Horizon 2020
European Union Funding
for Research & Innovation

EMULRADIO4RAIL

Deliverable D 3.3:
Design and implementation of radio access emulation tool

Project acronym:	EMULRADIO4RAIL
Starting date:	01/12/2018
Duration (in months):	18
Call (part) identifier:	H2020-S2R-OC-IP2-2018-03
Grant agreement no:	826152
Due date of deliverable:	Month 30/11/2020 (1rst amendment)
Actual submission date:	30/11/2020
Responsible/Author:	Raul Torrego - IKERLAN (IKL)
Dissemination level:	PU
Status:	Draft
Version	2.0

Reviewed: Christophe Gransart, Université Gustave Eiffel

Document history		
Revision	Date	Description
0	03/11/2020	First issue
1	24/11/2020	Integration of all the contributions
2	30/11/2020	Integration of final contributions

Report contributors		
Name	Beneficiary Short Name	Details of contribution
Raul Torrego	IKERLAN	Deliverable leader
Jose Soler, Ying Yan	DTU	Contribution on backhaul emulation platform
Ali Sabra, Marion Berbineau	Université Gustave Eiffel	Contribution on LTE emulation platform with OAI and control infrastructure
Virginie Deniau	Université Gustave Eiffel	Contribution on LTE emulation platform using CMW500 from R&S
Alessandro Vizzarri	RADIOLABS	Contribution on satellite emulation platform
Sofiane Kharbech, Laurent Clavier, Rédha Kassi, Philippe Mariage Virginie Deniau	ULILLE	Contribution on Wi-Fi emulation platform

Table of Contents

List of Figures

1. Executive Summary

The European EMULRADIO4RAIL project aims to develop an innovative emulation platform for tests and validation of various radio access technologies (RAT) like Wi-Fi, LTE, LTE-A and Sat-Coms. The project considers the integration of both channel emulators and network emulators into a single emulation platform so that the proposed testbed can offer a complete (from physical layer to IP level) test environment.

This deliverable provides the description of the different parts of the complete emulation platform that have been implemented. The EMULRADIO4RAIL project initially aimed to implement a complete and single emulation platform, which included several radio bearers and RF channel emulators in parallel (as described in Deliverable 3.1 [2]). However, due to COVID-19 and the travel restrictions that have been stablished across Europe, it has not been possible to complete all the integration with industrial partners, so this deliverable describes the individual parts and partial integrations that have been carried out.

The deliverable starts with a sum up of Deliverable D3.1 [2] in which the complete emulation platform is presented. Later, the implementation of the five different parts in which the platform is divided are presented: Wi-Fi emulation platform, LTE emulation platform, Satellite emulation platform, backhaul network emulation platform and control infrastructure.

2. Abbreviations and acronyms

Abbreviation / Acronyms	Description
ACS	Adaptable Communication System
AP	Access Point
COTS	Component Of The Shelf
DUT	Device Under Test
EPC	Evolved Packet Core
GUI	Graphical User Interface
HITL	Hardware in the Loop
HW	Hardware
IP	Internet Protocol
LTE	Long Term Evolution
LTE-A	Long Term Evolution Advanced
OAI	Open Air Interface
OPNET	OPtimized Network Engineering Tool
PC	Personal Computer
PLMN	Public Line Mobile Network
PSG	Performance Signal Generator
PXA	Extra performance Signal analyser
PXB	Base Band Generator
RF	Radio Frequency
Sat-Coms	Satellite Communications
SDR	Software-Defined Radio
SITL	System In The Loop
SW	Software
USB	Universal Serial Bus
USRP	Universal Software Radio Peripheral
Wi-Fi	Wireless Fidelity

3. Background

The present document, together with the hardware and software that have been developed, constitutes the Deliverable D3.3 "**Design and implementation of Radio access emulation tool**" according to Shift2Rail Joint Undertaking programme of the project titled "EMULATION OF RADIO ACCESS TECHNOLOGIES FOR RAILWAY COMMUNICATIONS" (Project Acronym: EMULRADIO4RAIL, under Grant Agreement No 826152).

In December 2018, the European Commission awarded a grant to the EMULRADIO4RAIL consortium of the Shift2Rail / Horizon 2020 call (H2020-S2RJU-OC-2018 S2R-CFM-IP2-01-2015).

EMULRADIO4RAIL is a project connected to the development of a new Communication System planned within the Technical Demonstrator TD2.1 of the 2nd Innovation Programme (IP2) of Shift2Rail JU: Advanced Traffic Management & Control Systems. The IP2 "Advanced Traffic Management & Control Systems" is one of the five asset-specific Innovation Programmes (IPs), covering all the different structural (technical) and functional (process) sub-systems related to control, command and communication of railway systems.

4. Objectives

This document has been prepared to provide the description of the implementation of the different parts of the complete emulation platform that have been carried out. Although it has not been possible to integrate the complete platform due to COVID-19 travel restrictions, this deliverable describes:

- Implementation of the Wi-Fi emulation setup
  - Setup with IKERLAN RF channel emulator
  - Setup with ULILLE RF channel emulator
- Implementation of the LTE emulation setup
  - Setup with IKERLAN RF channel emulator and DTU OAI platform
  - Setup with Propsim RF channel emulator and Uni Eiffel OAI platform
- Implementation of the Satellite emulation platform
  - Setup with RDL satellite emulator
- Implementation of the backhaul network emulation
  - Setup with DTU backhaul network emulation platform
- Implementation of the control infrastructure

5. Introduction

EMULRADIO4RAIL will provide an innovative platform for test and validation of various radio access technologies (Wi-Fi, LTE, LTE-A and Sat-Coms) that combines very new approaches for testing so called System In-The-Loop (SITL) and Hardware-In-The-Loop (HITL). The prototypes of the Adaptable Communication System (ACS) from X2RAIL-3 WP3, which constitute the devices under test (DUT), will be coupled to emulators of both the communication core network and various radio access technologies thanks to the coupling among discrete event simulator such as RIVERBED modeler (former OPNET modeler), Open Air Interface (OAI), various radio Channel emulators, Network emulators, models of IP parameters and real physical systems.

During Task 2.1 "Solutions for emulation of Radio bearer" and Task 3.1 "Development and implementation of the Radio access emulation tools for the different Radio access technologies" both the design and implementation of the emulation platform has been carried out. Deliverable D2.1 "Solutions to emulate the Radio Bearer (stream b)" [1] and D3.1 "High-level design of Radio access emulation tool (stream d)" [2] describe the proposed design, while this Deliverable D3.3 will present the final implementations of the different parts of the complete emulation platform that has been carried out. The proof of concept of integration of all the platforms is given by photography and also with the film of the project.

Deliverable D3.4 "Operation manual of Radio access emulation tool (stream f)" will describe the necessary steps that have to be followed in order to operate the platform, hence, that information is out of the scope of this deliverable D3.3 and only a short introduction will be done.

6. The complete emulation platform (extracted from D3.1)

The following picture shows the overall view of the EMULRADIO4RAIL platform. Five main parts can be distinguished:

- The Wi-Fi emulation platform which includes a RF channel emulator and COTS Wi-Fi devices
- The LTE emulation platform which includes a RF channel emulator and OAI and COTS LTE devices
- The satellite emulation platform operating only at IP level
- The backhaul network emulation platform.
- The control PC

The platform can also be configured with two LTE emulation platforms (one to emulate LTE private deployment and one for LTE public deployment) at the expense of the Wi-Fi platform.

All the sub-platforms were described in [D3.1].

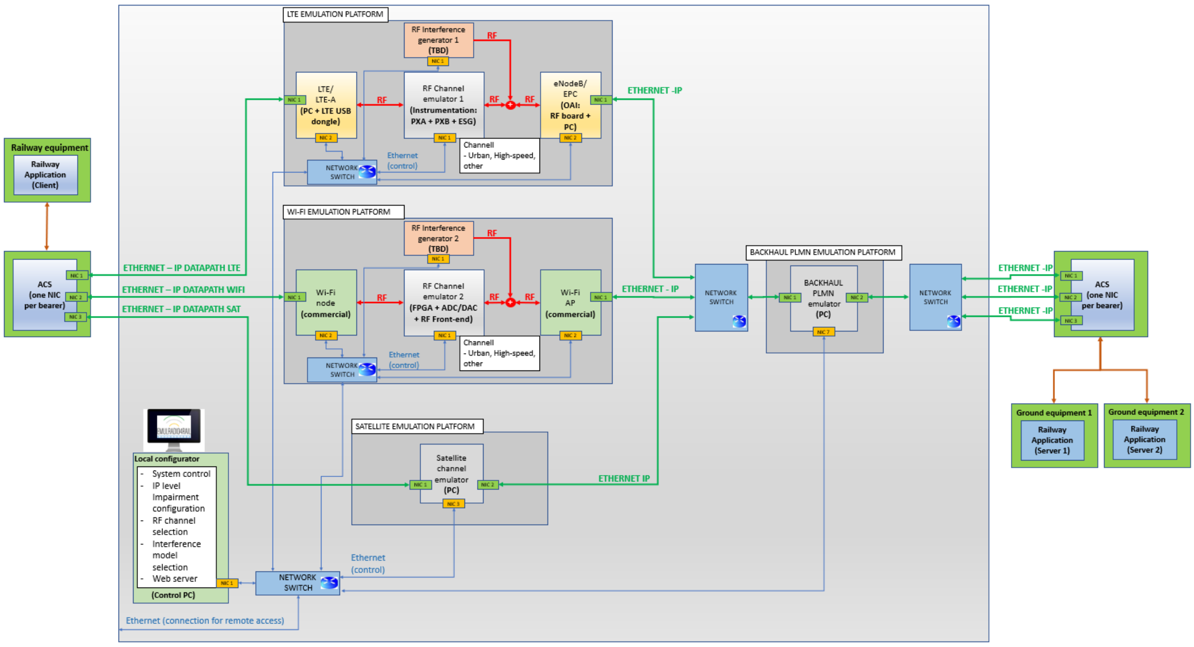

Figure 6.1: Overall view of the emulation platform

The following sections will individually describe the real implementations of the different parts of the platform that have been carried out.

7. Wi-Fi emulation platform

This section presents the different implementations of the Wi-Fi emulation platform that have been carried out. Two versions of the Wi-Fi emulation platform have been implemented:

- One with the IKERLAN RF channel emulator and a jammer
- One with the ULILLE RF channel emulator and a secondary Wi-Fi interference generator

7.1. Wi-Fi emulation platform with IKERLAN RF channel emulator

The Figure 7.1 below shows the Wi-Fi emulation platform implemented with the IKERLAN RF channel emulator:

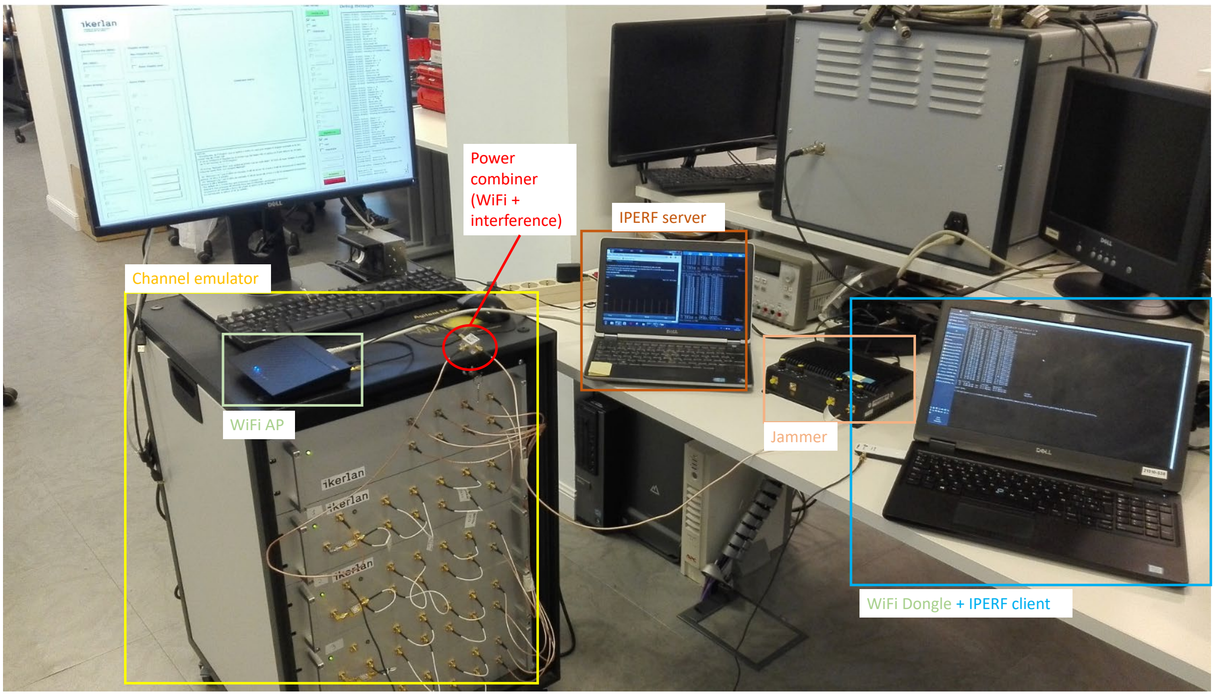

Figure 7.1: Wi-Fi emulation platform with the IKERLAN RF channel emulator

In the figure the following parts can be identified:

- The RF channel emulator built by IKERLAN for the EMULRADIO4RAIL project
- An ASUS RT-N12 Wireless Router acting as a wireless AP
- A TP-LINK TL-WN722N Wireless Dongle
- The TX4CA jammer from Projammers introduced into the Wi-Fi AP side with an RF splitter.

- The control PCs

Figure 7.2: WI-FI AP and dongle

The Wi-Fi emulation platform aims to receive IP frames either through the USB port of the dongle (if it is directly attached to the DUT (client)) or through the Ethernet port of the PC in which the dongle is attached (which acts as a bridge between the DUT (client) and the dongle). This IP frames are converted by the dongle into RF signal. This RF signal is modified by the RF channel emulator and interference is added to it by the jammer before arriving to the AP. The AP converts back the RF signals into IP packets thar are sent through a Ethernet port to the DUT (server). This datapath also works in the opposite direction, from the AP to the dongle.

In order to run the Wi-Fi emulation platform the next steps have to be carried out:

- Set-up the Wi-Fi AP and create a Wi-Fi network
- Connect the Dongle to the Wi-Fi network created by the AP
- Configure the RF channel emulator with the desired channel model
- Configure the Jammer with the desired jamming power
- Connect the DUT (client) to the Dongle
- Connect the DUT (server) to the AP
- Run the desired application in the DUT

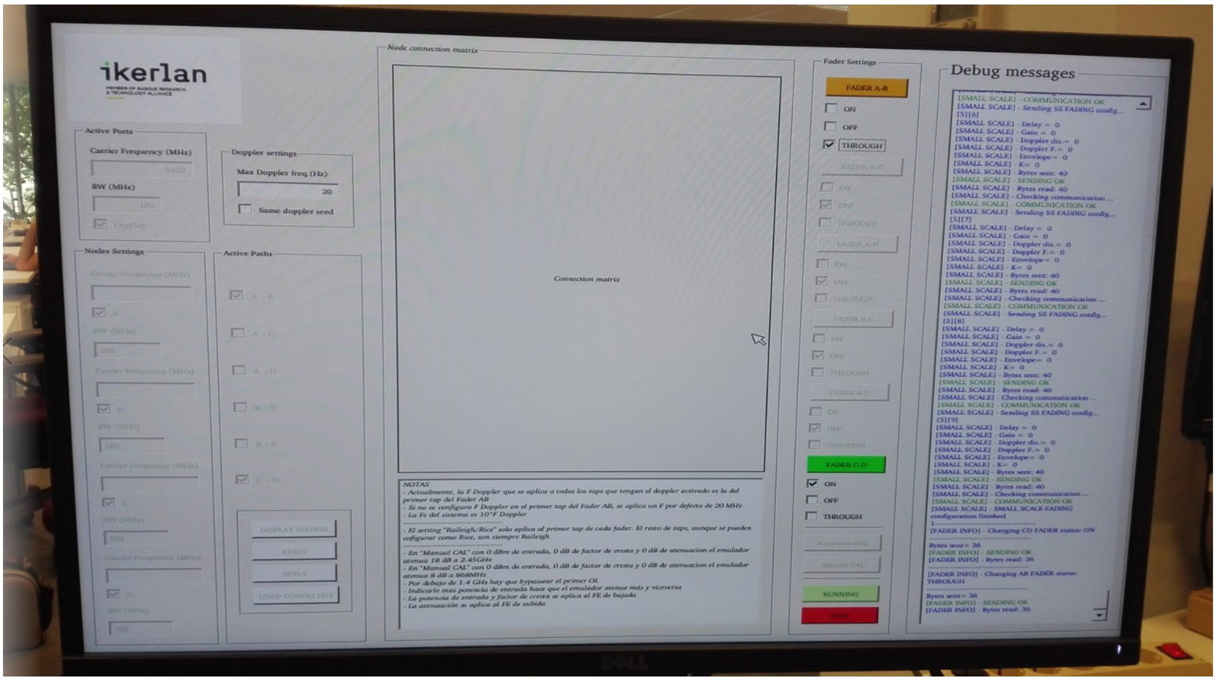

Figure 7.3: Detail of the RF channel emulator control GUI

7.2. Wi-Fi emulation platform with ULILLE RF channel emulator

Figure 7.4 shows the Wi-Fi emulation platform implemented with the ULILLE RF channel emulator:

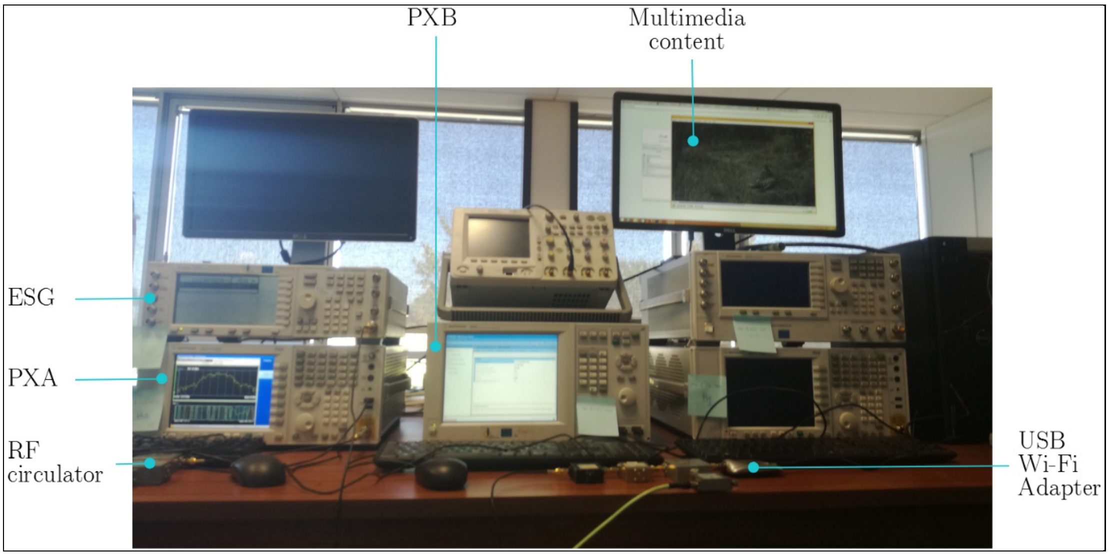

Figure 7.4: Wi-Fi emulation platform with ULILLE RF channel emulator

This second Wi-Fi channel emulation platform includes:

- The baseband generator and channel emulator PXB N5106A from Keysight Technologies
- Since it is a baseband emulator, and to be able to work with RF signals, the PXB requires:
  - RF demodulator (e.g. PXA) at its input
  - RF modulator (e.g., ESG) at its output

  This setup can be simplified with the newest channel emulator (Propsim), which can directly work with RF signals.

- Additional elements:
  - Wi-Fi USB adapters for signals reception
  - Two Wi-Fi Access points, one used the main communication and the other for the interference.
  - All the required RF elements (circulators, splitters, isolator, cables, etc.)
  - Two computers for supervising and exchanging traffic.

In order to introduce interferences from a secondary Wi-Fi network, a setup with several RF splitters and combiners has also been implemented as shown on Figure 7.5.

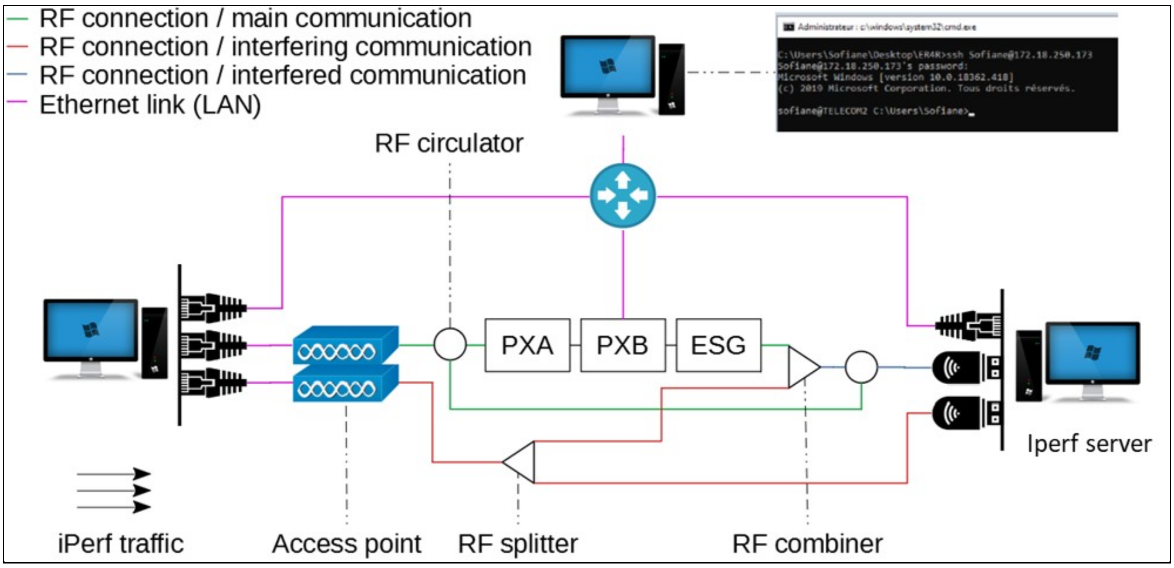

Figure 7.5: Global scheme of the set up

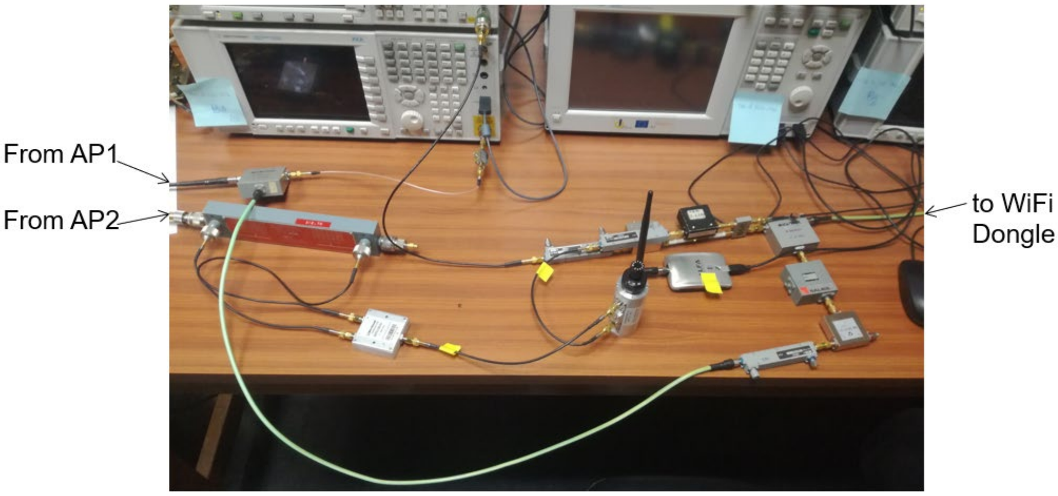

Figure 7.6: Interference addition setup

Once the testbed is connected following the diagram above, and all the instrumentations are turned on, the required steps to run an emulation scenario are:

- On the PXB, select the channel configuration that you want to emulate, or enter your own channel parameters.
- Using the configuration interfaces of the access points, select the Wi-Fi communication channels for both of them. Here, the choice depends on the type of the interference, e.g., we can choose Wi-Fi channel#1 for the main communication and channel#2 for the interfering link in the case of an adjacent channel interference.
- Generate traffic on both links (main and interfering links).
- Gathering results for statistical representation and interpretation.

8. LTE emulation platforms

This section presents the different implementations of the LTE emulation platform that have been carried out. Two versions of the LTE emulation platform have been implemented:

- One with the IKERLAN RF channel emulator and the OAI platform from DTU
- One with the Keysight Propsim RF channel emulator and CMW500 equipment from University Gustave Eiffel
- One with the Keysight Propsim RF channel emulator and the OAI platform from University Gustave Eiffel

8.1. LTE emulation platform with IKERLAN RF channel emulator and the OAI platform from DTU

Figure 8.1 shows the implemented LTE emulation platform with the integration of the OAI LTE platform from DTU and the RF channel emulator from IKERLAN.

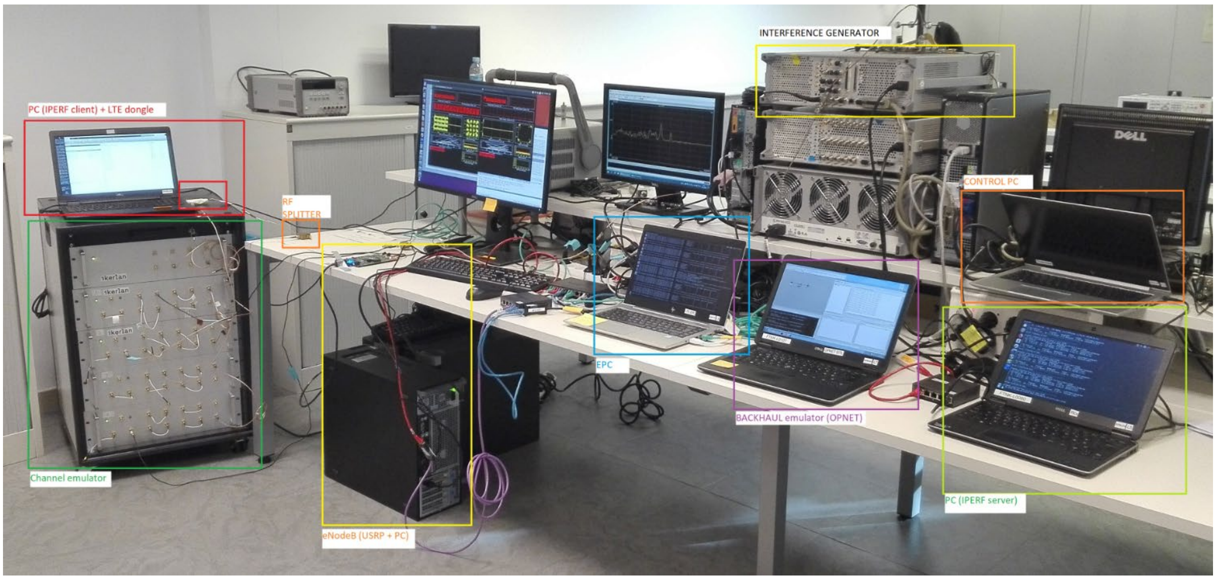

Figure 8.1: LTE emulation platform with IKERLAN RF channel emulator and DTU OAI platform

In the LTE emulation platform the following parts can be identified:

- The RF channel emulator built by IKERLAN for the EMULRADIO4RAIL project
- A HUAWEIE3372 LTE dongle
- The LTE eNodeB (USRP B210 SDR front-end together with a PC running the OAI stack)
- The LTE Evolved Packet Core (EPC)
- The backhaul network emulation platform (see section 10)
- The control PC

- The interference generator (Agilent E4438C RF signal generator and a splitter)

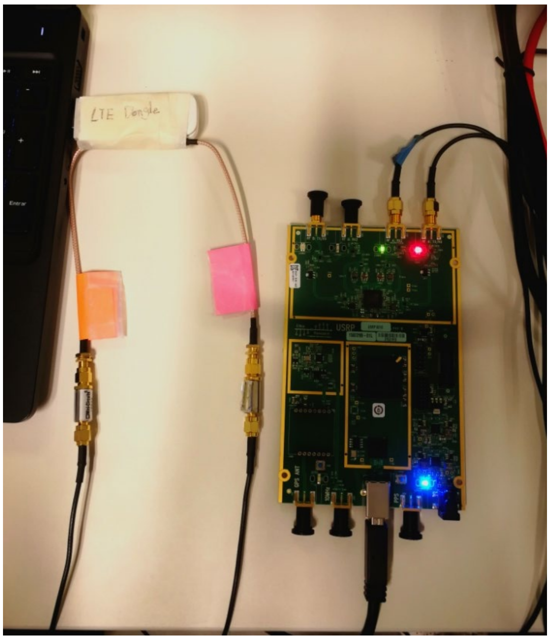

Figure 8.2: Detail on the LTE Dongle and the USRP B210 part of the LTE eNB

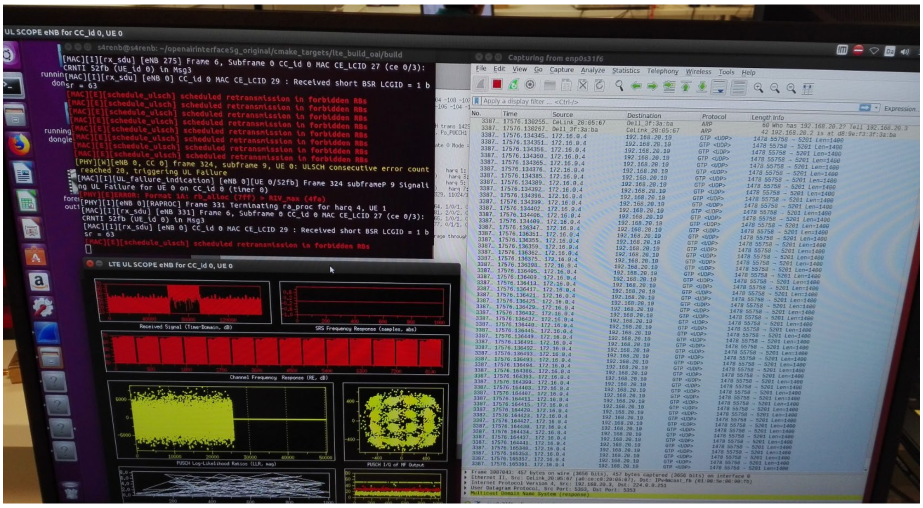

Figure 8.3: Detail of OAI eNB SW interface

The LTE emulation platform aims to receive IP frames either through the USB port of the dongle (if it is directly attached to the DUT (client)) or through the Ethernet port of the PC in which the dongle is attached (which acts as a bridge between the DUT (client) and the dongle). This IP frames are converted by the dongle into RF signal. This RF signal is modified by the RF channel emulator and interference is added to it by the signal generator before arriving to the OAI eNB. The OAI eNodeB converts back the RF signals into IP packets that are sent through a Ethernet port to the EPC. The EPC, after processing the frames, sends them to the backhaul network emulator which introduces more secondary IP traffic. Finally, all the traffic are sent to the DUT (server). This datapath also works in the opposite direction, from the OAI EPC/eNB to the dongle.

In order to run the LTE emulation platform, the next steps have to be carried out:

- Set-up the LTE OAI EPC and eNB and create the LTE cell
- Connect the Dongle (which has the correct SIM card) to the LTE network created by OAI
- Configure the RF channel emulator with the desired channel model
- Configure the signal generator with the desired interference setup
- Configure the backhaul network emulator with the desired traffic model
- Connect the DUT (client) to the Dongle
- Connect the DUT (server) to the backhaul network emulator PC
- Run the desired application in the DUT

8.2. LTE emulation platform with Propsim RF channel emulator and the OAI platform from University Gustave Eiffel

The Figure 8.4 below shows the second LTE emulation platform that has been implemented with the Propsim RF channel emulator and the OAI platform from UGE.

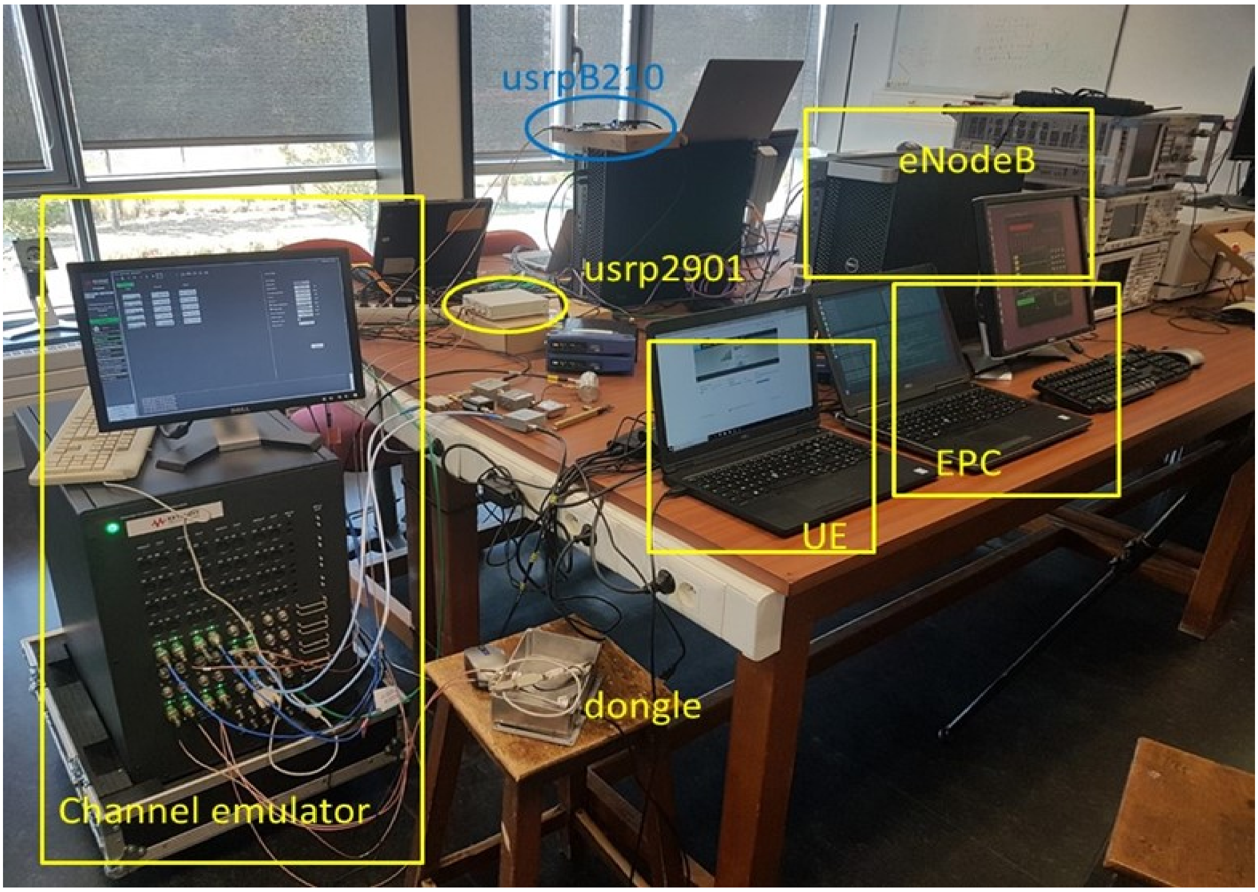

Figure 8.4: LTE emulation platform with Propsim RF channel emulator and UGE OAI platform

In Figure 8.4 the following devices can be identified:

- The Propsim RF channel emulator from Keysight
- A HUAWEI H8372 LTE dongle (ref 8372h-153) connected to the UE PC
- The LTE eNodeB (USRP 2901 SDR front-end together with a PC running the OAI stack)
- The LTE Evolved Packet Core (EPC)

The OAI platform developed at Université Gustave Eiffel is the exact replication of the one developed by DTU. We have considered the same version of the software.

As described in D3.2 [3], the eNodeB configuration files have an impact on the performance of the OAI platform. The platform is working in SISO mode (Single Input Single Output) in band 7 (2.66 GHz for Downlink and 2.54 GHz for Uplink). We have considered a transmission on 20 MHz. In the eNodeB configuration file, several parameters can be modified, such as TX and RX attenuation and also TX and RX gain, the number of resource block PRB (Physical Resource Block) and PDSCH (Physical Downlink Shared Channel) max power among other. These parameters are able to influence the eNodeB performances at Physical layer. It is very important to master all these aspects and to define at the end the best configurations. It is also important to note that it is possible to modify the transmission mode, namely the number of antennas considered, the frequency band and also duplexing mode (TDD or FDD). The eNodeB configuration can be modified easily in the configuration file.
Different SDR boards can be considered. For evolution towards 5G, X310 SDR board (or equivalent with MIMO and performant FPGA features) should be considered.

8.3. LTE emulation platform with Propsim RF channel emulator and the Rohde & Schwarz CMW500 equipment from University Gustave Eiffel

The setup is quite simple and the LTE parameters can be fixed with the CMW500 from Rohde & Schwarz. The UE is emulated with a LTE dongle. We used the Huawei dongle 8372-153h like for the OAI platform but with a specific SIM card from Rohde & Schwarz t connect the UE to the eNodeB. The signal is directly injected in the Propsim channel emulator, which emulates the channel as illustrated on Figure 8.5. The CMW500 emulates the eNobeB. FDD or TDD can be considered. In this set up we have emulated a FDD LTE signal. The intrinsic delay introduced by the channel emulator (4.5 µs) and attenuation (around 20 dB) should be known by the CMW500. Then the traffic can be sent at IP level and tests can be performed.

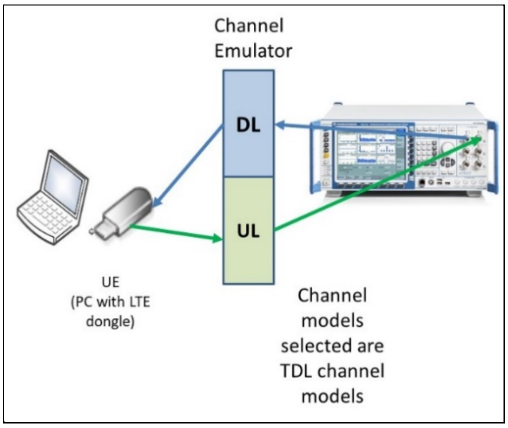

Figure 8.5: Drawing of the set up with CMW500

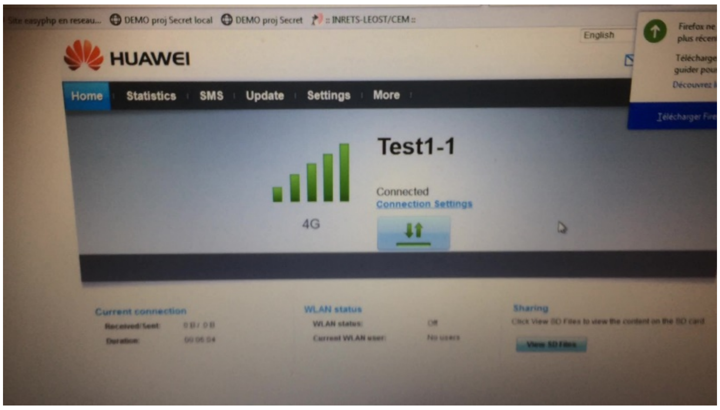

Figure 8.6: View of the UE connected

Figure 8.7: Photography of the CMW500 screen during the experimentation

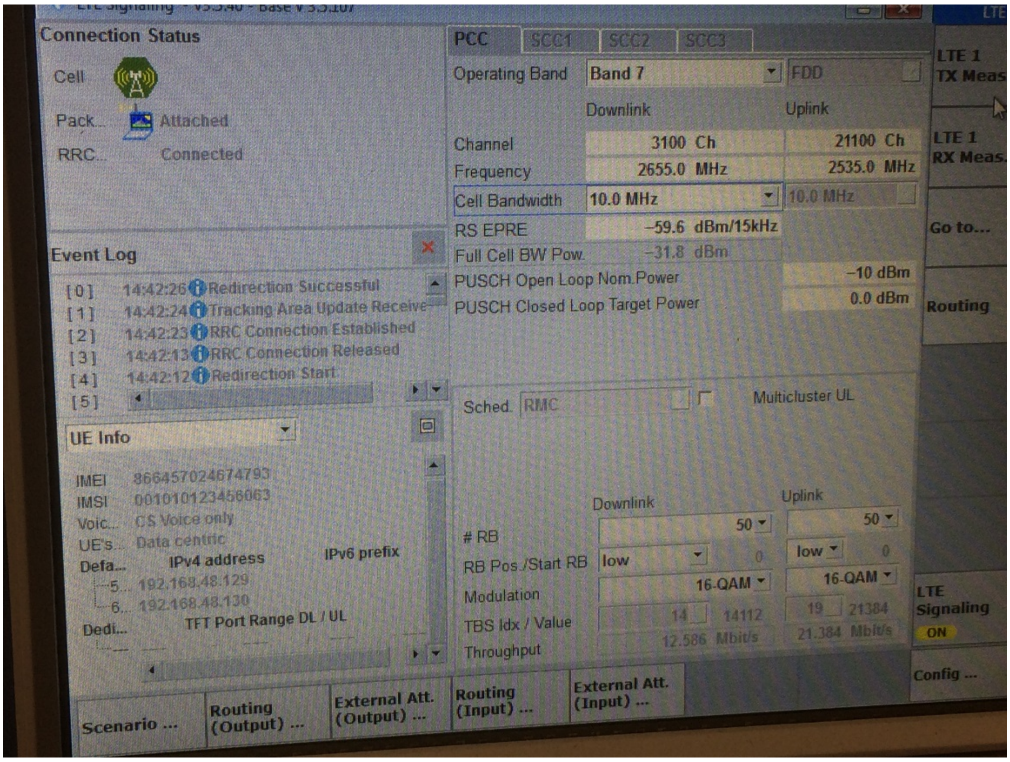

Figure 8.8: Photography of the CMW500 screen during the experimentation with confirmation of UE connection

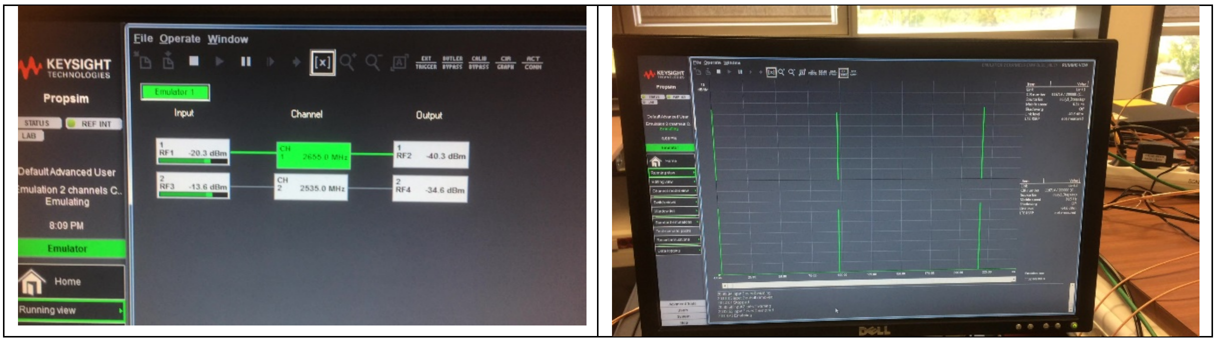

Figure 8.9: Photography of the Propsim screen with the 3taps hilly terrain model running

9. Satellite emulation platform

This section presents the satellite emulation platform that has been implemented. Besides, it also describes the integration work that has been carried out with the Hitachi rail prototype.

9.1. The Device Under Test (DUT): the Hitachi Rail Prototype

The Hitachi Rail Prototype, called MLCP, is the On Board Unit (OBU) and it is the Device Under Test (DUT) to be connected to the EMULRADIO4RAIL emulator platform via the Ethernet port. Due to the Covid19 emergency that represented an obstacle for the connection in the Hitachi Rail lab, the connection between the Hitachi Rail OBU and the EMULRADIO4RAIL satellite emulator was setup through a WAN connection over a VPN tunnel.

Figure 9.1: Satellite emulation platform - DUT

9.2. The Satellite IP emulator

The EMULRADIO4RAIL satellite network emulator is able to emulate at IP level the end-to-end transmission over a satellite link composed by the satellite modem and satellite earth station of both user and server side.
In this way it is possible to generate IP traffic from the user device and to send it to the application server (upstream link) and vice versa (downstream).
It is possible to inject the following IP impairments to alter the transmission:

- link bitrate (upstream/downstream)
- Link latency

- Packet loss

The impaired IP flow at the output of the satellite emulator is then delivered to the server running the railway application.

The satellite IP emulator is transparent to the traffic injected by an external device (traffic generator such as Iperf or nPing).

Taking this into account the satellite emulation platform runs into a single PC as can be seen in the following figures:

Figure 9.2: Satellite emulation platform

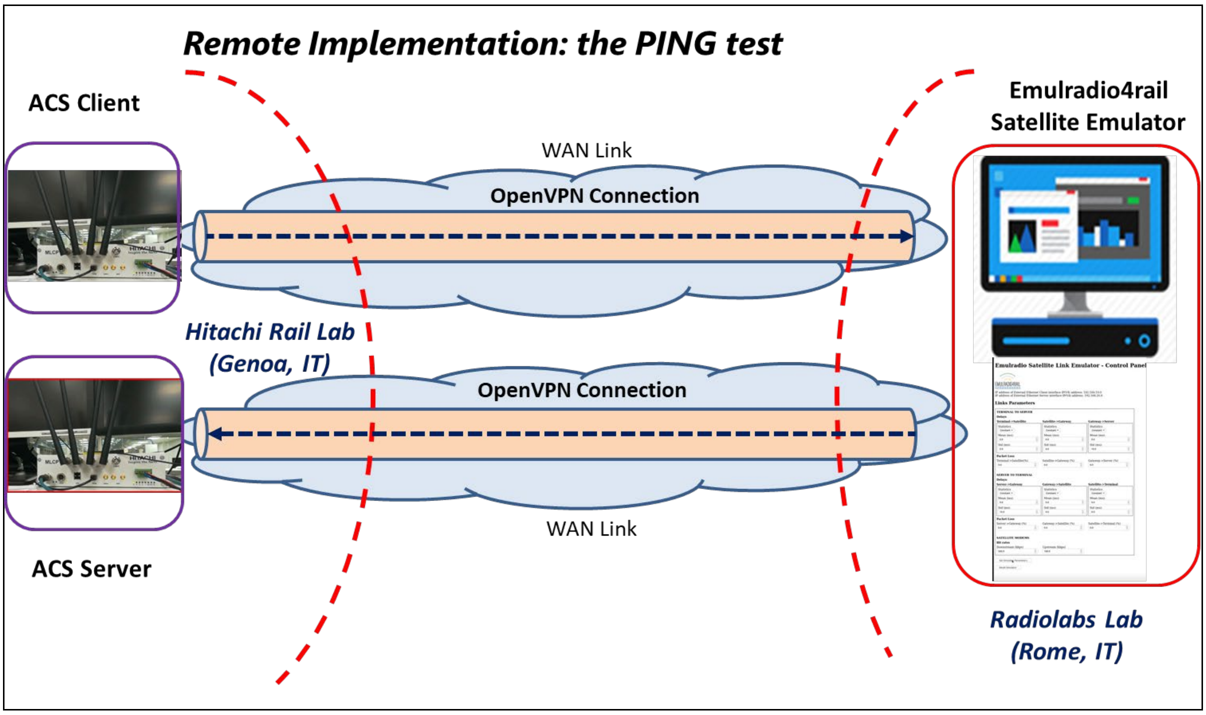

Figure 9.3: Remote implementation of emulradio4rail satellite emulator.

The solution considered for the integration of the Emulradio4Rail satellite IP emulator (located in Rome, Italy) with the testbench of Hitachi Rail (located in Genoa, Italy) is depicted in figure 9.3.

Two different VPN connections between the labs in Rome and in Genoa are established using OpenVPN software tool. The first one connects the ACS client to the Radiolabs satellite IP emulator. The second one connects the Radiolabs satellite IP emulator to the ACS server. Both upstream and downlink links are established successfully.
The connectivity test is performed using the Ping command from the ACS client (in Genoa) to ACS server (in Genoa) passing through the satellite IP emulator (in Rome).

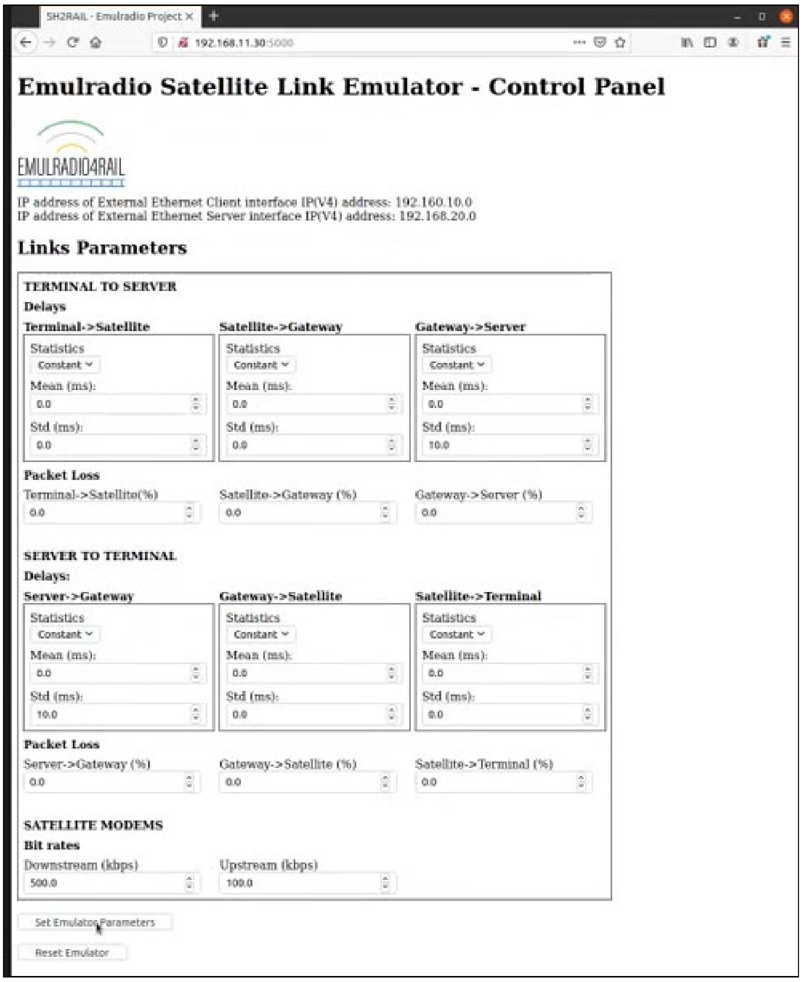

Figure 9.4: Web-based control interface of the satellite IP emulator

The connectivity test has been performed in two separated steps.
The first step consists in the measuring of the Round Trip Time (RTT) over the WAN link between the ACS client and server passing through the satellite IP emulator. In this step all IP impairments in the emulator are set to zero.
The second step of the test consists in:

a. Set the test configuration on the dedicated web-based control interface of the satellite IP emulator in terms of:

- Link bitrate (upstream/downstream)
- IP impairments: latency, packet loss
- statistical distribution models for IP impairments

b. perform the connectivity test using the PING command to observe the KPIs at IP level:
   - the Round Trip Time (RTT) of the application
   - the packet loss which is evidenced by observing the missing of some packets looking at the sequence number of the returned PINGs

10. Backhaul emulation platform

The backhaul simulation aims to analyse traffic transmission performance in a close-to-reality core network topology using Riverbed Modeler. Based on literature or previous measurements, the traffic types and network topology are defined. Like the satellite emulation platform, the backhaul emulation platform also works at IP level and therefore can be implemented in a single PC.

Figure 10.1 shows the PC in charge of running the backhaul network emulation (in blue) integrated together with the LTE emulation platform. This PC illustrated in the blue frame is a host of the network simulation software. The simulator is called Riverbed Modeler, previously called OPNET. Riverbed Modeler is a discrete event simulator. Using the simulation with its SITL package, System In the Loop. Using SITL, it is able to perform an integrated test with by combining both simulated and real environment.

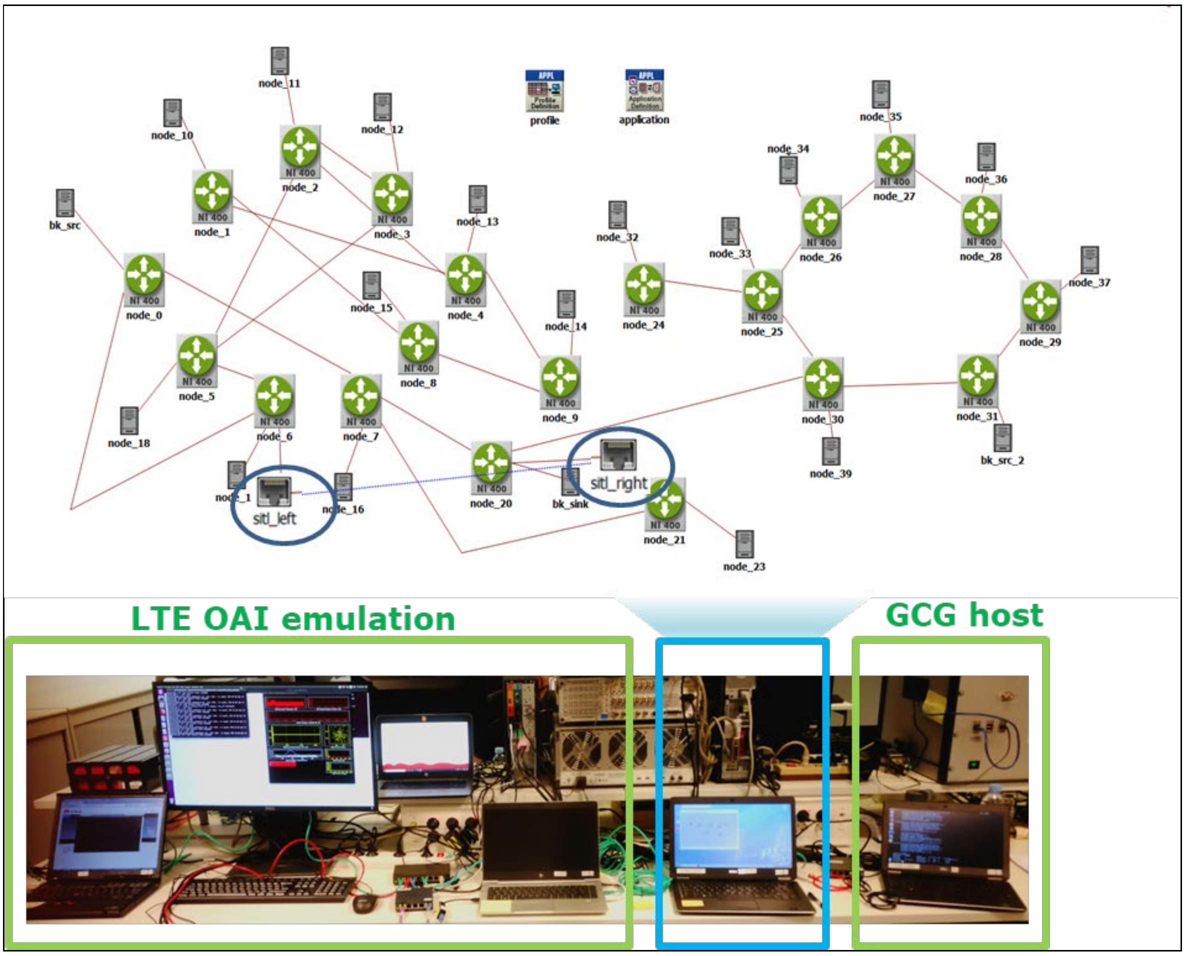

Figure 10.1: Backhaul network emulation platform (blue)

As shown in Figure 10.1, the real traffic from the LTE OAI emulation setup is injected via the *sitl_left* gateway. After transmitted through the backhaul network, the simulated traffic is converted and sent out to the GCG host via the *sitl_right* gateway. In this way, the real network traffic is converted into the simulated environment, and vice versa.

The designed backhaul network simulation is flexible in terms of:

1) can adjust the traffic parameters
2) can change types of devices/links;
2) can re-design the network topology;
3) can extend or reduce the number of nodes and the size of network.

As shown in the configuration of simulated backhaul traffic (Figure 10.2) and the results of monitored network performance (Figure 10.3), we have

1) injected simulated background traffic with mixed applications and mixed traffic load
2) collected analytical network performance results in respects of delay and throughput
3) integrated this simulated program with the physical setup via SITL interfaces (input – output traffic through the simulated network and together with simulated background traffic).

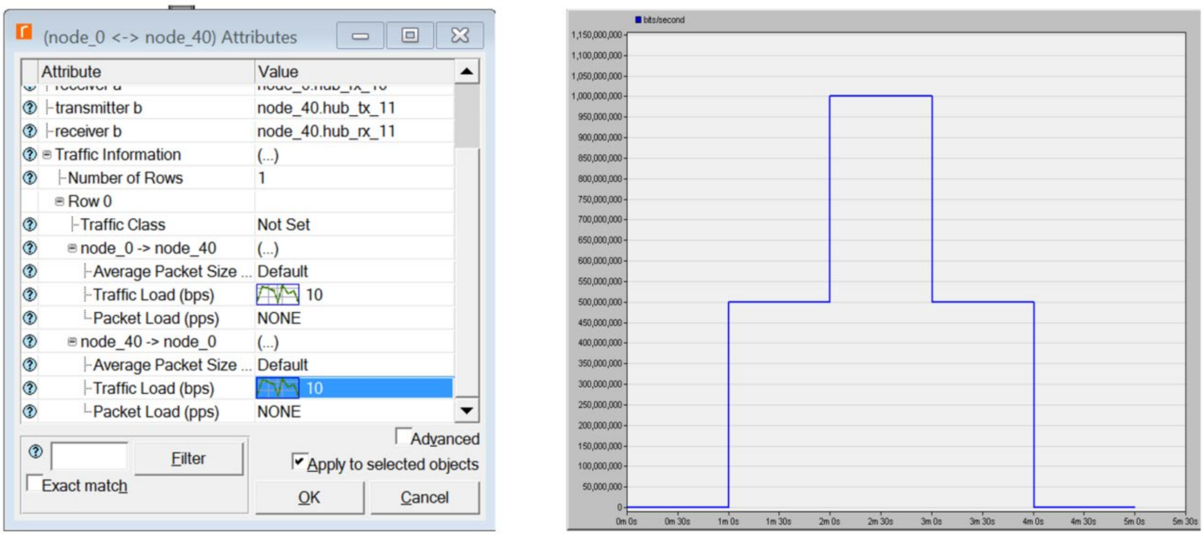

Figure 10.2: Configuration of background traffic fic load

In this example, the traffic load is assigned and changed along the simulation duration)

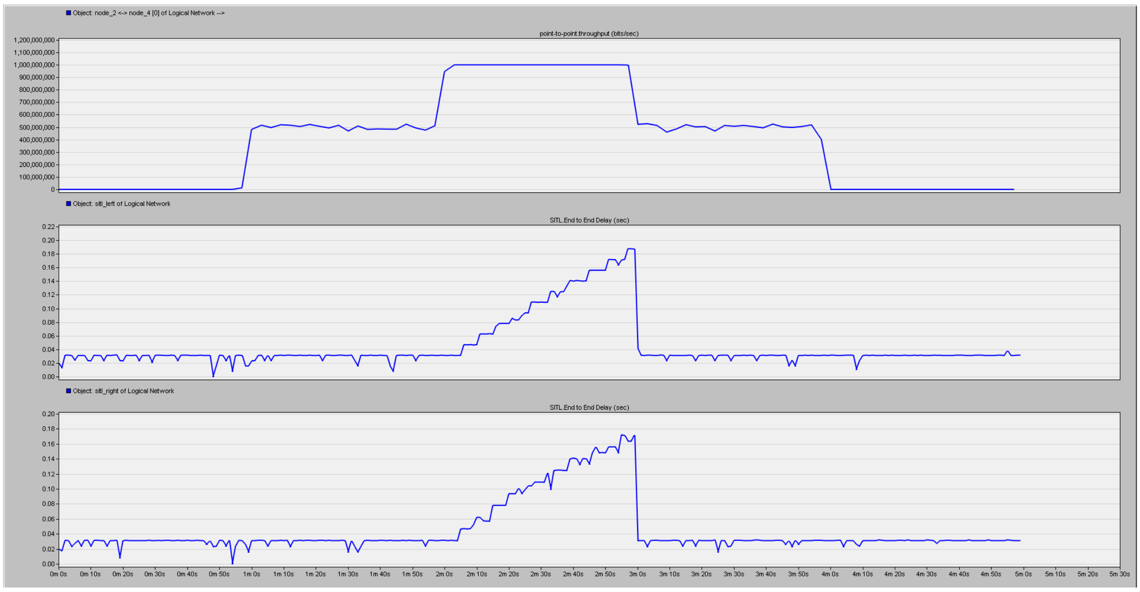

Figure 10.3: Result of throughput and delay

In this example, the upper figure shows the traffic throughput, the lower two figures illustrate the change of transmission delay due to the variation of traffic load)

11. Control infrastructure

Originally, all the configuration steps that have individually been described in each section, and that will be detailed in deliverable D3.4 "Operation manual of Radio access emulation tool (stream f)", had to be executed in the GUI of the corresponding device (GUI of the RF channel emulators, screen of the PCs in charge of implementing the OAI eNB/EPC, GUI of the satellite emulator… etc.)

In order to centralize this process and enable the configuration of the complete platform from a single device, a control infrastructure has been developed. Harnessing that all the devices in the emulation platform have both an operating system (either Windows or Linux) and an Ethernet port, a control infrastructure based on a remote desktop tool has been implemented.

In the final realization we have considered VNC (Virtual Network Computing) that allows controlling a remote server PC with a graphical desktop environment using a client PC. It makes managing files, software, and settings on a remote server easier for users who are not yet comfortable with the command line. Virtual Network Computing, or VNC, is an open source application that provides screen sharing services and is available for virtually all operating systems such as Windows, Linux, and of course OS X [4]. It has been necessary to install the tool in all the devices of the emulation platform and in the control PC. Besides, and in order not to interfere with the IP datapath under emulation, all the devices have been connected through a secondary Ethernet switch.

With this infrastructure it is possible to connect from the control PC to all the configurable devices and remotely execute the necessary configuration steps.

<u>Prerequisites:</u>
Switch
A server PC(s) with any version of Windows or Linux Ubuntu 16.04 (or 18.04)
A client PC with Linux Ubuntu or Windows
All PCs should be connected to the switch by an RG-45 cable, or to the same WI-FI access point. The diagram below in Figure 11.1 shows the overall network setup. Figure 11.2 shows the PC control screen with all the platforms connected (2 LTE and 1 Wi-Fi) and also the channel emulator. The same radio channel model is emulated at the same time as the train is running is the same environment (see GUI on the screen).

Each platform can be launch from the control PC thanks to all the windows on the screen that represent the different elements (Figure 11.2). This will allow to have the control PC in one room and the platforms in another room. Nevertheless, the OAI platforms generally needs some direct operator intervention.

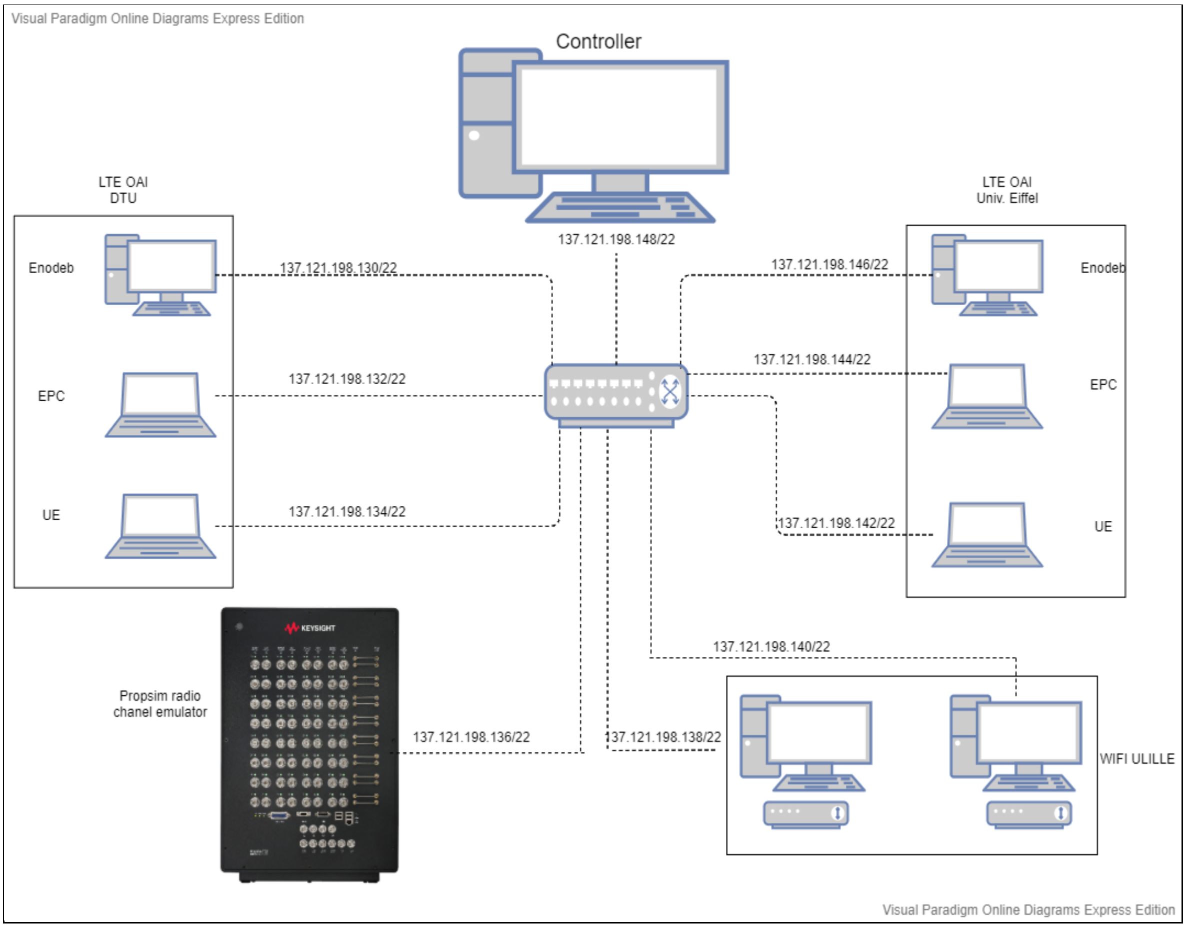

Figure 11.1: Network architecture for the global control

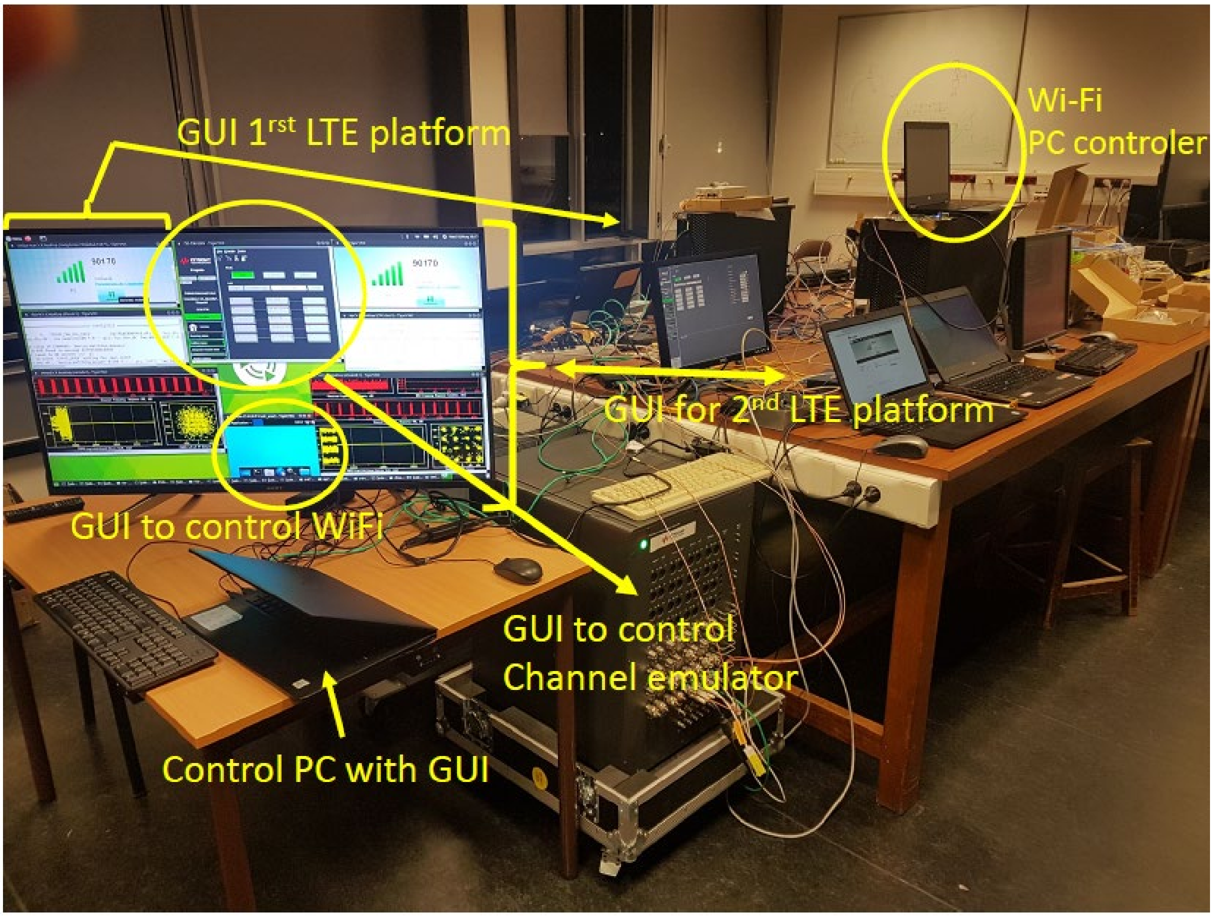

Figure 11.2: Client PC controlling remotely both LTE platforms (EPC, EPC and UE) and the 'Keysight' radio channel emulator

12. Conclusion

Deliverable D3.3, originally planned as a demonstration but replaced by this deliverable due to the sanitary situation, presents the final implementations of the different parts of the complete emulation platform that has been carried out. The proof of concept and integration of all the platforms is given by photography. It will also be demonstrated within the film of the project.

13. References

[1] EMULRADIO4RAIL DELIVERABLE D2.1, SOLUTIONS TO EMULATE THE RADIO BEARER (STREAM B), GRANT 826152, MAY 2019

[2] EMULRADIO4RAIL DELIVERABLE D3.1, HIGH-LEVEL DESIGN OF RADIO ACCESS EMULATION TOOL (STREAM B), GRANT 826152, OCTOBER 2019

[3] EMULRADIO4RAIL DELIVERABLE D3.2, EXPERIMENTAL ASSESSMENT OF THE COMMUNICATIONS BASED ON THE RADIO ACCESS EMULATION TOOL, GRANT 826152, NOVEMBER 2020

[4] HTTPS://WWW.SCIENCEDIRECT.COM/TOPICS/COMPUTER-SCIENCE/VIRTUAL-NETWORK-COMPUTING/PDF