FACULTY OF INFORMATION TECHNOLOGY AND ELECTRICAL ENGINEERING
DEGREE PROGRAMME IN WIRELESS COMMUNICATIONS ENGINEERING

MASTER'S THESIS

INTEGRATION AND CHARACTERISATION OF THE PERFORMANCE OF FIFTH-GENERATION MOBILE TECHNOLOGY (5G) CONNECTIVITY OVER THE UNIVERSITY OF OULU 5G TEST NETWORK (5GTN) FOR COGNITIVE EDGE NODE BASED ON FRACTAL EDGE PLATFORM

Author	Diluna Adeesha Warnakulasuriya
Supervisor	Dr. Konstantin Mikhaylov
Second Examiner	Dr. Tuomo Hänninen

April 2023

Warnakulasuriya D. A. (2023) Integration and characterisation of the performance of fifth-generation mobile technology (5G) connectivity over University of Oulu 5G Test Network (5GTN) for cognitive edge node based on FRACTAL edge platform. The University of Oulu, Faculty of Information Technology and Electrical Engineering, Degree Programme in Wireless Communications Engineering. Master’s Thesis, 88 p.

ABSTRACT

In recent years, there has been a growing interest in cognitive edge nodes, which are intelligent devices that can collect and process data at the edge of the network. These nodes are becoming increasingly important for various applications such as smart cities, industrial automation, and healthcare. However, implementing cognitive edge nodes requires a reliable and efficient communication network. Therefore, this thesis assesses the performance of direct cellular (5G) and IEEE 802.11-based Wireless Local Area Network (WLAN) technology for three network architectures, which has the potential to offer low-latency, high-throughput and energy-efficient communication, for cognitive edge nodes.

The study focused on evaluating the network performance metrics of throughput, latency, and power consumption for three different FRACTAL-based network architectures. These architectures include IEEE 802.11-based last mile, direct cellular (5G) backbone, and IEEE 802.11-based last mile over cellular (5G) backbone topologies. This research aims to provide insights into the performance of 5G technology for cognitive edge nodes.

The findings suggest that the power consumption of IEEE 802.11-enabled nodes was only slightly higher than the reference case, indicating that it is more energy-efficient than 5G-enabled nodes. Additionally, in terms of latency, IEEE 802.11 technology may be more favourable. The throughput tests revealed that the cellular (5G) connection exhibited high throughput for communication between a test node and an upper-tier node situated either on the internet or at the network edge. In addition, it was found that the FRACTAL edge platform is flexible and scalable, and it supports different wireless technologies, making it a suitable platform for implementing cognitive edge nodes.

Overall, this study provides insights into the potential of 5G technology and the FRACTAL edge platform for implementing cognitive edge nodes. The results of this research can be valuable for researchers and practitioners working in the field of wireless communication and edge computing, as it sheds light on the feasibility and performance of these technologies for implementing cognitive edge nodes in various applications.

Key words: FRACTAL, edge computing, cognitive edge nodes, latency, throughput.

TABLE OF CONTENTS

FOREWORD

This thesis focuses on examining the performance of 5G connectivity over the University of Oulu 5G Test Network (5GTN) for a cognitive edge node based on the FRACTAL edge platform as part of the ECSEL JU FRACTAL project (grant 877056). The research was conducted at the Centre for Wireless Communications (CWC) at the University of Oulu.

I would like to express my sincere gratitude to my supervisor assistant professor Dr. Konstantin Mikhaylov for the guidance, support, inspiration, and encouragement given throughout the period of my Master's studies. I also would like to thank Dr. Tuomo Hänninen at CWC for being the second examiner for this thesis. I am also grateful to the head of the CWC-Network and System (NS) research unit Professor Jari Iinatti for providing me with the opportunity to join and contribute to the ECSEL JU FRACTAL project. Also, I would like to thank professors and teachers, and other colleagues at CWC for their support and guidance throughout my studies at the university of Oulu. Especially, I would like to mention Dr. Onel Luis Alcaraz López and doctoral researcher Kenichi Komatsu at CWC. Special thanks to Dr. Kari Kärkkäinen for the support and guidance provided to me as the academic coordinator for the wireless communication Master's degree program. At last, I want to thank my parents, friends, and fellow students for their prayers and support.

I dedicate this work to my parents, who have provided me with unwavering support and encouragement throughout my Master's degree journey.

Oulu, April, 11 2023

Diluna Adeesha Warnakulasuriya

LIST OF ABBREVIATIONS AND SYMBOLS

1G	First Generation
4G	Fourth Generation
5G	Fifth Generation
6G	Sixth Generation
5GTN	Fifth Generation Test Network
AC	Alternative Current
AI	Artificial Intelligence
AP	Access Point
APN	Access Point Name
AT	Attention/Hayes
BLE	Bluetooth Low Energy
C-RAN	Cloud-Random Access Network
CM	Connect Manager
CR	Cognitive Radio
CSMA/CA	Carrier Sense Multiple Access With Collision Avoidance
DC	Direct Current
DHCP	Dynamic Host Configuration Protocol
DUT	Device Under Test
EC	Edge Computing
EPC	Evolved Packet Core
EVB	Evaluation Board
FDMA	Frequency Division Multiple Access
FNEC	Fractal Node Edge Computing
FRACTAL	Flexible Radio Access for Cognitive and Tactical Airborne Links
GUI	Graphical User Interface
HDMI	High-Definition Multimedia Interface
HW	Hardware
ID	Identifier
IoT	Internet of Things
IP	Internet Protocol
ISM	Industrial, Scientific, and Medical
ISP	Internet service provider
KPIs	Network Key Performance Indicators
LAN	Local Area Network
LoRa	Long Range
LoS	Line-of-Sight
LoRaWAN	Long Range Wide Area Network (LoRaWAN)
MAC	Media Access Control
MEC	Mobile Edge Computing
MIMO	Multiple-Input Multiple-Output
ML	Machine Learning
MMW	Mili-Meter Wave
NB-IoT	Narrow Band IoT
NR	New Radio
NSA	Non-Standalone
O-RAN	Open-Random Access Network

OAI	Open Air Interface
OFDMA	Orthogonal Frequency Division Multiple Access
ONNs	Oscillatory Neural Networks
OS	Operating System
PTP	Precision Time Protocol
PULP	Parallel Ultra-Low Power
QoE	Quality of Experience
QoS	Quality of Service
RAN	Radio Access Network
RATs	Radio Access Technologies
RISC-V	Reduced Instruction Set Computer-Five
RSRP	Reference Signal Received Power
RSRQ	Reference Signal Received Quality
RSSI	Received Signal Strength Indicator
RTT	Round-Trip-Time
RX	Receive Antenna
SDN	Software-Defined Networking
SFTP	Secure File Transfer Protocol
SIM	Subscriber Identity Module
SINR	Signal-to-Interference-Noise Ratio
SoC	System-on-Chip
SW	Software
TCP	Transmission Control Protocol
TDMA	Time Division Multiple Access
TS	Test Server
UE	User Equipment
UI	User Interface
USB	Universal Serial Bus
WiFi	Wireless Fidelity
WLAN	Wireless Local Area Network

µs	Microseconds
A	Ampere
bps	Bits per Second
dBm	Decibels Relative to One Milliwatt
Gbps	Giga Bits per Second
Hz	*Hertz*
MHz	*Mega Hertz*
ms	Milliseconds
V	Volts
W	Watts

1 INTRODUCTION

An enormous number of hardware devices and sensors have been produced and used globally as a result of the development of the Internet of Things (IoT). These hardware components are capable of sensing the physical environment around them and processing that information into data. Data consumers can access cloud data following their specific demands after these large data have been transported to the cloud for processing or storage. However, there are several problems in cloud computing. Such as low throughput, high latency, bandwidth bottlenecks, data privacy, centralized vulnerabilities, and additional costs [1]. This is where Edge Computing (EC) steps in.

This chapter summarizes the essential concepts and definitions that are critical to the thesis, specifically in sub-section 1.1 provides a brief introduction to the EC while sub-section 1.2 presents the concept of the Flexible Radio Access for Cognitive and Tactical Airborne Links (FRACTAL) and its strategic goals. An overview of network Key Performance Indicators (KPIs) has been given in sub-section 1.3. The main contributions of this thesis are presented in sub-section 1.4 while sub-section 1.5 present the structure of the rest of the thesis.

1.1 Edge computing

The fundamental concept behind the EC model is to move the data processing, storage, and computing tasks that the cloud originally demanded to the edge of the network close to terminal devices. This decreases the amount of time data must travel between devices and networks, speeds up device responses, lowers the cost of data transmission, and promotes decentralization [1]. EC represents a cloud computing system. It differs from the typical system in that it processes some of the data at the network's edge. It distributes the computational load to routers, switches, base stations, and other gateways. These edge nodes are positioned nearer to the data source than the cloud. Also, they have processing capability and can gather or filter the data, relieving the cloud from this task [2]. Edges ease the workload of the cloud by sending only aggregated data there. As a result, communication bandwidth and data transmission speed are improved. Only complex analyses are performed by cloud services on pre-processed data. A feedback loop manages the information transfer between cloud and edge nodes. Because of this, the system can act and respond in real-time inside a dynamic environment. Recent systems-on-a-chip (SoC) advancements make computer chips powerful enough to run extensive operating systems and algorithms. Edge nodes can be made cognitive and adaptive using Machine Learning (ML) and Artificial Intelligence (AI) techniques, which improve system performance. However, safe and scalable EC is still a theory in progress, and it is the forthcoming paradigm shift that the FRACTAL aims to be part of.

1.2 Concept of FRACTAL subsystem

FRACTAL is a European Union-funded project aimed at developing a decentralized, federated edge computing infrastructure for IoT applications. The FRACTAL architecture consists of a set of interconnected edge nodes, which can communicate with each other to share resources and data. The edge nodes are connected to a central cloud infrastructure, which provides additional resources and services. One of the key features of the FRACTAL architecture is its use of a fractal structure, which allows for flexible and dynamic resource allocation. This structure allows nodes to self-organize into clusters, with each cluster capable of performing specific tasks or services. The use of a fractal structure also enables easy scaling, as new nodes can be added to the system without requiring significant changes to the existing infrastructure. This building block is a robust computing platform node capable of constructing an industry-standard Cognitive Edge (a network that predicts and diagnoses). The FRACTAL node will be the building block of scalable decentralized IoT ranging from smart low-power computing systems to high-performance Computing edge nodes [3].

The FRACTAL communication subsystem also supports dynamic spectrum access, which allows devices to access unused or underutilized spectrum bands to increase available bandwidth and reduce interference. The FRACTAL communication subsystem supports multi-hop communication using a variety of routing protocols, including traditional IP-based routing as well as more specialized protocols designed for FRACTAL nodes.

The cognitive node it creates will extend intelligence out to the network's edge. A new generation of intelligent systems will thus be possible thanks to the FRACTAL node. In such systems, the cloud will only be responsible for controlling and managing the edge nodes. The node will become cognitive as a result of AI, supported by internal and external architectures, and be able to predict both its performance and the state of the environment. It will adapt in real-time delivering new services to meet the needs of the environment. This will enable the system's overall intelligence. The architecture of the node should be open, safe, reliable, and low power. It should have high-performance capabilities while still being secure, safe, and low power. To decrease the need for cloud services, it should also be cognitive and autonomous. Finally, the node will employ cutting-edge communication (i.e., Fifth Generation (5G)) and storage facilities. The network's scalability will be enabled by the fractality of its configuration [2].

The following are the strategic goals to be achieved to implement and prioritize the various needs of a FRACTAL node:

- Design and implement an open-safe reliable platform to build cognitive edge nodes of variable complexity.
- Guarantee extra-functional properties (dependability, security, timeliness, and energy efficiency) of FRACTAL nodes and systems built using FRACTAL nodes (i.e., FRACTAL systems).
- Evaluate and validate the analytics approach utilizing AI to help the identification of the largest set of working conditions still preserving safe and secure operational behaviours.
- To integrate fractal communication and remote management features into FRACTAL nodes.

FRACTAL cognitive edge nodes will be displayed in two flavours: (1) a commercial node that is partially constrained and has a short time to market based on the Xilinx VERSAL computing platform, and (2) a more research-oriented node that is fully flexible and customizable based on Reduced Instruction Set Computer-Five (RISC-V) cores and accelerators and has a longer time to market exploration based on the open-source Parallel Ultra-Low Power (PULP) platform. Lastly, FRACTAL technologies will be applied across a variety of applications and domains, first integrating and validating them in 4 industrial verification use cases, and then validating them in 5 industrial validation use cases [4].

The three-layer FRACTAL architecture depicted in Figure 1 is a network architecture that has been proposed as a way to address the limitations of current wireless networks. The architecture consists of three layers: (1) The Access Layer, (2) The Control Layer and (3) The Core Layer. The access layer is responsible for providing connectivity to end devices, sensors, and other IoT devices. It consists of edge nodes that can be deployed in different locations, including remote sites, factories, or homes. This layer is designed to be flexible, scalable, and able to adapt to different user requirements. The Control Layer is responsible for managing the access layer and ensuring the efficient use of network resources. It includes network controllers that coordinate the activities of APs and make decisions about resource allocation and network optimization. This layer is designed to be intelligent and dynamic, able to adjust to changing network conditions and user requirements. Finally, the Core Layer is responsible for connecting the control layer to the Internet and other external networks. It includes gateways that provide access to the Internet and other external networks and that enable communication between different FRACTAL networks. This layer is designed to be secure and reliable, able to protect network traffic and prevent unauthorized access.

In FRACTAL, there are three tiers of hardware nodes that correspond to the three layers of the FRACTAL architecture. The first tier, also known as the Access Tier, consists of devices such as sensors, cameras, and smartphones that are located at the edge of the network. The second tier, known as the Edge Tier, includes gateways and small servers that provide processing and storage capabilities at the edge of the network. Finally, the third tier, referred to as the Core Tier, consists of large-scale cloud data centres that offer massive storage and computing capabilities. Together, these tiers form a hierarchical architecture that enables the efficient processing and management of data across the network.

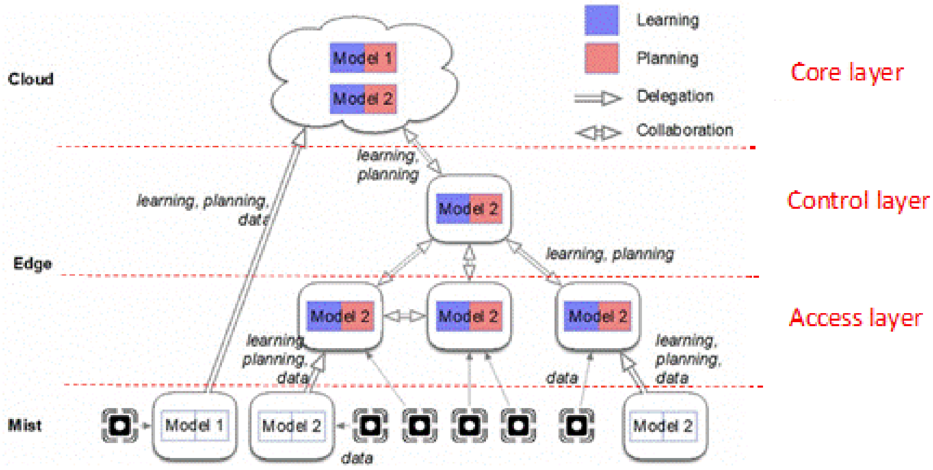

Figure 1. The three-layer FRACTAL architecture [5].

The architecture depicted in Figure 1 has a hybrid topology network with subnetworks that are formed using peer-to-peer, star, tree, and mesh. By combining these different topologies, the FRACTAL architecture can create subnetworks that are optimized for different purposes and can adapt to changing network conditions. For example, a mesh topology can be used to provide coverage over a large area, while a star topology can be used to provide efficient communication between a small number of devices. Overall, the hybrid topology of the FRACTAL architecture allows for a flexible and adaptable wireless infrastructure that can meet the diverse needs of modern communication systems. It should come as no surprise that we need to enable and support a variety of topologies and connection choices given the sheer diversity of the potential IoT use cases and applications and their unique requirements, which FRACTAL seeks to address. Two of these topologies can be regarded as the most prevalent ones:

- The tree (the star topology can be treated as a subcase of tree topology).
- The mesh (the peer-to-peer can be considered a subclass of mesh topology).

Note that the distance between the FRACTAL nodes may differ significantly, from a few meters to several kilometres depending on the desired application or use case. The application would also decide on the required wireless communication key performance metrics and traffic patterns. Generally, the FRACTAL node's capabilities (such as computational power and available resources) and communication requirements will rise as its tier increases. As a result, the upper-layer nodes are likely to use more complicated communication protocols to operate (typically build on top of Internet Protocol (IP)). This implies that IP-based communication must be supported by the underlying radio access technologies [5].

The FRACTAL communication subsystem is designed to provide a flexible and adaptive communication architecture that can support the unique requirements of Fractal Node Edge Computing (FNEC) networks. By incorporating cognitive radio, dynamic spectrum access, multi-hop communication, and cognitive networking, the FRACTAL communication subsystem can help to improve network performance, increase available bandwidth, and reduce power consumption.

1.3 Network Key Performance Indicators

Network KPIs are metrics for assessing a network's performance. They are frequently used to monitor many characteristics of a network including throughput, reliability, and efficiency. It is possible to measure a variety of network KPIs, and each one offers information on various characteristics of network performance. Some common network KPIs are latency, throughput, jitter, packet loss, bandwidth, network availability, power consumption, and packet loss.

Network KPIs are typically derived through a combination of monitoring, analysis, and modelling [6]. The first step in deriving network KPIs is to monitor the network. This involves collecting data on various network parameters, such as latency, throughput, jitter, packet loss, and bandwidth. Network monitoring tools such as network analyzers, protocol analyzers, and network probes can be used to collect data on network performance. Once network data has been collected, it is analyzed to derive network KPIs. This may involve calculating averages, standard deviations, and other statistical measures to gain insight into network performance. Analysis may also involve identifying trends and patterns in network data over time. Network KPIs can also be derived through modelling. This involves building mathematical models of the network and using these models to predict network performance under different conditions. For example, network models may be used to predict the effect of changes in network topology, traffic patterns, or hardware configuration on network performance. It is important to calibrate network KPIs to ensure that they accurately reflect network performance. This may involve comparing KPI measurements against known network benchmarks or testing the network under controlled conditions to validate KPI measurements. Finally, network KPIs are reported to stakeholders to provide insight into network performance. KPI reports may include charts, graphs, and other visualizations to help stakeholders understand network performance and identify areas for improvement.

Some KPIs that are commonly used to measure network performance may be less useful or even useless when applied to FNEC. This is because FNEC networks operate differently than traditional centralized networks and have unique characteristics that may make certain KPIs less relevant or accurate. Here are some reasons why some KPIs may be less useful in FNEC:

- Limited bandwidth: FNEC networks often have limited bandwidth compared to traditional networks, as they rely on edge devices to process and transmit data. This means that bandwidth-related KPIs may not accurately reflect network performance or may not be a primary concern in FNEC [7].
- Edge device processing power: FNEC networks rely heavily on the processing power of edge devices, which may vary in capability and may have different power requirements. This means that KPIs that focus solely on network performance may not take into account the processing capabilities and limitations of individual edge devices [8].
- Dynamic network topology: FNEC networks often have a dynamic topology [5], with edge devices joining and leaving the network frequently. This means that KPIs that are based on network topologies, such as network availability or packet loss, may not accurately reflect the current state of the network.
- Energy efficiency: FNEC networks often have a strong focus on energy efficiency, as edge devices may have limited power sources and high energy consumption can be a significant concern [5]. KPIs that do not take energy efficiency into account may not provide a complete picture of network performance.

In FNEC, the focus is typically on processing data efficiently and quickly, rather than on ensuring that the network is always available. Moreover, measuring network availability can be a complex task, as it requires monitoring the network continuously over time to ensure that it is operational. In some cases, it may be difficult or impractical to measure network availability accurately in an FNEC environment.

FNEC is designed to optimize data processing and reduce the amount of data that needs to be transmitted over the network. In this context, the focus is on processing the data efficiently at the edge, rather than transferring large amounts of data between edge devices and computing nodes. As a result, the amount of data transmitted over the network is typically low, which reduces the likelihood of packet loss. In some cases, a small amount of packet loss may not significantly impact the performance of FNEC systems. This is because the system is designed to handle data processing tasks in a distributed manner, which can help to mitigate the impact of packet loss. Thus, in the contest of FNEC, packet loss is not a critical KPI.

While bandwidth can still be a useful metric to consider in some cases, it may not be the most relevant or useful KPI for evaluating the performance of FNEC. Edge devices in Fractal Node Edge Computing have limited bandwidth compared to the computing nodes. Therefore, measuring bandwidth may not accurately reflect the actual performance of the system, as it does not account for the limitations of the edge devices. One of the key benefits of FNEC is that it allows for distributed data processing, which can reduce the amount of data that needs to be transferred between devices. As a result, bandwidth may not be a limiting factor for the performance of the system.

It is important to carefully consider which KPIs are relevant and useful in FNEC networks and to choose metrics that take into account the unique characteristics of these networks. Therefore, only latency, throughput, and power consumption KPIs are considered in this thesis.

1.3.1 Latency

Latency is a network KPI that measures the time it takes for data to travel from one point in the network to another. It is typically measured in milliseconds (ms) or microseconds (µs). Low latency is desirable as it indicates fast network performance and minimal delays in data transfer.

High latency can result in slow network performance and delays in data transfer. Latency can be affected by several factors, such as network congestion, the distance between network nodes, and the quality of network equipment. It can be measured using tools such as ping, traceroute, or network monitoring software. To improve latency, network administrators can take several steps, such as optimizing network configurations, upgrading network equipment, or using Quality of Service (QoS) mechanisms to prioritize important traffic.

Latency is a good KPI for Fractal Node Edge Computing (FNEC) because FNEC is a distributed computing paradigm that relies on real-time data processing and decision-making at the edge of the network. In FNEC, computing tasks are offloaded from centralized cloud servers to edge devices, such as smartphones, IoT sensors, or drones, to reduce latency and improve the efficiency of data processing.

Low latency is crucial for FNEC applications as it enables faster data processing and decision-making at the edge of the network. This is particularly important for applications that require real-time data processing, such as autonomous vehicles, smart grids, or medical devices. For example, in autonomous vehicles, low latency is critical for the timely detection of obstacles and safe navigation, while in medical devices, low latency can help ensure the timely diagnosis and treatment of patients. By measuring latency in FNEC, network administrators can identify potential bottlenecks or delays in data processing and take steps to optimize network performance. This can include optimizing network configurations, deploying edge devices closer to end users, or using QoS mechanisms to prioritize time-sensitive traffic. Overall, latency is an important KPI for FNEC as it directly affects the efficiency and effectiveness of data processing and decision-making at the edge of the network.

1.3.2 Throughput

Throughput is the amount of data that can be transmitted over a network in a given period. It is typically measured in bits per second (bps). High throughput is important for applications that require large amounts of data to be transmitted quickly, such as video streaming or file transfers. Throughput can be affected by several factors, such as network congestion, network equipment, network protocols, the behaviour of the end-user, the processing potential of the components of the system, limitations of the underlying physical medium, etc [9]. It can be measured using tools such as network monitoring software, speed tests, and traffic generators. When measuring throughput as a KPI, it's important to consider factors such as network congestion, network equipment, and network protocols that can affect the results. Network administrators should also establish a baseline for expected throughput and compare actual results against the baseline to identify potential bottlenecks or inefficiencies in data transfer. The following formula can be used to calculate the throughput of a network [10].

$$Throughput = (Total\ data\ transmitted) \ / \ (Time\ taken) \quad (1)$$

To improve the throughput, network administrators can take several steps, such as optimizing network configurations, upgrading network equipment, or using compression techniques to reduce the size of data transmitted over the network.

Throughput is a good KPI for FNEC for several reasons. Firstly, FNEC involves processing and analyzing data at the edge of the network, which requires efficient data transfer from the end devices to the edge nodes. Throughput directly impacts the speed and efficiency of data transfer, making it a critical KPI for FNEC. Secondly, FNEC involves processing and analyzing large amounts of data in real-time. This requires high bandwidth and low latency to ensure that data can be processed and analyzed on time. Therefore, throughput is an important factor in ensuring that the network can handle the data traffic required for FNEC. Moreover, FNEC often involves deploying edge devices in remote or challenging environments, where network connectivity may be limited. Measuring throughput as a KPI can help identify potential bottlenecks or inefficiencies in data transfer and allow network administrators to optimize network configurations or deploy additional edge devices to improve throughput and overall network performance.

1.3.3 Power consumption

Power consumption refers to the amount of energy used or consumed by network devices and infrastructure during their operation. Networks, including wired and wireless networks, such as local area networks (LANs), wide area networks (WANs), data centres, and telecommunications networks, require energy to power and operate the various components, including switches, routers, servers, access points, and other networking equipment. Energy-efficient networks can reduce operating costs and help reduce the environmental impact of network operations. To measure power consumption as a KPI in FNEC, it's important to consider factors such as the power consumption of individual devices, the efficiency of network infrastructure, and the power requirements of data storage and processing. This can be done through a combination of hardware and software monitoring tools, such as power meters and energy management software, that measure and analyze power consumption at different points in the network. The following formula can be used for power consumption calculations [11].

$$Power\ consumption\ =\ Voltage\ X\ Current \tag{2}$$

Power consumption represents the power consumed during the measurement period which can be expressed in watts or any other appropriate unit of power. To calculate power consumption, it is important to know the voltage and current values of the device or system being measured. This formula can be used to calculate the power consumption of various devices, systems, or processes based on their current consumption and the voltage for which they are operated or used. It's important to note that power consumption KPI can be measured at different levels of the FNEC network, such as individual edge devices, network switches, or the overall network. The formula above can be applied at any of these levels to measure the power consumption of that specific component of the network.

There are several reasons that make power consumption an important KPI for FNEC. FNEC often involves deploying edge devices in remote or hard-to-reach locations, where power supply may be limited or unreliable. Energy-efficient devices can help reduce power consumption and lower operational costs. On the other hand, FNEC can have a significant environmental impact, particularly if energy-intensive devices are deployed in large numbers. Measuring consumption as a KPI can help identify areas where power consumption can be reduced, leading to a smaller carbon footprint. Energy-efficient devices often have a longer lifespan than less-efficient devices, as they generate less heat and are less likely to overheat or fail. This can lead to lower maintenance costs and improved device reliability.

To measure power consumption as a KPI in FNEC, it's important to consider factors such as the current consumption of individual devices, the efficiency of network infrastructure, and the power requirements of data storage and processing. Thus, it is an important KPI for FNEC, as it can help reduce costs, lower environmental impact, and improve the lifespan and reliability of edge devices.

1.4 Summary of contributions

In this thesis, study the performance of 5G technology for cognitive edge nodes based on the FRACTAL edge platform over the University of Oulu 5G Test Network (5GTN). The study focuses on three KPIs: latency, throughput, and power consumption, and evaluates their performance in three different FRACTAL-based network architectures. The main contributions of this thesis are:

- Evaluation of 5G technology for cognitive edge nodes: The study investigates the suitability of 5G technology for cognitive edge nodes and provides an in-depth evaluation of its performance.
- A comprehensive evaluation of FRACTAL-based network architectures: The paper evaluates the performance of three different FRACTAL-based network architectures, providing a comparative analysis of their performance.
- Identification of factors influencing performance: The study identifies key factors that impact the performance of 5G connectivity for cognitive edge nodes, including network architecture, number of nodes, environment, and traffic type.
- Recommendations for optimizing performance: Based on the evaluation results, the paper provides recommendations for optimizing the performance of 5G connectivity for cognitive edge nodes, such as selecting the most suitable network architecture and optimizing nodes.

Overall, this thesis provides a valuable contribution to the field of wireless communication and cognitive edge computing by providing a comprehensive evaluation of the performance of 5G technology for cognitive edge nodes. The findings of this study can be useful for researchers, network operators, and policymakers working on wireless communication and cognitive EC.

1.5 Outline of thesis

The rest of this thesis is organized as follows: In Chapter 2, a review of the related literature is presented to provide a comprehensive understanding of the research area. Chapter 3 presents an analysis of the selected communication architectures for the cognitive edge node based on the FRACTAL edge platform. In Chapter 4, the implementation details of the selected communication architectures including the selection of Hardware (HW) and their configurations are discussed. Chapter 5 describes the experimental procedures used in the study, including the test plan and the tools used for testing, as well as the measurement procedures. Chapter 6 presents the obtained results from the experiments. In Chapter 7, the results are discussed, and conclusions are drawn in Chapter 8. Chapter 9 provides a summary of each main chapter of the thesis. Finally, Chapter 10 contains the list of references used in the study, and Chapter 11 presents the appendices, which include additional details and supporting information.

2 RELATED WORKS

Several research papers have been published on the FRACTAL architecture and its applications. Authors in [12] propose a machine learning-based approach for dynamic resource allocation in FRACTAL edge networks. The authors address the problem of limited resources in edge computing environments by using machine learning to predict the resource requirements of IoT applications. The proposed approach involves collecting data on resource usage patterns and training a machine learning model to predict resource needs based on historical data. The authors demonstrate the effectiveness of their approach through simulation experiments, showing that it can significantly improve resource utilization and reduce power consumption in FRACTAL edge networks. The paper concludes that machine learning can play a key role in enabling dynamic resource allocation in edge computing environments, allowing for efficient and scalable IoT applications.

The paper [13], provides an overview of the potential benefits of using Software-Defined Networking (SDN) in edge computing environments. The authors first introduce the concept of edge computing and its growing importance in supporting emerging applications such as IoT, augmented reality, and autonomous vehicles. They then explain the key features of SDN, which include centralized network management, programmability, and abstraction of network functions. The paper surveys existing research on the use of SDN in edge computing and identifies several use cases that can benefit from SDN, including network slicing, security, and energy efficiency. The authors also discuss some of the challenges and limitations of using SDN in edge computing environments, such as the need for low-latency communication and the potential for increased network complexity. The paper concludes with a discussion of future research directions, including the need for new SDN architectures and protocols that can better support edge computing, as well as the importance of considering the unique characteristics and requirements of edge computing applications when designing SDN solutions. Overall, the paper provides a useful overview of the potential benefits and challenges of using SDN in edge computing environments and highlights the need for further research in this area.

More recent work by [14] proposes a scalable IoT framework for energy management in connected buildings using the FRACTAL architecture. The authors address the challenges of energy management in connected buildings, including the need for real-time data analytics and dynamic resource allocation. The proposed Fractal IoT framework is based on the FRACTAL architecture, which is designed to support dynamic resource allocation and efficient data processing in edge computing environments. The authors demonstrate the effectiveness of their approach through a case study of a smart building, where they show that the Fractal IoT framework can improve energy efficiency and reduce energy costs. The framework involves the deployment of sensors and actuators throughout the building, which are used to collect data on energy consumption and environmental conditions. This data is processed using edge computing resources, and machine learning algorithms are used to predict energy consumption and optimize resource allocation.

Authors in [15] investigate the problem of resource allocation storms in edge computing environments. The authors identify reallocation storms as a key challenge in edge computing, where sudden and frequent changes in resource allocation can result in instability and reduced performance. The paper provides an analysis of reallocation storms in edge computing, using a simulation-based approach to evaluate the impact of different factors on storm severity. The authors consider various edge computing scenarios, including the allocation of resources to different tasks and the movement of tasks between different edge nodes. The authors also propose a dynamic resource allocation algorithm called Stormy, which is designed to mitigate the impact of reallocation storms in edge computing environments. The Stormy algorithm uses a reinforcement learning approach to optimize resource allocation and avoid sudden changes in resource allocation.

A useful overview of the FRACTAL project, highlighting the importance of developing secure and energy-efficient edge computing systems has been presented in [4]. The authors highlight the importance of edge computing in enabling new applications and services but also note the challenges of developing edge computing systems that are secure, energy-efficient, and scalable. The FRACTAL project aims to address these challenges by developing a hardware platform that supports cognitive and secure edge computing. The paper provides an overview of the FRACTAL hardware platform, which includes a set of microcontrollers, programmable logic devices, and sensors. The platform is designed to be energy-efficient, reliable, and secure, with support for hardware-based security features such as encryption and authentication. The authors also describe the FRACTAL software architecture, which is designed to support cognitive and secure edge computing applications. The software architecture includes a set of cognitive agents that can reason about the environment and make decisions based on data collected from sensors.

While the FRACTAL framework has provided valuable contributions to the field of edge computing, the relatively limited number of publications on this specific topic necessitates the examination of related research on AI-enabled EC, resource allocation, and optimization to provide a more comprehensive understanding of the current state of the art.

A paper published by Wang et al (2019) proposes the In-Edge AI framework, which combines Mobile Edge Computing (MEC), caching, communication, and federated learning to enable intelligent decision-making and data processing at the network edge [16]. They discuss the key components of the framework, such as the edge nodes, the caching layer, and the federated learning algorithm. Finally, the authors provide experimental results demonstrating the effectiveness of the In-Edge AI framework in improving the accuracy and efficiency of AI systems at the network edge. They conclude that In-Edge AI has the potential to enable a wide range of applications, including smart transportation, healthcare, and smart cities.

An overview of the role of the MEC in enabling edge intelligence for 5G and the IoT has been studied in [17]. The paper describes the architecture of MEC and its components, such as the MEC platform and MEC application, and discusses the potential benefits of MEC, such as improved performance, reduced network traffic, and increased security and privacy. It also explores the concept of edge intelligence, which refers to the use of machine learning and artificial intelligence algorithms at the edge of the network, and describes how MEC can facilitate the deployment of edge intelligence applications. The paper concludes by discussing the challenges and future research directions for MEC and edge intelligence, such as ensuring interoperability and standardization and developing novel machine-learning algorithms for resource-constrained edge devices. However, the paper does not present any specific results or empirical data related to these challenges or future research directions.

Research conducted by authors in [18] discusses the potential applications of edge intelligence, such as autonomous driving, smart healthcare, and industrial automation. The authors describe the challenges associated with developing edge intelligence, including limited computing power and storage on edge devices, and the need to develop new AI algorithms that are optimized for edge computing. The authors argue that edge computing offers significant advantages over traditional cloud computing, particularly in terms of latency, power consumption, and scalability. Overall, the paper is well-written and provides a comprehensive overview of the concept of Edge Intelligence. The authors provide a detailed analysis of the benefits of edge computing and argue that it offers a more efficient and effective approach to

AI than traditional cloud computing. The paper is particularly relevant in light of the increasing demand for low-latency, high-bandwidth applications, and the need to address privacy concerns associated with cloud computing.

A comprehensive review of performance evaluation metrics for cloud, fog, and edge computing has been presented in [19]. The paper presents a taxonomy of performance metrics and benchmarks that can be used to evaluate the performance of these computing paradigms. The paper identifies various challenges associated with evaluating the performance of cloud, fog, and edge computing, including the lack of standardized benchmarks, the diversity of hardware and software platforms, and the need to evaluate multiple performance metrics simultaneously. The authors provide a detailed taxonomy of performance evaluation metrics for cloud, fog, and edge computing, which includes metrics related to resource utilization, scalability, response time, reliability, security, and power consumption.

Authors in [20] provide valuable insights into the energy-performance trade-offs of Oscillatory Neural Networks (ONNs) based on VO2 devices for future edge AI computing. The study highlights the potential of ONNs based on VO2 devices as a promising approach for energy-efficient and high-performance edge AI computing and provides a basis for future research in this area. The results show that ONNs based on VO2 devices can achieve high energy efficiency and performance for edge AI computing. The authors also investigate the impact of different design parameters on the energy-performance trade-offs of ONNs, such as the number of neurons, the number of oscillators, and the frequency range. The study demonstrates that ONNs based on VO2 devices have the potential to overcome the limitations of traditional computing architectures for edge AI computing, such as high power consumption and limited processing capabilities. The authors also discuss the potential applications of ONNs based on VO2 devices in edge AI computing, such as speech recognition and image processing.

The paper [21], presents a model for admission control and latency-aware resource allocation for 5G MEC and evaluates its performance using simulations. The proposed admission control mechanism uses a two-stage approach, where incoming requests are first evaluated based on their latency requirements and then admitted based on the available resources at the MEC server. The authors also introduce a latency-aware resource allocation algorithm that considers the processing capacity and latency requirements of each application. The simulations demonstrate that the proposed admission control mechanism can effectively manage the admission of applications to the MEC server while meeting their latency requirements. The results show that the proposed mechanism can achieve a high acceptance rate of applications while maintaining a low latency and efficient resource utilization. The paper provides a valuable contribution to the field of 5G mobile edge computing by proposing a novel admission control mechanism that addresses the latency requirements of applications.

A novel approach for evaluating teaching quality using artificial intelligence and edge computing techniques has been proposed by the authors in [22]. The study evaluates the performance of the proposed approach using a dataset of teaching quality evaluation data and compares it with traditional approaches. The results demonstrate that the proposed approach can significantly improve the accuracy of teaching quality evaluation by leveraging the capabilities of artificial intelligence and edge computing. The study also highlights the potential of the proposed approach for various applications in the education sector, such as personalized learning and real-time feedback. The authors argue that the proposed approach can help to address the challenges of traditional teaching quality evaluation methods, such as subjectivity and time-consuming manual evaluations. Overall, the paper provides a valuable contribution to the field of edge computing and artificial intelligence by proposing a novel approach to teaching quality evaluation.

Authors in [23] provide a comprehensive review of the current state of the art in AI and edge computing, including various techniques and architectures for AI at the edge. They also discuss the potential of AI and edge computing for IoT-based applications, such as smart homes, smart cities, and industrial automation. The paper highlights the benefits of AI at the edge, including reduced latency, improved privacy and security, and increased scalability. The authors also identify some of the challenges and limitations of AI and edge computing, such as limited computing resources, heterogeneous networks, and data heterogeneity. To address these challenges, the authors propose a new perspective that combines AI and edge computing with other emerging technologies, such as blockchain, federated learning, and cloudlets.

The paper in [24] provides an overview of the various techniques and architectures for wireless power transfer, reviews the current research efforts in wireless-powered MEC networks, analyses the key performance metrics, and identifies future research directions and challenges. The authors review the current research efforts in wireless-powered MEC networks, including resource allocation, optimization, and performance evaluation. They discuss the challenges and opportunities in the design and implementation of these networks, such as energy harvesting, mobility management, and security. Furthermore, the paper provides a detailed analysis of the key performance metrics for wireless-powered MEC networks, such as power consumption, delay, throughput, and reliability. The authors compare the performance of different wireless power transfer techniques and architectures and provide insights into their trade-offs and suitability for different applications. The paper also identifies some of the future research directions and challenges in wireless-powered MEC networks, such as integrating renewable energy sources, developing efficient resource allocation and optimization algorithms, and ensuring the security and privacy of user data.

A study on the channel selection problem in cognitive radio-based IoT networks towards 5G has been presented in [25]. The authors investigate the impact of various Teletraffic attributes, such as data rate, packet size, and packet inter-arrival time, on channel selection performance in a cognitive radio network. The study is based on simulations using the NS-3 network simulator, and the authors evaluate the performance of three different channel selection algorithms: random selection, best channel selection, and threshold-based selection. The simulation results show that the threshold-based selection algorithm outperforms the other two algorithms in terms of throughput and packet delivery ratio, particularly when the Teletraffic attributes are taken into consideration. Moreover, the study shows that channel quality has a significant impact on channel selection performance and that network congestion can lead to increased packet loss and decreased throughput.

The paper [26] presents a promising approach to improve the energy efficiency of Cognitive Radio (CR) networks by leveraging the capabilities of MEC and cooperation among devices. The results show that the proposed cooperative MEC-enabled CR network architecture can achieve better energy efficiency and spectrum utilization compared to traditional cellular networks and MEC-enabled CR networks without cooperation. The simulation results also show that the proposed approach outperforms several benchmark schemes in terms of power consumption, network capacity, and fairness.

Authors in [27] propose a novel distributed spectrum management scheme for MEC-enabled CR IoT networks. The proposed scheme aims to improve spectrum utilization and reduce the power consumption of the network. The proposed scheme is based on a distributed learning algorithm, which allows the CR devices to learn the optimal spectrum access strategies through local interactions and the exchange of information with neighbouring devices. The proposed algorithm also takes into account the available resources of the MEC servers and offloads the computation tasks to the servers with sufficient resources. The results show that the proposed distributed spectrum management scheme can significantly improve spectrum utilization and reduce the power consumption of the network, compared to traditional CR networks without MEC. The simulation results also show that the proposed scheme outperforms several benchmark schemes in terms of spectrum utilization and energy efficiency.

The paper [28] proposes an anomaly detection scheme for cellular networks based on MEC and AI techniques. The proposed scheme employs MEC to enable real-time data processing and AI-powered anomaly detection to enhance detection accuracy. The authors evaluate the performance of the proposed scheme using a real-world dataset and compare it with traditional anomaly detection methods. The results show that the proposed scheme achieves better accuracy and detection rate while reducing the false alarm rate compared to the traditional methods. The authors conclude that the proposed scheme can effectively detect anomalies in cellular networks and has the potential to improve the quality of service in future 5G networks.

A framework for wireless edge computing that guarantees both latency and reliability for applications has been proposed by the authors in [29] which consists of three layers: a sensing layer, a computing layer, and a communication layer. The paper proposes a novel scheduling algorithm that considers both the latency and reliability requirements of the applications. The algorithm assigns resources to the applications based on their respective requirements, ensuring that the applications meet their deadlines and achieve a specified level of reliability. The proposed framework is evaluated using simulations, and the results show that it can provide latency and reliability guarantees for applications while efficiently utilizing the available resources. The paper also discusses several future research directions, including incorporating machine learning techniques into the framework and exploring the use of multiple wireless technologies for communication.

Previous studies ([16], [17], [21], [24], [27], [28]) have almost exclusively focused on MEC and its applications while the studies on dynamics resource allocation in edge computing ([12], [14], [15], [21], [24]) also widely discussed in the literature. The importance of the low latency requirement ([13], [18], [21], [29]) and energy efficiency ([14][14], [18], [20], [26], [27]) for future AI-enabled edge computing IoT applications has been also discussed in previous studies. Some limited publications particularly on FRACTAL architecture ([4], [12], [14]) can be found in the literature. Only a few works in the literature demonstrate the performance evaluation of edge computing ([19], [24]). However, these studies do not present any new experimental results or data regarding network performance such as latency, throughput, etc.

As far as I know, no previous research has investigated the characterization of the performance of wireless connectivity for cognitive node edge computing. Therefore, this thesis is focused on characterizing network KPIs of specific communication architectures based on the FRACTAL framework.

3 SELECTED COMMUNICATION ARCHITECTURES

FRACTAL nodes have various capabilities and communication requirements that allow them to operate effectively within the FRACTAL architecture. One key capability of FRACTAL nodes is their ability to operate using multiple Radio Access Technologies (RATs). This allows FRACTAL nodes to be deployed in different environments and to connect to different types of networks, depending on the specific needs of the system. Another important capability of FRACTAL nodes is their ability to operate in different network topologies, including peer-to-peer, star, tree, and mesh. This allows FRACTAL nodes to be deployed in various network configurations and to adapt to changing network conditions as needed. In terms of communication requirements, FRACTAL nodes require high-throughput and low-latency communication to support applications like video streaming, real-time gaming, and virtual reality. To achieve this, FRACTAL nodes utilize advanced modulation-coding schemes and media access control protocols, such as Orthogonal Frequency Division Multiple Access (OFDMA), to enable efficient use of the available frequency spectrum.

The selection of the best RAT for a particular EC application depends on several factors, including the required data transfer rates, latency, power consumption, and range. The choice of RAT will also depend on the specific requirements of the EC application and the availability of the required infrastructure. The communication requirements of the edge computing application should also be considered. For example, if the application requires high throughput, then a RAT topology that supports high data rates such as 5G should be selected.

FRACTAL aims to achieve low-latency communication for various applications, especially those that require real-time or near-real-time processing, such as industrial automation, augmented reality, and autonomous vehicles. In general, FRACTAL's target latency is in the order of milliseconds or even microseconds. Thus, the latency requirement is another important factor that should be considered when selecting a suitable RAT. If the application has stringent latency requirements, then a RAT topology that supports low latency such as IEEE 802.11 [32] should be selected. The range requirement of the application should also be considered. For example, if the application requires long-range communication, then a RAT topology that supports long-range communication such as cellular RATs [33] should be selected.

Power consumption is a critical consideration in wireless connectivity for FRACTAL nodes as they are often battery-powered devices with limited energy capacity. Low-power consumption is necessary to prolong battery life and reduce the need for frequent battery replacements or recharging. Various RATs have different power consumption profiles, with some being more power-efficient than others. For example, Zigbee and Bluetooth Low Energy (BLE) are designed for low-power IoT applications and consume very little power [34], while cellular technologies like 5G require more power due to their higher data rates and longer communication ranges. However, In addition to selecting a power-efficient RAT, other techniques can also be employed to minimize power consumption in FRACTAL nodes. These include duty cycling, where the device alternates between active and sleep states, and optimizing the data transmission protocol to reduce the amount of data transmitted and the frequency of transmission. By carefully managing power consumption, FRACTAL nodes can achieve long battery life and operate in remote or hard-to-reach locations where access to a power source is limited.

The infrastructure requirements of the application should also be considered. For example, if the application requires a large number of devices to be connected, then a RAT topology that supports mesh networking such as IEEE 802.11s or 6LoWPAN should be selected. The infrastructure requirements for FRACTAL wireless connectivity depend on the specific topology and radio access technology used. In general, FRACTAL networks require some form of infrastructure to support the communication between nodes. In the tree topology, a central gateway or base station is needed to manage communication with the nodes. In the mesh topology, nodes need to be able to communicate with each other directly or through intermediate nodes, which may require additional infrastructure such as relays or gateways. In the case of cellular connectivity, the infrastructure may include cell towers, base stations, and backhaul networks. The infrastructure also needs to support the power and latency requirements of FRACTAL networks. For example, low latency may require the use of high-speed backhaul networks, while low power consumption may require the use of energy-efficient equipment and infrastructure.

The cost of implementing a RAT topology in FRACTAL wireless connectivity can vary depending on various factors such as the number of nodes to be deployed, the distance between nodes, the required bandwidth, and the type of technology being used. For example, implementing a cellular-based RAT may require more infrastructure and investment than implementing an IEEE 802.11-based RAT. Similarly, implementing a mesh-based RAT may require more nodes to be deployed, which can add to the cost of the network. In general, the cost of implementing RATs in FRACTAL wireless connectivity should be considered alongside other factors such as the network's performance, scalability, and power requirements.

Ultimately, the choice of RAT for FRACTAL should be based on a thorough analysis of the specific use case and the trade-offs between performance, cost, and feasibility. By selecting the right RAT, FRACTAL can provide efficient and reliable wireless connectivity for edge computing, enabling a wide range of innovative applications in various industries.

As it was illustrated in Figure 1, FRACTAL is a three-layered architecture that aims to provide low-latency, reliable, and scalable wireless connectivity for edge computing. The choice of RAT is crucial to achieving these goals, as it determines the QoS and the efficiency of communication between the FRACTAL nodes. Several RATs are suitable for FRACTAL, each with its advantages and limitations. For instance, IEEE 802.11-based RATs, such as Wireless Fidelity (WiFi), provide high throughput and low latency under ideal conditions, but their range is limited, especially in the higher frequency bands. This makes them more suitable for last-mile connectivity within a FRACTAL subnetwork, rather than backbone connectivity between subnetworks. WiFi provides high data rates, with the latest WiFi 6 (802.11ax) standard offering speeds up to 9.6 Gbps while WiFi 4 (802.11n) practically offers a data rate of around 100-200 Mbps. Wi-Fi has low latency, with typical latencies ranging from a few milliseconds to a few tens of milliseconds. This is suitable for FRACTAL wireless connectivity, which requires low latency for real-time and interactive applications. WiFi has a large coverage area, with a range of up to hundreds of meters, depending on the environment and equipment used. This large coverage area is suitable for FRACTAL wireless connectivity, which requires connectivity over a wide area. On the other hand, WiFi infrastructure is widely available, with many public and private WiFi hotspots in cities, buildings, and homes. This infrastructure can be leveraged for FRACTAL wireless connectivity, reducing the need for additional infrastructure investments. In terms of implementation, WiFi is compatible with many devices, including smartphones, tablets, laptops, and IoT devices, making it a popular choice for wireless connectivity. This compatibility with existing devices can reduce the cost and complexity of deploying FRACTAL wireless connectivity.

On the other hand, cellular RATs, such as 5G, provide a wider range and better coverage, as well as support for mobility and handover. They can also provide high throughput and low latency, especially with the latest advances in radio technologies such as OFDMA. However, they require more infrastructure investments and can be more costly than IEEE 802.11-based

RATs, especially in terms of spectrum licensing fees. Compared to previous mobile networks, 5G offers significantly higher data rates than previous generations of cellular networks, with peak rates of up to 20 Gbps. This high data rate is achieved through the use of advanced modulation and coding schemes, as well as wider frequency bands. 5G networks have significantly lower latency with latency as low as 1 ms. This low latency is achieved through a combination of factors, including shorter transmission times and more efficient signalling protocols. 5G networks have significantly higher capacity than previous generations, due to the use of higher-frequency radio waves such as Milli-Meter Waves (MMW) and support of network slicing. This enables the network to support a large number of simultaneous connections with high data rates. In terms of security and scalability, 5G networks offer improved security features and are highly scalable.

Other RATs, such as Zigbee and Long-Range Wide Area Network (LoRaWAN), are not well-suited for FRACTAL connectivity, as they prioritize low-power, long-range communication over high-throughput and low-latency, which are essential for FRACTAL's edge computing applications. However, Zigbee can be suitable for certain applications in FRACTAL wireless connectivity, such as in smart homes and buildings where a large number of low-power devices need to be connected. Even though Zigbee has some potential as a suitable RAT for FRACTAL wireless connectivity, it has several limitations. Such as limited range, low data rate, interference, security, etc. Zigbee has a limited range of up to 10-100 meters, depending on the environment and interference. This may not be sufficient for some FRACTAL applications that require longer-range communication. Zigbee is designed for low data rates of up to 250 kbps, which may not be sufficient for some high-bandwidth FRACTAL applications. On the other hand, Zigbee has some security vulnerabilities, which can make it vulnerable to hacking and other cyber-attacks.

Bluetooth is also another RAT that is commonly used in IoT applications. While it offers several advantages like low cost, ease of use, and ubiquity, it may not be the most suitable wireless technology for FRACTAL wireless connectivity due to its limited range, interference issues, limited bandwidth, power consumption, and scalability. Bluetooth has a limited range of about 10 meters, which may not be sufficient for FRACTAL wireless connectivity, especially in large-scale deployments. While Bluetooth 5.0 has increased the range to around 100 meters, this is still less than what other wireless technologies like Wi-Fi and LoRaWAN offer. Bluetooth as well as Zigbee operates in the same 2.4 GHz frequency band, which is also used by other wireless technologies like WiFi. This can cause interference and affect the performance of Bluetooth devices, especially in environments with high wireless activity. Moreover, Bluetooth has a limited bandwidth of around 1 Mbps, which may not be sufficient for FRACTAL wireless connectivity use cases that require higher data rates. Bluetooth requires devices to be paired before they can communicate with each other, which may not be feasible in large-scale FRACTAL networks with a large number of nodes. These facts make Bluetooth less critical RAT for a wireless FRACTAL communication system.

LoRaWAN is a low-power, long-range radio access technology that uses an unlicensed radio spectrum to connect IoT devices. While it has many advantages for certain IoT applications, there are some technical reasons why it may not be the best choice for FRACTAL wireless connectivity. LoRaWAN has a limited bandwidth of up to 500 kbps [35], which may not be sufficient for some high-bandwidth FRACTAL applications that require faster data rates. LoRaWAN uses a random-access protocol, which can result in unpredictable latency and transmission delays. This may not be suitable for some FRACTAL applications that require low-latency communication. In terms of security requirements in FRACTAL, LoRaWAN uses a shared key security model, which can make it vulnerable to hacking and other cyber-attacks.

Thus, It may not be suitable in the contest of FRACTAL wireless connectivity. Additionally, LoRaWAN networks are limited in terms of scalability and the number of nodes that can be connected to the network. This may not be suitable for some FRACTAL applications that require a large number of connected devices.

The combination of WiFi and 5G can ensure seamless and uninterrupted connectivity. While WiFi is ideal for providing high-speed connectivity in local areas, it has a limited coverage range. On the other hand, 5G can provide extensive coverage and high-speed data transfer over long distances. By integrating these two RATs, users can enjoy uninterrupted connectivity in both local and remote areas. A hybrid WiFi and 5G RAT can improve network capacity and reliability. As the number of connected devices and data traffic increases, the WiFi network can become congested, leading to a decrease in network speed and reliability. With the 5G backbone, the load can be offloaded to the cellular network, increasing the network's capacity and reliability. Moreover, the hybrid RAT can enhance the overall user experience. By utilizing the best features of both RATs, users can enjoy fast and reliable connectivity, low latency, and an extended coverage range. This can significantly enhance the user experience, especially for bandwidth-intensive applications such as video streaming and online gaming.

Considering the above state-of-the-art of RATs, the following three distinct topologies have been selected for the characterization of the KPIs in FNEC.

- Topology 1: IEEE 802.11 based last mile.
- Topology 2: Direct cellular (5G) backbone.
- Topology 3: IEEE 802.11 based last mile over cellular (5G) backbone.

3.1 Topology 1: IEEE 802.11 based last mile

Figure 2 illustrates Topology 1, which indicates that the FRACTAL nodes use IEEE 802.11 WLAN radio access. This technology has been designed specifically for high-throughput star networks that operate in the 2.4 GHz and 5 GHz Industrial, Scientific, and Medical (ISM) bands. The technology combines wide frequency bands, advanced modulation-coding schemes, advanced media access schemes, and MIMO support, enabling high throughput and low latency under no or limited interference conditions. However, using above-2GHz ISM bands for operation prevents long-range communication. It is important to note that multiple implementations of mesh connectivity over IEEE 802.11 are available and have been reported in [30], [31]. In the case of FRACTAL nodes, the IEEE 802.11 technology can be used to provide connectivity to devices located in the last-mile region of the network. These devices can be connected to a FRACTAL node, which acts as a gateway to the rest of the network. The FRACTAL node can then communicate with other nodes in the network, either directly or through other gateways. The topology of the IEEE 802.11-based last mile in a FRACTAL network can be configured in various ways depending on the specific requirements of the network. For example, the last-mile connectivity can be provided through a mesh topology, where each node communicates with other nearby nodes to provide redundant paths and improve overall network resilience. Alternatively, a star topology can be used, where each device connects directly to a central FRACTAL node that acts as a hub for the network.

This topology is commonly used in residential and small business settings, where wired connections are not feasible or cost-effective. The IEEE 802.11 technology supports high-speed wireless communication and provides flexibility in network deployment, enabling easy extension and modification of the network coverage area. The use of IEEE 802.11 technology in the last mile topology has become increasingly popular due to its convenience, cost-effectiveness, and ease of use.

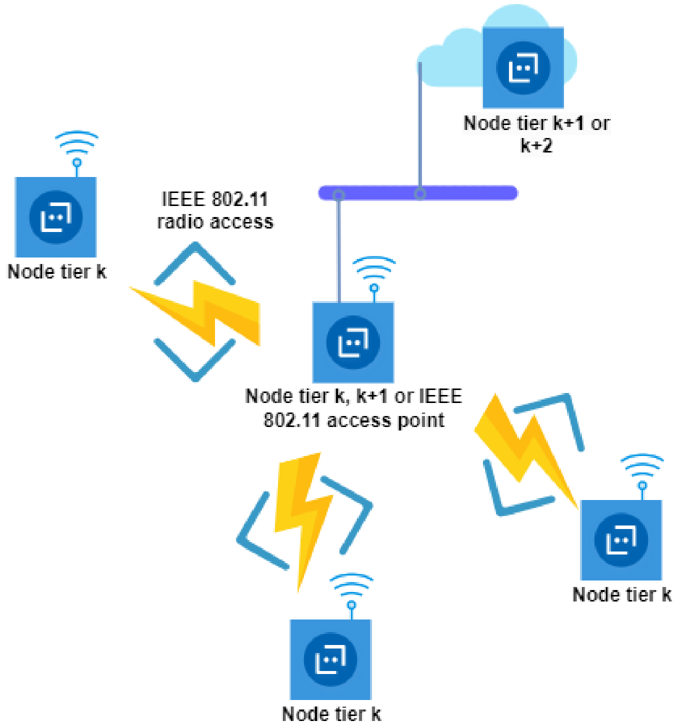

Figure 2. Topology 1: IEEE 802.11 based last mile.

3.2 Topology 2: Direct cellular (5G) backbone

Topology 2, as shown in Figure 3, proposes equipping FRACTAL nodes with cellular connectivity, specifically 5G. The use of OFDMA, new frequency bands, licensed spectrum, and controlled access to time-frequency resources offer both high throughput and extensive communication range. However, there are additional costs associated with enabling access to the spectrum, as well as significant infrastructure investments. It's worth noting that cellular technologies, ranging from First Generation (1G) to 5G, have been designed primarily with a tree-like topology in mind, where User Equipment (UE) communicates with a base station, and base stations merge into a centralized network with dedicated gateways providing Internet connectivity. As a result, peer-to-peer connectivity between two UEs, especially if they are nearby, may be inefficient. Additionally, it's important to note that the performance of a cellular network heavily relies on its configuration and available resources, such as the width of the frequency bands allocated for uplink and downlink communication in the case of Frequency Division Multiple Access (FDMA) or the amount of time allocated for uplink and downlink in the case of Time Division Multiple Access (TDMA). This performance may vary substantially from one network to another, between individual base stations, and even between sectors of a single base station within a network of a single operator.

This topology is commonly used in large-scale deployments, such as smart cities or industrial IoT applications, where high-speed and reliable connectivity is required between multiple devices and nodes. The use of 5G cellular technology in the backbone topology provides several benefits, such as high-speed data transmission, low latency, and the ability to handle many connected devices. Additionally, the direct cellular backbone topology eliminates the need for intermediate equipment, reducing infrastructure complexity and maintenance costs.

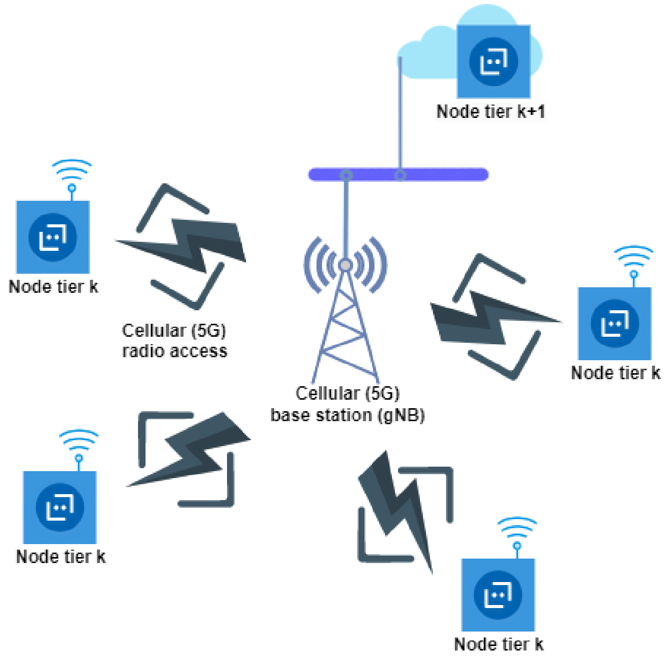

Figure 3. Topology 2: Direct cellular (5G) backbone.

3.3 Topology 3: IEEE 802.11 based last mile over cellular (5G) backbone

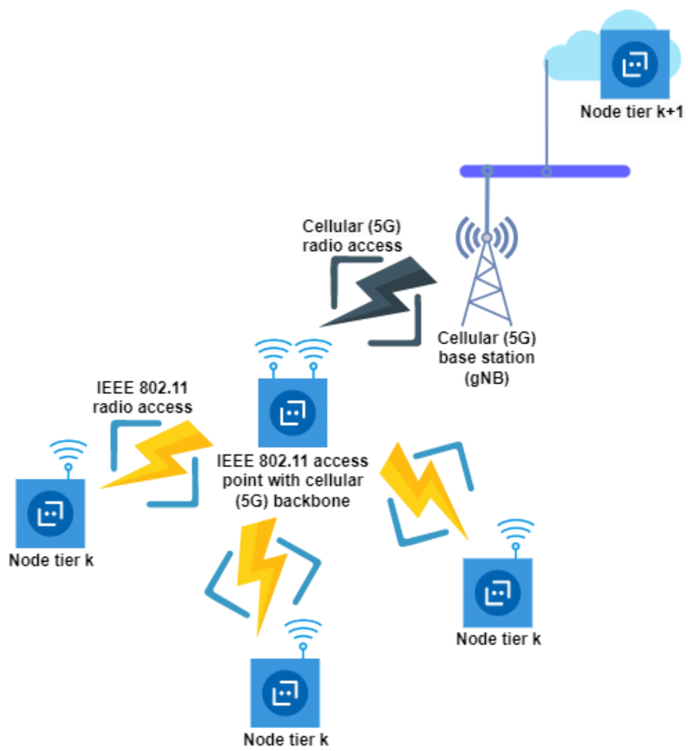

Figure 4. Topology 3: IEEE 802.11 based last mile over cellular (5G) backbone.

Topology 3, illustrated in Figure 4, focuses on the scenario where both technologies are combined. Specifically, a star or mesh IEEE 802.11 network is employed for communication between the same-tier FRACTAL nodes, and cellular (5G) radio access is used for communicating with the nodes at the upper tier. This hybrid topology is commonly used in scenarios where both high-speed connectivity and extended coverage are required. In this topology, the IEEE 802.11 wireless technology is used for the final leg of connectivity between the Internet service provider (ISP) and the end-user, while the 5G cellular technology provides connectivity between various network nodes, acting as a backbone for the last-mile network. The use of 5G cellular technology in the backbone topology provides high-speed and reliable connectivity between multiple devices and nodes, while the use of IEEE 802.11 technology in the last mile topology provides flexibility in network deployment and enables the easy extension and modification of the network coverage area. This hybrid topology provides a cost-effective solution for delivering high-speed connectivity over an extended coverage area.

4 IMPLEMENTATION

This chapter presents a summary of the implementation of the chosen communication architectures. The implementation includes the selection and configuration of different Hardware (HW) and Software (SW) platforms.

4.1 Topology 1: IEEE 802.11 based last mile

4.1.1 Hardware

To enable IEEE 802.11 wireless connectivity, two types of devices are required:

- An IEEE 802.11-compatible radio module for FRACTAL nodes.
- An IEEE 802.11-compatible AP to coordinate the communication between the FRACTAL nodes.

4.1.1.1 IEEE 802.11 radio module

Figure 5. Illustration of the test node equipped with TP-link AC1300 dongle.

Due to the popularity of the IEEE 802.11 standard, there are various choices for radio modules and modems that can be utilized. After evaluating factors such as cost, form factor, technical specifications, and driver availability for the software platforms, the TP-link AC1300 Archer T3U Plus WiFi Universal Serial Bus (USB) dongle [36] was chosen. The accompanying image in Figure 5 depicts the test node with the attached dongle. This dongle uses the USB 3.0

interface for connection and supports both the 2.4 GHz and 5 GHz frequency bands while operating according to IEEE 802.11 a/b/g/n/ac standards. The maximum supported transmit power is 18 dBm in the 2.4 GHz band and 20 dBm in the 5 GHz bands.

4.1.1.2 IEEE 802.11 access point

To enhance the capabilities of the system, The TeleWell Industrial 5G AP [37] was chosen as the AP. This router is specifically designed for M2M applications and is shown in Figure 6. This AP is compatible with LTE bands B1 – B5, B7, B8, B12 – B14, B17 – B20, B25, B26, B28 – B30, B46, B66, B71, as well as 5G bands n1 – n3, n5, n7, n8, n12, n20, n28, n38, n40, n41, n48, n66, n71, n77 – n79. It features four cellular 5G antennas and two WLAN antennas and can deploy both 2.4 GHz and 5 GHz networks supporting standards 802.11 b/g/n. It has two SIM card slots, and a cellular connection can be established via SIM-A or SIM-B. The AP can be powered with 9 to 36 V DC power input, and DC 12 V/2.5 A power adapter was used for this purpose. The manual for the AP [37] provides instructions on how to prepare it for use (such as power supply and antenna connection, Subscriber Identity Module (SIM) card), and how to deploy it. It should be noted that enabling cellular connectivity is not necessary for IEEE 802.11-based communication alone, but this functionality will be utilized later as part of Topology 3 and discussed in Section 4.3. The decision to choose this AP was primarily due to its flexibility in being able to be used for both IEEE 802.11 and integrated IEEE 802.11 and cellular (5G) tests.

Figure 6. Illustration of the TeleWell Industrial 5G AP used as IEEE 802.11 AP.

4.1.2 Software and configurations

4.1.2.1 Enabling support of WLAN at the test node

To enable the support of WLAN using Archer T3U Plus AC1300 dongle at the test node, it is required to install Realtek RTL88x2BU WLAN USB Driver. Step by step guide for the installation of this driver has been given in [38].

4.1.2.2 Configuration of WLAN access point

To access the router's web User Interface (UI), the user needs to log in using the router IP address, which is set to 192.168.123.254 by default. To do this, the user should enter http://192.168.123.254 in a web browser on any computer that is connected to the same WLAN. By default, the username and password for logging in are both 'admin', but for security reasons, it is recommended to change them. The steps for changing the login credentials can be found in the router's user manual [37].

To enable cellular connectivity, it is necessary to set the Access Point Name (APN) according to the ISP. This can be done through the connection setup window (Basic Network > WAN & Uplink > Connection Setup) of the router's web UI. By changing the Dial-Up profile option to 'Manual-configuration', the user is allowed to set the APN for either SIM card.

Since the IP address is assigned by the cellular network only to the AP, in case a node needs to be accessible by the upper tiers node, there is a need to enable port forwarding. For this, we enable the virtual computer port forwarding function (Basic Network > Port Forwarding > Virtual Server & Virtual Computer) using AP's web UI. Virtual Computer allows us to assign Local Area Network (LAN) hosts to global IP addresses so that they can be visible to the outside world [37]. The example configurations are shown in Figure 7.

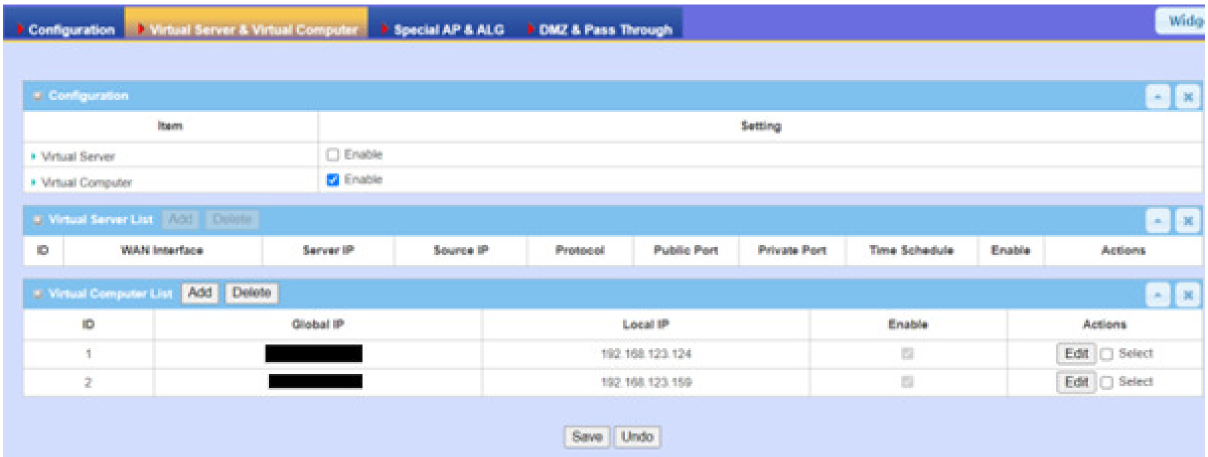

Figure 7. Illustration of port forwarding configuration for TeleWell Industrial 5G AP.

Note that the IP address allocated by the cellular network to the AP can be obtained from WAN Interface IPv4 Network Status in Status > Basic Network > WAN & Uplink. As a result of these configurations, once the node is connected to AP through WLAN, the local IP address of the node can be obtained from the Status > LAN & VLAN. It is also possible to configure a static IP address on the test node using the Media Access Control (MAC) address of the WiFi dongle and an IP address chosen from the IP Pool of the AP Dynamic Host Configuration Protocol (DHCP) server. To do this, we need to navigate to the DHCP Server window from Basic Network > LAN & VLAN > DHCP Server. By clicking on the Fixed Mapping button, we can enter the MAC address of the WiFi dongle and an IP address chosen from the IP Pool of the DHCP server. The MAC address of the WiFi dongle can be obtained from 'ifconfig' Linux terminal command once the test node is connected to the WiFi network. Further information about the configuration of the AP can be obtained, if needed, from its manual [37].

4.2 Topology 2: Direct cellular (5G) connectivity

4.2.1 Hardware

To establish a direct cellular (5G) connection, the Quectel RMU500-EK Evaluation Board (EVB) which is equipped with the RM500Q 5G module was chosen. The user guide for the EVB with RM500Q [39] provides detailed instructions on how to prepare the EVB for use, including power supply, disassembly and assembly, SIM card insertion, and attachment of the module to the evaluation board. To power the EVB, it can be connected to an external power adapter via the power jack (J101) on the EVB, or it can be powered through the USB-C interface using the power jack (J301) on the EVB. For this thesis, a power jack (J301) was used to connect the EVB with the host. At the time when development was carried out (i.e., in 2022 and early 2023) the RM500Q is the only 5G module commercially available and adapted for IoT usage, thus selection of this platform had no alternatives. Figure 8 shows the EVB with RM500Q (to the right) connected to the test node (to the left) using a USB-C cable.

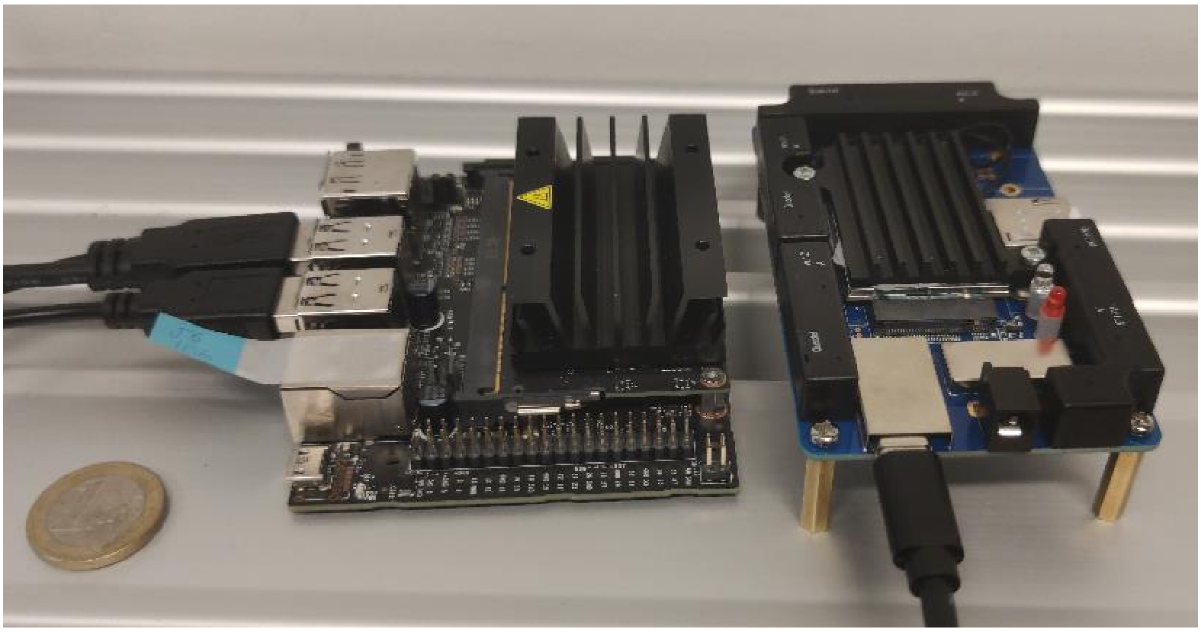

Figure 8. Illustration of test node connected to Quectel RM500Q 5G board over USB-C.

4.2.2 Software and configurations

To establish a connection to the 5G network using the Quectel RM500Q 5G module, it is necessary to install Linux 5G USB drivers on the test node. After testing various options, it was determined that the 'GobiNet' driver is the optimal choice for establishing a 5G connection. Once the driver installation is complete, it is essential to execute the appropriate attention/Hayes (AT) commands to the RM500Q 5G module to establish the connection. To issue AT commands to the RM500Q, it is necessary to install the Quectel USB Serial Option Driver. Additionally, the 'Qmi_WWAN' driver, which is also a 5G USB driver, was installed. Depending on the USB network adapter, either the 'GobiNet' or 'Qmi_WWAN' driver will be utilized. To set up a data call, Quectel's Connect Manager (quectel-CM) tool was used.

The Sequential instructions for the installation of the 5G drivers on the test node, quectel-CM tool, and the testing of the AT commands functionality including other dependencies have been given in Appendix 1 and Appendix 2. Additionally, Once the above steps were completed, AT commands given in Appendix 3 were used for a successful connection to the 5GTN.

4.3 Topology 3: IEEE 802.11 based last mile over cellular (5G) backbone

Topology 3 is a hybrid version of Topology 1 and Topology 2. Thus, the same HW and SW used in section 4.1 were also used here.

5 EXPERIMENTAL PROCEDURES

This chapter is started by discussing the produced test plan, then details the measurement and other tools used for testing and describes the way the required functionalities (e.g., time synchronization to enable accurate measuring of the delay or deployment of a Secure File Transfer Protocol (SFTP) server used for stress traffic testing). In the last subsection, details of the measurement procedures are presented.

5.1 Test plan

Table 1. Produced test plan and notations.

Test identifier (ID)	Description of the test
Test 1	A test node communicating to an upper-tier node (server) over cellular (5G) connection; office environment
Test 2	A test node communicating to a same-tier node over cellular (5G) connection; office environment
Test 3	A test node communicating to an upper-tier node (server) over IEEE 802.11 with cellular (5G) backbone connection; office environment
Test 4	A test node communicating to a same-tier node over IEEE 802.11 with cellular (5G) backbone connection; office environment
Test 5	A test node communicating to an upper-tier node (server) over IEEE 802.11 connection; office environment
Test 6	A test node communicating to a same-tier node over IEEE 802.11 connection; office environment
Test 7	A test node communicating to an upper-tier node (server) over cellular (5G) connection; direct line-of-sight to the base station
Test 8	A test node communicating to a same-tier node over cellular (5G) connection; direct line-of-sight to the base station
Test 9	A test node communicating to an upper-tier node (server) over IEEE 802.11 with cellular (5G) backbone connection; direct line-of-sight to the base station
Test 10	A test node communicating to a same-tier node over IEEE 802.11 with cellular (5G) backbone connection; direct line-of-sight to the base station

The set of tests specified in Table 1 was created to comprehensively test and measure key performance metrics of interest for different configurations. Test IDs 5 and 6 correspond to Topology 1, IDs 1, 2, 7, and 8 focus on Topology 2, and IDs 3, 4, 9, and 10 correspond to the hybrid topology which is Topology 3. It is worth noting that for Topologies 2 and 3, measurements were carried out for two radio channel conditions of the cellular link, one with a direct line-of-sight (LoS) between the cellular modem and the base station antenna, and another with the cellular modem located indoors without LoS to the base station. For additional information regarding the locations and measurement environment, please refer to Chapter 5. Furthermore, each of the test IDs consists of three subtests, as detailed in Table 2.

In total, 10 tests x 3 subtests = 30 experiments have been executed. In each experiment, along with the metrics of interest, we also monitored and logged the relevant information about the radio channel conditions (e.g., Received Signal Strength Indicator (RSSI), Reference Signal Received Power (RSRP), Reference Signal Received Quality (RSRQ), Signal-to-Interference-Noise Ratio (SINR) and Identifier (ID) of the cell, to which the nodes were connected). More details about the test procedure are discussed in the following sections and Figures 11-16 further illustrate the network architecture during the different tests.

Table 2. Subtest composition in each test

Subtest identifier (ID)	**Description of a subtest**
Subtest x.a, x=1..10	Measurement focuses on estimating the delay for small-size packets. The *ping* command is used to generate the traffic. The payload of the ping packet is set to 10 bytes and the period of transmissions is set to 1 second. Unless stated otherwise, the measurement for 1000 packets is done.
Subtest x.b, x=1..10	Measurement focuses on estimating the delay for bigger packets. The *ping* command is used to generate the traffic. The payload of the ping packet is set to 900 bytes and the period of transmissions is set to 1 second. Unless stated otherwise, the measurement for 1000 packets is done.
Subtest x.c, x=1..10	Measurement focuses on estimating the throughput. The throughput capability is measured through two approaches: (i) the conventional speed tests and (ii) by deploying an SFTP server and measuring the time required to upload and download a test file (1.6 GB[1], unless stated otherwise)

[1] As the test file we used the Nvidea OS image, available: https://developer.nvidia.com/embedded/-l4t/r32_release_v7.1/t210/tegra_linux_samplerootfilesystem_r32.7.1_aarch64.tbz2

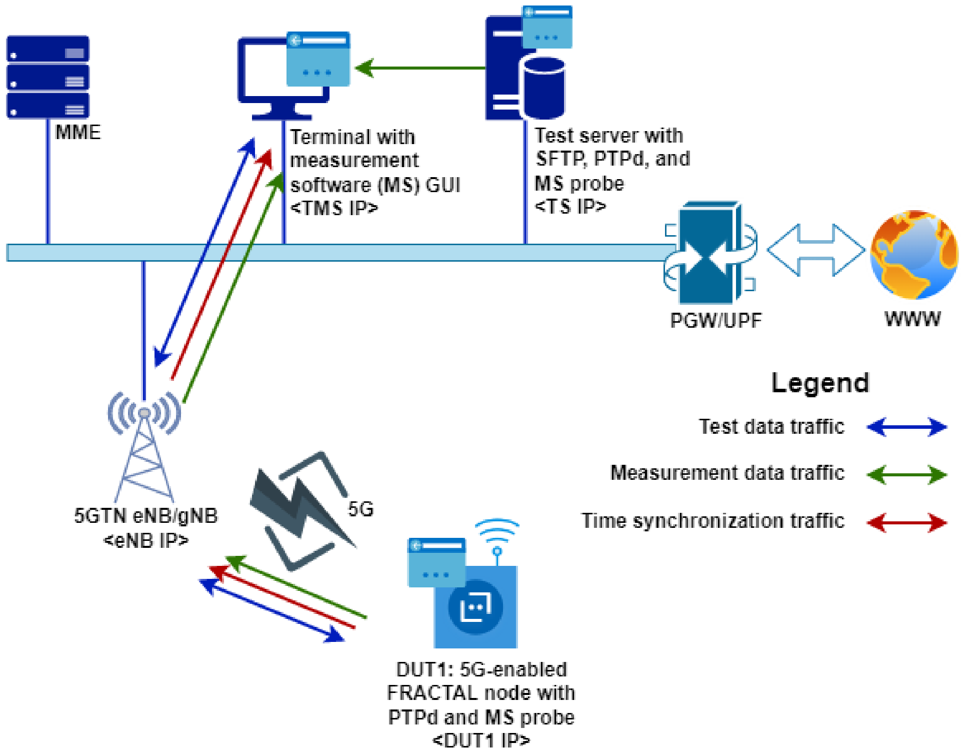

Figure 9. Network architecture for Tests 1 and 7.

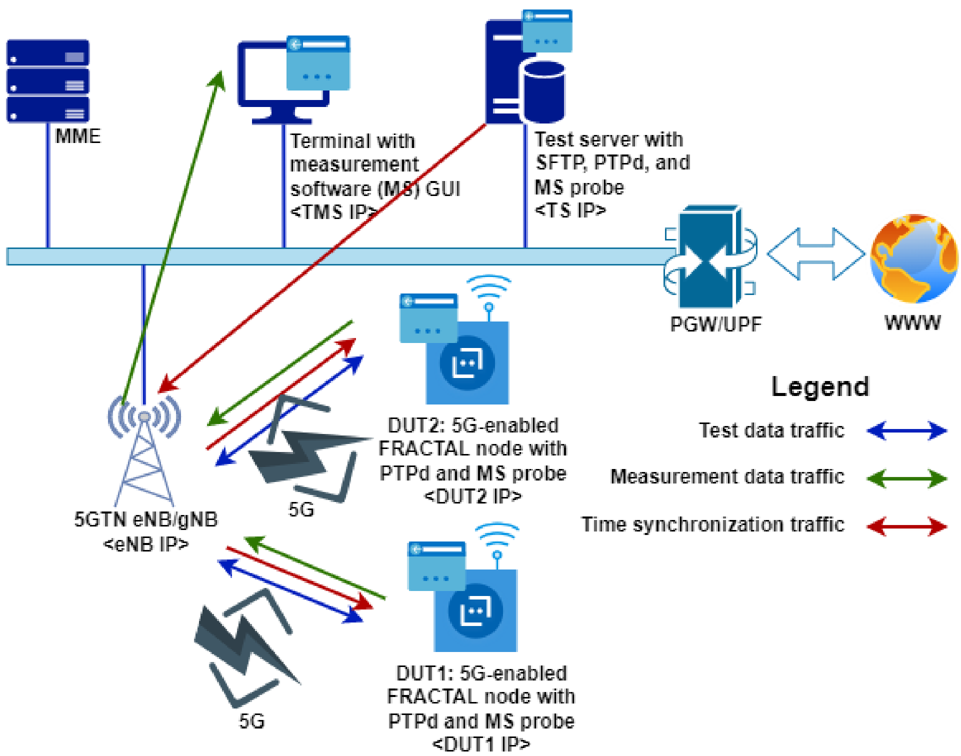

Figure 10. Network architecture for Test 2 and 8.

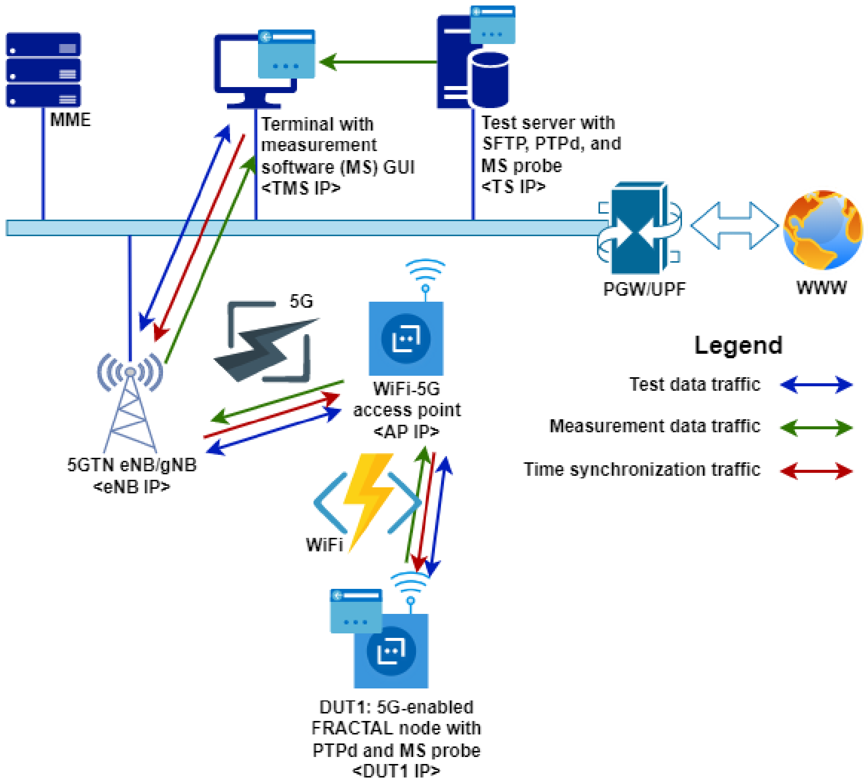

Figure 11. Network architecture for Tests 3 and 9.

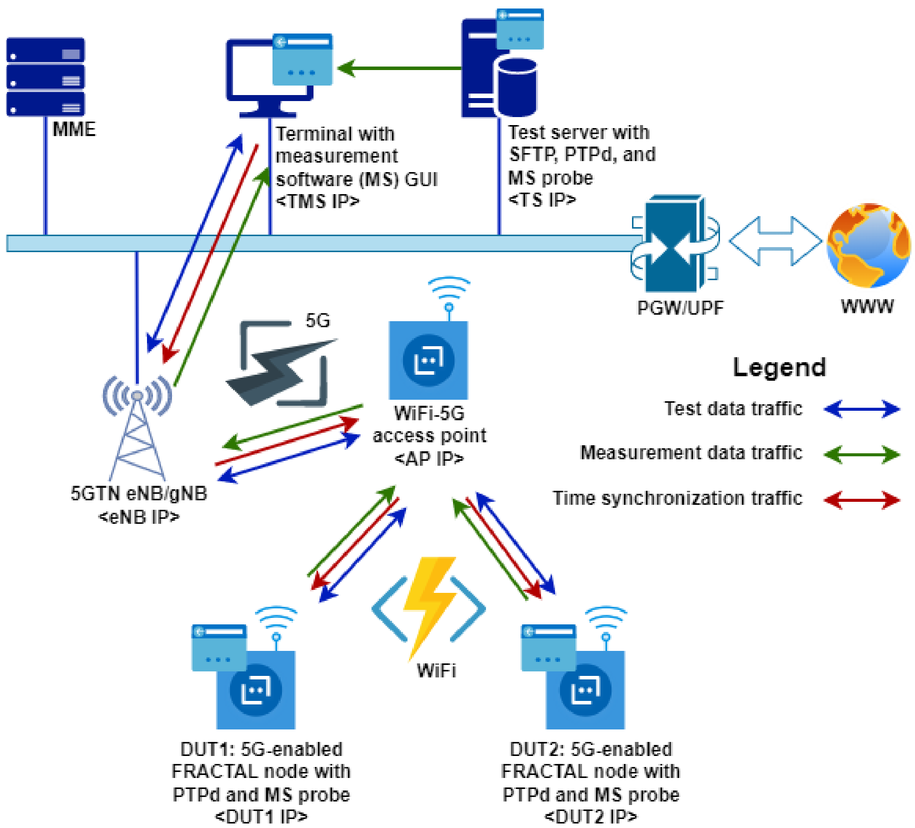

Figure 12. Network architecture for Tests 4 and 10.

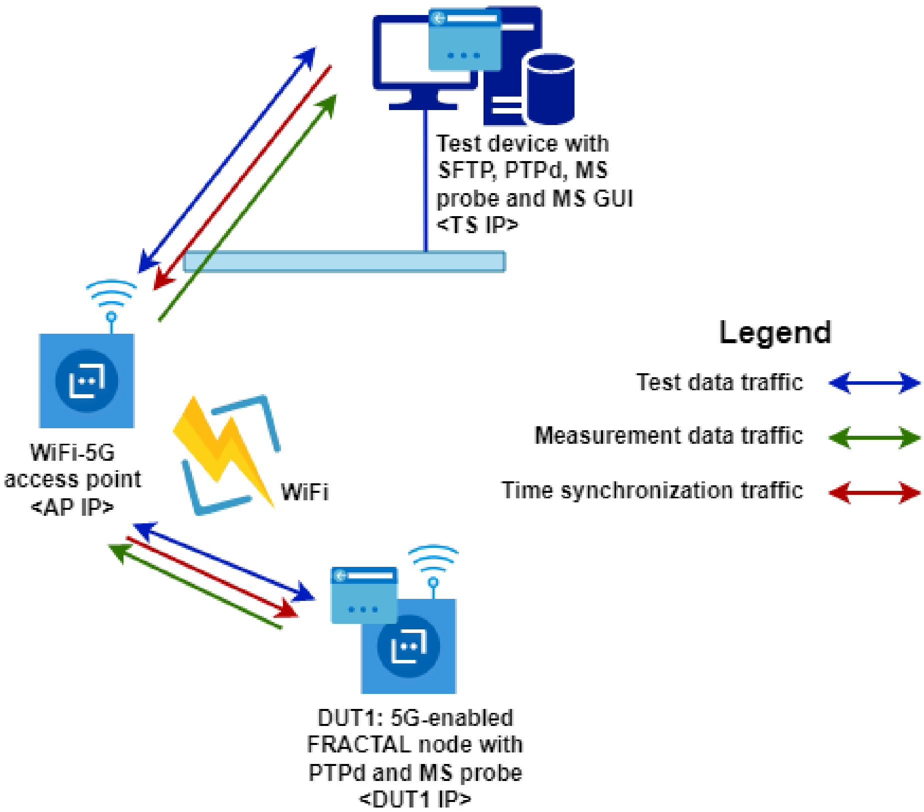

Figure 13. Network architecture for Test 4.

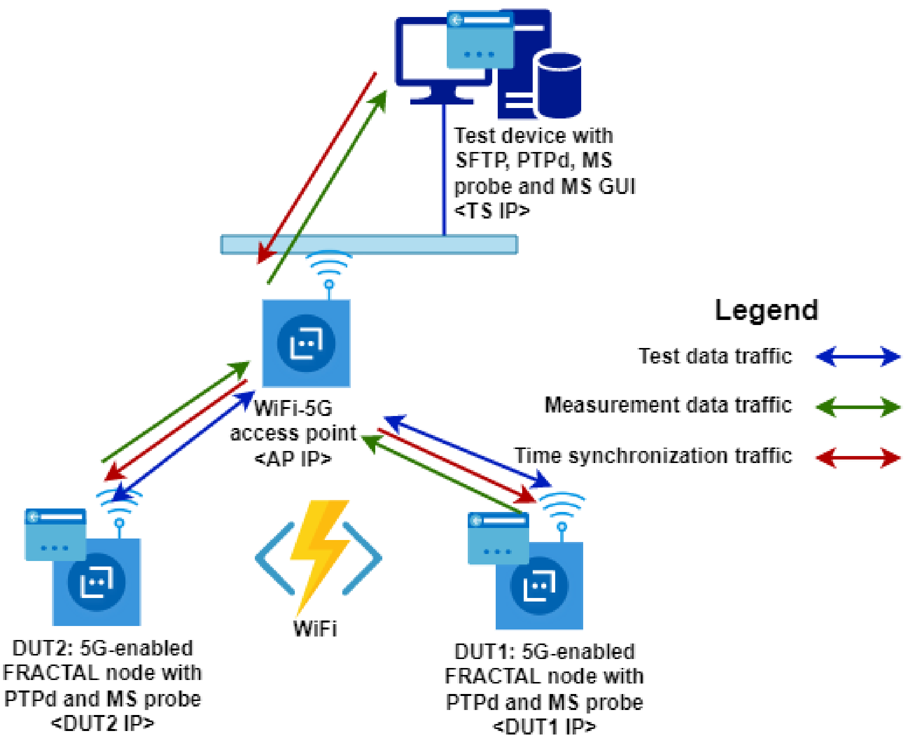

Figure 14. Network architecture for Test 5.

5.2 Experiment environment

The experiments and measurements were conducted within the 5GTN, which is a complete 5G test system and the world's first open 5G test network. The 5GTN is a national Finnish joint effort involving the University of Oulu, the Technical Research Center of Finland, and fifteen industry partners, and it functions as a full-scale micro-operator. Currently, the 5GTN is an integral part of the 6G Flagship Program being undertaken by the University of Oulu, Finland.

The 5GTN has been designed and implemented to support various research and industry needs and experiments while remaining scalable. It offers radio coverage in several locations throughout Finland, including Oulu, Tampere, Ii, Sodankylä, and Ylivieska. Specifically, in Oulu, radio coverage is provided in several areas, such as the Oulu city centre, University Hospital, VTT, Technology Park, University of Oulu, and University of Applied Sciences campuses, as well as industry research and development premises in Rusko.

The 5GTN possesses several main features, including the use of both non-standalone (NSA) and standalone (SA) 5G architecture. This enables dual connectivity for compatible devices that can utilize both LTE and New Radio (NR) access. Additionally, the 5GTN supports both fourth Generation (4G) and 5G connections through 4G and 5G Base Stations. Moreover, the 5GTN has several advanced features, such as the capability to use its own SIM cards, a core network implemented in a cloud environment, and the ability to utilize four different Evolved Packet Core (EPC) systems, including EPC (CMM17), OpenEPC, Cumucore EPC, and Open5GS. Additionally, it has a Bluetooth-based tracking system with 200 nodes, LoRa, WiFi, and IoT networks, and over 400 IoT sensor platforms in operation at the campus. The 5GTN also has both centralized and distributed computing servers and Graphics processing units (GPUs), with edge servers and MEC available. Frequencies used by 5GTN include 700MHz (B28), 2100MHz (B1), 2300MHz (B40), 2600MHz (B7), 3.5GHz (n78), and 26GHz (n258). The network has multiple base stations and antennas, including three 5G Macro cells (n78), indoor 5G radios (n78), one 5G mmW cell (n258), Macrocell (B28) with Narrow Band (NB)-IoT and Cat-M, Macrocell (B7) LTE-FDD, Macrocell (B42) LTE-TDD, and 20+ Pico Base Stations.

Cloud-RAN (cRAN)/Open-RAN (o-RAN) capable HW is being implemented to enable cloud-based RAN architecture and the adoption of the o-RAN standard. This will provide greater flexibility and scalability in network deployment and operation. The introduction of a larger Open Air Interface (OAI) capability will enhance the capacity and performance of the 5G network. OAI is an open-source SDN platform that enables the implementation of various wireless communication standards. The newest MEC and its application technologies will be introduced to enable low-latency and high-bandwidth computing at the edge of the network. This will enable new applications and services that require real-time data processing and analysis. In the 5GTN, The 100 Gbit/s backbone network will provide high-speed and high-capacity connectivity between different network elements and data centres. The SDN-based core network infrastructure will provide greater flexibility and control in network management and configuration. More dynamic 5G network slicing will enable the creation of more customized and optimized network slices for specific use cases and applications. Thus, The enlargement of 5G coverage in the Oulu area will provide more extensive test environments and use cases for researchers and industry partners.

The network is under constant development and evolution towards a more comprehensive 5G and future Sixth Generation (6G) test network, with plans to introduce MMW technology in FR2 in several phases, the newest UE's available, the renewal of 5G base stations, and the 5G core renewal. The structure of the 5GTN and its primary components are illustrated in Figure 15.

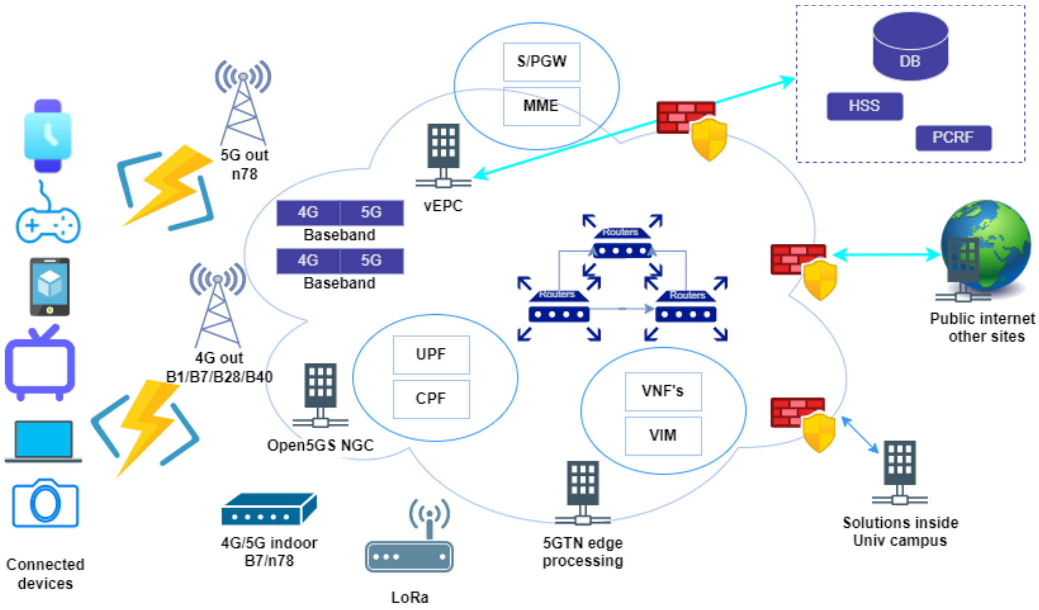

Figure 15. 5GTN structure and key elements.

In the experiments described below, a 5G NSA macro three-cell base station operating in the n78 frequency band and using TDMA was utilized. The available resources included 10 MHz of spectrum for 4G and 60 MHz of spectrum for 5G. The antenna tower that hosted the base station's antennas and its surroundings is depicted in Figure 16. The experimental setup and test locations are presented in Figure 17. The minimum ground-level distance from the antenna location to the test site was approximately 61 meters for test IDs 1-6 and 48 meters for test IDs 7-10.

Figure 16. The antenna tower of the University of Oulu hosting the 5GTN base station's antenna used in the tests.

Figure 17. Bird’s view on the experimental location (orange triangle denotes the position of 5GTN base station’s antenna, green circle marks the test location for tests ID 7-10, and purple circle signalizes the test location for tests 1-6.

5.3 Measurement tools

This section of the report describes the hardware and software tools used during the experiments to collect data and generate traffic. The test network consisted of one or two FRACTAL nodes for each test, an upper-tier FRACTAL node that served as an edge server in the 5GTN core network, and a telecommunication infrastructure that varied depending on the test.

When testing cellular (5G) communication, the telecommunication infrastructure included the base station of 5GTN and the 5GTN core network, while IEEE 802.11 AP was used for IEEE 802.11-based communication. For tests with cellular connectivity, an additional computer located in the 5GTN core network was used to synchronize measurement probes using Precision Time Protocol (PTP) and collect data. To ensure security and prevent possible attacks, a laptop equipped with an SFTP server, PTP, and measurement probes was used to conduct tests based on IEEE 802.11 without cellular connectivity.

Furthermore, to facilitate the execution of the experiments and switching between the different tests and subtests, an analysis of the required HW and SW components was carried out, and then instrumented two test FRACTAL nodes with all the needed SW components to allow all the planned tests. The summary of the components used in every test and the overall set of components installed on different devices is presented in Table 3. The deployment of the “communication-enabling” SW components has been already detailed in Section 4 while the following section individually discusses the HW and SW elements utilized in the tests.

Table 3. SW components of the individual network elements required for the tests

Test and subtest	**Test node 1 (device-under-test 1)**	**Test node 2 (device-under-test 2)**	**Test server (i.e., node of the upper tier)**	**Test laptop**
1 and 7	- cellular drivers - Qosium probe -SFTP client -PTPd master/slave	n/a	- Qosium probe - Qosium scope -SFTP server -PTPd master/slave	n/a
2 and 8	- cellular drivers - Qosium probe -SFTP client -PTPd master/slave	- cellular drivers - Qosium probe -SFTP server -PTPd master/slave	- Qosium probe - Qosium scope -PTPd master/slave	n/a
3 and 9	- IEEE 802.11 drivers - Qosium probe -SFTP client -PTPd master/slave	n/a	- Qosium probe - Qosium scope -SFTP server -PTPd master/slave	n/a
4 and 10	- IEEE 802.11 drivers - Qosium probe -SFTP client -PTPd master/slave	- IEEE 802.11 drivers - Qosium probe -SFTP server -PTPd master/slave	- Qosium probe - Qosium scope - PTPd master/slave	n/a
5	- IEEE 802.11 drivers - Qosium probe -SFTP client -PTPd master/slave	n/a	n/a	- Qosium scope - Qosium probe - PTPd master/slave
6	- IEEE 802.11 drivers - Qosium probe -SFTP client -PTPd master/slave	- IEEE 802.11 drivers - Qosium probe -SFTP server -PTPd master/slave	n/a	- Qosium scope - Qosium probe - PTPd master/slave
Overall	*- IEEE 802.11 drivers* *- cellular drivers* *- Qosium probe* *-SFTP client* *-PTPd master/slave*	*- IEEE 802.11 drivers* *- cellular drivers* *- Qosium probe* *-SFTP client* *-SFTP server* *-PTPd master/slave*	*- Qosium probe* *- Qosium scope* *-SFTP server* *-PTPd master/slave*	*- Qosium probe* *- Qosium scope* *-SFTP server* *-PTPd master/slave*

5.3.1 Test node

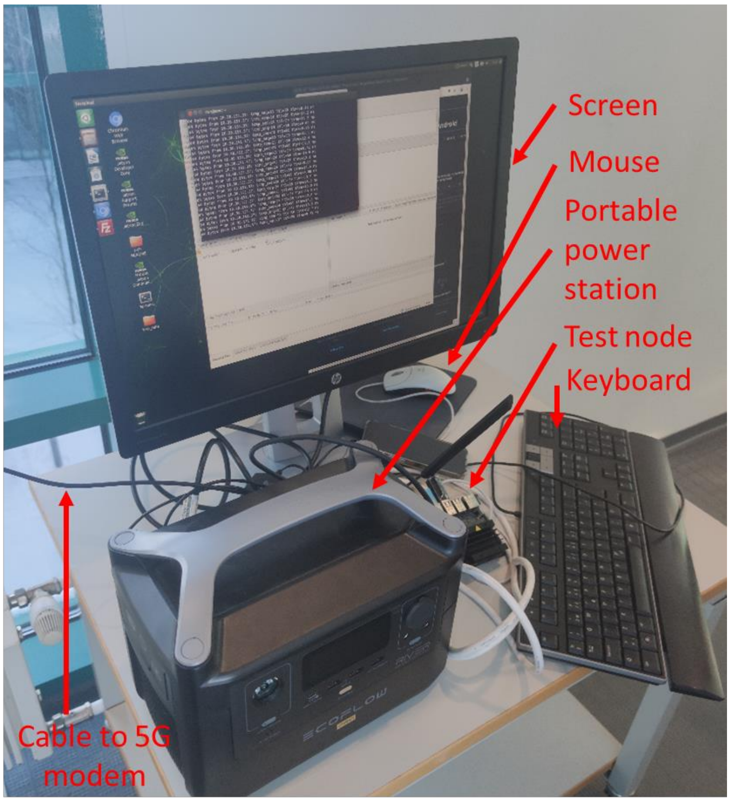

Figure 18. Photo of a test bed (pre-trials of Test 7) & the key elements of the test bed.

The experiments utilized two test nodes, each equipped with a complete set of required SW components (refer to Table 3). The 5G modem was attached to the test nodes during tests 1, 2, 7, and 8, while the IEEE 802.11 modem was attached during other tests. Additionally, A USB mouse and keyboard were installed in each node for control and interaction purposes, and the node's status was monitored using a screen connected via a High-Definition Multimedia Interface (HDMI) cable. Figure 18 illustrates the test bed used during pre-trials for Test 7. The test nodes and AP were powered using alternating current (AC) and powered throughout the tests, except for power supply measurements.

5.3.2 Qosium tool

The Qosium tool from Kaitotek OY [41] was utilized for accurately measuring the communication key performance indicators. Qosium is a passive QoS and Quality of Experience (QoE) real-time performance measurement and monitoring system for wired and wireless networks. It has features for network performance visualization and measures the QoS of real applications on the network without causing major disruptions. The tool is composed of two components - the network probes, which are the passive measurement tools deployed at the points to be measured, and a single Scope SW tool used to control the measurements and log the data. More information about the tool, its capabilities, and set-up procedures are available from its manual [42]. Note that the manufacturer provided a specialized pre-compiled version of the Qosium probes for deployment on the test nodes, and the deployment was done following the instructions provided by the manufacturer. In Qosium, traffic statistics are calculated in single points only. It's worth noting that sent traffic for the primary measurement point is received traffic from the perspective of the secondary measurement point. An illustration of the Qosium Scope Graphical User Interface (GUI) taken during pre-trials is depicted in Figure 19.

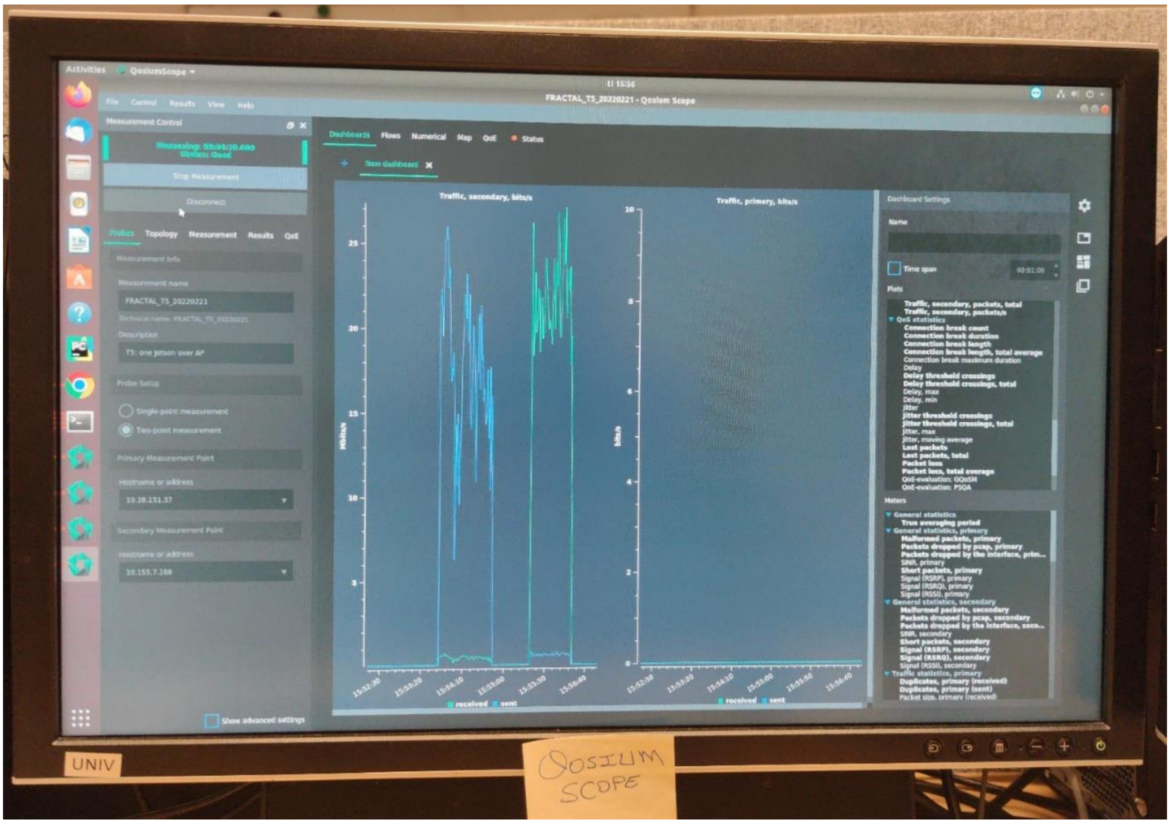

Figure 19. Photo of Qosium Scope interface taken during pre-trials (illustrating two-point measurement of throughput between a test node and a server for a 100 MB file).

5.3.3 DC power analyzer

The N6705B Direct Current (DC) power analyzer from Agilent/Keysight was employed to characterize the power consumption of the test devices. The tool allows for configuring the output voltage from 0 to 20 V and observing and recording the current consumption with a sampling rate of up to 50,000 samples per second. The accuracy of voltage configuration is below 0.1%, and the accuracy of current measurement is below 0.1%. More information on the tool is available from [43]. For the tests, the power analyzer was configured to operate in data logger mode, recording the current consumption profile of the test node through all phases of its operations. The collected data were further imported for post-processing in MATLAB software. The test bed utilized for power consumption measurements is illustrated in Figure 20.

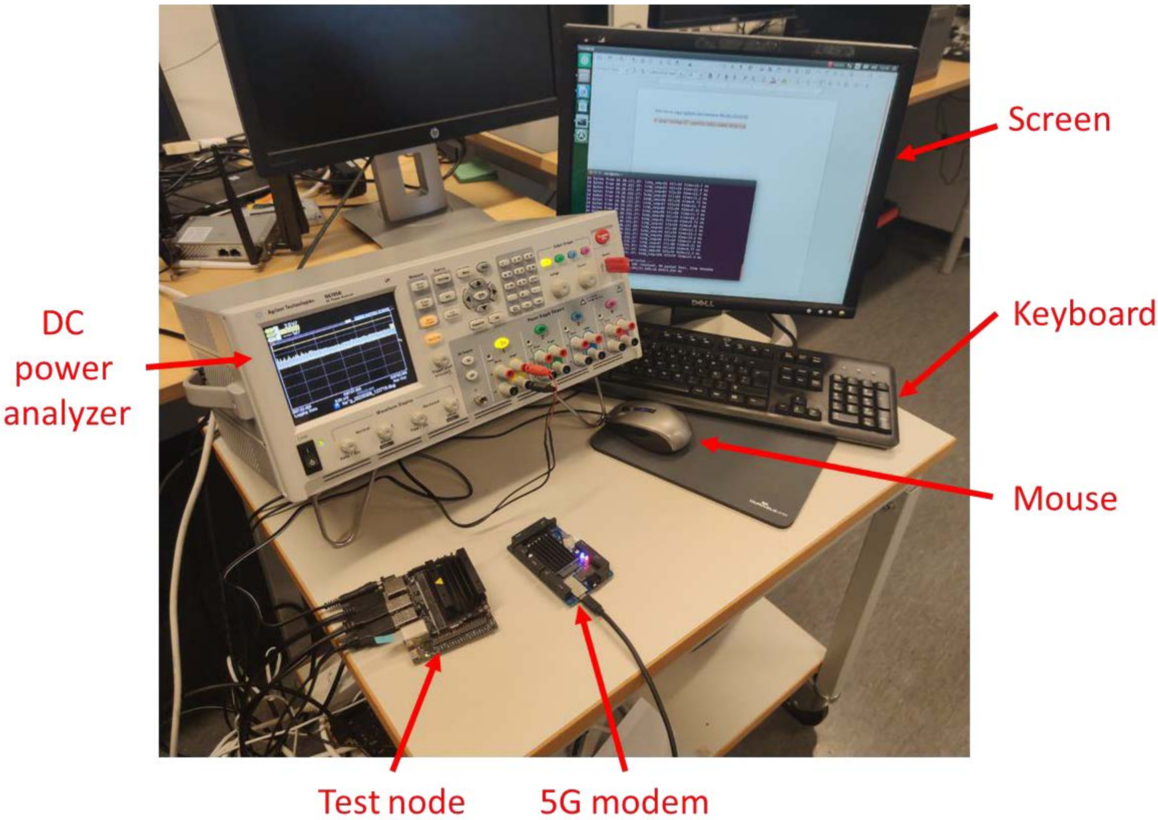

Figure 20. Photo of test bed during power consumption measurement (for cellular (5G).

It should be noted that to measure power consumption, the test device must be connected and directly powered by the DC power analyzer. To facilitate this, a cable was instrumented to allow the connection of the DC power analyzer directly to the barrel connector of the test node, and the jumpers were modified to make the test node operate from the supply provided to the barrel connector. It should also be noted that the USB mouse and keyboard, which were used to control the test nodes, were powered from the test node's USB interface, and their consumption was included in the total measured value. However, the screen was equipped with its power supply unit, and thus its consumption (except for the consumption for communication and current leakage through HDMI) was not accounted for. In the same way, another cable was instrumented to measure the consumption of the AP, which was connected directly to the DC power analyzer.

5.3.4 Test laptop

For conducting tests that required a direct Ethernet connection between the AP and an external computer (tests ID 5 and 6), we used the Lenovo ThinkPad T15 notebook computer, which is battery-powered and equipped with an Intel Core i5 64-bit processor and Ubuntu 20.04.4 LTS operating system. The Ethernet drivers required for the tests are pre-installed in Lenovo computers, eliminating the need for additional configurations. To obtain the local IP address of the notebook computer, we connected it to the AP via Ethernet and accessed the menu Basic Network > LAN & VLAN on the AP. The software components required for the tests (as listed in Table 6) were installed on the notebook computer similarly to how they were installed on the test node.

5.3.5 Portable power station

During the tests, the power supply issue in different locations was addressed by powering the test bed from an EcoFlow 720Wh Pro Portable Power Station [44]. Each of the three AC power outputs is capable of providing 600 W power, which is well above the expected consumption level of the test bed.

5.3.6 Speed test

For validating the connection and obtaining additional information about the performance of connectivity from a test device to the Internet (note, that the main test points for the ten tests described above are located in the local network, radio access network of 5GTN, or in the core network of 5GTN and thus these results are not directly comparable to the results, which can be obtained with the speed test), a popular Internet speed test tool https://www.speedtest.net/ was utilized. The GUI of the Speedtest is shown in Figure 21.

Figure 21. Photo of Speedtest interface taken during pre-trials.

5.3.7 Time synchronization

The System clock synchronization step is mandatory for those systems where applications rely on the system clock to schedule traffic or present data. In our tests, PTP was needed to ensure that the Qosium probes feature the same time reference. PTP is a protocol used to synchronize clocks in a network. The PTP-synchronized clocks are arranged in a master-slave hierarchy. The slaves are in sync with PTP masters which may be slaves to other masters. The PTP is a master-slave clock synchronization algorithm, which automatically builds and updates the hierarchy [45]. Several Linux distributions provide a package for Linux PTP. In this thesis, the

PTPd Linux package was used. Installation and configurations of the PTPd on the test nodes have been given in Appendix 4.

5.3.8 Traffic generation: SFTP

To generate traffic load, the SFTP is used, which is a network protocol designed for secure file access, transfer, and management of large files and sensitive data over a Transmission Control Protocol (TCP)/IP network via port 22. SFTP requires both a client and a server to function. Users can utilize the SFTP client to connect to a server and store files, while the SFTP server is responsible for storing and retrieving files. When a file is clicked on by a user, a request is sent over the network and eventually reaches the server. The requested data is then received by the requesting device. Moreover, SFTP ensures that all data transfers are encrypted before being transferred.

FileZilla software was used as the SFTP client. FileZilla is a free SFTP client software and can be downloaded from the software centre in a Linux Operating System (OS). SFTP server installation can be done using the steps given in Appendix 4.

5.3.9 Testing

To ensure the correct operation of the individual network components and the whole testbed, as well as to ensure the integrity of the planned tests and their results, a set of pre-tests was carried. First, the elements of the testbed were tested independently to ensure the stability of their operation. Next, three sets of pre-trials were organized during the three first weeks of February 2023 to validate interoperability and detect potential conflicts between the test bed components, familiarize with the tools used, their configurations, and data formats, and validate and tune the measurement procedures. During these trials, all the critical deficiencies, capable to prevent the execution of the measurement campaign, were detected and eliminated.

5.4 Measurement procedures

There were three target topologies as described in Section 5, each with two subcases. The first subcase involved two test nodes communicating with each other, while the second subcase involved a test node communicating with a Test Server (TS). In total, we considered six measurement scenarios, and for two of the topologies, measurements were carried out in two different radio channel conditions, resulting in 10 tests in total. For each measurement scenario, two types of measurements were performed. The first type estimated delay (subtests x.a and x.b, x=1..10), which was carried out using the Ping command. The second type addressed throughput and was carried out using an SFTP server and client (subtest x.c, x=1..10). For measuring the delay, the following ping command was used.

- sudo ping <secondary probe local IP> -s <size in bytes> -c <number of pings>

For throughput measurements, the SFTP server (located either at the server at Edge or on another test node, depending on the test) has been used. First, the client established the connection to the server. Next, a test file was uploaded to the SFTP server. After completing, the very same file was downloaded by the client from the server. The SFTP client was controlled through its GUI. The general measurement procedure for each test is composed of several steps as listed below.

- Step 1: Power up the test devices and give them time to boot up.
- Step 2: Enable PTP time synchronization for accurate timestamping.
- Step 3: Launch measurement in Qosium Scope.
- Step 4: Perform the Speetest at Device Under Test (DUT)1 and log the results.
- Step 5: Use Traceroute to check the route between the test nodes (i.e., DUT1 and DUT2/TS) and log the results.
- Step 6: Measure the radio channel conditions and log the results.
- Step 7: Start the experiment (either latency or throughput, as discussed above).
- Step 8: During the execution of the experiment notes were filled in containing the timestamps of the measurement, the observed throughput, and other relevant notes on the experiment run.
- Step 9: After the end of the experiment: log the results (i.e., start/end time and total time required for transferring the test file in uplink/downlink as reported by SFTP client, or log the results of executing the ping command).
- Step 10: Stop the measurement in Qosium Scope.
- Step 11: (done after all the measurements are completed): Copy the logs and collected by Qosium results and stored for further processing.

A more detailed step-by-step guide to the measurement procedure is given in Appendix 5. Figures 22-28 illustrate the test bed and environment, which were taken during the tests.

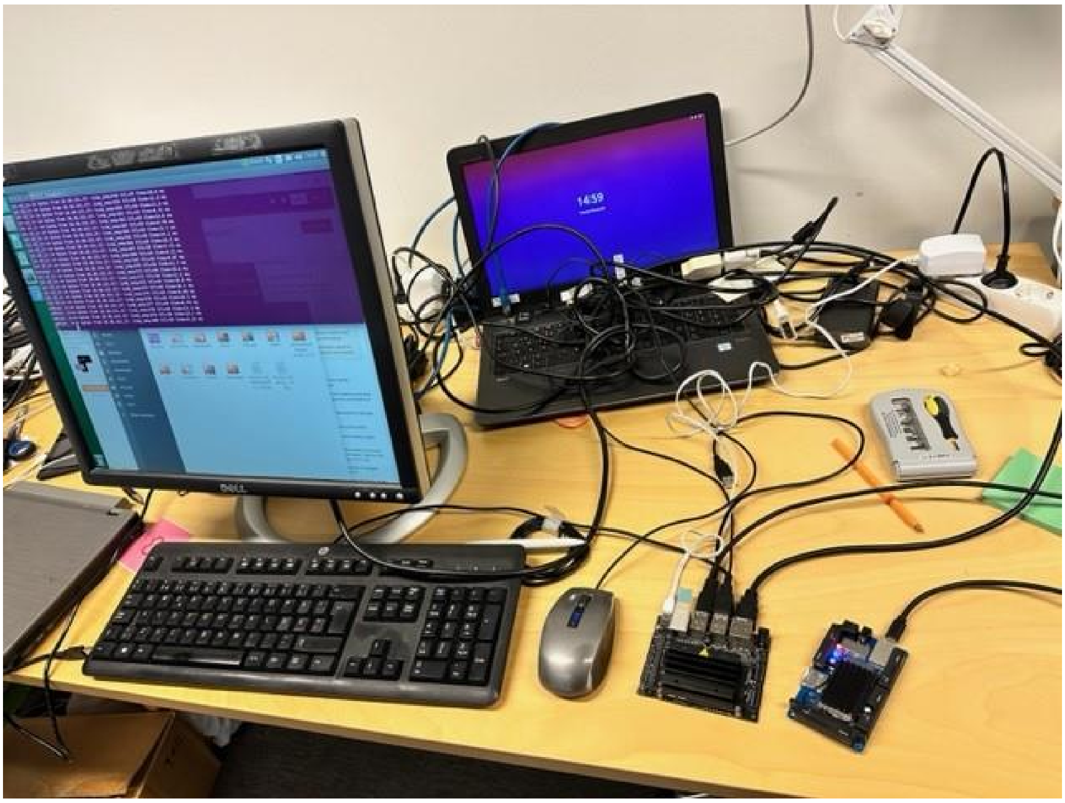

Figure 22. Test bed during Test ID 1.

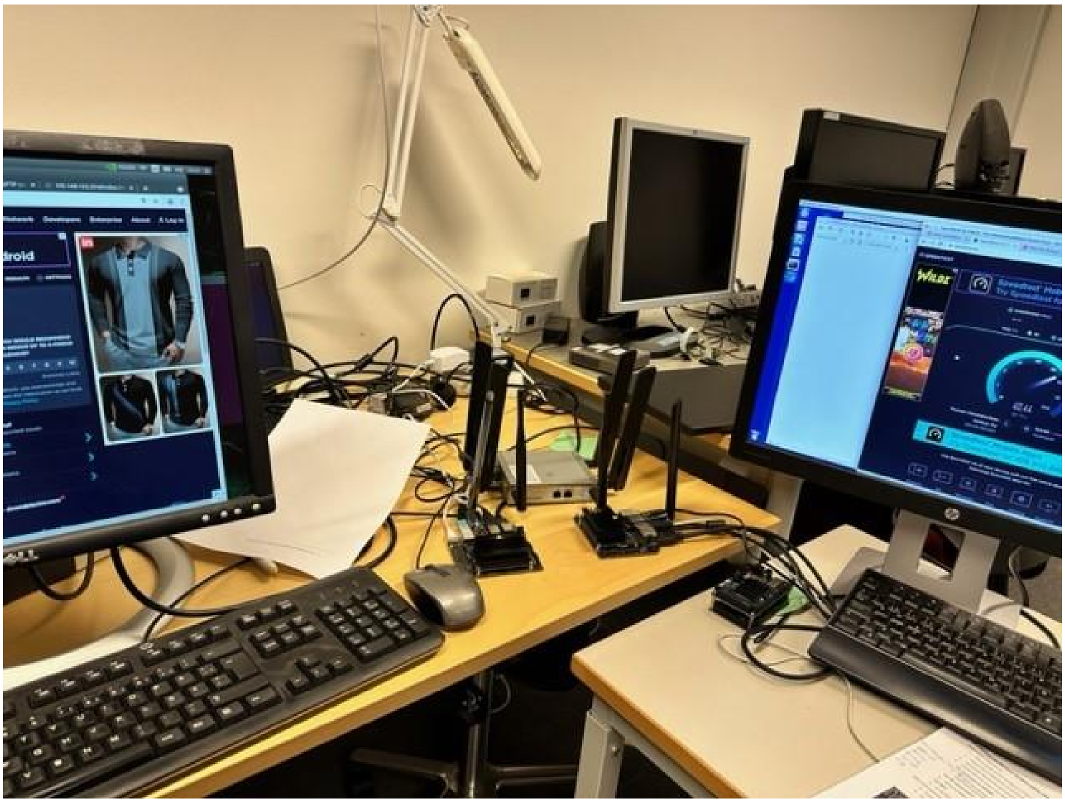

Figure 23. Test bed during Test ID 4.

Figure 24. Test bed during Test ID 5.

Figure 25. Test bed during Test ID 6.

Figure 26. Test bed during Test ID 7.

Figure 27. Test bed during Test ID 8.

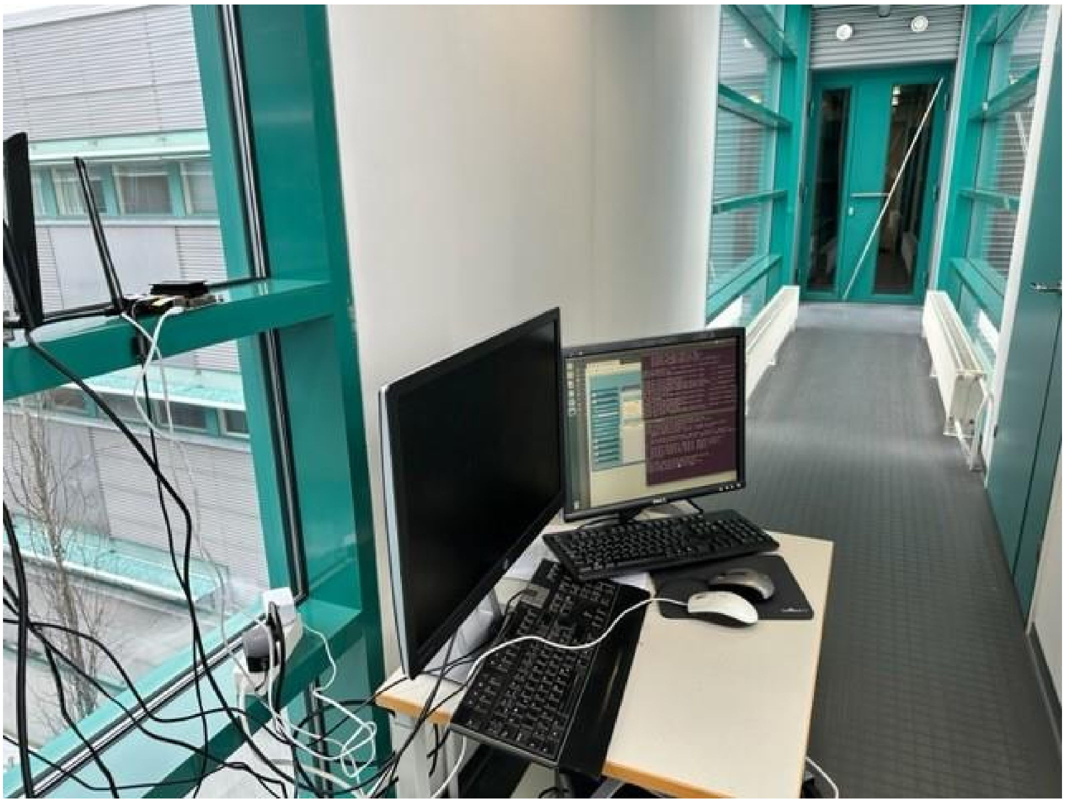

Figure 28. Test bed during Test ID 9.

For the power consumption measurements, three different cases were considered. First, the reference case in which the test nodes were not connected to any radio transceiver (i.e., neither the 5G modem nor the WiFi dongle). Note that all the other peripherals (i.e., the mouse, keyboard, and screen) were connected to the node. During the measurements, the power logger was first started, then the power was applied to the node for it to boot up, and then no commands have been given to the node until second 549 from the start of the experiment when the command for powering down the node was issued using the mouse and the GUI of the node. Further, the power was disconnected from the node and the logger was stopped. This offers a reference, allowing us to estimate the background consumption of all the components of the node except for the radio transceivers.

Secondly, the power consumption of a node connected to the 5G modem in an indoor environment. The measurements were carried out by enabling the logger and appending the power supply to the node. Two minutes after starting the node, a command was given for it to connect to the 5GTN. The connection status, signal strength, and connected cell ID were then checked, and a ping test (100 messages with 16 bytes payload, interval one second) and a speed test were launched. The run tests are shown on the chart as frames with the start and end times relative to the start of the experiment indicated above. After 26 seconds of completing the speed test, a command was issued for the node to power down. Finally, the power supply of the node was removed, and the logger was stopped.

The power consumption of a node connected to an IEEE 802.11 modem (no cellular (5G) modem was connected to the node during these measurements) and the power consumption of the AP with one single node connected to it was the other two test cases. The testing procedures for these two cases are the same as those described in the above paragraph. Figures 29 – 31 show the test beds during the power consumption measurement tests.

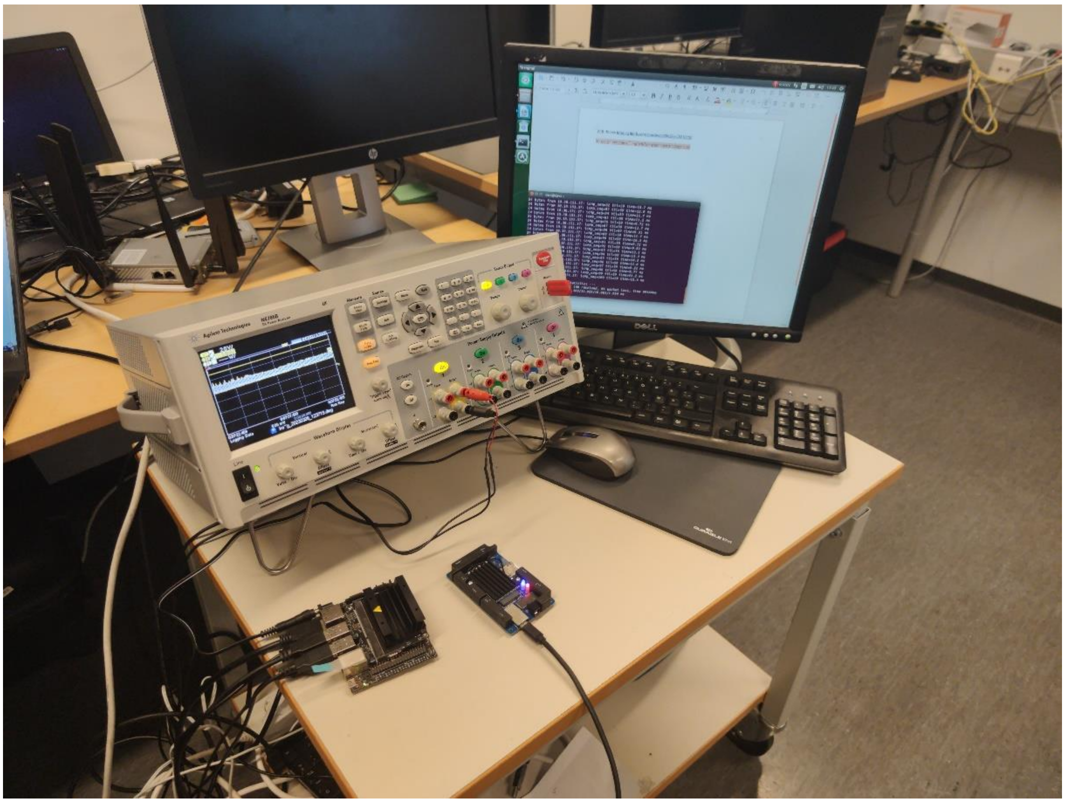

Figure 29. Test bed for the power consumption measurement of the node with 5G modem.

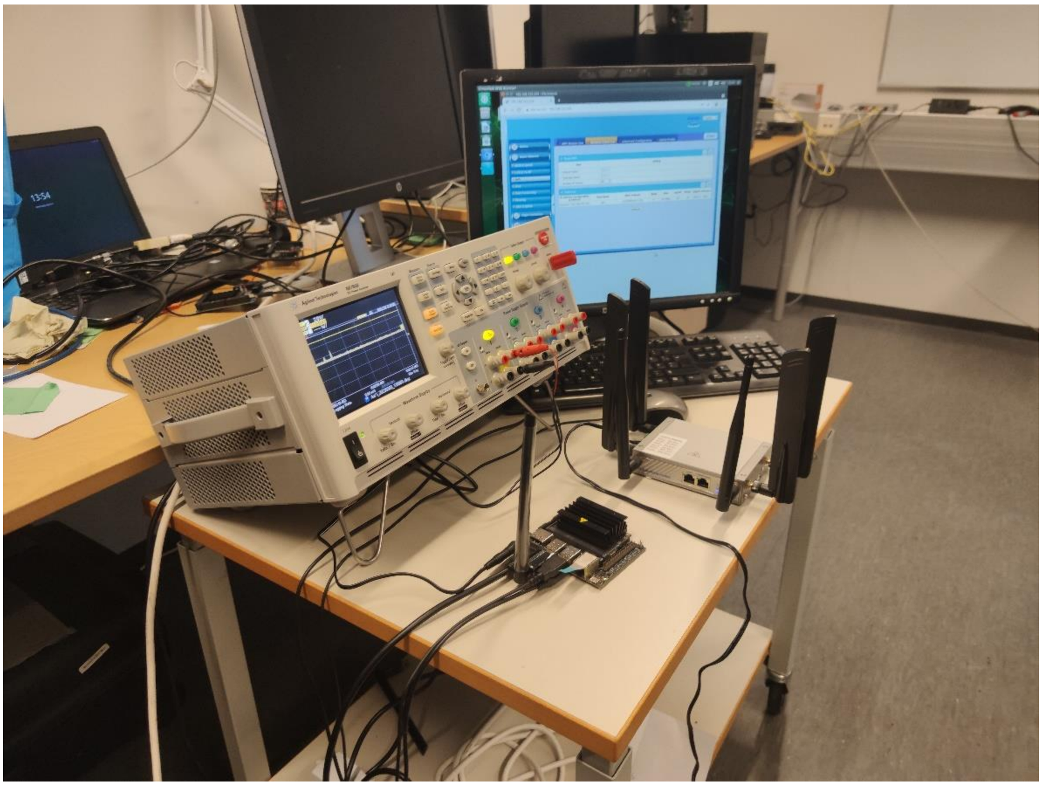

Figure 30. Test bed for the power consumption measurement for the node with an IEEE 802.11 modem.

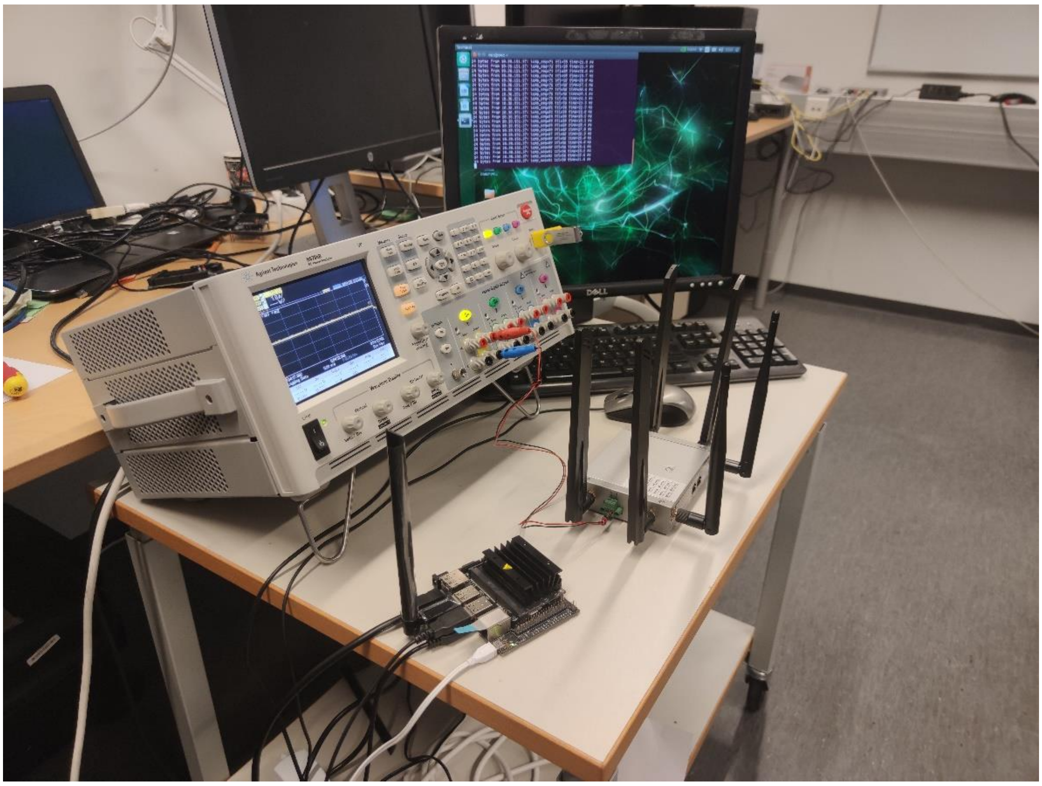

Figure 31. Test bed for the power consumption measurement for the AP with connected test node.

6 RESULTS

Each of the three following subsections focuses on one single KPI result obtained from the measurements – the throughput, the delay, and the power consumption. Meanwhile, Table 4 summarizes the radio channel-related metrics for each of the tests done. For cellular (5G) communication the table includes the data for SINR, RSRQ, RSRP, and RSSI. For the three former metrics, the results are provided for four antennas: primary receive (RX) (PRX), diversity RX (DRX), MIMO RX2, and MIMO RX. Also, a single RSSI value is reported. For the cases when communication was done using IEEE 802.11, the RSSI values for two antennas are listed. Note that these radio channel-related metrics were measured before each of the tests.

Table 4. Radio channel-related metrics measured during the tests

Test ID	SINR (dB)				RSRQ (dB)				RSRP (dBm)				RSSI (dBm)	
	PRX	DRX	RX2	RX3	PRX	DRX	RX2	RX3	PRX	DRX	RX2	RX3	RSSI0	RSSI1
1.a	15	0	2	-1	-10	-13	-12	-16	-98	-100	-98	-117	-71	Not supported
1.b	15	1	3	2	-10	-14	-12	-12	-98	-101	-100	-108	-71	
1.c	15	-7	5	8	-10	-13	-11	-11	-98	-102	-95	-105	-71	
2.a	-1	-3	-2	3	-17	-14	-16	-11	-112	-101	-108	-99	-75	
2.b	1	2	2	9	-14	-13	-12	-11	-108	-100	-100	-96	-75	
2.c	8	0	-3	12	-10	-14	-13	-11	-99	-98	-99	-100	-69	
3.a	Not supported for this topology												-15	-28
3.b													-16	-28
3.c													-15	-28
4.a													-15	-28
4.b													-16	-28
4.c													-23	-18
5.a													-16	-28
5.b													-13	-27
5.c													-13	-32
6.a													-12	-34
6.b													-23	-18
6.c													-23	-18
7.a	30	26	-20	-20	-10	-10	-10	-10	-66	-68	-68	-73	-51	Not supported
7.b	30	25	24	30	-10	-10	-10	-10	-66	-68	-69	-75	-51	
7.c	30	26	25	30	-10	-10	-10	-10	-66	-67	-68	-75	-51	
8.a	31	26	25	30	-10	-10	-10	-10	-67	-68	-68	-75	-51	
8.b	26	26	29	32	-10	-10	-10	-10	-76	-66	-63	-68	-51	
8.c	26	26	29	32	-10	-10	-10	-10	-75	-65	-63	-68	-51	
9.a	Not supported for this topology												-19	-22
9.b													-19	-21
9.c													-19	-22
10.a													-23	-27
10.b													-23	-27
10c													-23	-27

6.1 Throughput measurement results

Table 5. Summarized results showing average throughput for different test IDs

Test ID	Average Speed Test Result (Mbps)		Average Throughput Result from SFTP client (Mbps)		Average Qosium Scope Results (Mbps)	
	Uplink	Downlink	Uplink	Downlink	Uplink	Downlink
1.c	13,2	169,67	11,20	40,00	11,78	42,39
2.c	6,85	149,29	2,40	2,90	2.75	3.76
3.c	26,29	48,31	26,40	21,60	26,83	22,24
4.c	27,52	47,46	32,90	49,80	33,32	53,92
5.c	25,63	47,23	170,40	126,00	169	131,37
6.c	30,00	47,00	52,24	62,14	52,77	61,92
7.c	46,05	185,83	42,40	75,20	46,81	82,4
8.c	31,86	147,29	14,40	18,16	14,59	16,96
9.c	36,36	44,42	36,80	18,00	38,00	18,67
10.c	36,61	41,24	80,00	55,20	81,34	55,38

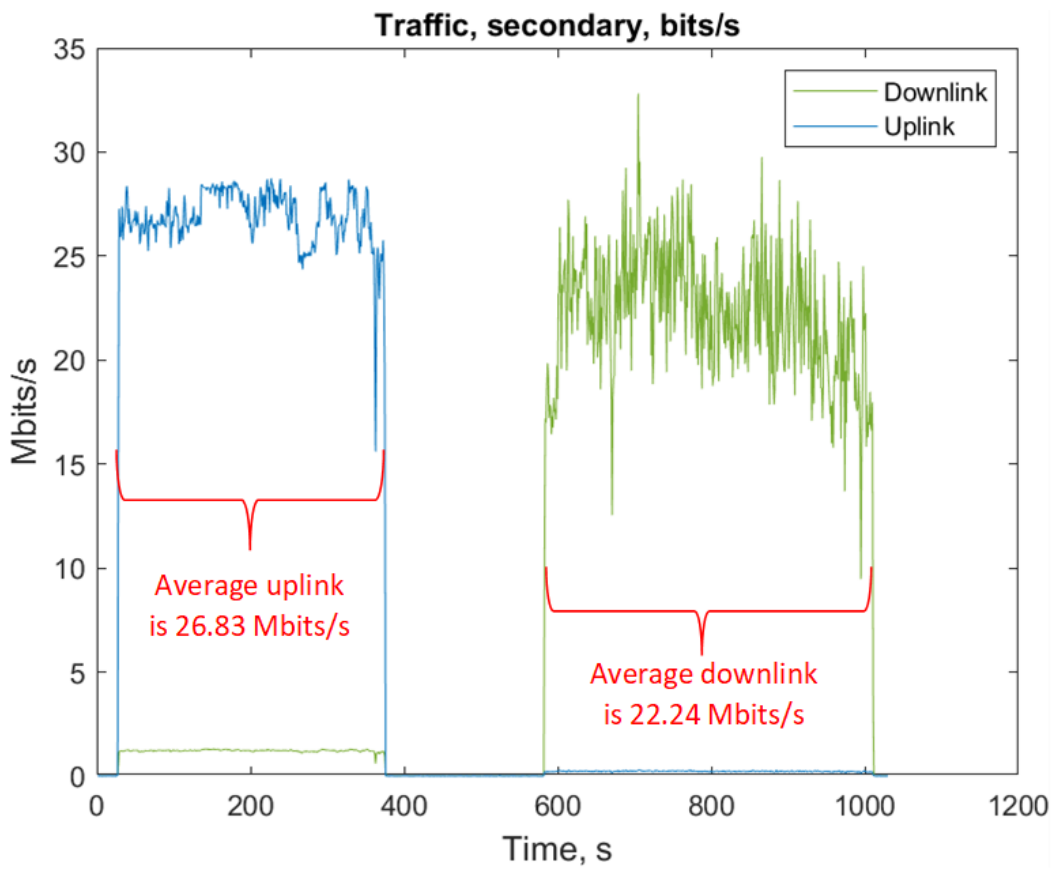

Figure 32. Illustration of the throughput test traffic in Mbit/s from secondary Qosium probe (i.e., the FRACTAL node) for Test ID 3. The file is uploaded to a remote SFTP server (high uplink traffic, small downlink traffic acknowledging reception) during the first phase, and then the file is downloaded from the SFTP server (high downlink traffic, small uplink traffic acknowledging reception) in the second phase. Here ‘uplink’ is referred to the received traffic

from the primary probe and the 'downlink' is referred to the sent traffic from the secondary probe.

Table 5 summarizes the average uplink and downlink data rate obtained through three different methods: the speed test (measuring Internet access speed to the closest server of a specialized subset [46]), the SFTP measurement (discussed in section 5.4), and analyzing the measurement results of Qosium Scope for each test ID. For calculating the average throughput from Qosium results, these have been imported in MATLAB and further processed. The average throughput has been separately calculated over periods of uplink and downlink traffic, as illustrated in Figure 32. Plots of throughput obtained from Qosium for selected illustrative test cases, showing how these metrics have changed throughput during the experiment are presented in figures 33 – 41. It's important to note that in the context of primary traffic figures in KaitoTec Qosium Scope, "received" refers to incoming traffic to the primary probe, which corresponds to traffic sent from the secondary probe. On the other hand, "sent" refers to outgoing traffic from the primary probe, which corresponds to the traffic received by the secondary probe. Similarly, in the context of secondary traffic figures, "received" refers to incoming traffic to the secondary probe, which corresponds to traffic sent from the primary probe. And "sent" refers to outgoing traffic from the secondary probe, which corresponds to the traffic received by the primary probe. These distinctions are significant in understanding the direction of traffic flow between the primary and secondary probes in Qosium Scope and can provide valuable insights into the performance and behaviour of the network traffic being measured.

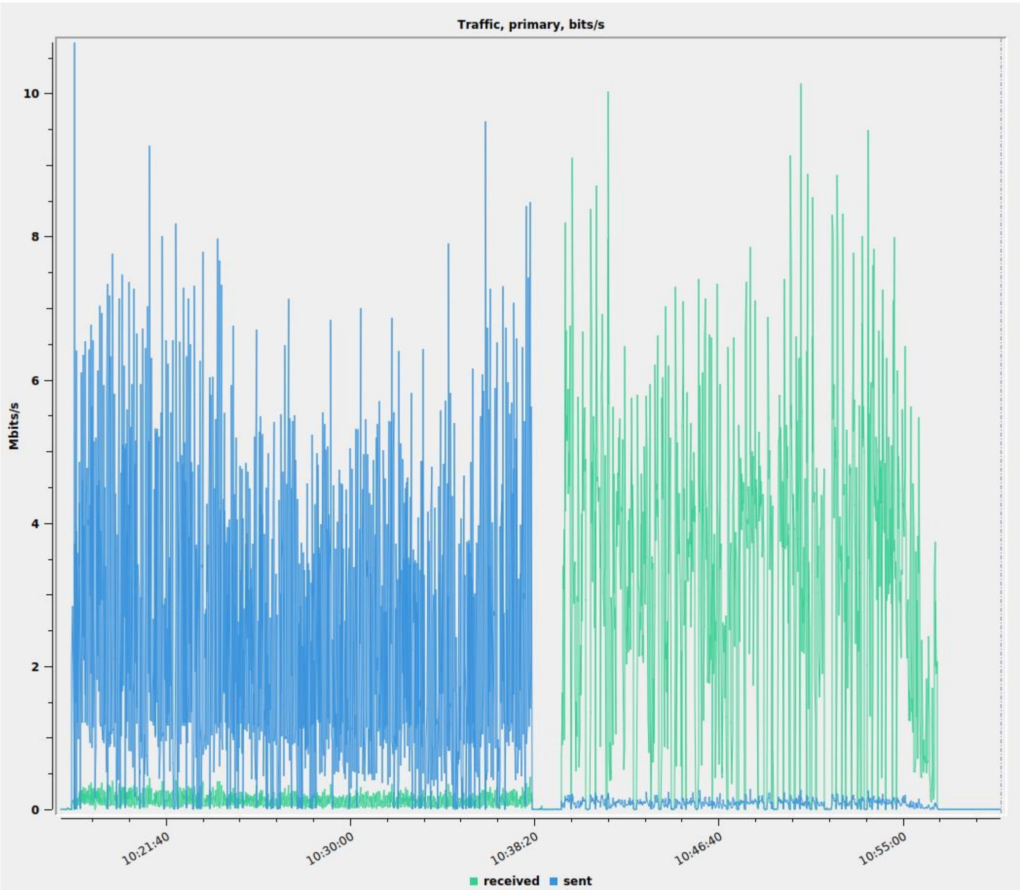

Figure 33. Illustration of the throughput measured by Qosium Scope during Test ID 2.c. One test node is the primary probe while the second test node is the secondary probe.

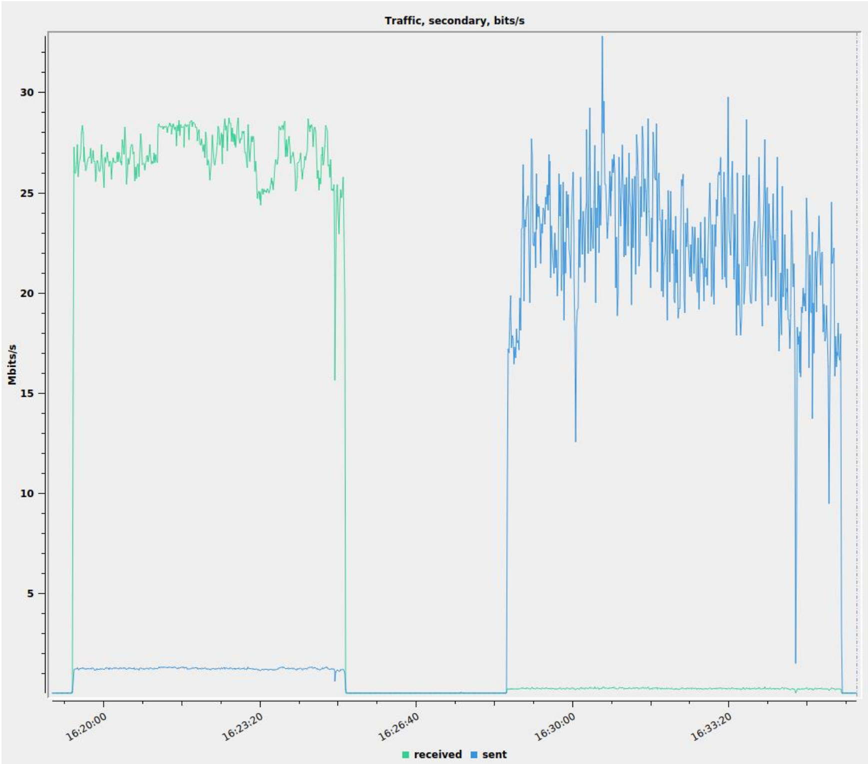

Figure 34. Illustration of the throughput measured by Qosium Scope for Test ID 3.c. The primary probe is the test saver while the secondary probe is the test node.

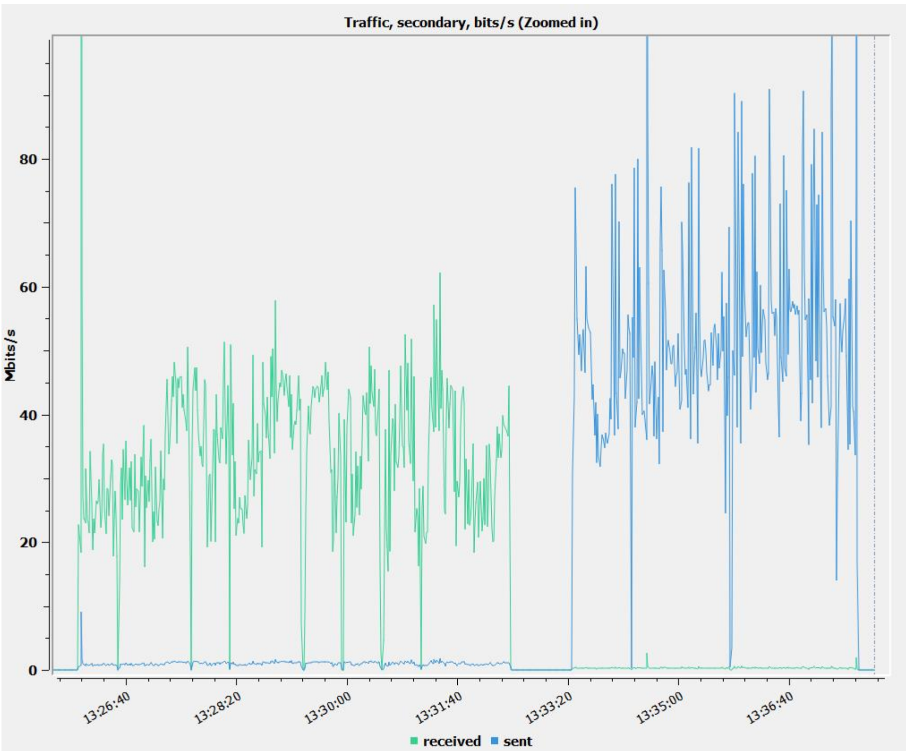

Figure 35. Illustration of the throughput measured by Qosium Scope for Test ID 4.c. One test node is the primary probe while the second test node is the secondary probe.

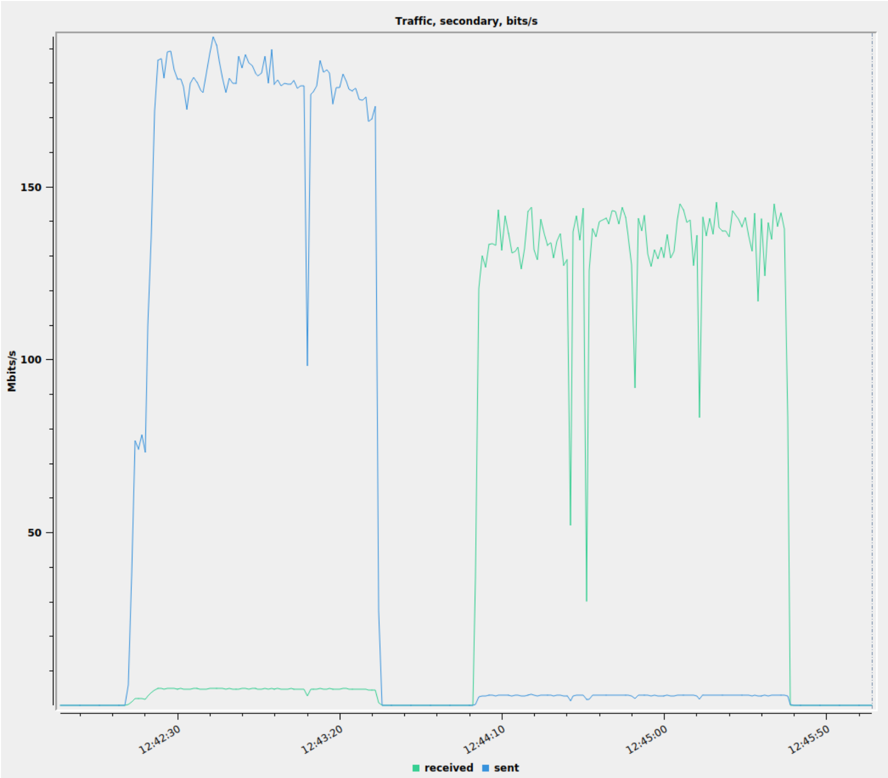

Figure 36. Illustration of the throughput measured by Qosium Scope for Test ID 5.c. The primary probe is the test node while the secondary probe is the test laptop.

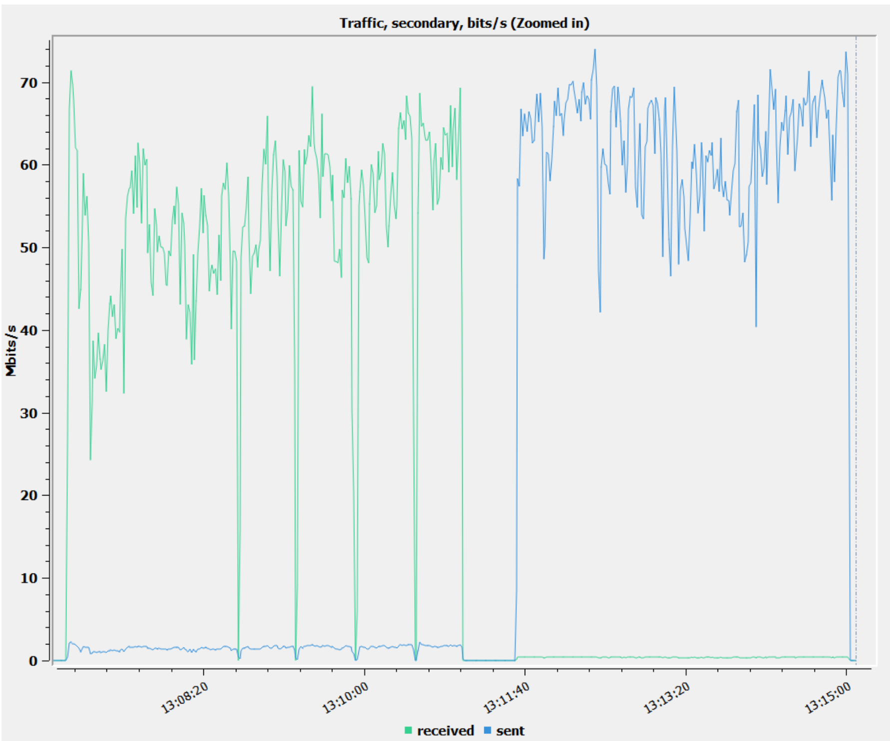

Figure 37. Illustration of the throughput measured by Qosium Scope for Test ID 6.c. One test node is the primary probe while the second test node is the secondary probe.

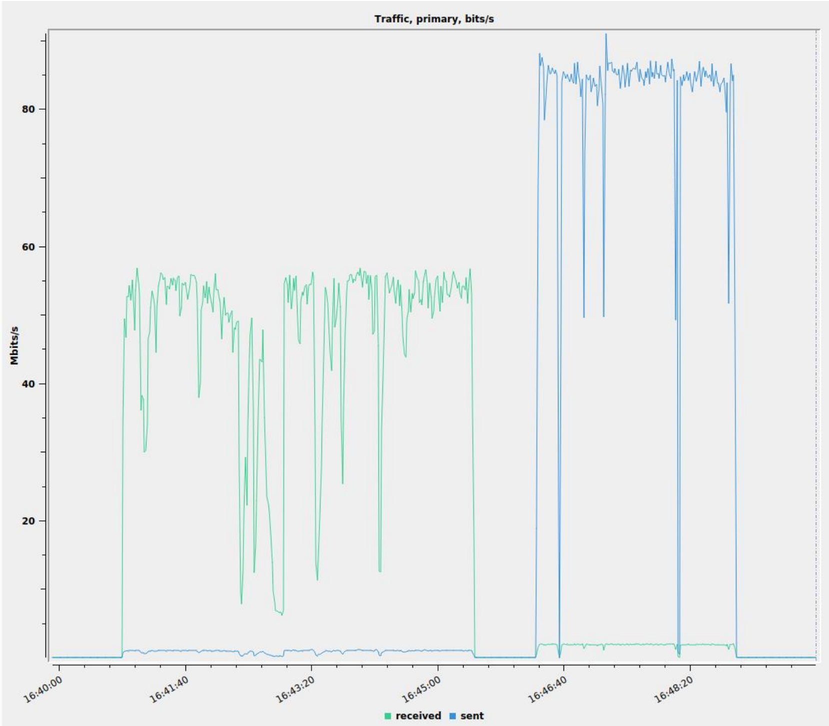

Figure 38. Illustration of the throughput measured by Qosium Scope for Test ID 7.c. The primary probe is the test saver while the secondary probe is the test node.

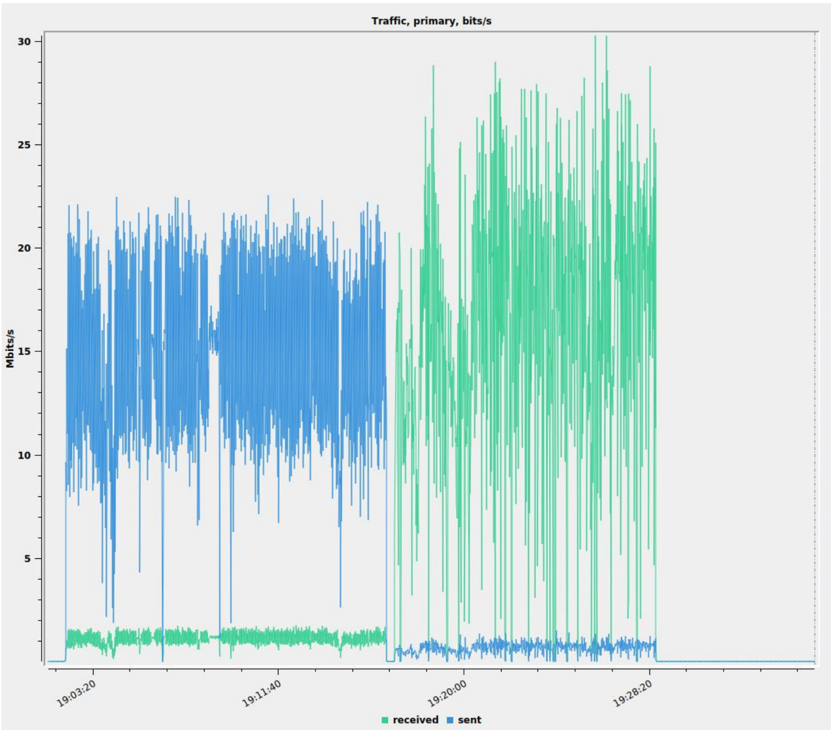

Figure 39. Illustration of the throughput measured by Qosium Scope for Test ID 8.c. One test node is the primary probe while the second test node is the secondary probe.

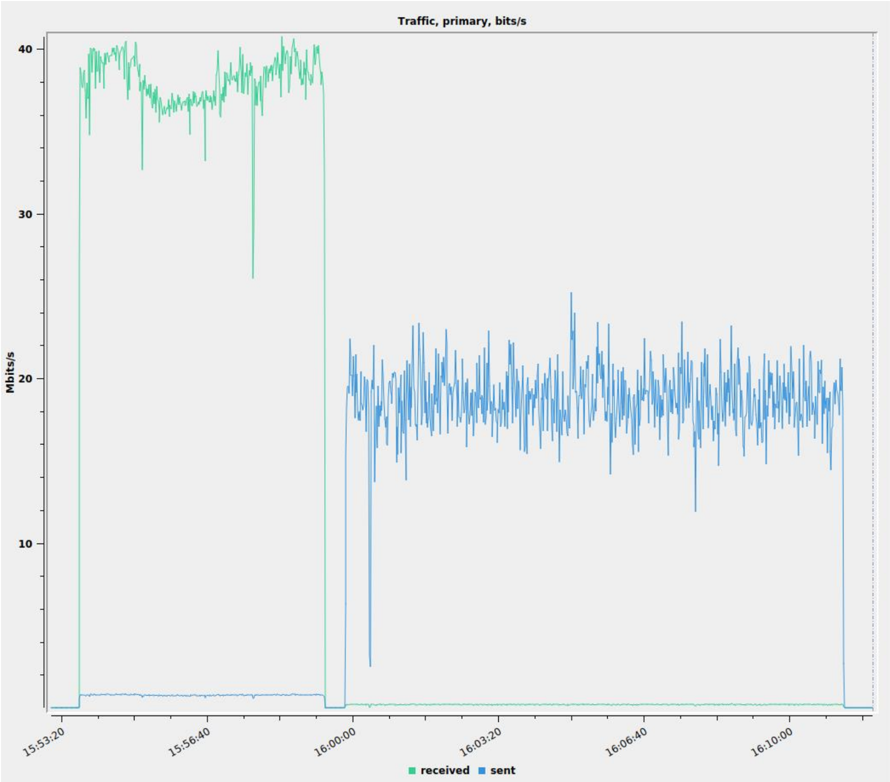

Figure 40. Illustration of the throughput measured by Qosium Scope for Test ID 9.c. The primary probe is the test saver while the secondary probe is the test node.

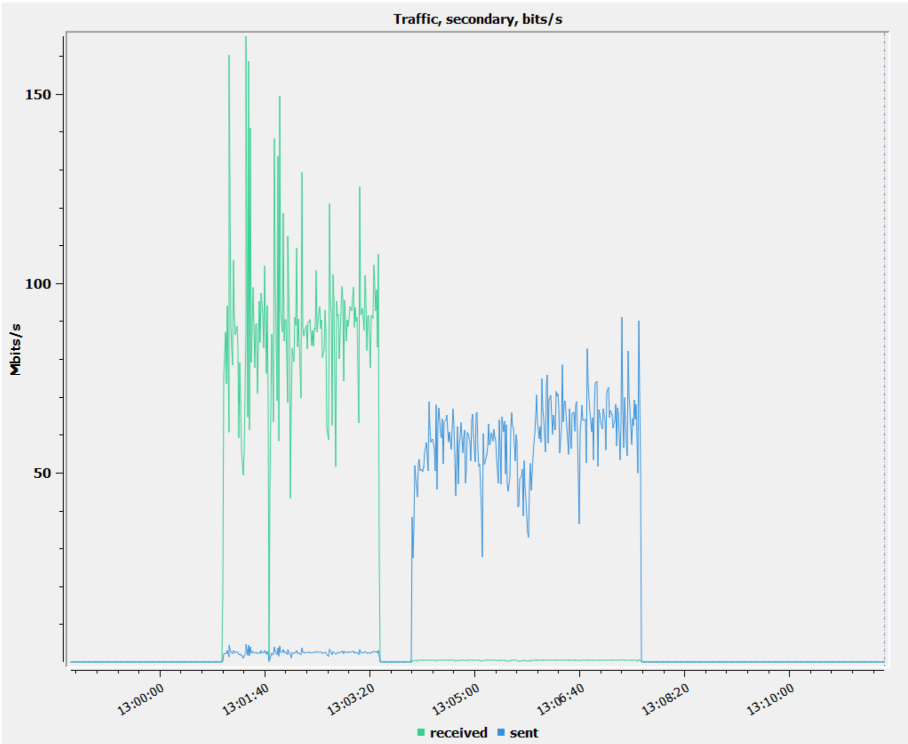

Figure 41. Illustration of the throughput measured by Qosium Scope for Test ID 10.c. One test node is the primary probe while the second test node is the secondary probe.

6.2 Latency measurement results

Table 6 offers a summary of the results for latency measurements, combining the results obtained from ping commands showing the round-trip-time (RTT) with the delay measured by the Qosium Scope tool in uplink (received) and downlink (sent) directions. Note, the results are presented for two different message payload sizes.

Table 6. RTT and average delay

Test ID	Size of a message[2] (bytes)	RTT (ms) reported by Ping				Qosium Scope delay measurement (ms)	
		Minimum	Average	Maximum	Mean Deviation	Average Received Delay	Average Sent Delay
1.a	24	7,336	11,897	30,798	2,331	6,96	6,19
1.b	908	11,189	17,309	122,911	5,608	7,43	6,99
2.a	24	14,692	24,202	1598,861	55,737	16,24	13,63
2.b	908	22,73	48,142	2089,212	130,567	18,58	14,73
3.a	24	13,361	24,958	44,06	4,469	Data unavailable[3]	
3.b	908	20,484	34,219	63,844	5,373		
4.a	24	1,38	1,936	15,48	0,681	1,04	0,94
4.b	908	1,693	2,263	10,479	0,642	0,91	1,06
5.a	24	1,293	1,632	3,142	0,273	2,89	0,91
5.b	908	1,412	2,14	7,695	0,705	3,33	1,03
6.a	24	1,489	2,007	6,447	0,608	0,87	1,02
6.b	908	1,757	2,409	25,759	1,216	1,03	1,12
7.a	24	7,636	11,856	27,22	2,371	6,83	5,78
7.b	908	10,928	16,528	31,741	2,728	7,55	5,77
8.a	24	14,461	20,429	36,084	3,186	11,96	11,77
8.b	908	20,169	29,911	44,859	3,364	12,88	12,77
9.a	24	13,806	24,092	72,038	4,046	Data unavailable[3]	
9.b	908	20,792	31,252	45,046	3,704		
10.a	24	1,502	2,418	13,909	1,272	1,26	1,51
10.b	908	1,655	2,833	11,164	1,201	1,34	1,54

Figures 39 - 46 present the plots of delay obtained from Qosium for selected illustrative test cases, showing how the delay changed in time throughout the experiment. Note that in the following figures 'received' refers to the delay experienced by the network traffic when it is received at the primary probe while 'sent' refers to the delay experienced by the network traffic when it is sent from the secondary probe.

[2] Including headers; the ping size argument was set to either to 10 or to 900 bytes.
[3] Despite multiple attempts, the Qosium scope was unable to measure the delay, which remains unexplained and requires further investigation, beyond the scope of the thesis.

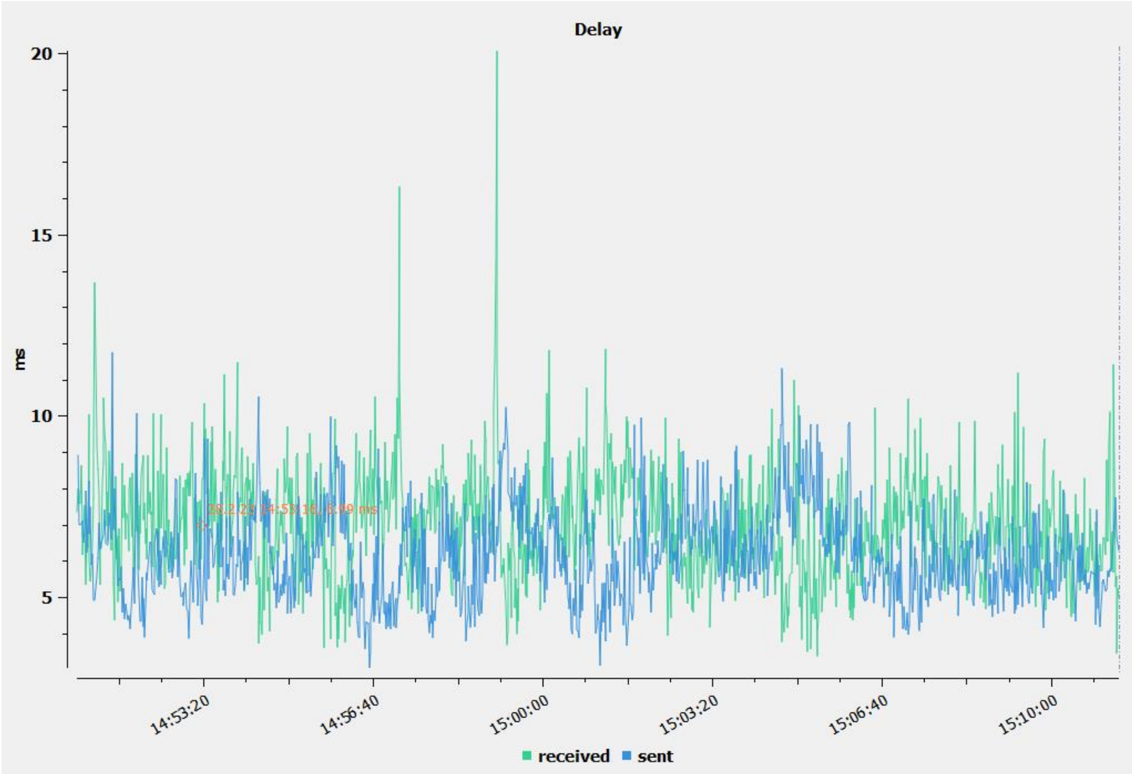

Figure 42. Illustration of the delay measured by Qosium Scope for Test ID 1.a. The primary probe is the test saver while the secondary probe is the test node.

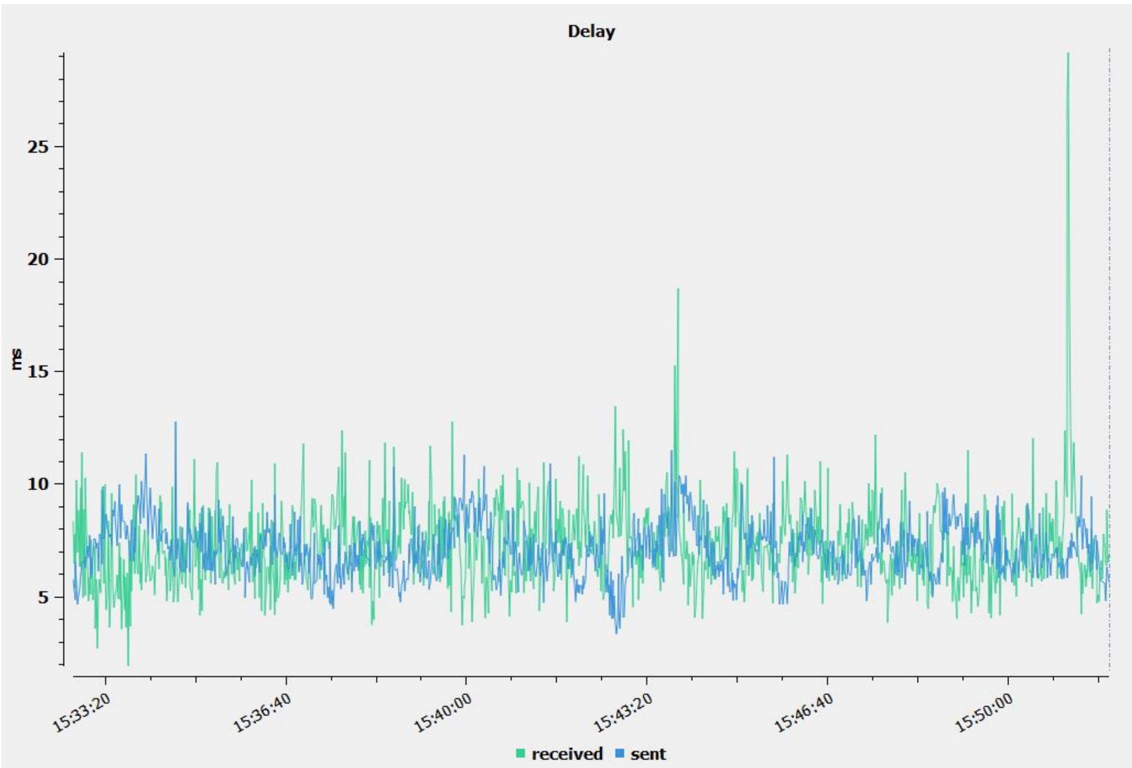

Figure 43. Illustration of the delay measured by Qosium Scope for Test ID 1.b. The primary probe is the test saver while the secondary probe is the test node.

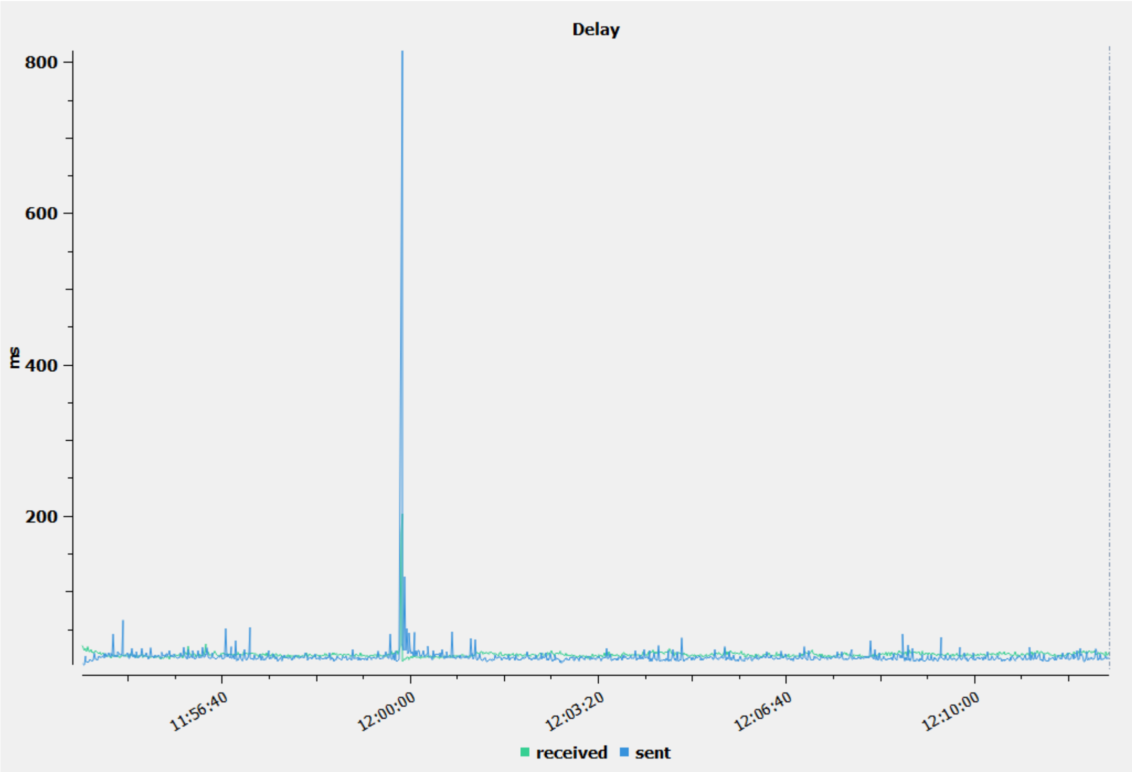

Figure 44. Illustration of the delay measured by Qosium Scope for Test ID 2.a. One test node is the primary probe while the second test node is the secondary probe.

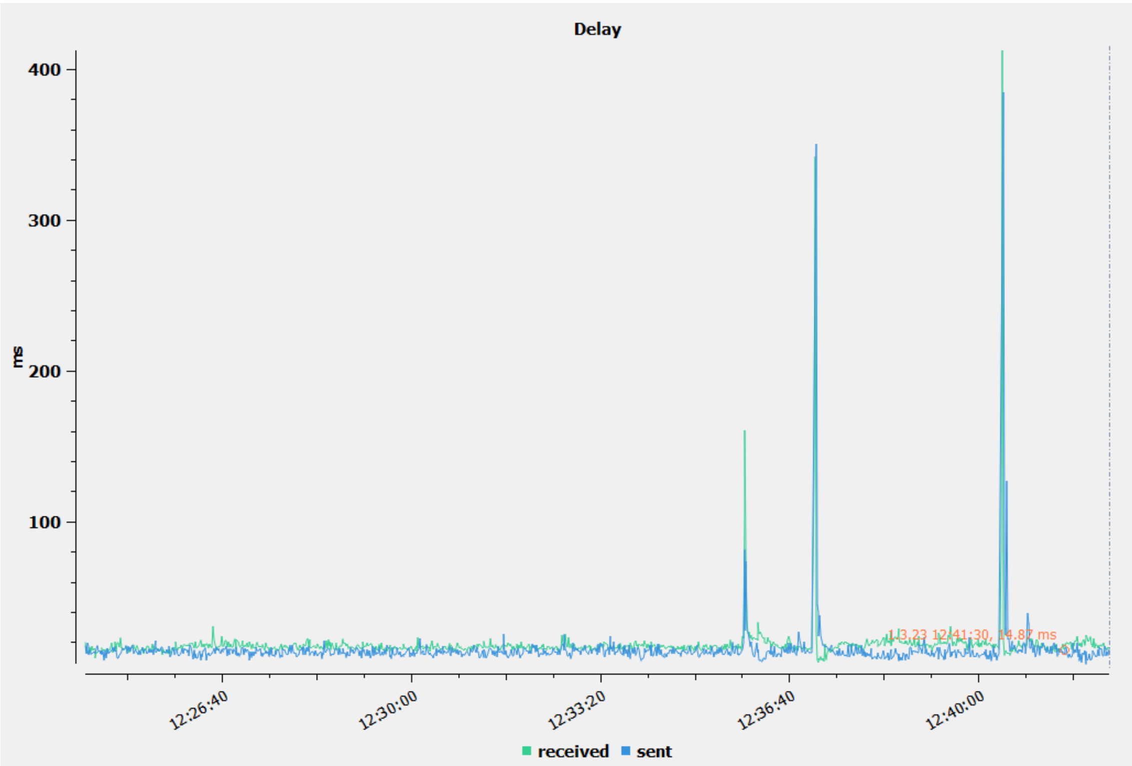

Figure 45. Illustration of the delay measured by Qosium Scope for Test ID 2.b. One test node is the primary probe while the second test node is the secondary probe.

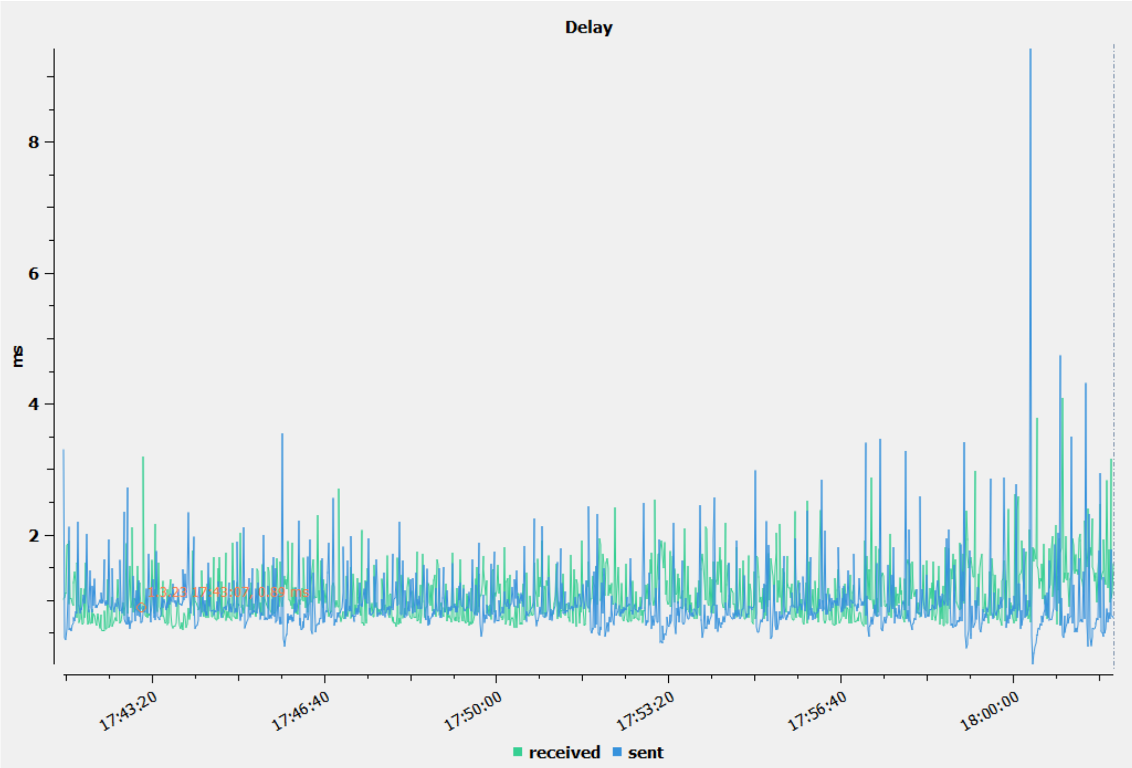

Figure 46. Illustration of the delay measured by Qosium Scope for Test ID 4.a. One test node is the primary probe while the second test node is the secondary probe.

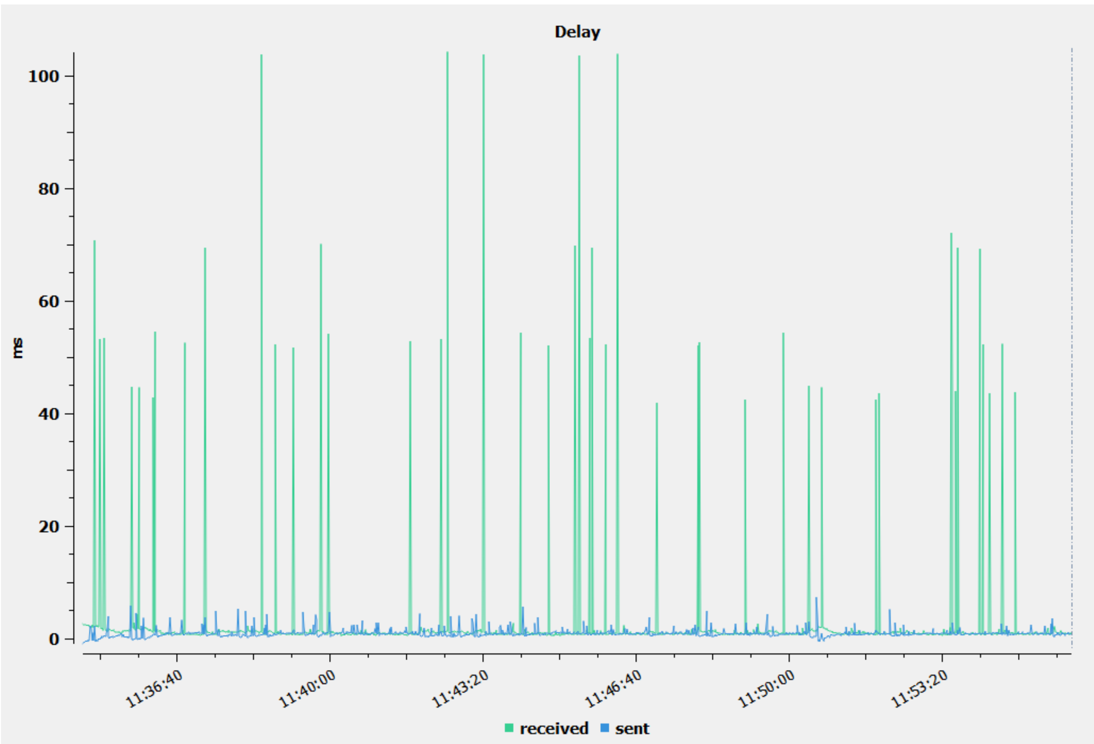

Figure 47. Illustration of the delay measured by Qosium Scope for Test ID 5.a. The primary probe is the test node while the secondary probe is the test laptop.

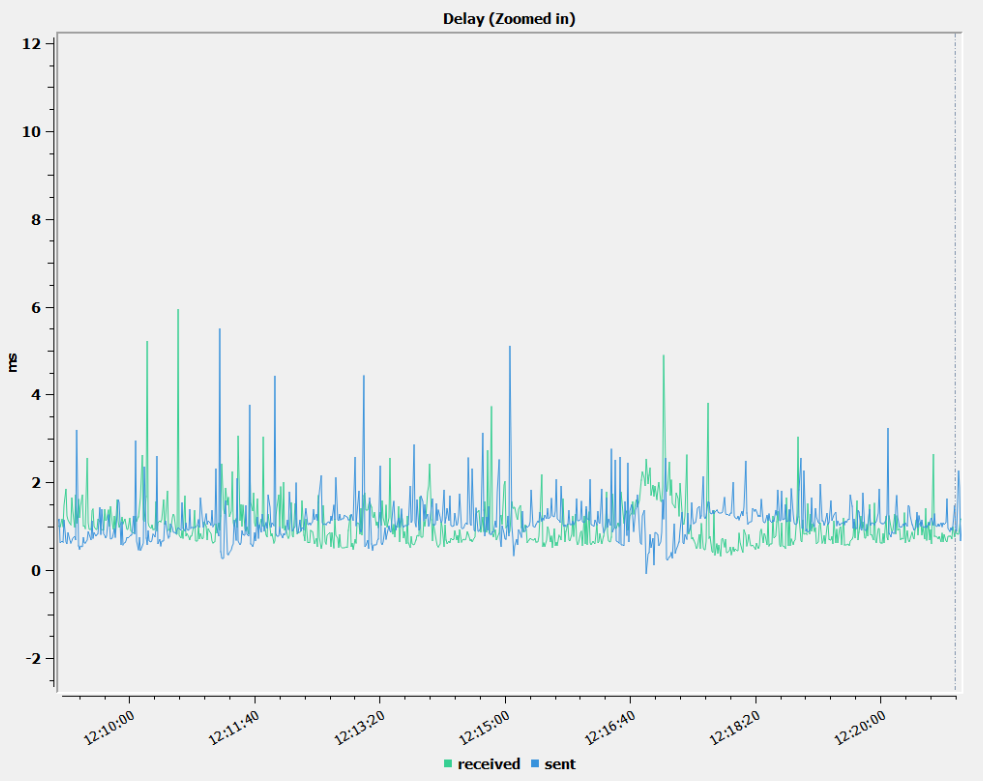

Figure 48. Illustration of the delay measured by Qosium Scope for Test ID 6.b. One test node is the primary probe while the second test node is the secondary probe.

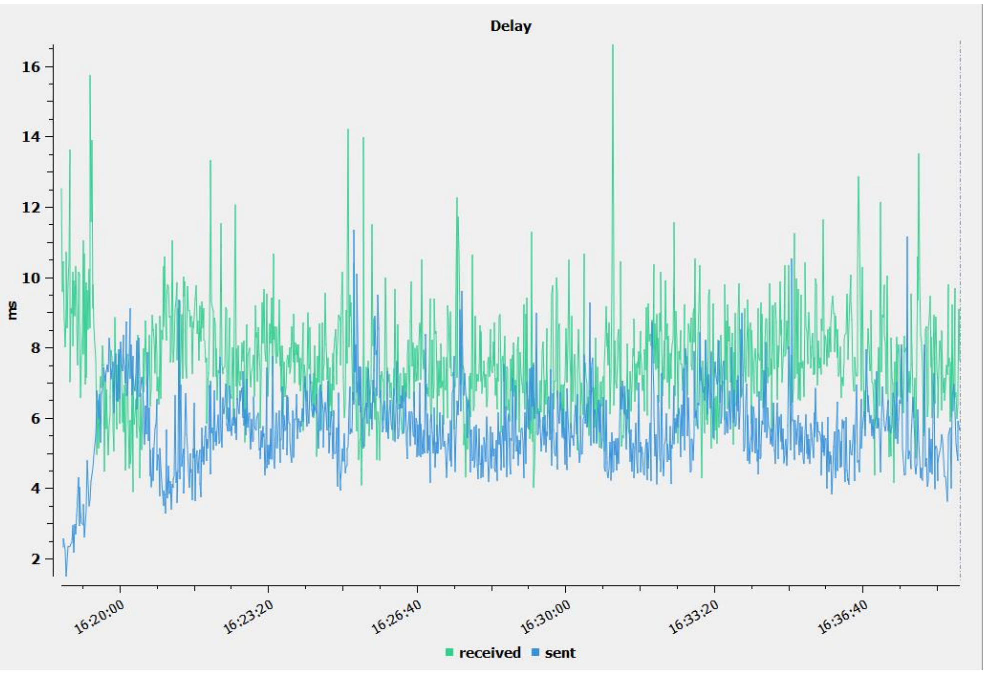

Figure 49. Illustration of the delay measured by Qosium Scope for Test ID 7.b. The primary probe is the test saver while the secondary probe is the test node.

6.3 Power consumption results

Finally, Table 7 provides some insight into the power consumption of the node for communication. Moreover, Figures 47 - 50 illustrate the current consumption profile of the test node for different topologies measured by the DC power analyzer with the different phases of the experiment marked. Note that the different phases marked in the below figures were identified from the time noted down during the experiments.

Table 7. - Average power consumption

Average power consumption in Watts									
Reference	**Direct 5G connection**			**WiFi connection**			**AP**		
	overall	**Ping test**	**Speed test**	**overall**	**Ping test**	**Speed test**	**overall**	**Ping test**	**Speed test**
2,3895	5,141	4,5536	9,3772	3,3897	2,9296	5,5021	3,8038	3,6845	5,1333

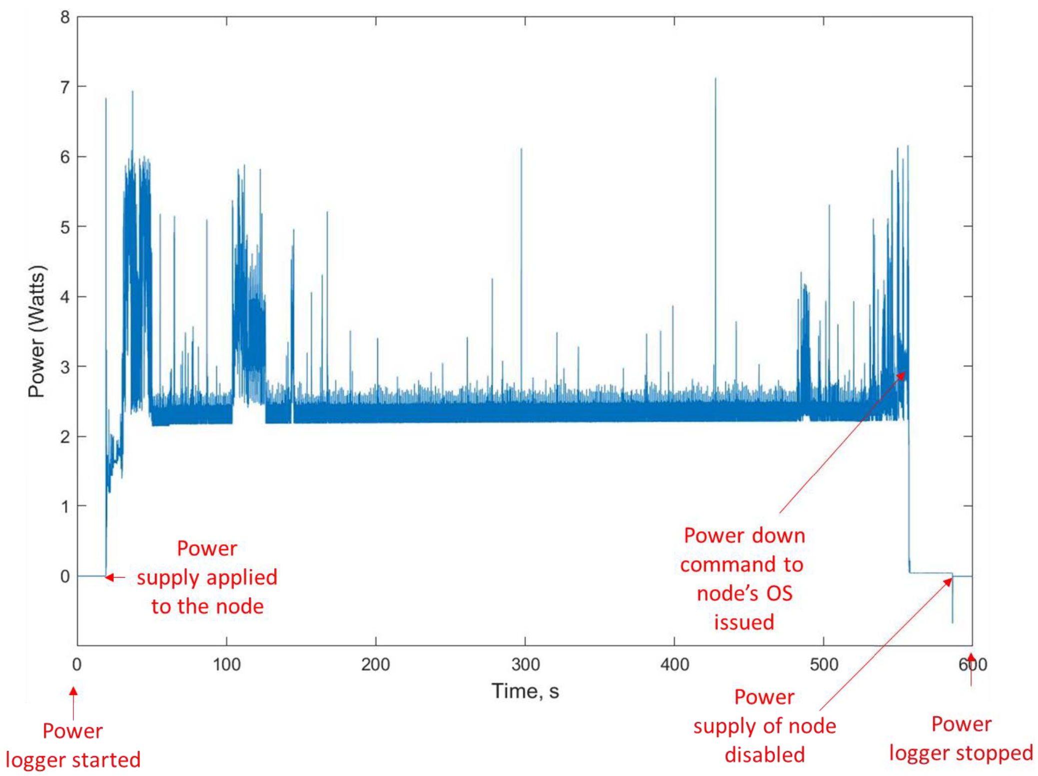

Figure 50. Illustration of the power consumption for the reference case.

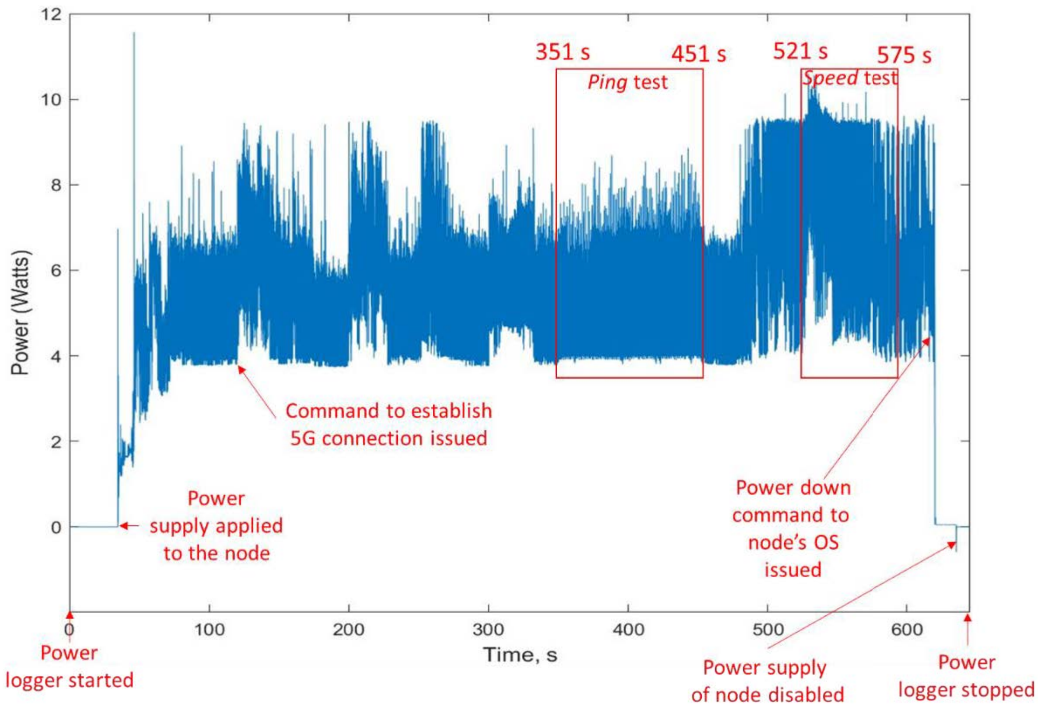

Figure 51. Illustration of the power consumption for the node with 5G modem.

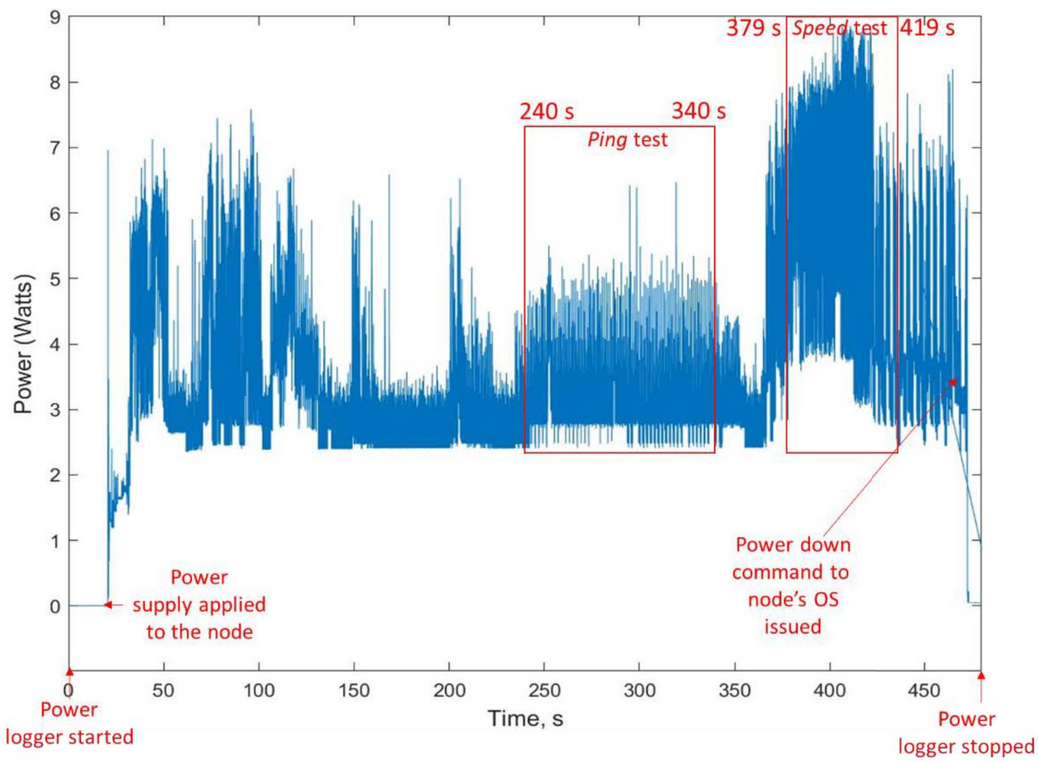

Figure 52. Illustration of the power consumption for the node with an IEEE 802.11 modem.

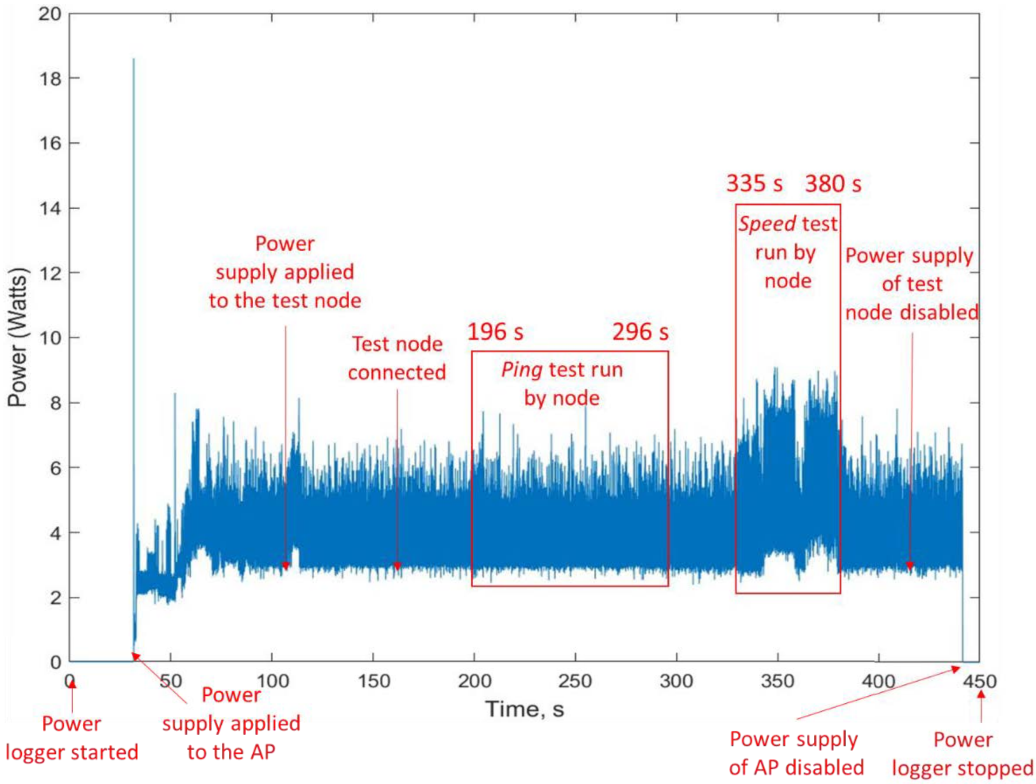

Figure 53. Illustration of the power consumption for the AP with connected test node.

7 DISCUSSION

This chapter discusses in detail the KPIs results presented in section 6. Subsection 7.1 discuss the throughput performance and subsection 7.2 discuss the performance for latency. Subsection 7.3 is dedicated to a discussion on the performance of power consumption.

The results presented in Table 4 indicate that for IEEE 802.11, the signal level during testing was between -12 to -23 dBm, which implies a highly favourable channel that can support the fastest modulation-coding schemes which is beneficial for achieving high data rates and low latency. On the other hand, for cellular (5G), the signal level ranged from -100 dBm for indoor tests to -66 to -75 dBm for LoS tests. This suggests excellent channel conditions for LoS tests and mid-cell-to-cell edge conditions for indoor tests [47]. The appropriate signal level will depend on a range of factors, such as the distance between the wireless devices, the level of interference from other devices, and the specific modulation and coding schemes that are being used. In some cases, a lower signal level may be acceptable if the network can still provide reliable connectivity and sufficient data rates for the intended applications. Additionally, the radio conditions remained relatively stable and consistent within each group of tests. FRACTAL aims to improve the performance of wireless networks by dynamically adjusting their configurations based on changing network conditions. Therefore, the quality of the channel is an important factor in achieving this objective.

7.1 Throughput performance

Table 5 shows that the average Qosium scope results closely align with the results obtained from SFTP client software, although the results from speed tests may differ substantially. In some tests (such as IDs 1.c and 2.c), the speed test results were notably higher than the results of other measurement methods. However, in other tests (such as 4.c and 6.c), the throughput through SFTP transfer was higher than the speed test. The primary reason for these differences lies in the different routing and protocols used on top of IP. During the speed test, all traffic is sent to the nearest internet gateway and from there to the test server. In contrast, in cases involving IEEE 802.11 protocol and node-to-node communication (like test ID 6.c), data transfer through a local network may occur during SFTP tests. The speed test results can be influenced by the current load on the network. If there are many users or devices connected to the network at the time of the speed test, it can result in increased network congestion and reduced throughput. This can lead to varying speed test results even if the signal quality is high. Another reason could be the performance of the speed test server. Speed tests rely on a server to measure the upload and download speeds, and if the speed test server is experiencing high load or congestion, it also can result in inconsistent speed test results.

Based on the presented results, it can be observed that for exclusive 5G communication, the maximum average throughput achieved during speed tests was 185 Mbps in the downlink and 46 Mbps in the uplink. Similarly, during SFTP tests, the throughput was around 80 Mbps in the downlink and 46 Mbps in the uplink. The discrepancy in the results can be attributed to (i) the use of different protocols, and (ii) different routing paths. Different protocols have varying overheads and requirements, which can affect the available bandwidth for data transfer. For example, some protocols may introduce more delay or require more processing power, which can reduce the amount of data that can be transferred within a given time. Additionally, different routing paths can lead to varying latencies and packet losses, which can further impact the throughput. In an FNEC system, where multiple nodes are interconnected and collaborate to process data, the impact of different protocols and routing paths can be even more pronounced. The choice of protocols and routing paths can affect not only the performance of individual nodes but also the overall performance of the system. The difference in the uplink and downlink performance is also expected since the 5GTN is configured to allocate the available spectrum unequally between the uplink and downlink, with more resources available for the downlink.

The results of test ID 2.c reveal that the throughput of communication between two nodes located under the same 5G base station is significantly lower compared to the throughput observed when a node is communicating with an upper-tier node, such as an edge server. This is because 5G communication does not support the sidelink, and therefore all the traffic has to go through a base station in case of two 5G devices communicate with each other. Additionally, if two devices are situated close to each other, they may have to share the same beam, which can further reduce the maximum throughput. It is also important to note that the results of the tests may have been influenced by the operation of other users, as the cellular network operates under spectrum-resource-limited conditions.

From the results for test IDs 5 and 6 revealing the throughput for IEEE 802.11 WLAN connectivity, one can see that the uplink throughput reached 170 Mbit/s and downlink 130 Mbit/s. Notably, it can be noted that the throughput for IEEE 802.11-based communication between the two test nodes exceeded 70-80 Mbit/s in the uplink and 50 Mbit/s in the downlink. The physical layer (PHY) of IEEE 802.11 WLAN uses higher frequency bands (2.4 GHz or 5 GHz) and wider bandwidths (20 MHz or 40 MHz) [48] compared to lower frequency bands (e.g., sub-6 GHz) used in some 5G cellular networks. Higher frequency bands and wider bandwidths allow for more data to be transmitted in a given period, resulting in higher throughput. Another reason could be the MAC layer of IEEE 802.11 WLAN uses carrier sense multiple access with collision avoidance (CSMA/CA) protocol to access the channel [48]. This protocol ensures that multiple nodes do not transmit simultaneously, thereby avoiding collisions and thereby increase the throughput compared to the TDMA protocol used in 5GTN. Cellular networks, especially in urban or densely populated areas, can experience higher levels of network congestion due to the larger number of users and devices connected to the network. This can result in decreased throughput as the available bandwidth is shared among multiple users. On the other hand, WLAN-based communication may have less congestion, especially in localized environments or small-scale deployments, resulting in higher throughput.

This can also be seen that the throughput performance for test IDs 4 and 10 is decently close to the results observed for test ID 6. This is not surprising, in the presence of an AP with a 5G backbone. The 5G backbone link is not utilized when transferring the data between two devices connected to the same AP over IEEE 802.11. The 5G backbone link is typically used for routing data traffic between the AP and the core network in a cellular network infrastructure. This link can provide high bandwidth and low latency, as it is designed to handle large amounts of data traffic from multiple devices. However, in the case of local data transfer between two devices connected to the same AP, the data does not need to be routed through the 5G backbone link. Instead, the data is transferred directly between the devices over the IEEE 802.11 wireless link. Therefore, the performance of the connectivity between the two devices in this scenario is defined solely by the throughput of IEEE 802.11 wireless technology, which is determined by factors such as the wireless signal strength, the distance between the devices, the number of other devices sharing the same wireless channel, and the quality of the wireless network infrastructure. It is important to note that the utilization of the 5G backbone link may come into play when the data needs to be routed to a remote location outside the local network, or when multiple devices are connected to different APs within the same network and need to communicate with each other. In such cases, the performance of the connectivity will be impacted by both the throughput of IEEE 802.11 technology and the quality and capacity of the 5G backbone link.

However, it is important to note two things. Firstly, these results reflect averages over a sufficiently long period, and short-term results may differ. Additionally, the results are dependent on various factors, such as the location of the test nodes (e.g., in the radio access network, Internet, or in a core cellular network), traffic parameters and patterns (e.g., file size), and the protocol used. Secondly, it should be noted that the throughput performance is affected by network configuration and its parameters, such as the availability of resources and their distribution between uplink and downlink, and various logical channels for cellular networks. Additionally, the version of the IEEE 802.11 protocol in use can also impact performance. The performance is also strongly influenced by other users and external interference. Therefore, the results obtained should be considered indicative rather than conclusive. It is more important to focus on trends rather than actual values.

7.2 Latency Performance

The results for test IDs 1,2,7 and 8 reveal that for the nodes directly connected to the cellular (5G) network the average RTT was 12 and 24 ms when communicating to an upper-tier node for packets of 24 and 908 bytes, respectively. For the direct communication between same-tier nodes connected over 5G, the delay is about twice higher – 24 and 48 bytes, respectively. The higher delay observed in direct communication between same-tier nodes connected over 5G compared to the nodes directly connected to the cellular (5G) network can be attributed to the additional processing and routing required in the network. In an FNEC sub-system, the nodes are organized hierarchically, with upper-tier nodes responsible for aggregating data from lower-tier nodes and performing more complex processing tasks. When a lower-tier node communicates with an upper-tier node, may involve multiple hops. Each hop in the communication path introduces a certain amount of delay due to the time taken to transmit the data from one node to another.

The results of Qosium Scope, which show the delay for uplink and downlink communication separately, enable us to detail the results more. This can be seen that the delay is distributed not uniformly – the downlink delay is slightly (about 10%) lower than the uplink one. Comparing the results for the indoor position of the test nodes (test ID 1) and LoS (test ID 7) this can be seen that for both cases the latency in uplink remains about the same, while for downlink LoS allows getting slightly lower latency, especially for the higher payload value. However, this can also be seen that the LoS condition also substantially reduces the mean deviation. Still, especially when the communication is between the nodes of the same level the deviation of the RTT is quite high: 55 and 130 ms for test IDs 2.a and 2.b, respectively. The lower delay in downlink communication compared to uplink communication can be attributed to the way cellular networks are designed. In a cellular network, the base station (connected to the cellular network) is responsible for transmitting data to the mobile device (downlink) and receiving data from the mobile device (uplink). The base station has more powerful antennas and better transmission power, allowing for more efficient downlink communication compared to uplink communication. Additionally, the base station can perform more advanced signal processing on downlink data, leading to a further reduction in latency. When comparing the results for the indoor position of the test nodes (test ID 1) and LoS (test ID 7), the latency in uplink communication remains about the same because the signal strength and processing capabilities are relatively similar in both scenarios. However, in LoS conditions, there is less interference and obstacles for the downlink signal, allowing for a slightly lower latency compared to indoor conditions, especially for higher payload values. In cellular networks, resources such as time, frequency, and bandwidth are allocated in an asymmetric manner, with typically more resources allocated to downlink communication compared to uplink communication. This is because downlink communication typically involves broadcast or multicast transmissions from the base station to multiple user devices, while uplink communication involves individual transmissions from multiple user devices to the base station. The asymmetrical resource allocation allows for more efficient and higher data rate transmission in the downlink direction, resulting in lower delay compared to uplink communication. Moreover, The MAC procedures employed in cellular networks, including scheduling, queuing, and prioritization, can impact the delay characteristics of downlink and uplink communication. For example, the base station can use advanced scheduling algorithms to efficiently allocate resources and prioritize downlink transmission based on the QoS requirements of different user devices. In contrast, the uplink transmission may involve contention-based access, where user devices compete for resources, which can introduce additional delay due to contention and queuing delays.

The substantial reduction in mean deviation in LoS conditions can be attributed to the reduced interference and obstacles in the path of the signal. As mentioned above, FNEC subsystem nodes are organized hierarchically, with upper-tier nodes responsible for aggregating data from lower-tier nodes and performing more complex processing tasks. When lower-tier nodes communicate with same-level nodes, the data has to traverse multiple nodes in the network, which can increase the deviation of the RTT due to variable queuing and processing delays. However, when the communication is in LoS conditions, there are fewer obstacles for the signal to overcome, resulting in a more consistent and reliable signal strength. This consistency leads to a reduction in the deviation of the RTT, as there are fewer variations in the queuing and processing delays. On the other hand, when the communication is between nodes of the same level, there may be competition for resources and congestion in the network, leading to variable delays in the queuing and processing of packets. This variability can further increase the deviation of the RTT.

This can be seen that communication using short-range IEEE 802.11 technology (test cases 5 and 6) demonstrates lower latency, than communication over a cellular (5G) network. The difference is especially notable for the case when two same-tier nodes communicate between themselves. For example, comparing the results of test IDs 6 and 8, one can see that while IEEE 802.11 enables a one-way delay of around 1 ms, the cellular (5G) delay is around 12-13 ms. This is also worth noting that the mean deviation for RTT over IEEE 802.11 links is also very low – well below 1 ms. This can also be seen that the increase of payloads from 10 to 900 bytes resulted in a decently small increase in latency of less than 0.5 ms. When two same-tier nodes communicate over short-range IEEE 802.11 technology, the communication path is direct and involves fewer nodes. This results in lower latency and more consistent communication compared to the longer, more complex path involved in cellular network communication. Additionally, IEEE 802.11 technology is optimized for short-range communication, which leads to faster transmission and lower latency compared to the more general-purpose cellular network technology. Moreover, the increase in payload from 10 to 900 bytes in IEEE 802.11 communication resulted in a small increase in latency because the technology uses a frame-based communication protocol. In frame-based protocols, data is transmitted in packets with a fixed size, and the communication overhead (such as header information) is the same regardless of the packet size. This means that regardless of the actual amount of data being transmitted in the packet, the communication overhead remains constant. Therefore, the increase in payload size has a minimal effect on the latency in IEEE 802.11 communication. On the other hand, cellular (5G) networks are designed for long-range communication with a focus on providing high throughput and supporting large numbers of devices. The cellular network architecture is hierarchical, with multiple layers of nodes involved in processing data, which can increase latency.

The combination of the two technologies and the use of an AP (test IDs 3,4,9 and 10) shows somewhat contradictive trends. On the one hand, when communication is done to an upper-tier node (test IDs 3 and 9) the RTT exceeds that for the 5G only link (test IDs 1 and 7) by more than 10 ms. This is caused due to the need or using two wireless legs (i.e., test node > AP > 5G base station) based on the different technologies, as well as due to the need to relay the packets between the two radio transceivers inside the AP. The combination of IEEE 802.11 and cellular (5G) technologies through the use of an AP introduces additional communication overhead and complexity, which can lead to higher latency. When communicating with an upper-tier node using the AP, the communication involves two wireless legs, one from the test node to the AP over IEEE 802.11 and the other from the AP to the 5G base station over cellular (5G). This introduces additional delays due to the need for relaying the packets between the two different wireless legs and processing the packets at the AP. Furthermore, the relayed packets must be processed by the two radio transceivers inside the AP, which can introduce additional processing delay. This processing delay can also be influenced by the workload of the AP and the congestion of the wireless channels. Not surprisingly, this also results in a higher deviation of the latency compared to 5G-only-based communication. However, communication between the two nodes of the same tier is done using the IEEE 802.11 technology and thus the latency is of the same order as for IEEE 802.11-only scenario.

It also can be seen that when the cellular network is used in the indoor environment, the delay has undergone substantial sudden temporal fluctuations. It can be happened due to various factors such as signal interference, congestion, network load, and mobility of the devices. For instance, when there is congestion in the cellular network, packets may have to wait for a longer time before being transmitted, leading to increased delay. Similarly, when there is signal interference, the quality of the signal may be degraded, leading to increased packet loss and retransmissions, which also increase the delay. Such sudden temporal fluctuations in delay can have a significant impact on real-time and delay-critical applications. For example, in applications that require low latency, such as online gaming, video conferencing, or industrial control systems, any sudden increase in delay can result in a poor user experience or even system failures.

7.3 Power consumption performance

Table 10 indicates that the average power consumption for the 5G-enabled node was more than double compared to the reference case, with an increase from 2.38 W to 5.1 W. During the speed test, the average consumption was as high as 9.38 W, and the peak consumption went over 10 W. The increase in power consumption when using the 5G modem can be attributed to several factors. First, 5G technology requires higher processing power, which leads to increased power consumption by the node's processor. Additionally, 5G networks use higher frequency bands, which require more power to transmit and receive signals. Finally, during the speed test, the node is likely using the 5G modem at its maximum capacity, resulting in even higher power consumption.

During the ping test, the average consumption was 4.55 W. Also, The consumption peaks during the modem's connection to the network, and other actions such as checking network status or preparing for the speed test can be attributed to the energy-intensive processes involved in establishing and maintaining a stable network connection. These processes involve exchanging control information between the modem and the network and adjusting the transmission power to maintain a reliable signal. All of these operations require additional power consumption from the node, resulting in the observed peaks.

According to Table 7 and Figure 52, the overall power consumption of the IEEE 802.11-enabled node was only slightly higher, at one Watt, compared to the reference case, and 1.8 W lower than the 5G-enabled node. The ping tests resulted in a power consumption that was just 0.5 W higher than the reference case, while the speed test resulted in a power consumption that was 3.1 W higher than the reference case. Notably, the peak power consumption of the node during the experiments was quite high and approached 9 W.

The higher power consumption of the IEEE 802.11-enabled node during the speed test can be explained by the increased data transfer rate and associated processing required by the wireless interface. Additionally, the peak consumption approaching 9 W could be due to various factors such as the initialization of the wireless interface and due to the various network management tasks, such as maintaining a connection, managing contention for the wireless medium, and handling error correction, which can also consume additional power. It is also possible that the hardware or firmware implementation of the wireless interface may play a role in the observed power consumption.

As we can see from Figure 53, the power consumption of the AP throughout the test remained mostly stable at around 3.7 W; however, when the node was executing the Speed test the AP's consumption increased to about 5.1 W. The high peak in the power consumption observed when starting the AP is likely caused by the initial charging of the capacitors.

8 CONCLUSION

The appropriate signal level for wireless networks is dependent on various factors, such as the distance between wireless devices, the level of interference, and the specific modulation and coding schemes being used. A stable and consistent radio condition is essential for achieving FRACTAL's objective of improving the performance of wireless networks by dynamically adjusting their configurations based on changing network conditions. It is important to consider the specific requirements of each application and to adjust the signal level accordingly to ensure reliable connectivity and sufficient data rates.

The results of the throughput tests show that the cellular (5G) connection provides high throughput for communication between a test node and an upper-tier node located on the Internet or at the network Edge. However, the throughput available for communication between two nodes interconnected through 5G is substantially lower. The IEEE 802.11 communication also allows for high throughput but is affected by interferences and has limited communication ranges. Combining both approaches through utilizing an AP with 5G backbone and IEEE 802.11 local network resulted in a balanced solution, enabling good throughput for both communications of a node to an upper-tier node and between two nodes. However, the maximum achievable throughput in a communication system depends on various factors, including the communication technology, the specific test scenario, the position of the test node relative to the base station or AP, and the presence of interferences.

Based on the presented study, it can be concluded that the performance of wireless communication systems depends on various factors such as the protocols used, routing paths, interference, and distance between the communicating nodes. The use of different protocols can have a significant impact on the available bandwidth and latency of data transfer, which in turn affects the throughput of the system. The location and quality of the access points can also influence the speed and reliability of wireless connections. Furthermore, the number of nodes and the type of protocol used for medium access control can also impact the overall throughput of the system. Therefore, it is essential to consider these factors when designing and evaluating wireless communication systems. Moreover, it can be concluded that latency in communication can be influenced by various factors such as network architecture, communication technology, signal strength, and interference. In a hierarchical network architecture like the FNEC sub-system, the delay can be higher when lower-tier nodes communicate with upper-tier nodes due to the additional processing and routing required. Communication over short-range IEEE 802.11 technology can result in lower latency and more consistent communication compared to cellular network technology, especially for same-tier communication. The difference in latency between uplink and downlink communication can be attributed to the design of cellular networks where the base station has more powerful antennas and better transmission power for downlink communication. The reduction in mean deviation in LoS conditions can be attributed to the reduced interference and obstacles for the signal to overcome, resulting in a more consistent and reliable signal strength.

In conclusion, the experiments conducted on power consumption performance showed that the power consumption of wireless nodes and access points can vary significantly depending on the type of wireless technology used and the type of activity being performed. The 5G-enabled node had the highest power consumption, with an increase of over double compared to the reference case, while the IEEE 802.11-enabled node had only slightly higher power consumption than the reference case. The power consumption of the access point remained mostly stable throughout the test, but increased during the speed test, likely due to the increased data transfer rate and associated processing required by the wireless interface. These findings highlight the importance of considering power consumption when selecting wireless technologies and designing wireless networks.

The results presented in the study provide valuable insights for selecting and implementing wireless connectivity for FRACTAL nodes, covering a wide range of expected and foreseen use cases. Additionally, the procedures used for enabling 5G connectivity for IoT nodes are innovative and can have applications beyond the FRACTAL project. However, it is important to note that wireless systems are affected by the environment and configurable parameters, which can change over time. Therefore, it is recommended to interpret the numeric results with caution and consider them indicative rather than conclusive. It is more advisable to focus on the trends observed rather than specific numerical values. The findings of throughput, latency, and power consumption are all useful in understanding the performance characteristics of the different wireless networking technologies tested. Overall, these findings can help network designers and engineers make informed decisions about which wireless networking technology to use based on the specific needs of the application or use case, balancing factors such as data transfer rate, responsiveness, and power consumption.

There are many potential areas for future research in the field of wireless connectivity over FRACTAL nodes such as exploring other network architectures, Investigating the impact of different environments, Studying the impact of different device configurations, and investigating the impact of network slicing. While this thesis is examined three network architectures, there may be other architectures that could provide additional insights. For example, one can investigate the performance of a hybrid network that combines both wired and wireless connectivity. Although this thesis considered indoor and LoS environments, it may be valuable to examine how the performance of the three network architectures changes in other different environments. For example, in an urban environment with high levels of interference or in a rural environment with fewer signal obstructions. The performance of wireless networks is heavily dependent on the configuration of the devices involved. Future research could examine how different device configurations impact the performance of the three network architectures. For example, one can examine the impact of using different antennas or different wireless chipsets. This thesis examined the performance of the three network architectures in a general sense. Therefore, future research could focus on specific use cases, such as industrial IoT or smart cities based on FNEC. 5G networks offer network slicing capabilities, which allow the network to be divided into multiple virtual networks. Future research could examine how the performance of the three network architectures discussed in the thesis changes when network slicing is implemented. This could provide insights into the potential benefits and drawbacks of using network slicing in wireless networks. By continuing to investigate these topics, researchers can provide valuable insights into how to optimize the performance of wireless networks with FRACTAL nodes in a variety of settings and use cases.

9 SUMMARY

The Introduction of the thesis has described the EC and its benefits of processing data closer to the source. The FRACTAL project aims to develop a decentralized, federated edge computing infrastructure for IoT applications using a fractal structure that allows for flexible and dynamic resource allocation. The FRACTAL architecture consists of a set of interconnected edge nodes, which can communicate with each other to share resources and data and are connected to a central cloud infrastructure. The FRACTAL communication subsystem also supports dynamic spectrum access and multi-hop communication. The project seeks to design and implement an open, safe, reliable platform to build cognitive edge nodes of variable complexity, guarantee extra-functional properties, evaluate and validate the analytics approach utilizing AI, and integrate fractal communication and remote management features into FRACTAL nodes. The FRACTAL cognitive edge nodes will come in two flavours, a commercial node, and a more research-oriented node.

The related works section has discussed various research papers related to FRACTAL project, edge computing, and its applications. These papers explore topics such as the FRACTAL project, dynamic resource allocation using machine learning, the benefits and challenges of using SDN in edge computing environments, a scalable IoT framework for energy management in connected buildings, resource allocation storms in edge computing, the In-Edge AI framework, and the role of MEC in enabling edge intelligence for 5G and IoT. Overall, the papers highlight the potential benefits of edge computing for enabling new applications and services but also address the challenges of developing secure, energy-efficient, and scalable edge computing systems.

The Experimental Procedures section of the thesis has outlined the process followed to conduct the experiments. This section has several sub-sections, including the test plan, experimental environment, measurement tools, and measurement procedures The experimental environment sub-section has described the setting in which the experiment was conducted. This includes details about the physical space in which the experiments were conducted. The measurement tools sub-section has detailed the tools used to measure the variables in the experiment. This includes HW tools and SW tools with the configurations of these tools. At last, the measurement procedures sub-section has detailed the specific procedures followed to collect the data during the experiments.

The results section of the thesis has reported the data collected from different test cases and reported in separate sub-sections. The discussion about these results especially on the performance of the throughput, latency, and power consumption has been given in the discussion section. Starting from the discussion and the analysis of the obtained results and the possible reasons for the nature of these results.

The conclusion section has concluded that the IEEE 802.11-enabled node had only slightly higher power consumption than the reference case, indicating that it is more energy-efficient compared to 5G-enabled nodes. In terms of latency, IEEE 802.11 technology may be preferable. Moreover, it has also concluded that the IEEE 802.11 WLAN connectivity provided higher throughput for peer-to-peer communication compared to cellular connectivity. The conclusion section has also discussed the possible future research areas on this topic.

Finally, the appendices section has given more detailed instructions that have been used during the implementation of the different test cases. This includes programming codes, AT commands, and other relevant steps used during the configuration of the HW and SW as well as during the measurements. This could be useful for anyone who wants to redo the measurements discussed in this thesis.

10 REFERENCES

[1] H. Hua, Y. Li, T. Wang, N. Dong, W. Li, and J. Cao, "Edge computing with Artificial Intelligence: A machine learning perspective," ACM Computing Surveys, vol. 55, no. 9, pp. 1–35, 2023.

[2] "Fractal research project – fractal project.". Available: https://fractal-project.eu/wp-content/uploads/2021/06/FRACTAL_Brochure.pdf.

[3] Quévremont et al., "WP2 Specifications & Methodology," Project deliverable FRACTAL, pp. 148, 2022.

[4] A. Lojo, L. Rubio, J. M. Ruano, T. Di Mascio, L. Pomante, E. Ferrari, I. G. Vega, F. K. Gurkaynak, M. L. Esnaola, V. Orani, and J. Abella, "The ECSEL Fractal Project: A cognitive fractal and secure edge based on a unique open-safe-reliable-low power hardware platform," 2020 23rd Euromicro Conference on Digital System Design (DSD), pp. 393-400, 2020.

[5] K. Mikhaylov, W. D. Adeesha, "D6.5 FRACTAL communication subsystem validation," Project deliverable FRACTAL, pp. 73, 2023.

[6] A. A.Obiniyi, M. B. Soroyewun, and M. M. Abur, "New Innovations in performance analysis of Computer Networks: A Review," International Journal of Applied Information Systems, vol. 6, no. 8, pp. 1–10, 2014.

[7] P. Garcia Lopez, A. Montresor, D. Epema, A. Datta, T. Higashino, A. Iamnitchi, M. Barcellos, P. Felber, and E. Riviere, "Edge-centric computing," ACM SIGCOMM Computer Communication Review, vol. 45, no. 5, pp. 37–42, 2015.

[8] Z. Zhou, X. Chen, E. Li, L. Zeng, K. Luo, and J. Zhang, "Edge intelligence: Paving the last mile of artificial intelligence with Edge Computing," Proceedings of the IEEE, vol. 107, no. 8, pp. 1738–1762, 2019.

[9] A. Saxena, "Throughput in computer networks," Scaler Topics, 09-Dec-2022. [Online]. Available: https://www.scaler.com/topics/throughput-in-computer-networks/.

[10] Tetcos, "understand measures of network performance: Throughput and delay." Available: https://www.tetcos.com/pdf/v13/Experiments/Understanding-Measure-of-Network-Performance-Throughput-and-Delay.pdf.

[11] R. Tucker and K. Hinton, "Energy-Efficient Networking," GreenTouch. [Online]. Available:https://www.belllabs.com/greentouch/uploads/documents/OFC+2014+Short+Course.pdf.

[12] K. K. Leung, T. Salonidis, and B. J. Ko, "Machine Learning for Dynamic Resource Allocation at network edge," Ground/Air Multisensor Interoperability, Integration, and Networking for Persistent ISR IX, p. 18, 2018.

[13] A. C. Baktir, A. Ozgovde, and C. Ersoy, "How can edge computing benefit from software-defined networking: A survey, use cases, and future directions," IEEE Communications Surveys & Tutorials, vol. 19, no. 4, pp. 2359–2391, 2017.

[14] M. S. Anjana, M. V. Ramesh, A. R. Devidas, and K. Athira, "Fractal IoT: A Scalable IoT Framework for Energy Management in Connected Buildings," Proceedings of the 1st ACM International Workshop on Technology Enablers and Innovative Applications for Smart Cities and Communities, pp. 10-17, 2019.

[15] L. Lovén, E. Peltonen, L. Ruha, E. Harjula, and S. Pirttikangas, "A dark and stormy night: Reallocation storms in edge computing," EURASIP Journal on Wireless Communications and Networking, vol. 2022, no. 1, 2022.
[16] X. Wang, Y. Han, C. Wang, Q. Zhao, X. Chen, and M. Chen, "In-edge ai: Intelligentizing mobile edge computing, caching and communication by Federated Learning," IEEE Network, vol. 33, no. 5, pp. 156–165, 2019.
[17] Y. Liu, M. Peng, G. Shou, Y. Chen, and S. Chen, "Toward edge intelligence: Multiaccess edge computing for 5G and internet of things," IEEE Internet of Things Journal, vol. 7, no. 8, pp. 6722–6747, 2020.
[18] Z. Zhou, X. Chen, E. Li, L. Zeng, K. Luo, and J. Zhang, "Edge intelligence: Paving the last mile of artificial intelligence with Edge Computing," Proceedings of the IEEE, vol. 107, no. 8, pp. 1738–1762, 2019.
[19] M. S. Aslanpour, S. S. Gill, and A. N. Toosi, "Performance evaluation metrics for cloud, fog and edge computing: A review, taxonomy, benchmarks and standards for future research," Internet of Things, vol. 12, p. 100273, 2020.
[20] C. Delacour, S. Carapezzi, M. Abernot, and A. Todri-Sanial, "Energy-Performance Assessment of oscillatory neural networks based on VO2 devices for future edge ai computing," IEEE Transactions on Neural Networks and Learning Systems, pp. 1-14, 2022.
[21] B. Carpentieri and F. Palmieri, "A scheduling framework for latency optimization on 5G mobile edge computing infrastructures," IEEE Globecom Workshops (GC Wkshps), pp. 1-6, 2019.
[22] F. Li and C. Wang, "Artificial Intelligence and edge computing for teaching quality evaluation based on 5G-enabled wireless communication technology," Journal of Cloud Computing, vol. 12, no. 1, 2023.
[23] A. Bourechak, O. Zedadra, M. N. Kouahla, A. Guerrieri, H. Seridi, and G. Fortino, "At the confluence of Artificial Intelligence and edge computing in IOT-based applications: A Review and new perspectives," Sensors, vol. 23, no. 3, p. 1639, 2023.
[24] X. Wang, J. Li, Z. Ning, Q. Song, L. Guo, S. Guo, and M. S. Obaidat, "Wireless Powered Mobile Edge Computing Networks: A survey," ACM Computing Surveys, p. 3579992, 2023.
[25] A. Kundu and W. Arif, "An investigation on Teletraffic attributes for channel selection of IOT objects in cognitive radio internet of Things Networks towards 5G," International Journal of Communication Networks and Distributed Systems, vol. 29, no. 1, p. 1, 2023.
[26] B. Liu, J. Wang, S. Ma, F. Zhou, Y. Ma, and G. Lu, "Energy-efficient cooperation in Mobile Edge Computing-enabled Cognitive Radio Networks," IEEE Access, vol. 7, pp. 45382–45394, 2019.
[27] F. Z. Benidris and S. Limam, "A novel distributed spectrum management in mobile edge computing based Cognitive Radio Internet of Things Networks," Multiagent and Grid Systems, vol. 17, no. 4, pp. 367–382, 2022.
[28] B. Hussain, Q. Du, A. Imran, and M. A. Imran, "Artificial Intelligence-powered mobile edge computing-based anomaly detection in Cellular Networks," IEEE Transactions on Industrial Informatics, vol. 16, no. 8, pp. 4986–4996, 2020.
[29] M. S. Elbamby, C. Perfecto, C.-F. Liu, J. Park, S. Samarakoon, X. Chen, and M. Bennis, "Wireless edge computing with latency and reliability guarantees," Proceedings of the IEEE, vol. 107, no. 8, pp. 1717–1737, 2019.

[30] B. Gokalgandhi, M. Tavares, D. Samardzija, I. Seskar and H. Gacanin, "Reliable Low-Latency Wi-Fi Mesh Networks," IEEE Internet of Things Journal, vol. 9, no. 6, pp. 4533-4553, 15 March, 2022.

[31] Y. Tian and T. Yoshihiro, "Traffic-Demand-Aware Collision-Free Channel Assignment for Multi-Channel Multi-Radio Wireless Mesh Networks," IEEE Access, vol. 8, pp. 120712-120723, 2020.

[32] Aijaz, Adnan. "High-Performance Industrial Wireless: Achieving Reliable and Deterministic Connectivity Over IEEE 802.11 WLANs." IEEE Open Journal of the Industrial Electronics Society, vol. 1, pp. 28–37, 2020.

[33] A. Bourechak, O. Zedadra, M. N. Kouahla, A. Guerrieri, H. Seridi, and G. Fortino, "At the confluence of Artificial Intelligence and edge computing in IOT-based applications: A Review and new perspectives," Sensors, vol. 23, no. 3, p. 1639, 2023.

[34] D. A. Warnakulasuriya, K. Mikhaylov, and O. L. Alcaraz Lopez, "Wireless power transfer for Bluetooth Low Energy based IOT device: An empirical study of energy performance," 2022 14th International Congress on Ultra Modern Telecommunications and Control Systems and Workshops (ICUMT), pp. 1-7, 2022.

[35] K.-H. Phung, H. Tran, Q. Nguyen, T. T. Huong, and T.-L. Nguyen, "Analysis and assessment of Lorawan," 2018 2nd International Conference on Recent Advances in Signal Processing, Telecommunications and Computing (SigTelCom), pp. 241-246, 2018.

[36] TP-Link, "Archer T3U Plus," available (2023.03.10): https://www.tplink.com/us/home-networking/usb-adapter/archer-t3u-plus/

[37] TeleWell, "5G WAN: User Manual," available (2022.03.10): https://telewell.fi/files/-TeleWell_5G_Industrial_router.pdf

[38] Morrownr, "Morrownr/88X2BU-20210702: Linux driver for USB WIFI adapters that are based on the RTL8812BU and RTL8822BU chipsets - v5.13.1," GitHub. Available: https://github.com/morrownr/88x2bu-20210702.

[39] Quectel,"Quectel_rmu500-EK_User_Guide_V1.0," Quectel, 10-Dec-2021. Available: https://www.quectel.com/download/quectel_rmu500- ek_user_guide_v1-0.

[40] Quectel,"Quectel_rg50xq&RM5xxQ_Series_AT_Commands_Manual_V1. 2," Quectel, 10-Dec-2021. Available: https://www.quectel.com/download/quectel_rg50xqrm5xxq-_series_at_comman ds_manual_v1-1.

[41] KAITOTEK, "Qosium," available (2023.03.13): https://www.kaitotek.com/qosium

[42] Kaitotek Oy, Qosium Documentation. [Online]. Available: https://docplayer.net/210774069-Qosium-documentation-kaitotek-oy-qosium-documentation-qosium-documentation.html.

[43] Keysight, "N6705B DC Power Analyzer, Modular, 600 W, 4 Slots," available (2023.03.13): https://www.keysight.com/fi/en/product/N6705B/dc-poweranalyzer-modular-600-w-4-slots.html

[44] Ecoflow, "EcoFlow RIVER Pro Portable Power Station," available (2023.03.13): https://us.ecoflow.com/products/river-pro-portable-powerstation

[45] Red Hat, "Chapter 23: configuring PTP using PTP4L Red Hat Enterprise Linux 6," Red Hat Customer Portal. Available: https://access.redhat.com/documentation/enus/-red_hat_enterprise_linux/6/html/deployment_guide/chconfiguring_ptp_using_ptp4l.

[46] Speedtest, "Speedtest," available (2023.03.14): https://www.speedtest.net/about

[47] CableFree, “LTE Metrics including RSRP, RSRQ and SINR,” available (2023.03.15): https://www.cablefree.net/wirelesstechnology/4glte/lte-rsrq-sinr/

[48] "Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications - Redline," in IEEE Std 802.11-2020 (Revision of IEEE Std 802.11-2016) - Redline , vol., no., pp.1-7524, 26 Feb. 2021.

[49] “LTE and 5G Linux USB driver user guide.” [Online]. Available: https://sixfab.com/wpcontent/uploads/2020/12/Quectel_LTE5G_Linux_USB_Driver_User_Guide_V2.0.pdf.

11 APPENDICES

Appendix 1 Compile the Linux kernel modules on the test node

In this thesis, Nvidia's Jetson Nano device used as the test node and it is required additional configuration in order to use 5G modules. Following steps were tested in an Intel Core i5, Ubuntu 20.04.4 LTS, 64-bit Notebook.

Step 1: Create directories in *$HOME*
- cd $HOME

Step 2: Create a directory to place all the JETSON files and built kernel.
- mkdir -p l4t-gcc
- mkdir -p l4t-gcc/kernel_out/build
- mkdir l4t-gcc/kernel_out/modules

Step 3: Download BSP, Sample root filesystem, BSP sources and GCC toolchain tool from here. Move all files to l4t-gcc directory.

Step 4: Untar these tarball files
- cd $HOME/l4t-gcc

Following will create a directory named `Linux_for_Tegra`
- tar -xvf Jetson-210_Linux_R32.7.1_aarch64.tbz2

Following will extract under `Linux_for_Tegra/source/public`
- tar -xvf public_sources.tbz2

Following will extract toolchain folder at the same level with `Linux_for_Tegra`
- tar -xvf gcc-linaro-7.3.1-2018.05-x86_64_aarch64-linux-gnu.tar.xz

If there is no `rootfs` folder in `Linux_for_Tegra`, please create it with root.
- sudo tar -xvf Tegra_Linux_Sample-Root-Filesystem_R32.7.1_aarch64.tbz2 -C Linux_for_Tegra/rootfs

Step 4: Set the environment variables for compling later.
- export JPATH=$HOME/l4t-gcc/Linux_for_Tegra
- export JROOTFS=$JPATH/rootfs
- export TOOLCHAIN_PREFIX=$(pwd)/gcc-linaro-7.3.1-2018.05-x86_64_aarch64-linux-gnu/bin/aarch64-linux-gnu-
- export LOCALVERSION=-tegra
- export CROSS_COMPILE=${TOOLCHAIN_PREFIX}

Step 5: Untar kernel source code from public directory.
- cd Linux_for_Tegra/source/public

This extracts both kernel code and dtb files.
- tar -xvf kernel_src.tbz2

Step 6: Compile kernel
- cd $HOME/l4t-gcc

Step 7: Set kernel and modules path and output folder
- export JKERNEL=$JPATH/source/public/kernel/kernel-4.9/
- export TEGRA_KERNEL_OUT=$HOME/l4t-gcc/kernel_out/build
- export TEGRA_MODULES_OUT=$HOME/l4t-gcc/kernel_out/modules

Step 8: Set tegra config for the use
- make -C $JKERNEL ARCH=arm64 O=$TEGRA_KERNEL_OUT CROSS_COMPILE=${TOOLCHAIN_PREFIX} tegra_defconfig

Step 9: Modify kernel configuration

- make -C $JKERNEL ARCH=arm64 O=$TEGRA_KERNEL_OUT CROSS_COMPILE=${TOOLCHAIN_PREFIX} menuconfig
- Include (Y) and save USB driver for GSM and CDMA modem as given in [49] chapter 3.2.6 step 4.
- Include (Y) and save Multi-purpose USB Networking Framework as given in [49] chapter 3.3.2 steps 4.
- Include (Y) and save QMI_WWAN driver for Qualcomm MSM based 3G and LTE modems as given in [49] chapter 3.4.2 steps 4.
- Include (Y) and save PPP (point-to-point protocol) supports given in [49] chapter 3.5 Step 4.

Step 10: Build dtb, module and Image files. j{n} n depends on the number of process can run on your host device.

- make -C $JKERNEL ARCH=arm64 O=$TEGRA_KERNEL_OUT CROSS_COMPILE=${TOOLCHAIN_PREFIX} -j{n}

Step 11: Install modules.

- make -C $JKERNEL ARCH=arm64 O=$TEGRA_KERNEL_OUT INSTALL_MOD_PATH=$TEGRA_MODULES_OUT modules_install

Step 12: Copy the built kernel Image in kernel_out/ build/arch/arm64/boot/Image to kernel directory in Linux_for_Tegra directory.

- cp $TEGRA_KERNEL_OUT/arch/arm64/boot/Image $JPATH/kernel/

Step 13: Copy the files in kernel_out/ build/arch/arm64/boot/dts to kernel/dtb directory in Linux_for_Tegra directory.

- cp $TEGRA_KERNEL_OUT/arch/arm64/boot/dts/* $JPATH/kernel/dtb/

Step14: Copy built modules in kernel_out/modules/lib to rootfs directory in Linux_for_Tegra directory.

- cp -a $TEGRA_MODULES_OUT/lib/* $JROOTFS
- Make sure the device has already gone into the recovery mode (remove the jumper in power pin and put the Jumper between REC and GND pins).
- Connects PC host to Jetson device through MicroUSB.
- Execute lsusb commend to check if your Ubuntu has been detected the USB device.

You should see this string < Bus 001 Device 069: ID 0955:7f21 NVidia Corp. > printing out, otherwise please check the USB that is attached to Jetson.

Step 15: Finally, execute flash.sh in the Linux_for_Tegra directory as below.

- cd Linux_for_Tegra
- sudo ./apply_binaries.sh
- sudo ./flash.sh jetson-nano-qspi-sd mmcblk0p1

If a monitor has been connected to the Jetson nano device, Jetson device should boot up successfully at this stage.

Appendix 2 Installation of the 5G drivers on the test node

Before begins the installation of the 5G drivers on the test node. The necessary configurations must be done to the Linux kernel of the test node as given in Appendix 1.

Step 1: Connect and power the device using power cable.
Step 2: Download the Quectel 5G USB drivers from here.
Step 3: Extract all the zip files and rename the folders so that the folder names do not contain "&" symbol. Otherwise, this will show errors.

- sudo make ARCH=arm64 CROSS_COMPILE=aarch64-linux-gnu- install

Step 4: Go to the extracted Qmi_wwan directory and install the driver.

- sudo make ARCH=arm64 CROSS_COMPILE=aarch64-linux-gnu- install

Step 5: Go to the extracted GobiNet directory and install the driver

- sudo make ARCH=arm64 CROSS_COMPILE=aarch64-linux-gnu- install

Step 6: Go to the extracted SerialOption/v4.9.111 directory and install the driver. Driver version is based on the kernel version (Uname -r)

- sudo make ARCH=arm64 CROSS_COMPILE=aarch64-linux-gnu- install

Install the Quectel-CM
Step 1: Go to the QConnect directory and install the QConnect manager.

- sudo make ARCH=arm64 CROSS_COMPILE=aarch64-linux-gnu-

Step 2: Prepare "busybox udhcpc" tool

Quectel-CM will call "busybox udhpc" to obtain IP and DNS, and "busybox udhpc" will call script file */usr/share/udhcpc/default.script* to set IP, DNS and routing table for Linux board.

The source codes of "busybox udhpc" tool can be downloaded from here, then enable CONFIG_UDHCPC with the command below and copy the script file *[BUSYBOX]/examples/udhcp/simple.script* to Linux board (renamed as */usr/share/udhcpc/default.script*).

- sudo make ARCH=arm64 CROSS_COMPILE=aarch64-linux-gnu- menuconfig

go to Netwok Utilities > udhcpc and press Y to enable it. Press ENTER and exit. Finally save the configuration.

- sudo make ARCH=arm64 CROSS_COMPILE=aarch64-linux-gnu-
- sudo make ARCH=arm64 CROSS_COMPILE=aarch64-linux-gnu- install

Test AT Commands
Step 1: Install and run UART port tools such as "minicom", "busybox microco", "socat".

- sudo apt-get install socat

When the USB serial option driver has been installed in the module, the device files named as ttyUSB0, ttyUSB1, ttyUSB2, etc. will be created in directory /dev. The AT port is usually /dev/ttyUSB2, which is the second ttyUSB port created by the USB serial option driver.
Step 2: Run the Socat on ttyUSB2 and test 'at+cops' command.

- sudo socat - /dev/ttyUSB2,crnl

Appendix 3 AT commands for 5G connection

Set the <APN>

- AT+CGDCONT=1, "IP","<APN>", "0.0.0.0"

Search for LTE band B7 (Note: the band depends on the telecom operator, B7 is the configuration in the 5GTN.

- AT+QNWPREFCFG="lte_band",7

Search for 5G New Radio (NR) Non-standalone (NSA) band n78

- AT+QNWPREFCFG="nsa_nr5g_band",78

Set network search mode to search only LTE & NR5G bands

- AT+QNWPREFCFG="mode_pref",LTE:NR5G

Set not to disable 5G NR standalone (SA)/NSA

- AT+QNWPREFCFG="nr5g_disable_mode",0

Query current Serving Cell

- AT+QENG="servingcell"

Should be able to see 5G NR cell if the 5G network is available.
Unlock for uncommercial network capability.

- AT+QMBNCFG="Select","ROW_Commercial"

Set roaming Preference to home network.

- AT+QNWPREFCFG= "roam_pref",1

Alternatively, one can also use the following single line instead giving AT commands one by one.

- AT+CGDCONT=1,"IP","5gtnoulu","0.0.0.0";+QNWPREFCFG="lte_band",7;+QNWPREFCFG="nsa_nr5g_band",78;+QNWPREFCFG="mode_pref",LTE:NR5G;+QNWPREFCFG="nr5g_disable_mode",0;+QENG="servingcell";+QMBNCFG="Select","ROW_Commercial";+QNWPREFCFG= "roam_pref",1

In addition to these one can get the signal quality output (such as signal strength; network mode; serving cells;) from following AT commands. Definitions for these commands can be found from AT commands manual [40].

- AT+QENG="servingcell";+qnwinfo;+qsinr;+qrsrq;+qrsrp;+csq

Setup a data Call from quectel-CM tool
Step 1: Open a terminal from quectel-CM directory.
Step 2: Give root access and setup a data call from the APN.

- Sudo su
- ./quectel-CM -s <APN

Appendix 4 Installation and configuration of PTPd and SFTP saver

Installation of PTPd and configuration of PTP master and PTP slave.

Step 1: Installation of PTPd

- Sudo apt -y install ptpd

Step 2: execute PTP master

- sudo ptpd -I – masteronly -U -u -C

where <interface> is the identifier of the interface used (can be obtained with command ifconfig) and <slave IPi> is the IP address of a slave PTP client.

Step 3: Execute PTP client

- sudo ptpd -I – masteronly -U -u -C

where <interface> is the identifier of the interface used (can be obtained with command ifconfig) and <Master IPi> is the IP address of a PTP Master.

Configuration of SFTP client

Step 1: Install Secure Shell Protocol (SSH)

- sudo apt -y install ssh

Step 2: Change SSHD configuration for SFTP group

- sudo nano /etc/ssh/sshd_config
- paste the following lines at the end or bottom of the file

```
Match group sftp
ChrootDirectory /home
X11Forwarding no
AllowTcpForwarding no
ForceCommand internal-sftp
```

Step 3: Restart SSH services

- sudo systemctl restart ssh

Step 4: Create SFTP users group

- sudo addgroup sftp

Step 5: Create a new SFTP user and set the password for this user

- sudo useradd -m sftp_user -g sftp
- sudo passwd sftp_user

Step 6: Grant full permissions to the specific directory

- sudo chmod 700 /home/sftp_user/

Appendix 5 Measurement procedure

Step 1: Power up the test devices and give them time to boot up.
Step 2: Enable PTP time synchronization for accurate timestamping as given in Appendix 4.
Step 3: Launch measurement in Qosium Scope:

- Substep 3.a: Select the Two-port measurement in Qosium Scope Probe Setup section in Probes tab.
- Substep 3.b: Enter an appropriate test name and description of the test in Qosium Scope Measurement Info section in Probes tab.
- Substep 3.c: Enter the local IP address of the test node in Qosium Scope Primary Measurement Point section Probes tab. (Note: depending on the test to run, the IP address should be obtained either with ifconfig commend or by checking the GUI of AP)
- Substep 3.e: Enter the IP address of the TS in Qosium Scope Secondary Measurement Point section in Probes tab. (Note: depending on the test to run, the IP address should be obtained either with ifconfig commend or by checking the GUI of AP)
- Substep 3.f: Click on the Connect to Qosium Probe in Measurement Control section in Qosium Scope.
- Substep 3.g: Select the correct primary probe and secondary probe network interfaces from Topology tab in Qosium Scope and leave all the other settings as default.
- Substep 3.h: From Measurement tab, chose the Packet filter mode to manual filter and configur the manual filter between test node (local IP) and the target node. Leave all the other settings as default.
- Substep 3.i: In Qosium Scope Results tab, enable the “get packet results” and enabled all the options in Save to File Settings section.
- Substep 3.k: Click on start measurements in Qosium Scope Measurement Control section.

Step 4: Perform the Speed test (see Section 6.2.6) at DUT1 and log the results.
Step 5: Use traceroute to check the route between the test nodes (i.e., DUT1 and DUT2/TS) and log the results.
Step 6: Measure the radio channel conditions and log the results.

For tests IDs 3,4,5,6,9 and 10 – check the GUI of the AP (some parameters such as RSSI each device connected to AP through WiFi can be seen from Basic Network > WiFi > WiFi client list in router’s web UI.) o For test IDs 1,2,7 and 8 – the RSRP, RSRQ, SINR and RSSI measurements were obtained using the AT as given in Appendix 3 and Examplary results of this command’s execution are depicted in the following figure.

Step 6: Start the experiment (either latency or throughput, as discussed above);
Step 7: During the execution of the experiment the experiment notes were filled containing the timestamps of the measurement, the observed throughput and
other relevant notes on experiment run.
Step 8: After the end of experiment: log the results (i.e., start/end time and total time required for transferring the test file in uplink/downlink as reported by SFTP client, or log the results of executing ping command).
Step 9: Stop the measurement in Qosium Scope.

• Step 10 (done after all the measurements are completed): Copy the logs and collected by Qosium results and store them for further processing.

```
>>sudo socat - /dev/ttyUSB2,crnl
>>[sudo] password for cwc:
>>at+qeng="servingcell";+qnwinfo;+qsinr;+qrsrq;+qrsrp;+csq
       at+qeng="servingcell";+qnwinfo;+qsinr;+qrsrq;+qrsrp;+csq
       +QENG: "servingcell","NOCONN"
       +QENG: "LTE","FDD",244,27,57805,70,3000,7,3,3,89,-61,-8,-37,19,15,60,-
       +QENG: "NR5G-NSA",244,27,74,-65,32,-11,636000,78,8,1
       +QNWINFO: "FDD LTE","24427","LTE BAND 7",3000
       +QSINR: 26,26,29,32,NR5G
       +QRSRQ: -10,-10,-10,-10,NR5G
       +QRSRP: -75,-65,-63,-68,NR5G
       +csq: 31,99
OK
```

Figure 54. Listing of exemplary at+qeng="servingcell" command results showing the status of the 5G connection and encoded RSSI, RSRP, RSRQ and SINR values