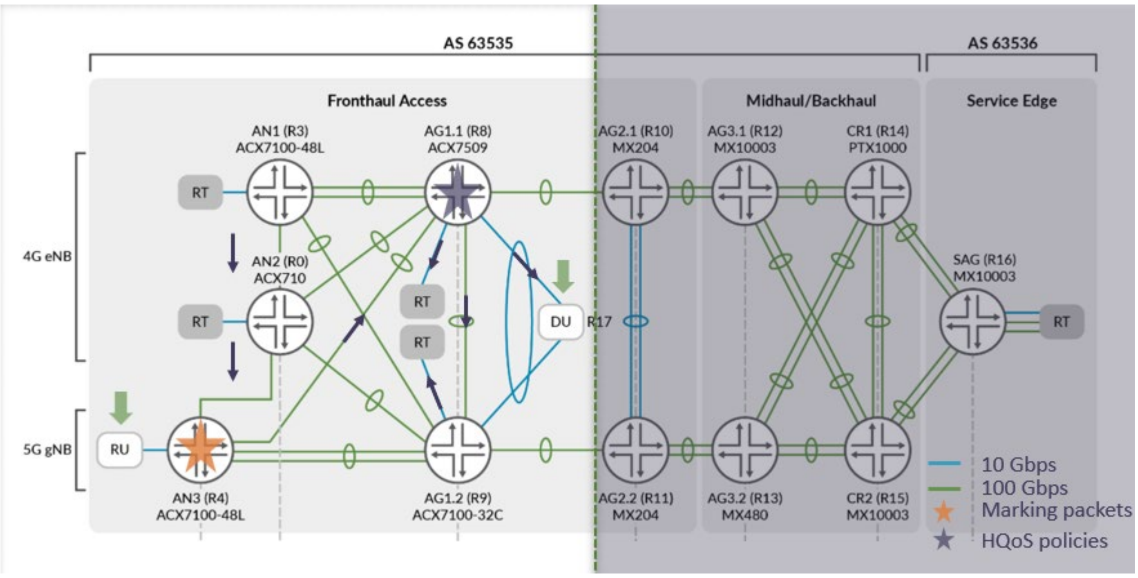


FIGURE 48. FRONTHAUL, MIDHAUL AND BACKHAUL TYPICAL NETWORK TOPOLOGY

At its core, QoS refers to the set of techniques and policies implemented to manage network resources and ensure that different classes of traffic receive appropriate treatment according to their priority, latency, jitter, and loss requirements. Without proper QoS strategies, critical applications may suffer degradation in performance, leading to poor user experience, violation of service-level agreements (SLAs), and ultimately, loss of revenue and reputation for service providers.

Implementing unified QoS strategies at the transport layer facilitates seamless integration and consistent treatment of traffic across different segments of the network.

One of the primary reasons for defining QoS at the transport layer is to enable the differentiation and prioritization of traffic flows. For example, in a multi-service aggregation scenario, voice and real-time video traffic require low latency and minimal jitter to maintain call quality and video smoothness. Conversely, non-critical data traffic such as email or file downloads can tolerate higher delays and variable throughput. By classifying and marking packets based on their service requirements, transport networks can allocate resources dynamically, ensuring that high-priority traffic is transmitted promptly, while lower-priority traffic utilizes remaining bandwidth without causing congestion.

Traffic management components such as classification, scheduling, queuing, shaping, and policing form the building blocks of an effective QoS framework. Classification involves identifying traffic flows based on parameters like IP addresses, protocol types, or QoS markings (e.g., DSCP, 802.1p, MPLS EXP). Scheduling and queuing mechanisms then determine the order and rate at which packets are transmitted. For instance, strict priority queueing ensures that delay-sensitive traffic is serviced ahead of others, while weighted fair queueing allows proportional bandwidth allocation among different classes. Shaping controls the traffic flow to smooth bursts and matches the available bandwidth, preventing congestion downstream. Policing enforces traffic profiles by dropping or remarking packets that exceed agreed limits, thus protecting the network and other users.

A well-defined QoS strategy also supports scalability and flexibility, which are vital as network demands evolve. Transport networks must accommodate increasing bandwidth, new service types, and dynamic traffic patterns without compromising service quality. Hence, QoS designs should allow for easy adaptation, such as adding new traffic classes or adjusting priority levels, without requiring extensive reconfiguration or causing service disruption.

Preserving QoS markings end-to-end across the transport network is another critical aspect. Many services rely on layered encapsulations and VLAN tagging, which can complicate the maintenance of QoS indicators. Therefore, transport equipment and protocols must support mechanisms to classify, rewrite, and preserve QoS codepoints through various encapsulation layers, ensuring that traffic receives consistent treatment from ingress to egress points.

In addition to performance benefits, effective QoS strategies enable service providers to offer differentiated services tailored to customer needs. By guaranteeing specific performance levels for premium services, providers can meet stringent SLAs, enhance customer satisfaction, and create new

Financiado por la Unión Europea NextGenerationEU
Plan de Recuperación, Transformación y Resiliencia
UNICO I+D
GOBIERNO DE ESPAÑA
cttc Centre Tecnològic de Telecomunicacions de Catalunya

revenue streams. This differentiation is increasingly important in competitive markets where customers expect customized service experiences.

Finally, transport QoS strategies must also consider resilience and reliability. Networks should maintain QoS guarantees even under failure scenarios such as link congestion, device restarts, or configuration changes. Robust QoS implementations include mechanisms to detect and mitigate issues like congestion-induced packet loss or delay spikes and support traffic prioritization across redundant paths.

2.9.2 Implementation

2.9.2.1 Experimental characterization of Transport Network for supporting FH

From the previous analysis, the FH emerges as the network segment presenting more stringent requirements. It is then key to understand to what extent a router can impact in terms of performance metrics to the FH traffic flows.

In order to better understand such impacts, an experimental test was performed by Telefónica at the European Open Testing and Integration Centre (OTIC) in Madrid[9] hosted in the Technology and Automation lab in Telefónica Headquarters.

The basic scenario of testing is described in Figure 49.

[9] https://www.o-ran.org/otics/european-otic-in-madrid

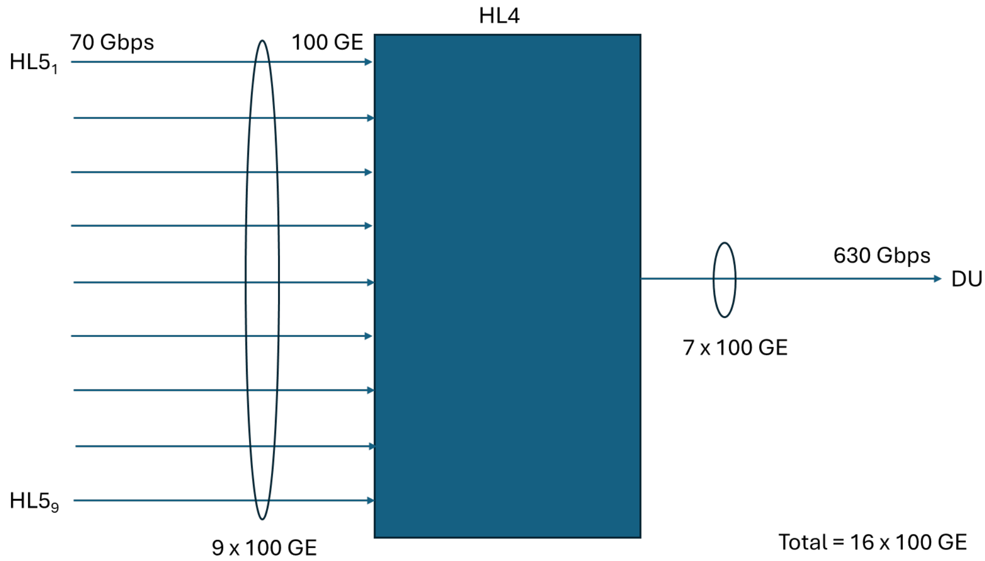


FIGURE 49. TEST SCENARIO AT TELEFONICA OTIC

The device under test is a Cisco ASR9000 representing a state-of-the-art router. It is assumed that the router aggregates up to nine sites consisting of 3 O-RUs each of them which are supposed to be aggregated by additional routers playing the role of cell site routers, denoted as $HL5_1$ to $HL5_9$. The device under test, noted as HL4, is considered to have the O-DU directly connected to it.

The dimensioning scenario for the FH traffic considers up to 70 Gbps per radio site, which implies 1x 100 GE per site, making a total of 630 Gbps to be delivered towards the O-DU. In order to forward that traffic, it is assumed that the O-DU is connected to the device under test by means of 7x 100 GE interfaces.

In order to characterize the behavior of the router in terms of latency and jitter, a traffic generator, the Keysight AresONE is used, injecting the flows described before. Different packet sizes are used for that. From the point of view of traffic volume, 630 Gbps are injected as baseline even though 700 Gbps are also tested in order to understand the behavior of the router in situation close to link saturation. The test is repeated in both upstream (from O-RU to O-DU) and downstream (from O-DU towards the O-RUs) directions for completeness, despite the fact that downstream is the direction congruent with the real flows in O-RAN (i.e., the greatest volume of traffic happens from the O-DU to the O-RUs). Figure 50 illustrates one of the samples of latency and jitter collection as an example of the test run.

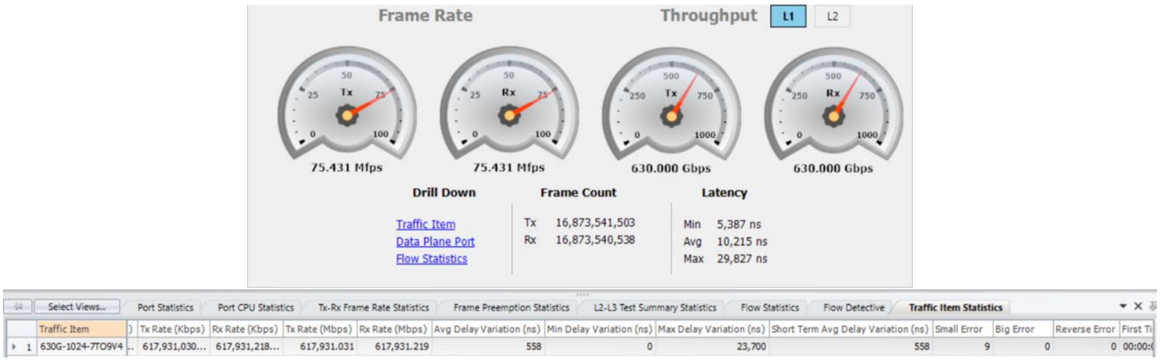


FIGURE 50. SCREENSHOT EXAMPLE OF THE CAPTURED DATA DURING FH CHARACTERIZATION EXAMPLES

2.9.3 Results

The following Table 20 summarizes the measurements performed.

TABLE 20. SUMMARY OF TESTED RESULTS IN TELEFONICA OTIC

Direction	Traffic [Gbps]	Packet Size [B]	Avg. Latency [µs]	Avg. Jitter [ns]
Downstream	630	128	9,83	104
		512	9,99	330
		1024	10,21	558
	700	128	9,77	82
		512	10,08	263
		1024	10,25	449
Upstream	630	128	9,59	65
		512	9,89	220
		1024	10,07	383
	700	128	10,54	47
		512	10,73	169
		1024	11,00	322

Financiado por la Unión Europea NextGenerationEU
Plan de Recuperación, Transformación y Resiliencia
UNICO I+D
GOBIERNO DE ESPAÑA
cttc Centre Tecnològic de Telecomunicacions de Catalunya

2.9.4 Conclusions

The results obtained during the experimental characterization of Fronthaul (FH) traffic at Telefónica's OTIC environment demonstrate that latency and jitter metrics remain within extremely stable margins, even under traffic conditions close to link saturation. In all evaluated cases—both downstream and upstream, and with different packet sizes—average latencies consistently stayed around 10 µs, while jitter remained below 0.6 µs (600 ns), reaching values as low as 47 ns in certain configurations. These results confirm that an aggregation router such as the Cisco ASR9000 can deliver deterministic performance, which is critical for the transport of high-demand FH traffic.

Given that O-RAN deployments involve the coexistence of fronthaul traffic with other services (such as residential, enterprise, or backhaul traffic) over a converged transport network, it is essential to define and enforce strict Quality of Service (QoS) strategies that guarantee the logical isolation of FH traffic. In this context, assigning the highest priority to FH traffic—especially in split 7-2x scenarios—is key to meeting its latency, jitter, and synchronization requirements, shielding it from congestion caused by less critical services.

These timing considerations are particularly crucial in disaggregated radio access scenarios, where precise coordination between O-DU and O-RU units depends on strict adherence to synchronization and delay requirements. The experimental evidence gathered supports the technical feasibility of transporting FH traffic over general-purpose routers, provided that an appropriate QoS design is in place to ensure its prioritization. These results are applied in Use Case 2, PoC 1: Joint Fronthaul Capacity Control/Placement, QoS management and Flexible Functional Splits for characterizing the network. In particular, this PoC explores the viability of achieving centralization in an O-RAN mobile network, and these latency results were key to analysing feasible distances for locating O-RAN components while meeting the strict requirements across the different segments of the transport network.

2.10 SMART-K3.1: AI/ML Workflows in O-RAN

2.10.1 System Design

AI/ML workflows are fundamental to the O-RAN (Open Radio Access Network) architecture, enabling a data-driven and highly automated RAN. O-RAN introduces an open, disaggregated, and modular framework in which intelligence is embedded through the Near-Real-Time RIC (near-RT RIC) and Non-Real-Time RIC (non-RT RIC), both of which orchestrate AI/ML models and policy-driven control [39].

Financiado por la Unión Europea NextGenerationEU
Plan de Recuperación, Transformación y Resiliencia
UNICO I+D
GOBIERNO DE ESPAÑA
cttc Centre Tecnològic de Telecomunicacions de Catalunya

At the core of AI/ML enablement in O-RAN is the O-RAN AI/ML Workflow Framework, which defines how ML models are trained, validated, deployed, and monitored using standardized interfaces and data structures [40]. These components interact through the A1, E2, and O1 interfaces, which coordinate model lifecycle management, data collection, and inference execution [41], [42].

Key technical roles of AI/ML workflows in O-RAN include:

1. Data Collection and Feature Engineering: RAN data such as CQI reports, PRB utilization, mobility events, and interference measurements is collected via O1 and E2 telemetry. These datasets feed ML pipelines for training and continuous optimization, enabling accurate anomaly detection and network state prediction [40], [42].
2. Model Training and Optimization: The non-RT RIC executes advanced AI/ML tasks including supervised learning (e.g., traffic forecasting), clustering (e.g., load balancing), and reinforcement learning (e.g., resource allocation). Its cloud-native design provides the computational scale needed for model retraining and performance tuning [40].
3. Inference in the Near-Real-Time Loop: Once deployed, models run in near-RT RIC xApps that support inference within tight latency bounds (10 ms–1 s). These xApps enable closed-loop optimization functions such as mobility robustness, interference management, and dynamic spectrum allocation [39].
4. Model Lifecycle Management: The A1 interface allows the non-RT RIC to push ML model updates, enrichment information, or high-level policies to the near-RT RIC. This ensures continuous adaptation of the models to changing traffic conditions and operator objectives [41].
5. Interoperability and Vendor-Neutral Integration: O-RAN standardization ensures that AI/ML workflows remain interoperable across multi-vendor RAN components. Model formats, data schemas, and service models (E2SMs) are defined to avoid proprietary lock-in and to accelerate innovation [39], [42].

Overall, AI/ML workflows in O-RAN are essential for moving toward a fully autonomous RAN capable of self-optimization, self-configuration, and predictive management. They enable operators to leverage data-driven intelligence, improve energy efficiency, enhance spectral usage, and support increasingly complex 5G and 6G services In line with this design, we also aim to integrate Large Language Models directly into the workflow. The goal is to develop an O-RAN monitoring platform that not only collects and visualizes RAN metrics but uses LLMs to interpret the data, detect issues, and support automated decision making. This creates a clear pathway toward an LLM-enhanced monitoring system for O-RAN.

Financiado por la Unión Europea NextGenerationEU
Plan de Recuperación, Transformación y Resiliencia
UNICO I+D
GOBIERNO DE ESPAÑA
cttc Centre Tecnològic de Telecomunicacions de Catalunya

2.10.2 Implementation

Our proposed monitoring and controlling system leverage the parallel processing capabilities of Large Language Models (LLMs) to extend its functionality across a wide variety of tasks within the O-RAN system. By utilizing the long context length of LLMs, tool integrations, Retrieval-Augmented Generation (RAG) frameworks, and agentic workflows, the system achieves enhanced efficiency and adaptability.

A detailed description of the proposed adaptive monitoring system is outlined and developed together with 6G BLUR JOINT Key Concept 3.1 *Agentic Framework in the RAN monitoring platform*. The core idea of this approach is to interact with O-RAN components (deployed on bare metal, Docker, or Kubernetes), create a centralized logging system and make it searchable, diagnose problems based on logs and infrastructure status, and provide detailed documentation or tools to the agents whenever needed by creating a custom RAG solution. Finally, to orchestrate all these activities, we developed an advanced agentic workflow that integrates all components seamlessly.

The proposed framework can also be used for controlling O-RAN components through the deployment of xApps. For example, the LLM-based agentic workflow can be thought of as an equivalent to a sophisticated text-based policy augmented with arbitrary conditions defined through conditional instructions for agent operations.

Upon observing unexpected changes in the system (by means of using a monitoring xApp), the xApp can modify network behaviour by optimizing the resources available to the UEs. In all these scenarios, the provided agentic workflow can utilize external tools, such as optimization packages or different Python libraries, to generate more efficient solutions.

2.10.3 Results

The workflow is organized around the coordinated operation of two complementary xApps that exploit the O-RAN Key Performance Measurements (KPMs) and RAN Control (RC) capabilities developed by srsRAN[10]. The first xApp is dedicated to continuously monitoring traffic levels and PRB utilization across the radio access network. Using the standardized O-RAN KPM service, it gathers fine-grained performance indicators such as throughput, load distribution, PRB occupancy, and

[10] https://github.com/srsran/oran-sc-ric/tree/main

temporal fluctuations in demand. These measurements are processed in real time to build an accurate and up-to-date representation of network conditions, providing essential situational awareness regarding the behaviour of the radio interface. This xApp effectively acts as the sensing layer of the system, supplying the detailed operational data needed for higher-level decision making.

The second xApp consumes these KPMs and focuses on analysing traffic conditions to determine whether the current allocation of PRBs is optimal. By applying decision logic and policies enabled through the srsRAN implementation of the O-RAN RC framework, the xApp evaluates potential imbalances, congestion scenarios, or resource underutilization. When necessary, it formulates and issues RC control actions to modify PRB assignments dynamically, adapting resource allocation to better match the observed traffic patterns. This may involve increasing PRBs for heavily loaded cells, reducing allocations where demand is low, or balancing resources across the network to improve overall performance and spectral efficiency.

Together, these two xApps establish a continuous, closed-loop process that follows the observe–analyze–act paradigm. The monitoring xApp ensures constant visibility into real-time network behaviour, while the decision-making xApp transforms these insights into targeted control actions aimed at optimizing the radio resources. This workflow demonstrates how O-RAN-compliant xApps, supported by the srsRAN implementation of the KPM and RC specifications, can enable intelligent, autonomous, and responsive RAN management capable of adapting proactively to dynamic traffic conditions.

2.10.4 Conclusions

This KC is focused on the AI/ML workflows in O-RAN components, enabling the self-adaptation of RAN components to changes in traffic. The key point is the integration of all necessary components at the RAN level with a LLM platform to facilitate the monitoring and controlling of O-RAN networks. This approach empowers B5G/6G network management to dynamically adapt and scale according to operational requirements, facilitating autonomous optimization decisions driven by closed-loop feedback mechanisms. The outcomes of the work developed in SMART K1.1, K2.1 and JOINT K3.1, key concepts, are the pillars for the realisation of UC3PoC3 presented in Section 3.2.3.

3 Use cases and E2E results

3.1 UC2: Joint RAN and Transport mechanisms for 6G Disaggregated Mobile networks

3.1.1 PoC1: Joint Fronthaul Capacity Control/Placement, QoS management and Flexible Functional Splits

3.1.1.1 Introduction

This proof of concept explores and evaluates the various possibilities that emerge when implementing different distributed approaches within a mobile network infrastructure, specifically incorporating O-RAN components in the RAN domain. Given that centralization is one of the most prominent and beneficial scenarios in Open RAN, offering significant advantages to Mobile Network Operators (MNOs), a key objective of this PoC is to assess the viability of achieving centralization beyond the conventional limits and to explore deployment flexibility. Traditionally, such centralization has been constrained to distances of approximately 10–20 km, relying on dedicated transport solutions to meet strict round-trip time (RTT) requirements in the fronthaul. Instead, this PoC investigates deeper levels of centralization by pushing past these boundaries, aiming to support centralization over distances exceeding 20 km (equivalent to RTTs over 250 µs), by relocating the O-DU away from the cell site.

This works builds upon several key developments:

- **SMART-K1.1** – CU and DU implementation: implementation of O-RAN Central Unit (CU) and Distributed Unit (DU) components using the srsRAN software.
- **SMART-K1.2** – RAN orchestration: enhancement of the FH traffic conditions by integrating various FH control methods.
- **SMART-K2.1** – Development and characterization of FH and MH: extension and improvement of the O-FH library within the srsRAN Project CU/DU solution.
- **SMART-K2.2** – Characterization of the FH and MH transport: study specially on the FH to understand its behaviour under centralized topologies.
- **SMART-K2.3** – Flexible functional split: calculation of the fronthaul requirements that would be needed to support a 6G network and implementation of various functional splits.
- **SMART-K2.4** – Radio stack optimization: adaptations in DU SW to allow for longer FHs.

Financiado por la Unión Europea NextGenerationEU
Plan de Recuperación, Transformación y Resiliencia
UNICO I+D
GOBIERNO DE ESPAÑA
cttc Centre Tecnològic de Telecomunicacions de Catalunya

- **SMART-K2.5** – QoS Management for time-critical immersive applications: design and implementation of generalized QoS MAC schedulers and Lyapunov-based schedulers.
- **SMART-K2.6** - QoS strategies at the transport for multi-service aggregation.
- **JOINT-K2.1** – Transport network optimization: optimization of the transport network within an ORAN-based mobile network.
- **JOINT-K2.2** – Entities' placement decisions: algorithms development commonly known as Virtual Network Embedding (VNE).
- **JOINT-K2.3** – QoS policies and scheduling: implementation of HQoS-aware (Hierarchical Quality of Service) policies.

3.1.1.2 System Design Considerations

As previously seen, UC2 PoC1 combines a great number of key concepts. The following subsections provide details about the interactions among different key concepts by grouping the main contributions of this PoC into different topics.

3.1.1.2.1 FH characterization, Network characterization and O-DU optimization

3.1.1.2.1.1 FH characterization

First, in order to characterize different FH profiles within centralized topologies, we developed a custom software using MATLAB[11]. This tool was designed to, based on a set of input parameters related to the network, the O-RU and the O-DU, determine the most suitable centralization scenario.

By processing these inputs, the software is capable of analyzing multiple configuration possibilities and selecting the optimal deployment strategy. As output, it provides the specific configuration settings required for the O-DU to implement the chosen centralization scheme effectively.

The tool not only automates the evaluation process but also facilitates informed decision-making by offering a clear mapping between the network conditions and the necessary adjustments on the O-DU side. This allows for a systematic and efficient characterization of FH profiles, ensuring that the proposed configuration maximizes performance while aligning with network requirements and constraints. Detailed information about this development can be found in Section 2.5 (SMART-K2.2).

3.1.1.2.1.2 Network characterization

For this work, we have considered Telefónica IP/Fusion network, whose initial aggregation levels are defined as HL5, HL4, and HL3. Typically, the O-DU is deployed directly at the cell site, corresponding

[11] Available online at: https://gitlab.cttc.es/mdi-6gblur/smart/-/tree/main/KeyConcepts/SMART-K2.2%20Characterization%20of%20the%20FH%20and%20MH%20transport

to the HL5 level. However, in this study, we have carried out an in-depth analysis of the distribution of routers within the network, as well as their geographical distances and aggregation ratios, in order to identify the temporal and capacity-related parameters necessary to enable the relocation of the O-DU to higher aggregation levels—specifically, to HL4 (an intermediate scenario) or HL3 (the ideal scenario).

This analysis involved mapping the network topology to understand how traffic is aggregated and propagated through these hierarchical layers. By assessing both the physical distances between nodes and the logical capacity constraints at each level, we were able to estimate critical metrics such as transport latency, bandwidth availability, and the scalability impact associated with centralizing the O-DU functions further upstream.

The ultimate objective of this evaluation is to determine the feasibility and performance trade-offs of shifting the O-DU from the edge (HL5) towards more centralized locations (HL4 or HL3), thereby optimizing resource utilization, simplifying network management, and potentially reducing operational costs, all while ensuring compliance with the stringent latency and capacity requirements inherent to fronthaul networks.

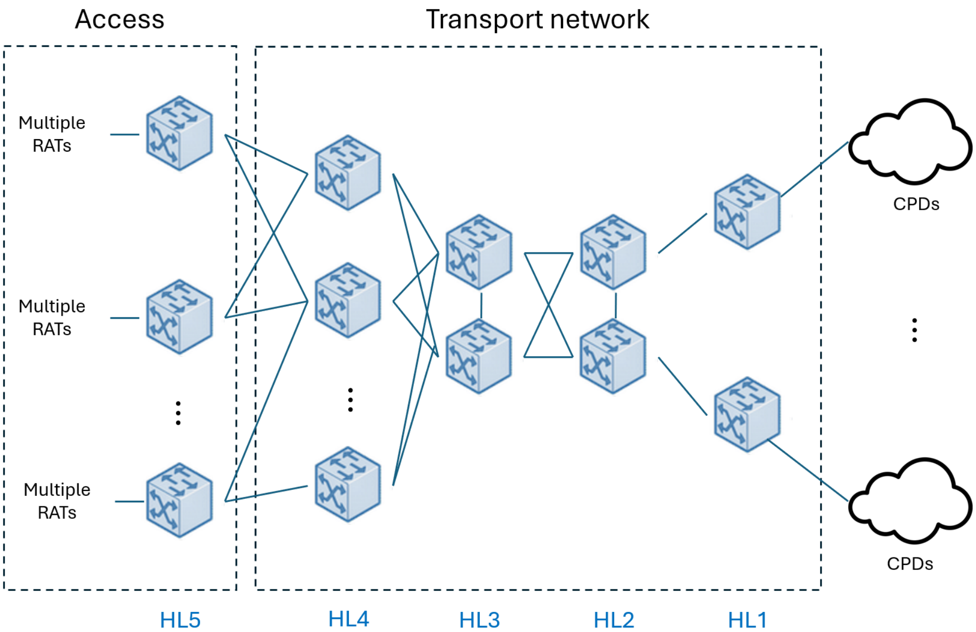


FIGURE 51. TELEFÓNICA´S TRANSPORT NETWORK

In addition, we have identified and dimensioned the necessary link capacities required to handle the estimated traffic in both DL and UL that would arise under this centralization scenario. This involved a detailed assessment of the expected traffic load associated with relocating the O-DU to higher aggregation levels and quantifying the bandwidth demands that such a shift would impose on the underlying transport network.

As part of this process, we analyzed a determined cell-site configuration to estimate peak data rates in both DL and UL directions. The reference capacity is based on a tri-sectorial site configuration where one sector operates at peak capacity while the other two sectors run at average capacity, applying C-band (MIMO: 32T32R; BW: 100 MHz) and B1/B3 bands (MIMO: 4T4R; BW: 20 MHz), and leveraging pooling gains from centralization. Combining the three sectors and using BFP9 compression method, each site will gather, in the worst case, a traffic of $\approx$70 and $\approx$28 Gbps, in DL and UL, respectively. Based on these projections, we determined the minimum link capacities that would be needed to ensure reliable transport of fronthaul data without congestion or degradation of service quality, with an aggregated capacity of 700 Gbps that was required for the HL5–HL4 links, and 1700 Gbps for the HL4–HL3 and HL3–HL3 links.

Furthermore, we carefully evaluated empirically the timing behavior of the routers involved under these conditions. This included calculating the end-to-end latency introduced by the additional transport distance, as well as examining processing delays, queuing effects, and jitter at each network node. More information is explained in SMART K2.6 (section 2.9).

For that purpose, we defined the networks routers including a switching matrix with a maximum capacity of approximately 300 Gbps for HL5, 2 Tbps for HL4 and 4 Tbps for HL3, being commercial numbers available in the network equipment and greater that the worst-case traffic to ensure enough capacity.

In this sense, a testbed was set up to characterize the delay introduced by HLx routers. Specifically, an HL4 setup was implemented using a commercial router. By injecting traffic at capacities of 630, 700, and 701 Gbps, combined with packet sizes of 128, 512, and 1024 bytes, it was observed that average delays remained consistently around 10 µs, with jitter staying below 0.5 µs across all scenarios. These findings suggest that similar performance can be expected at other HLx levels, since increases in aggregate capacity are typically matched by corresponding scaling in the switching matrix.

Furthermore, considering that O-FH traffic will likely share transport with other services (e.g., fixed broadband) over a multi-service network, it is essential to enforce strict prioritization of O-FH packets. Due to their stringent timing constraints, giving O-FH traffic the highest priority ensures that it remains isolated from congestion effects, maintaining stable delay and minimal jitter regardless of surrounding traffic conditions.

Financiado por la Unión Europea NextGenerationEU
Plan de Recuperación, Transformación y Resiliencia
UNICO I+D
GOBIERNO DE ESPAÑA
cttc Centre Tecnològic de Telecomunicacions de Catalunya

These timing considerations are critical in fronthaul scenarios, where strict latency and synchronization requirements must be met to ensure proper coordination between the O-DU and O-RU.

3.1.1.2.1.3 Network impairments simulations

In order to realistically replicate the transport conditions and analyze the delay experienced from the O-DU to the O-RU over a FH deployed within the transport network, we carried out simulations using the ns-3 event-driven simulator. The impairments modeled in ns-3 can be summarized as follows: (i) baseline transport delay, which includes propagation, transmission, and switching times; and (ii) maximum delay, which results from delay variation caused by network load and traffic dynamics. The various capabilities developed in this work are summarized in JOINT K2.3.

To enhance realism, we integrated packet processing delays introduced by each network node, based on measurements obtained from Telefónica's testbed, into the simulations. Additionally, the network topology was configured to reflect the architecture of Spain's IP/Fusion network.

Network-induced variability was evaluated by analyzing the one-way delay, focusing solely on the downlink direction. Each site aggregates traffic due to a tri-sector configuration under a defined load level, as previously described. In this context, the impact of network impairments, together with propagation delay, is assessed in a realistic scenario involving fragmentation, with an MTU set to 1500 bytes.

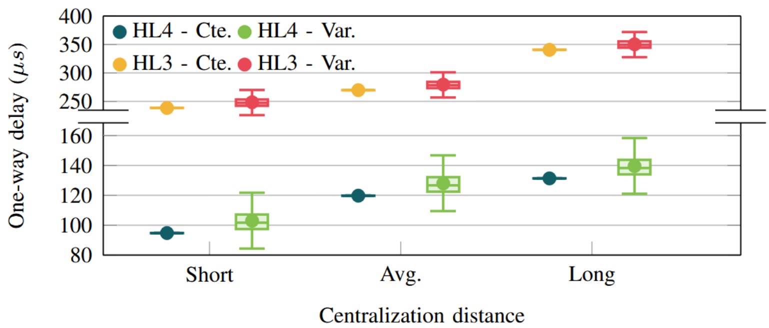


FIGURE 52. ONE-WAY O-FH DELAY PERFORMANCE FOR O-DU AT HL3 AND HL4

Router-induced delays were considered using two approaches. In the first, each node is assigned a constant processing delay, matching the average measured value of 10 µs. This case is shown in Figure 52 as HL3-Cte and HL4-Cte. Each boxplot illustrates the variability of the open fronthaul delay, accounting for both propagation time and router processing delay, with queuing delays being negligible (on the order of nanoseconds).

In the second approach, delays at each node are modeled as a shifted exponential distribution, with minimum, average, and maximum delays of 5 μs, 10 μs, and 35 μs, respectively, based on measurements from the HL4 testbed. As shown in Figure 52 (HL3-Var and HL4-Var), the accumulation of variability across multiple nodes results in a dispersion of approximately ±20 μs (represented by the lower and upper whiskers) around the mean. As can be seen, the variability introduced by the network interfaces is minimal. However, the processing tasks performed by intermediate routers can significantly contribute to the overall delay variability.

3.1.1.2.1.4 FH characterization analytical results

To ensure accurate calculations of the temporal relationships, the developed tool is based on the CUS-planes specification [22] defined by the O-RAN Alliance. This specification outlines the precise timing relationships that must be maintained between the O-RU and the O-DU, including the corresponding time windows, delay management use cases, and other critical synchronization parameters.

By adhering to the guidelines established in the CUS-planes specification, the tool is able to rigorously model the FH interface and correctly account for the stringent latency and jitter requirements inherent to this link. The specification provides detailed descriptions of the allowable delay budgets, synchronization tolerances, and transport layer behaviors necessary to ensure seamless interaction between the O-RU and O-DU under different deployment scenarios. For example, in Figure 53 timing relations per symbol IQ in DL direction for control planes and user planes are shown.

Financiado por la Unión Europea NextGenerationEU
Plan de Recuperación, Transformación y Resiliencia
UNICO I+D
GOBIERNO DE ESPAÑA
cttc Centre Tecnològic de Telecomunicacions de Catalunya

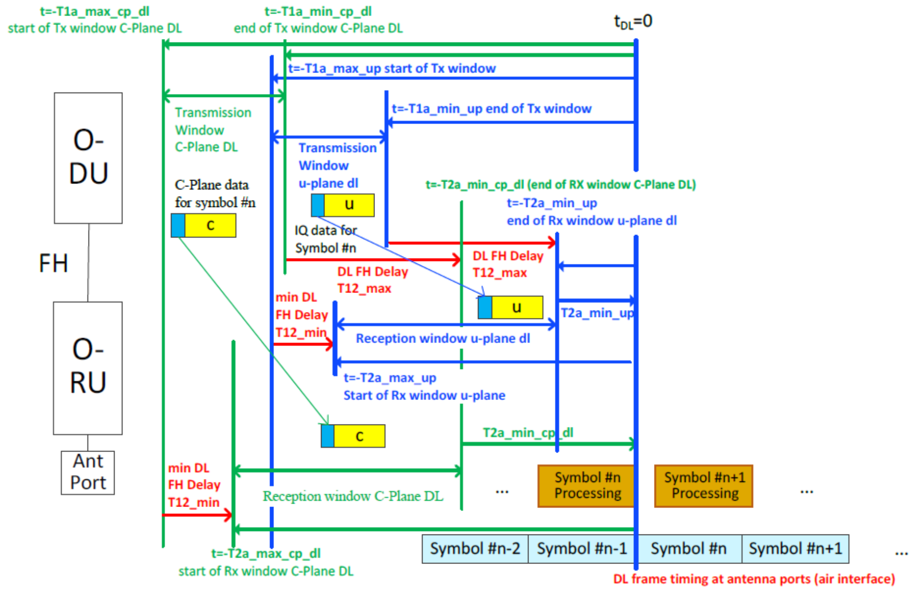


FIGURE 53. DIAGRAM OF TIMING RELATIONS IN FH IN DL DIRECTION

Incorporating these standardized constraints into the tool ensures that all calculated network configurations are not only theoretically optimal but also compliant with industry-accepted performance criteria. This guarantees that any proposed relocation of the O-DU or modification of the network architecture will continue to meet both system requirements and operational demands imposed by O-RAN-based disaggregated radio access networks.

Given the specific centralization conditions of our scenarios, in our case it is possible to extend the degree of centralization simply by relaxing the transport requirements, that is, by loosening the constraints imposed on the network routers. By allowing for slightly less stringent transport conditions, such as increased latency budgets or reduced synchronization precision along the fronthaul links, the O-DU can be relocated to higher aggregation levels more easily, thereby broadening the centralization possibilities.

Visually, the performed changes in the DL can be seen in Figure 54. It can be observed that we have successfully extended T12_max by Δt time units in order to align it with the optimal transmission window (dashed line) of the O-DU. By doing so, we have effectively increased the maximum allowable fronthaul (FH) delay, thereby providing greater flexibility in the transport network design. By adjusting T12_max to match the O-DU's optimal transmission window, we ensure that data transmission remains synchronized and compliant, even as we push the limits of centralization by relocating the O-DU further upstream.

Financiado por la Unión Europea NextGenerationEU
Plan de Recuperación, Transformación y Resiliencia
UNICO I+D
GOBIERNO DE ESPAÑA
cttc Centre Tecnològic de Telecomunicacions de Catalunya

Practically, this means that the fronthaul network can accommodate longer physical distances while still maintaining the performance requirements. The adjustment creates a more forgiving delay budget that can absorb variabilities in transport, such as processing delay and queuing effects, which would otherwise constrain centralization options.

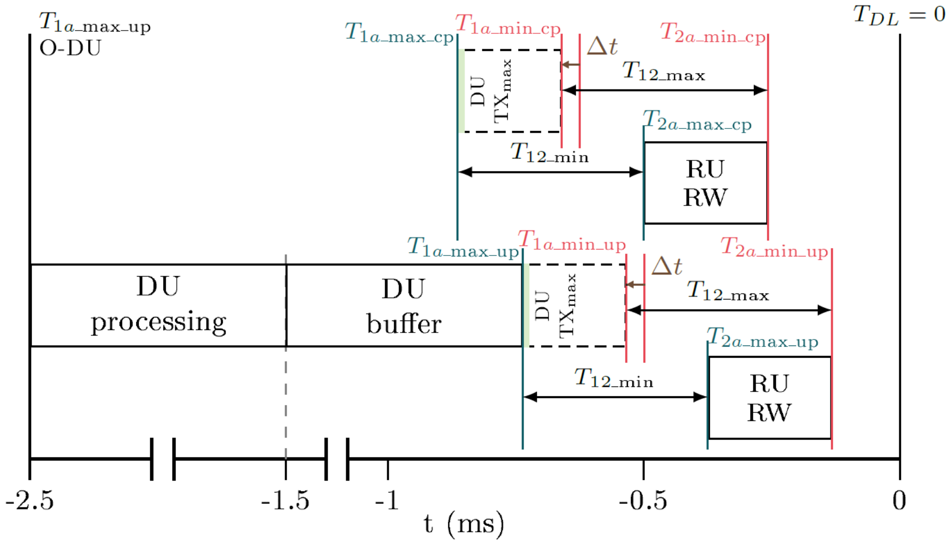


FIGURE 54. FH TIMING SCENARIO IN DL AFTER CHANGES APPLIED

Consequently, when the FH times are modified, these changes also have an impact on the UL, whose diagram with the changes produced is shown in the Figure 55. This implies that the control plane changes in the same way as in DL, and in the user plane the DU RW extends in time Δt time units.

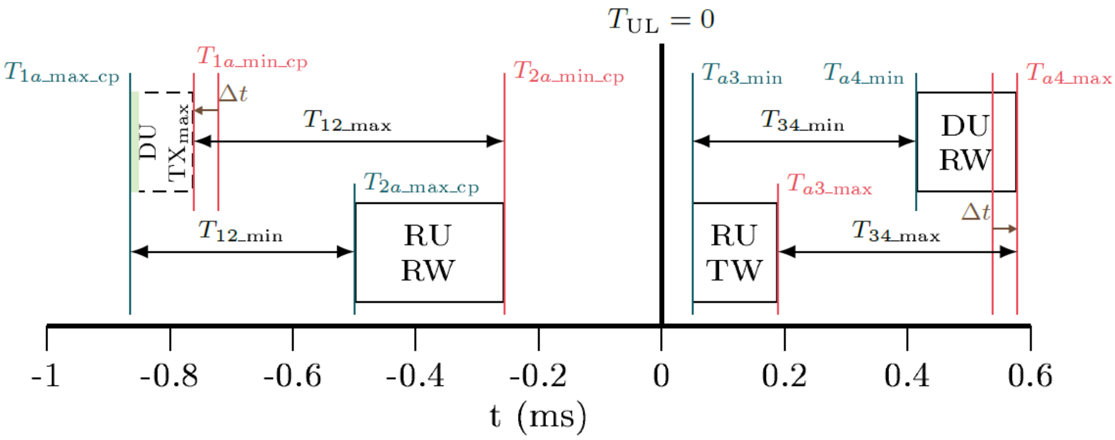


FIGURE 55. FH TIMING SCENARIO IN UL AFTER CHANGES

The implemented tool is capable of generating multiple types of results, both numerical and visual, which facilitate the interpretation and analysis of the evaluated scenario.

On the numerical side, the tool provides precise quantitative outputs such as calculated latency values, delay budgets, and other relevant metrics that characterize the performance and feasibility of the proposed FH configuration.

In parallel, the tool also produces a variety of visual outputs, including graphical representations of network topologies and timing diagrams with the initial and final scenarios. These visualizations are invaluable in offering an intuitive understanding of complex relationships between network parameters and in highlighting the effects of different centralization strategies on overall system performance.

By combining both numerical precision and graphical clarity, the tool enables us to comprehensively interpret the implications of their design decisions. This dual-output approach ensures that technical teams can both validate detailed performance metrics and communicate key insights effectively to broader audiences, supporting informed decision-making throughout the planning and optimization process.

Based on the exportation of results, and by running the program multiple times to analyze the variability of the outcomes according to the network conditions, it becomes possible to thoroughly examine the obtained scenarios and their distribution. For example, Figure 56 shows the FH timings for the O-DU located at HL4 and for the different degrees of centralization (see Table 6). It can be seen how *T12_min* remains constant, while *T12_max* increases to match the optimal value of the O-DU TW.

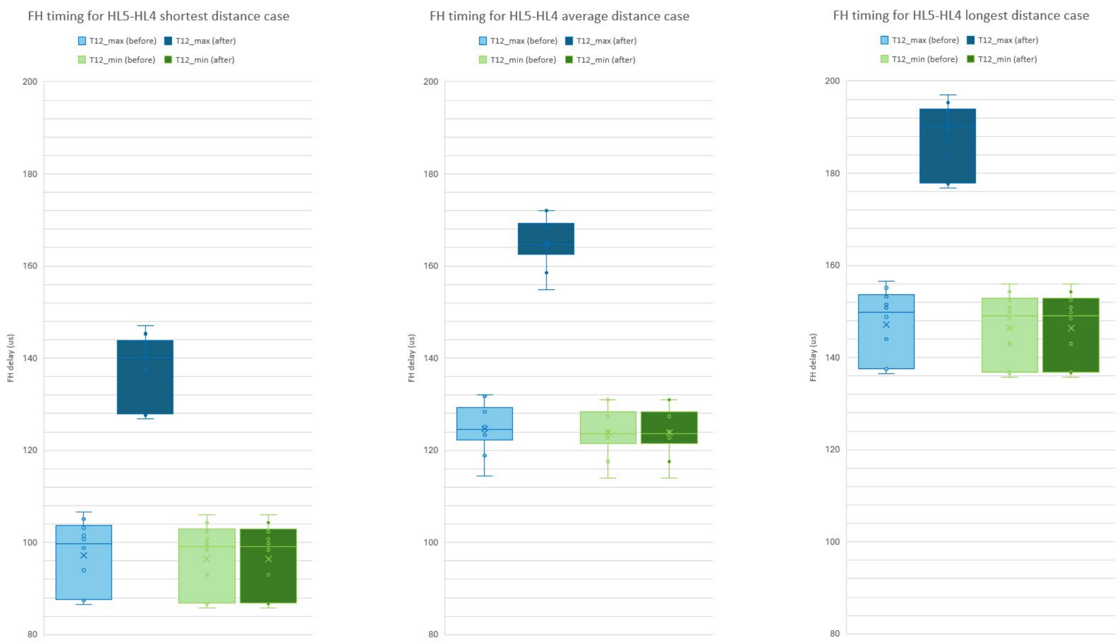


FIGURE 56. FH DELAYS FOR O-DU AT HL4

Likewise, Figure 57 shows the centralization distance achieved in each simulation, going from approximately 10 km to 25 km.

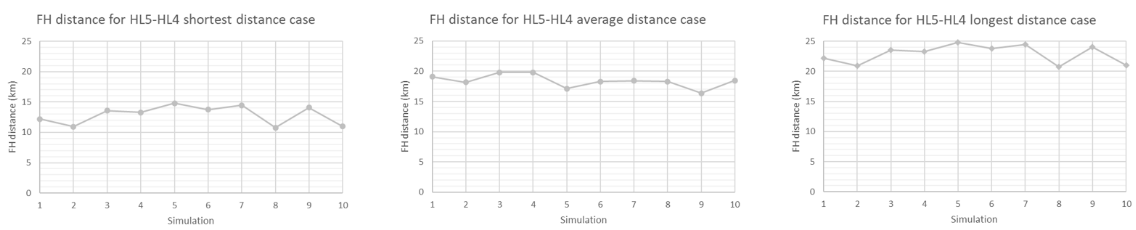


FIGURE 57. FH DISTANCES FOR O-DU AT HL4

Now, for the case in which the O-DU is further centralized and set at HL3, we have the following FH delays plotted in Figure 58, of course longer than in the previous case.

Financiado por la Unión Europea NextGenerationEU
Plan de Recuperación, Transformación y Resiliencia
UNICO I+D
GOBIERNO DE ESPAÑA
cttc Centre Tecnològic de Telecomunicacions de Catalunya

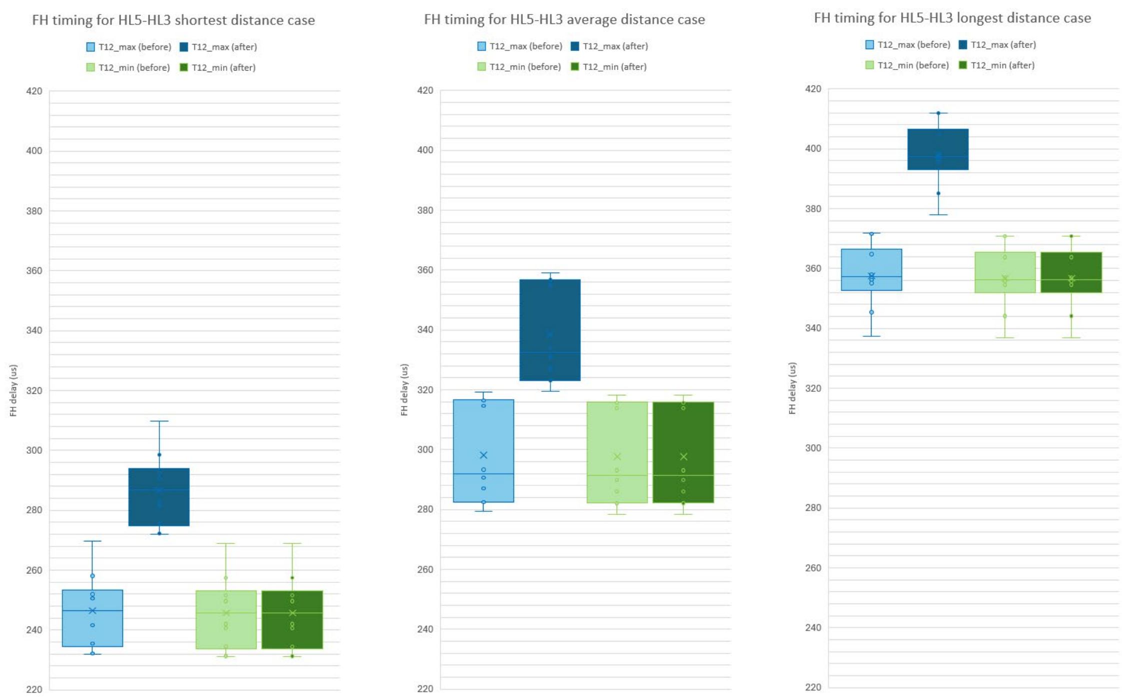


FIGURE 58. FH DELAYS FOR O-DU AT HL3

In turn, it is shown in Figure 59 the centralization distance corresponding with these delays, going from approximately 37 km to 65 km.

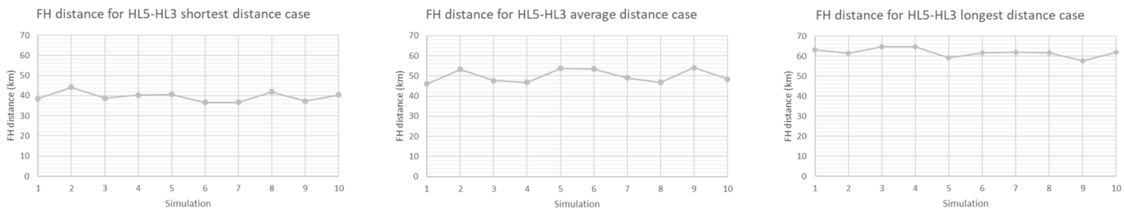


FIGURE 59. FH DISTANCES FOR O-DU AT HL3

In order to offer a clearer and more comprehensive understanding of the results, the following section presents a series of tables detailing the specific modifications implemented at each level of centralization and for each simulation. These tables present values that are not exactly the same that those tested in the testbed and presented at the end of this section, but they are extracted in the same conditions and are presented here intended to illustrate not only the extent of the changes but also the rationale and contextual application behind each adjustment, showing also the variability between simulations due to the inherent variability in network conditions and the specific centralization distance. For this purpose, ten simulations results are presented for each centralization degree.

Financiado por la Unión Europea NextGenerationEU
Plan de Recuperación, Transformación y Resiliencia
UNICO I+D
GOBIERNO DE ESPAÑA
cttc Centre Tecnològic de Telecomunicacions de Catalunya

Table 21 specifically highlights two key aspects for every scenario analysed: the centralization distance obtained both in the preliminary scenario with the inputs provided, and after applying the timing changes to optimize the scenario, and the corresponding study case applied, which outlines the context or methodology adopted in that particular instance. It can be seen that in all the cases, the optimization method applied is the study case 1 (see Figure 29), based on adjusting the O-DU TW. So, the applied changes consist in extending *T12_max* (and consequently *T1a_min*) to relax conditions in transport NW routers, thus reducing the O-DU TW to its optimum value, that is six symbol times (*Ts*), i.e. 200 µs. For this reason, the centralization distance does not change since in the preliminary scenario the good conditions in the network already allow us to reach the proposed distances.

TABLE 21. DETAILED RESULTS OF DISTANCE AND STUDY CASE APPLIED FOR EACH SIMULATION

	Simulation	Case study	Centralization distance (before)		Centralization distance (after)	
HL4 – Shortest distance (10-15 km)	1	1	12,19	km	12,20	km
	2	1	10,93	km	10,94	km
	3	1	13,55	km	13,56	km
	4	1	13,28	km	13,28	km
	5	1	14,80	km	14,80	km
	6	1	13,76	km	13,76	km
	7	1	14,45	km	14,46	km
	8	1	10,75	km	10,76	km
	9	1	14,07	km	14,08	km
	10	1	10,98	km	10,98	km
	Avg.		**12,88**	**km**	**12,88**	**km**
HL4 - Average distance (15-20 km)	**Simulation**	**Case study**	**Centralization distance (before)**		**Centralization distance (after)**	
	1	1	19,07	km	19,08	km

	2	1	18,16	km	18,16	km
	3	1	19,79	km	19,80	km
	4	1	19,79	km	19,80	km
	5	1	17,11	km	17,12	km
	6	1	18,28	km	18,28	km
	7	1	18,39	km	18,40	km
	8	1	18,28	km	18,28	km
	9	1	16,38	km	16,38	km
	10	1	18,47	km	18,48	km
	Avg.		**18,37**	**km**	**18,38**	**km**
HL4 - Longest distance (20-25 km)	**Simulation**	**Case study**	**Centralization distance (before)**		**Centralization distance (after)**	
	1	1	22,19	km	22,20	km
	2	1	20,93	km	20,94	km
	3	1	23,55	km	23,56	km
	4	1	23,28	km	23,28	km
	5	1	24,80	km	24,80	km
	6	1	23,76	km	23,76	km
	7	1	24,45	km	24,46	km
	8	1	20,75	km	20,76	km
	9	1	24,07	km	24,08	km
	10	1	20,98	km	20,98	km
	Avg.		**22,88**	**km**	**22,88**	**km**
HL3 – Shortest distance (35-45 km)	**Simulation**	**Case study**	**Centralization distance (before)**		**Centralization distance (after)**	

Financiado por la Unión Europea NextGenerationEU
Plan de Recuperación, Transformación y Resiliencia
UNICO I+D
Gobierno de España
cttc Centre Tecnològic de Telecomunicacions de Catalunya

	1	1	38,52	km	38,52	km
	2	1	44,17	km	44,18	km
	3	1	38,80	km	38,80	km
	4	1	40,31	km	40,32	km
	5	1	40,69	km	40,70	km
	6	1	36,62	km	36,62	km
	7	1	36,66	km	36,66	km
	8	1	41,89	km	41,90	km
	9	1	37,29	km	37,30	km
	10	1	40,38	km	40,38	km
	Avg.		**39,53**	**km**	**39,54**	**km**
HL3 – Average distance (45-55 km)	**Simulation**	**Case study**	**Centralization distance (before)**		**Centralization distance (after)**	
	1	1	46,07	km	46,08	km
	2	1	53,17	km	53,18	km
	3	1	47,60	km	47,60	km
	4	1	46,82	km	46,82	km
	5	1	53,69	km	53,70	km
	6	1	53,53	km	53,54	km
	7	1	49,02	km	49,02	km
	8	1	46,84	km	46,84	km
	9	1	54,03	km	54,04	km
	10	1	48,38	km	48,38	km
	Avg.		**49,92**	**km**	**49,92**	**km**
HL3 - Longest distance (55-65 km)	**Simulation**	**Case study**				

			Centralization distance (before)		Centralization distance (after)	
	1	1	63,15	km	63,16	km
	2	1	61,32	km	61,32	km
	3	1	64,58	km	64,58	km
	4	1	64,57	km	64,58	km
	5	1	59,22	km	59,22	km
	6	1	61,56	km	61,56	km
	7	1	61,79	km	61,80	km
	8	1	61,55	km	61,56	km
	9	1	57,77	km	57,78	km
	10	1	61,95	km	61,96	km
	Avg.		**61,75**	**km**	**61,75**	**km**

Regarding the O-DU timing, Table 22 presents the TW boundaries again before and after applying the changes thanks to the MATLAB tool analysis. It can be seen that while *T1a_max* does not change, *T1a_min* is extended (giving more budget for *T12_max*) so the difference between both values is 200 µs, that is, the O-DU TW optimum size. This allows to have more relaxed conditions in the network, what translates to have more margin for possible queueing delays that may appear in the worst conditions. These changes are consequently applied also in UL for the O-DU RW.

TABLE 22. DETAILED RESULTS OF O-DU TW TIMING FOR EACH SIMULATION

	Simulation	E2E delay															
		T1a_max						T1a_min						Difference			
		Before		After		Correction		Before		After		Correction		Before		After	
HL4 - Shortest distance (10-15 km)	1	468	us	468	us	0	us	228	us	268	us	40	us	240	us	200	us
	2	461,7	us	461,7	us	0	us	221,5	us	261,7	us	40,2	us	240,2	us	200	us
	3	474,8	us	474,8	us	0	us	234,7	us	274,8	us	40,1	us	240,1	us	200	us
	4	473,4	us	473,4	us	0	us	232,8	us	273,4	us	40,6	us	240,6	us	200	us
	5	481	us	481	us	0	us	240,6	us	281	us	40,4	us	240,4	us	200	us
	6	475,8	us	475,8	us	0	us	235,5	us	275,8	us	40,3	us	240,3	us	200	us
	7	479,3	us	479,3	us	0	us	239,1	us	279,3	us	40,2	us	240,2	us	200	us

Financiado por la Unión Europea NextGenerationEU
Plan de Recuperación, Transformación y Resiliencia
UNICO I+D
GOBIERNO DE ESPAÑA
cttc Centre Tecnològic de Telecomunicacions de Catalunya

	8	460,8	us	460,8	us	0	us	220,5	us	260,8	us	40,3	us	240,3	us	200	us
	9	477,4	us	477,4	us	0	us	237,2	us	277,4	us	40,2	us	240,2	us	200	us
	10	461,9	us	461,9	us	0	us	221,6	us	261,9	us	40,3	us	240,3	us	200	us
	Avg.	**471,41**	**us**	**471,41**	**us**	**0**	**us**	**231,15**	**us**	**271,41**	**us**	**40,26**	**us**	**240,26**	**us**	**200**	**us**
HL4 - Average distance (15-20 km)	**Simulation**	**E2E delay**															
		T1a_max						**T1a_min**						**Difference**			
		Before		**After**		**Correction**		**Before**		**After**		**Correction**		**Before**		**After**	
	1	502,4	us	502,4	us	0	us	262,4	us	302,4	us	40	us	240	us	200	us
	2	497,8	us	497,8	us	0	us	257,3	us	297,8	us	40,5	us	240,5	us	200	us
	3	506	us	506	us	0	us	266	us	306	us	40	us	240	us	200	us
	4	506	us	506	us	0	us	265,7	us	306	us	40,3	us	240,3	us	200	us
	5	492,6	us	492,6	us	0	us	252,9	us	292,6	us	39,7	us	239,7	us	200	us
	6	498,4	us	498,4	us	0	us	258,3	us	298,4	us	40,1	us	240,1	us	200	us
	7	499	us	499	us	0	us	258,9	us	299	us	40,1	us	240,1	us	200	us
	8	498,4	us	498,4	us	0	us	257,9	us	298,4	us	40,5	us	240,5	us	200	us
	9	488,9	us	488,9	us	0	us	248,4	us	288,9	us	40,5	us	240,5	us	200	us
	10	499,4	us	499,4	us	0	us	259,1	us	299,4	us	40,3	us	240,3	us	200	us
	Avg.	**498,89**	**us**	**498,89**	**us**	**0**	**us**	**258,69**	**us**	**298,89**	**us**	**40,2**	**us**	**240,2**	**us**	**200**	**us**
HL4 - Longest distance (20-25 km)	**Simulation**	**E2E delay**															
		T1a_max						**T1a_min**						**Difference**			
		Before		**After**		**Correction**		**Before**		**After**		**Correction**		**Before**		**After**	
	1	518	us	518	us	0	us	278	us	318	us	40	us	240	us	200	us
	2	511,7	us	511,7	us	0	us	271,5	us	311,7	us	40,2	us	240,2	us	200	us
	3	524,8	us	524,8	us	0	us	284,7	us	324,8	us	40,1	us	240,1	us	200	us
	4	523,4	us	523,4	us	0	us	282,8	us	323,4	us	40,6	us	240,6	us	200	us
	5	531	us	531	us	0	us	290,6	us	331	us	40,4	us	240,4	us	200	us
	6	525,8	us	525,8	us	0	us	285,5	us	325,8	us	40,3	us	240,3	us	200	us
	7	529,3	us	529,3	us	0	us	289,1	us	329,3	us	40,2	us	240,2	us	200	us
	8	510,8	us	510,8	us	0	us	270,5	us	310,8	us	40,3	us	240,3	us	200	us
	9	527,4	us	527,4	us	0	us	287,2	us	327,4	us	40,2	us	240,2	us	200	us
	10	511,9	us	511,9	us	0	us	271,6	us	311,9	us	40,3	us	240,3	us	200	us
	Avg.	**521,41**	**us**	**521,41**	**us**	**0**	**us**	**281,15**	**us**	**321,41**	**us**	**40,26**	**us**	**240,26**	**us**	**200**	**us**
HL3 - Shortest distance (35-45 km)	**Simulation**	**E2E delay**															
		T1a_max						**T1a_min**						**Difference**			
		Before		**After**		**Correction**		**Before**		**After**		**Correction**		**Before**		**After**	
	1	615,6	us	615,6	us	0	us	375,6	us	415,6	us	40	us	240	us	200	us
	2	643,9	us	643,9	us	0	us	403,8	us	443,9	us	40,1	us	240,1	us	200	us
	3	617	us	617	us	0	us	376,3	us	417	us	40,7	us	240,7	us	200	us

Financiado por la Unión Europea NextGenerationEU
Plan de Recuperación, Transformación y Resiliencia
UNICO I+D
Gobierno de España
cttc Centre Tecnològic de Telecomunicacions de Catalunya

	4	624,6	us	624,6	us	0	us	384,5	us	424,6	us	40,1	us	240,1	us	200	us
	5	626,5	us	626,5	us	0	us	385,9	us	426,5	us	40,6	us	240,6	us	200	us
	6	606,1	us	606,1	us	0	us	365,9	us	406,1	us	40,2	us	240,2	us	200	us
	7	606,3	us	606,3	us	0	us	366,1	us	406,3	us	40,2	us	240,2	us	200	us
	8	632,5	us	632,5	us	0	us	392,1	us	432,5	us	40,4	us	240,4	us	200	us
	9	609,5	us	609,5	us	0	us	369,4	us	409,5	us	40,1	us	240,1	us	200	us
	10	624,9	us	624,9	us	0	us	384,7	us	424,9	us	40,2	us	240,2	us	200	us
	Avg.	**620,69**	**us**	**620,69**	**us**	**0**	**us**	**380,43**	**us**	**420,69**	**us**	**40,26**	**us**	**240,26**	**us**	**200**	**us**
HL3 - Average distance (45-55 km)	**Simulation**	**E2E delay**															
		T1a_max						**T1a_min**						**Difference**			
		Before		**After**		**Correction**		**Before**		**After**		**Correction**		**Before**		**After**	
	1	653,4	us	653,4	us	0	us	413,3	us	453,4	us	40,1	us	240,1	us	200	us
	2	688,9	us	688,9	us	0	us	448,6	us	488,9	us	40,3	us	240,3	us	200	us
	3	661	us	661	us	0	us	421,1	us	461	us	39,9	us	239,9	us	200	us
	4	657,1	us	657,1	us	0	us	416,4	us	457,1	us	40,7	us	240,7	us	200	us
	5	691,5	us	691,5	us	0	us	451,1	us	491,5	us	40,4	us	240,4	us	200	us
	6	690,7	us	690,7	us	0	us	450,4	us	490,7	us	40,3	us	240,3	us	200	us
	7	668,1	us	668,1	us	0	us	427,3	us	468,1	us	40,8	us	240,8	us	200	us
	8	657,2	us	657,2	us	0	us	416,6	us	457,2	us	40,6	us	240,6	us	200	us
	9	693,2	us	693,2	us	0	us	453,2	us	493,2	us	40	us	240	us	200	us
	10	664,9	us	664,9	us	0	us	424,6	us	464,9	us	40,3	us	240,3	us	200	us
	Avg.	**672,6**	**us**	**672,6**	**us**	**0**	**us**	**432,26**	**us**	**472,6**	**us**	**40,34**	**us**	**240,34**	**us**	**200**	**us**
HL3 - Longest distance (55-65 km)	**Simulation**	**E2E delay**															
		T1a_max						**T1a_min**						**Difference**			
		Before		**After**		**Correction**		**Before**		**After**		**Correction**		**Before**		**After**	
	1	738,8	us	738,8	us	0	us	498,8	us	538,8	us	40	us	240	us	200	us
	2	729,6	us	729,6	us	0	us	489,1	us	529,6	us	40,5	us	240,5	us	200	us
	3	745,9	us	745,9	us	0	us	505,9	us	545,9	us	40	us	240	us	200	us
	4	745,9	us	745,9	us	0	us	505,6	us	545,9	us	40,3	us	240,3	us	200	us
	5	719,1	us	719,1	us	0	us	479,4	us	519,1	us	39,7	us	239,7	us	200	us
	6	730,8	us	730,8	us	0	us	490,7	us	530,8	us	40,1	us	240,1	us	200	us
	7	732	us	732	us	0	us	491,9	us	532	us	40,1	us	240,1	us	200	us
	8	730,8	us	730,8	us	0	us	490,3	us	530,8	us	40,5	us	240,5	us	200	us
	9	711,9	us	711,9	us	0	us	471,4	us	511,9	us	40,5	us	240,5	us	200	us
	10	732,8	us	732,8	us	0	us	492,5	us	532,8	us	40,3	us	240,3	us	200	us
	Avg.	**731,76**	**us**	**731,76**	**us**	**0**	**us**	**491,56**	**us**	**531,76**	**us**	**40,2**	**us**	**240,2**	**us**	**200**	**us**

Likewise, in the case of FH timing *T12_min* does not change and *T12_max* is extended enough to reach O-DU TW optimum size, as can be seen in Table 23.

TABLE 23. DETAILED RESULTS OF FH TIMING FOR EACH SIMULATION

	Simulation	FH delay															
		T12_max						T12_min						Difference			
		Before		After		Correction		Before		After		Correction		Before		After	
HL4 - Shortest distance (10-15 km)	1	94	us	134	us	40	us	93	us	93	us	0	us	1	us	41	us
	2	87,5	us	127,7	us	40,2	us	86,7	us	86,7	us	0	us	0,8	us	41	us
	3	100,7	us	140,8	us	40,1	us	99,8	us	99,8	us	0	us	0,9	us	41	us
	4	98,8	us	139,4	us	40,6	us	98,4	us	98,4	us	0	us	0,4	us	41	us
	5	106,6	us	147	us	40,4	us	106	us	106	us	0	us	0,6	us	41	us
	6	101,5	us	141,8	us	40,3	us	100,8	us	100,8	us	0	us	0,7	us	41	us
	7	105,1	us	145,3	us	40,2	us	104,3	us	104,3	us	0	us	0,8	us	41	us
	8	86,5	us	126,8	us	40,3	us	85,8	us	85,8	us	0	us	0,7	us	41	us
	9	103,2	us	143,4	us	40,2	us	102,4	us	102,4	us	0	us	0,8	us	41	us
	10	87,6	us	127,9	us	40,3	us	86,9	us	86,9	us	0	us	0,7	us	41	us
	Avg.	97,15	us	137,41	us	40,26	us	96,41	us	96,41	us	0	us	0,74	us	41	us
	Simulation	**FH delay**															
		T12_max						**T12_min**						**Difference**			
		Before		**After**		**Correction**		**Before**		**After**		**Correction**		**Before**		**After**	
HL4 - Average distance (15-20 km)	1	128,4	us	168,4	us	40	us	127,4	us	127,4	us	0	us	1	us	41	us
	2	123,3	us	163,8	us	40,5	us	122,8	us	122,8	us	0	us	0,5	us	41	us
	3	132	us	172	us	40	us	131	us	131	us	0	us	1	us	41	us
	4	131,7	us	172	us	40,3	us	131	us	131	us	0	us	0,7	us	41	us
	5	118,9	us	158,6	us	39,7	us	117,6	us	117,6	us	0	us	1,3	us	41	us
	6	124,3	us	164,4	us	40,1	us	123,4	us	123,4	us	0	us	0,9	us	41	us
	7	124,9	us	165	us	40,1	us	124	us	124	us	0	us	0,9	us	41	us
	8	123,9	us	164,4	us	40,5	us	123,4	us	123,4	us	0	us	0,5	us	41	us
	9	114,4	us	154,9	us	40,5	us	113,9	us	113,9	us	0	us	0,5	us	41	us
	10	125,1	us	165,4	us	40,3	us	124,4	us	124,4	us	0	us	0,7	us	41	us
	Avg.	124,69	us	164,89	us	40,2	us	123,89	us	123,89	us	0	us	0,8	us	41	us
	Simulation	**FH delay**															
		T12_max						**T12_min**						**Difference**			
		Before		**After**		**Correction**		**Before**		**After**		**Correction**		**Before**		**After**	
HL4 - Longest distance (20-25 km)	1	144	us	184	us	40	us	143	us	143	us	0	us	1	us	41	us
	2	137,5	us	177,7	us	40,2	us	136,7	us	136,7	us	0	us	0,8	us	41	us
	3	150,7	us	190,8	us	40,1	us	149,8	us	149,8	us	0	us	0,9	us	41	us
	4	148,8	us	189,4	us	40,6	us	148,4	us	148,4	us	0	us	0,4	us	41	us

Financiado por la Unión Europea NextGenerationEU
Plan de Recuperación, Transformación y Resiliencia
UNICO I+D
GOBIERNO DE ESPAÑA
cttc Centre Tecnològic de Telecomunicacions de Catalunya

	5	156,6	us	197	us	40,4	us	156	us	156	us	0	us	0,6	us	41	us
	6	151,5	us	191,8	us	40,3	us	150,8	us	150,8	us	0	us	0,7	us	41	us
	7	155,1	us	195,3	us	40,2	us	154,3	us	154,3	us	0	us	0,8	us	41	us
	8	136,5	us	176,8	us	40,3	us	135,8	us	135,8	us	0	us	0,7	us	41	us
	9	153,2	us	193,4	us	40,2	us	152,4	us	152,4	us	0	us	0,8	us	41	us
	10	137,6	us	177,9	us	40,3	us	136,9	us	136,9	us	0	us	0,7	us	41	us
	Avg.	147,15	us	187,41	us	40,26	us	146,41	us	146,41	us	0	us	0,74	us	41	us
HL3 - Shortest distance (35-45 km)	**Simulation**	**FH delay**															
		T12_max						**T12_min**						**Difference**			
		Before		**After**		**Correction**		**Before**		**After**		**Correction**		**Before**		**After**	
	1	241,6	us	281,6	us	40	us	240,6	us	240,6	us	0	us	1	us	41	us
	2	269,8	us	309,9	us	40,1	us	268,9	us	268,9	us	0	us	0,9	us	41	us
	3	242,3	us	283	us	40,7	us	242	us	242	us	0	us	0,3	us	41	us
	4	250,5	us	290,6	us	40,1	us	249,6	us	249,6	us	0	us	0,9	us	41	us
	5	251,9	us	292,5	us	40,6	us	251,5	us	251,5	us	0	us	0,4	us	41	us
	6	231,9	us	272,1	us	40,2	us	231,1	us	231,1	us	0	us	0,8	us	41	us
	7	232,1	us	272,3	us	40,2	us	231,3	us	231,3	us	0	us	0,8	us	41	us
	8	258,1	us	298,5	us	40,4	us	257,5	us	257,5	us	0	us	0,6	us	41	us
	9	235,4	us	275,5	us	40,1	us	234,5	us	234,5	us	0	us	0,9	us	41	us
	10	250,7	us	290,9	us	40,2	us	249,9	us	249,9	us	0	us	0,8	us	41	us
	Avg.	246,43	us	286,69	us	40,26	us	245,69	us	245,69	us	0	us	0,74	us	41	us
HL3 - Average distance (45-55 km)	**Simulation**	**FH delay**															
		T12_max						**T12_min**						**Difference**			
		Before		**After**		**Correction**		**Before**		**After**		**Correction**		**Before**		**After**	
	1	279,3	us	319,4	us	40,1	us	278,4	us	278,4	us	0	us	0,9	us	41	us
	2	314,6	us	354,9	us	40,3	us	313,9	us	313,9	us	0	us	0,7	us	41	us
	3	287,1	us	327	us	39,9	us	286	us	286	us	0	us	1,1	us	41	us
	4	282,4	us	323,1	us	40,7	us	282,1	us	282,1	us	0	us	0,3	us	41	us
	5	317,1	us	357,5	us	40,4	us	316,5	us	316,5	us	0	us	0,6	us	41	us
	6	316,4	us	356,7	us	40,3	us	315,7	us	315,7	us	0	us	0,7	us	41	us
	7	293,3	us	334,1	us	40,8	us	293,1	us	293,1	us	0	us	0,2	us	41	us
	8	282,6	us	323,2	us	40,6	us	282,2	us	282,2	us	0	us	0,4	us	41	us
	9	319,2	us	359,2	us	40	us	318,2	us	318,2	us	0	us	1	us	41	us
	10	290,6	us	330,9	us	40,3	us	289,9	us	289,9	us	0	us	0,7	us	41	us
	Avg.	298,26	us	338,6	us	40,34	us	297,6	us	297,6	us	0	us	0,66	us	41	us
HL3 - Longest distance (55-65 km)	**Simulation**	**FH delay**															
		T12_max						**T12_min**						**Difference**			
		Before		**After**		**Correction**		**Before**		**After**		**Correction**		**Before**		**After**	

Financiado por la Unión Europea NextGenerationEU
Plan de Recuperación, Transformación y Resiliencia
UNICO I+D
GOBIERNO DE ESPAÑA
cttc Centre Tecnològic de Telecomunicacions de Catalunya

	1	364,8	us	404,8	us	40	us	363,8	us	363,8	us	0	us	1	us	41	us
	2	355,1	us	395,6	us	40,5	us	354,6	us	354,6	us	0	us	0,5	us	41	us
	3	371,9	us	411,9	us	40	us	370,9	us	370,9	us	0	us	1	us	41	us
	4	371,6	us	411,9	us	40,3	us	370,9	us	370,9	us	0	us	0,7	us	41	us
	5	345,4	us	385,1	us	39,7	us	344,1	us	344,1	us	0	us	1,3	us	41	us
	6	356,7	us	396,8	us	40,1	us	355,8	us	355,8	us	0	us	0,9	us	41	us
	7	357,9	us	398	us	40,1	us	357	us	357	us	0	us	0,9	us	41	us
	8	356,3	us	396,8	us	40,5	us	355,8	us	355,8	us	0	us	0,5	us	41	us
	9	337,4	us	377,9	us	40,5	us	336,9	us	336,9	us	0	us	0,5	us	41	us
	10	358,5	us	398,8	us	40,3	us	357,8	us	357,8	us	0	us	0,7	us	41	us
	Avg.	**357,56**	**us**	**397,76**	**us**	**40,2**	**us**	**356,76**	**us**	**356,76**	**us**	**0**	**us**	**0,8**	**us**	**41**	**us**

With all this information at hand, and once we had determined the input values required by the MATLAB tool, namely, the O-RU processing timing and the FH conditions, we were able to generate the centralization scenarios for each degree of centralization shown in Table 24. These scenarios allowed us to extract FH latencies and O-DU processing times, which we then applied within the testbed environment. This enabled us to empirically validate the results obtained through simulation, ensuring that the theoretical findings aligned with practical performance under real or emulated network conditions.

TABLE 24. CENTRALIZATION SCENARIOS

Centralization degree	Case study applied	Centralization distance [km]	O-DU timing				FH timing			
	Before [μs] After [μs]	Before [μs]	After [μs]	Before [μs]	After [μs]	Before [μs]	After [μs]			
			T1a_max		T1a_min		T12_max		T12_min	
HL4-shortest	1	12,8	471,4	471,4	231,2	271,4	97,2	137,4	96,4	96,4
HL4-average	1	18,38	498,9	498,9	258,7	298,9	124,7	164,9	123,9	123,9
HL4-longest	1	22,88	521,4	521,4	281,2	321,4	147,2	187,4	146,4	146,4
HL3-shortest	1	39,54	620,7	620,7	380,4	420,7	246,4	286,7	245,7	245,7
HL3-average	1	49,92	672,6	672,6	432,3	472,6	298,3	338,6	297,6	297,6
HL3-longest	1	61,96	731,8	731,8	491,6	531,8	357,6	397,8	356,8	356,8

Financiado por la Unión Europea NextGenerationEU
Plan de Recuperación, Transformación y Resiliencia
UNICO I+D
GOBIERNO DE ESPAÑA
cttc Centre Tecnològic de Telecomunicacions de Catalunya

3.1.1.2.1.5 O-DU optimization

As it can be observed, in the scenario with the highest degree of centralization, we were able to achieve a distance of nearly 65 km between the O-DU and the O-RU. Along with the corresponding FH delay values derived for this configuration, we utilized the srsRAN software to validate system behavior under these conditions.

Through these tests, it was verified that, for this level of centralization, it is not necessary to modify or fine-tune the O-DU protocols to accommodate the increased delays introduced by the extended transport distance. This potential adjustment of O-DU protocols remains an open area for future research. As such, the only modification required in our current study was the adjustment of the window parameters on the O-DU side to align with the extended fronthaul conditions. More information about this can be found in Section 2.7 (SMART-K2.4).

Specifically, srsRAN Project is the open-source RAN solution developed by SRS (one of the project partners), which is fully compliant with the 3GPP and O-RAN specifications. Given the requirements of the PoC, we opted for the monolithic version of the software (CU and DU running in the same executable), since it is easier to configure for high bandwidth cells with strict latency requirements. All the timing settings discussed above are easily configurable in the "ru_ofh" section of the srsRAN Project yaml configuration file:

```
ru_ofh:
  t1a_max_cp_dl    # T1a maximum value for downlink Control-Plane. Supported: [0 – 1960]
  t1a_min_cp_dl    # T1a minimum value for downlink Control-Plane. Supported: [0 - 1960]
  t1a_max_cp_ul    # T1a maximum value for uplink Control-Plane. Supported: [0 - 1960]
  t1a_min_cp_ul    # T1a minimum value for uplink Control-Plane. Supported: [0 - 1960]
  t1a_max_up       # T1a maximum value for User-Plane. Supported: [0 - 1960]
  t1a_min_up       # T1a minimum value for User-Plane. Supported: [0 - 1960]
  ta4_max          # Ta4 maximum value for User-Plane. Supported: [0 - 1960]
  ta4_min          # Ta4 minimum value for User-Plane. Supported: [0 - 1960]
```

where all values are interpreted as a time interval in microseconds. It is worth remarking that these values are later converted by the software into an integer number of OFDM symbols (approximately 33 µs with a subcarrier spacing of 30 kHz). Note that, when modifying these time windows, the operator may also need to fine tune the following parameter in the “expert_phy” section:

```
expert_phy:
  max_proc_delay   # Maximum allowed DL processing delay in slots. Supported: [1 - 30]
```

Roughly speaking, this delay (expressed in units of slot duration, that is 0.5 ms with a subcarrier spacing of 30 kHz) encompasses all the steps required by a DL slot from its scheduling until its transmission at the antenna port at the RU, thus including the delay introduced by the O-FH protocol when transferring data from the DU to the RU.

The timing results obtained analytically for the FH have been validated in an Open RAN end-to-end (E2E) testbed set up at the 6G-SANDBOX premises. This validation process was carried out to cross-check the simulation outcomes and to extract practical, real-world results within a controlled laboratory environment.

By replicating the centralization scenarios and fronthaul conditions in the testbed, we were able to empirically verify that the analytical models and software tool predictions closely aligned with the observed system behavior. The testbed allowed us to assess latency metrics under realistic conditions across an Open RAN disaggregated architecture.

3.1.1.2.2 QoS scheduling schemes

We have proposed various MAC schedulers to address QoS requirements of delay-critical XR traffics. In particular:

- **QoS scheduler**: considers the resource type (GBR, non-GBR, dc-GBR), the priority level of the flow, the packet delay budget (PDB), the head-of-line (HOL) delay and the PF metric.
- **DPP scheduler**: considers the guaranteed flow bit rate (GFBR) requirement, the resource utilization, and stability of the queues, using Lyapunov control theory.
- **PDU set-aware scheduler**: extends DPP scheduler, using Lyapunov control theory, with PDU set information, like the PDU set delay budget and PDU set size.

They have been implemented in ns-3 5G-LENA and evaluated through system-level simulations using mixed AR/VR/CG scenarios. This work is summarized in SMART K2.5 and has contributed to publications [14],[1],[18], [5].

3.1.1.2.3 FH control methods

We have developed various FH control methods to control the downlink data bulks that go through a limited-capacity fronthaul link. In particular:

- Dropping
- Postponing
- OptimizeRbs
- OptimizeMcs

- AdjustRbs

They have been implemented in ns-3 5G-LENA and evaluated through system-level simulations using mixed AR/VR/CG scenarios. This work is summarized in SMART K1.2 and has contributed to publications [9], [11].

3.1.1.2.4 Flexible Functional Splits

We have derived the capacity and latency requirements that transport networks (fronthaul and midhaul) will need to meet in future 6G deployments, for different functional split options. We have analyzed the impact of different functional split options using the 5G-LENA simulator. This represents the first step toward implementing dynamic Flexible Functional Splits in 5G-LENA. This work is summarized in SMART K2.3 and has contributed to publication [19].

3.1.1.3 Experimental Implementation and Assessment

3.1.1.3.1 Testbed Description

A structured and modular testbed is designed to validate the theoretical analysis in practical scenarios (see Figure 60). This testbed emulates a scenario in which a O-DU/O-CU is connected to an O-RU through an O-FH that introduces a significant delay. The O-FH delay is introduced by a Keysight instrument, Network Emulator 3 (NE3), which has the ability to introduce a programmable delay to all packets that travers it in either direction. With this, we emulate the operation of (a part of) a system in which O-DU/O-CUs are centralized at a significant distance from the O-RUs, as illustrated in Figure 61.

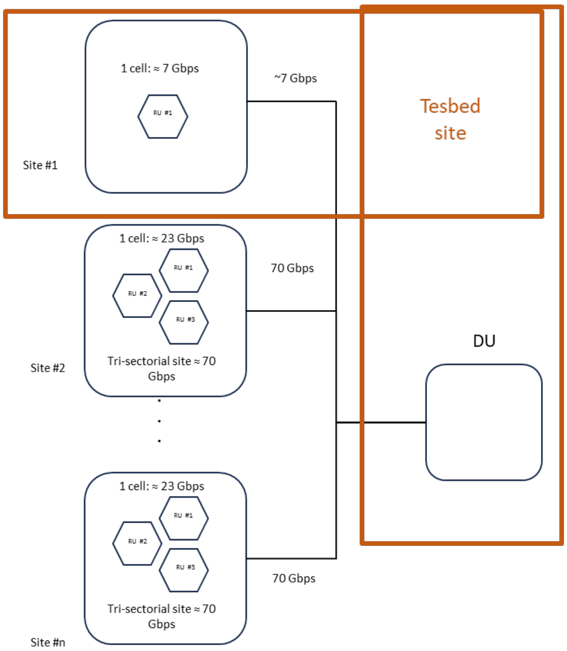


FIGURE 60. CONSIDERED SYSTEM MODEL, WITH THE PART EMULATED BY THE TESTBED HIGHLIGHTED IN THE ORANGE FRAME

3.1.1.3.1.1 Overall testbed

The conceptual architecture and practical implementation of this testbed are shown in Figure 61. It consists of seven components, together forming an end-to-end cellular network emulation environment with automated and centralized control. We explain each component and the corresponding functionalities below in more detail.

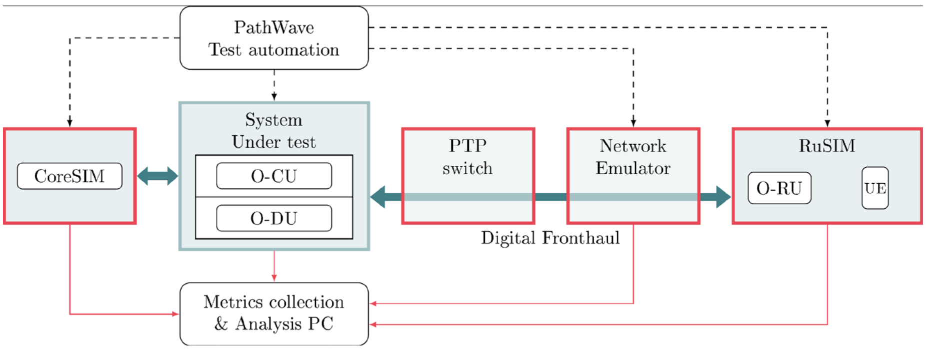


FIGURE 61. DIAGRAM OF THE TESTBED COMPONENTS

3.1.1.3.1.2 RuSIM

RuSIM is a Keysight emulator of O-RUs and UEs. It can emulate multiple O-RUs with configurable cell settings while creating IP traffic and mimicking thousands of devices running real voice and data connections. This makes RuSIM a powerful tool for testing the complete protocol stack for both 4G and 5G (NSA and SA) through the eCPRI interface, and particularly suited to test the performance of the O-FH interface as is the focus in this PoC. The configuration and control of RuSIM is governed by AirMosaic, a Keysight proprietary software that is also used to supervise the generation of user traffic.

The server running RuSIM has the following hardware specifications:

- Manufacturer: Supermicro
- CPU
    - Model: Intel Xeon Gold 6238R CPU @ 2.20GHz
    - Sockets: 2
    - Cores per socket: 28
    - Threads per core: 1
    - CPU max frequency: 4.0 GHz
    - CPU min frequency: 1.0 GHz
- RAM
    - DDR version: DDR4
    - Frequency: 3200 MHz
    - Memory per module: 16 GiB
    - Number of modules: 8
    - Total memory: 128 GiB
- O-FH NIC
    - Type: fiber
    - Link mode: 10000baseCR/Full
    - Speed: 10 Gbps

3.1.1.3.1.3 NE3

To emulate the network impairments in the fronthaul link, we utilize Keysight Network Emulator 3 (NE3). This instrument enables users to accurately introduce the real conditions that occur over live production networks, such as packet drops, delays, reordering, duplications, etc. In this particular testbed, NE3 is used to emulate the delays that will be present in the O-FH in the case of centralization of O-DUs, due to the large physical length of the line and possible switches and routing devices in its path.

Financiado por la Unión Europea NextGenerationEU
Plan de Recuperación, Transformación y Resiliencia
UNICO I+D
GOBIERNO DE ESPAÑA
cttc Centre Tecnològic de Telecomunicacions de Catalunya

3.1.1.3.1.4 PTP Switch

Given features such as increased radio bandwidth or carrier aggregation, where each aggregated component carrier needs to be accurately time-aligned, one rising design challenge in 5G and 6G networks is achieving precise timing. This becomes even more important with the introduction of functional splits in the O-RAN architecture, since equipment from different vendors may be placed separately at different sites, making it impossible to share the same local clock across equipment. Therefore, tight time synchronization between different network components is necessary in 5G O-RAN deployments. For instance, it is essential to synchronize O-DU and O-RU in Split 7.2. To achieve this, O-RAN Alliance defines various techniques, as specified in [29]. In our testbed, we implement PTP (IEEE 1588 Precision Time Protocol) synchronization between O-RU and O-DU via the S-plane based on the LLS-C3 configuration.

We utilize a timing-aware O-RAN switch, Falcon-RX [30], as the primary PTP source. To generate and transmit the PTP synchronization signal to the secondary nodes, it connects to an external GPS antenna and receives the GPS signal from multiple satellites. Then, the lower layers emulation server (RuSIM, functioning as O-RU) and the system under test (functioning as O-DU) connect to the switch via fibers for both S-plane and C-/U-plane traffic. Note that for the lower layers emulation server, the S-plane and C-/U-plane share the same interface, but for the system under test, as recommended by the srsRAN setup, the S-plane and C-/U-plane are separated into two independent interfaces, connected by different cables.

On the RuSIM side, ptp4l runs on the server, using the received PTP signal to synchronize its network interface card (NIC) clock to the primary PTP source. Then, timing-related values such as system frame number (SFN) are calculated based on the synchronized NIC clock. On the other hand, since the gNB utilizes the system clock on the hosting server, we execute ptp4l and phc2sys on the test server, which use the PTP signal to synchronize the server's system clock to that of the primary clock.

3.1.1.3.1.5 srsRAN gNB

The O-RU emulated by RuSIM is controlled by the srsRAN gNB solution from SRS. The srsRAN Project is the software-based complete RAN (O-CU and O-DU) solution from SRS. It is compliant with 3GPP Release 17 and O-RAN Alliance specifications. The software, distributed as open source, is entirely written by SRS with minimal external dependencies, it includes the full L1/2/3 stack and is portable across processor architectures. The srsRAN features that are most relevant in the context of the PoC are:

- Support for all FR1 bands, both TDD and FDD, and for all bandwidths up to 100 MHz; 4x4 downlink MIMO;
- Support for Split 7.2 enabled by the in-house O-FH library;

Financiado por la Unión Europea NextGenerationEU
Plan de Recuperación, Transformación y Resiliencia
UNICO I+D
GOBIERNO DE ESPAÑA
cttc Centre Tecnològic de Telecomunicacions de Catalunya

- AVX512-accelerated kernels for LDPC encoding/decoding and O-FH IQ samples compression/decompression;
- Exhaustive configuration via yaml files.

More details about srsRAN, and especially on the advances in the O-FH library driven by this PoC, are provided as part of the key concept SMART-K2.1. In this experimental setup, the SRS software is deployed in a server with the following characteristics:

- Manufacturer: Supermicro
- CPU
  - Model: Intel Xeon Platinum 8260 CPU @ 2.40GHz
  - Sockets: 2
  - Cores per socket: 24
  - Threads per core: 2
  - CPU max frequency: 3.9 GHz
  - CPU min frequency: 1.0 GHz
- RAM
  - DDR version: DDR4
  - Frequency: 2400 MHz
  - Memory per module: 64 GiB
  - Number of modules: 8
  - Total memory: 512 GiB
- O-FH NIC
  - Type: fiber
  - Link mode: 10000baseCR/Full
  - Speed: 10 Gbps
- Back-haul NIC
  - Type: fiber
  - Link mode: 10000baseCR/Full
  - Speed: 10 Gbps

In order to facilitate that the software runs in real time and no processing delays impair the operation of the O-DU/O-CU, Data Plane Development Kit (DPDK) is enabled at the server. DPDK [31]bypasses the kernel networking stack and directly accesses the NIC. Thus, it can reduce the overhead of packet processing. Given that the gNB is supposed to process a large number of packets in real-time, packet processing can significantly affect the overall performance. Based on this observation, to achieve high throughput and stable performance when running highly demanding cell scenarios (e.g., a cell with 100 MHz bandwidth using 4 antenna ports at the base station and 4 MIMO layers), we install

Financiado por la Unión Europea NextGenerationEU
Plan de Recuperación, Transformación y Resiliencia
UNICO I+D
GOBIERNO DE ESPAÑA
cttc Centre Tecnològic de Telecomunicacions de Catalunya

DPDK on the test server to improve the packet processing performance of gNB. To utilize and maximize the benefits of DPDK, we further configure the test server for DPDK execution, such as enabling 1 GB HugePages and binding the C-/U-Plane network interface to DPDK. By enabling DPDK, we notice a prominent gNB performance improvement in demanding scenario executions. For example, in the aforementioned 4 × 4, 100 MHz cell, running the gNB without DPDK causes BLER and low throughput to connected users, while this degradation does not happen with DPDK.

3.1.1.3.1.6 CoreSIM

In order to be able to emulate the performance of an end-to-end network, a 5G Core emulator, CoreSIM, is used. CoreSIM is a UPF and AMF Core Network emulator developed by Keysight [32]. It simplifies RAN testing by eliminating Core Network unwanted dependencies, thereby allowing an easily controllable and repeatable testing environment. CoreSIM is highly scalable, as it can handle up to 100 independent test lines in parallel, supporting 10000 UEs and 1000 registrations/sec. It also offers great flexibility through simulators for both EPC NSA and 5G SA configurations, supporting various data, storage, voice, and video protocols.

3.1.1.3.1.7 Open RAN Studio Player and Capture Appliance

As a capturing and debugging device, we use Keysight's S5040A Open RAN Studio Player and Capture Appliance [33]. It is an instrument grade test and measurement appliance designed to operate with Keysight's PathWave Signal Studio and Open RAN Studio to emulate a Distributed Unit (O-DU), capture O-RAN uplink communications, and perform the measurements necessary to validate an O-RU's functional operation and performance.

3.1.1.3.1.8 KS8400 – OpenTAP

To control different equipment and run desired commands, we make use of the Keysight PathWave Test Automation tool (KS8400) [34], which is based on the open-source OpenTAP, to implement and sequence all test steps. The key point is that this tool can execute the test steps and corresponding logic specified in a test plan file (in tapplan format). Using different OpenTAP plugins, KS8400 can orchestrate the operation of the instruments and pieces of software included in the testbed in a synchronous manner, as well as performing some basic recording for a postprocessing of results.

3.1.1.3.2 Scenario description and testbed configuration parameters

In order to facilitate the presentation of results, we describe here the different scenarios that have been emulated using the described testbed, as well as the different configuration parameters for these.

3.1.1.3.2.1 Scenarios

Using the testbed described above, different O-DU/O-CU centralization scenarios can be emulated. The key aspect in the emulation of these scenarios concerns mainly the settings of the NE3 instrument, as it is used to introduce different O-FH delay characteristics that mimic the signal propagation delays and delay variation that is expected to be encountered in such scenarios.

As it will be detailed in the following, we consider three main types of scenarios:

- A baseline scenario, representing the standard distributed deployment where the O-DU is located at the cell site, with no or minimal spatial separation between O-RU and O-DU. The O-FH introduces a minimal delay.
- Centralization scenarios in which pooling of multiple O-DU is performed in a centralized location to jointly serve a set of multiple O-RUs placed at different cell-sites with geographically distributed locations. In this situation, the O-RUs may be at a significant distance of the O-DU, leading to an O-FH that introduces substantial delays in the traffic traversing it.
- Finally, in order to investigate the ultimate limitations in terms of delays that the O-FH can tolerate, we also evaluate some additional scenarios in which larger O-FH delays than those of the centralized case are considered, even if not realistic in current commercial deployments.

Scenarios comprise, first and foremost, NE3 settings, specifically, delay distribution and values. Specifically, the following distributions and delay parameters have been studied in this project:

- Constant distribution: delay
- Uniform uncorrelated distribution: minimum delay, maximum delay
- Uniform saw-toothed distribution: minimum delay, maximum delay

Constant distribution simply adds a constant delay to all packets passing through NE3. An option exists to make this impairment transparent for PTP packets, although it has been disabled for our tests. Disabling it allows to emulate physically long O-FH connection, where the signal propagation delays equally both PTP and non-PTP packets.

Uniform uncorrelated distribution adds a uniformly random delay independently to subsequent packets. It is important to know that no reordering is allowed with this impairment, and thus the real distribution might be shifted towards the larger value, depending on the packet size, pipe capacity, distribution width, etc. Separate impairments for packet reordering exist, but have not been used within the scope of this project.

Uniform saw-toothed distribution deterministically affects packet delay following a sawtooth shape. Delays for consecutive packets start from the maximum value and slowly decrease towards the minimum value. Once this is reached, delays start again from the maximum value, as show in Figure 62.

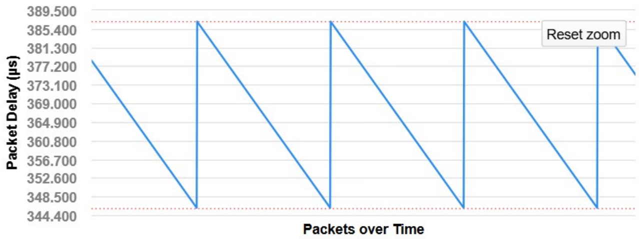


FIGURE 62. UNIFORM SAWTOOTHED DISTRIBUTION

3.1.1.3.2.1.1 Baseline scenario

The Baseline test scenario emulates the standard case in which the O-DU is co-located with the O-RU and, hence, the O-FH introduces negligible delay. The placement of NE3 in between RuSIM and the server running SRS' O-DU software implies, however, that the processing time of packets traversing NE3 introduces a small inherent delay that cannot be avoided. Hence, in the baseline scenario, described in Table 25, the O-FH introduces a constant delay equal to 15 us.

TABLE 25. DESCRIPTION OF THE BASELINE TEST SCENARIO

Test Scenario	Centralization Distance	Minimum O-FH delay	Maximum O-FH delay
BASELINE	0 km	15 us	15 us

3.1.1.3.2.1.2 O-DU Centralization Scenarios

We now turn our attention to the scenarios of interest for this PoC, which represents situations in which a pool of O-DUs is centralized at given location with the goal of serving a number of O-RUs that are deployed in cell sites geographically distributed around that polling location. The number of O-RUs served by the pool of O-DUs depends on the hierarchy level (HL) of the network at which the pooling is affected. In this case, and following the scenarios theorized by Telefónica based on its Spain´s transport network, we consider two different hierarchy levels at which pooling of O-DUs may occur: HL4 and HL3. The former corresponds to the aggregation of 9 cell sites, while the latter corresponds to the aggregation of ~23 cell sites. In addition, for each of these aggregation levels, we can consider different distance ranges for the cell sites, depending on the cell layout. In particular, we consider 3 distance ranges (short, average, and long), thus determining 6 possible scenarios, as described in Table 26.

TABLE 26. DESCRIPTION OF THE CENTRALIZATION TEST SCENARIOS

Test Scenario	Centralization Distance	Minimum O-FH delay	Maximum O-FH delay
HL4-SHORT	14.8 km	98 us	139 us
HL4-AVERAGE	19.8 km	123 us	164 us
HL4-LONG	22.12 km	134 us	175 us
HL3-SHORT	41.56 km	244 us	285 us
HL3-AVERAGE	47.78 km	275 us	316 us
HL3-LONG	61.96 km	346 us	387 us

As can be observed in Table 26, all 6 scenarios are characterized by a range of delays of 41 us (maximum FH delay – minimum FH delay) that the O-FH can introduce. In our testbed, this is implemented in different ways, depending on whether a constant or a variable delay is introduced by NE3:

- In the case of a constant delay, 3 possible subcases are considered for each of the test scenarios, which we coin as MIN, MID, and MAX. As their name indicate, in these three cases NE3 introduces a delay that corresponds to the minimum, mid-point, and maximum delay of the considered O-FH delay range.
- In the case of a variable delay (whether uniformly or saw-toothed distributed), NE3 introduces a delay that varies along the whole range 41 us range of delay introduced by the O-FH in each of the test scenarios.

3.1.1.3.2.1.3 Extra-long O-FH scenarios

Finally, the third set of scenarios is one in which the assumed O-FH delayed is increased beyond that of the scenarios presented in Table 26, which were based on real deployment cases in a real operator network. These scenarios do not correspond to any centralization setup and, therefore, we do not associate any centralization distance to them. They are synthetic scenarios merely meant to investigate how much can the delay introduced at the O-FH be stretched before losing all connectivity between O-DU and O-RU, and their characteristics are displayed in Table 27. As it can be seen, these scenarios are defined by a constant O-FH delay ranging from 0.5 to 1 ms.

TABLE 27. DESCRIPTION OF TEST SCENARIOS WITH EXTRA LONG O-FH

Test Scenario	Centralization Distance	Minimum O-FH delay	Maximum O-FH delay
XLONG-500	N/A	500 us	500 us
XLONG-600	N/A	600 us	600 us
XLONG-700	N/A	700 us	700 us
XLONG-800	N/A	800 us	800 us
XLONG-900	N/A	900 us	900 us
XLONG-1000	N/A	1000 us	1000 us

3.1.1.3.2.2 Delay management parameter settings

In order to adapt to the different O-FH test scenarios described above, in which packet exchanges between O-RU and O-DU are subject to different delays depending on the scenario considered, the O-DU needs to adapt the transmission (TX) and reception (RX) windows of the packets it will transmit / receive over the O-FH. The O-DU TX and RX windows are determined by its delay management parameters, mainly the parameters coined as *T1a* and *Ta4*. In our investigation, we consider two types of setting:

- A baseline setting, inspired by the setting recommended by the O-RAN Fronthaul Inter-Operability Test (IOT) Profile 1, described in [35].
- A modification of this setting that is designed to adapt to longer O-FH delays.

We present both of these settings next.

3.1.1.3.2.2.1 IOT Profile 1 Delay Management Parameters

The delay management parameters for this case are described in Table 28.

TABLE 28. O-FH DELAY MANAGEMENT PARAMETERS IN O-RAN IOT PROFILE 1

Parameter	**Value (us)**
FH Parameters	
T12_max, T34_max	160
T12_min, T34_min	0
RuSIM Parameters	

Tcp_adv_dl	125
T2a_max_cp_dl	500 (=T2a_max_up + Tcp_adv_dl)
T2a_min_cp_dl	259 (=T2a_min_up + Tcp_adv_dl)
T2a_max_cp_ul	336
T2a_min_cp_ul	125
T2a_max_up	375 (>=345)
T2a_min_up	134 (<=134)
Ta3_max_up	171
Ta3_min_up	50
SRS O-DU Parameters (usec)	
T1a_max_cp_dl	500
T1a_min_cp_dl	419
T1a_max_cp_ul	336
T1a_min_cp_ul	285
T1a_max_up	375
T1a_min_up	294
Ta4_max_up	331
Ta4_min_up	50

3.1.1.3.2.2.2 Scenario-adjusted

When considering the case of O-DU pooling at centralized locations, it is important to adjust the O-DU TX and RX window parameters in order to account for the extra delays introduced by the O-FH in such situation, so that the packets transmitted by the O-DU arrive within the O-RU RX window, and the O-RU packet transmissions arrive within the O-DU RX window. To do so, we adjust the O-DU parameters based on the maximum (**max_FH_delay**) and minimum (**min_FH_delay**) O-FH delays reported in Table 26 and Table 27, as detailed in Table 29. In Table 30, we show comprehensively the value of all delay management parameters resulting from applying the adjustment to the O-DU pooling test scenarios.

Financiado por la Unión Europea NextGenerationEU
Plan de Recuperación, Transformación y Resiliencia
UNICO I+D
GOBIERNO DE ESPAÑA
cttc Centre Tecnològic de Telecomunicacions de Catalunya

TABLE 29. SCENARIO-ADJUSTED O-FH DELAY MANAGEMENT PARAMETERS

Item	Value (us)
FH Parameters	
T12_max, T34_max	**max_FH_delay**
T12_min, T34_min	**min_FH_delay**
RuSIM Parameters	
Tcp_adv_dl	125
T2a_max_cp_dl	500 (=T2a_max_up + Tcp_adv_dl)
T2a_min_cp_dl	259 (=T2a_min_up + Tcp_adv_dl)
T2a_max_cp_ul	336
T2a_min_cp_ul	125
T2a_max_up	375 (>=345)
T2a_min_up	134 (<=134)
Ta3_max_up	171
Ta3_min_up	50
SRS O-DU Parameters (usec)	
T1a_max_cp_dl	375 + **min_FH_delay** + 125 (= T1a_max_up + Tcp_adv_dl = T2a_max_cp_dl + **min_FH_delay**)
T1a_min_cp_dl	134 + **max_FH_delay** + 125 (= T1a_min_up + Tcp_adv_dl = = T2a_min_cp_dl + **max_FH_delay**)
T1a_max_cp_ul	336 + **min_FH_delay** (= T2a_max_cp_ul + T12_min)
T1a_min_cp_ul	125 + **max_FH_delay** (= T2a_min_cp_ul + T12_max)
T1a_max_up	375+ **min_FH_delay** (= T2a_max_up + T12_min)
T1a_min_up	134 + **max_FH_delay** (= T2a_min_up + T12_max)
Ta4_max_up	171 + **max_FH_delay** (= Ta3_max_up + T34_max)
Ta4_min_up	50 + **min_FH_delay** (= Ta3_min_up + T34_min)

TABLE 30. SCENARIO-ADJUSTED DELAY MANAGEMENT PARAMETERS FOR THE O-DU POOLING TEST SCENARIOS

Test Scenario	HL3	HL3	HL3	HL3	HL3	HL3	HL4	HL4	HL4	HL4	HL4	HL4	HL4	HL4	HL4	HL3	HL3	HL3
	Longest	Longest	Longest	Shortest	Shortest	Shortest	Shortest	Shortest	Shortest	Average	Average	Average	Longest	Longest	Longest	Average	Average	Average
	max	mid	min	max	mid	min	max	mid	min	max	mid	min	max	mid	min	max	mid	min
NE3 Delay	387	366	346	285	264	244	139	118	98	164	143	123	175	154	134	316	295	275
max delay	387	387	387	285	285	285	139	139	139	164	164	164	175	175	175	316	316	316
min delay	346	346	346	244	244	244	98	98	98	123	123	123	134	134	134	275	275	275
t1a_max_cp_dl	846	846	846	744	744	744	598	598	598	623	623	623	634	634	634	775	775	775
t1a_min_cp_dl	646	646	646	544	544	544	398	398	398	423	423	423	434	434	434	575	575	575
t1a_max_cp_ul	682	682	682	580	580	580	434	434	434	459	459	459	470	470	470	611	611	611
t1a_min_cp_ul	512	512	512	410	410	410	264	264	264	289	289	289	300	300	300	441	441	441
t1a_max_up	721	721	721	619	619	619	473	473	473	498	498	498	509	509	509	650	650	650
ta1_min_up	521	521	521	419	419	419	273	273	273	298	298	298	309	309	309	450	450	450
ta4_max	558	558	558	456	456	456	310	310	310	335	335	335	346	346	346	487	487	487
ta4_min	396	396	396	294	294	294	148	148	148	173	173	173	184	184	184	325	325	325
t2a_max_cp_dl	500	500	500	500	500	500	500	500	500	500	500	500	500	500	500	500	500	500
t2a_min_cp_dl	259	259	259	259	259	259	259	259	259	259	259	259	259	259	259	259	259	259
t2a_max_cp_ul	336	336	336	336	336	336	336	336	336	336	336	336	336	336	336	336	336	336
t2a_min_cp_ul	125	125	125	125	125	125	125	125	125	125	125	125	125	125	125	125	125	125
t2a_max_up	375	375	375	375	375	375	375	375	375	375	375	375	375	375	375	375	375	375
t2a_min_up	134	134	134	134	134	134	134	134	134	134	134	134	134	134	134	134	134	134
ta3_max_up	171	171	171	171	171	171	171	171	171	171	171	171	171	171	171	171	171	171
ta3_min_up	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50
Tcp_adv_dl	125	125	125	125	125	125	125	125	125	125	125	125	125	125	125	125	125	125

Financiado por la Unión Europea NextGenerationEU
Plan de Recuperación, Transformación y Resiliencia
UNICO I+D
Gobierno de España
cttc Centre Tecnològic de Telecomunicacions de Catalunya

3.1.1.3.2.3 Other testbed configuration parameters

We report next other miscellaneous settings used in the testbed.

3.1.1.3.2.4 Air-interface and O-FH U-Plane Compression Parameters

We start by parameters to be configured in RuSIM, the O-RU emulation. RuSIM emulates the transmission of DL packets by the O-RU over the air interface, as well as the transmission of UE packets to the O-RU in the UL. Hence, the assumed parameters for the air interface emulation need to be determined. In addition to this, different compression options for the packets that the O-RU transmits and receives over the O-FH are possible, as these need to be set in agreement with the O-DU. Both types of parameters, along with the settings we have used, are reported in Table 31.

TABLE 31. AIR-INTERFACE EMULATION AND O-FH COMPRESSION PARAMETERS

Category	Item	Value
General	Radio access technology	NR TDD
	TDD configuration	DDDDDDDSUU
	Nominal sub-carrier spacing	30 kHz
	Total channel bandwidth	100 MHz
	Number of spatial/antenna streams	4
	Fronthaul Ethernet Link	10 Gbps x 1 lane
	RU category	Category A
IQ Compression	U-Plane data compression method	Block floating point
	U-Plane data IQ bitwidth	9
	IQ data frame format not including udCompHdr field	TRUE

3.1.1.3.2.4.1 Traffic characteristics

Traffic is based on Keysight’s UDG protocol. This protocol is natively integrated within both RuSIM and CoreSIM, and allows the user to precisely control UL and DL traffic independently. It is possible to have simultaneous UL+DL traffic, or just one of them. If only mono-directional traffic is performed, it is important to know that control messages in the other direction will still be present, thus making traffic in the other direction non-zero.

Financiado por la Unión Europea NextGenerationEU
Plan de Recuperación, Transformación y Resiliencia
UNICO I+D
GOBIERNO DE ESPAÑA
cttc Centre Tecnològic de Telecomunicacions de Catalunya

Two types of traffic definitions have been used throughout all experiments:

1. Default traffic burst.
2. Traffic bursts for average E2E latency estimation.

As shown in Figure 63, the *Default scenario* comprises a single flow type, a single UE with static mobility, performing a set of actions defined in two subsequent sessions. Static mobility allows us to control parameters such as position, speed, frequency offset, CQI, AWGN, RSRP, RI, etc. In fact, UEs are emulated within RuSIM, thus enabling full control of the radio conditions between the UEs and the O-RU. Importantly, given that our setup comprises 4 spatial streams, we set the Rank Indicator (RI) equal to 4, allowing us to obtain the maximum throughput through MIMO multiplexing in DL.

Two sessions have been created: a first session related to registration and PDU session establishment and a second one related to the actual data exchange. This allows us to logically separate the UE connection setup from the actual data exchange. In case multiple UEs are part of the scenario, e.g., in the *Average E2E latency scenario*, the second session does not start until the first one is completed. This allows having all UEs correctly attached and with a PDU Session already established, and thus starting the data bursts all simultaneously.

In the second session, a set of increasing DL-only traffic is generated from CoreSIM towards the UE. Traffic bursts last 60 seconds, while pauses between bursts last 5 seconds. A longer 20 second pause is performed before starting the UL-only traffic, i.e., traffic created from the RuSIM's UE going towards CoreSIM. Pauses between bursts allow post-processing scripts to easily detect bursts, greatly simplifying the data analysis. The traffic bursts created in this session are illustrated in Figure 64.

Finally, the UE releases the PDU session and deregisters from the network.

Financiado por la Unión Europea NextGenerationEU
Plan de Recuperación, Transformación y Resiliencia
UNICO I+D
GOBIERNO DE ESPAÑA
cttc Centre Tecnològic de Telecomunicacions de Catalunya

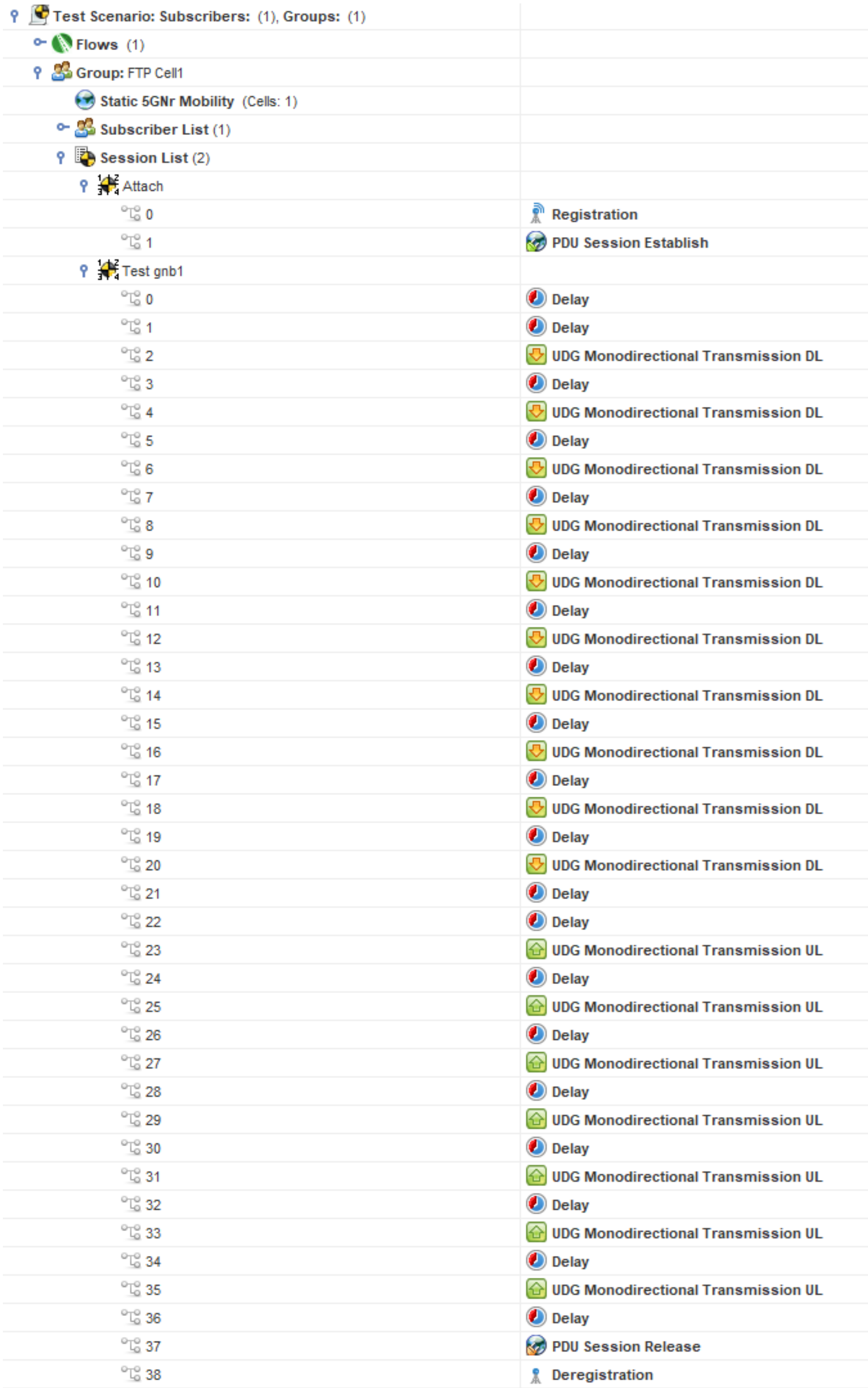


FIGURE 63. DEFAULT TRAFFIC SCENARIO AS CONFIGURED IN THE UDG TRAFFIC GENERATOR

The *Average E2E latency scenario* resembles the default one just described. To more accurately extract E2E latencies, though, a single burst 5-minutes burst is performed during the second session, allowing us to capture more latency data. Furthermore, we found that the number of UEs also affects some results, and this variable is also added to the scenario. When multiple UEs are connected, traffic is considered to be aggregate on all UEs, conversely, each UE experiences $\frac{1}{\#UEs}$ of the total traffic.

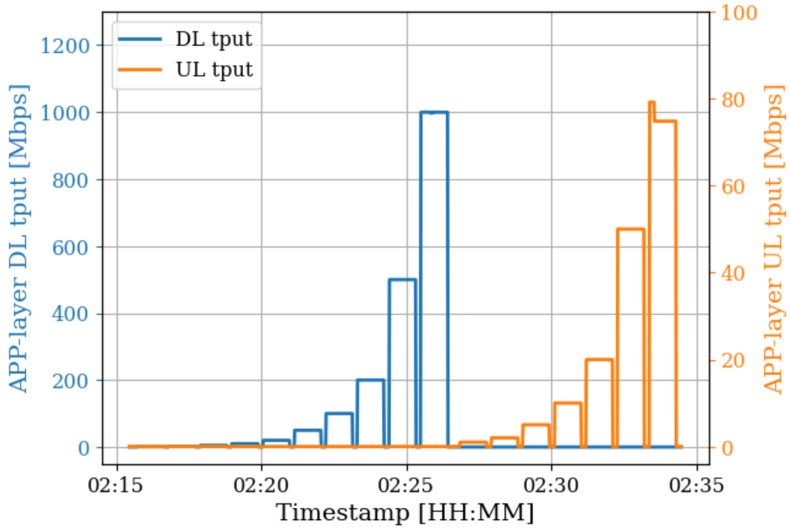


FIGURE 64. ILLUSTRATION OF THE DL (BLUE) AND UL (ORANGE) TRAFFIC BURSTS USED IN THE DEFAULT TRAFFIC SCENARIO

3.1.1.3.2.4.2 UL SNR and DL BLER

RuSIM allows emulating some impairments over the wireless channel. Specifically, it allows adding AWGN noise to UL signals, and BLER (separately or each MCS) to DL signals. Specifically, these settings can be controlled by the mobility model. For simplicity, within this project we only used the *Static 5GNr Mobility*, meaning that a configuration is maintained constant throughout the entire scenario duration.

UL AWGN noise has to be enabled first, then it can be set relative to the UL signal power in dB units. This means that $SNR_{dB} = -AWGN_{dB}$, e.g., $AWGN = -5\ dB \rightarrow SNR = 5\ dB$.

DL BLER, instead, can be set independently for each MCS. This means that, for example, the user can set DL BLER for MCS1=0, for MCS2=0.0001, for MCS3=0.0002, etc.

3.1.1.3.2.4.3 O-DU/O-CU: max_proc_delay (MPD)

One additional srsRAN O-DU parameter used within this project is `expert_phy:max_proc_delay` (MPD). As explained in Sections 3.1.1.2.1.5, ..his parameter controls the maximum allowed DL processing delay (in number of slots), which defaults to 5 slots (or 2.5 ms considering the 30 kHz SCS used). The idea is that longer MPD trades higher E2E latency to allow longer processing latency, possibly due to an exceptionally long O-FH or to underperforming hardware.

3.1.1.3.2.4.4 SUT load

As the O-DU needs to perform heavy signal processing tasks in real time and respecting the tight timing constraints imposed by the O-FH timing management model, it is expected that the transmission of packets over the O-FH may be affected by the computational load of the server running the O-DU. To test this, we run some tests in which the server running the O-DU is loaded with some extra computing tasks, and we compare the results with those when the server is exclusively running the O-DU and O-CU tasks.

3.1.1.3.3 Results

We now begin reporting the results obtained in the testbed. We will focus mostly on the evaluation of the O-DU pooling scenarios defined in Table 26, and how the performance of the system when using the scenario-adjusted delay management parameters compares with respect to when using the baseline parameters.

3.1.1.3.3.1 Centralization of O-DU/CUs

We start by considering the scenarios of Table 26. First, we inspect the statistics collected by the O-DU and the O-RU in terms of O-FH packets that arrive on-time, early, or late at each of the ends. Then, we focus on the throughput and block-error rate results achieved. Throughout this investigation, we assume an ideal air-interface situation, in which no packet errors exist in the over-the-air transmission. This allows us to isolate the effect of the O-FH in the end-to-end system performance, without this analysis being obscured by the inevitable performance degradation that will be introduced by wireless channels.

3.1.1.3.3.1.1 Ideal air-interface

3.1.1.3.3.1.1.1Packet timing with different scenarios

Next, we depict the rate of packets that arrive early, on-time, and late in each of the scenarios. For each scenario, two tests are run: one using the baseline delay management parameters in Table 28 (labelled as *IOT profile 1* in the legend of the figures), and another test using the adjusted parameters of Table 29 (labelled as *Adjusted* in the legend of the figures). In both tests, NE3 introduces two types of delay respectively: either a constant delay or a random delay uniformly distributed within ±20 µs around the value in the corresponding scenario.

In all subsequent figures, a green vertical line at 160 µs indicates the timing threshold of the baseline configuration: under an O-FH which incurs delays smaller than 160 µs (i.e., to the left of the green line), all O-FH packets transmitted by O-DU and O-RU in their respective TWs should be received within the RW of the counterpart.

3.1.1.3.3.1.1.1.1 UL User Plane

For UL user plane traffic, both scenarios exhibit similar behaviour when the O-FH delay is below 160 µs (left of the green line). However, when the O-FH delay exceeds 160 µs, a noticeable divergence emerges. Specifically, in the baseline configuration (blue lines), the late packet ratio grows significantly, whereas it remains stable under the scenarios with adjusted parameters. Correspondingly, on-time packet ratio in the baseline configuration shows a markable decrease, indicating a strong sensitivity to delay. As for the early packet, the ratio remains consistently low across all tests, as expected in theory analysis.

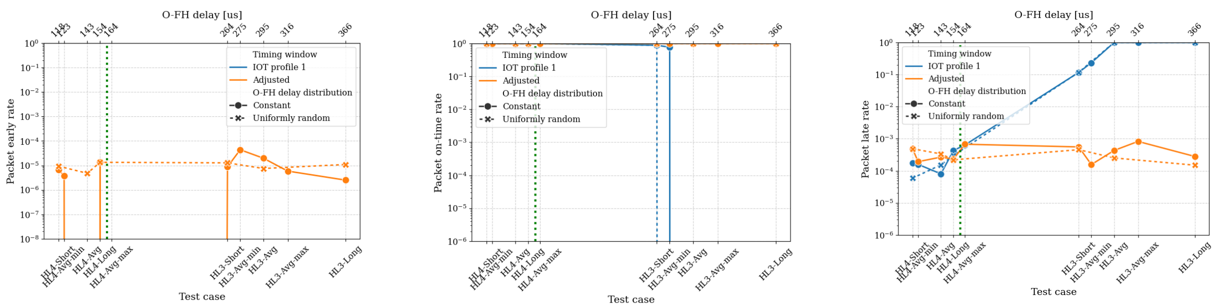


FIGURE 65. RATE OF EARLY (LEFT), ON-TIME (MIDDLE) AND LATE (RIGHT) PACKETS OBSERVED IN THE UL IN USER-PLANE TRAFFIC

3.1.1.3.3.1.1.1.2 UL Control Plane

The timing evaluation of UL control plane packets, conducted at the O-RU, presents results consistent with those observed in the user plane: a significant increase happens in the late packet ratio when the O-FH delay grows in baseline scenarios. In contrast, this is not observed in the scenarios using adjusted parameters, where the late packet ratio remains 0. This behaviour also leads to the different trends for the on-time packet ratio: in baseline scenarios, it decreases sharply once the O-FH delay is larger than 160 µs, while remaining stable in the adjusted scenarios. As for early packets, all tests report a value of 0, aligning with the previous observations.

Financiado por la Unión Europea NextGenerationEU
Plan de Recuperación, Transformación y Resiliencia
UNICO I+D
GOBIERNO DE ESPAÑA
cttc Centre Tecnològic de Telecomunicacions de Catalunya

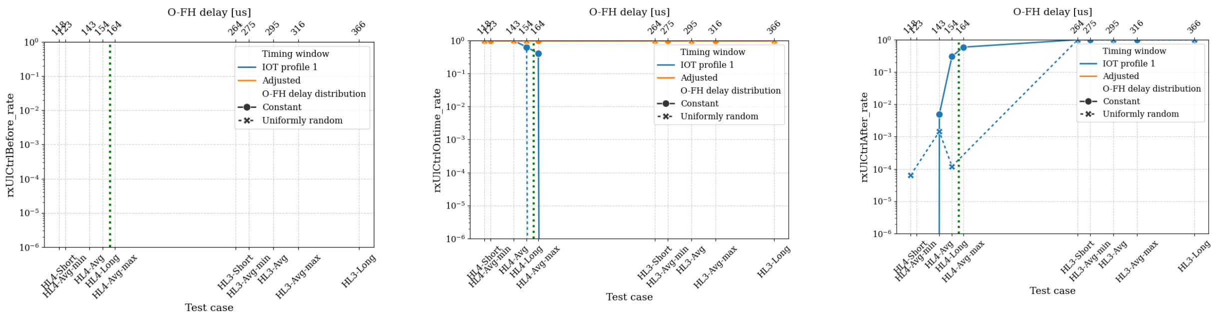


FIGURE 66. RATE OF EARLY (LEFT), ON-TIME (MIDDLE) AND LATE (RIGHT) PACKETS OBSERVED IN THE UL IN CONTROL-PLANE TRAFFIC

3.1.1.3.3.1.1.1.3 DL User Plane

The O-RU also collects packet timing statistics for DL traffic. Specifically, the user plane traffic on DL shows a markable increase in the late packet ratio in baseline scenarios as the O-FH delay increases, while this ratio remains at a low and stable level in tests conducted with adjusted timing window parameters.

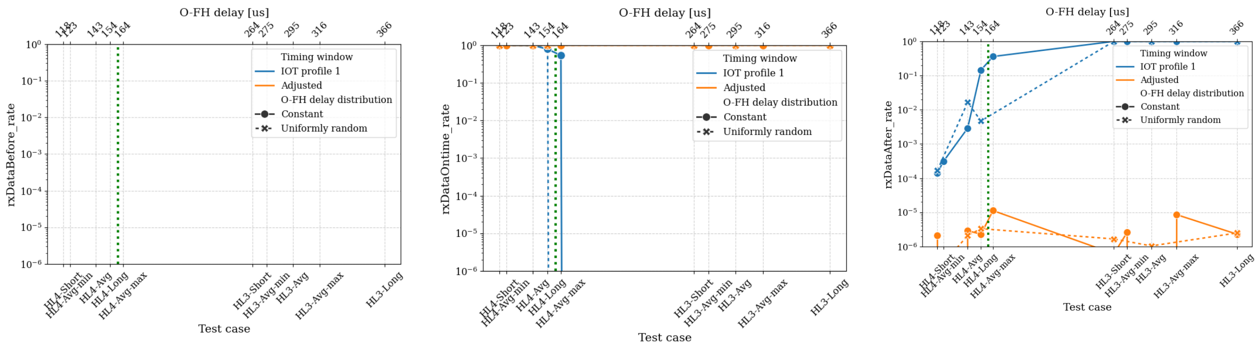


FIGURE 67. RATE OF EARLY (LEFT), ON-TIME (MIDDLE) AND LATE (RIGHT) PACKETS OBSERVED IN THE DL IN USER-PLANE TRAFFIC

3.1.1.3.3.1.1.1.4 DL Control Plane

Similarly, the packet timing analysis for DL control plane traffic reveals trends consistent with those observed in other traffic types. In baseline scenarios, increasing O-FH delay results in a significant impact on the late and on-time packet ratios, while this has minimum effect in scenarios using adjusted parameters.

The above observations clearly demonstrate the strong influence of O-FH delay on packet timing. However, by properly adjusting the timing window parameters, packet timing violation can be substantially reduced even in long O-FH delay cases.

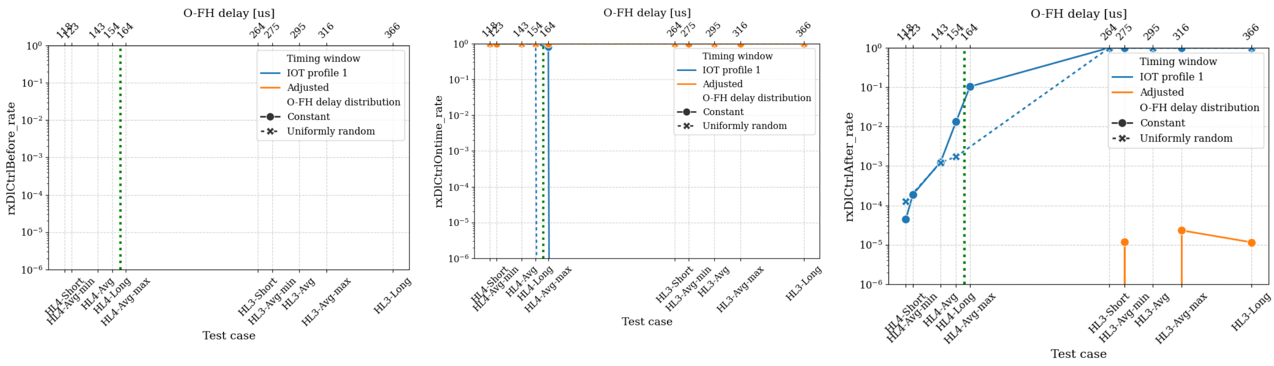


FIGURE 68. RATE OF EARLY (LEFT), ON-TIME (MIDDLE) AND LATE (RIGHT) PACKETS OBSERVED IN THE DL IN CONTROL-PLANE TRAFFIC

3.1.1.3.3.1.1.2 Throughput and BLER results

Next, we report the throughput and BLER achieved in either UL (PUSCH) or DL (PDSCH), depicted in Figure 69 and Figure 70. These figures present consistent results with the previous packet timing analysis. The results indicate that the test cases with O-DU aggregation at the HL4 level can all be accomplished without any noticeable performance degradation with the default TW and RW settings. For the test cases with O-DU aggregation at the HL3 level, though, only the two cases with shorter O-FH delay provide decent performance, even if a small degradation can be observed in the downlink throughput. For the rest of HL3 test cases, which imply longer delays in the OFH, communication between the O-RU and the O-DU along the O-FH interface is not possible with the default TW/RW settings, and no throughput is obtained. When, on the other hand, the TW/RW boundaries are adjusted as described before, we observe that the throughput obtained in all test cases hovers around the offered traffic, and no degradation of the performance can be observed with increasing centralization distance. Remarkably, packet delay variation across the O-FH results in no degradation of the performance. These results confirm that, with careful adjustment of the TW/RW parameters at the O-DU, O-FH of significantly larger length can be deployed with no E2E performance penalty.

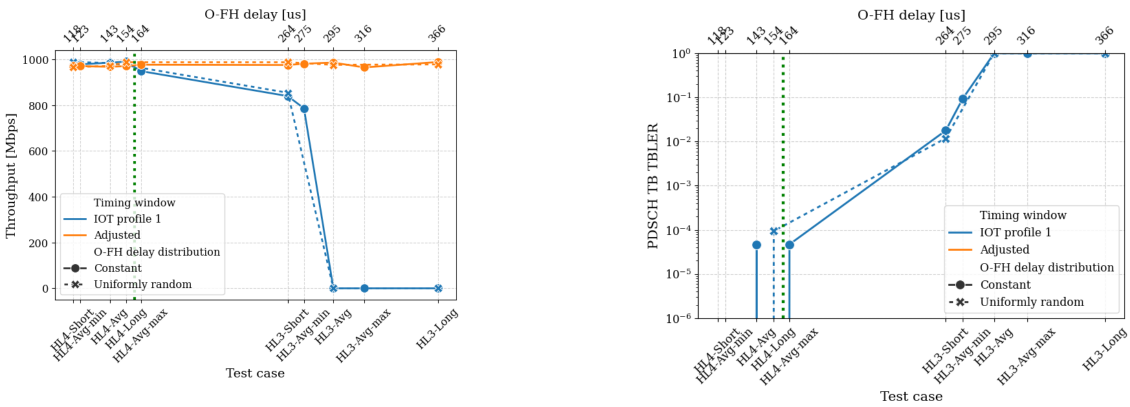


FIGURE 69. THROUGHPUT (LEFT) AND BLER (RIGHT) OBSERVED IN THE DOWNLINK (PDSCH)

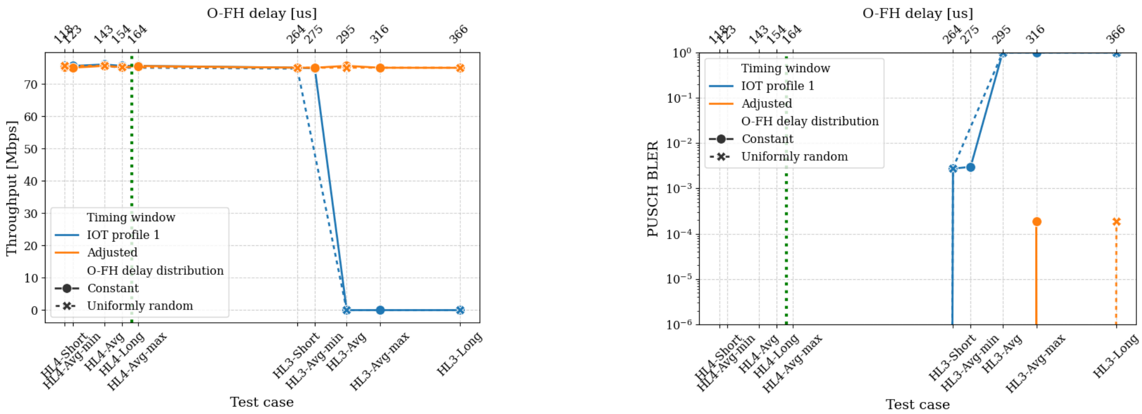


FIGURE 70. THROUGHPUT (LEFT) AND BLER (RIGHT) OBSERVED IN THE UPLINK (PUSCH)

3.1.1.3.3.1.2 Realistic SNR + Multiple UEs

In the following results, the assumption of an idealistic air-interface is dropped, and results assuming and UL SNR of 0 dB are reported.

Under this condition, the results in Figure 71 indicate that the UL MCS is reduced to 4, and PUSCH throughput can no longer reach the configured value of 78 Mbps, but is instead limited to approximately 8.3 Mbps in all O-FH timing configurations. Notably, the UL BLER remains at a relatively stable value of 0.01 across all tests. This may be attributed to the O-DU configuration parameter *olla_target_bler*, which defines the target BLER in Outer-Loop Link Adaptation (OLLA) algorithm.

Although noise is only applied on the UL channel, we notice in Figure 72 that the DL MCS and throughput are also heavily affected, despite the DL BLER remaining at 0. Specifically, the MCS reduces to 20, while PDSCH throughput decreases from 1 Gbps (under ideal air interface conditions) to approximately 380 Mbps.

Besides, all these figures demonstrate that the impact of introducing a realistic SNR on the UL channel is consistent across different O-FH timing configurations, as both throughput and BLER remain unchanged in tests with different O-FH delays and timing window settings.

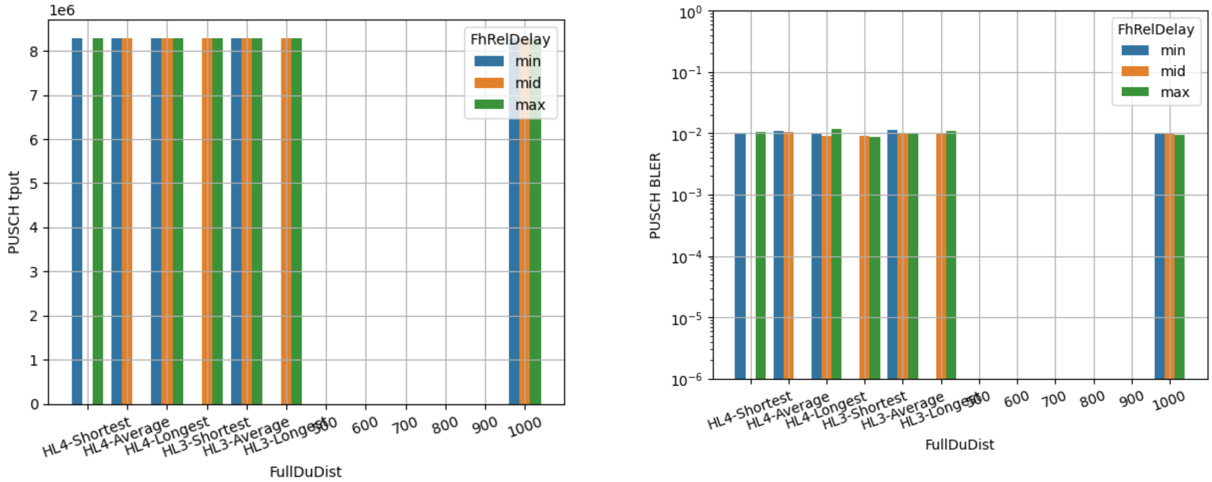


FIGURE 71. THROUGHPUT (LEFT) AND BLER (RIGHT) OBSERVED IN THE UPLINK WITH UL SNR SET TO 0 DB

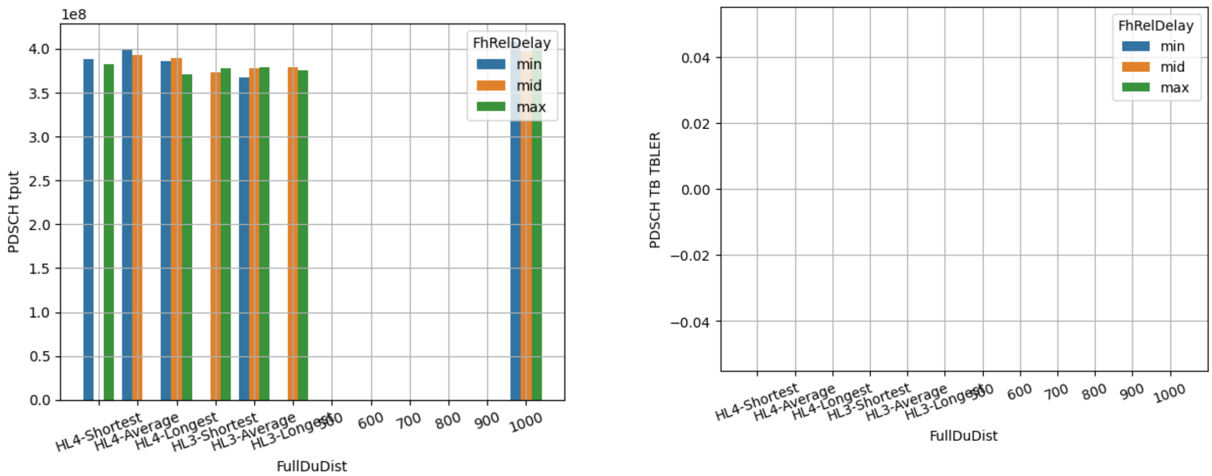


FIGURE 72. THROUGHPUT (LEFT) AND BLER (RIGHT) OBSERVED IN THE DOWNLINK WITH UL SNR SET TO 0 DB

		DL								UL									
pci	rnti	cqi	ri	mcs	brate	ok	nok	(%)	dl_bs	pusch	rsrp	mcs	brate	ok	nok	(%)	bsr	ta	phr
1	4606	15	4.0	20	366M	871	679	43%	4.86M	0.1	-61.2	4	102k	25	0	0%	0	0	38
1	4606	15	4.0	20	393M	934	616	39%	4.78M	0.0	-61.2	4	106k	26	0	0%	0	0	38
1	4606	15	4.0	20	348M	829	721	46%	4.86M	0.1	-61.2	4	102k	25	0	0%	0	0	38
1	4606	15	4.0	20	375M	892	658	42%	4.48M	0.1	-61.2	4	102k	25	0	0%	0	0	38
1	4606	15	4.0	20	367M	889	661	42%	5.16M	0.1	-61.2	4	102k	25	0	0%	0	0	38
1	4606	15	4.0	20	332M	790	760	49%	5.16M	0.1	-61.2	4	102k	25	0	0%	0	0	38
1	4606	15	4.0	20	431M	1025	525	33%	5.16M	-0.0	-61.2	4	106k	26	1	3%	0	0	38
1	4606	15	4.0	20	371M	885	665	42%	4.48M	0.0	-61.2	4	102k	25	1	3%	0	0	38
1	4606	15	4.0	20	345M	822	728	46%	5.16M	0.1	-61.2	4	102k	25	0	0%	0	0	38
1	4606	15	4.0	20	336M	801	749	48%	4.78M	0.1	-61.2	4	102k	25	0	0%	0	0	38
1	4606	15	4.0	20	114M	275	189	40%	0	-0.0	-61.2	4	41k	10	0	0%	0	0	38

		DL								UL									
pci	rnti	cqi	ri	mcs	brate	ok	nok	(%)	dl_bs	pusch	rsrp	mcs	brate	ok	nok	(%)	bsr	ta	phr
1	4606	15	4.0	20	61k	37	0	0%	0	0.2	-61.2	4	4.0M	198	0	0%	300k	0	38
1	4606	15	4.0	20	121k	75	0	0%	0	0.2	-61.2	4	8.2M	396	4	1%	300k	0	38
1	4606	15	4.0	20	119k	74	0	0%	0	0.2	-61.2	4	8.2M	396	4	1%	300k	0	38
1	4606	15	4.0	20	119k	74	0	0%	0	0.2	-61.2	4	8.2M	396	4	1%	300k	0	38
1	4606	15	4.0	20	121k	74	0	0%	0	0.2	-61.2	4	8.2M	396	4	1%	300k	0	38
1	4606	15	4.0	20	119k	74	0	0%	0	0.2	-61.2	4	8.2M	396	4	1%	300k	0	38
1	4606	15	4.0	20	126k	74	0	0%	0	0.2	-61.2	4	8.2M	396	4	1%	300k	0	38
1	4606	15	4.0	21	128k	74	0	0%	0	0.2	-61.2	4	8.2M	396	4	1%	300k	0	38
1	4606	15	4.0	21	130k	75	0	0%	0	0.2	-61.2	4	8.2M	396	4	1%	300k	0	38
1	4606	15	4.0	21	128k	74	0	0%	0	0.2	-61.2	4	8.2M	396	4	1%	300k	0	38
1	4606	15	4.0	21	131k	74	0	0%	0	0.2	-61.2	4	8.2M	396	4	1%	300k	0	38

FIGURE 73. SNAPSHOT OF PUSCH AND PDSCH METRICS PROVIDED BY SRSRAN O-DU

The results in Figure 74 illustrate how different levels of UL air-interface noise influence network KPIs under different O-FH configurations. We can observe a strong negative correlation between PHY-layer throughput and noise level. However, this correlation behaves differently in DL and UL. In the UL, the throughput decreases gradually as the SNR drops below 22 dB. However, the DL throughput remains nearly constant down to 4 dB SNR, and then declines sharply below this threshold.

Figure 75 also indicates that noise has no observable effect on PDSCH BLER, which remains zero across different SNR values. In contrast, the PUSCH BLER is sensitive to SNR, as it is 0 for SNR values above 20 dB, but increases to 0.01 once the SNR falls below this threshold. Note that this value is consistent with the threshold observed for the drop in PUSCH throughput.

DL one-way latency is also affected by the noise level, particularly under high throughput or noise conditions. Specifically, in Figure 76 we notice that increased noise leads to higher latency at 1 Gbps throughput, while its impact is minimal at lower throughput levels. Additionally, the DL latency increases sharply as the SNR drops from 4 to 2 dB but remains relatively constant at higher SNR values. Moreover, the figure also demonstrates that the impact of air-interface noise is independent of the O-FH configuration. No matter whether the O-FH delay is low (15 µs), medium (366 µs), or high (1000 µs), the trends are similar.

Financiado por la Unión Europea NextGenerationEU
Plan de Recuperación, Transformación y Resiliencia
UNICO I+D
GOBIERNO DE ESPAÑA
cttc Centre Tecnològic de Telecomunicacions de Catalunya

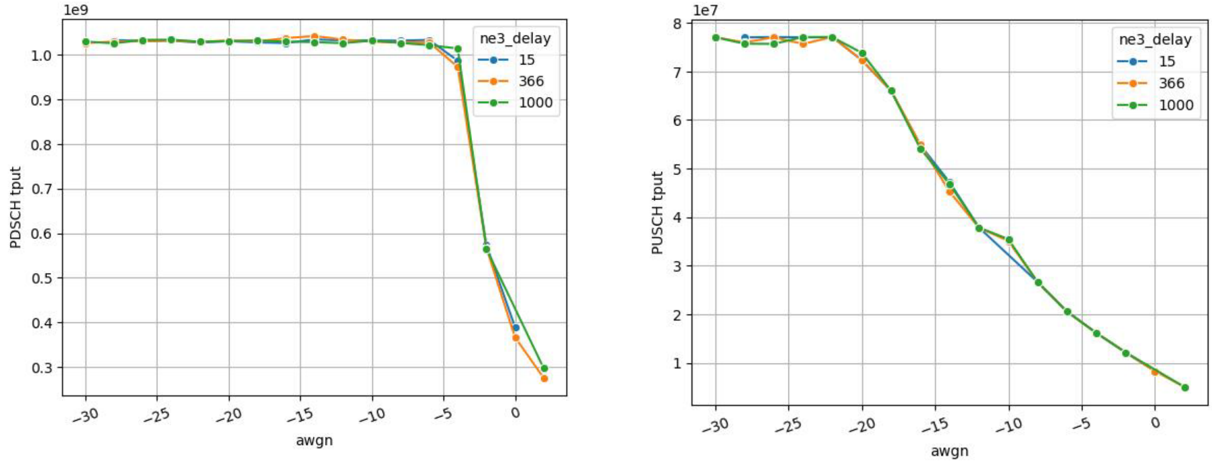


FIGURE 74. THROUGHPUT ACHIEVED AT THE PDSCH (LEFT) AND PUSCH (RIGHT) VS THE UL NOISE LEVEL

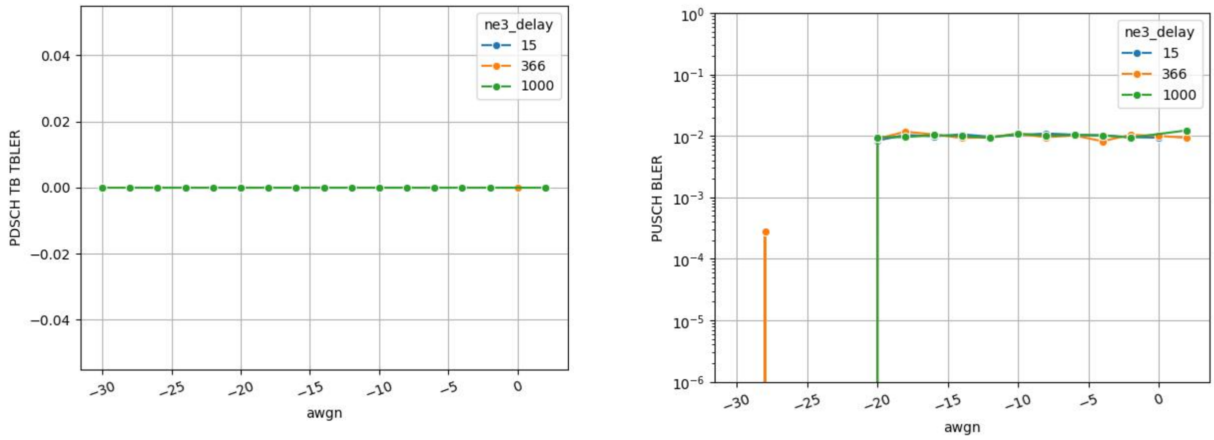


FIGURE 75. BLER ACHIEVED AT THE PDSCH (LEFT) AND PUSCH (RIGHT) VS THE UL NOISE LEVEL

Financiado por la Unión Europea NextGenerationEU
Plan de Recuperación, Transformación y Resiliencia
UNICO I+D
GOBIERNO DE ESPAÑA
cttc Centre Tecnològic de Telecomunicacions de Catalunya

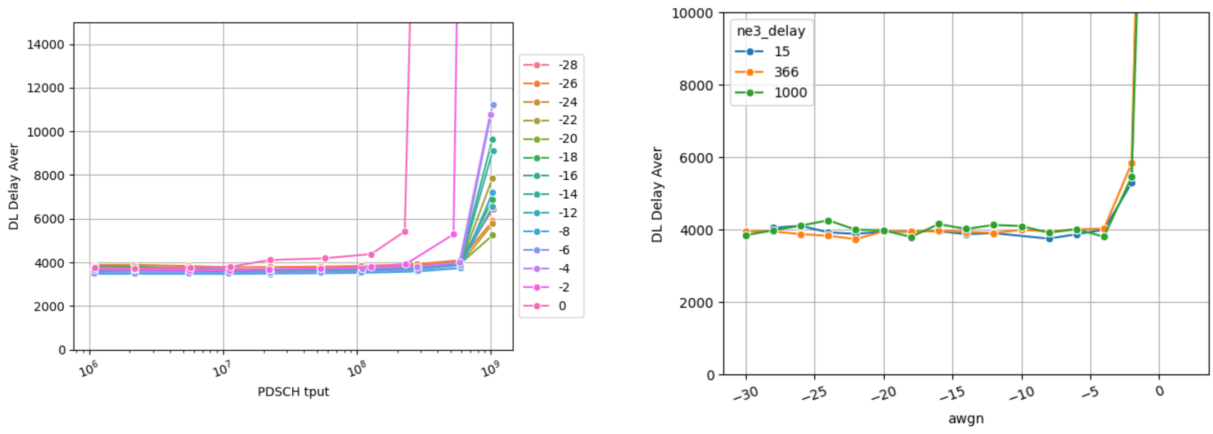


FIGURE 76. LEFT. DL LATENCY ACHIEVED AT DIFFERENT UL NOISE LEVELS WITH AN O-FH DELAY OF 15 US. RIGHT: DL LATENCY ACHIEVED WITH DIFFERENT O-FH DELAYS FOR A FIXED PDSCH THROUGHPUT OF 500 MBPS

3.1.1.3.3.2 Packet timing analysis

Within the scope of the project, some metrics collected from the DU in the uplink, suggested the DU was reporting a portion of UL packets arriving late in time compared to their expected allowed temporal window. Addressing the related investigation involved conducting repeated incremental testing to rule out any random artifact or misbehaviours and after than required in depth root cause analysis.

As any timing mismatch needs to be always observed from both potential failing sides, we tried to first check initially the traffic capturing from the sending and receiving servers. But in doing so we found that direct capture in the servers at such high rates was also impacting the accuracy/precision in the timestamping and could even incorrectly suggest lost packets because of the limitation of the software capture process itself.

Finally, we decided to use the hardware-based capture and analysis tools from Keysight Open RAN Studio and from the Network Emulator 3 hardware equipment. By doing that, and cross comparing with the timing of associated PTP messages, we were able to confidently confirm that the RU was not transmitting packets before the start of the transmission window. In Figure 77 it can be observed that in an explicit initial analysis all packets at a specific subframe and symbol captured at transmission port plane consistently have timestamps above the expected 50 microseconds. Then, a more detailed analysis in Figure 78 using the delay management analysis features in Open RAN studio provided further evidence on the absence of actual early in excess packets.

Financiado por la Unión Europea NextGenerationEU
Plan de Recuperación, Transformación y Resiliencia
UNICO I+D
GOBIERNO DE ESPAÑA
cttc Centre Tecnològic de Telecomunicacions de Catalunya

Thanks to the confidence that these set of analysis tests and tools provided, the consortium partners were able to set higher focus on investigating the uplink receiving part and identify the actual root cause for the initially offending metric.

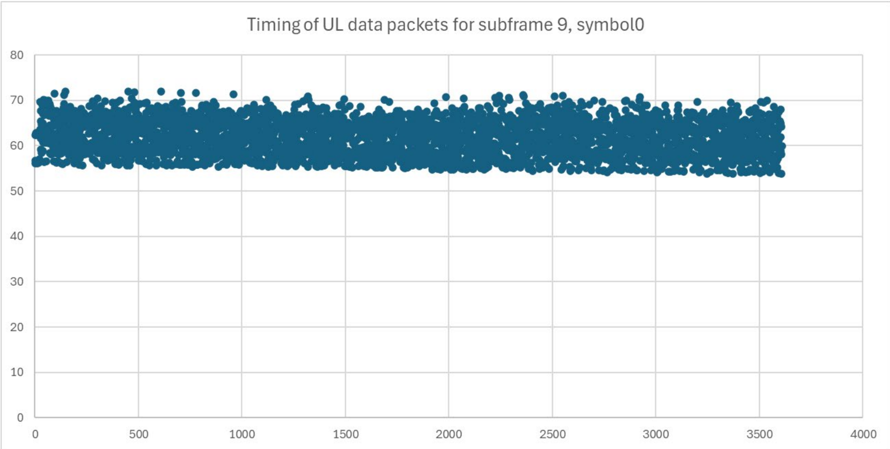


FIGURE 77. TIMING ANALYSIS ON OPEN-RAN STUDIO PCAP

Financiado por la Unión Europea NextGenerationEU
Plan de Recuperación, Transformación y Resiliencia
UNICO I+D
GOBIERNO DE ESPAÑA
Centre Tecnològic de Telecomunicacions de Catalunya

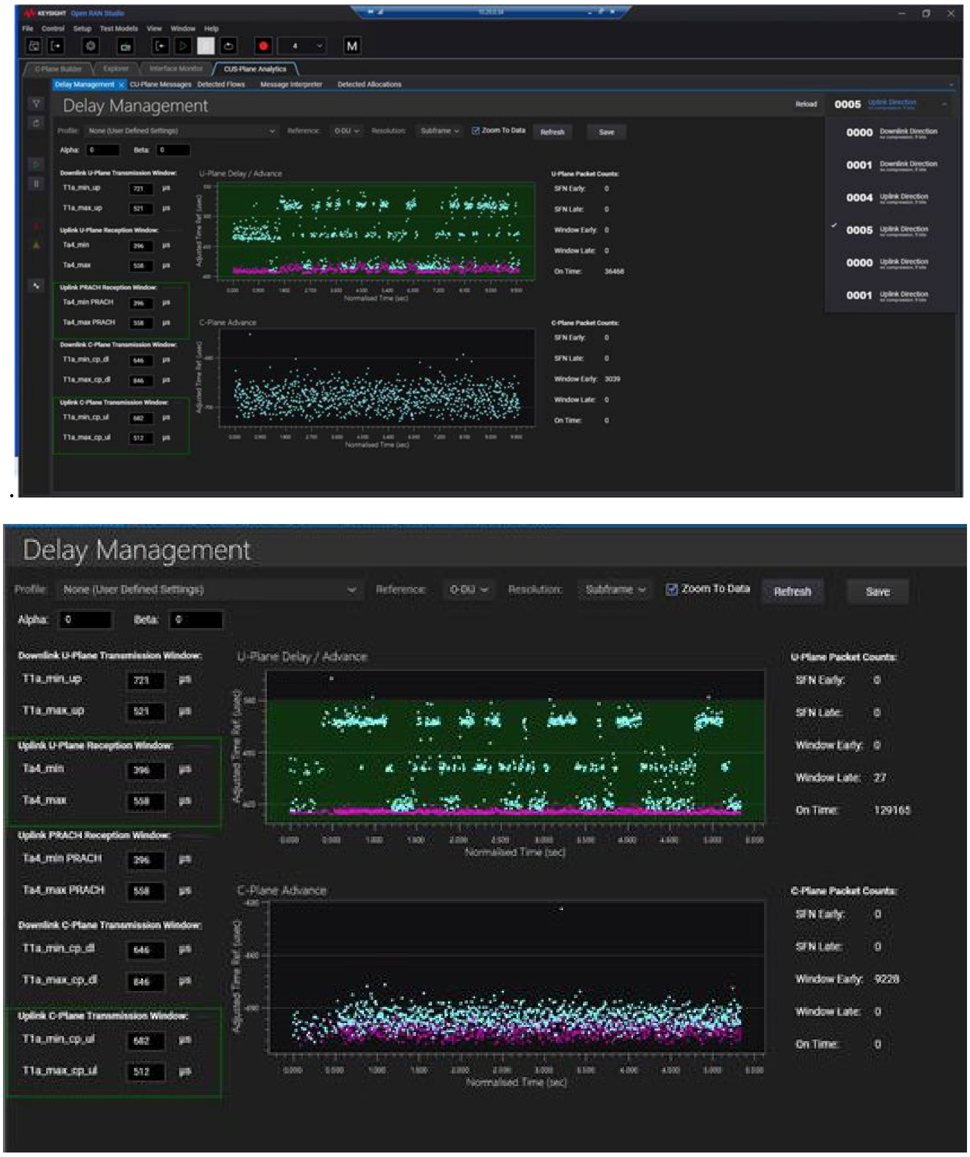


FIGURE 78. TIMING ANALYSIS ON OPEN-RAN STUDIO

3.1.1.3.3.3 E2E latency studies

Below we discuss our experiments on E2E latency evaluations.

3.1.1.3.3.3.1 Challenges in measuring latency

For DL one-way latency, RuSIM only provides histogram buckets with relatively larger intervals (e.g., [0,5 ms), [5,10 ms), etc.) or long-term averages starting from the beginning of the emulation. The coarse granularity of these latency buckets causes most latency measurements to fall within the first or second bucket across different configurations, making it indistinguishable between different configurations. As for the long-term average, considering that we are emulating multiple bursts with different throughput values in one test, this may mask the difference over averaging. A potential way to alleviate this limitation is to calculate latency over short intervals (e.g., per second) using the available long-term average and the total number of packets. Figure 79 below presents an example of this method, which clearly reveals the impact of O-FH delay on DL latency.

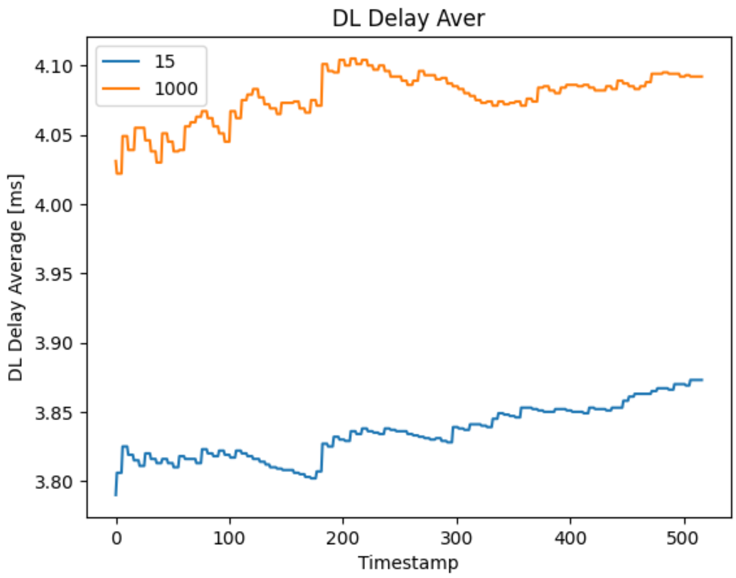


FIGURE 79. ESTIMATED INSTANT LATENCY FOR O-FH DELAYS OF 15 NS (BLUE) AND 1 US (ORANGE)

On the contrary, UL one-way latency measurement is not natively supported in RuSIM. Instead, we can estimate it through manual packet capture and postprocessing. In the figure below, we present the calculated UL latency histogram. However, this approach requires extensive manual effort and complex timing calculations, making it impractical for our large-scale testing.

Financiado por la Unión Europea NextGenerationEU
Plan de Recuperación, Transformación y Resiliencia
UNICO I+D
GOBIERNO DE ESPAÑA
cttc Centre Tecnològic de Telecomunicacions de Catalunya

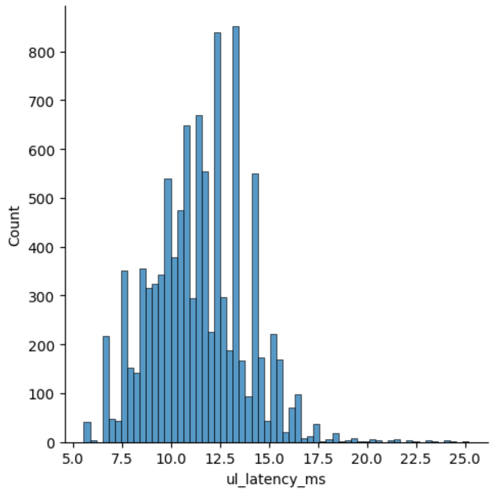


FIGURE 80. HISTOGRAM OF THE UL LATENCY THAT CAN BE OBTAINED MANUALLY VIA PACKET CAPTURES

Therefore, considering all these factors, we only consider the DL latency with a simplified test setup to ensure reliable measurement. In each test with a specific configuration, instead of running a series of bursts with varying throughput values, we conduct a single 5-minute DL throughput burst. This approach allows the long-term average to effectively reflect DL latency under this configuration.

Another important consideration is that the one-way latency measurement relies on synchronization between the UE (RuSIM) and the Core (CoreSIM). RuSIM is synchronized via PTP to the PTP switch, while CoreSIM is synchronized via NTP to the PTP switch. To assess synchronization accuracy, we conduct a one-day drift monitoring for both methods.

From Figure 81, we see that NTP synchronization between CoreSIM and the switch can exhibit drifts of up to 1 ms. In contrast, PTP synchronization for RuSIM generally maintains nanosecond-level accuracy, with only occasional spikes. This means that the one-way latency measurement may be subject to inaccuracies on the order of 1 ms due to the limitations in NTP synchronization.

Financiado por la Unión Europea NextGenerationEU
Plan de Recuperación, Transformación y Resiliencia
UNICO I+D
GOBIERNO DE ESPAÑA
cttc Centre Tecnològic de Telecomunicacions de Catalunya

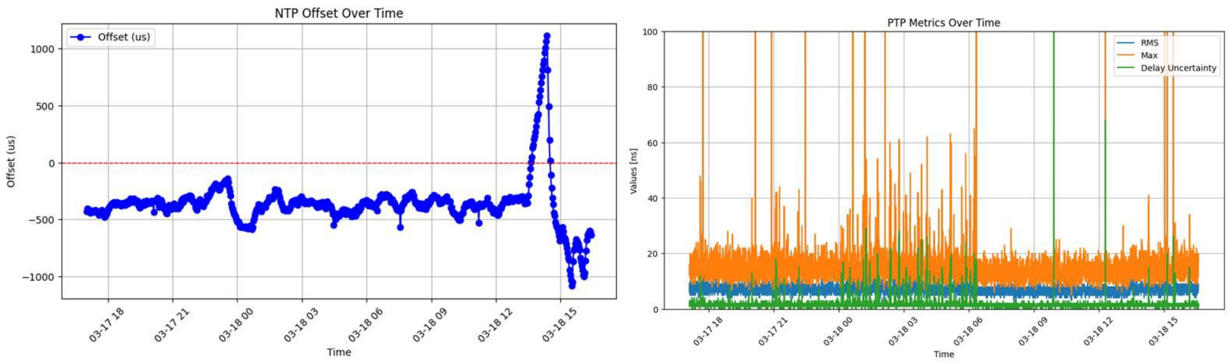


FIGURE 81. TIMING OFFSETS ACHIEVED WITH NTP (LEFT) AND PTP (RIGHT), MEASURED TO ASSESS THE ACCURACY OF THE LATENCY ESTIMATES

3.1.1.3.3.3.2 Interaction of max_proc_delay, data-rates, SNR, O-FH delay

In this section, we evaluate the effect of several scenario parameters on the end-to-end latency that is observed in DL user traffic. The parameters that we explore are described next:

- UL SNR on the air interface: we set the UL SNR to 40, 10, and 0 dB (corresponding to AWGN settings of -40, -10, and 0, respectively) in order to explore how the quality of the UEs uplink connection may impact the system's latency. Degradation on the UL air interface will lead to the loss of ACKs from the UEs, triggering unnecessary retransmissions and increasing latency as a result.
- O-FH delay: with NE3, we emulate O-FHs that introduce different delays in the transmission of packets from O-DU to O-RU and vice versa. In particular, we evaluate O-FH delays of 15, 118, 336, and 1000 us.
- Application-layer DL throughput: we also explore how the overall DL throughput measured at the application layer affects the average DL latency.
- The time allowed for the O-DU to process DL packets: srsRAN O-DU has a configurable parameter that determines the maximum time that the O-DU is allowed to spend processing a certain DL packet before it is forwarded to the O-RU. This parameter, `max_proc_delay`, determines the number of slots prior to the transmission of a packet over the air that the O-DU can start processing the packet. Increasing values of this parameter increase the E2E latency, but settings too small may lead to the O-DU dropping packets in situations of high computational load, as it will not have enough time to carry out all the necessary operations.
- The number of UEs active in the considered cell, which may impact the latency related to signalling for the different users.

The results of different tests sweeping over the above parameters are presented in Figure 82 to Figure 87, which report the average downlink latency observed under different configurations.

Without exhaustively describing each of the figures, we note the following overall observations from the obtained results:

- DL latency is degraded as the UL SNR decreases. This is likely due to packet losses in the UL control information that is related to DL traffic.
- Generally, an increasing number of UEs concurrently accessing the cell leads to an increase in latency. Similarly, a higher DL throughput increases the DL latency as well. Hence, unsurprisingly, the end-to-end latency is negatively impacted by the increasing system load.
- DL latency is increased by larger settings of the O-DU processing time, as expected. However, care has to be taken when trying to lower the value of `max_proc_delay`. Low settings for this parameter can only be successfully used whenever the O-FH does not introduce a large delay; otherwise, the O-DU needs to be provided sufficient processing time to compensate for the O-FH propagation delay. Similarly, larger number of UEs imposes tougher processing tasks for the O-DU (e.g., the scheduler operations), and thus more conservative values of the `max_proc_delay` parameter need to be used in those conditions.

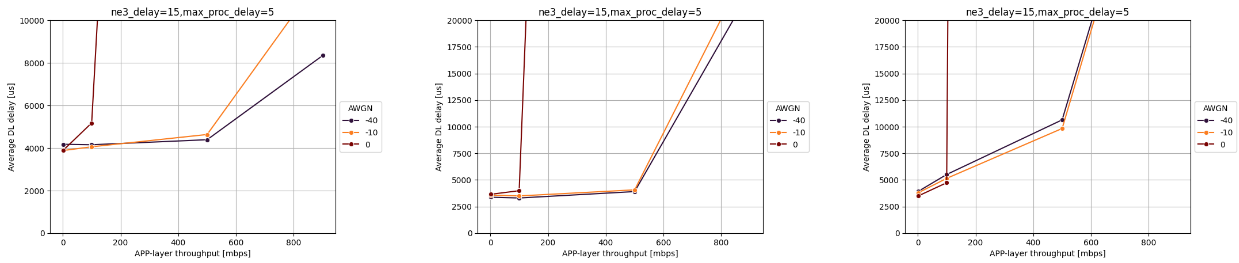


FIGURE 82. AVERAGE DOWNLINK DELAY AS A FUNCTION OF THROUGHPUT AND SNR, FOR DIFFERENT NUMBER OF UES: 1 (LEFT), 10 (CENTER), AND 50 (RIGHT)

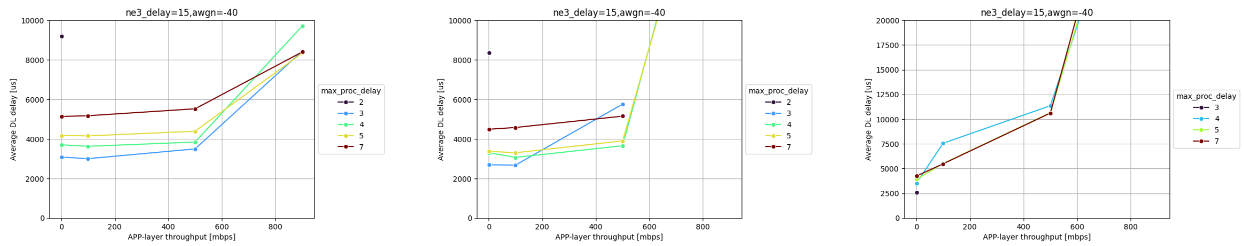


FIGURE 83. AVERAGE DOWNLINK DELAY AS A FUNCTION OF THROUGHPUT AND O-DU PROCESSING TIME, FOR DIFFERENT NUMBER OF UES: 1 (LEFT), 10 (CENTER), AND 50 (RIGHT)

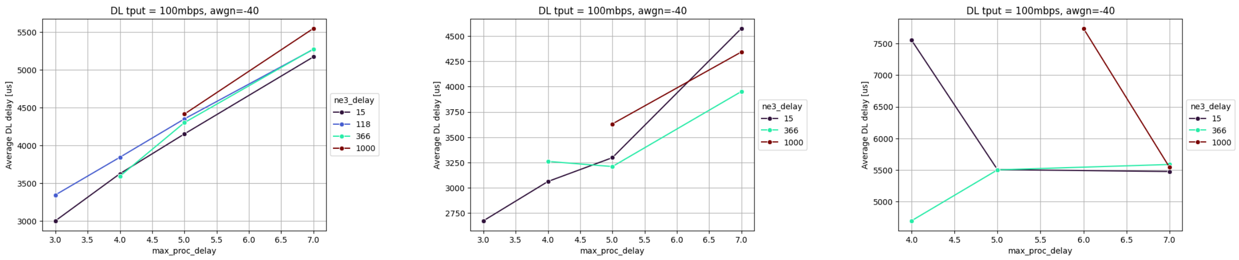


FIGURE 84. AVERAGE DOWNLINK DELAY AS A FUNCTION OF O-DU PROCESSING TIME AND O-FH DELAY, FOR DIFFERENT NUMBER OF UES: 1 (LEFT), 10 (CENTER), AND 50 (RIGHT)

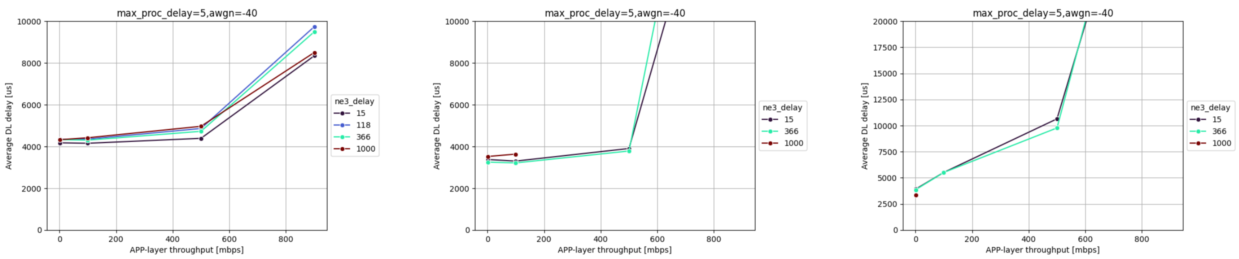


FIGURE 85. AVERAGE DOWNLINK DELAY AS A FUNCTION OF THROUGHPUT AND O-FH DELAY, FOR DIFFERENT NUMBER OF UES: 1 (LEFT), 10 (CENTER), AND 50 (RIGHT)

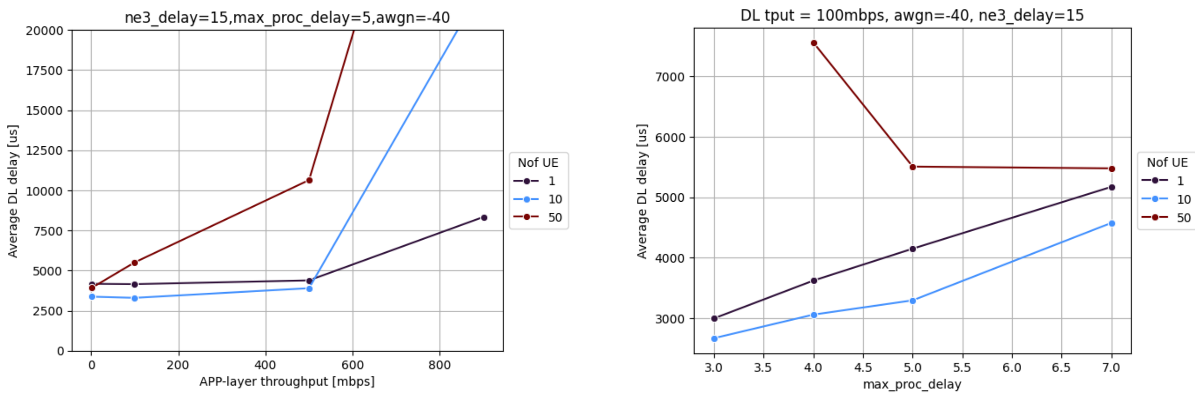


FIGURE 86. AVERAGE DOWNLINK DELAY AS A FUNCTION OF THE NUMBER OF UES AND DL THROUGHPUT (LEFT) AND O-DU PROCESSING TIME (RIGHT)

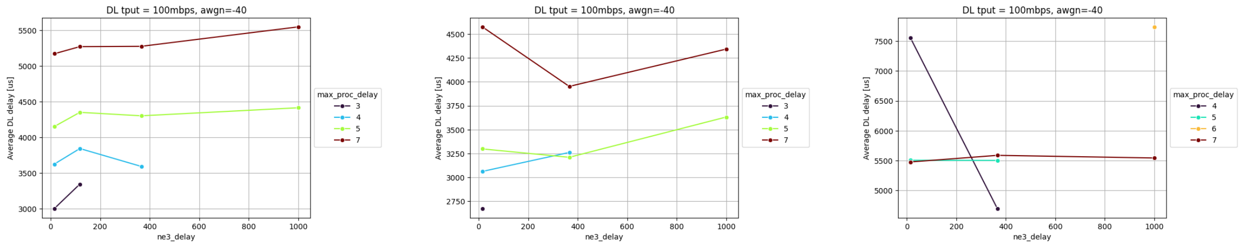


FIGURE 87. AVERAGE DOWNLINK DELAY AS A FUNCTION OF O-FH DELAY AND O-DU PROCESSING TIME, FOR DIFFERENT NUMBER OF UES: 1 (LEFT), 10 (CENTER), AND 50 (RIGHT)

3.1.1.3.3.4 Computational Performance of O-DU/CU server

During exploratory tests, we run several Linux perf tools on the server hosting the O-DU/CU to monitor performance metrics such as CPU and memory usage. However, we observe that the execution of these tools can also interfere the packet timing analysis.

3.1.1.3.3.4.1 Tests with / without perf running

The resulting figures, such as Figure 88 and Figure 89, indicate that running perf tools consistently leads to an increased ratio of late packets compared to tests where no additional processes are active, across all O-FH configurations. An initial explanation is that running perf introduces additional workload on the CPU, which slows down the O-DU/CU processes and causes more late packets. A reduction in the ratio of early packets is also observed when perf is running. However, considering the very small absolute number of early packets, this conclusion may not be statistically significant.

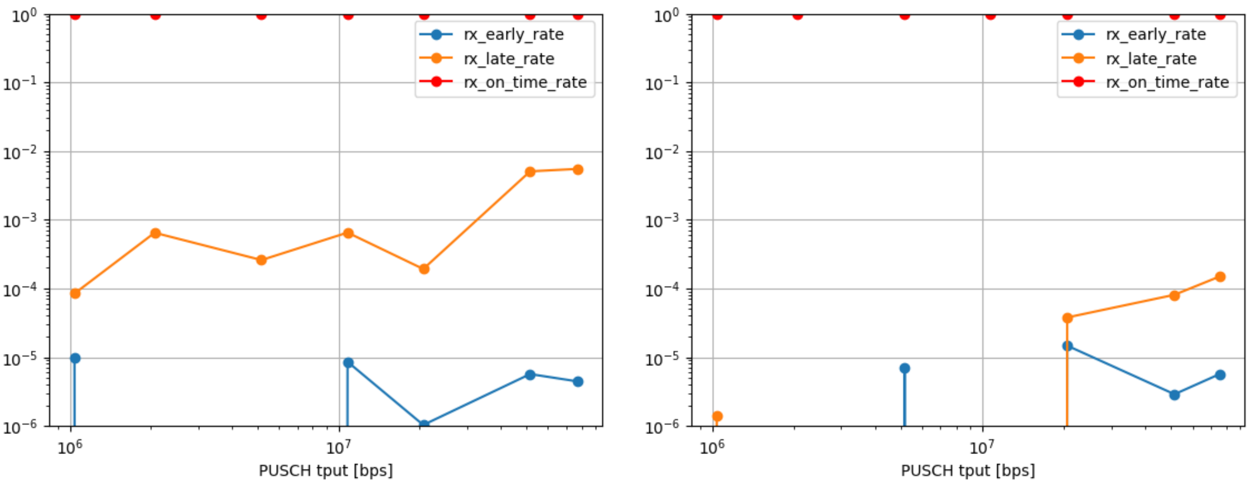


FIGURE 88. MEASURED RATE OF EARLY, LATE AND ON-TIME UL U-PLANE PACKETS IN THE HL3-LONG MID DELAY SCENARIO AND 78 MBPS OF UL TRAFFIC WHEN CONCURRENTLY RUNNING PERF (LEFT) AND WITHOUT IT (RIGHT)

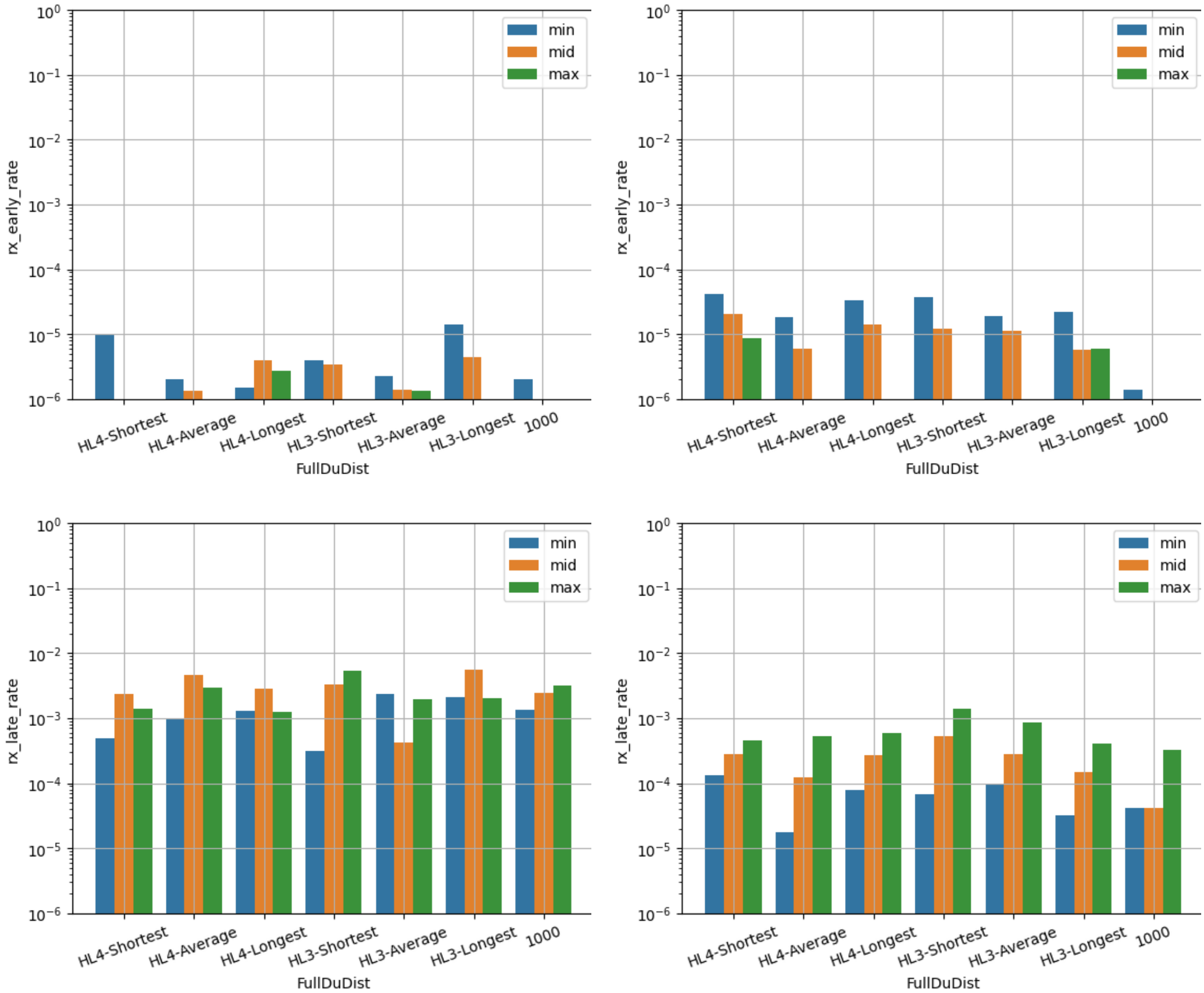


FIGURE 89. UL U-PLANE PACKET TIMING FOR MAX UL THROUGHPUTPUT (77 MBPS) WHEN CONCURRENTLY RUNNING PERF (LEFT) AND WITHOUT IT (RIGHT)

To further validate this observation and explanation, we conduct the same tests while explicitly introducing CPU workload by running the stress-ng command during the emulation. The results, shown in Figure 90, indicate that by introducing 20% CPU load on all CPU cores, there is a significant increase in the late packets detected by the O-DU. Thus, we conclude that the computing resources of the hosting server play a key role in the O-DU/CU packet timing.

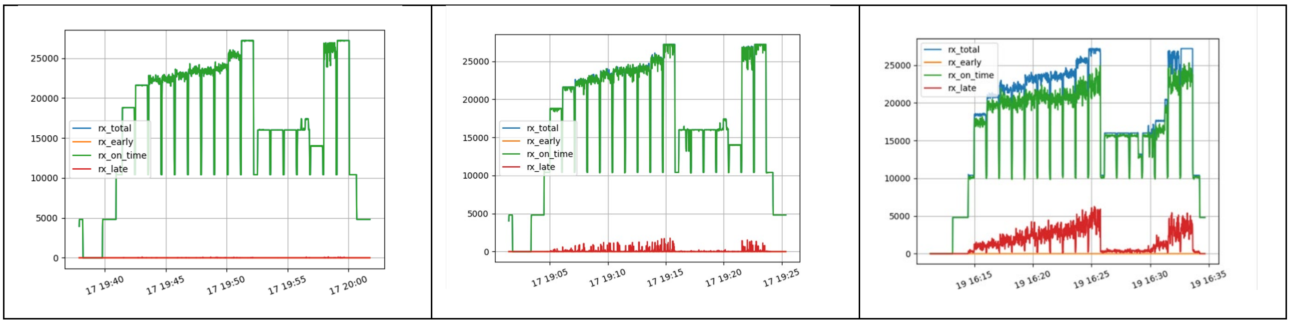


FIGURE 90. UL U-PLANE PACKET TIMING FOR MAX UL THROUGHPTUPUT (77 MBPS) WHEN CONCURRENTLY RUNNING PERF (LEFT) WITHOUT IT (CENTER) AND WHEN INTRODUCING A 20% CPU LOAD VIA STRESS-NG (RIGHT)

3.1.1.3.3.5 O-FH limits

3.1.1.3.3.5.1 Stretching O-FH delay to the limit

By default, the *max_proc_delay* parameter in the SRS O-DU/CU software is set to 5 slot intervals. As shown in previous sections, reducing the value of this parameter can lead to smaller latency. However, doing so also reduces the time margin for packet processing at the O-DU/CU, increasing the risk of failure in scenarios with longer O-FH delays. The figure below illustrates that, for a specific configuration, there exists a threshold on the O-FH delay: as long as the delay is below this threshold, all tests complete successfully, but once it exceeds this value, tests will consistently fail.

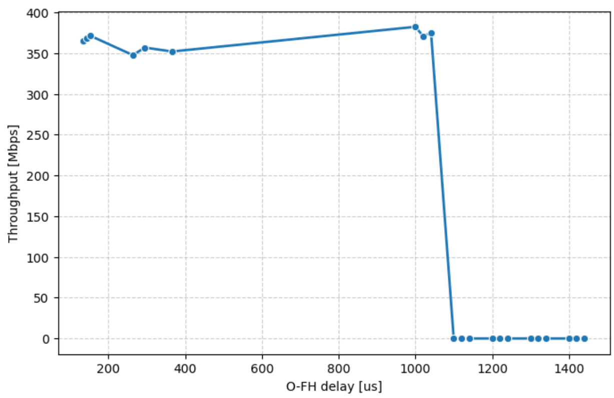


FIGURE 91. AWGN=0, MAX_PROC_DELAY=5, TESTS WITH OFH DELAY LARGER THAN 1100 US ALL FAILED

Therefore, to evaluate this trade-off, we conduct a series of tests to examine the maximum O-FH delay that the system can tolerate when applying different *max_proc_delay* values. Figure 92 shows whether a successful connection (marked as a green dot) was achieved for a given O-FH delay (y-axis, in us), or whether the connection was unsuccessful (marked with a red dot). A clear correlation between the *max_proc_delay* value and the failure threshold for O-FH delay. Obviously, reducing

Financiado por la Unión Europea NextGenerationEU
Plan de Recuperación, Transformación y Resiliencia
UNICO I+D
GOBIERNO DE ESPAÑA
cttc Centre Tecnològic de Telecomunicacions de Catalunya

max_proc_delay lowers the threshold on the O-FH delay that the system can work without failure. This finding aligns with theoretical expectations, as both a decreased *max_proc_delay* and an increased O-FH delay puts more stringent constraints on processing time, making the system more prone to failure. Thus, we can see a trade-off emerges between smaller latency and longer achievable O-FH distance.

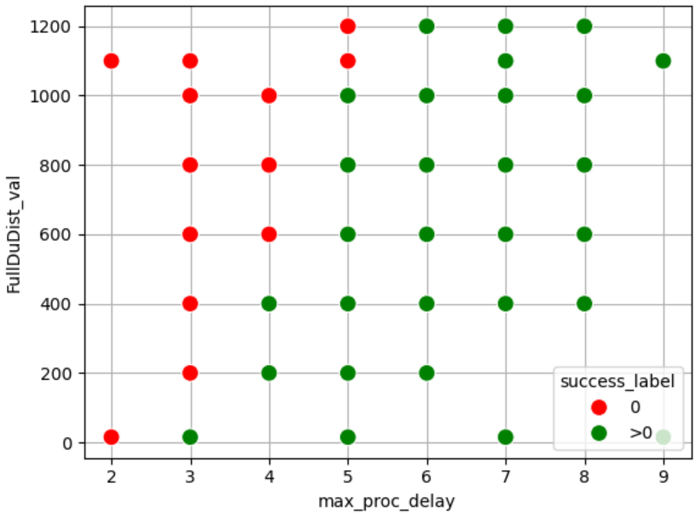


FIGURE 92. RELATIONSHIP BETWEEN MAX_PROC_DELAY AND MAX ACHIEVABLE O-FH DELAY

3.1.1.3.3.5.2 Effects of max_proc_delay, SNR, data-rate, etc.

To extend the previous analysis, we conduct similar tests with different data rates, SNR levels, and numbers of UEs. We can draw the conclusions from the below Figure 93 and Figure 94: increasing the throughput, number of UEs, or air-interface noise level tends to reduce the maximum O-FH delay under which the system can successfully complete the test. In other words, higher throughput, more UEs, or lower SNR values lead to a shift of the O-FH delay threshold toward smaller values. These conclusions align with the expectations, as the increase in all the three factors implies higher computational workload at the O-DU/CU, thus requiring a larger processing time margin. Then to meet this constraint, the system can only tolerate shorter O-FH delays.

FIGURE 93. SUCCESSFUL AND UNSUCCESSFUL CONNECTIONS UNDER DIFFERENT O-FH DELAYS, DL DATA RATES, UL SNRS, AND SETTINGS FOR THE O-DU PROCESSING TIME. THE FIGURE COMPARES THE RESULTS OBTAINED WITH 1 UE VS THOSE OBTAINED WITH 10 UES IN THE NETWORK

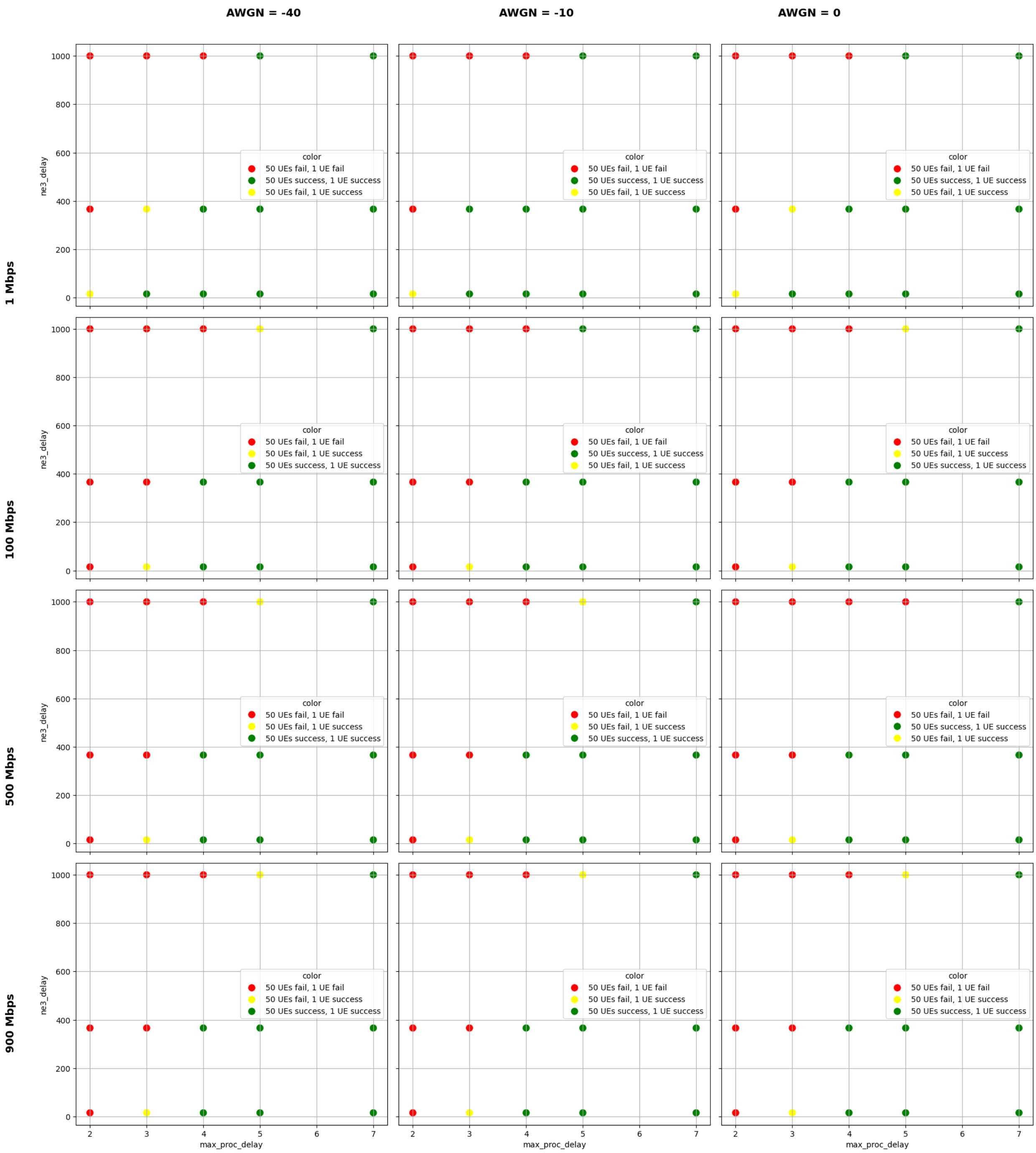


FIGURE 94. SUCCESSFUL AND UNSUCCESSFUL CONNECTIONS UNDER DIFFERENT O-FH DELAYS, DL DATA RATES, UL SNRS, AND SETTINGS FOR THE O-DU PROCESSING TIME. THE FIGURE COMPARES THE RESULTS OBTAINED WITH 1 UE VS THOSE OBTAINED WITH 50 UES IN THE NETWORK

3.1.1.3.4 Conclusions

To conclude, the experimental validation in UC2-PoC1 has shown that the concept of O-DU pooling at centralized locations is indeed feasible with the current O-RAN O-FH specifications, provided that the relevant parameters of the O-DU are appropriately adjusted. From the testbed implementation of the system, we can draw the following conclusions:

- It is crucial to adjust the O-DU TX and RX windows in order to reflect the O-RU timing and the packet delay and jitter introduced by the O-FH. With careful adjustment of these parameters, good end-to-end performance can be attained for the centralization distances targeted by this PoC.
- The E2E latency achieved by a system with a long O-FH is affected by multiple factors: number of registered UEs, computational capabilities of the servers running the RAN functions, SNR at the air interface, etc. Adjusting the time available for processing at the O-DU can be used to optimize the E2E latency, but conservative settings should be prioritized to avoid connectivity problems when the system is under high load conditions – both high traffic loads and high computational loads.
- The results indicate that, with our current testbed emulation of the considered disaggregated system, O-FH delays of up to 1 millisecond can be dealt with if the timing parameters are appropriately adjusted.

Nonetheless, we remark as well that our experimental validation is based on a simplified representation of the system, where the data-rates in the air interface and through the O-FH have to necessarily be scaled down in order to fit the hardware limitations present in a lab setup. While these results are encouraging, corroboration of them through a pilot implementation in a live, commercial network would be the natural next step.

3.2 UC3: RAN Control Architectures and Smart Algorithms

3.2.1 PoC1: Data-Driven RAN optimization for NPN deployments

3.2.1.1 Introduction

This PoC focuses on data-driven optimization of Radio Access Network (RAN) within non-public network (NPN) deployments, leveraging new capabilities in 5G network management. Enhanced gNBs now support traffic optimization and prioritization based on predefined network slices. The PoC successfully introduced machine learning-based exposure APIs that reveal Network Digital Platform capabilities for optimizing radio resources in gNBs, developed as part of UC1/PoC1 and UC2/PoC2.

Financiado por la Unión Europea NextGenerationEU
Plan de Recuperación, Transformación y Resiliencia
UNICO I+D
GOBIERNO DE ESPAÑA
cttc Centre Tecnològic de Telecomunicacions de Catalunya

3.2.1.2 System Design

The PoC evolved through various phases, starting with a knowledge-based approach, progressing to an experience-based approach, and eventually incorporating an AI/ML closed-loop algorithm[12]. This AI-based agent was designed to consider resource efficiency while meeting service level agreements (SLAs). Autonomous operations, administration, and management (OAM) mechanisms effectively monitored self-integration status and the SLA of the NPN 5G system, employing online learning to optimize resource utilization.

Integrating AI/ML techniques into Beyond 5G (B5G) and 6G networks significantly enhanced network performance and resource management. This was achieved through intelligent allocation and optimization of resources using real-time data analysis and predictive modelling. The approach for this PoC included:

- Developing a theoretical model based on domain knowledge.
- Creating an empirical model through network performance monitoring and resource management analysis.
- Applying advanced machine learning techniques to enhance the model.
- Training the AI model with actual network usage patterns and performance data.

The PoC demonstrated and evolved through these phases, enabling an NPN 5G system to support a wide range of configurations for 5G connectivity services and flexible deployment of application services. Activities included static monitoring and optimal deployment of 5G Core Network Functions (CNFs) and RAN configuration to ensure improved network performance and efficient resource utilization. Smart, autonomous resource configuration in NPN deployments involved ML-powered optimization for better network performance and resource use, facilitated through an embedded analytics engine in the network digital twin system, enhancing operational efficiency and optimization.

The Network Exposure Function (NEF) was deployed at the 5TONIC lab, providing KPI data and configuration capabilities of selected NPNs to non-trusted Application Functions (AFs) developed in the 6G BLUR project. Key RAN configuration capabilities affecting network behaviour included transmission bandwidth, Time Division Duplex (TDD) pattern, transmission power, and modulation order. Utilizing NEF exposure capabilities, this PoC explored NPN optimization scenarios with external AFs actively participating in the optimization process.

[12] Here what we mean by a closed-loop is that NDT platform continuously trains ML models to learns from newer use case validations to refine prediction capabilities.

Financiado por la Unión Europea NextGenerationEU
Plan de Recuperación, Transformación y Resiliencia
UNICO I+D
GOBIERNO DE ESPAÑA
cttc Centre Tecnològic de Telecomunicacions de Catalunya

3.2.1.3 Implementation

Building upon the results of UC1 PoC1 and UC2 PoC2, the NPN was deployed at CTTC facilities in Barcelona and integrated with the Public Network (advanced 5G Core) in the 5Tonic Lab, Leganes, Madrid. As part of this PoC, we illustrate:

- How an enterprise can use data-driven recommendations and configurations to reduce resource usage, and how analytics APIs can help us validate that data-driven recommendations meet the target KPIs.
- When introducing new use cases, how predictions can be used to assess KPI fulfilment versus demand requirements.
- Lastly, how the enterprise can use data-driven recommendations and configurations to find the best configurations to reduce resource usage.

3.2.1.4 Results

Based on the Analytical API, enterprises can identify regular traffic patterns, determine actual traffic requirements, validate the initial recommendation, and identify potential optimizations. Once they identify potential optimizations, they can use data-driven recommendations to optimize NPN configurations based on resource optimization criteria.

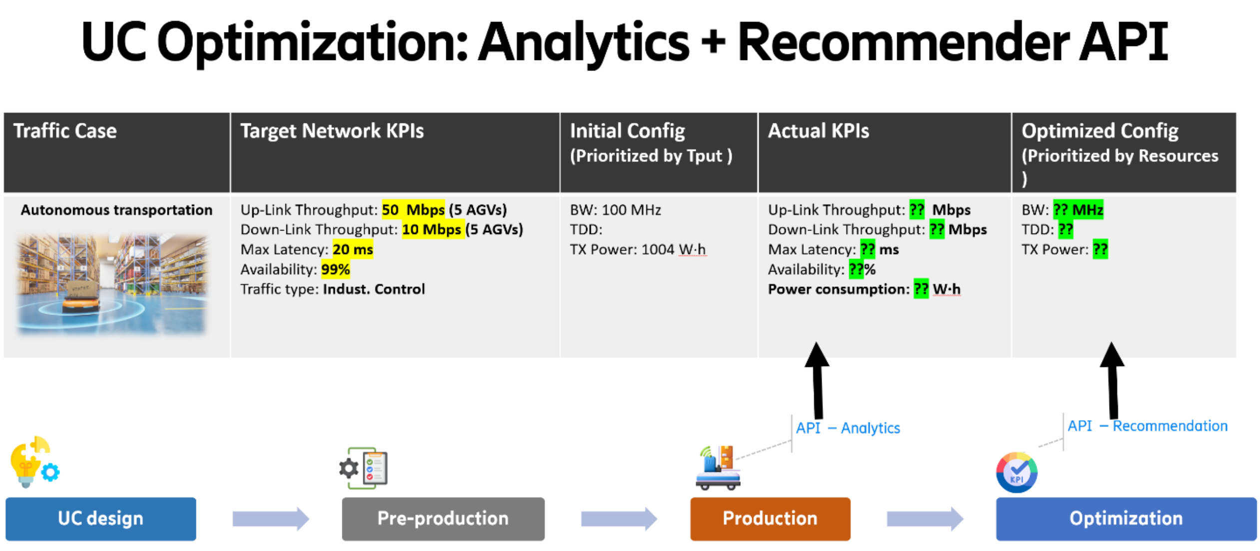


FIGURE 95. OPTIMIZATION PROCESS

We will demonstrate how to use the Design Recommendations API for an enterprise to deploy new connectivity use cases. Further, we will illustrate the use of Analytics APIs to identify regular traffic

Financiado por la Unión Europea NextGenerationEU
Plan de Recuperación, Transformación y Resiliencia
UNICO I+D
GOBIERNO DE ESPAÑA
cttc Centre Tecnològic de Telecomunicacions de Catalunya

patterns, determine actual traffic requirements, validate the initial recommendation, and identify potential optimizations.

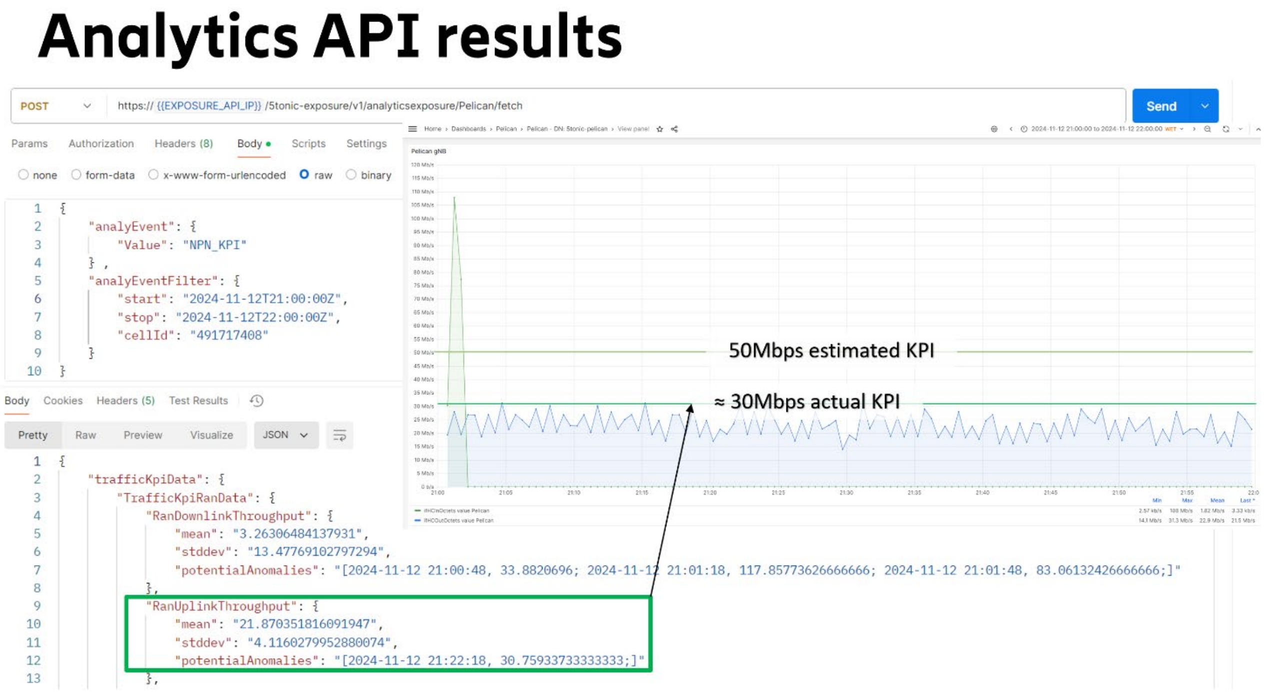


FIGURE 96. ANALYTICS RESPONSE

Using results from the Analytical API, we can identify regular traffic patterns, determine actual traffic requirements, validate the initial recommendation, and identify potential optimizations.

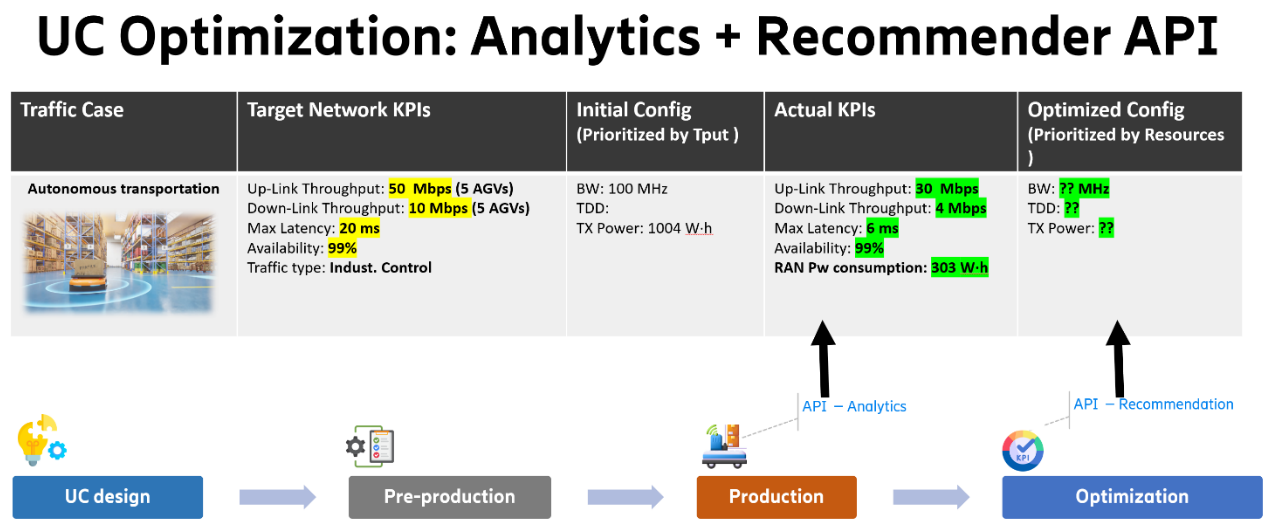
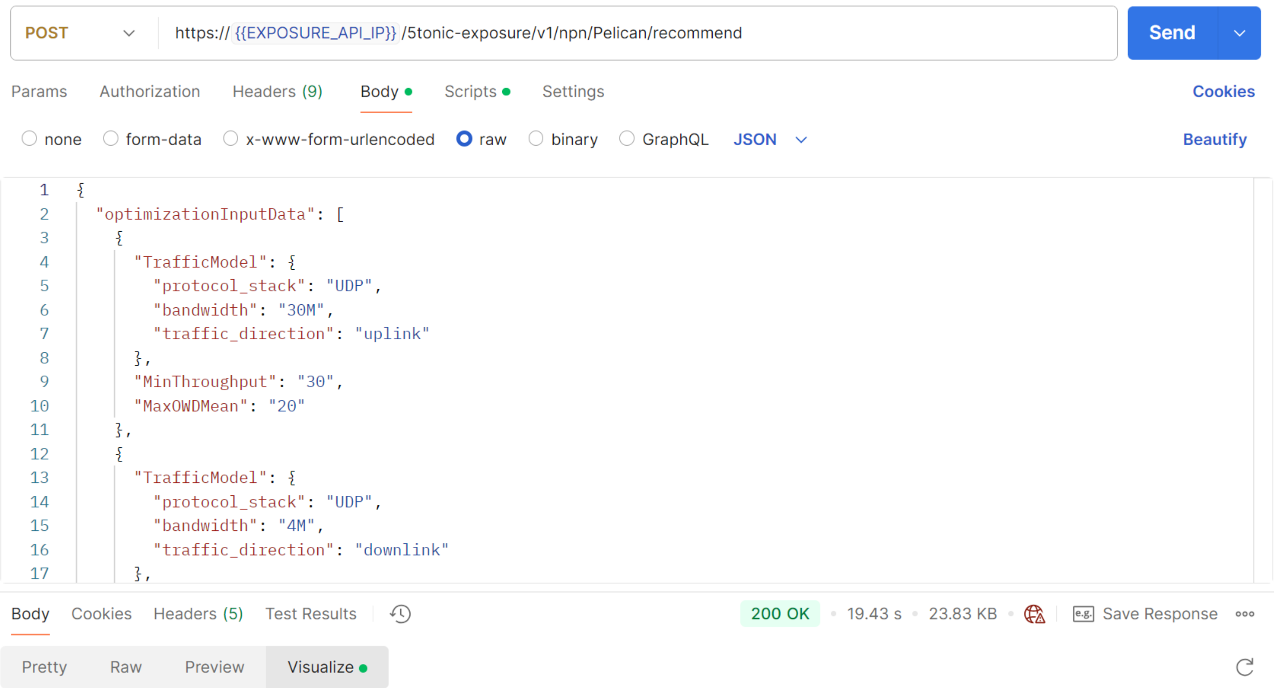


FIGURE 97. OPTIMIZATION RECOMMENDATION

Financiado por la Unión Europea NextGenerationEU
Plan de Recuperación, Transformación y Resiliencia
UNICO I+D
GOBIERNO DE ESPAÑA
cttc Centre Tecnològic de Telecomunicacions de Catalunya

Based on regular traffic pattern analysis, we can ask the Digital Twin platform's data-driven recommendation API to look for alternative configurations focused on saving resources, such as bandwidth, without deteriorating KPIs. We query the NDT platform's data-driven recommendation API, providing the use case traffic model and expected traffic KPI requirements, and request the platform to suggest the most suitable configurations. The Design Recommendation API response suggests seven optimal configurations that fulfill the target KPI requirements, recommending the first-priority configuration.



Recommend API Inputs: KPI's requirements

Recommendation Criteria	Uplink KPI Tput (Mbps)	Uplink KPI OWD (ms)	Uplink KPI Power (W)	Downlink KPI Tput (Mbps)	Downlink KPI OWD (ms)	Downlink KPI Power (W)
RESOURCE_OPTIMIZATION	30	20	N/A	4	20	N/A

Recommend API Outputs: Configurations sorted by RESOURCE_OPTIMIZATION

Priority	Bandwidth (MHz)	TDD Pattern	Uplink Max ach. Tput (Mbps)	Uplink Pred. OWD (ms)	Uplink Pred. Power Con. (W)	Downlink Max ach. Tput (Mbps)	Downlink Pred. OWD (ms)	Downlink Pred. Power Con. (W)
1	40	DDDSUDDSUU(10:2:2)	38.794	13.561	167.737	320.773	6.949	263.859
2	60	DDDSU(10:2:2)	42.455	11.041	181.185	563.655	5.530	277.887
3	60	DDDSUDDSUU(10:2:2)	60.617	11.904	186.287	489.633	5.470	279.888
4	80	DDDSU(10:2:2)	57.651	11.461	179.756	744.465	5.514	243.742
5	80	DDDSUDDSUU(10:2:2)	81.972	11.558	185.867	633.796	5.993	276.553
6	100	DDDSU(10:2:2)	73.013	13.093	174.616	833.945	5.781	286.482
7	100	DDDSUDDSUU(10:2:2)	97.081	12.296	176.783	833.713	5.536	262.274

First recommendation: Configuration details

Priority	Sector Bandwidth (MHz)	TDD Pattern	Tx Power (W)	Uplink Modulation	Uplink MIMO	Downlink Modulation	Downlink MIMO	Cell Name	Sector Name	5G NR Band	Spectrum Usage Technique	Antennas
1	40	DDDSUDDSUU(10:2:2)	1004	64-QAM	1	256-QAM	4	PELICAN_5G_DOT-1	S1	n77 , n78	TDD	RD4479B78L

FIGURE 98. RESOURCE OPTIMIZATION RECOMMENDATION

The enterprise can use the Configuration API to apply the recommended configurations to the NPN system and utilize the Configuration Retrieval API to validate the applied configuration.

Financiado por la Unión Europea
NextGenerationEU
Plan de Recuperación, Transformación y Resiliencia
UNICO I+D
GOBIERNO DE ESPAÑA
cttc
Centre Tecnològic de Telecomunicacions de Catalunya

Let's illustrate when the enterprise introduces new use cases along with the existing ones, how they can use the NDT platform's predictions to assess KPI fulfilment versus demand requirements.

UC Design: Prediction + Recommender API

Traffic Case	Purpose	Macro Context	Traffic Model	5G Private Network KPIs
Autonomous transportation 	**Transportation of goods and materials** within factory and warehouse.	Users: **5 AGVs** Concurrent: **YES** Time: **Day Shift**	UL (1) + DL (1) Protocol: UDP UL Tput UDP: **30 Mbps** DL Tput UDP: **4 Mbps** Traffic type: Periodic Availability: 99%	UL Tput UDP : **40 Mbps (AGVs + Drone)** DL Tput UDP: **5 Mbps (AGVs + Drone)** Max OWD: **20 msec.** Min Availability: 99%
Inventory management 	**Drone based inventory** management, Inventory audit, Item search, Buffer stock control to increase the inventory accuracy, decrease labor costs and minimize dangerous tasks for the workforce Uplink: HD Video streaming, Positioning, Downlink: Operation control, Collision control	Users: **1 Drone** Concurrent: **NO** Time: **Day Shift**	UL (1) + DL (1) Protocol: UDP UL Tput UDP: **10 Mbps** DL Tput UDP: **1 Mbps** Availability: 99% Traffic type: HD Video streaming, Positioning, Downlink: Operation control, Collision control	

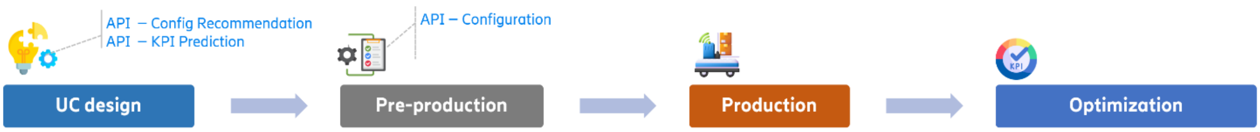


FIGURE 99. PREDICTION RECOMMENDATION

We query the NDT platform's Prediction API, providing the NPN configuration and use case traffic model requirements, and request the NDT platform to accurately predict the achievable KPIs. Based on the prediction results, we may find that the achievable KPIs do not match the required target KPIs.

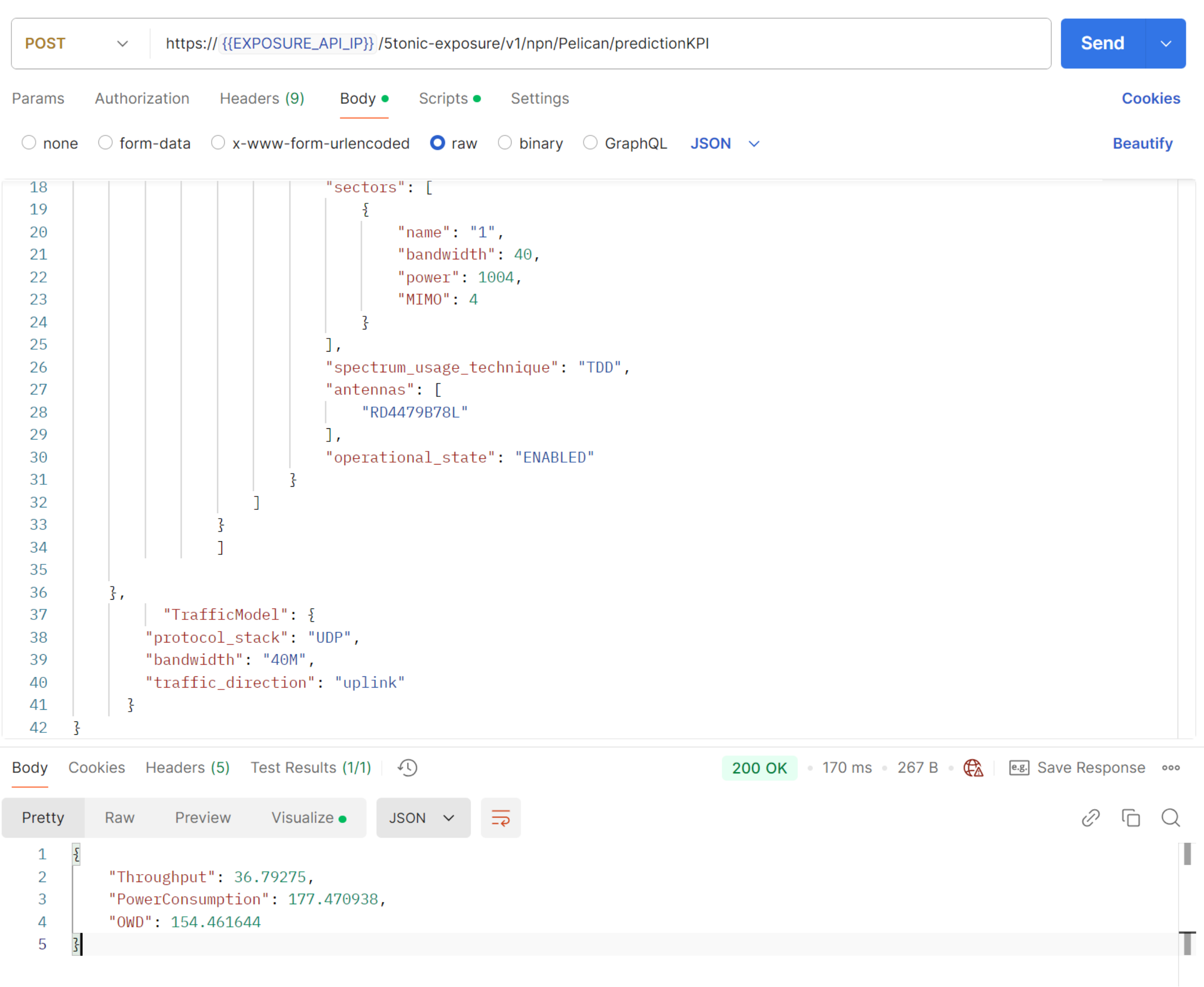
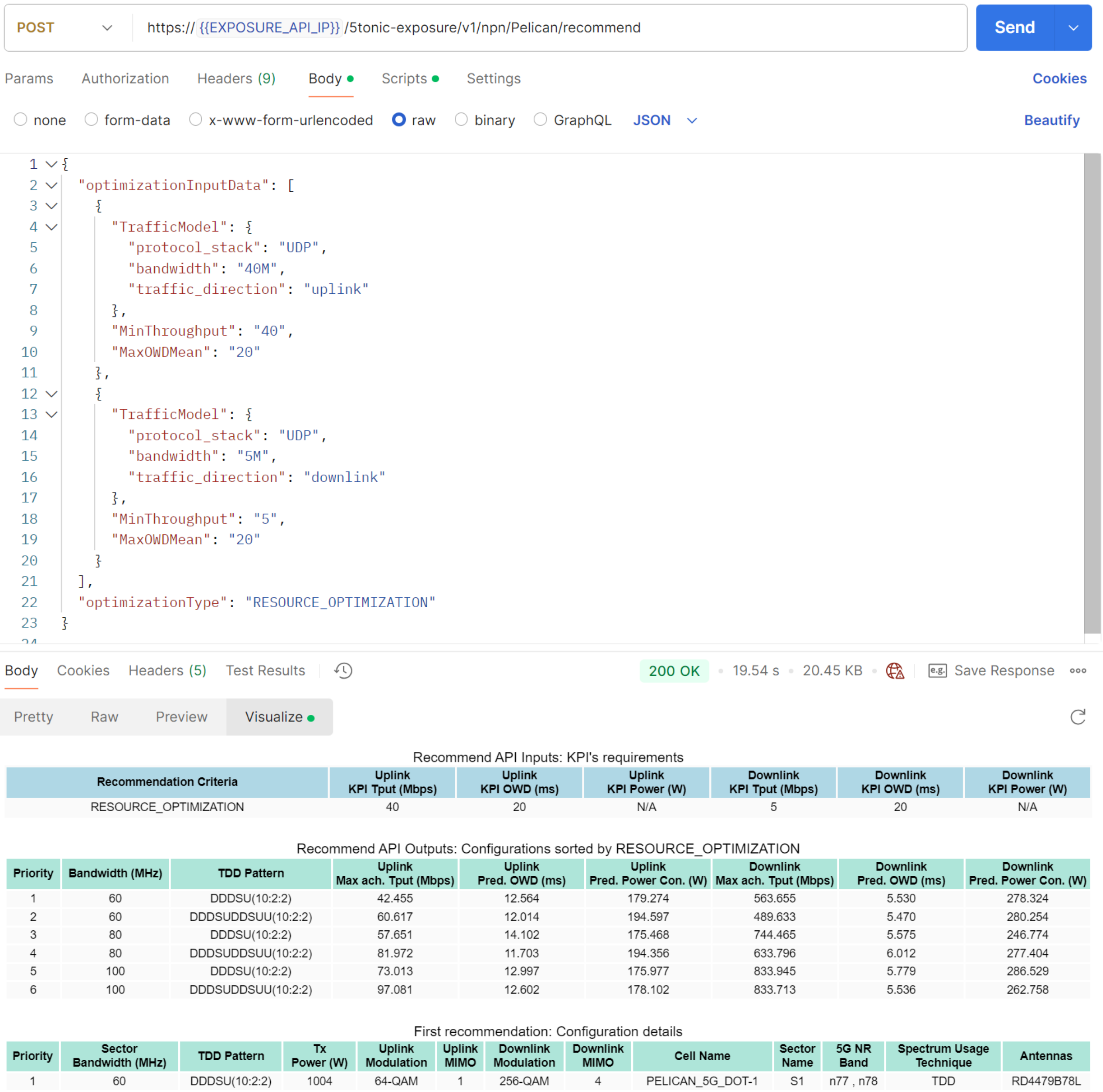


FIGURE 100. PREDICTION API RESPONSE

We query the NDT platform's Design Recommendation API, providing the use case traffic model and expected traffic KPI requirements, and request the platform to suggest the most suitable configurations. The Design Recommendation API response suggests six optimal configurations that fulfil the target KPI requirements, recommending the first-priority configuration.




```json
{
  "optimizationInputData": [
    {
      "TrafficModel": {
        "protocol_stack": "UDP",
        "bandwidth": "40M",
        "traffic_direction": "uplink"
      },
      "MinThroughput": "40",
      "MaxOWDMean": "20"
    },
    {
      "TrafficModel": {
        "protocol_stack": "UDP",
        "bandwidth": "5M",
        "traffic_direction": "downlink"
      },
      "MinThroughput": "5",
      "MaxOWDMean": "20"
    }
  ],
  "optimizationType": "RESOURCE_OPTIMIZATION"
}
```


Recommend API Inputs: KPI's requirements

Recommendation Criteria	Uplink KPI Tput (Mbps)	Uplink KPI OWD (ms)	Uplink KPI Power (W)	Downlink KPI Tput (Mbps)	Downlink KPI OWD (ms)	Downlink KPI Power (W)
RESOURCE_OPTIMIZATION	40	20	N/A	5	20	N/A

Recommend API Outputs: Configurations sorted by RESOURCE_OPTIMIZATION

Priority	Bandwidth (MHz)	TDD Pattern	Uplink Max ach. Tput (Mbps)	Uplink Pred. OWD (ms)	Uplink Pred. Power Con. (W)	Downlink Max ach. Tput (Mbps)	Downlink Pred. OWD (ms)	Downlink Pred. Power Con. (W)
1	60	DDDSU(10:2:2)	42.455	12.564	179.274	563.655	5.530	278.324
2	60	DDDSUDDSUU(10:2:2)	60.617	12.014	194.597	489.633	5.470	280.254
3	80	DDDSU(10:2:2)	57.651	14.102	175.468	744.465	5.575	246.774
4	80	DDDSUDDSUU(10:2:2)	81.972	11.703	194.356	633.796	6.012	277.404
5	100	DDDSU(10:2:2)	73.013	12.997	175.977	833.945	5.779	286.529
6	100	DDDSUDDSUU(10:2:2)	97.081	12.602	178.102	833.713	5.536	262.758

First recommendation: Configuration details

Priority	Sector Bandwidth (MHz)	TDD Pattern	Tx Power (W)	Uplink Modulation	Uplink MIMO	Downlink Modulation	Downlink MIMO	Cell Name	Sector Name	5G NR Band	Spectrum Usage Technique	Antennas
1	60	DDDSU(10:2:2)	1004	64-QAM	1	256-QAM	4	PELICAN_5G_DOT-1	S1	n77 , n78	TDD	RD4479B78L

FIGURE 101. DESIGN RECOMMENDATION FOR RESOURCE OPTIMIZATION

The enterprise can use the Configuration API to apply the recommended configurations to the NPN system and utilize the Configuration Retrieval API to validate the applied configuration.

In summary, by utilizing the Network Digital Twin platform's exposure APIs, we illustrated how enterprises can effectively deploy connectivity use cases with design-level recommendations tailored

Financiado por la Unión Europea NextGenerationEU
Plan de Recuperación, Transformación y Resiliencia
UNICO I+D
GOBIERNO DE ESPAÑA
cttc Centre Tecnològic de Telecomunicacions de Catalunya

to their KPIs. We explored how analytical APIs provide insights into traffic patterns and KPI requirements, enabling data-driven decisions that optimize resource usage. We also discussed how predictions help assess new use cases against demand and performance expectations. This PoC demonstrated the NDT platform's capabilities for data-driven prediction, recommendation and optimization, as well as the methodology's applicability to various real-world use cases, as detailed in section 3.9/5.1 on the AI/ML Platform. The 'Final Models Selection' subsection highlights the superior performance of the chosen models in predictions and recommendations across different scenarios.

The following Key Performance Indicators (KPIs) were successfully implemented in the PoC (Refer to the KPI in UC2 PoC2 of 6GBLUR JOINT E4 deliverable section): network performance KPIs (Downlink throughput per user, Uplink throughput per user, Downlink cell capacity, Uplink cell capacity, Application-level latency, Reliability, Coverage, and Service availability), Network Resource Utilization KPIs (Spectrum Utilization, Bandwidth Utilization, Resource Block Utilization, and Resource utilization). The Network Digital Twin platform provided a framework to collect these KPIs and expose them via Network Exposure Analytics APIs, as previously demonstrated. All listed KPIs were collected, logged, and made accessible through the Network Exposure Analytics API, except for "KPI Reliability." Although reliability was not explicitly provided in the API output, it was derived from the primary KPI of latency by enriching the NEF information of the average latency value with a standard deviation measurement. These KPIs are to monitor, measure, and optimize the efficient use of network performance & resources within 5G NPN system

Experimentation and real-world use case datasets were collected using the Network Exposure Analytics API. A demonstration video of this PoC was presented to partners and it is publicly available[13].

For the development and implementation of this model, we have made use of a series of complementary technologies that we have integrated, allowing us to offer massive customization at a reasonable cost. These complementary technologies include:

- Zero Touch mechanism
- Distributed Architecture
- Edge computing
- Machine Learning
- Network Digital Twin
- Network Slicing
- Network Exposure APIs

[13] Available online at: https://gitlab.cttc.es/mdi-6gblur/smart/-/tree/main/PoCs/UC3PoC1%20Data-Driven%20RAN%20optimization%20for%20NPN?ref_type=heads

Financiado por la Unión Europea NextGenerationEU
Plan de Recuperación, Transformación y Resiliencia
UNICO I+D
GOBIERNO DE ESPAÑA
cttc Centre Tecnològic de Telecomunicacions de Catalunya

3.2.1.5 Conclusions

This Data-Driven RAN optimization for NPN deployments PoC, by utilizing the Network Digital Twin platform, successfully demonstrated that a Communication Service Provider or enterprise customer can use AI/ML-based network exposure APIs to easily design, deploy, and optimize 5G connectivity use cases. These APIs offer exposure capabilities that go far beyond the standard, extending well beyond a NEF. We successfully demonstrated how this rich set of ML-based exposure APIs can be used at different stages of the innovation cycle for connectivity use cases.

3.2.2 PoC2: Optimization of DU-RU VNFs

3.2.2.1 Introduction

This PoC implements only a subset of the broader design principles described in SMART K1.4 Section 2.3, and it does so in a pragmatic and domain-specific manner. From the design stage in SMART K1.4, the PoC concretely adopts the idea of a distributed control structure spanning non-RT, near-RT, and real-time loops, and demonstrates how these layers interact through clearly defined information flows. It also operationalises the SMART K1.4 emphasis on multi-component coordination by combining a non-RT forecasting function with multiple near-RT control modules that are responsible for distinct but interdependent tasks, in this case, RAN slicing decisions and FPGA resource reconfiguration. The PoC similarly reflects the SMART K1.4 view that heterogeneous compute elements must be jointly managed, as shown by the explicit use of FPGA accelerators and the mechanisms that allow control decisions to influence hardware behaviour. Another aspect reproduced from the design is the separation between predictive optimisation and rapid reactive control, where the LSTM-based traffic forecasting, the slicing xApp, and the FPGA reconfiguration xApp each fulfil specialised roles that map directly to the envisioned layered functionality. The PoC also embodies the concern raised in SMART K1.4 for system reliability through its alert-based safety mechanism, in which immediate hazard signals bypass higher-level controllers and trigger deterministic actions; this matches the design’s notion that certain conditions must override optimisation goals and propagate directly to the lowest-latency control path. Finally, the PoC mirrors the SMART K1.4 requirement for observability and state exchange, as it establishes a data flow from non-RT measurements to near-RT decisions and onward to the hardware layer, reflecting the multi-stage information pipeline anticipated in the design document. Although simplified, these elements collectively demonstrate how the originally proposed distributed, multi-layer control logic can be realised in a concrete, running system.

Financiado por la Unión Europea NextGenerationEU
Plan de Recuperación, Transformación y Resiliencia
UNICO I+D
GOBIERNO DE ESPAÑA
cttc Centre Tecnològic de Telecomunicacions de Catalunya

The micro-orchestration framework developed in this PoC is designed as a hierarchical system comprising three functional layers that operate in concert to balance energy efficiency with performance and safety requirements. The foundational layer is the predictive planning component, implemented as an emulated rApp that hosts a LSTM neural network. This rApp forecasts aggregated PRB utilization across the gNB by consuming Key Performance Measurement (KPM) metrics and generating probabilistic forecasts that drive proactive resource allocation decisions.

Above this predictive layer sits the reactive control layer, which comprises two orchestration applications (xApps) that consume forecasts and apply RAN control actions at different temporal scales. The slicing xApp dynamically provisions PRBs to connected User Equipment (UE) across premium priority and secondary slices with QoS differentiation, enabling flexible allocation based on anticipated demand. The micro-orchestrator xApp manages FPGA SoC function lifecycle, executing scaling and migration of compute functions between processing elements based on forecasted load and predefined rule-based policies.

Overlaying both predictive and reactive layers is a safety supervision component that provides event-driven integration with external alert systems, such as smart city computer vision applications. This safety layer can override standard policies when critical hazard scenarios are detected, ensuring that safety-critical events can trigger immediate resource reallocation regardless of energy optimization objectives. A high-level representation of the micro-orchestration framework is seen in Figure 102.

Financiado por la Unión Europea NextGenerationEU
Plan de Recuperación, Transformación y Resiliencia
UNICO I+D
GOBIERNO DE ESPAÑA
cttc Centre Tecnològic de Telecomunicacions de Catalunya

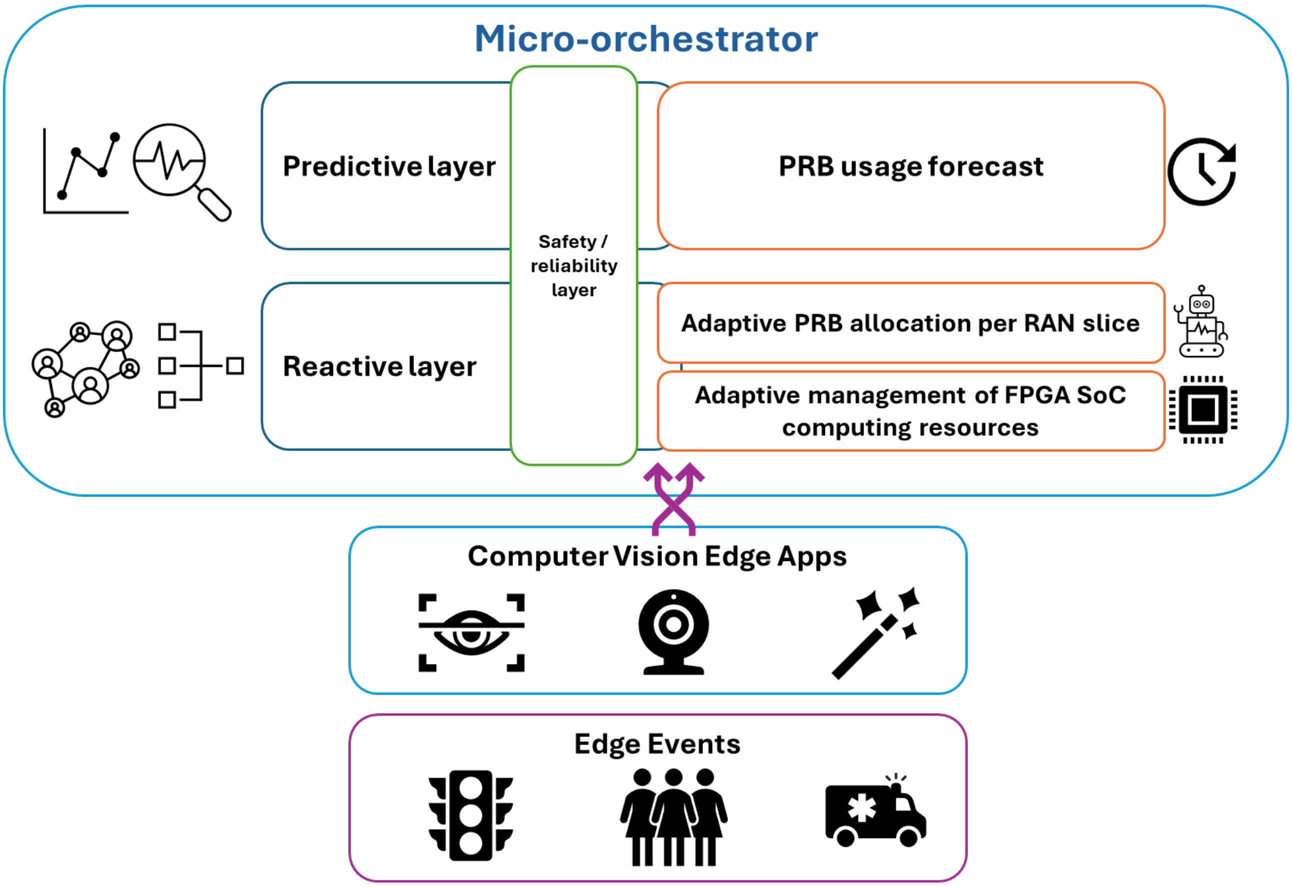


FIGURE 102. THE DIFFERENT LAYERS COMPRISING THE MICRO-ORCHESTRATION FRAMEWORK

The system is explicitly designed to operate across multiple control loop frequencies, each addressing different aspects of the resource management problem. The non-real-time layer executes at minute-to-hour granularity, where the LSTM model periodically infers new forecast sequences based on accumulated KPM data. These forecasts are then stored in a shared database accessible to downstream control functions.

The near-real-time control loop operates at RAN frame granularity, typically on the order of milliseconds. The slicing xApp reads the most recent LSTM forecasts and updates PRB allocation policies across slices. This timescale allows the system to adapt resource distribution in response to anticipated load trends while maintaining compatibility with standard RAN scheduling constraints.

The real-time control loop executes with sub-millisecond decision latency and is responsible for instantaneous FPGA SoC reconfiguration decisions. The micro-orchestrator xApp operates at this timescale, scaling function parameters or triggering migrations based on current system state and recent forecast trends.

Financiado por la Unión Europea NextGenerationEU
Plan de Recuperación, Transformación y Resiliencia
UNICO I+D
GOBIERNO DE ESPAÑA
cttc Centre Tecnològic de Telecomunicacions de Catalunya

At the highest priority is the critical real-time layer, which executes whenever external hazard alerts are generated. The safety supervisor applies deterministic hard constraints immediately upon alert reception, potentially overriding the decisions made by lower-priority control loops. This ensures that safety-critical requirements are never compromised by energy optimization or performance tuning objectives.

When the system receives both forecast-driven commands and external alert-driven commands, a clear prioritization hierarchy is applied. Real-time alerts from external systems take absolute precedence over energy-optimization objectives. This prioritization ensures that if a computer vision system detects a hazardous condition that requires low-latency signal processing, the micro-orchestrator immediately configures the FPGA SoC to prioritize responsiveness, even if this increases energy consumption. The forecast-driven policy layer operates normally during non-critical periods, allowing the system to achieve energy efficiency gains when safety is not at risk.

This hierarchical approach represents a deliberate design choice that recognizes the fundamental difference between energy minimization, which is an optimization objective, and safety compliance, which is a hard constraint. By implementing explicit prioritization, the system avoids the scenario where energy-minimization actions could inadvertently compromise safety by increasing latency or reducing processing capability.

3.2.2.2 System Design

Deployment Infrastructure and Testbed Configuration

The demonstration was deployed on CTTC's cloud-native O-RAN testbed constructed with BubbleRAN's MX-PDK solution [38]. The physical infrastructure comprises six rack-mounted servers hosting a Kubernetes cluster that manages containerized workloads. Two O-RAN-compliant RUs with antenna arrays are connected through a dedicated front-haul switch, enabling controlled radio frequency testing. An RF-shielding enclosure allows for repeatable over-the-air testing without external interference. For UE emulation, the testbed includes six Quectel 5G modems and two mobile phones that can connect to the deployed gNB and exercise realistic user traffic patterns. Figure 103 shows the hardware setup of the BubbleRAN’s testbed operating at CTTC’S ADAPT RU lab.

The software stack is built using BubbleRAN capabilities at both the radio access and core network layers. The 5G Core Network is instantiated using either OpenAirInterface or Open5GS, depending on experimental requirements. The gNB implementation uses either OpenAirInterface or srsRAN variants, allowing flexibility in radio protocol implementation. The RAN Intelligent Controller (RIC) framework is provided by FlexRIC, which natively supports xApp and rApp deployment through Kubernetes service definitions. The xApps include the KPM service module for metrics collection, a slicing service module for PRB allocation control, and the micro-orchestrator service module for FPGA SoC function management.

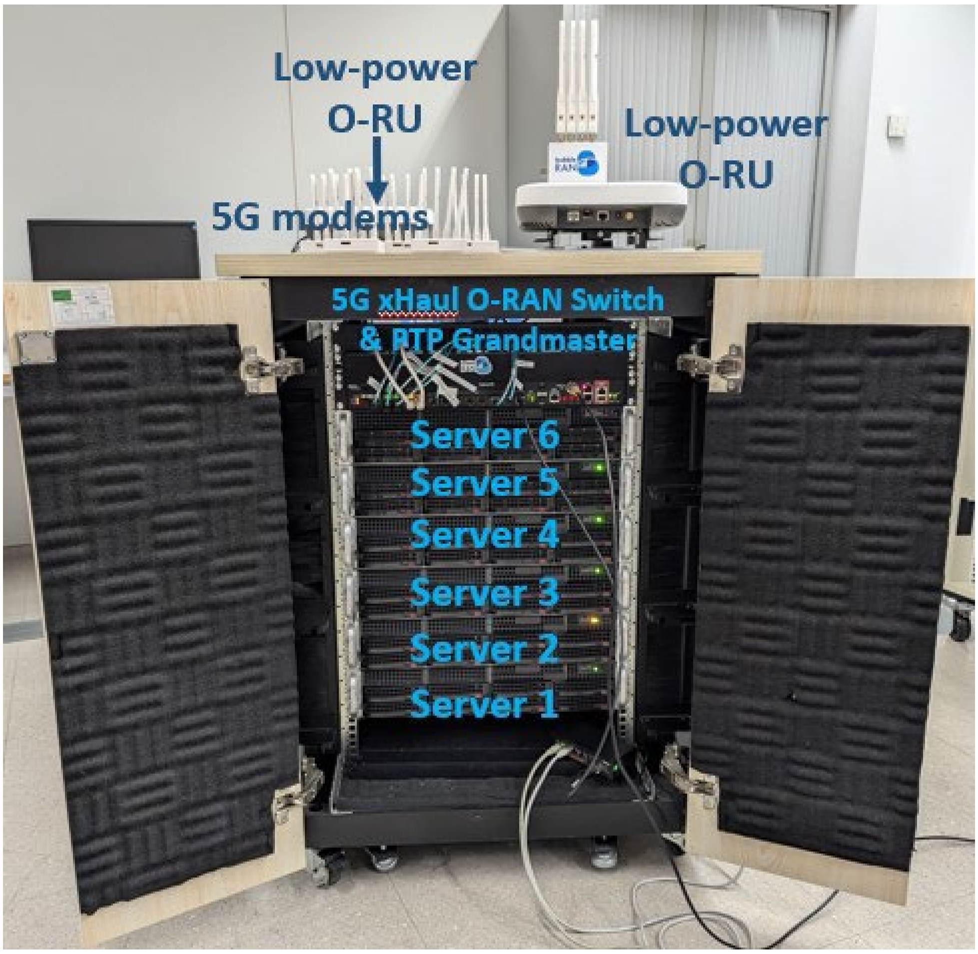


FIGURE 103. THE BUBBLERAN TESTBED

Supporting infrastructure for observability includes Prometheus for time-series metric storage and MySQL for transactional data persistence. Energy monitoring is provided by the Kepler exporter, which accesses hardware performance counters to estimate per-pod power consumption within the Kubernetes cluster. Grafana dashboards provide real-time visualization of all collected metrics and operational status. Service management and orchestration functions are performed by a non-real-time RIC instance (Odin) with a command-line interface for administrative operations.

The network topology and all system parameters were specified declaratively using YAML configuration files, enabling reproducible deployments and version control of the system configuration. The radio configuration specifies operation on band n78 with 40 MHz bandwidth and 30 kHz subcarrier spacing, appropriate for sub-6 GHz 5G deployments. The TDD configuration defines a 5 ms periodicity with 7 downlink slots, 6 downlink symbols, 2 uplink slots, and 4 uplink symbols, providing substantial downlink resource for the demonstration scenario.

The slicing configuration establishes enhanced Mobile Broadband (eMBB) service type with both IPv4 and IPv6 protocol support. Two UE terminal devices are configured, each with distinct IMSI identities and cryptographic credentials. Both terminals are initially configured to connect to the same slice but can be logically separated through the slicing xApp during operation.

The networking infrastructure includes overlay networks for internal cluster communication and external DNS resolution through public nameservers. Kubernetes services expose the gNB, 5G Core Network, FlexRIC, and supporting applications through well-defined service endpoints accessible to network functions and external monitoring tools.

The system is designed to support multiple data pathways connecting the various architectural components. KPM metrics flow from the gNB through the FlexRIC framework into both Prometheus for time-series storage and into the rApp for consumption by the LSTM forecaster. Forecasts generated by the rApp are written to a shared MySQL database, which serves as the data plane for communication between the non-real-time and near-real-time control layers.

The xApps read both KPM metrics and rApp forecasts from their respective stores and generate RAN control actions that are applied through the gNB's scheduling and slicing engines. For the micro-orchestrator xApp, control commands are sent to the FPGA SoC resource management layer, which physically reconfigures the accelerator hardware. Energy metrics collected by the Kepler exporter and application-level performance metrics are aggregated in Grafana for real-time visualization and post-analysis.

External alert sources, such as the smart city computer vision application from a parallel demonstration, integrate through a direct network socket connection established specifically to minimize latency. This direct connection bypasses the database pathway used for other data types, ensuring that critical safety alerts can reach the micro-orchestrator within milliseconds rather than incurring the latency of database roundtrips.

In Figure 104 we present a general flow diagram for the operation of the system.

Financiado por la Unión Europea NextGenerationEU
Plan de Recuperación, Transformación y Resiliencia
UNICO I+D
GOBIERNO DE ESPAÑA
cttc Centre Tecnològic de Telecomunicacions de Catalunya

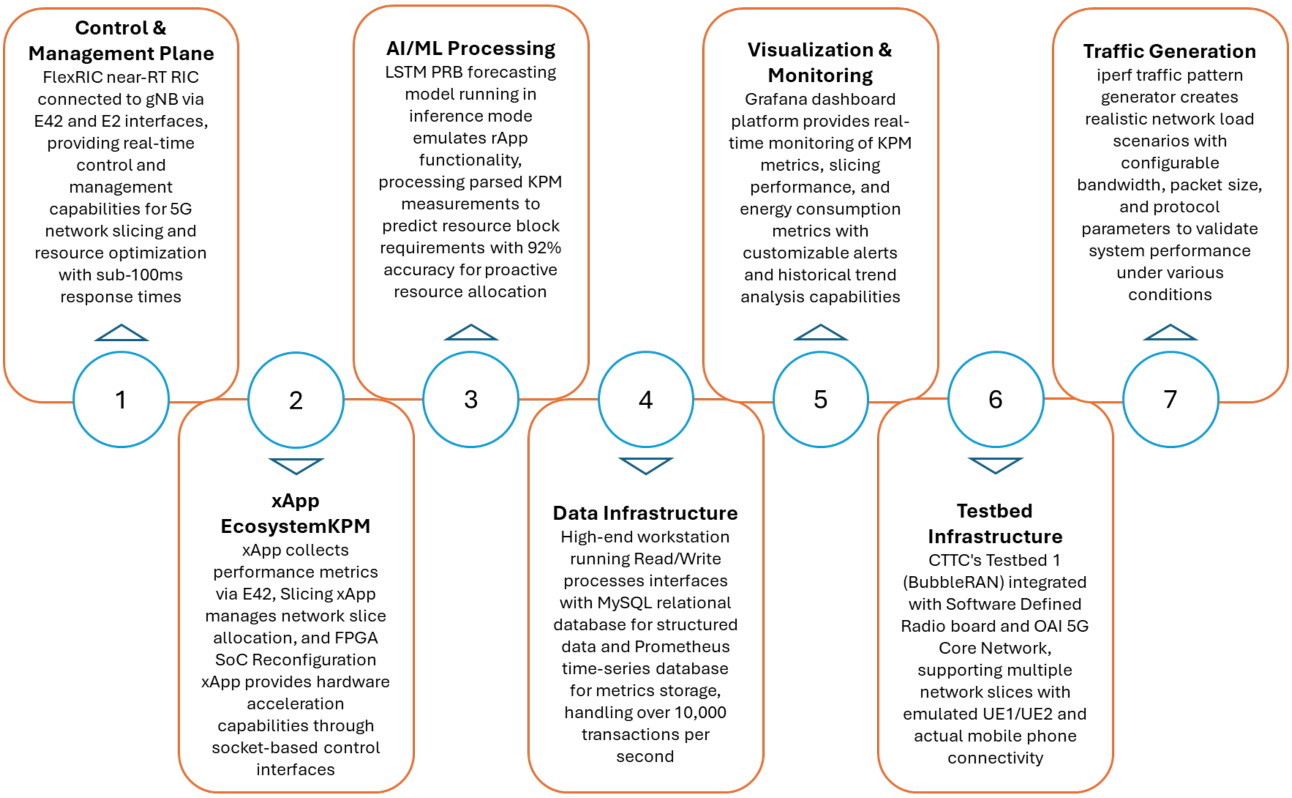


FIGURE 104. OVERVIEW OF THE SYSTEM OPERATION

3.2.2.3 Implementation: Machine Learning Forecasting and Control Policy Framework

Dataset Production and Model Training

The foundational input to the entire system is historical PRB utilization data collected over a five-week period from a bare-metal OpenAirInterface deployment. The data collection setup was configured with controlled traffic injection patterns to ensure that diverse operational scenarios were captured. The iperf traffic generation tool was used to inject flows with rates ranging from 10 to 100 Mbps, following predefined temporal profiles that simulate realistic RAN load variations including peaks, valleys, and transient surges.

The collected dataset[14] includes multivariate time-series observations at TTI granularity, capturing not only aggregate PRB utilization but also slice-specific allocation patterns, downlink and uplink load distribution, and associated system metrics including latency and throughput. This

[14] https://github.com/CTTCADAPTRU/OAI-KPM-dataset

Financiado por la Unión Europea NextGenerationEU
Plan de Recuperación, Transformación y Resiliencia
UNICO I+D
GOBIERNO DE ESPAÑA
cttc Centre Tecnològic de Telecomunicacions de Catalunya

comprehensive instrumentation ensures that the machine learning model has access to sufficient context to learn meaningful patterns.

The LSTM architecture was selected specifically for its demonstrated effectiveness in multi-step time-series forecasting and its ability to capture non-linear temporal dependencies. Stacked LSTM layers provide hierarchical feature extraction, while attention mechanisms enable the model to explicitly learn which historical timesteps are most relevant for predicting future PRB utilization. The model processes multivariate input features and generates probabilistic forecasts that include not just point predictions but also confidence intervals reflecting forecast uncertainty.

The training process used standard backpropagation with Mean Squared Error loss function and a validation split of 10 to 20 percent of the dataset to monitor generalization performance. Convergence was achieved within typical computational budgets, and visual inspection of training trajectories confirmed that the model successfully learned to predict PRB patterns with acceptable accuracy across the test horizon. The trained model weights are persisted and loaded at inference time for continuous operation within the rApp.

Rule-Based Policy Definition and Implementation

The operational policies are organized into two distinct branches, each managed by a different xApp component but coordinated through shared forecast data. The first policy branch governs slice-level PRB allocation and is executed by the slicing xApp. This component implements a control loop that reads the most recent four minutes of KPM metrics from the MySQL database to understand current allocation trends. It then retrieves the latest LSTM forecast outputs, typically the last three prediction samples, which are averaged to reveal the overall trend of anticipated PRB usage.

Based on this combined understanding of current state and forecasted trends, the slicing xApp applies a policy that determines what percentage of available PRBs should be allocated to the priority slice hosting UE1 and what percentage should be reserved for the secondary slice hosting UE2. The policy enforces a hard guarantee that the priority slice always receives sufficient resources to maintain service, while the secondary slice receives whatever capacity remains. This adaptive allocation allows the system to maintain priority slice performance during periods of overall high load while maximizing secondary slice capacity during low-load periods.

The second policy branch manages FPGA SoC function lifecycle and is executed by the micro-orchestrator xApp. This component manages both function scaling and function migration across processing elements. Function scaling involves adjusting the FFT size parameter from a discrete set of options: 8 points, 1024 points, 2048 points, or 4096 points. Smaller FFT sizes reduce computational demand and power consumption but sacrifice spectral resolution, potentially degrading signal quality. Function migration involves moving the FFT function execution from the Programmable

Logic (PL) area of the FPGA, which offers high performance but high-power consumption, to the Application Processing Unit (APU), which is an ARM A53 quad-core processor offering lower power consumption but higher latency.

The relationship between detected hazard alerts and applied reconfigurations is formalized in a mapping table. When a low hazard level is detected, the system configures the FFT to use 8 points and executes on the APU, maximizing energy efficiency. As hazard level increases to medium-low, the FFT size increases to 1024 points but remains on the APU. Medium-high hazard level triggers migration to the PL with 2048-point FFT to provide better latency and spectral resolution. Finally, high hazard level uses the full 4096-point FFT on the PL, providing maximum spectral resolution and processing capability regardless of energy cost.

Forecast-Driven Decision Making

The LSTM forecast output serves as input to both the slicing xApp and the micro-orchestrator xApp, enabling proactive decisions that anticipate future resource demands. When the forecast indicates increasing PRB utilization in the near future, the slicing xApp can pre-emptively allocate additional resources to the priority slice before the load actually arrives, ensuring smooth service continuity. Similarly, when the forecast indicates decreasing load, the micro-orchestrator can plan migrations to more power-efficient configurations in anticipation of the reduced computational demand.

The probabilistic nature of the LSTM output is explicitly leveraged through confidence intervals that quantify forecast uncertainty. During periods of high forecast confidence, the system can apply more aggressive optimization strategies. During periods of high uncertainty, the system applies conservative policies that maintain safety margins. This adaptive confidence-based control represents a principled approach to handling model uncertainty rather than treating forecasts as deterministic ground truth.

The forecast horizon H typically spans ten or more timesteps into the future, allowing the system to make planning decisions with sufficient advance notice to execute migrations and reconfiguration operations without disrupting active traffic. However, this planning horizon must be carefully balanced against forecast accuracy degradation over longer prediction windows. The system is designed to be robust to forecast errors through the reactive control layer, which can quickly adjust to deviations between predicted and actual PRB utilization.

Alert-Driven Safety Constraints

In parallel with forecast-driven proactive control, the system integrates event-driven reactive control triggered by external alerts. When a hazard alert is generated by an external smart city application, the alert is immediately transmitted to the micro-orchestrator xApp via a direct network socket

Financiado por la Unión Europea NextGenerationEU
Plan de Recuperación, Transformación y Resiliencia
UNICO I+D
GOBIERNO DE ESPAÑA
cttc Centre Tecnològic de Telecomunicacions de Catalunya

connection. This direct pathway was chosen specifically to minimize latency, as database-based communication introduced unacceptable delays for safety-critical signals.

Upon receiving an alert, the safety supervisor layer applies hard deterministic overrides to the ongoing control logic. The micro-orchestrator immediately reconfigures the FPGA SoC according to the alert-specified policy, potentially overriding the decisions that would have been made by the forecast-driven policy layer. This hierarchical control ensures that safety objectives always take precedence.

The rationale for this alert-driven override capability is grounded in the practical realities of smart city deployment. When a hazardous event is detected, such as a pedestrian unauthorized entry into a tram safety zone, the system must immediately improve signal processing capability to enable rapid emergency response. This might require migrating functions to the lower-latency PL and using higher FFT sizes for better spectral resolution, even though such decisions increase energy consumption. The temporary energy cost is justified by the safety criticality of the event.

3.2.2.4 Results

Baseline Characterization of Function Offload Performance

The first phase of operational demonstration established baseline performance characteristics for FPGA SoC function offloading. In this phase, deterministic computer vision events were used to drive function reconfigurations, enabling controlled measurement of performance metrics across different placement and scaling combinations. This controlled experimental setting provided repeatable and measurable data on the energy-latency trade-offs achievable through offloading.

Execution time measurements were collected for the FFT function operating on both the APU and PL across multiple FFT sizes. The results clearly demonstrated the latency advantage of PL placement: when executing 2048-point or 4096-point FFTs, the PL implementation achieved acceleration factors up to 15 times compared to APU execution. This substantial latency improvement is crucial for meeting the stringent timing requirements of 5G signal processing, where function execution must complete within a single slot duration.

Power consumption analysis was performed by monitoring the voltage rails serving different FPGA SoC components: the Processing System (PS) containing the APU, the Programmable Logic (PL) fabric, and the DDR memory subsystem. The measurement data revealed that migrating the FFT function from PL to APU reduced total power consumption by approximately 32.6 percent for a 2048-point FFT configuration. This power reduction arises from reduced PL fabric switching activity when executing simpler algorithms on the lower-power APU, combined with the ability to power-gate unused PL regions when functions migrate away.

Financiado por la Unión Europea NextGenerationEU
Plan de Recuperación, Transformación y Resiliencia
UNICO I+D
GOBIERNO DE ESPAÑA
cttc Centre Tecnològic de Telecomunicacions de Catalunya

An important practical finding was the validation that FFT size 2048 remains executable within 5G NR timing constraints. For the test configuration with 30 kHz subcarrier spacing and 40 MHz bandwidth, both PL and PS placements could complete 2048-point FFT processing of input samples within the allocated time budget per NR slot. This confirms that the largest FFT size in the policy table represents a feasible operating point that does not compromise radio protocol correctness.

Closed-Loop Demonstration with LSTM-Driven PRB Allocation

Following baseline characterization, the system was operated in full closed-loop configuration with active LSTM forecasting and adaptive PRB slice allocation. The demonstration scenario involved two UEs connected to the gNB, with UE1 assigned to a priority slice and UE2 assigned to a secondary slice. Traffic was injected using iperf with configurable flow rates following predefined temporal profiles that simulate realistic load variations.

Throughout the demonstration period, the LSTM model continuously consumed KPM metrics and generated updated forecasts of PRB utilization. The slicing xApp periodically read these forecasts and updated the PRB allocation policy accordingly. Real-time monitoring via Grafana dashboard provided visibility into the system behaviour. The throughput of UE1 remained stable and near its maximum achievable value, confirming that the priority slice guarantee was properly maintained throughout the test. Meanwhile, the throughput of UE2 varied in response to load variations and allocation adjustments.

The PRB allocation itself exhibited clear adaptive behaviour. During periods of peak traffic load, when the LSTM forecast indicated sustained high PRB demand, the slicing xApp allocated approximately 70 percent of available PRBs to the priority slice and 30 percent to the secondary slice. During low-load periods, when the forecast indicated reduced aggregate demand, the allocation became more balanced while still maintaining the priority guarantee. The real-time adjustment of allocation percentages was observable in the Grafana dashboard as slice configuration changed every few seconds in response to updated forecasts.

The energy footprint of the system, as exported by the Kepler exporter and visualized in Grafana, showed correlation with the micro-orchestrator's placement decisions. During periods when the forecaster indicated reduced load and the micro-orchestrator migrated functions to the lower-power APU, the overall system energy consumption decreased measurably. This energy reduction was particularly pronounced during the low-load periods in the traffic pattern, demonstrating that the energy-optimization objective was achieving practical benefits.

Figure 105 provides a screen capture of the Grafana dashboard where the adaptive PRB allocation based on the forecast provided by the LSTM rApp is demonstrated.

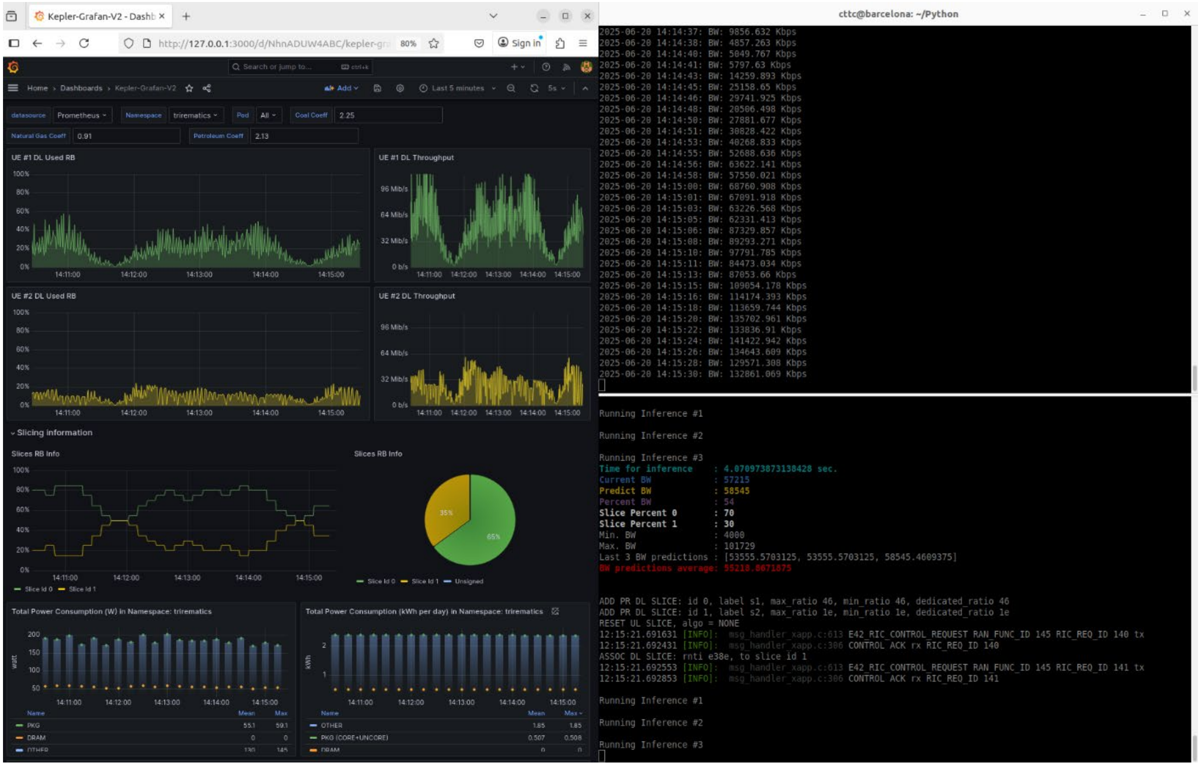


FIGURE 105. THE GRAFANA DASHBOARD SHOWING THE PRB ALLOCATION AND THROUGHPUT PER SLICE ALONG WITH ENERGY MEASUREMENTS PROVIDED BY THE KEPLER EXPORTER

Event-Driven Reconfiguration and Safety Integration

When the smart city alert system was integrated, the demonstrator exhibited reactive behaviour responsive to external events. A representative scenario involved a smart camera detecting a pedestrian entering a tramway safety zone, resulting in generation of a "Medium-High" (level 2) hazard alert. The alert generation system created the corresponding hazard classification and transmitted it to the micro-orchestrator via the direct socket connection.

Upon receiving this alert, the micro-orchestrator immediately applied the corresponding reconfiguration: the FFT function was scaled to 2048 points and migrated to the PL. This reconfiguration occurred within milliseconds of alert generation, ensuring that the system was ready to process the emergency signal with optimized latency and spectral resolution. The reconfiguration was captured in the monitoring dashboard, showing the instantaneous change in function placement and power consumption metrics.

The freed APU resources resulting from migration of the FFT function became available for allocation to other edge services or emergency applications. This resource freeing capability represents a key value proposition of the micro-orchestrator architecture: by intelligently moving functions between processing elements in response to changing demands, the system can optimize resource utilization and enable sharing of expensive edge compute resources among multiple concurrent services.

Following the alert event, the system returned to normal operation. The LSTM forecast layer continued to provide load predictions, and the forecast-driven slicing xApp adapted PRB allocation in response to the evolving traffic conditions. The hazard level eventually decreased, and the micro-orchestrator applied a new reconfiguration corresponding to the lower hazard level, potentially migrating functions back to more energy-efficient placements. This seamless transition between event-driven and forecast-driven control modes demonstrated the system's ability to handle diverse operational scenarios.

Key Performance Indicators and Measurable Outcomes

The demonstration quantified improvements across multiple performance dimensions. The energy efficiency outcome was achieved through the observed **power reduction of 32.6 percent** when migrating FFT processing from the PL to the APU. This energy saving resulted directly from the micro-orchestrator's intelligent placement decisions and represented a tangible reduction in the device's power consumption during low-load periods.

Latency reduction was demonstrated through the **15-fold acceleration** of FFT processing when using PL placement compared to APU placement. This latency improvement directly contributed to the project objective of reduced end-to-end signal processing latency and enabled the system to maintain timing compliance even when processing larger FFT sizes that improve spectral resolution.

Resource utilization efficiency was demonstrated through the adaptive PRB allocation mechanism. The slicing xApp's dynamic provisioning ensured that both priority and secondary slices received appropriate resource levels based on anticipated demand. During high-load periods, the priority slice maintained service quality while the secondary slice still received meaningful capacity. During low-load periods, the allocation became more balanced without compromising priority guarantees.

The flexibility and scalability objectives were addressed through the demonstrated ability to dynamically reconfigure FPGA SoC functions. The system successfully scaled FFT sizes, migrated functions between processing elements, and adapted allocation policies in response to both predicted load trends and real-time event signals. These dynamic capabilities would enable the architecture of this PoC to support evolving service requirements and deployment scenarios.

3.2.2.5 Conclusions and Future Outlook

This PoC documents a comprehensive demonstration of adaptive micro-orchestration for FPGA SoC-based RU functions in cloud-native 5G RAN architectures. By combining machine learning-based PRB usage forecasting with hierarchical reactive control and explicit safety prioritization, the system achieves measurable improvements in energy efficiency (32.6% power reduction), latency performance (15x acceleration), and resource utilization while maintaining safety and reliability constraints.

The end-to-end deployment on production-grade O-RAN infrastructure using standard cloud-native platforms demonstrates technical feasibility and provides a foundation for future research into increasingly sophisticated adaptive orchestration mechanisms. The demonstrated interplay between probabilistic forecasting, constrained optimization, and real-time event response establishes a replicable pattern for RAN control problems involving multiple competing objectives and uncertain environmental conditions.

Future work will focus on extending the architecture to support heterogeneous multi-RAN scenarios, integrating with standardized O-RAN SMO interfaces, and incorporating advanced machine learning techniques for continuous online learning and adaptation. The framework positions edge-deployed FPGA SoCs as dynamic, intelligent compute resources that can be seamlessly integrated into emerging 6G network architectures supporting diverse service requirements and sustainability objectives. Through these enhancements and extensions, the micro-orchestration architecture will continue to evolve to meet the increasingly sophisticated demands of next-generation telecommunications networks.

3.2.3 PoC3: Data-Driven O-RAN optimization based on xApps and rApps

3.2.3.1 Introduction

This PoC focuses on offering a closed-loop system for monitoring and controlling O-RAN networks. An adaptative monitoring schema with integrated intelligence into the O-RAN stack has been developed and tested by means of including LLMs (Large Language Models), to automate tasks and reduce human feedback.

This work builds upon several of the key concepts described previously. More specifically, the involved key concepts are:

- **SMART-K1.1 – CU and DU Implementation**: Implementation of O-RAN Central Unit (CU) and Distributed Unit (DU) components using the srsRAN software.
- **SMART-K3.1 – AI/ML Workflows in O-RAN:** Implementation of monitoring and controlling xApps and integrated in an agentic based platform.
- **JOINT-K3.1 – Monitoring Platform**: Agentic based monitoring elements of RAN components.

Financiado por la Unión Europea NextGenerationEU
Plan de Recuperación, Transformación y Resiliencia
UNICO I+D
GOBIERNO DE ESPAÑA
cttc Centre Tecnològic de Telecomunicacions de Catalunya

3.2.3.2 System design

The PoC has been developed in different phases that will be explained below.

First, with improvements to the E2 interface within the srsRAN CU/DU solution. We recall here that E2 is the O-RAN interface that allows the near-RT RIC to connect to the CU and the DU, extract metrics and control procedures and functionalities, thus enabling an ML-driven RAN. All the code is available from the srsRAN Project repository (more specific pointers are provided below).

The specific E2 components that have been developed in the context of 6G BLUR-SMART are listed next.

- The E2 agent at the RLC layer inside the DU, that allows the collection and exposition of metrics about data volume at the RLC SDU level (see https://github.com/srsran/srsRAN_Project/tree/main/lib/rlc). The implementation of new metrics, such as *DRB.RlcSduTransmittedValumeUL_Filter* and *DRB.RlcSduTransmittedValumeDL_Filter,* for measuring the UL (DL, respectively) volume of data counted at the RLC SDU level[15].
- E2SM-RC Control Style 2, Action 6, that controls the maximum number of PRBs that can be allocated to a UE[16].

Secondly, an Agentic framework for monitoring and optimizing the O-RAN environments. This system connects with various O-RAN components (whether on bare metal, Docker, or Kubernetes), centralizes logs for analysis, and employs an agent-based workflow enhanced with RAG capabilities to diagnose issues, generate documentation, and coordinate all tasks automatically. Furthermore, the proposed solution can function as an xApp that controls O-RAN components, dynamically adjusting network behaviour in response to unexpected events such as congestion in specific areas. For instance, it can increase resources to improve performance. The framework can also integrate external tools and Python libraries to produce more efficient solutions, while automating xApp deployment and making policies clearer and easier for human operators to manage.

[15] https://github.com/srsran/srsRAN_Project/blob/main/lib/e2/e2sm/e2sm_kpm/e2sm_kpm_du_meas_provider_impl.cpp

[16] https://github.com/srsran/srsRAN_Project/blob/main/lib/e2/e2sm/e2sm_rc/e2sm_rc_control_action_du_executor.cpp

Financiado por la Unión Europea NextGenerationEU
Plan de Recuperación, Transformación y Resiliencia
UNICO I+D
GOBIERNO DE ESPAÑA
cttc Centre Tecnològic de Telecomunicacions de Catalunya

Finally, the development of two xApps: a monitoring one, that collects the KPMs of E2 nodes and stores them in a database, and an actuating one, which based on the collected measurements, it suggests the necessary PRBs needed to fulfil the traffic of the UEs.

With these three main contributions, we are able to demonstrate a closed-loop in an O-RAN environment, able to monitor the state of the different components, collect performance metrics by means of using E2 interface and RIC controller, and actuate to optimize its resources, in terms of PRBs per user, all managed by an agentic framework that reduces the complexity to manage it.

3.2.3.3 Implementation

The PoC has been built as a fully containerized, modular O-RAN testbed integrating monitoring, control, and intelligent automation components. The architecture leverages open-source frameworks to emulate a realistic RAN and core environment while enabling AI-driven decision loops through a Retrieval-Augmented Generation (RAG) layer and specialized xApps.

At the infrastructure level, all major components were deployed using Docker Compose, ensuring portability, ease of orchestration, and reproducible experimentation. This containerized approach allowed each subsystem—RAN, Core, RIC, analytics, and AI—to operate independently while communicating through standardized O-RAN interfaces and network bridges.

The RAN layer was implemented using srsRAN 5G, configured in a split architecture with separate Central Unit (CU) and Distributed Unit (DU) containers. This setup provided realistic RLC, MAC, and PHY behaviour while exposing the necessary telemetry and control hooks for O-RAN integration. UEs were represented using simulated or SDR-based setups, enabling controlled traffic scenarios for testing PRB allocation strategies.

The 5G Core Network consisted of a fully dockerized Open5GS stack (AMF, SMF, UPF, AUSF, UDM, etc.), providing full 5G SA functionality. This ensured compatibility with the srsRAN CU/DU and allowed end-to-end registration, PDU session establishment, and data-plane traffic to flow through the system.

For RAN control and intelligence, the platform used the O-RAN Software Community (OSC) Near-Real-Time RIC version in srsRAN Github[17], also deployed in containers. The near-RT RIC hosted the two custom xApps developed for the PoC:

[17] https://github.com/srsran/oran-sc-ric/tree/main

Financiado por la Unión Europea NextGenerationEU
Plan de Recuperación, Transformación y Resiliencia
UNICO I+D
GOBIERNO DE ESPAÑA
cttc Centre Tecnològic de Telecomunicacions de Catalunya

1. Monitoring xApp – responsible for subscribing to E2 telemetry from the srsRAN CU. It collected PRB usage, throughput, KPI distributions, and UE-level resource metrics, pushing these measurements into the analytics pipeline.
2. PRB-Control xApp – responsible for receiving AI-generated decisions and applying them by issuing E2 control messages to the RAN, modifying PRB allocations or scheduling configurations according to the optimization logic.

To enable observability, a time-series analytics stack based on InfluxDB and Grafana was integrated. The monitoring xApp streamed metrics to InfluxDB, allowing for real-time visualization of PRB consumption, UE behaviour, and RAN state. Grafana dashboards supported debugging, KPI validation, and operator-level insights during the evaluation.

A central innovation of the PoC was the AI-driven decision layer, powered by a RAG pipeline and LLM-based agents. This component ingested the monitoring data, retrieved relevant domain knowledge (O-RAN specifications, documentation, past measurements), and synthesized actionable recommendations. The agentic system generated PRB-optimization decisions and passed them to the PRB-control xApp through well-defined APIs.

This arrangement allowed the LLM to perform high-level reasoning—interpreting RAN behaviour, identifying bottlenecks, and aligning decisions with operator policies—while delegating the low-level actuation to the O-RAN-compliant xApps inside the near-RT RIC.

Overall, the implementation demonstrated an end-to-end pipeline where RAN telemetry → analytics → AI reasoning → RIC-driven control → PRB reconfiguration operates cohesively in a fully open, containerized environment. The modularity of the components enabled rapid testing, isolation of subsystem behaviour, and seamless integration of intelligence into an O-RAN ecosystem.

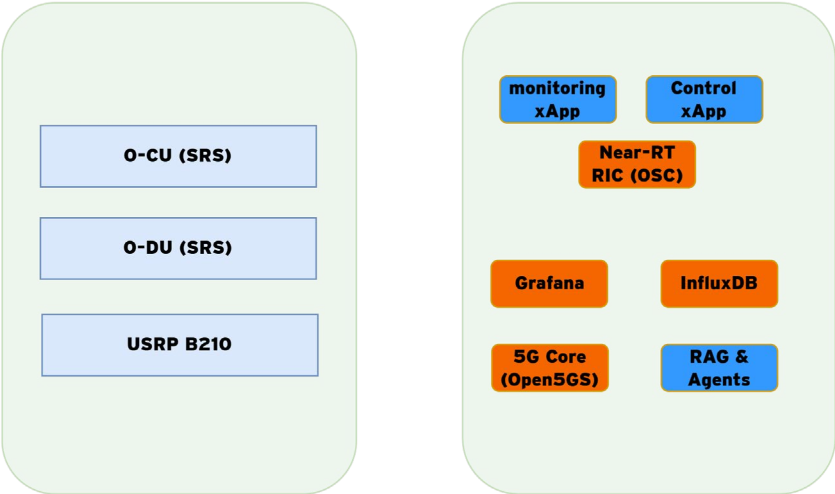


FIGURE 106. DIFFERENT ELEMENTS USED FOR THE UC3POC3

3.2.3.4 Results

In order to demonstrate the feasibility of controlling the assigned PRBs to a given UE through the Agentic framework, Figure 107 illustrates the request issued within the RAG framework to modify PRB allocation. Figure 108 then shows how the PRBs monitored by the monitoring xApp increase in accordance with the actions taken by the control xApp.

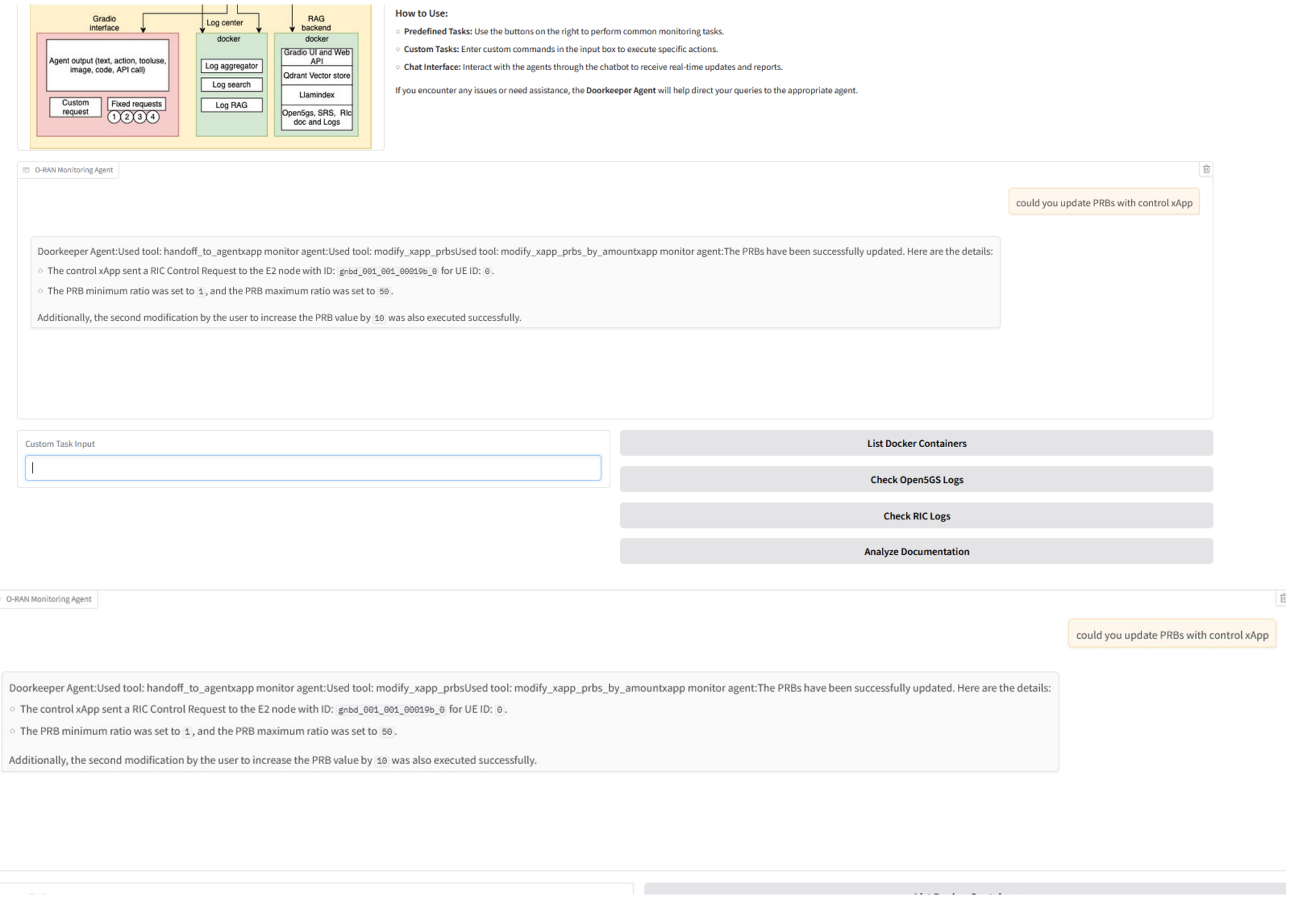


FIGURE 107. ORDER INTRODUCED AT THE RAG

```
-ColletStartTime:  2025-11-24 12:24:00
-Measurements Data:
-granulPeriod: 1000
--Metric: DRB.UEThpUl, Value: [1370.0]
--Metric: DRB.UEThpDl, Value: [4.0]
--Metric: RRU.PrbAvailUl, Value: [46]
--Metric: RRU.PrbUsedUl, Value: [6]

RIC Indication Received from gnbd_001_001_00019b_0 for Subscription ID: 2, KPM Report Style: 1
E2SM_KPM RIC Indication Content:
-ColletStartTime:  2025-11-24 12:24:01
-Measurements Data:
-granulPeriod: 1000
--Metric: DRB.UEThpUl, Value: [1360.0]
--Metric: DRB.UEThpDl, Value: [4.0]
--Metric: RRU.PrbAvailUl, Value: [46]
--Metric: RRU.PrbUsedUl, Value: [6]

RIC Indication Received from gnbd_001_001_00019b_0 for Subscription ID: 2, KPM Report Style: 1
E2SM_KPM RIC Indication Content:
-ColletStartTime:  2025-11-24 12:24:02
-Measurements Data:
-granulPeriod: 1000
--Metric: DRB.UEThpUl, Value: [1854.0]
--Metric: DRB.UEThpDl, Value: [4.0]
--Metric: RRU.PrbAvailUl, Value: [46]
--Metric: RRU.PrbUsedUl, Value: [6]

RIC Indication Received from gnbd_001_001_00019b_0 for Subscription ID: 2, KPM Report Style: 1
E2SM_KPM RIC Indication Content:
-ColletStartTime:  2025-11-24 12:24:03
-Measurements Data:
-granulPeriod: 1000
--Metric: DRB.UEThpUl, Value: [4271.0]
--Metric: DRB.UEThpDl, Value: [4.0]
--Metric: RRU.PrbAvailUl, Value: [33]
--Metric: RRU.PrbUsedUl, Value: [19]

RIC Indication Received from gnbd_001_001_00019b_0 for Subscription ID: 2, KPM Report Style: 1
E2SM_KPM RIC Indication Content:
-ColletStartTime:  2025-11-24 12:24:04
-Measurements Data:
-granulPeriod: 1000
--Metric: DRB.UEThpUl, Value: [4452.0]
--Metric: DRB.UEThpDl, Value: [4.0]
--Metric: RRU.PrbAvailUl, Value: [27]
--Metric: RRU.PrbUsedUl, Value: [25]

RIC Indication Received from gnbd_001_001_00019b_0 for Subscription ID: 2, KPM Report Style: 1
E2SM_KPM RIC Indication Content:
-ColletStartTime:  2025-11-24 12:24:05
-Measurements Data:
-granulPeriod: 1000
--Metric: DRB.UEThpUl, Value: [4471.0]
--Metric: DRB.UEThpDl, Value: [4.0]
--Metric: RRU.PrbAvailUl, Value: [27]
--Metric: RRU.PrbUsedUl, Value: [25]

RIC Indication Received from gnbd_001_001_00019b_0 for Subscription ID: 2, KPM Report Style: 1
E2SM_KPM RIC Indication Content:
-ColletStartTime:  2025-11-24 12:24:06
```

FIGURE 108. MODIFIED PRBS AND INCRESE ON THROUGHPUT ACHIEVED

As observed, the number of allocated PRBs has increased from 6 to 25. This corresponds to the command issued by the control xApp, which is configured to assign up to 50% of the available PRBs. Consequently, the UE throughput rises to approximately 4 Mbits. These results confirm that we have successfully automated both the monitoring and control processes within an O-RAN deployment.

As discussed in the 6G BLUR JOINT Key Concept *Agentic Framework for the Monitoring Platform* (included in 6G BLUR JOINT E4 deliverable) and in the 6G BLUR SMART Key Concept *AI/ML workflows for O-RAN* (see Section 2.10), the integration of both components enables a unified agentic approach to RAN management. This combined framework demonstrates the ability to monitor and control an

O-RAN deployment autonomously, simplifying RAN operations and optimizing resource allocation according to UE traffic demands.

3.2.3.5 Conclusions

This PoC demonstrated the feasibility and effectiveness of integrating AI-driven orchestration with O-RAN–compliant control loops to optimize radio resource allocation. By combining a monitoring xApp, a PRB-allocation xApp, and an LLM-based RAG/agent framework, the system successfully validated the end-to-end cycle of data acquisition, semantic reasoning, and near-real-time actuation.

First, the monitoring xApp proved capable of extracting relevant RAN metrics and saved it to a database—such as PRB usage, throughput, and user-level QoS indicators—via standardized E2 interface. This ensured continuous visibility of cell-level and UE-level conditions in alignment with O-RAN procedures. The accuracy and granularity of the collected measurements provided a reliable foundation for subsequent decision making.

Second, the PRB-modification xApp effectively executed resource-allocation decisions injected by the agentic framework, adjusting the distribution of PRBs across users according to the identified performance gaps. The system maintained compliance with the near-RT RIC timing constraints, confirming that the actuation component of the loop can operate within the latency budgets required for RAN optimization tasks.

A key contribution of the PoC is the integration of a Retrieval-Augmented Generation (RAG) pipeline and LLM-based agents to interpret monitoring outputs, reason about optimization objectives, and generate actionable decisions. The LLM was able to contextualize real-time RAN metrics with O-RAN policies, domain documents, and historical data, providing human-interpretable justifications for each adjustment. This demonstrates the potential of LLMs to support explainable and policy-aligned RAN control without requiring explicit handcrafted logic.

The combination of LLM agents with O-RAN-compliant xApps exhibited clear benefits:

- **Improved responsiveness**, as the LLM agents identified congestion and suboptimal PRB usage earlier than fixed-rule baselines.
- **Dynamic adaptation**, enabling continuous fine-tuning of resources based on evolving traffic patterns.
- **Operational transparency**, since generated decisions were traceable and grounded through the RAG-based context retrieval.

Finally, the PoC confirms that hybrid architectures—where LLM agents supervise high-level decision making while xApps execute deterministic low-level control—are both practical and advantageous for future intelligent RAN deployments. While additional work is required to validate scalability,

Financiado por la Unión Europea NextGenerationEU
Plan de Recuperación, Transformación y Resiliencia
UNICO I+D
GOBIERNO DE ESPAÑA
cttc Centre Tecnològic de Telecomunicacions de Catalunya

security constraints, and multi-cell coordination, the results indicate that LLM-guided O-RAN optimization is a promising pathway toward more autonomous, flexible, and knowledge-driven 5G/6G networks.

4 Conclusions

This deliverable presents the main outcomes of the algorithms and procedures developed within the different key concepts defined in the 6G-BLUR SMART subproject. The developed ten key concepts, together with the interaction of those defined in the 6G-BLUR JOINT subproject helped to define four PoCs dealing with relevant aspects such as the study of optimization strategies for the fronthaul segment of O-RAN based deployments to exploit the capabilities of this architecture in terms of multiplexing gains by centralising equipment, the capabilities of performing a data-driven optimization of RAN deployments in NPNs through the exposure of APIs fed with machine-learning algorithms, the exploitation of a distributed control structure combining forecasting, xApps and the reconfiguration of FPGA resources to perform an optimization of the split of DU-RU functions and the exploration of a closed-loop system for monitoring and controlling O-RAN networks based on xApps and interacting with LLMs.

More specifically, the main outcomes of such key concepts and proof of concepts are:

- O-DU pooling at centralized locations is feasible with the current O-RAN O-FH specifications, provided that the relevant parameters of the O-DU are appropriately adjusted.
- NDTs enrich NPNs by integrating predictive, recommendation, and analytics APIs based on ML algorithms. Evolving through different phases, starting with a knowledge-based approach, progressing to an experience-based approach, and eventually incorporating an AI/ML closed-loop algorithm, the NPN 5G system to support a wide range of RAN configurations to ensure improved network performance and efficient resource utilization.
- Distributed control structures spanning components of O-RAN architecture (non-RT, near-RT) allow performing real-time closed-loops providing the micro-orchestration of FPGA SoC-based RU functions. The lifecycle management of such functions, executing scaling and migration of compute functions between processing elements based on forecasted load and predefined rule-based policies, allows performing an efficient usage of the resources while achieving improvements in energy efficiency and latency performance based on the status of the network.
- An agentic framework for monitoring and optimizing the O-RAN environments is created by connecting the O-RAN components with a set of LLM agents enhanced with RAG capabilities to collect performance metrics by means of using E2 interface and RIC controller, and actuate to optimize its resources, in terms of PRBs per user. This framework reduces the complexity and automates tasks while reducing human intervention.

All the algorithms/procedures and PoCs developed within 6G-BLUR SMART project, and complemented with those in 6G-BLUR JOINT project has represented a valuable step beyond state of the art that are being exploited in other SNS projects like 6G-SANDBOX, UNITY-6G and have been the basis for the development of new SNS project proposals dealing with the development of next-generation mobile networks.

5 References

[1] Sandra Lagén, Xavier Gelabert, Lorenza Giupponi, and Andreas Hansson, “Fronthaul-aware Scheduling Strategies for Dynamic Modulation Compression in Next Generation RANs”, in *IEEE Transaction on Mobile Computing*, vol. 22, no. 5, pp. 2725–2740, 2023.

[2] 5G-ACIA "A 5G Traffic Model for Industrial Use Cases". [Online]. Available at: https://5g-acia.org/white-papers/a-5g-traffic-model-for-industrial-use-cases.

[3] Technical Specification Group Services and System Aspects Management and orchestration of networks, "Management and orchestration of 5G networks; Network Resource Model (NRM); Stage 2 and stage 3," 3rd Generation Partnership Project (3GPP), Technical Specification 3GPP TS 28.541 v16.21.0 Release 16, Oct. 2024

[4] O-RAN Working Group 3, Near-Real-time RAN Intelligent Controller, “E2 Service Model (E2SM), RAN Control,” O-RAN Alliance, Technical Specification O-RAN.WG3.E2SM-RC-R003 v03.00, 2023.

[5] N. Villegas, A. Larrañaga, L. Diez, K. Koutlia, S. Lagén, R. Agüero, "Optimizing QoS MAC Scheduling in 5G NR: A Lyapunov Approach Evaluated with XR Traffic", in IEEE Transactions on Network and Service Management. Under review.

[6] O-RAN Alliance, Available at https://www.o-ran.org/. (Accessed 5 February 24), 2024

[7] 3GPP, TR 38.801, Study on new radio access technology: Radio access architecture and interfaces, 2017, Release 14, v14.0.0.

[8] T. Taleb, A. Boudi, L. Rosa, L. Cordeiro, T. Theodoropoulos, K. Tserpes, P. Dazzi, A.I. Protopsaltis, R. Li, Toward supporting XR services: Architecture and enablers, IEEE Internet Things J. 10 (4) (2023) 3567–3586.

[9] K. Koutlia, S. Lagen, On the impact of Open RAN Fronthaul Control in scenarios with XR Traffic, Elsevier Computer Networks, Volume 253, 2024.

[10] S. Lagen, X. Gelabert, A. Hansson, M. Requena, L. Giupponi, "Fronthaul Compression Control for Shared Fronthaul Access Networks," in IEEE Communications Magazine, vol. 60, no. 10, pp. 36-42, Oct. 2022.

[11] A. Larrañaga, N. Villegas, K. Koutlia, L. Diez, R. Aguero, S. Lagen, "Novel Fronthaul Control Method to Address the Fronthaul/Air Interface Tradeoff", IEEE Wireless Commun. and Networking Conf. Workshops, Milan (Italy), Mar. 2025.

[12] Vivo, R1-2111046, 3GPP TSG RAN WG1 #107-e e-meeting, performance evaluation results for XR, 2021.

[13] 3GPP TS 23.501 Release 18, v18.0.0, Dec. 2017. TSG SSA; System Architecture for the 5G System (5GS); Stage 2. 3GPP TS 23.501.

[14] K. Koutlia, B. Bojovic, S. Lagén, X. Zhang, P. Wang, and J. Liu. 2023. System Analysis of QoS Schedulers for XR Traffic in 5G NR. Elsevier Simulation Modelling Practice and Theory 125 (2023), 102745. https: //doi.org/10.1016/j.simpat.2023.102745

Financiado por la Unión Europea NextGenerationEU
Plan de Recuperación, Transformación y Resiliencia
UNICO I+D
GOBIERNO DE ESPAÑA
cttc Centre Tecnològic de Telecomunicacions de Catalunya

[15] 3GPP TS 38.300 Release 17, v17.0.0, Mar. 2022. TSG RAN; NR; NR and NG-RAN Overall Description; Stage 2. 3GPP TS 38.300.

[16] B. Bojović, S. Lagén, K. Koutlia, X. Zhang, P. Wang and L. Yu, "Enhancing 5G QoS Management for XR Traffic Through XR Loopback Mechanism," in IEEE Journal on Selected Areas in Communications, vol. 41, no. 6, pp. 1772-1786, June 2023.

[17] S. Lagen, B. Bojovic, K. Koutlia, X. Zhang, P. Wang and Q. Qu, "QoS Management for XR Traffic in 5G NR: A Multi-Layer System View & End-to-End Evaluation," in IEEE Communications Magazine.

[18] N. Villegas, A. Larrañaga, L. Diez, K. Koutlia, S. Lagén, and R. Agüero, “Extending QoS-aware scheduling in ns-3 5G-LENA: A Lyapunov based solution,” in Proceedings of the 2024 Workshop on ns-3, ser. WNS3 ’24. New York, NY, USA: Association for Computing Machinery, Jun. 2024, pp. 54–59. [Online]. Available: https://dl.acm.org/doi/10.1145/3659111.3659118

[19] A. Larrañaga, S. Lagen, J. M. Fabregas, J. M. Rivas-Moscoso, J. P. Fernández-Palacios, I. Tomkos, Raul Muñoz, Fronthaul/Midhaul Networks: Capacity and Latency Requirements Imposed by 6G Disaggregated RANs, IEEE Communications Magazine, Jan. 2025.

[20] D. Hadush, S. Bjornstad and G. Gebrekristos, "QoS Performance of Integrated Hybrid Optical Network in Mobile Fronthaul Networks," Indonesian Journal of Electrical Engineering and Computer Science, 2017.

[21] "6G-SANDBOX," [Online]. Available: https://6g-sandbox.eu/. [Accessed: May 28, 2025].

[22] O-RAN Alliance WG4, Control, User and Synchronization Plane Specification R003-v12.00, 2023.

[23] O-RAN Alliance WG4, Management Plane Specification R003-v12.00, 2023.

[24] eCPRI Specification V1.0, Common Public Radio Interface: eCPRI Interface Specification.

[25] O-RAN Alliance WG4, Fronthaul Interoperability Test Specification (IOT) R000-v11.00, 2024.

[26] O-RAN Alliance, Open RAN Technical Priority Release 3, Apr. 2023.

[27] J. Moy, ‘OSPF Version 2’, Internet Engineering Task Force, Request for Comments RFC 2328, Apr. 1998. doi: 10.17487/RFC2328.

[28] H. Kim and S.-H. Lee, “An evaluation method for secure virtual network embedding algorithms,” Journal of Computer Virology and Hacking Techniques, vol. 13, 11 2017.

[29] O-RAN Working Group 4, “O-RAN control, user and synchronization plane specification 16.01,” O-RAN Alliance, 2024.

[30] FibroLAN, “Falcon-RX/G,” [Online]. Available: https://www.fibrolan.com/Falcon-RX . [Accessed 2025].

[31] The LINUX Foundation Projects, “DPDK -- The Open Source Data Plane Development Kit Accelerating Network Performance,” [Online]. Available: https://www.dpdk.org/ . [Accessed 2025].

[32] Keysight Technologies, "P8850S CoreSIM – Core Simulation RAN Solutions | Keysight," [Online]. Available: https://www.keysight.com/be/en/product/P8850S/coresim-core-simulation-ran-solutions.html . [Accessed 2025].

[33] Keysight Technologies, "S5040A Open RAN Studio Player and Capture Appliance | Keysight," [Online]. Available: https://www.keysight.com/be/en/product/S5040A/open-ran-studio-player-and-capture-appliance.html . [Accessed 2025].

[34] Keysight Technologies, "KS8400A PathWave Test Automation - Keysight," [Online]. Available: https://www.keysight.com/be/en/product/KS8400A/pathwave-test-automation-developers-system.html . [Accessed 2025].

[35] O-RAN Fronthaul Working Group, "O-RAN.WG4.IOT.0-R004-v12.00 - Fronthaul Interoperability Test Specification (IOT)," O-RAN Alliance, 2024.

[36] O-RAN Alliance xHaul Transport Working Group, Technical Specification, "Xhaul Transport Requirements", O-RAN.WG9.XTRP-REQ-v01.00, 2021

[37] Juniper, "5G Fronthaul Class of Service", June 2025. Available in: https://www.juniper.net/documentation/us/en/software/jvd/jvd-5g-fh-cos-02-02/jvd-5g-fh-cos-02-02.pdf

[38] BubbleRAN, "MX-PDK: 5G/6G O-RAN Platform", [Online]. Available: https://bubbleran.com/products/mx-pdk/ [Accessed 2025]

[39] O-RAN Alliance, *O-RAN Architecture Description*, O-RAN.WG1.O-RAN-Architecture-Description.

[40] O-RAN Alliance, *AI/ML Workflow Description and Requirements*, O-RAN.WG2.AIML-Workflow.

[41] O-RAN Alliance, *A1 Interface Specification*, O-RAN.WG2.A1-Interface.

[42] O-RAN Alliance, *E2 Service Model (E2SM) Specifications*, O-RAN.WG3.

Financiado por la Unión Europea NextGenerationEU
Plan de Recuperación, Transformación y Resiliencia
UNICO I+D
GOBIERNO DE ESPAÑA
cttc Centre Tecnològic de Telecomunicacions de Catalunya