edgeFLEX

D5.5

5G use case validation results in laboratory tests

The research leading to these results has received funding from the European Union's Horizon 2020 Research and Innovation Programme, under Grant Agreement no 883710.

Project Name	edgeFLEX
Contractual Delivery Date:	30.09.2022
Actual Delivery Date:	30.09.2022
Author(s):	Felix Maier, Alexandra Bach, Fiona Williams, Efe Demirok, Robert Farac, Tuana Ensezgin
Workpackage:	WP5 – Field Trials of Dynamically Controlled VPP Solutions
Security:	P = Public
Nature:	R
Version:	V1.0
Total number of pages:	50

Abstract
An extensive range of tests of edgeFLEX services and enablers were undertaken in the live 5G networks of Ericsson in an indoor laboratory setting with a live 5G network transmitting at low power. The latency of the transmission was the main parameter tested on standard 5G networks and with advanced prototypes of the forthcoming URLLC 5G networks not yet available on the market. In addition, Ericsson's 5G Device Management API proof-of-concept was integrated and tested with selected edgeFLEX platform components, and its features beneficial for DSOs and VPP operators were presented. The tests demonstrated that 5G can fulfil the requirements of the edgeFLEX services and enablers. Using the URLLC test infrastructure, latency was significantly reduced. Additionally, it was demonstrated that the 5G Device Management API improves the ease of use of 5G for power providers and security in data exchange between DSOs and VPP operators.

Keyword list
5G, URLLC, 5G performance tests, voltage and frequency control, inertia estimation, VPP optimisation, edgePMU, energy services, 5G Device Management API.

Disclaimer
All information provided reflects the status of the edgeFLEX project at the time of writing and may be subject to change.

Executive Summary

This document describes an extensive range of 5G performance and functionality tests performed in three infrastructures in Ericsson laboratory settings: a 5G standalone network, a 5G prototype network with Ultra-Reliable Low Latency Communications (URLLC) feature not yet available on the market and a 5G prototype network with the 5G Device Management API proof-of-concept. The established test infrastructures are explained in detail including the used components. In these infrastructures, representative data streams of edgeFLEX services and enablers were synthetically generated and transmitted over the 5G link. The latency of the transmission was the main parameter tested on the 5G standalone network and the prototype network with URLLC. Conclusions were also drawn on throughput of the communication links. The utilization of 5G and edge cloud shall enhance the edgeFLEX services and enablers with low communication latency and high reliability, supporting different traffic patterns of the services and enablers. Recommendations were provided in this deliverable on how to reduce latencies to a minimum. The performed tests aimed to show how the latencies of transmitted messages of edgeFLEX services and enablers were affected under different configurations. In addition, Ericsson's 5G Device Management API proof-of-concept was integrated and tested with some of the edgeFLEX platform components, and its features beneficial for power system operators and VPP operators were demonstrated with the newly implemented GUI.

The following test series were undertaken:

1. 5G performance tests of edgeFLEX services using synthetic data,
2. 5G performance tests of edgePMU data streams,
3. 5G performance tests of frequency control data streams with the edgeFLEX platform,
4. 5G Device Management API tests with the edgeFLEX platform.

In each test series, test cases were defined, and results of the tests were summarized and interpreted. The test results demonstrated that 5G can fulfil the requirements of the edgeFLEX services and enablers. Using the URLLC test infrastructure, radio link latency was significantly reduced to less than 2 ms. Optimisations in the edgeFLEX platform resulted in lower processing times and therefore lower round-trip times for frequency control services.

Additionally, it was demonstrated that the 5G Device Management API and its GUI improves the ease of use of 5G for power providers and reduces their need to interact with mobile network operators. New features introduced by the API enables DSOs and VPP operators to attach their devices to the 5G network easily, create device groups, set the QoS levels of the communications within their groups or for individual devices, and monitor the connection quality of their devices. These features can enable secure data exchange and flexibility trading in the future, which can increase the grid stability and the integration of more renewable energy assets into grid balancing.

Our 5G latency test results demonstrate that 5G enables a wide range of new innovative power grid services for VPPs, DSOs and TSOs and form the basis for future innovation. Our 5G API tests and demonstration show how the new functionality makes 5G easier to use for power system operators and VPP operators.

Authors

Partner	Name	e-mail
EDD		
	Felix Maier	felix.maier@ericsson.com
	Alexandra Bach	alexandra.bach@ericsson.com
	Fiona Williams	fiona.williams@ericsson.com
	Efe Demirok	efe.demirok@ericsson.com
	Robert Farac	robert.farac@ericsson.com
	Tuana Ensezgin	tuana.ensezgin@ericsson.com

Table of Contents

1. Introduction

1.1 Objectives of the deliverable and context of the tests undertaken

The objective of the work was to validate that 5G can meet the performance requirements of the edgeFLEX services and enablers developed in the project, to identify relevant new 5G features that would be beneficial to edgeFLEX services and enablers and energy providers, and to improve the ease of use of 5G for them by using a new 5G Device Management API. Selected services and the edgePMU enabler will be tested in field trials during the course of the project. The results reported in this deliverable provide recommendations and feedback to WP1, 2, 3, 4 and 5 so that the algorithms and their implementations as services supporting the edgeFLEX services and enablers could be optimised.

Additionally, the results provide representative data on the performance of 5G networks which partners could expect to be observed in a live 5G network operating at field trial sites and in the large-scale deployment of services commercially as exploitation of the project results in the market.

1.2 Structure of the deliverable

This deliverable reports on the observed performance and functionality of 5G networks in tests conducted in live 5G laboratory infrastructures.

The edgeFLEX services and enablers and the requirements they place on 5G networks are described in Section 2. We aimed to validate the requirements the services and enablers place on 5G in our tests.

The 5G laboratory infrastructures, the methods used to measure latency, and the limitations of test infrastructures are described in Section 3.

All test series conducted during the project including the 5G performance tests and the functionality tests with the 5G Device Management API are described in Section 4. Various test cases, their test conditions, the detailed results of these tests, and the analysis on the implications of the results are given as well.

In Section 5, the conclusions of our work and their future use in the project work is described.

In addition to the main sections, Annex A.1 provides formulas which take radio propagation models as input and provide the distance between sender and receiver as output. These formulas provide a rough approximation of how variations in attenuation of radio signals relate to the distance between the sender and receiver of data.

1.3 How to read this deliverable

The content of this deliverable can be understood once the reader has gained a basic understanding of the edgeFLEX project work. We recommend deliverable D3.1 before reading this deliverable to gather an overview and understanding of 5G in edgeFLEX as well as the 5G ICT requirements and solutions of edgeFLEX services and enablers. This deliverable describes the complete series of 5G laboratory tests conducted by EDD and no further tests have been planned in the project context.

For readers who want more detailed background information, the edgeFLEX deliverables D4.1 [1], D4.2 [2] and D4.3 [3] provide useful information on the edgeFLEX platform and the development of the edgeFLEX edgePMU.

2. The edgeFLEX services and enablers and their requirements on 5G

The 5G ICT requirements of the edgeFLEX services and enablers have been investigated in depth and reported in detail in D3.1. In this section, we provide a short summary of these services and enablers and an overview of their 5G ICT requirements as background information to the description of the 5G network performance tests of these services and enablers reported in this deliverable.

2.1 An overview of the edgeFLEX services and enablers

A range of edgeFLEX services and the edgePMU enabler was defined and developed within the project. A detailed description of the edgeFLEX services and enablers can be found in other deliverables. For simplicity, we provide a summary of the edgeFLEX services and enablers is given below:

- **Voltage Control** has the goal to avoid energy losses by providing continuous energy and by balancing the grid. It uses current and voltage measurements from simulators or from field devices in the power grid. To solve the voltage violation, setpoints are sent back to the asset.
- **Inertia Estimation** monitors the system's inertia in an unregulated and uncertain environment. Based on frequency and power measurements it gives back a single estimated value of inertia.
- **Frequency Control** services help to reduce the frequency variance by considering power provisioning from energy storages and renewable energy resources etc. It calculates setpoints which are sent back to renewable energy resources and energy storage assets.
- **Frequency Regulation Metering** assesses the frequency regulation that is provided by grid-connected devices and determines which modify the frequency and which do not. It uses the frequency measurements from the devices as an input and calculates the Rate-of-Change-of-Power (RoCoP).
- **VPP Coordinated Frequency Control** coordinates the Fast Frequency Regulation (FFR) offered at the device level by the Distributed Energy Resources (DERs) of the VPP with the purpose of stabilizing the power grid frequency. Based on power and frequency measurements, a control stream is computed and sent back to the DERs.
- **VPP Automatic Generation Control** aims to restore the grid frequency to the nominal value and keep the VPP power injection at the scheduled value. The service uses the reference power signal from the TSO and the power injections of DERs of the VPP as an input. A control signal with modified setpoints is distributed back to the DERs.
- **VPP Optimisation** is a tool to financially optimize the VPP operations and to maximize revenue. It takes several input arguments, e.g., weather information, flexibility schedules etc. Setpoints are sent back to the VPP to change the on-site equipment.
- **Advanced Flexibility Trading** is responsible for grid management and flexibility trading. It uses data relevant for trading from sensors, meters, actuators and energy management systems to define a flexibility offer. The flexibility offer is then provided to the devices that can be utilised by prosumers and DSOs.
- The **edgePMU** enables low-cost real-time data monitoring by taking raw voltage and current measurements and send them to the edge cloud for further processing. The algorithm deployed in the edge cloud processes the data, e.g., it calculates the phasor.

Within each edgeFLEX service and enabler, a range of specific scenarios and use cases have been considered in the course of the project. In this deliverable, we focus on typical data streams and communication aspects of the edgeFLEX services and enabler.

2.2 5G requirements of the edgeFLEX services and enablers

The most relevant requirements that edgeFLEX services and enablers state to communications were defined by telecom and power grid experts in the project. The requirements were defined in the context of the deployments of edgeFLEX services in power grids in the project field trials and in laboratory 5G networks. Note that the values expressed are the best guess that partners could provide at present. The requirements are summarized in Table 2-1. More detailed information about the requirements of the edgeFLEX services and enablers are specified in D3.1.

Table 2-1 5G requirements of edgeFLEX services and enablers

	Comm. Service Availability	Maximum End-to-End Latency	Bitrate [kbps]	Message Rate [Hz]
Voltage Control	99.999%	100 - 500 ms	> 100	0.017 – 0.1
Inertia Estimation	99.99%	100 ms	20 - 2000	10 - 1000
Frequency Regulation Metering	99.99999%	NA	> 12	10
VPP Coordinated Frequency Control	99.9999%	< 50 ms	> 6	10
VPP Automatic Generation Control	99.99%	NA	> 3	1
VPP Optimisation	99.99%	< 5000 ms	> 1	0.001
Advanced Flexibility Trading	99.99%	< 50 ms	> 100	0.067 - 1000
edgePMU	99.999%	< 20 ms	> 5000	1000 – 33.333

In the following section, we describe the 5G test infrastructures used and their relationship to full scale 5G networks deployed commercially. We provide the details of the tests and their results and relate them to the field trials of the project and later commercial exploitation.

2.3 Conclusion regarding the edgeFLEX services and enablers and their requirements on communications

The requirements depicted in Table 2-1 are related to current deployments and the current time intervals used to manage services in the power network and the current volume of messages associated with these services. In future, the requirements of edgeFLEX services and enablers on communications are expected to become more stringent, meaning that communication service requirements on availability, bitrates and message rates are expected to increase, while maximum end-to-end latency requirements are expected to decrease.

While 4G can fulfil most of the current requirements of the edgeFLEX services and enablers, as the requirements will evolve and become more stringent, the capabilities that 5G networks offer will be needed. The 5G performance tests reported in this deliverable have been undertaken in the context of the increasing communication service requirements of the evolving power network services.

3. 5G infrastructures for laboratory tests

To test the 5G performance of edgeFLEX services and enablers and to perform the functionality testing and end-to-end data transmission tests with the 5G Device Management API, Ericsson provided laboratory infrastructures with both standard and prototype 5G networks. The tests were conducted on the following three infrastructures provided by Ericsson:

- 5G standalone network
- 5G prototype network with URLLC feature
- 5G prototype network with the 5G Device Management API proof-of-concept

For the 5G performance tests, the test infrastructure was composed of 5G network products available on the market as well as 5G network prototypes (for URLLC). The products used that are available on the market have the same equipment as those which would be used by power network operators owning a 5G network meaning that our results are representative of the results of commercially deployed 5G networks. Power operators are buying and operating their own private networks, often in a combination with public networks, to support the operation of their power networks. Public communication network operators could own and operate the components described later in this section.

For the 5G Device Management API tests with the edgeFLEX platform including the functionality tests of the API features and the end-to-end data transmission tests with an edgeFLEX use case, a local area private 5G network including the API proof-of-concept implementation was used in the laboratory. The test setup with the 5G Device Management API consists of a prototype 5G network. This network does not have the same components as the standard 5G networks in the market, since the standardization activities related to 5G APIs for device management purposes are not finalised and are still ongoing. The prototype network used in edgeFLEX is the world's first implementation of such functionalities developed as an API.

A brief overview of the configuration and of the equipment of the test infrastructure as well as its limitations is provided in the sub-sections below.

3.1 The mobile network configurations implemented for the 5G tests

For the 5G performance tests, two test setups were developed. First, the 5G Standalone (SA) [4] infrastructure and secondly, the URLLC prototype infrastructure. Based on these setups, it is explained how the latency was measured. For the 5G Device Management API tests, one test setup was developed, which was used both for the functionality tests of the API features and the end-to-end data transmission performance tests with an edgeFLEX use case.

3.1.1 The 5G standalone infrastructure

For the performance tests of edgeFLEX services and enablers using synthetic data and for the tests with the edgePMU device, a 5G standalone infrastructure was deployed.

The network used the radio frequency band B78A and a bandwidth of 100 MHz. The radio signal transmission power was 1 mW and the uplink and downlink frequency bands were 3550 +/- 50 MHz. Several components were integrated:

- 5G edge cloud hosted in the 5G network
- 5G radio dot (see Figure 3-1)
- User Equipment (UE): 5G industry router (see Figure 3-2)
- Radio signal attenuator with coaxial cables
- PC with Intel Xeon E5430 (2.66GHz) CPU and Ubuntu 18.04 operating system
- Data generator (varies per test series)

- NTP server for time synchronization between PC and mobile network

Figure 3-2 Ericsson 5G Radio Dot. Source: Ericsson

Figure 3-1 5G industry router. Source: Ericsson

Experimenting with a range of radio conditions using attenuation devices

To simulate bad radio conditions, such as obstacles or large distances between UE and base station in the field, attenuators were installed in the previously explained setup. They were integrated into the communication link between the synthetic data generator and the edge cloud, i.e., between the 5G industry router and the radio dot using coaxial cabling. There were four variable attenuators with a basic attenuation level of 15dB. The attenuation was adjusted by turning regulators. In a range of attenuation levels, latency and packet loss were evaluated. The goal was to validate the performance of 5G under poor radio conditions.

In Annex A.1, we provide formulas which take radio propagation models as input and provide the distance between sender and receiver as output. These formulas provide a rough approximation of how variations in attenuation of radio signals relate to the distance between the sender and receiver of data in a line-of-sight scenario, i.e., no obstacles between sender and receiver.

3.1.2 The URLLC infrastructure

URLLC is the abbreviation for Ultra-Reliable Low Latency Communication. It is a new feature of 5G defined in Rel. 16 of 3GPP [5]. It provides low latency and ultra-high reliability for use cases which place requirements on reducing the latency of communications to a minimum. When URLLC capabilities are available in commercial 5G network products, 5G radio networks are expected to achieve an average latency of 0.5 ms and to successfully transmit a 32-byte message over the 5G radio link within 1 ms with a reliability of up to 99.9999%. Low latency is achieved with shorter time slots of 0.125 ms in the radio transmission and the possibility to prioritize data without waiting for slot boundaries. High reliability can be provided by extra-robust transmission modes and multi-antenna transmission for extra redundancy and packet duplication.

The URLLC test infrastructure consisted of the following URLLC prototype and edgeFLEX components:

- UE Cabinet attached to a first PC with Intel Xeon E5430 (2.66GHz) CPU and Ubuntu 18.04 operating system,
- gNB cabinet attached to a second PC with Ubuntu 18.04 with GUI as operating system, and
- Data generator (varies per test series).

3.1.3 The 5G Device Management API infrastructure

For the 5G Device Management API tests with the edgeFLEX platform including the functionality tests of the API features and the end-to-end data transmission tests with an edgeFLEX use case, a prototype 5G network for local area was used in the laboratory. The following components were integrated:

- Raspberry Pi 4 Model B hosting the data generator,
- 5G edge cloud hosting the data receiver,
- 5G Device Management API hosted in the 5G network,
- Graphical user interface (GUI) hosted in the 5G network,
- 5G radio dot,
- 5G industry router.

3.2 Latency measurements and synchronization

To simulate the data streams observed in the field trials, similar data streams were generated within the infrastructures described above. The transmission of data streams used very low power and test frequencies rather than the frequencies used by public mobile networks to avoid any interference with public network services. Their transmission in the 5G infrastructure was time stamped and recorded to determine the latency of the messages due to the radio transmission over the air and due to other components of the infrastructure, i.e., the time between the sending and the receiving of messages end-to-end. In this sub-section, we describe how latency measurements were conducted.

3.2.1 5G SA infrastructure

To validate the performance of the 5G SA network, measurements were made of the latency of the transmission of synthetic messages for each edgeFLEX service and for the edgePMU messages. We captured one-way latencies, meaning that messages in the uplink (sent from PC to the edge cloud) and downlink (direction vice versa) were traced separately. To measure the latency of the transmitted data, the messages were recorded with the Linux tool 'tcpdump' [6]. Two measurement points where timestamped messages were captured were placed in the setup: the first one on the PC interface towards the 5G industry router and the second one in the edge cloud.

The captured timestamps of the sent and received message, were then extracted from the recordings. Then, the latency was computed by subtracting the arrival times of the received messages from those of the sent messages. The resulting latencies can be considered as end-to-end latencies, which include delay times caused by the 5G industry router, the 5G radio access network and the 5G core network.

Timestamped data were needed for the measurement of one-way latency. Since the timestamps were extracted from two separate entities (PC and edge cloud), which were running on two different clocks, synchronization of those clocks was required. This was achieved by using the Network Time Protocol (NTP) [7]. NTP servers were installed in the network so that the cloud could directly obtain the clock information from the NTP server. To synchronize the PC, it was connected to the network using an ethernet cable and obtained the clock information from the same server. Thus, the offset for the clock information on the PC side was minimized.

To verify that the synchronization was working precisely, a test with ping messages was performed between the PC and the cloud. Ping requests with 1 second intervals were sent from the PC to the cloud. The latency of the ping requests was measured using tcpdump on the two measurement points. Moreover, the latency of ping responses, transmitted from the edge cloud to the PC, was calculated. Then, the latencies of corresponding requests and replies were summed. Finally, the sums were compared with the ping (round trip time) values. The average difference between the sum of one-way latencies and the RTTs was 0.01739 ms. The standard deviation of them was 0.0102 ms. This result verified the correct performance of the NTP synchronization for precise latency measurements.

3.2.2 URLLC infrastructure

In the URLLC infrastructure, the Round-Trip Time (RTT) was computed instead of one-way latency because no time synchronization could be implemented due to the lack of NTP servers in the gNB cabinet. The messages were recorded with ‘tcpdump’. The formula below highlights the composition of the RTT:

RTT = Radio link latency (UL and DL) + processing time

The processing time was determined by the edgeFLEX platform, and it was the time difference between the arrival of the packet and the sending of the corresponding Acknowledgement (ACK). The radio link latency was computed by subtracting the processing time from the RTT. Note that the resulting latencies cannot be considered as end-to-end latencies because the URLLC infrastructure did not host 5G core as a whole and a number of radio access network functions that are typical for commercial 5G networks.

3.3 Limitations of the test infrastructure

In commercially deployed 5G networks, the 5G performance is influenced by many factors which could not be reproduced in the laboratory test environment available for use for the tests. The following key differences between the test and a commercial field infrastructure can be summarized as:

- Since the distance between sender and receiver was fixed and one cell was used, **no handover** issues between multiple cells could be investigated. In field deployment, the distance between user and base station can change and this has an influence on the signal strength and other radio link characteristics.
- In the lab infrastructure, users have **optimal connection** to the antenna resulting in nearly no interference. Contrarily, real infrastructure suffers from reflections, refractions and scattering resulting in a delayed received message and interference.
- Field applications use a wide range of traffic types. In test environment, there is a **limited number of traffic types**.
- In the lab infrastructure, **small packet loss** occurred in contrast to a real environment and the traffic load on the test network was low. Testing under high load conditions was not performed.
- The standard 5G network used in these tests is **optimized for mobile broadband services** with focus on throughput performance, not for latency critical services.
- **End-to-end latency** measurements were undertaken, using only two measurement points, meaning that the effect of individual components on latency could not be investigated.

The following key differences and limitations between the test and a commercial field infrastructure were specific to the URLLC test infrastructure:

- **Round-trip latency** was measured. No one-way latency measurements were possible.
- A PC needed to be attached to the gNB cabinet to deploy software, **no edge cloud** environment was available.
- **Prototype hardware** was used as components, and it typically had a larger size than optimised products would have (large size of UE).

In the laboratory tests with the 5G SA infrastructure, the latest available hardware equipment was used. During the tests, the following limitations regarding the hardware components were observed:

- The highest possible **message rate of the MQTT data streams** is 1000 Hz. Above that rate, too many packets were aggregated by the 5G industry router, and evaluation of the results could not be performed.
- The **edgePMU signal-generator** software was used in the tests to simulate data traffic produced by an edgePMU hardware. The signal-generator deployed on a PC could not generate the highest sampling rate that the edgePMU hardware could generate. This limited the possibilities to test the full range of possible sampling rates that would be generated by the edgePMU hardware.

The following limitations were specific to the 5G Device Management API test infrastructure:

- The 5G network used during the tests of the 5G Device Management API proof-of-concept was a prototype network assembled specifically to provide and test this functionality. The prototype was assembled with modified commercial network components to enable new 5G Device Management API features.
- Protocol Data Unit (PDU) is a single data stream transmitted end-to-end from a 5G UE to a specific data network (e.g., internet or an energy application hosted on edge cloud). At present, many chips in 5G devices do not allow more than one PDU session for a device. Therefore, a device could only have one active connection (as well as a PDU session) in a device group created by the API to send its data to an application deployed on edge cloud. That's why, during the API tests, it was needed to move a device from one group to another to enable that device to send its data to the application on the new group. If devices did not have this limitation, it would be possible to make the device members in both groups and it could share its data with two applications at the same time.

3.4 Conclusions regarding the 5G test infrastructures for laboratory tests

The laboratory 5G SA and URLLC infrastructures used for the 5G performance tests were designed to enable 5G latency performance testing of the edgeFLEX services and enablers. The infrastructures provided state-of-the-art testbeds for the series of tests. The limitations of the test infrastructures mentioned above were those which are the normal limitations of infrastructures used for the laboratory testing of mobile networks. The test undertaken in these laboratory infrastructures provide representative results. Similar results could be expected in commercial mobile networks.

The laboratory infrastructure used for the 5G Device Management API tests was designed to enable exposure of device management capabilities of the 5G networks to power network service or users managing the system for power system operators through an API. The tests undertaken in this laboratory infrastructure demonstrate the functionality of new 5G features enabled through the API.

4. Test series of 5G performance and API functionality tests

This section describes the series of tests undertaken to validate performance and functionality of 5G for edgeFLEX use cases. First, the 5G performance tests of edgeFLEX services using synthetic data are described. Then the 5G performance tests of edgePMU are shown. The 5G performance tests of the edgeFLEX frequency control service tested with the novel 5G URLLC features are presented. Finally, the functionality and end-to-end data transmission tests with the 5G Device Management API are described. For each test series, the test cases, e.g., how the data was generated, the test sequence and the test results are summarized.

The following test series were conducted:

1. 5G performance tests of edgeFLEX services using synthetic data,
2. 5G performance tests of edgePMU data streams,
3. 5G performance tests of frequency control data streams with the edgeFLEX platform,
4. 5G Device Management API tests with the edgeFLEX platform.

The results of the test series focusing on latency are presented in three ways:

- The average is also known as the arithmetic mean indicating the sum of all samples collected and divided by the number of samples.
- The nth percentile is used as a second statistical measure which means that n% of all samples are smaller than this value. It is a measure of the probability that packets arrive within a time window, and it is closely linked to reliability.
- Improvements describe the difference between lowest and largest average value given in percentage which could be achieved when using optimal configurations compared to the weakest configuration. To show the improvements of the optimizations, the 99th percentile or the 99.9th percentile. The two figures are good measures of the distribution so that reliability can be analysed.

To ensure representative statistical results, two measures were undertaken: a high number of samples to minimize untypical behaviour and appropriate figures to interpret the results. Therefore, each test was repeated several times collecting between 3,000 to 10,000 samples in the 5G SA infrastructure and 100,000 samples in the URLLC infrastructure to demonstrate the high reliability. The latency results presented in this deliverable are given by different statistical means.

4.1 Test series 1: 5G performance tests of edgeFLEX services using synthetic data

This test series describes the 5G performance tests of the edgeFLEX services with synthetic data streams. The data streams were transmitted by utilising the MQTT protocol. One set of the tests was conducted in good radio conditions (test case 1) and another set of tests was conducted in poor radio conditions (test case 2).

A schematic diagram of the setup used in this test series is illustrated in Figure 4-1.

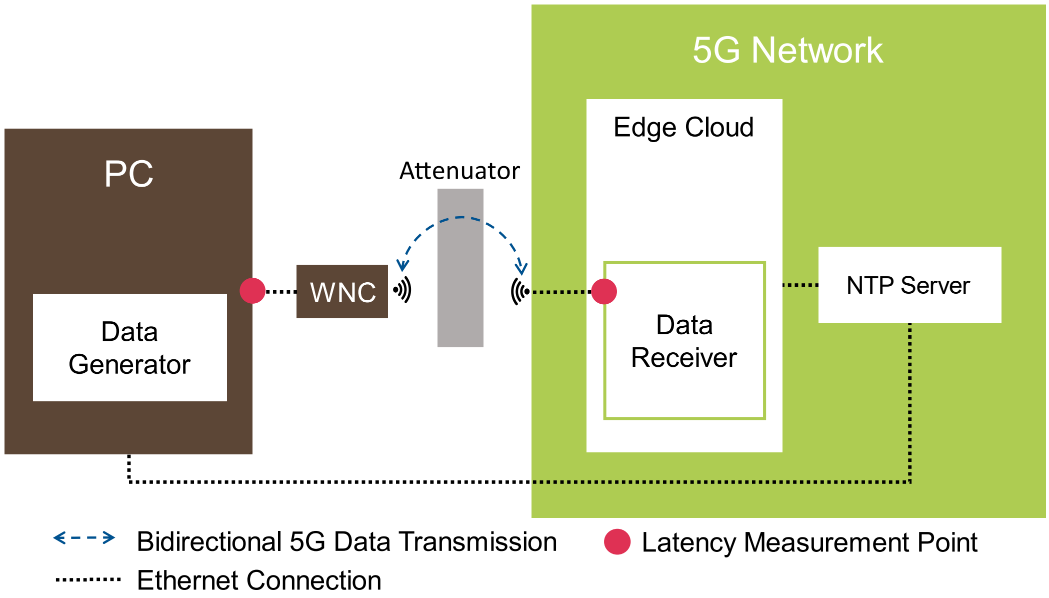

Figure 4-1 Schematic diagram of 5G SA infrastructure with attenuator

Objective of the test series

The objective of the test series was to evaluate the latency performance of 5G supporting edgeFLEX services both in good and in bad radio conditions.

Generating the data streams of the test series

The MQTT protocol works in a publish-subscribe architecture, meaning that a client transmits data by publishing and receives data by subscribing. A client can be publisher, subscriber, or both at the same time. The clients do not establish a point-to-point connection, but all clients are connected to a MQTT broker, and every message is sent to it. Thereby, the broker acts as a central unit and handles the data published by the clients by distributing it to the corresponding subscribers. [8] The technology utilised for the MQTT broker is provided by Eclipse mosquitto. [9]

In our setup, the PC was both the publisher and the subscriber. It generated packets using synthetic data as input. Then, the packets were sent via a wired link to the 5G industry router from which the packets were transmitted to the radio dot over air. Then, the packets were transmitted in MQTT messages over the 5G radio access to the 5G network. The broker was deployed in the edge cloud of the 5G network and received the messages. If a bidirectional link was to be tested, the broker sent the packets or control messages back to the subscriber on the PC.
Both uplink and downlink traffic were traced, so that measurements of one-way latency were performed separately. The end points used for the latency measurement were the 5G industry router interface of the PC and the external interface of the edge cloud.

The traffic patterns of the MQTT data streams were specified by message size and message rate. The message size is defined by the sample data received from the project partners. It ranged from 207 bytes up to 1163 bytes and it was fixed per edgeFLEX service. The message rate defines how many messages are generated per second. It will be specified in Hz, as the number of messages per second. To investigate the possibilities of the 5G link to handle more frequent packets, a higher message rate than specified in the requirements was used in some test cases. This will become more relevant for future power scenarios. The message rates used in the tests ranged between 0.1 and 1000 Hz. The resulting throughput of the transmitted data depended on the selected message rate. A higher rate resulted in more messages transmitted per second and therefore higher data throughput. The following Table 4-1 gives an overview of the traffic patterns of the five edgeFLEX services that were tested.

Table 4-1 Traffic patterns of the services used in the tests with synthetic data

edgeFLEX service	Communication Direction	Message Rates tested [Hz]	Message Size [bytes]	Expected max. Bitrate [kbps]
Voltage Control	Uplink, Downlink	0.1, 0.2, 1	299	0.67
Inertia Estimation	Uplink	50, 100, 200	202	345
Frequency Control	Uplink, Downlink	1, 10, 100, 1000	207	9.86
VPP Optimisation	Uplink, Downlink	1	253	2.2
Energy Flexibility Trading	Uplink, Downlink	0.1, 1, 10, 100, 1000	1163	4.82

4.1.1 Test case 1: 5G one way latency test in good radio conditions

4.1.1.1 Test conditions

- 5G SA infrastructure described in Section 3.1.1
- Synthetically generated MQTT data streams reflecting edgeFLEX services described in Section 2.1
- Measurement of one-way latency in uplink and downlink direction

4.1.1.2 Test sequence

- The tcpdump was started on the PC,
- The tcpdump was started on the 5G edge cloud,
- The MQTT broker and subscriber were started on the 5G edge cloud,
- A data stream was configured based on the message size and message rate requested by the service,
- The MQTT publisher was started on the PC,
- A specified number of MQTT messages were sent,
- The tcpdump was stopped on the PC and the cloud,
- The trace files for each test sequence were collected, and finally the
- Latencies for each test sequence were computed.

Several separate test sequences were conducted with the five edgeFLEX services and various message sizes and message rates provided in the table above.

4.1.1.3 Test results

Since the message rate was the only parameter to vary for each service, the best achieved latencies with the corresponding optimal message rate are summarized in the following table. In addition, the achieved improvement is specified except for the VPP optimisation service because only one message rate was simulated.

Table 4-2 Synthetic data test effect of message rate

Test Case	Comm. direction	Best average latency [ms]	Latency requirements [ms]	Optimal message rate [Hz]	Latency improvement with optimal message rate [%]
Voltage Control	UL	4.47	100 - 500	1	26.4
	DL	10.7		1	13.4
Inertia Estimation	UL	6	100	50	43.8
Frequency Control	UL	4.11	< 50	10	53.8
	DL	7.38		100	39.9
VPP Optimisation	UL	6.94	< 5000	1	Fixed message rate
	DL	11.2		1	
Energy Flexibility	UL	8.12	< 50	10	19.0
	DL	8.89		100	22.6

From the table it becomes clear that the 5G network can fulfil the latency requirements of all edgeFLEX services as the 5G network achieves average one-way latencies between 4.11 to 11.2 ms. Moreover, configuring the optimal message rate can reduce the average latency by up to 53.8%.

4.1.2 Test case 2: 5G one-way latency tests under a range of radio conditions

4.1.2.1 Test conditions

- 5G SA infrastructure described in Section 3.1.1 including an attenuator integrated between the 5G router and the radio dot
- Synthetically generated MQTT data streams reflecting the energy flexibility service as a representative for any edgeFLEX service with message rate of 10 Hz. 5000 messages were transmitted in each test round.
- Measurement of one-way latency in uplink and downlink direction

4.1.2.2 Test sequence

- The attenuation level was set,
- The tcpdump was started on the PC,
- The tcpdump was started on the 5G edge cloud,
- The MQTT broker and subscriber were started on the 5G edge cloud,
- A data stream was configured with the energy flexibility service and a message rate of 10 Hz,
- The MQTT publisher was started on the PC,
- 5000 MQTT messages were sent,
- The tcpdump was stopped on the PC and the cloud,
- The trace files for each test sequence were collected, and finally the

- Latencies for each test sequence were computed.

Several separate test sequences were conducted with a range of attenuation levels increasing the attenuation from 15 dB to 83 dB.

4.1.2.3 Test results

The attenuation is described by the attenuation level which consists of 15 dB basic attenuation and a variable attenuator. The equivalent distance to the attenuation level for a 5G frequency of 3.6 GHz is calculated to give a non-abstract understanding. The figure shall reflect a rough approximation of the distance between the 5G antenna and the device in a commercial deployment with direct line-of-sight propagation. The performance test results are displayed with the average latency and packet loss in uplink and downlink in the following Table 4-3.

Table 4-3 5G latency and packet loss performance under poor radio conditions

Attenuation Level [dB]	Equivalent distance [m]	Uplink		Downlink	
		Average Latency [ms]	Packet Loss Rate [%]	Average Latency [ms]	Packet Loss Rate [%]
15	0.373	10.43	0	8.21	0
45	11.8	9.14	0	8.54	0
75	373	10.85	0	8.86	0
77	469	9.02	0	8.52	0
79	591	8.6	0	8.51	0
81	744	8.51	0	8.81	0
82	835	8.7	0	8.46	0
83	937	8.83	0.06	10.09	0.34

When attenuation levels were increased stepwise from 15 to 82 dB, no increase of the packet loss was observed which proved the good performance of the 5G link under bad radio conditions. The average latency measured in downlink was almost constant around 8.5 ms for all attenuation levels while the average latency in uplink showed varying results between 8.51 to 10.85 ms. The tendency of lower latency for higher attenuation could be explained with the 5G network adapting to changing radio conditions by using more robust modulation and coding schemes for data transmission.

The connection of the radio link was lost when the total attenuation level reached to 83 dB. It was the only level where packet loss occurred. Shortly before the connection was lost a few messages were sent with an increased latency. After connection loss, no more messages were received at the edge cloud.

It was shown that 5G fulfils the latency requirements of edgeFLEX services specified in Section 2 even under poor radio channel conditions.

4.2 Test series 2: 5G performance tests of edgePMU data streams

In this series of tests, the edgePMU software was used to perform tests in a wide range of conditions.

For the tests with edgePMU data streams, the same underlying setup was used as for the tests of edgeFLEX services using synthetic data. However, there were adjustments made on the PC and the 5G network in terms of generating the data and configuration of the traffic patterns. The adjustments are described in the following. A schematic diagram of the adjusted infrastructure is illustrated in Figure 4-2.

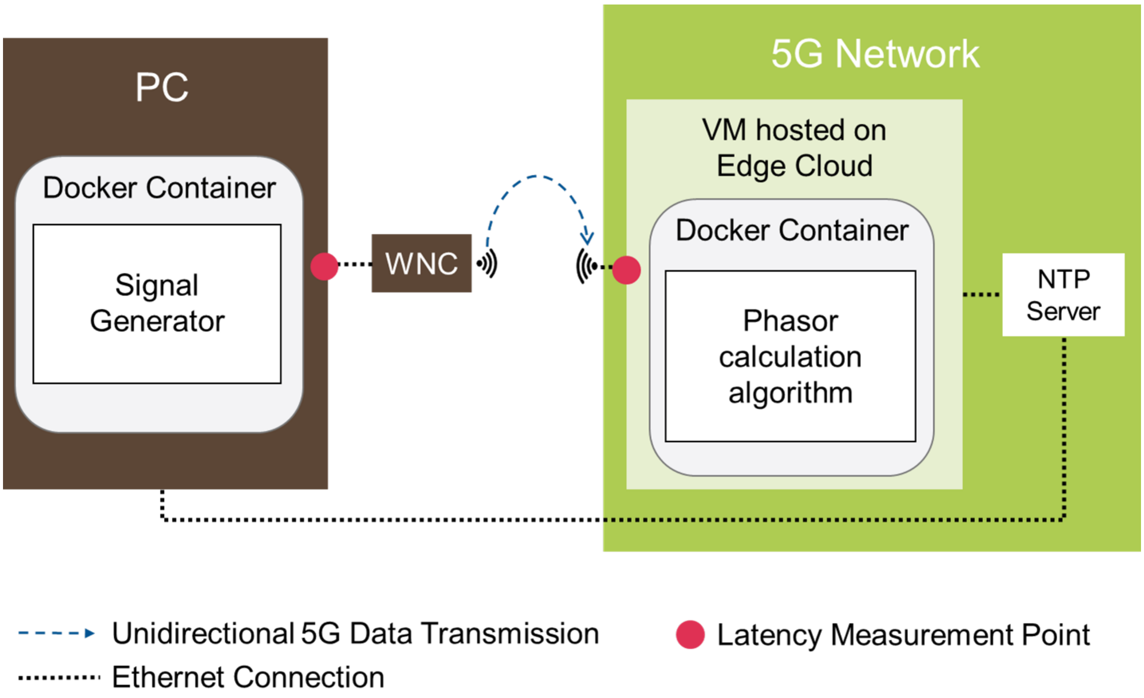

Figure 4-2 5G SA infrastructure for tests with edgePMU data streams

Objective of the test series

The objective was to show that 5G latency performance supports the requirements that the edgePMU has on communication. The edgePMU use cases set the most stringent requirements on communication latency among other edgeFLEX use cases.

Generating the data streams

A signal-generator software was provided by RWTH which generated signals typical of an edgePMU sensor deployed in the field. The software is based on Docker containers.

The Docker container for the signal-generator was deployed in the PC. Signals were generated on the PC using VILLASnode [10] software developed by RWTH. VILLASnode is a gateway which implements the exchange of simulation data in real-time. The generated signals were handed over to the communication socket, where the signals were transmitted as UDP packets. The packets were transmitted over the 5G radio access to the 5G network. The edgePMU algorithm, which was deployed on a Virtual Machine running in the edge cloud, received the packets and performed the phasor calculation. In a real deployment, the output of the phasor-calculation algorithm could be published via MQTT to the corresponding broker. The MQTT broker would typically be deployed outside of the VM for further data analysis. In our test infrastructure, we have analysed the traffic between the edgePMU device and the phasor-calculation algorithm in the edge cloud, and not the traffic between the algorithm and the control centre typically used by a power grid operator.

The latency measurement points used for the one-way latency measurement were the 5G industry router interface towards the PC and the interface of VM towards the PC. For the edgePMU, only uplink communication from the PC to the VM was used.

We changed two parameters in the signal generator: sampling rate and vectorize. Sampling rate defines how many samples are generated per second. In our test cases, it ranged from 1000 to 30,000 Hz. Sampling rates in real deployments in a power grid would exceed this number up to

60,000 Hz or more, but the PC used as signal generator could not handle such high load. The vectorize parameter defines how many samples one UDP message contains. UDP messages are eventually transmitted via the 5G link. A vectorize parameter of one sample per message results in a message size of 24 bytes. The MTU (Maximum Transfer Unit) of UDP is 1472 bytes, meaning that one UDP message can have a maximum length of 1472 bytes. Therefore, the highest value of vectorize in our tests was 50, resulting in a message size of 1200 bytes. Every message size exceeding this limit would result in packet fragmentation which made it infeasible to evaluate a representative amount of latency samples. The throughput is the product of the message size and the message rate (quotient of sampling rate over vectorize). Compared to the synthetic data tests, the edgePMU tests required larger message rates which resulted in higher throughputs. An overview of all traffic patterns is given in Table 4-4.

Table 4-4 Traffic patterns of edgePMU data streams used in the test

Sampling rate [samples/s]	Vectorize [samples/message]	Message size [bytes]	Message rate [messages/s]	Throughput [kbps]
1,000 - 30,000	1 - 50	24 - 1,200	20 - 1,000	192 - 5,760

4.2.1 Test case 1: 5G one way latency test with edgePMU data streams

4.2.1.1 Test conditions

- 5G SA infrastructure described in Section 3.1.1
- UDP data streams generated by the edgePMU signal-generator software
- Measurement of one-way latency in uplink direction

4.2.1.2 Test sequence

- The tcpdump was started on the PC,
- The tcpdump was started on the VM in the 5G edge cloud,
- The phasor calculation algorithm was started on the VM in the 5G edge cloud,
- The signal generator was configured based on sampling rate and vectorize,
- The signal generator was started on the PC,
- A specified number of UDP messages were sent,
- The tcpdump was stopped on the PC and the edge cloud,
- The trace files for each test sequence were collected, and finally the
- Latencies for each test sequence were computed.

A range of tests was conducted with various parameters of sampling rate and vectorize. In the first test round, the vectorize parameter was fixed to 50 and the sampling rate was increased from 1000 to 30,000. At 30,000, the signal-generator’s output showed that a lot of samples were missed during generation, because the limit of the PC hosting the signal-generator was reached. That is why, we could not increase it above 30,000, despite the sampling rate in real deployments of the edgePMU would go up to 60,000 or higher. In the second test round, the sampling rate was fixed to 1000 and vectorize was decreased from 50 to 1.

4.2.1.3 Test results

The test results are presented in Table 4-5 and Table 4-6 with the average and the 90th percentile latency for a fixed vectorize parameter and a fixed sampling rate.

For a fixed vectorize parameter, the smaller sampling rates showed the best latency results with an average latency around 9 ms and 90th percentile values below 11ms. Sampling rates of 7500 and higher had average latencies from 11 to 12 ms. Their 90th percentile latencies were between 15 and 16 ms. The results show that higher sampling rates led to higher latencies. After choosing a lower sampling rate, we observed a maximum improvement of 30.8% compared to higher sampling rates.

Table 4-5 Average latency for fixed vectorize parameter of 50

Sampling rate	1000	2000	5000	7500	10000	12500	15000	20000	30000
Average latency [ms]	8.89	8.42	9.28	10.54	12.2	11.5	11.6	11.8	10
90th percentile [ms]	10.9	8.7	10.7	15.0	16.3	16.0	15.6	16.0	15.1

For a fixed sampling rate of 1000, the lower vectorize parameters had an average latency around 4 ms and 90th percentile between 4 ms and 5 ms. Higher vectorize led to average values up to 9 ms and 90th percentile values which were between 0.5 ms and 2 ms higher. By choosing an optimal vectorize parameter, an improvement of 57.9% was achieved.

Table 4-6 Average latency for fixed sampling rate of 1000

Vectorize	1	5	10	40	50
Average latency [ms]	4.34	3.74	6.03	7.80	8.89
90th percentile [ms]	5.14	4.16	7.67	8.35	10.9

In general, it can be concluded: the lower the vectorize and the lower the sampling rate, the lower the average latency. The optimal configuration for low latency is provided in Table 4-7. It was therefore demonstrated that 5G fulfils the challenging latency and throughput requirements of the edgePMU.

Table 4-7 Optimal configuration of edgePMU for lowest latency

Sampling rate	**Vectorize parameter**	**Minimum average latency**	**Maximal improvement**
1000	5	3.74 ms	57.9%

4.3 Test series 3: 5G performance tests of frequency control services

In this series of 5G performance tests, the edgeFLEX platform [2] was used. In contrast to test series 1, here the data streams were generated by the edgeFLEX platform simulation tool developed by project partner WIT [1]. For that, the edgeFLEX platform was deployed in two test infrastructures, the 5G standalone infrastructure as well as in the URLLC infrastructure. The edgeFLEX platform deployed in the infrastructure comprises the following components:

- Data generator tool to simulate three different frequency control services:
  - AGC: VPP Automatic Generation Control
  - COORD: VPP Coordinated Frequency Control
  - ROCOP: Frequency Regulation Metering (Rate-of-Change-of-Power)

- Frequency control algorithms and edgeFLEX backbone: edgeFLEX databus (hosts MQTT broker), edgeFLEX persistence (data storage), edgeFLEX visualization (via Grafana)

Objectives of the test series

The latency performance of the edgeFLEX platform in standard 5G test network and the URLLC prototype network were under investigation. The goal was to show that the frequency control services can be supported by the latency performance of standard 5G and URLLC networks and compare the performance of both infrastructures. Additionally, the overall RTT of the data communications was tested and optimised.

The test results of the test cases in this test series are not presented with one-way latency as in the test cases before. Since no time synchronization could be implemented in the URLLC infrastructure, the two-way radio link latency (sum of UL and DL latency) was measured in both infrastructures. Additionally, the Round-Trip Time (RTT) was computed and analysed. The RTT includes the radio link latency and the processing time in the edgeFLEX platform (see more in Section 3.2).

Generating the data streams of the test series in the standard 5G SA infrastructure

The data generation in the standard 5G network worked very similar as in the previous test series 1 with synthetic data streams. First, the MQTT messages were generated by data generator deployed on the PC. Messages were transmitted via the 5G connection to the MQTT broker hosted by the edgeFLEX databus in the edge cloud and sent back from the cloud to the PC in downlink. A schematic diagram of the 5G SA infrastructure including components of the edgeFLEX platform is shown in Figure 4-3.

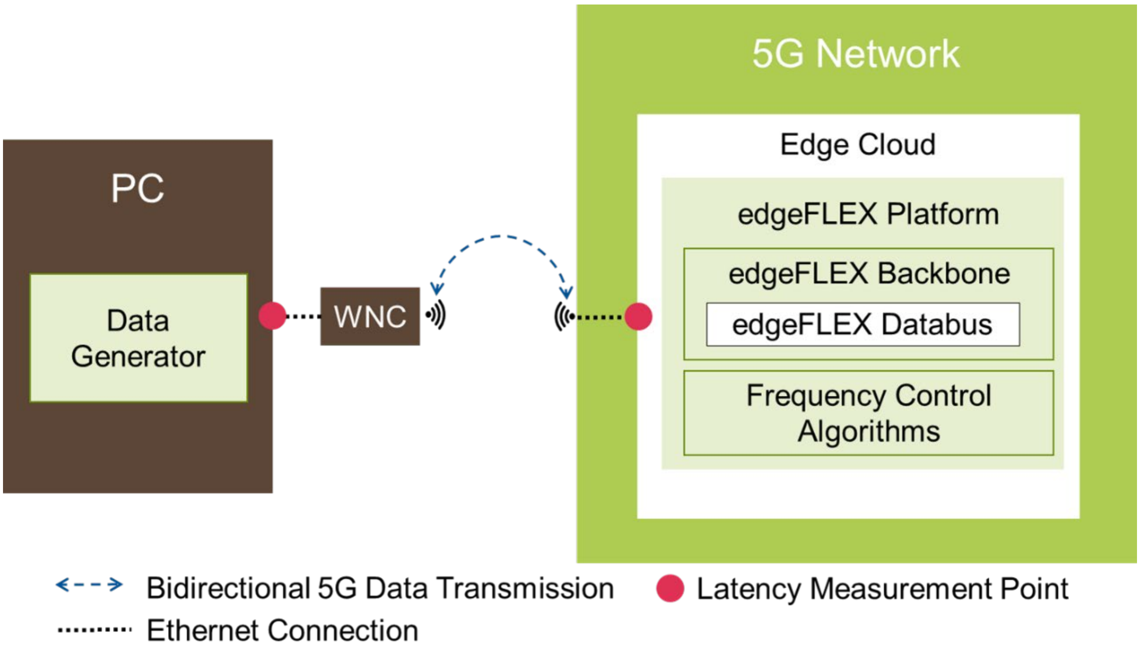

Figure 4-3 Schematic diagram of edgeFLEX platform deployment in 5G SA setup

Generating the data streams of the test series in the URLLC infrastructure

As a second infrastructure to test the three services of frequency control, the URLLC infrastructure was used. The PCs attached to the UE and gNB cabinet hosted the relevant software. The data generator was hosted by the first PC, and it generated the service specific MQTT messages which were sent via the MQTT client to the edgeFLEX platform in uplink. The edgeFLEX platform on the gNB side hosted the databus as well as the edgeFLEX backbone and the frequency control algorithms. The MQTT broker in the edgeFLEX databus received the messages and forwarded them to the visualization components of the edgeFLEX platform and the service algorithms. The algorithms performed the calculations and generated the control messages. In downlink, the control messages were sent back from the edgeFLEX databus to the first PC. A schematic view of the URLLC infrastructure including components of the edgeFLEX platform is shown in Figure 4-4.

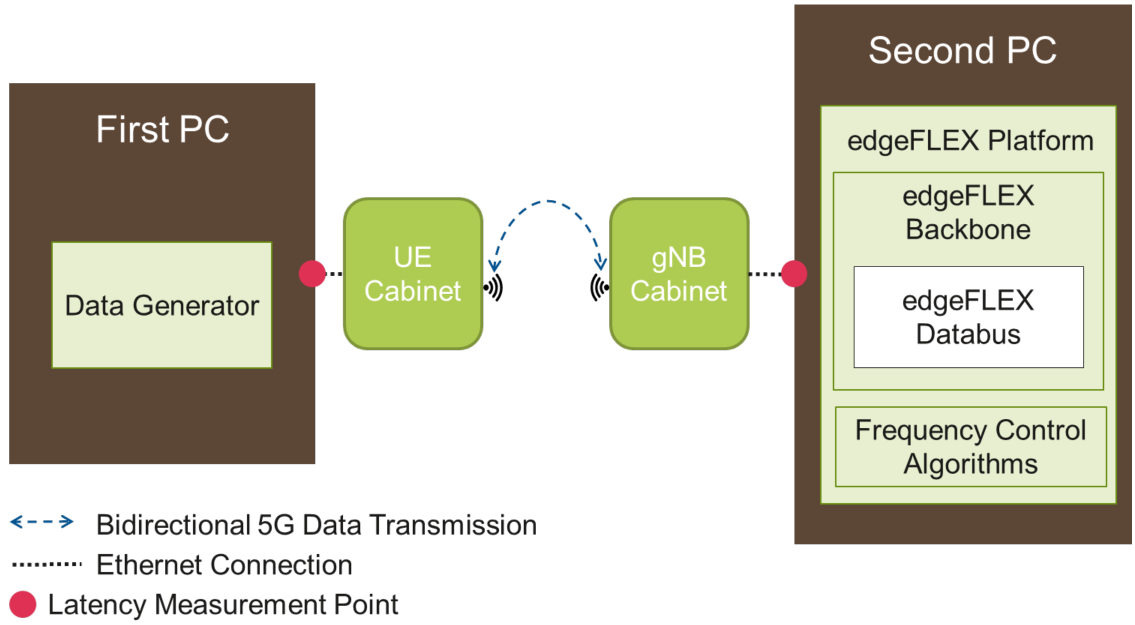

Figure 4-4 Schematic Diagram of edgeFLEX platform deployment in URLLC setup

Traffic patterns of frequency control data streams

In this test series, each frequency control service showed specific traffic patterns predefined by the data generator. The following patterns were generated by the data generator:

Table 4-8 Traffic patterns of frequency control services

	AGC		COORD		ROCOP	
	UL	DL	UL	DL	UL	DL
Message Size [bytes]	237	718	222	175-275	219-249	179-300
Message rate [Hz]	100		50		100	
Throughput [kpbs]	15	57	56	71	37	82

4.3.1 Test case 1: 5G two-way radio latency test of frequency control data streams in 5G SA

4.3.1.1 Test conditions

- 5G SA infrastructure described in Section 3.1.1
- MQTT data streams of frequency control services generated by the edgeFLEX platform
- Measurement of two-way latency

4.3.1.2 Test sequence

- Deployment of the edgeFLEX platform in the 5G SA infrastructure,

- The tcpdump was started on the PC,
- The tcpdump was started on the VM in the 5G network,
- The data generator was configured based on the service under investigation,
- The data generator was started,
- Each service was repeated multiple times to collect an appropriate number of messages,
- The tcpdump was stopped on both PCs,
- The trace files for each test sequence were collected, and finally the
- Two-way latencies for each test sequence were computed.

Test results

The average two-way latencies that 5G SA provided for the three frequency control services were in the range of 11.09 ms (COORD) to 12.32 ms (AGC) as illustrated in Table 4-9. The results are comparable to the results shown in test series 1 as the traffic patterns were similar and the same protocol MQTT was used.

Table 4-9 Latency results of frequency control services in 5G SA infrastructure

Frequency control service	COORD	ROCOP	AGC
Average two-way latency [ms]	11.09	11.66	12.32

4.3.2 Test case 2: 5G two-way radio latency test of frequency control data streams in URLLC

4.3.2.1 Test conditions

- URLLC infrastructure described in Section 3.1.2
- MQTT data streams of frequency control services generated by the edgeFLEX platform
- Measurement of two-way latency

4.3.2.2 Test sequence

- Deployment of the edgeFLEX platform in the URLLC infrastructure,
- The tcpdump was started on the UE PC,
- The tcpdump was started on the gNB PC,
- The data generator was configured based on the service under investigation,
- The data generator was started,
- Each service was repeated multiple times to collect an appropriate number of messages,
- The tcpdump was stopped on both PCs,
- The trace files for each test sequence were collected, and finally the
- Two-way latencies for each test sequence were computed.

4.3.2.3 Test results

In the tests of the three frequency control services in the URLLC infrastructure, the average two-way radio link latency was below 2 ms for all three services. Likewise, the 99.9th percentile of the two-way radio latencies lay in a slightly higher range but still below 2 ms as illustrated in Table 4-10. It shows that the URLLC feature of 5G provides high reliability to receive messages within a certain timeframe, especially relevant for time sensitive services.

Table 4-10 Latency results of frequency control services in URLLC infrastructure

Frequency control service	COORD	ROCOP	AGC

Average two-way latency [ms]	1.58	1.63	1.67
99.9th percentile of two-way latencies	1.80	1.70	1.72

4.3.3 Test case 3: Comparison of 5G two-way latency tests with frequency control data streams between 5G SA and URLLC

Test conditions and test sequences were described in the previous two test cases.

4.3.3.1 Test results

When comparing the two-way radio link latencies of the two infrastructures used for the tests with data streams of edgeFLEX services, the expected outcome of the test case was yielded, showing that URLLC was able to significantly reduce latency compared to the 5G SA infrastructure. Figure 4-5 proves that the URLLC infrastructure achieved much lower average latencies than the 5G SA infrastructure. Average two-way latencies of 5G SA were in the range of 11.1 to 12.3 ms while URLLC was able to provide figures in the range of 1.6 ms for all three frequency control services.

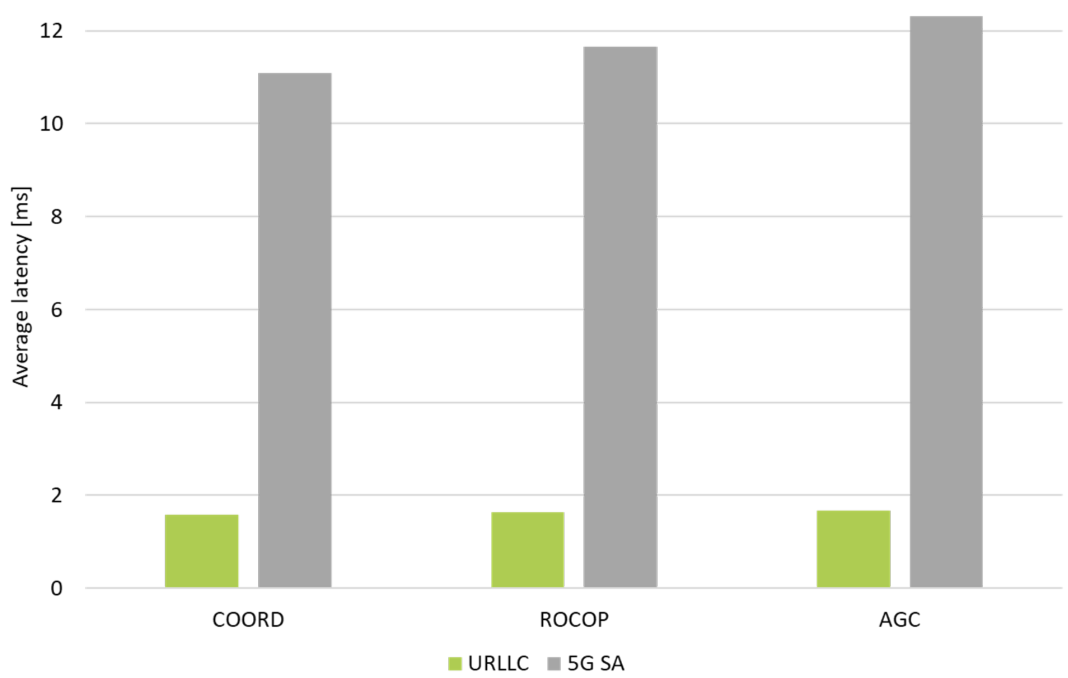

Figure 4-5 Comparison of average two-way radio link latencies in 5G standard network and URLLC

When validating the range of the radio link latency distribution from minimum to maximum, we observed that the variance of the URLLC infrastructure was significantly reduced. Whilst the 5G SA infrastructure yielded a variance of up to 22.62 ms the URLLC infrastructure achieved a much lower variance of 5.96 ms.

This high performance of URLLC was possible thanks to the focus on latency by reducing transmission slots in the radio among other new URLLC features that are not implemented in the standard 5G SA infrastructure. The 5G SA infrastructure could not compete with those results because it is optimised for high throughput as it is an eMBB (enhanced Mobile Broadband) system. Additionally, it hosted a core network and radio access network functions used in market mobile network solutions which the URLLC prototype infrastructure did not contain.

4.3.4 Test case 4: RTT test with frequency control data streams in URLLC

Test conditions and test sequences were analogous to those described in test case 2. In this test case, however, the focus is on the RTT, which is the sum of the two-way radio link latency and the processing time of the platform.

4.3.4.1 Test results

Looking at the RTTs in contrast to the radio link latency, illustrated in Table 4-11, we saw significant increases caused by the processing time. In the test with the COORD service, the average RTT was 2.75 ms. The 99.9th percentile of RTTs was 26.4 ms. In ROCOP, the average RTT was higher around 6 ms with a 99.9th percentile around 31ms. In AGC, the smallest values were observed. The average RTT was around 3 ms. The 99.9th percentile was around 4 ms.

Table 4-11 RTT results of frequency control services in URLLC infrastructure

Frequency control service	COORD	ROCOP	AGC
Average RTTs [ms]	2.75	6.03	3.19
99.9th percentile of RTTs	26.4	30.6	4.27

Figure 4-6 proves that the processing time adds up a major part of the RTTs in the 99.9th percentile, comprising 60% of the AGC service and 90% in the COORD and ROCOP services. To optimize the overall RTTs of the frequency control services, processing times will be optimized in test case 5 of this test series.

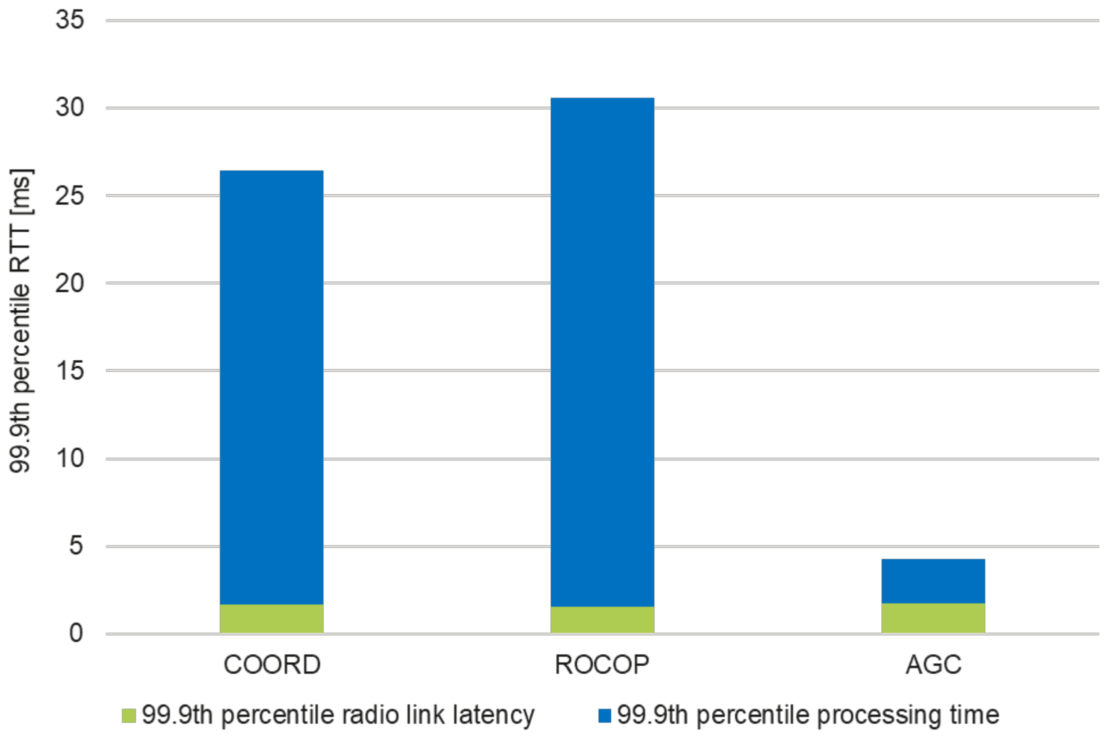

Figure 4-6 99.9th percentile processing time and radio link latency in default URLLC configuration

4.3.5 Test case 5: Optimising RTT with frequency control data streams in URLLC

To improve the results of test case 4, the impact of disabling Nagle's algorithm was investigated. This algorithm is used in TCP/IP protocols and thus also in MQTT. It is responsible for the congestion control and limits the transmissions by aggregating multiple packets instead of sending each directly and flooding the channel. To disable the algorithm, the 'set_tcp_nodelay' option in the mosquitto configuration of the MQTT broker was set to true [9].

Test conditions and test sequences were analogous to those described in test case 2.

4.3.5.1 Test results

The results of this test case are presented in Table 4-12 showing the average RTTs for all services in the range of 2.53 to 3.13 ms and RTTs in the 99.9th percentile in the range of 3.17 to 4.2 ms.

Table 4-12 Optimised RTT results of frequency control services in URLLC infrastructure

Frequency control service	COORD	ROCOP	AGC
Average RTTs [ms]	2.53	2.65	3.13
99.9th percentile of RTTs	3.17	3.16	4.2

In contrast to the results of test case 4, the outliers and the 99.9th percentile of the processing times were significantly reduced for the services COORD and ROCOP, illustrated in Figure 4-7. It shows that disabling Nagle's algorithm was beneficial and had a positive impact, i.e., reducing the processing time in the platform in the 99.9th percentile down to 1.5 ms for the ROCOP service and down to 1.4 ms for the COORD service.

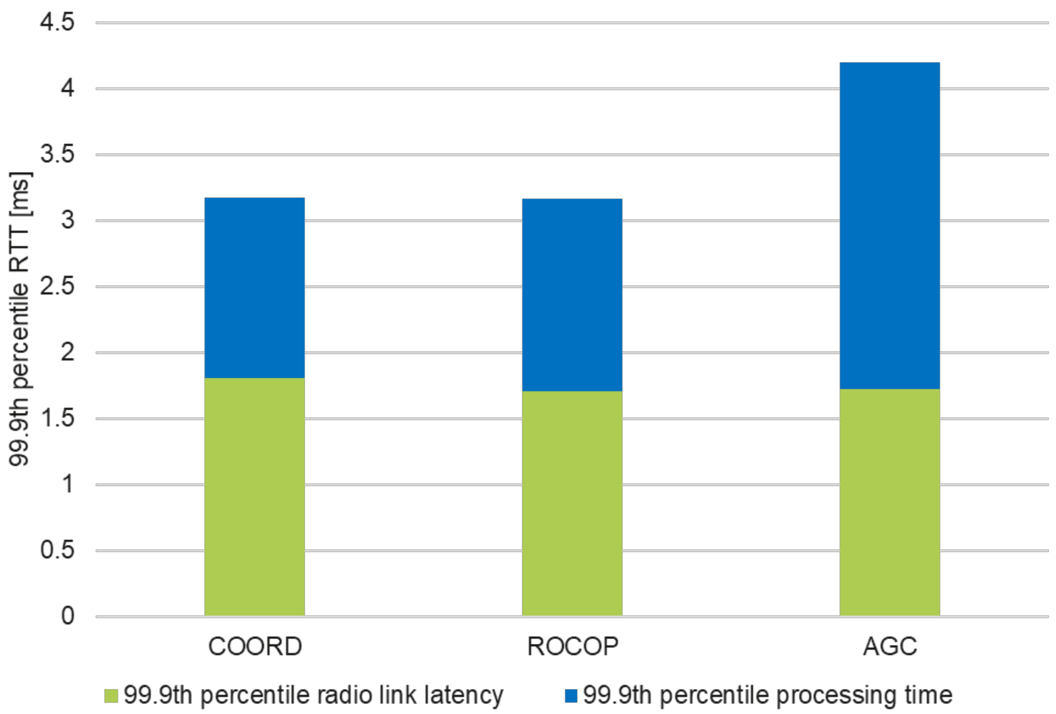

Figure 4-7 99.9th percentile of RTTs in URLLC with disabled Nagle's algorithm

4.3.6 Test case 6: Comparison of tests with frequency control data streams between optimized and non-optimized RTTs

Test conditions and test sequences were analogous to those described in test case 2.

4.3.6.1 Test results

The biggest improvements were observed in the 99.9th percentile of the processing times as depicted in Figure 4-8. Consequently, the RTTs were reduced as well. For the ROCOP service, the RTTs in the 99.9th percentile were reduced by up to 90% going from 31 ms down to 3.2 ms. The 99.9th percentile of RTT with the COORD service was also lowered from 26.4 ms down to 3.2 ms while the AGC service showed an equal RTT of 4.2 ms.

After disabling Nagle's algorithm, the measured average RTT was lower thanks to the reduced processing times in the edgeFLEX platform while the average radio link latency stayed around 2 ms. Especially, the average RTT of the ROCOP service was reduced from 6 ms to 2.7 ms corresponding to a reduction of 55% when Nagle's algorithm was disabled. The COORD service had an average RTT around 2.5 ms and the processing time comprises 40% of the RTT after optimisation. Results of the average RTT of the AGC service remained nearly the same as for the configuration with enabled Nagle's algorithm, in the range of 3 ms. Since the AGC service achieved only minor reduction (2%) of processing time, this service seemed to be independent from the influence of the Nagle's algorithm.

Comparing the results to the previous test cases 1 to 4, it becomes clear that the URLLC infrastructure with disabled Nagle's algorithm achieved the overall best performance in terms of RTTs. The effect of disabling the algorithm means that packets are not aggregated which increases the load on the network.

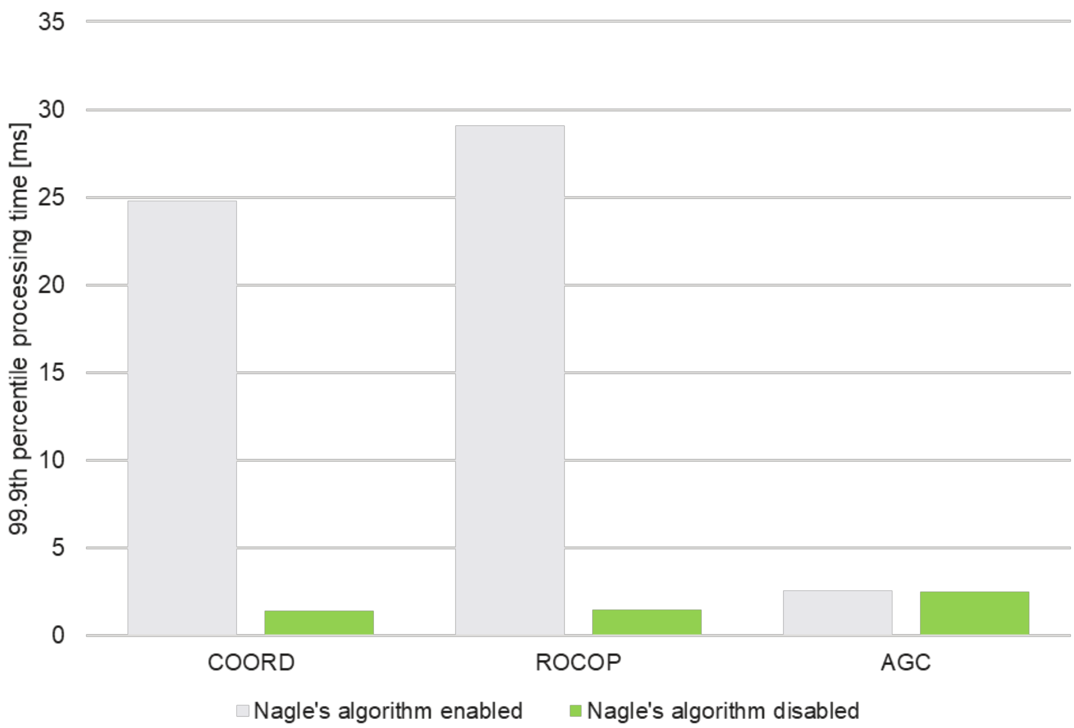

Figure 4-8 Comparison of the 99.9th percentile processing times with and without Nagle's algorithm

4.4 Test series 4: 5G Device Management API tests with the edgeFLEX platform

This sub-section describes the functionality tests performed with the 5G Device Management API in relation with edgeFLEX services as well as the end-to-end data transmission tests with synthetic data streams of edgeFLEX services. To demonstrate the features of the 5G Device Management API and evaluate the potential benefits of the API on the energy sector, some components from the edgeFLEX platform were integrated to the 5G Device Management API and test data flows were simulated related to the edgeFLEX services. A schematic diagram of the setup used in these test series is illustrated in Figure 4-9. The overall test setup enables to manage and monitor the 5G connectivity of devices, establish end-to-end communications, and transmit the synthetic data streams from 5G UEs to the edgeFLEX platform components.

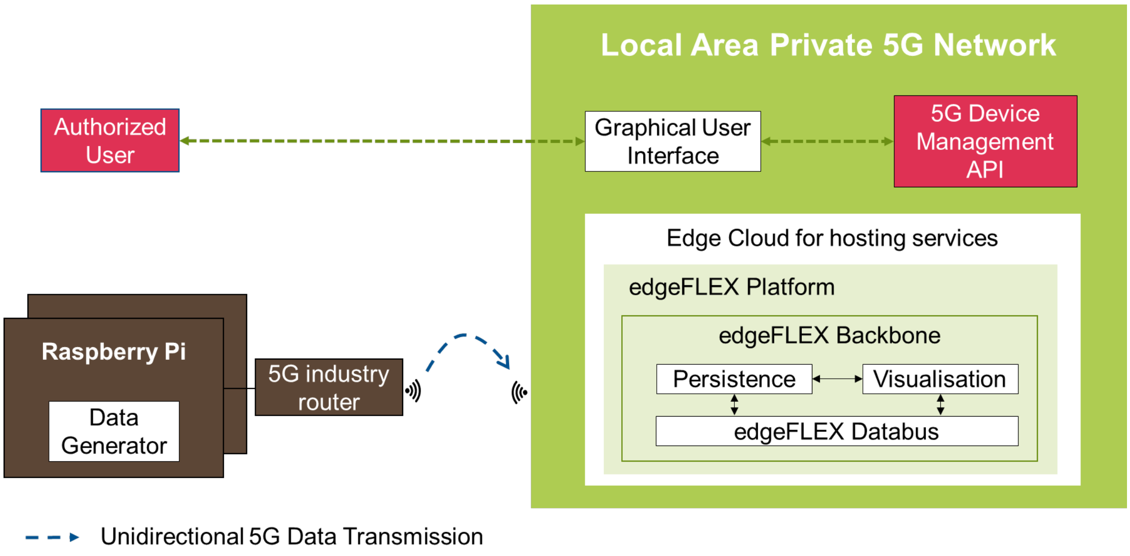

Figure 4-9 The laboratory infrastructure for the 5G Device Management API functionality tests and end-to-end data transmission tests with the edgeFLEX platform

In the laboratory, a local area private 5G network having the 5G Device Management API proof-of-concept, multiple Raspberry Pis connected to 5G industry routers forming the 5G UEs, and the edgeFLEX platform components were deployed and used. In this prototype network, it was possible to change the 5G network configuration to develop and test new 5G Device Management API features. This network was a shared network with other colleagues, therefore any configuration change in the network was affecting the progress of the work.

The required 5G connectivity during the tests was established with the use of the 5G Device Management API graphical user interface (GUI) as seen in Figure 4-9. The green line indicates the data transmission between the authorized user, the GUI, the 5G Device Management API and 5G core network functions. It gives the user the capability to be able to change configurations in the 5G network, manage the connectivity of devices and monitor the connectivity status of devices. Before being able to send a data from a 5G UE to the edgeFLEX platform, it was required to use the 5G Device Management API GUI to attach devices to the 5G network, create device groups and establish all end-to-end connections.

While establishing the required connections, the functionality of relevant features of the 5G Device Management API were also tested, meaning to check whether the devices were successfully attached to the 5G network, the device groups were successfully created, and they are isolated from each other etc. When the connections were successfully achieved using the 5G Device Management API, the end-to-end data transmission tests were performed to test whether the synthetic data streams representative of edgeFLEX services were successfully transmitted from 5G UEs to the edgeFLEX databus.

Objective of the test series

The aim of functionality tests and end-to-end data transmission tests with the 5G Device Management API was to use the 5G Device Management API, demonstrate and validate the functionality of its features, and evaluate the potential benefits of the API on the energy sector.

Integrating the components of the edgeFLEX platform into the 5G network

As it was described in D4.3 [3], the edgeFLEX platform is one of the main developments of the project. From the edgeFLEX platform components, the edgeFLEX databus, persistence and the visualization tools were integrated into the 5G laboratory network where the 5G Device Management API proof-of-concept was available.

Figure 4-9 represents the communications between various endpoints or components, which are needed to have successful data transmissions between field devices and services deployed in real power grids or VPPs. The first communication endpoint named as “Data Generators” represents field devices, which can be deployed in and connected to the power grid. In this test series, 5G industry routers were used to transmit the data over-the-air coming from data generators to the edgeFLEX platform components. The edgeFLEX platform components were provided as Docker images from WP4 and deployed on the edge cloud in the laboratory. The edgeFLEX databus, persistence and visualization tools were representatives of the communications and visualization tools that could be used by a power system operator to digitalize their operations.

Generating the data streams for end-to-end data transmission tests

The lower half of the Figure 4-9 illustrates the end-to-end data transmission between 5G UEs and the edgeFLEX platform components hosted on the edge cloud. The blue dotted arrow in Figure 4-9 indicates the unidirectional 5G data transmission over-the-air.

The Raspberry Pi’s were deployed with data generators, and they were configured to simulate measurement data of edgeFLEX services, namely the frequency control and voltage control services, and to send relevant data periodically. During the end-to-end data transmission tests, some of the synthetic data generators were configured to send voltage control related data from devices reserved for DSOs, whereas the others were configured to send frequency control related data from devices reserved for VPP operators. The communication protocol used during the end-to-end data transmission tests was the MQTT protocol. The data generators in the Raspberry Pi’s published MQTT messages that were transmitted to the edgeFLEX databus through the 5G industry routers. The edgeFLEX databus acted as an MQTT broker, receiving messages from data generators, and routing them to the persistence and visualization tools. The persistence tool was one of the subscribers of the data, acting as a data storage and providing the storage functionalities. The Visualization tool was another subscriber of the data, which had a GUI to illustrate the incoming data in graphs to the user.

5G-enabled secure and reliable communication between power grid actors

The 5G Device Management API gives the capability of managing and monitoring the connectivity of field devices to the power system operators and VPP operators. It gives these operators the ability to manage the connectivity of their own devices, add their devices to the 5G network or remove devices from the network, split and group multiple devices into various device groups, set different QoS levels per device or per group according to the requirements of devices, and monitor the QoS of the connections. To test and evaluate the benefits of the 5G Device Management API in the energy sector, various tests were performed considering a 5G-enabled enhanced use case defined in edgeFLEX, which allows secure data exchange between DSOs and VPP operators. As VPP operators and DSOs are two different organizations, one of the challenges for them to collaborate and coordinate to stabilize the grid faster is to maintain security on their operations and data transmissions. The 5G Device Management API and its features were considered in edgeFLEX to minimize these challenges. The API will enable these operators to securely exchange data in order to increase grid stability, to integrate more renewable energy assets into grid balancing and to enable and optimize the energy flexibility trading for VPP operators. In this way, both operators would benefit from the collaboration, and they can flexibly decide and manage when to exchange data between each other.

For the use case, it was assumed that there is a DSO and a VPP operator that own various assets such as measuring devices, batteries, and solar panels etc. As described above, the use case is about enabling secure and reliable data exchange between a DSO and a VPP operator that would result in faster grid stabilisation for the DSO, and better optimization on the flexibility trading for the VPP operator. To achieve that, the VPP operator would like to exchange data with the DSO. As a result, the VPP operator could trade their flexibility at the flexibility market at the time when the DSO needs it. On the other hand, the DSO needs to keep the grid stable, therefore they would also benefit from a bi-directional data exchange with a VPP operator to stabilize the grid. However, this data exchange should be secure and both operators would like to be sure that their own data transmissions on their premises are isolated from the traffic of any other operator. One of the solutions for enabling a secure data exchange between a DSO and VPP operator, giving operators the flexibility to share data whenever they want or need, and performing all these operations with an easy-to-use tool is the “5G Device Management API”. The operators could simply use the API in order to improve the security in data sharing by creating isolated device groups, by connecting or disconnecting some data resources from the groups whenever they need and by using the easy-to-use GUI of the API.

4.4.1 Test case 1: Functionality test of the device provisioning and onboarding feature of the API

In this subsection, the API feature for device provisioning and onboarding is described. A short description of those procedures is given below:

- Device provisioning – It is the procedure of providing the subscription related information of a 5G UE (e.g., subscription identification numbers such as GPSI and IMSI) to the 5G network to be authenticated by the 5G network.
- Device onboarding – It is the procedure of the acceptance of a 5G UE by the 5G network and being attached to the 5G network, giving the 5G UE the communication capability.

4.4.1.1 Test conditions

- The 5G Device Management API infrastructure in the lab is available and functional,
- Multiple 5G UEs (Raspberry Pi's connected to 5G industry routers) are turned on in the lab,
- Each 5G industry router has the correct authentication certificate available that enables successful authentication and onboarding of the device to the 5G network,
- The authorized user has access to the GUI of the 5G Device Management API.

4.4.1.2 Test sequence

- The authorized user provides unique subscription identification numbers (e.g., GPSI and IMSI) of devices to the 5G Device Management API’s GUI to allow devices to be accepted by and attached to the 5G network,
- The 5G Device Management API forwards the unique identifiers to the relevant 5G core network functions in the background to onboard requested devices to the 5G network,
- The successful onboarding of the devices is achieved, and this operation is illustrated in the GUI.

4.4.1.3 Test results

After having the test conditions in place, the test sequence was followed to see whether the device provisioning and onboarding feature of the 5G Device Management API was functioning correctly. It was seen that all four devices (5G UEs) were successfully accepted by the 5G network and onboarded to the network. A screenshot of the GUI showing the provisioned and onboarded devices in the 5G network can be seen in Figure 4-10. As these devices were not added to any device groups yet, they were all onboarded to the “Default” group with default QoS levels.

Figure 4-10 Screenshot of the GUI showing the onboarded devices

4.4.2 Test case 2: Functionality test of the device group and device connectivity management features of the API

4.4.2.1 Test conditions

- The 5G Device Management API infrastructure in the lab is available and functional,
- The authorized user has access to the GUI of the 5G Device Management API,
- Devices (5G UEs) are already onboarded to the 5G network.

4.4.2.2 Test sequence

- The authorized user wants to create two device groups (one group created for the DSO and one group created for the VPP operator) by using the GUI,
- The user defines the name of the groups and the QoS levels for each group before sending the requests to the API using the GUI. As one of the QoS parameters to change, the user sets the priority level of the traffic to “high”,
- Created device groups are shown in the GUI as a result of a successful device group creation event,
- Using the “device group management” feature of the API, two DSO devices are added to the DSO group, whereas two VPP devices are added to the VPP group to isolate their traffic between each other,
- Lastly, the user enables the in-group communication through the “device connectivity management” feature of the API to establish all necessary end-to-end connections between devices (5G UEs) and edgeFLEX platform (DSO and VPP platform instances) hosted on the edge cloud

4.4.2.3 Test results

After having the test conditions in place, the test sequence was followed to see whether the device group management and the device connectivity management features of the 5G Device Management API was functioning correctly. It was seen that two device groups named as “DSOgroup” and “VPPgroup” were successfully created in the 5G network with requested priority levels as seen in Figure 4-11.

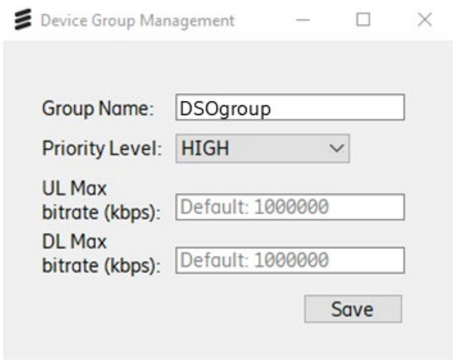

Figure 4-11 Screenshot of the GUI showing the creation of a DSO device group with needed QoS levels

A screenshot of the GUI showing the created device groups in the 5G network, and the members of groups added to these groups can be seen in Figure 4-12. By having two groups, the data transmissions of DSO devices and VPP devices were isolated from each other as well as other data transmissions that would happen in the 5G network. In addition, the groups were configured to route the synthetic data streams of particular devices to the particular edgeFLEX platforms: “DSOgroup” traffic routed to the DSO Platform, “VPPgroup” traffic routed to the VPP Platform. Therefore, only the members of DSOgroup could transmit their data to the DSO Platform, whereas only the members of VPPgroup could send their data to the VPP Platform.

Figure 4-12 Screenshot of the GUI showing the created groups and members of these groups

4.4.3 Test case 3: Functionality test of the device connectivity monitoring feature of the API

4.4.3.1 Test conditions

- The 5G Device Management API infrastructure in the lab is available and functional,
- The authorized user has access to the GUI of the 5G Device Management API,
- Devices (5G UEs) are already onboarded to the 5G network,
- Separate device groups for the DSO and the VPP operator are created in the network,

- The QoS parameters for each device group are set through the GUI,
- End-to-end connections between the devices and edge cloud are established using the 5G Device Management API.

4.4.3.2 Test sequence

- The authorized user wants to monitor the connectivity of the devices onboarded to the 5G network by using the GUI,
- The user defines the time period to monitor the connectivity of their devices,
- The user selects which connectivity related parameters to be monitored and accordingly, the round-trip-time latency and packet loss measurements are performed for the selected devices,
- The successful monitoring of the device connectivity is achieved, and the latency and packet loss values are illustrated to the user in the GUI.

4.4.3.3 Test results

After having the test conditions in place, the test sequence was followed to see whether the device connectivity monitoring feature of the 5G Device Management API was functioning correctly. It was seen that the packet loss and round-trip-time latency of the communication links of devices could be monitored, and the values were illustrated in the GUI. As an example, the packet loss and latency values of the communication link of a DSO device can be seen in Figure 4-13.

As a benefit of this feature: if users from the DSO know that the current packet loss is zero, then they could trust the data streams coming from field devices and those data could be processed on the energy service hosted on their edge cloud. On the other hand, in case the packet loss value is high, then they could know that there is a communication issue in their network and the incoming data could be erroneous. Therefore, they could ignore the data received in the last seconds or minutes, and the estimation of the energy service would still be functional.

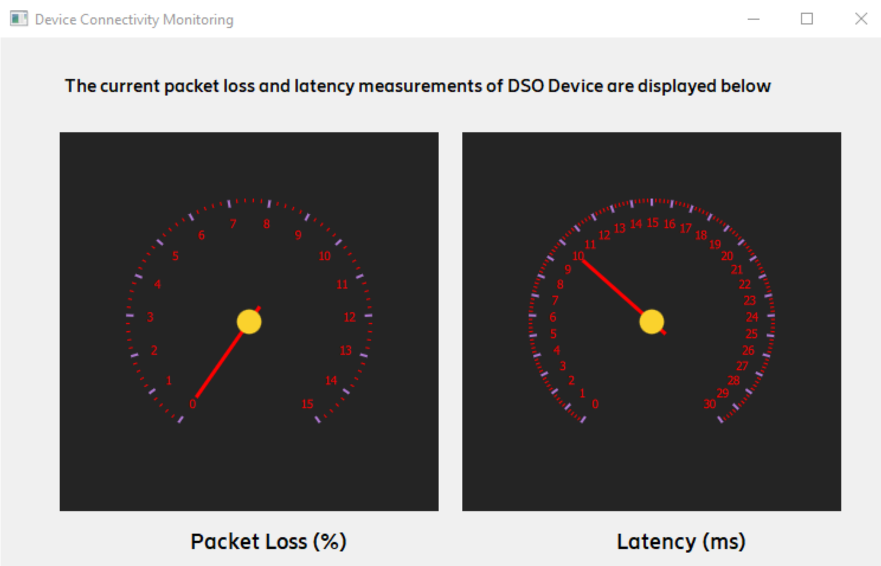

Figure 4-13 Screenshot of the GUI showing the recent packet loss and latency values of the communication link of the DSO Device

4.4.4 Test case 4: End-to-end data transmission tests with an edgeFLEX service

4.4.4.1 Test conditions

- The 5G Device Management API infrastructure in the lab is available and functional,

- The authorized user has access to the GUI of the 5G Device Management API,
- Devices (5G UEs) are already onboarded to the 5G network,
- Separate device groups for the DSO and the VPP operator are created in the network,
- The QoS parameters for each device group are set through the GUI,
- Two instances of the edgeFLEX platform components (i.e., DSO Platform and VPP Platform, both hosting the edgeFLEX databus, persistence and visualization tools) were deployed on edge cloud as data receivers and they were instantiated separately,
- End-to-end connections between the 5G devices (i.e., MQTT data generator deployed on Raspberry Pis) and edgeFLEX platform components (i.e., DSO Platform and VPP Platform, two different instances of the edgeFLEX databus, persistence and visualization tools hosted on edge cloud) are established using the 5G Device Management API.

4.4.4.2 Test sequence

- The authorized user wants to send data streams from devices (5G UEs) onboarded to the 5G network to the databuses of the DSO and the VPP operator,
- The authorised user checks the API’s GUI and the IP addresses of relevant devices in order to start sending right data streams from DSO devices to the DSO databus and vice versa,
- As sending the actual data streams is not a feature that can be initiated from the GUI, the user needs to connect to the Raspberry Pi terminal of the DSO Device,
- To demonstrate that the data streams are received on the edge cloud side, the user needs to connect to the DSO Databus terminal,
- The successful reception of the data streams coming from the DSO Device is shown in the DSO Databus terminal,
- To demonstrate that the other two edgeFLEX platform components (i.e., persistence and visualization tools) are also functional, the user also checks the received data streams in the GUI of visualization tool (Grafana),
- To demonstrate that the data streams of DSO Device are not received by the VPP Databus, the user also connects to the VPP Databus terminal and checks whether the data from DSO Device are captured or not,
- It is seen in the VPP Databus terminal that data streams coming from the DSO Device are not received in the VPP Databus.

4.4.4.3 Test results

After having the test conditions in place, the test sequence was followed to see whether the end-to-end data transmission between 5G UEs and the edge cloud hosting edgeFLEX platform components could be performed. It was seen that the data generators deployed in Raspberry Pi’s were functioning correctly and their data was received by the edgeFLEX databus instances. Firstly, the recent status of the GUI can be seen in Figure 4-14, which illustrates the successful functionalities of device provisioning and onboarding, device group management and device connectivity management features of the API.

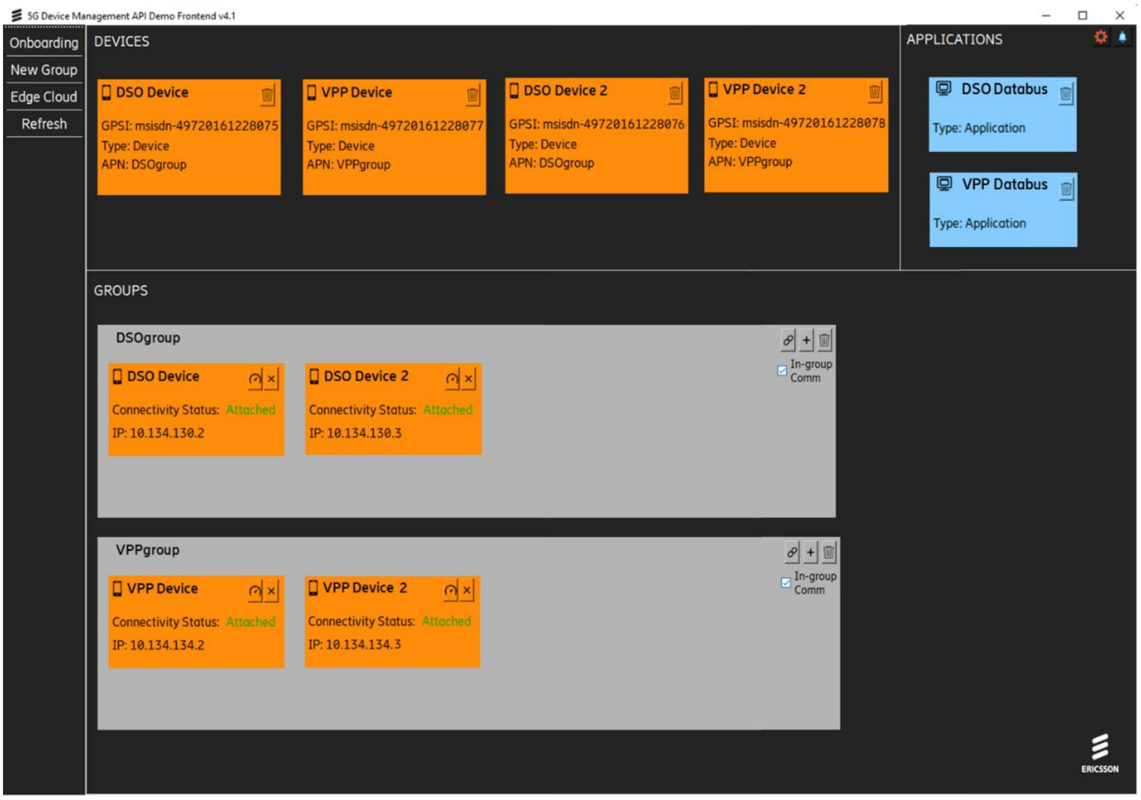

Figure 4-14 Screenshot of the GUI showing the onboarded devices, created groups and members of the groups

In order to test the end-to-end connections established through the use of the 5G Device Management API, synthetic data streams were generated by the Raspberry Pi's as explained in Section 4.4 and the received data was observed from the edgeFLEX databus side, through the terminal window and the Grafana GUI from the edgeFLEX platform.

Figure 4-15 shows screenshots taken from terminal windows of the databuses "DSO Databus" and "VPP Databus" connected to two separate groups. The upper two terminal windows belong to a virtual machine which hosts the DSO Databus. It illustrates that the DSO Databus could receive the measurement data simulated and transmitted from the DSO Device with the IP address "10.134.130.2" (pink box), whereas it could not receive data from the VPP Device with the IP address "10.134.134.2" (orange box), as the DSO Device and the VPP Device are members of two different groups and their traffic are isolated from each other. Similarly, the lower picture indicates a similar result, in which the VPP Databus could only receive the measurement data transmitted from the VPP Device, but not from the DSO Device. With the screenshot seen in Figure 4-15, we both proved that the end-to-end transmissions between 5G UEs and the edgeFLEX Databus instances connected to them were successful, and the data transmissions of devices located in two different DSO and VPP groups are isolated from each other.

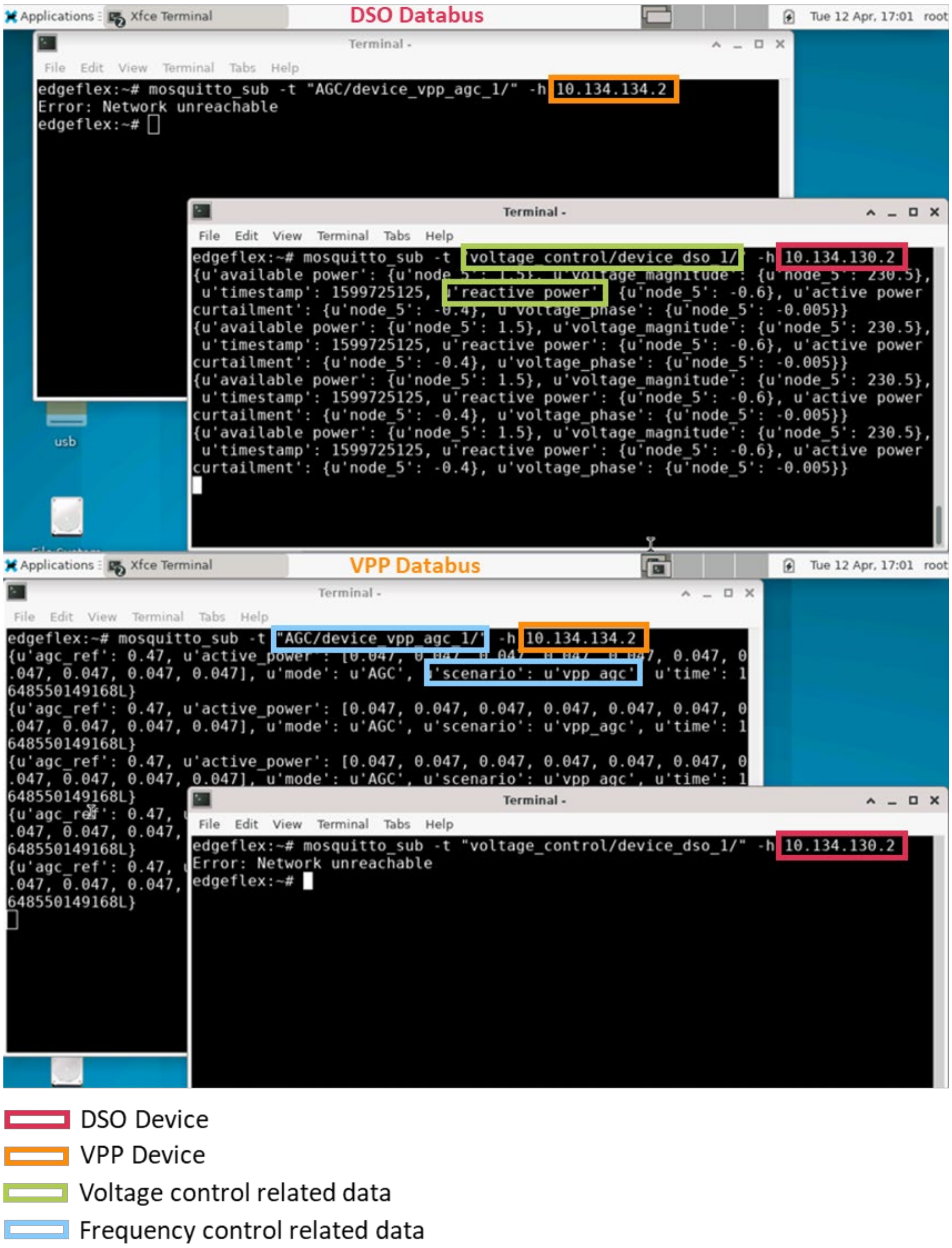

Figure 4-15 Screenshot of the virtual machines hosting the DSO Databus and the VPP Databus showing the successful end-to-end data transmissions and isolation between groups

The end-to-end data transmissions and the correct functionality of the edgeFLEX visualization tool were also tested during the tests. The Grafana GUI was run in a browser on the virtual machine where the edgeFLEX databus, persistence and visualization tools were hosted. The received data was observed and plotted on the dashboards in Grafana as shown in Figure 4-16. As an example, the dashboards in Figure 4-16 illustrate the data transmitted from the VPP Device to the VPP Databus. The data streams were simulated data of the frequency control service in edgeFLEX, which included the active power and reference power values together with timestamps. The figure proves that the measurement data was successfully transmitted to the edgeFLEX platform, received by the edgeFLEX databus, stored in the persistence tool and forwarded to the visualization tool Grafana.

Figure 4-16 Screenshot of the edgeFLEX visualization tool showing the successfully received data from a 5G UE

The results show that the 5G Device Management API has succeeded in onboarding devices, creating and managing device groups, enabling isolation between separate groups and establishing end-to-end connections. Furthermore, it is seen that the edgeFLEX platform components could be integrated smoothly into the 5G network where the 5G Device Management API was available.

4.4.5 Test case 5: Enabling data exchange between a DSO and a VPP operator as an enhanced edgeFLEX use case

4.4.5.1 Test conditions

- The 5G Device Management API infrastructure in the lab is available and functional,
- The authorized user has access to the GUI of the 5G Device Management API,
- Devices (5G UEs) are already onboarded to the 5G network,
- Separate device groups for the DSO and the VPP operator are created in the network,
- The QoS parameters for each device group are set through the GUI,
- Two instances of the edgeFLEX platform components (i.e., DSO Platform and VPP Platform, both hosting the edgeFLEX databus, persistence and visualization tools) were deployed on edge cloud as data receivers and they were instantiated separately,
- End-to-end connections between the 5G devices (i.e., MQTT data generator deployed on Raspberry Pi's) and edgeFLEX platform components (i.e., DSO Platform and VPP Platform, two different instances of the edgeFLEX databus, persistence and visualization tools hosted on edge cloud) are established using the 5G Device Management API.

4.4.5.2 Test sequence

- The authorized user wants to enable data exchange between a DSO and a VPP operator,
- Using the "device group management" feature of the API, the group membership of the VPP Device is updated and the device is moved from VPP group to the DSO group,
- Updated device groups are shown in the GUI as a result of a successful device group creation event,
- Thus, the VPP Device can start sending its measurement data to the DSO Databus,
- To demonstrate that the data streams from VPP device are received on the DSO Databus side, the user needs to connect to the DSO Databus terminal,
- The successful reception of the data streams coming from the VPP Device is shown in the DSO Databus terminal.

4.4.5.3 Test results

After having the test conditions available in place, the test sequence was followed to see whether the enhanced use case of enabling data exchange between a DSO and a VPP operator as described in Section 4.4 could be realised.

As an example, the VPP Device was moved from the VPP group to the DSO group. It was seen that the device group management feature of the API could be used to perform this operation and change the membership of a device from a group. The status of the GUI after the VPP Device moved from VPP group to the DSO group can be seen in Figure 4-17.

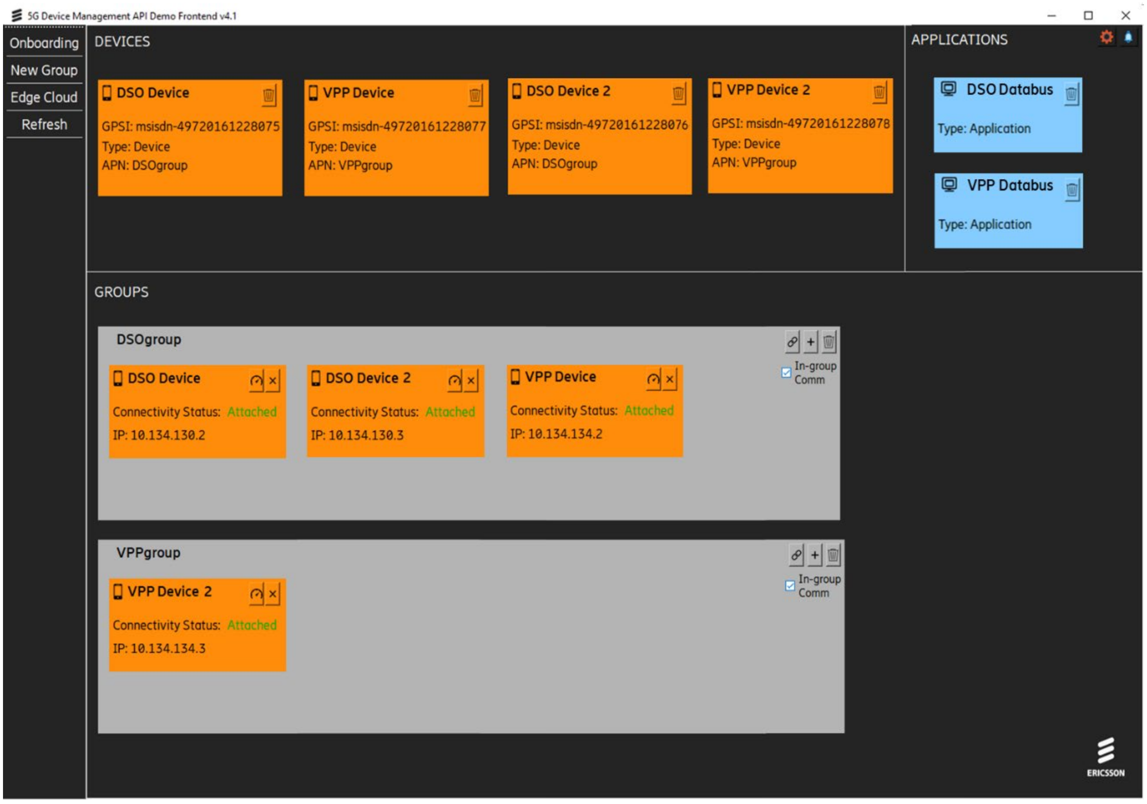

Figure 4-17 Screenshot of the GUI after performing the enhanced edgeFLEX use case of enabling data exchange between a DSO and a VPP operator

In order to test and validate that the data streams generated in the Raspberry Pi of the VPP Device could be transmitted over-the-air to the DSO Databus, the received data was observed from the DSO Databus side through the terminal window. The result can be seen in Figure 4-18.

As it was shown previously in Figure 4-15, the DSO Device was transmitting the voltage control related data to the DSO Databus, whereas the VPP Device was transmitting frequency control related data. As in Figure 4-18, the DSO Databus can receive both the voltage control and frequency control related data, because the VPP Device is now a member of the DSO Group.

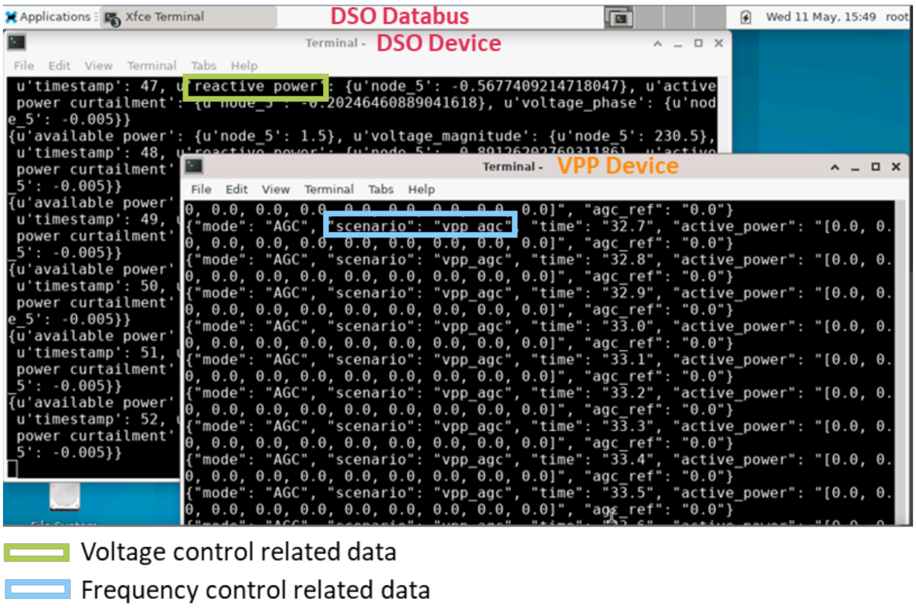

Figure 4-18 Screenshot of the virtual machine hosting the DSO Databus showing the successful data transmissions initiated from the DSO Device and the VPP Device

4.5 Conclusions regarding the results of the test series

4.5.1 Achieved improvements

In all 5G latency performance tests, it was shown that 5G fulfils the edgeFLEX service requirements in terms of latency. The measures of the average one-way latency and of round-trip time were used. In the 5G performance tests of edgeFLEX services using synthetic data, the average one-way latency was between 4 ms and 11 ms. The positive impact of the optimal message rate reduced the latency up to 50%. Under bad radio conditions, nearly no differences in latency changes were observed. A connection loss occurred for 83 dB corresponding to 937 m distance between sender, i.e., the sensor in the field, and receiver, i.e., the nearby 4G or 5G base station. In the tests with edgePMU data streams, the average one-way latency ranged between 4 ms and 15 ms. By using an optimal sampling rate and optimal vectorize, the latency resulted in a reduction of 58%. In the tests with frequency control services, the 5G SA infrastructure average radio link latency was around 10 ms, while the URLLC infrastructure achieved radio link latencies of under 2 ms. The best performance of RTTs was observed with the URLLC infrastructure and when Nagle's algorithm was disabled resulting in RTTs below 3 ms with an improvement of up to 80%. The highest reduction of RTT was achieved in the 99.9th percentile and with an improvement of up to 92%.

In all tests with the 5G Device Management API, it was shown that the 5G Device Management API features and the GUI developed for the API were functional, and the prototype network supported to deploy and run the developments from edgeFLEX project, such as edgeFLEX databus, persistence and visualization tools. With the monitoring of the synthetic data streams in the terminal, it was demonstrated that the operations such as attaching devices to the 5G network, creating isolated device groups, setting the QoS levels of the communications within these groups and monitoring the quality of device connections were possible to perform and simpler to do with the GUI.

4.5.2 Performance of 5G in relation to the requirements for current and future edgeFLEX services and enablers

In all tests with synthetic data, the 5G network was able to fulfil the latency performance requirements of edgeFLEX services and enablers. Moreover, 5G can fulfil even more stringent requirements that will be required when the edgeFLEX services and enablers evolve in the future.

By using near real-time digital management of the power grid, a faster visualization of the results will be possible. Renewable energy generation sources will have a greater use resulting in a reduction of CO2 emissions of energy generation. Moreover, services and enablers with dynamic mechanisms will be enabled which will require fast interactions between the components of the grid dependent on low latencies.

For the tests of edgePMU data streams, the 5G link worked properly when sending the messages in a high frequency resulting in a high throughput. edgePMUs will generate a high volume of data per se. In specific cases such as incidents, the volume of data to be communicated will be greatly increased due to the need for more precision in the measurements. This implies that the future power grid will place higher requirements on the throughput of communications services. The throughput of the 5G network used in the laboratory tests was so high that we did not see packet loss under the loads we could generate with the available equipment. The maximum throughput of 5.76 Mbps measured in the test series did not challenge the 5G network as the maximum available bandwidth for uplink was 100 Mbps.

In the test series of frequency control data streams, it was shown that URLLC outperforms the standard 5G network. Since COORD and ROCOP are time critical services, a low RTT is important to ensure the performance of these services. Our tests showed that standard 5G networks can already support those services. For future scenarios, using the URLLC feature of 5G can be recommended for ultra-low latency and high reliability of the communications network.

In the test series of the 5G Device Management API, the enhanced use case of enabling data exchange between DSOs and VPP operators was tested in the laboratory network. The API offers simplicity and flexibility to any power system operator or VPP operator through the grouping functionality. The grouping functionality can isolate the traffic between different devices groups and provides system operators with the ability to define which devices are added to each group. For instance, DSOs and VPP operators will be able to use the 5G Device Management API to isolate their data traffic, to improve the security of data exchange, and to enable flexibility trading, increasing the stability of the power grids and enabling the integration of more renewable energy sources into grid balancing.

5. Conclusions and value of the results in 5G laboratory tests

In this chapter we describe the value of the test results reported in this deliverable to stakeholders and to other WPs of edgeFLEX and we summarize the key conclusions of this work.

5.1 The value of our 5G performance tests results to a range of energy stakeholder groups

We summarise the value of the 5G latency performance test results of test series 1-3 to a range of energy stakeholder groups in Table 5-1 below. Test series 1-3 described the 5G latency performance under a range of conditions and with a range of representative edgeFLEX data streams.

Table 5-1 The value of the 5G latency performance tests results to energy stakeholder groups

Stakeholder groups addressed by our results	Value of our results to the group
5G system and device manufacturers	Our results demonstrate the potential to optimize and reduce 5G latency in the network enabling the 5G system and device manufacturers to support a wider range of utility and other vertical sector critical infrastructure use cases in both private and public 5G network infrastructures.
Public 5G network providers	Our results demonstrate the potential to optimize and reduce 5G latency in the radio network enabling public 5G network providers to support a wider range of utility and other vertical sector critical infrastructure use cases using public 5G network infrastructures.
Power network operators	Our results demonstrate that 5G has capabilities which can go far beyond the requirements of today's use cases for Distribution Network Automation and many other use cases in the energy sector. 5G can support the requirements of VPPs as their use cases evolve towards enabling real-time optimization and management of their energy use cases and their use of time critical use cases becomes common. Power network operators can choose to use private 5G networks purchased by the power operators or public 5G infrastructure provided by public operators.
VPP operators and energy asset owners	Our results demonstrate that existing and future latency requirements of power network use cases can be met using 5G networks and that optimizations of 5G radio networks offer improvements in latency to use cases with demanding latency requirements. VPP operators and energy asset owners can choose to use private 5G networks purchased by the VPP operators or public 5G infrastructure provided by public operators.
Academic institutions	Our results bring together the fields of communications and power network automation demonstrating how 5G networks can offer flexible, low latency communications support for advanced VPP and Energy Community use cases, such as those developed by the edgeFLEX project partners.

5.2 The value of our 5G Device Management API functionality test results to a range of energy stakeholder groups

We summarise the value of the 5G Device Management API functionality test results of test series 4 to a range of energy stakeholder groups in Table 5-2 below. Test series 4 describes functionality tests of the features of the 5G Device management API.

Table 5-2 The value of the 5G Device Management API test results to energy stakeholder groups

Stakeholder groups addressed by our results	Value of our results to the group
5G system and device manufacturers	Our results demonstrate that the 5G Device Management API functionality increase the ease of use for power operators, improve the security and reliability features in relation to power system use cases and form the basis for further innovative use cases. Once standardisation has been completed, this opens new global markets for 5G system and device manufacturers.
Public 5G network providers	Once standardisation has been completed, commercial implementations of the 5G Device Management API functionality will enable public network providers to support increased ease of use for power operators, improved security and reliability features in relation to power system use cases and to offer functionality forming the basis for further innovative use cases.
Power network operators	Power network operators can avail of the increased ease of use, improved security and reliability features in relation to power system use cases and new functionality forming the basis for further innovative use cases. Power operators can choose to use this new functionality in privately owned 5G networks or in publicly provided 5G networks or in hybrid networks combining public and private infrastructures.
VPP operators and energy asset owners	VPP operators and energy asset owners can avail of the increased ease of use, improved security and reliability features in relation to power system use cases and new functionality forming the basis for further innovative use cases. VPP operators and energy asset owners can choose to use this new functionality in privately owned 5G networks or in publicly provided 5G networks or in hybrid networks combining public and private infrastructures.
Academic institutions	Once standardisation has been completed, academic institutions can build on the functionality adopted in the 3GPP global standards to develop innovations such as adapters or Network Applications (NetApps) [11]. An approach to developing adapters for IoT use cases is followed in the IoT-NGIN project [12]. An approach to the development of Network Applications is being followed in a range of Horizon 2020 projects [11].

5.3 The value of our test results to other edgeFLEX WPs

We summarise the value of the 5G performance and 5G API functionality test results of test series 1-4 to the edgeFLEX WPs in Table 5-3 below.

Table 5-3 The value of the 5G performance and 5G API functionality test results to the edgeFLEX WPs

WP	Value of our results to the WP
WP1 and WP2	We have provided feedback to all partners developing edgeFLEX services and enablers in terms of communication latency performance that a standard 5G SA network can provide.
WP3	Results and feedback have been provided to RWTH as input for the further development of the edgePMU hardware.
WP4	Results and feedback have been provided to WIT for the further development of the edgeFLEX platform.
WP5	We have collected experience through deployments of the edgeFLEX services and enablers in the laboratory test infrastructure which will be useful for the deployments in the project field trials and in commercial power networks in the future. Furthermore, we have provided the expected latency performance of the services and enablers in the 5G network when the URLLC feature would be deployed.
WP6	We have provided feedback and input to business case development and regulatory aspects based on the 5G performance and API functionality test results and the value they offer to stakeholder groups.

5.4 Main conclusions of the test series of 5G performance and API functionality tests

The result of the 5G performance tests described in Section 4 demonstrate that 5G fulfils the requirements and supports the expected evolution of the edgeFLEX services and enablers concerning low message delay, i.e., latency and round-trip time.

Moreover, the results show that the edgeFLEX services and enablers' performance can be further improved by optimizing different traffic patterns influencing the 5G performance by optimal setting of specific parameters:

- In the tests with synthetic data streams of edgeFLEX services, we optimised the parameter of message rate (number of messages per second) to reduce latency to a minimum
- In the tests with edgePMU data streams, we optimised the edgePMU parameters of vectorize (number of samples per message) and sampling rate (number of edgePMU

samples per second) to reduce latency to a minimum. It was beneficial to keep the vectorize and sampling rate parameters as low as possible.

- In the tests of frequency control services, we optimised latency by replacing the 5G SA network with a prototype URLLC network. It was successfully shown that the URLLC prototype network has lower latency than the standard 5G SA network providing the opportunity to VPPs and power system operators to bring the management of the power grid to a close to real-time operational status.

When optimising the parameters of the 5G network and the edgeFLEX services and enablers as described above, latencies between 4 - 11 ms are achieved in tests with synthetic data streams even under bad channel conditions. In tests with edgePMU data streams, 5G average latencies between 3 - 12 ms are observed. In the frequency control services using URLLC, RTTs between 3 - 4 ms are recorded. The tests demonstrated that the 5G can provide communications for the edgeFLEX services fulfilling the ICT requirements and ensuring reliability.

These results demonstrate the potential to enable further optimisations of the power grid in the context of increasing power production using volatile renewable energy sources, the need to reduce CO2 emissions related to power production and enable the further optimisations in the integration of VPPs and energy asset owners into the management of the power grid

Our results demonstrate that the 5G Device Management API functionality increase the ease of use for power operators and VPP operators, improve the security and reliability features in relation to power system use cases and form the basis for further innovative use cases. Once standardisation has been completed, this opens new global markets for 5G system and device manufacturers, as well as for 5G network operators. Power system operators will be able to implement innovative services themselves enabled by the increased ease of use which the 5G Device Management API provides.

6. List of Tables

7. List of Figures

8. References

[1] edgeFLEX, “D4.1 - Description of edgeFLEX platform design,” 31 March 2021. [Online]. Available: https://edgeflex-h2020.eu/files/content-edgeflex/Content_Pages/Progress/Deliverables/edgeFlex_883710_D4.1.pdf. [Accessed 28 March 2022].

[2] edgeFLEX, “D4.2 - Description of edgeFLEX MVP,” 30 September 2021. [Online]. Available: https://edgeflex-h2020.eu/files/content-edgeflex/Content_Pages/Progress/Deliverables/edgeFLEX_883710_D4.2.pdf. [Accessed 28 March 2022].

[3] edgeFLEX, “D4.3 - Description of internal interfacesfor conrol services,” 30 March 2022. [Online]. Available: https://edgeflex-h2020.eu/files/content-edgeflex/Content_Pages/Progress/Deliverables/883710_Deliverable_D4.3.pdf. [Accessed 21 June 2022].

[4] “3GPP TR 21.915 ; 3rd Generation Partnership Project; Technical Specification Group Services and System Aspects; Release 15 Description; Summary of Rel-15 Work Items (Release 15),” 2019.

[5] “3GPP TR 21.916 ; 3rd Generation Partnership Project; Technical Specification Group Services and System Aspects; Release 16 Description; Summary of Rel-16 Work Items,” 2019.

[6] “tcpdump,” [Online]. Available: https://www.tcpdump.org/.

[7] “Network Time Protocol,” [Online]. Available: http://www.ntp.org/.

[8] “MQTT,” [Online]. Available: https://mqtt.org/. [Accessed 6 October 2021].

[9] “mosquitto.conf man page,” [Online]. Available: http://mosquitto.org/man/mosquitto-conf-5.html. [Accessed 18 February 2022].

[10] “VILLASnode,” [Online]. Available: https://villas.fein-aachen.org/doc/node.html.

[11] E. C. CORDIS, “5G PPP – 5G innovations for verticals with third party services,” [Online]. Available: https://cordis.europa.eu/programme/id/H2020_ICT-41-2020. [Accessed 21 July 2022].

[12] IoT-NGIN. [Online]. Available: https://iot-ngin.eu/. [Accessed 21 July 2022].

9. List of Abbreviations

ACK	Acknowledgement
AGC	VPP Automatic Generation Control
API	Application Programming Interface
COORD	VPP Coordinated Frequency Control
CPU	Central Processing Unit
DL	Downlink
DER	Distributed Energy Resource
eMBB	enhanced Mobile Broadband
DSO	Distribution System Operator
FFR	Fast Frequency Regulation
gNB	G Node B (5G base station)
GPSI	Generic Public Subscription Identifier
GUI	Graphical User Interface
ICT	Information and Communications Technology
IMSI	International Mobile Subscriber Identity
IP	Internet Protocol
LOS	Line of sight
MQTT	Message Queueing Telemetry Transport
MTU	Maximal Transmission Unit
NTP	Network Time Protocol
PDU	Protocol Data Unit
PMU	Phasor Measurement Unit
ROCOP	Rate-of-Change-of-Power from Frequency Regulation Metering
RTT	Round Trip Time
SA	Standalone
TCP	Transmission Control Protocol
TSO	Transmission System Operator
UDP	User Datagram Protocol
UE	User Equipment
UL	Uplink
URLLC	Ultra-reliable Low Latency Communication
VM	Virtual Machine
VPP	Virtual power plant
WP	Work package

ANNEX

Annex 1 provides formulas which take as input radio propagation models and provide as output the distance between sender and receiver. These formulas provide a rough approximation of how variations in attenuation of radio signals relate to the distance between the sender and receiver of data.

A.1 Attenuation Test Equations

To compute the distance, it was assumed that the attenuation level is equivalent to the path loss. Then the equation for line of sight (LOS) attenuation can be used to calculate the basic distance d_0 corresponding to a basic attenuation level of 15dB.

$$PL_{dB} = 20\log_{10}\frac{4\pi d}{\lambda} = 20\log_{10}\frac{4\pi d f}{c}$$

Then, the equation for log-normal shadowing can be used to compute the other distances based on the basic attenuation and distance.

$$PL(d) = PL(d_0) + 10n\log_{10}\frac{d}{d_0}$$

Manipulating and combining the equations leads to the following end equation:

$$d = d_0 \cdot 10^{\frac{1}{10n}(PL(d)-PL(d_0))} = \frac{c}{4\pi f} \cdot 10^{\frac{1}{20}PL(d)}$$