Cloud-radio access network functional split over ethernet-based fronthaul analysis and performance improvement for fronthaul

Mountaser, Ghizlane

Awarding institution:
King's College London

Cloud-Radio Access Network Functional Split over Ethernet-based Fronthaul: analysis and performance improvement for fronthaul.

Ghizlane Mountaser

Supervisor: Dr. Toktam Mahmoodi

Department of Engineering
King's College London

This dissertation is submitted for the degree of
Doctor of Philosophy
at
Center for Telecommunications Research
School of Natural and Mathematical Sciences

King's College London, UK February 2021

Declaration

I hereby declare that except where specific reference is made to the work of others, the contents of this dissertation are original and have not been submitted in whole or in part for consideration for any other degree or qualification in this, or any other university. This dissertation is my own work and contains nothing which is the outcome of work done in collaboration with others, except as specified in the text and Acknowledgement.

Acknowledgement

First of all, I would like to express my heartfelt gratitude to Dr. Toktam Mahmoodi, my PhD supervisor. I owe her a great sincere gratitude for her support, motivation, inspiration and valuable guidance. I would like to thank her for her continued efforts to show me the way forward.

I am also deeply thankful to Prof. Mischa Dohler, my PhD co-supervisor who has engaged me in a number of interesting meetings and discussions that were inspiring, challenging, and useful experience throughout my research.

I would like to express my deep gratitude to the entire team at the Center for Telecommunications Research and special thanks to the 5G lab team, in particular Dr. Fragkiskos Sardis. I would also like to thank my colleagues with whom I had the opportunity to collaborate with including Dr. Maria Lema Rosas, Dr, Massimo Condoluci, Maliheh Mahlouji and Enric Pardo Grino.

I take this opportunity to specifically thank Prof. Hamid Aghvami for his amazing discussions on various interesting research topics and Prof. Osvaldo Simeone for his thoughts and supportive suggestions for my work. His support was one of the important contributions to this study.

I would like to thank BT for funding my PhD and especially for the internship in their office. I am also grateful to Dr. Richard MacKenzie for his mentorship during my BT internship.

I would like to express my thankfulness to the good friends at Engineering department for supporting each other.

Finally, I would like to express my sincere gratitude to my family, my relatives, and to my husband and my daughters in particular, because of their love and support throughout my PhD journey, which gave me the strength to make it happen.

Abstract

The emerging Cloud-Radio Access Network (RAN) architecture within the fifth generation (5G) of wireless networks plays a vital role in enabling higher flexibility and granularity. Cloud-RAN is able to scale and enhance network efficiency to support new features over the next several years by means of centralisation. In Cloud-RAN architecture, baseband functions are centrally deployed providing several benefits in terms of offering high level of cooperation between base stations and RAN sharing. These aspects allow dynamic reconfiguration of resources enabling to fulfil diverse requirements of various vertical industrial applications.

Cloud-RAN significantly benefits from the emerging technologies such as softwarization and virtualisation. Softwarization/virtualisation within Cloud-RAN allows adaptive allocation of RAN functions between components of Cloud-RAN introducing the so-called flexible functional split. Flexible functional split is a promising approach aiming at providing greater flexibility to sufficiently fulfil diversified service requirements needed by 5G. However, selecting the appropriate functional split is an important factor.

On the other hand, flexible functional split enables the use of different types of the transport network. The Ethernet-based fronthaul can be an attractive solution for Cloud-RAN. On the one hand, deployment of Ethernet-based fronthaul enables Cloud-RAN to provide more diverse, flexible and cost-efficient solution. On the other hand, Ethernet-based fronthaul requires packetised communication, which imposes challenges in delivering stringent latency requirements between RAN functionalities. To this end, in this thesis, an implementation of Cloud-RAN with functional split over Ethernet-based fronthaul has been considered to benefit from versatility and flexibility provided by functional split and the use of commodity and low-cost industry standard equipment.

In the first part of the thesis, the feasibility of the Medium Access Control (MAC) and physical (PHY) split over the Ethernet-based fronthauling has been addressed. An implementation and deployment of hardaware-based Cloud-RAN has been developed and the impact of packetisation on the fronthaul has been evaluated.

In the second part, three alternatives of RAN function splits have been implemented on a hardware platform supporting Cloud-RAN with Ethernet-based fronthauling. The impact that each functional split have on the fronthaul in the delivery of 5G services has been identified through a system-level evaluation. Then, recommendation on the most appropriate split for a given 5G scenario has been proposed.

In the third part, a solution to improve the performance of Ethernet-based fronthaul by means of multi-path diversity and erasure coding has been proposed. Under a probabilistic model that assumes a single service, the average latency required to obtain reliable fronthaul transport and the reliability-latency trade-off are first investigated. The analytical results are then validated and complemented by a numerical study that accounts for the heterogeneous service.

In the last part, the proposal to improve reliability of the Ethernet-based fronthaul has been validated by evaluating the performance of the model in industry-garde testbed,

Finally, the thesis is concluded with some future researches.

Table of contents

List of figures

List of tables

Chapter 1

Introduction

1.1 Introduction

The fifth generation of mobile communication networks (5G) is facing different multiple challenges compared to previous generations, with an ever increasing number of use cases, it is intended to satisfy all the new applications being considered by so-called *industry verticals*. Besides giving users mobile broadband services, 5G is also expected to provide technological solutions for time sensitive communications with the rising of different vertical domains, such as automated cars, the Tactile Internet, or various scenarios within Internet of Things. The wide range of services being provided is changing the paradigm of cellular networks, and concepts as common as cells are no longer relevant. In fact, 5G is evolving to a device-centric approach, where the configurability (and reconfigurability) of the network is key to satisfy the quality of service.

To support this new type of network, trends like softwarization and centralisation are being widely considered by both the research community and standardisation bodies [1]. In particular, centralisation in the radio access network (RAN) has been discussed in the context of Centralised or Cloud-RAN [2, 3].

In Cloud-RAN architecture, baseband functions are decoupled from radio elements and centrally deployed on a shared pool of resources at a data centre running over commercial off-the-shelf equipment, e.g. servers, offering advantages such as cooperative solutions, interference mitigation, improved load balancing and RAN sharing, among others.

Recent developments in softwarization and virtualisation of mobile networks provide the platform for deployment of Cloud-RAN solutions through flexible functional split. Flexible functional split is a promising approach aiming at providing greater flexibility by freely splitting baseband functions between central cloud and distributed entities providing different level of centralisation each with different requirements that would create more business models.

The split could be predefined for different network slice, or it can be dynamically changed for different types of traffic or depending on the network conditions, which could be configured via top-level network controller and offered as a service. This aspect allows for dynamic reconfiguration and resource management in an effective, programmable and rapid manner, leading to greater flexibility and diversity. This flexibility/diversity provides an architecture that allows for openness leading to the so-called *open RAN*. This allows operators to customise the network on the basis of their own specifications, leading to the best possible implementation and, ultimately, to more effective innovation and commercial deployment.

As such, Cloud-RAN is envisaged as a promising solution for the cellular network to support diverse requirements of the envisioned 5G vertical industrial applications by leveraging the two essential approaches which are cloudification and flexible functional split.

However, the new interface that is between centralised and distributed units, called the fronthaul, may have a negative effect on the efficiency of the network if the performance of the interface is not properly managed. Therefore, the ability of the fronthaul to flexibly scale up with data rate and deliver stringent requirements of latency and reliability to support 5G has become critical to the success of Cloud-RAN. The need for flexibility in the fronthaul has opened up the possibility of flexibly splitting RAN functionalities between centralised and distributed units. The advantage of such an architectural approach is the use of different transport such as packet-based fronthaul that allows using commodity and low-cost industry standard equipment. This fact has triggered several standardisation bodies to define packet-based fronthaul as a possible solution for Cloud-RAN fronthaul. Nonetheless, packet-based networks make it more challenging to ensure the high reliability and low-latency Key Performance Indicators (KPI) expected by 5G systems.

Defining the splitting point, and maintaining the tight interaction between different functionalities in RAN is, however, critical. The success of such architecture depends on the selection of the appropriate functional split and the efficiency of the fronthaul. Therefore, the main questions to raise in this research are as follows: how packet-based fronthaul can support the levels of latency and reliability needed for 5G innovative applications and use cases; and which functionality split can be the best architectural choice for a given scenario. In this context, this thesis aims to answer these questions.

1.2 Motivation and Contributions

To this end, in this thesis, a Cloud-RAN with different functional split options has been implemented over Ethernet-based fronthaul. The main motivations behind this choice is to benefit from the followings:

1. Flexibility and versatility provided by flexible functional split: flexible functional split is a promising approach aiming at providing greater flexibility to sufficiently fulfil diversified service requirements of vertical industrial applications. Flexible functionality split provides a variety of options on what functions to locate in a distributed unit versus at the central unit. There is no one-option-fits-all that can be implemented in Cloud-RAN that can meet the needs of different vertical industrial applications. The selection of the appropriate functional split is an important factor, as a number of parameters have to be considered. It is therefore important to research the effect of the flexibility in RAN configuration on the delivery of 5G services in order to determine which split can be most suitable for each service.

2. Deployment of fronthaul network on the existing Ethernet infrastructure: cloud-RAN with Ethernet-based fronthaul has received tremendous attention given that Ethernet is a widely deployed technology, it is cost-effective, and it relies on off-the-shelf standard equipment. Another major benefit of Ethernet is its capability of flexibly scaling with the dynamic nature of data traffic providing more diverse, flexible and cost-effective infrastructure. Taking full advantage of the current infrastructure would be of the utmost importance for the business success of operators. On the other hand, Ethernet-based fronthaul requires packetised communication, which imposes challenges in delivering desired latency and reliability to meet the 5G service requirements. This yet remains challenging for the adoption of packet-based fronthaul in Cloud-RAN as latency and reliability play a crucial role in future wireless networks. As such, there is also a need to enhance performance over Ethernet link to support 5G services.

Being motivated by the advantages of the two points 1 and 2 that they bring to the cellular system and their challenges, which are still subject to active discussion in the research community, the contribution of this study is then focused on Cloud-RAN with different functional split options over Ethernet-based fronthaul. The main contributions to this study can be summarised as follows:

- First time feasibility study of packetising data and fronthauling over the Ethernet with MAC-PHY split in OAI platform. In this context, the amendment to OAI is implemented to decouple the LTE RAN protocol stack into two entities: one entity consisting of RRC, PDCP, RLC and MAC modules and the other entity consisting of the PHY module in order to be executed as standalone entities. Further amendment is implemented to packetise PDUs exchanged between the two entities and to transmit/receive packets over the fronthaul interface. To this end, a testbed based on the amended OAI platform, including Ethernet fronthaul, IP packet generator, monitoring and logging tools, is being deployed in this thesis to demonstrate the feasibility of the MAC-PHY split over the Ethernet. The feasibility is demonstrated in terms of the correct operation of the overall system and the compliance of the latency and jitter results with the standard requirements under the particular system parameters set out in this thesis.
- Implement alternate RAN splits that are PDCP-RLC and intra-PHY splits in the testbed to evaluate the impact of each functional split on the fronthaul in the delivery of 5G services.
- Proposal for reliable low-latency Ethernet-based fronthaul. The proposed solution improves the reliability of Ethernet-based fronthaul while maintaining the latency below a strict latency bound required by 3GPP.

1.3 Thesis Outline

The rest of the thesis is organised as follows.

- Chapter 2 introduces the architecture of Cloud-RAN and provides the technical background for understanding the research field in this thesis. The chapter includes a brief summary of different functional splits and different fronthaul network technologies.
- In Chapter 3, experimenting with splitting MAC and PHY layer with fronthauling through Ethernet is carried out. The hardware experimental setup

is detailed in this Chapter and the performance of the system is examined from latency and jitter perspectives.

- In Chapter 4, three different functionality splits are implemented in Cloud-RAN, and the impact of each of these splits on communication latency and jitter is examined. The traffic for 5G classes is modelled based on the 3GPP traffic model. Then the Chapter provides a recommendation, based on the results obtained, which split can be the most suitable for each 5G classes of traffic.

- Chapter 5, presents a model to improve the reliability of the fronthaul while ensuring the latency requirement is met by means of multi-path diversity and erasure coding of the fronthaul packets. Two strategies of sharing of fronthaul resources are used, namely non-orthogonal and orthogonal. The performance of the fronthaul is then analysed analytically and through simulations making an assumption of purging scenario. The proposed solution is compared with conventional single-path fronthaul transport and multi-path methods based on duplication.

- Chapter 6, presents an implementation of a hardware testbed of the system model proposed in Chapter 5. The main reason why the hardware test is presented in a separate Chapter, is that in the previous Chapter, we made an assumption of purging scenario, i.e. redundant blocks are removed from the queues. In this Chapter, however, the system is non-purging. In this Chapter the reliability-latency trade-off is investigated.

- Finally, concluding remarks are in chapter 7 with some directions for future works.

1.4 Publications

The publications related to the main contributions of this thesis are stated as follows:

1. G. Mountaser, M. L. Rosas, T. Mahmoodi and M. Dohler, "On the Feasibility of MAC and PHY Split in Cloud RAN," 2017 IEEE Wireless Communications and Networking Conference (WCNC), San Francisco, CA, 2017, pp. 1-6, doi: 10.1109/WCNC.2017.7925770.

In this paper, I implemented and tested MAC-PHY split over Ethernet in an industry-grade hardware testbed, and as the lead author, was responsible for the main writing of the paper while receiving suggestions and advice on results presentations and the relevant state of the art from my co-author, Dr Maria Lema. This publication has contributed to Chapter 3.

2. G. Mountaser, M. Condoluci, T. Mahmoodi, M. Dohler and I. Mings, "Cloud-RAN in Support of URLLC," 2017 IEEE Globecom Workshops (GC Wkshps), Singapore, 2017, pp. 1-6, doi: 10.1109/GLOCOMW.2017.8269135.

   In this publication, I extended the implemented in 1 to two additional functional splits and I tested the system by modelling different classes of 5G traffic. As the lead author, I was responsible for the main writing of the paper while my co-author, Dr Massimo Condoluci, helped with traffic modelling and result presentations. This publication has contributed to Chapter 4.

3. G. Mountaser, T. Mahmoodi and O. Simeone, "Reliable and Low-Latency Fronthaul for Tactile Internet Applications," in IEEE Journal on Selected Areas in Communications, vol. 36, no. 11, pp. 2455- 2463, Nov. 2018, doi: 10.1109/JSAC.2018.2872299.

   In this publication, I modelled the solution for reliable low-latency Ethernet-based fronthaul and evaluated it in MATLAB. My co-author, Prof. Osvaldo Simeone, has provided guidance in developing the encoding model. This publication has contributed to Chapter 5.

4. G. Mountaser, M. Mahlouji and T. Mahmoodi, "Latency Bounds of Packet-Based Fronthaul for Cloud-RAN with Functionality Split," ICC 2019 - 2019 IEEE International Conference on Communications (ICC), Shanghai, China, 2019, pp. 1-6, doi: 10.1109/ICC.2019.8761906.

   In this publication we derive theoretical lower and upper bounds on Ethernet-based fronthal delay, and evaluated through simulations in MATLAB. In this work, my co-author, Maliheh Mahluji, worked on the analytical derivation, and hence those are not included in this thesis. As the lead author, I had the main writing role, simulated MATLAB model and analysed the results.

5. Ghizlane Mountaser; Toktam Mahmoodi, "Flexible Function Split Over Ethernet Enabling RAN Slicing," in Radio Access Network Slicing and Virtualization for 5G Vertical Industries , IEEE, 2021, pp.209-220, doi: 10.1002/9781119652434.ch11.

In this book chapter, we present our overall research vision on how flexible functional split and multi-path Ethernet-based fronthauling can be implemented as an enabler for RAN slicing.

Chapter 2

Background

2.1 Introduction

Cloud-RAN is one of the innovative architectural solutions for mobile networks aiming at providing an infrastructure satisfying the communication needs of a wide range of services and deployments. The Cloud-RAN was proposed by China Mobile in 2009 to support the significant increase in base station density needed by the evolution of mobile communications from 2G to 4G and to help operators to deal with many different challenges. Cloud-RAN have been adopted by many organisations, such as Next Generation Mobile Networks (NGMN). In fact a dedicated C-RAN project, called P-CRAN, was founded in NGMN in 2011 to promote the idea of Cloud-RAN given its advantages.

2.1.1 Cloud-RAN Architecture

Base station handles the transmission/reception of user and control data to/from several users using multiple access protocols in air interface. The processing of the signal is made up of two parts: radio processing and baseband processing.

In the legacy cellular system architecture, the base station combines radio and baseband processing capabilities. The baseband is connected to the antenna module, which is usually located within a few metres of the radio module, through a coaxial cable. Each base station supports its own components and has a dedicated housing facility that is not shared with other base stations. The base station is connected to the core network, which is normally located far from it. Figure 2.1(a) illustrates this configuration, where the base station is placed near antenna.

The Cloud-RAN concept changed the paradigm of the cellular system by decoupling the radio unit from the baseband unit allowing BBU equipment to be placed in a more convenient place with lower leasing and maintenance costs. The architecture of the Cloud-RAN, as shown in Figure 2.1(b) consists of three main components, namely Remote Radio Head (RRH), the BaseBand Unit (BBU) and the fronthaul network. The RRH performs radio processing (digital to analogue conversion and analogue to digital conversion) and RF functions (amplification and filtering) and is deployed at the remote site close to the antenna. The BBU performs the baseband processing in charge of several cells. Multiple BBUs from several sites are clustered into the BBU pool at the central office or data centres with powerful computation capability and storage. BBUs are inter-connected via a high bandwidth and low latency switching network. This switching enables cells information to be exchanged in the BBU pool efficiently and with low-latency which facilitates multi-cell processing. The BBU computational resources can be dynamically allocated and shared between different cell sites enabling utilisation of resources efficiently based on data traffic and enabling significant multiplexing gains leading to substantial reduction in total computing resources. Multiple RRH are connected to the BBU pool via a low latency and high bandwidth optical link. The separation distance can be up to 40 km. Nevertheless, the length of the link is limited by the timing requirements of the radio. The CPRI interface is the most widely used interface.

In a more evolved solution, BBUs processing is virtualised by leveraging the two key approaches that are softwarization and virtualisation. In virtual Cloud-RAN, each BBU is a virtual node over abstracted physical hardware. Virtual BBUs are connected through a virtual link. This architecture allows the effective utilisation of network resources leading to a programmable network, which increases the scalability and flexibility of the system [4]. This architecture is shown in Figure 2.1(c).

The Cloud-RAN approach provides several advantages over traditional cellular networks.

- It is easy to maintain due to two perspectives. The first is simpler radio equipment, as the latter holds only radio frequency functionality. The second is placing BBUs of many cells at the same location which facilitates to add them and upgrade them easily, thereby improving scalability. These can reduce network operational costs spent on cooling, site maintenance and RAN upgrade which helps to reduce the cost of operating expense (OPEX) & capital expenditure (CAPEX). In particular, the recent trial from China

Mobile has shown OPEX and CAPEX can be lowered by 53% and 30% respectively using Cloud-RAN [5].

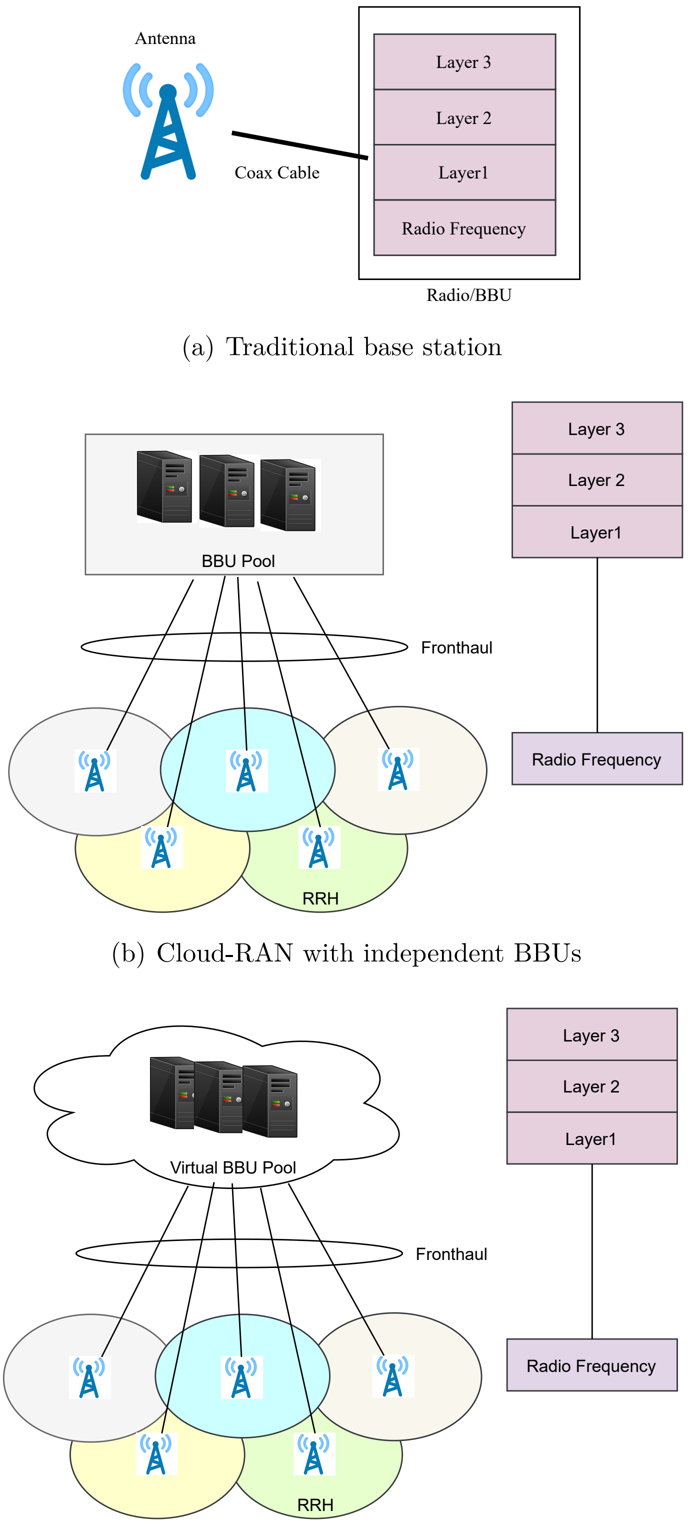
- Improving network performance and efficiency by sharing centralised signal processing resources. The resources are used on-demand, depending on the traffic load. This ensures that resources are dynamically and cooperatively used to fulfil the needs of the network efficiently.
- Optimising network capacity by enabling effective coordination between adjacent cells. With centralised BBUs, cell information can be effectively shared and exchanged due to tighter cooperation between BBUs. In addition, cell information can be easily processed due to the large and powerful computations of the BBU pool. This advantage facilitates the implementations of schemes that mitigate inter-cell interference such as Enhanced Inter-cell Interference Coordination and Coordinated Multipoint Transmission and Reception that are two important features in LTE-Advanced.
- It facilitates upgrade and allows new features to be introduced without any modifications to the radio equipment. Therefore, it can evolve easily and integrate emerging technologies to support new business models.

2.1.2 Full Centralisation and CPRI

The traditional Cloud-RAN corresponds to the so-called *full centralisation* where the layer 1, layer 2 and layer 3 RAN functions are holds in BBU.

In full centralisation architecture, fronthaul transports data in the form of in-phase and quadrature (IQ) samples over typically Common Public Radio Interface (CPRI). CPRI is the most commonly used open interface in conventional Cloud-RAN deployments. It standardises the protocol interface between BBU and RRH, enabling the interoperability of equipment from different vendors. CPRI is designed to transport radio signal over other media and it supports single and multiple hops with chain, tree and ring topology.

In DL, LTE baseband signal is generated in the BBU, and then the data in the form of IQ is transmitted over the fronthaul to RRH where digital to analogue conversion is done. The data of one antenna for one carrier is mapped to one IQ data flow that is carried by one CPRI link. CPRI specifies three different flows that are multiplexed over the interface using layer 1 and layer 2 protocols. The three flows are:



(a) Traditional base station

(b) Cloud-RAN with independent BBUs

(c) Cloud-RAN with virtualised BBUs

Fig. 2.1 Base station evolution architecture.

- User plane data: transmitted in the form of one or more IQ flows, with each IQ data flow representing data from one antenna for one carrier, the so-called antenna carrier.
- Control and management plane: control data is responsible for synchronisation and error detection and correction. While management data is for operation, administration and maintenance of the CPRI.
- Synchronization and timing: holds synchronisation and timing information that can be used to ensure the precision of the frequency and timing of RF signal transmission and reception on the air interface.

Despite the attractive advantages of conventional Cloud-RAN, the architecture faces several implementation challenges: the first challenge is the constant data rate of CPRI that is independent of user activity. In fact, it requires continuous transport of CPRI stream even if no user traffic is present. The Equation (2.1) shows CPRI data rate demands over the fronthaul. The capacity for CPRI transmission scales linearly with the number of RF chains (N_{RFchain}) and the sampling rate R_s which depends on the transmission bandwidth as shown in Table 2.1. Given that CPRI scales linearly with both carrier bandwidth and number of antenna ports, CPRI poses an important limiting factor for 5G where massive number of antenna ports are expected to be used.

$$R_{\text{CPRI}} = 2 \times N_{\text{RFchain}} \times R_s \times N_{(res,\text{CPRI})} \times N_{\text{ovhd}} \times N_{\text{LineCode}}, \tag{2.1}$$

where:

- 2 accounts for the complex nature of the samples(IQ) data
- N_{RFchain} corresponds to the number of RF chains. Traditionally, each antenna has individual RF chain. However, when hybrid beamforming is employed number of RF chains might be much lower than the number of antennas,
- R_s is the sampling rate (15.36 MHz per 10 MHz bandwidth [6]),
- $N_{(res,\text{CPRI})}$ is the resolution of binary representation of symbols to be transported (the number of bits per I or Q sample is 8-20 bits for downlink, 4-20 bits for uplink [6]),

- N_{ovhd} is the CPRI overhead (16/15 because a basic frame consists of 16 words, the first word is reserved for control, while the other 15 words are used to carry IQ data samples),
- N_{LineCode} is the overhead due to either 8B/10B coding (10/8) or 64B/66B code (66/64) [6].

Table 2.1 CPRI bandwidth requirement in 5G for one PHY-RF based split. Considering $N_{(res,\text{CPRI})} = 16$ bits and $N_{(res,\text{CPRI})} = 10/8$ and assuming number of RF chains equal to the number of antenna ports.

Transmission bandwidth	10 MHz	20 MHz	100 MHz	1 GHz
Sampling rate (R_s)	15.36 MHz	30.72 MHz	150.36 MHz	$1,500.36$ MHz
Number of antenna ports = 2	1 Gbps	2 Gbps	20 Gbps	100 Gbps
Number of antenna ports = 8	4 Gbps	8 Gbps	80 Gbps	400 Gbps
Number of antenna ports = 64	32 Gbps	64 Gbps	640 Gbps	$3,200$ Gbps
Number of antenna ports = 256	128 Gbps	256 Gbps	$2,560$ Gbps	$12,800$ Gbps

Table 2.1 shows the data rate demands for transporting IQ data using the CPRI interface for different transmission bandwidths and antenna port numbers supported in 5G [7]; assuming only digital beamforming is employed for precoding. In this case, the number of RF chains is equal to the antenna number and, as such, the fronthaul data rate is proportional to the number of antenna ports. It can be seen that the transmission of IQ data requires high transport capacity of 8 Gbps when considering a 20 MHz transmission bandwidth and eight antenna ports (4G specifications). The bandwidth requirement becomes even more demanding when considering massive MIMO with up to 256 antenna ports and larger bandwidth that are introduced in 5G, reaching capacities as high as $2,560$ Gbps when considering a 100 MHz transmission bandwidth and 256 antenna ports and higher capacity of $12,800$ Gbps when considering transmission bandwidth of 1 GHz and 256 antenna ports expected to be supported in 5G new radio. Thereby, the high throughput requirement poses challenges for the fronthaul interface and the system may encounter capacity bottleneck in 5G.

However with massive MIMO, RF chains might be limited due to cost and hardware complexity as well as in order to reduce the power and energy consumption [8]. In limited RF chains, RRH is equipped with number of RF chains that is smaller than number of antenna ports. This is achieved by deploying hybrid beamforming i.e., digital and analogue beamformings [9] in which BBU performs digital beamforming for RF-chains and RRH performs analogue beamforming for antenna ports. Thus, in the case of limited RF chains, digitised radio signals are transmitted in proportion to the number of RF chains; thus, the data rate requirement set out in Table 2.1 will be reduced by a factor (number of antennas/number of RF chains).

The second challenge is that delay and jitter values must be kept to a minimum. Values for permissible Round Trip Time of the fronthaul range from 100 μs, to up to 400 μs. NGMN's guideline is to design a network such that the one-way latency is below 100 μs, and the jitter retained on the nanosecond scale. The Third challenge is the accurate synchronisation requirement of 8.138 nanoseconds is needed by CPRI with a frequency variation of (± 0.002 ppm) to obtain clock information correctly [6]. These critical requirements are making CPRI very challenging.

2.2 RAN Functional Decomposition

Functional splitting is one of the core innovative concepts of Cloud-RAN. Functional splitting is introduced in order to relax the excessive bandwidth and latency requirements, as well as to enhance the flexibility of the fronthaul by allowing for a more flexible placement of baseband functions between central and distributed units. This approach would therefore allow for a variety of deployment options.

2.2.1 RAN Split in 3GPP

From 3GPP perspective, Cloud-RAN architecture in 5G composes of CU that is responsible for non-real-time functions and DU component responsible for real-time functions. 3GPP defined eight functional split options whereby less baseband functionalities are centralised providing different levels of centralisation from a fully centralised to a fully distributed architecture. Figure 2.2 shows the eight possible function splits that are defined by 3GPP taking LTE protocol stack as a reference stack for discussion. Figure 2.2 also shows an illustration of the trade-off between latency and transport requirements versus radio complexity. The eight functional split options are as follows:

- Option 1: only RRC is in CU, PDCP, RLC, MAC, PHY and RF are in DU. The main benefits of this split are that the fronthaul data rate scale

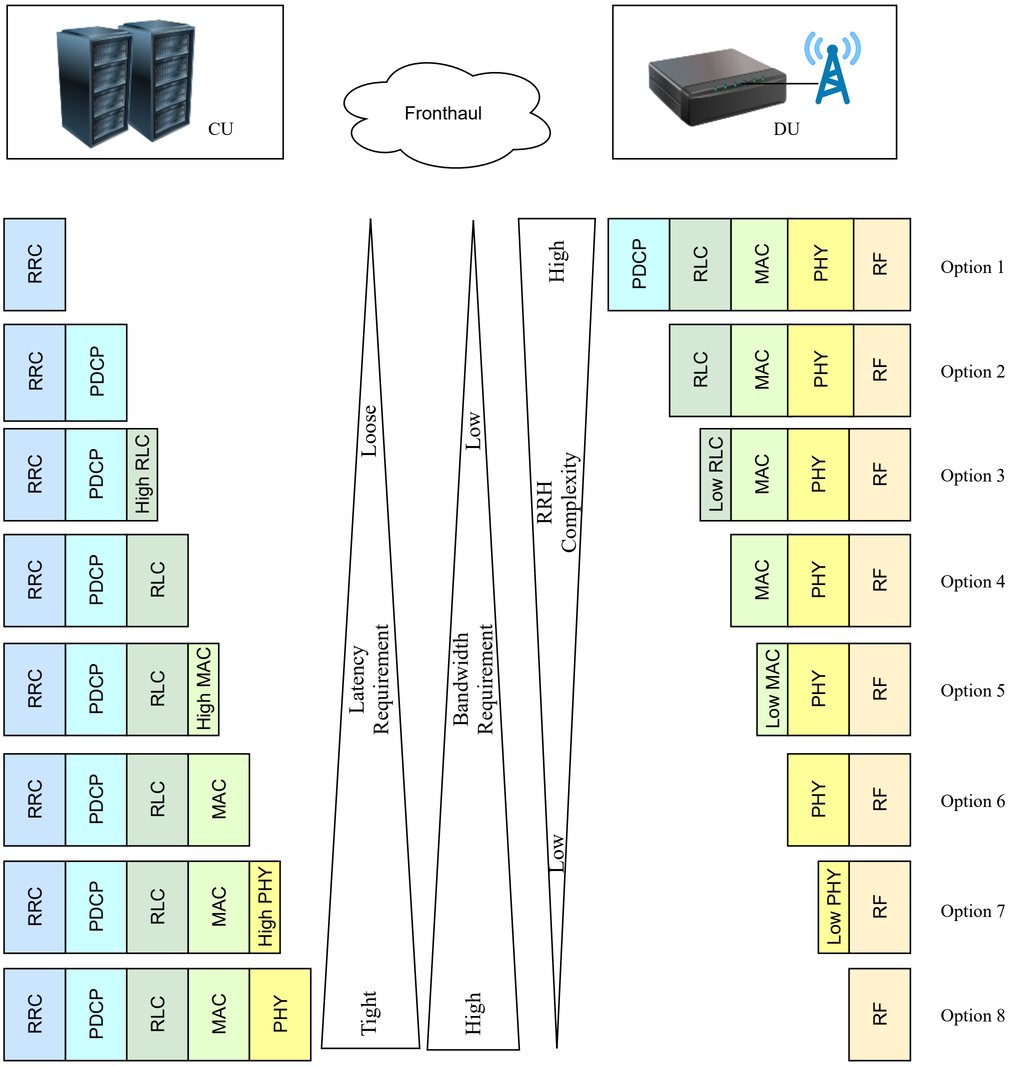


Fig. 2.2 Options for functional split for Cloud-RAN.

flexibly according to the user plane traffic and the interface typically cope with relatively larger latency. Moreover, having all the user plane protocol in DU, i.e. closer to the edge, this split is beneficial for low latency use cases. However, because PDCP which performs security is in DU, this split requires distribution of security key.

- Option 2: for this split, RRC and PDCP are centralized whereas RLC, MAC, PHY and RF are distributed. This split point is intensively considered by standard bodies and researchers and it is like split 3C which has been standardized in LTE dual connectivity [10]. Having PDCP in CU, the split is effectively suitable for aggregation at PDCP level because it doesn't necessarily require a strict lower layer synchronization and it is not subject to the restrictions of real-time. It's also suitable for mobility and handover as it enables reducing handover failure probability. As in the case of split option 1, fronthaul data rate scales relative to user traffic.

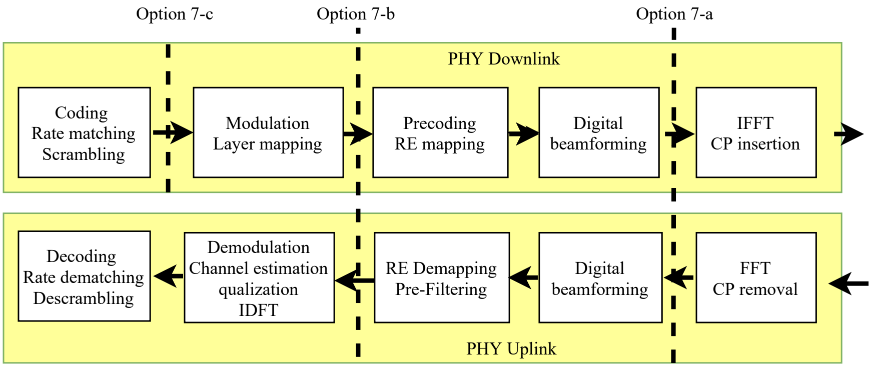


Fig. 2.3 Split option 7 adopted in 3GPP standardisation.

- Option 3: the split is performed within RLC sublayer. From 3GPP point of view, two options are defined in this split. The first option centralizes RLC automatic repeat request which can help to recover from fronthaul interface failure using automatic repeat request recover mechanism. However, since automatic repeat request is responsible for end-to-end re-transmission, this option is more sensible to latency than split option 2. The second option separates transmit and receive RLC where transmit RLC is in DU whereas receive RLC is in CU. There is, thus, no constraint on DL data transmission. However, placing transmit RLC in DU increases processing and buffer requirements in DU.

- Option 4: RRC, PDCP and RLC are in CU. MAC, PHY and RF are located in DU. The fronthaul interface transports RLC PDU; thus, the data rate of the interface is dependent on user activity. The downside is that splitting is not easy to implement and could be impractical because of the tight interaction between MAC and RLC. For example, the scheduling mechanism in DL is the interaction between MAC and RLC. This interaction should not take a long time, so that the deadline for scheduling is met.

- Option 5: the split is within MAC, where part of the MAC functionality, such as scheduling decisions, is in CU, while the time sensitive MAC processing is in DU.

  Option 6: for this option, the split is between MAC and PHY wherein only PHY and RF are in DU. The split offers a high level of centralisation and pooling gain compared to options above. Transport blocks are transmitted over the fronthaul interface, hence the data rate of the fronthaul scales with user traffic. The latency requirement is relaxed compared to CPRI.

- Option 7: this is intra-PHY split whereby part of PHY functions is in CU. The other part of PHY and RF are in DU. There are three variants of this choice which are 7-a, 7-b and 7-c as shown in Figure 2.3. Key advantage of this option is the high degree of centralisation with a significant reduction on the fronthaul data rate requirement compared to CPRI by moving antenna related operations to the DU (e.g., DL antenna mapping, FFT, etc.), However, the DU in this option is more complex than the one in option 8.

- Option 8: this option corresponds to fully centralized RAN architecture. While this option benefits from the advantages of full centralisation, it has a very high data rate requirement on the fronthaul due to the transmission of IQ data in time domain.

In light of 3GPP standardisation activities towards new radio Release 15, functional splits in Cloud-RAN are categorised into two types according to the latency requirements of fronthaul as follows:

- Higher layer split: split options 1, 2 and 3 (according to terminology in Figure 2.2) are all suitable candidates for this category from 3GPP point of view. Option 2, however, is selected as the higher layer functional split architecture by 3GPP in new radio release 15 specification [3]. The right figure in Figure 2.4 shows such a split whereby RRC and PDCP are in CU while the other layers are in DU. The latency requirement of the interface is relaxed and will be determined from the latency-based target service. The interface between CU and DU in higher layer split is standardised as F1 interface [3] that supports an IP transport network layer to be carried over the underlying Carrier Ethernet network.

- Lower layer split: this split is preferable in scenarios to realise enhanced performance. Within 3GPP, it was concluded in [3] that the splits 6 and 7 (according to terminology in Figure 2.2) are possible options for a lower layer split architecture. The left figure in Figure 2.4 shows an example of low layer split with option 6.

The 5G Cloud-RAN architecture further splits CU into control and user blocks to allow for greater flexibility in 5G architecture. The control and user blocks can

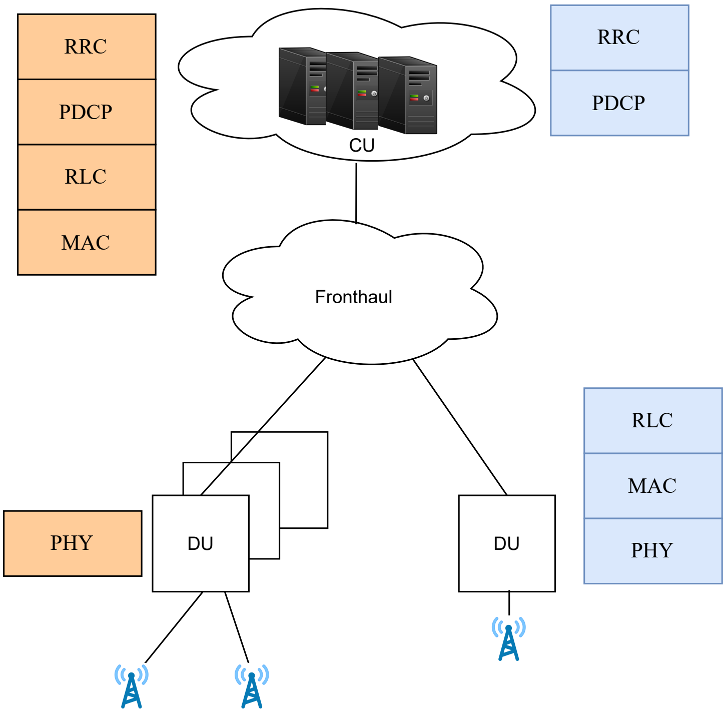


Fig. 2.4 High layer split and low layer split options.

exist on the same hardware, on separate hardware, on the same site or on separate sites. For example, user data may be decoupled from control data in order to place user plane block near to the user to support applications with stringent latency requirement.

To this end, flexible functional split is a promising approach providing different levels of centralisation. Each of these envisioned split options imposes different requirements on fronthaul. Depending on the split point, the latency and bandwidth requirements change as shown in Table 2.2 [3]. There is a trade-off between latency and transport requirements versus radio complexity as shown in Fig. 2.2. In general, the lower the split point, the higher the centralisation degree, the higher is the required interface data rate and the tighter the latency requirement.

Table 2.2 Bandwidth and latency requirements for different split points in 5G.

Split Point Option	One-way Latency	DL Bandwidth	UL Bandwidth
2 (PDCP-RLC)	1.5-10 ms	4016 Mbps	3024 Mbps
6 (MAC-PHY)	250 μs	4133 Mbps	5640 Mbps
7-c (Intra-PHY)	250 μs	10.1-22.2 Gbps	53.8-86.1 Gbps

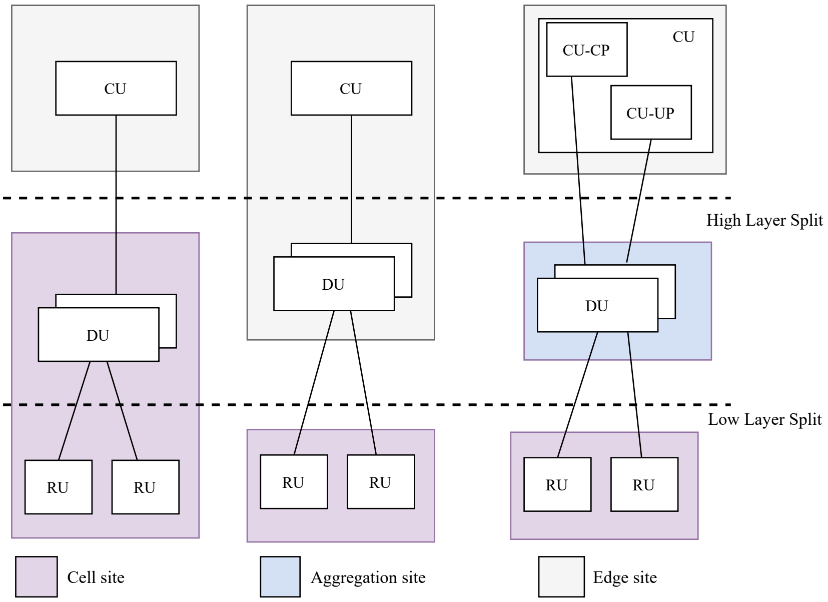


Fig. 2.5 Example of placement scenarios for Cloud-RAN components in NGMN

2.2.2 RAN Split in NGMN

NGMN has focused their attention on Cloud-RAN and has contributed actively in many projects to provide the industry the guidance on building and implementing Cloud-RAN. As one of their deliverables of Cloud-RAN is a study in which different function split options are presented and analysed for low and high latency fronthaul. NGMN followed up with providing an overview of the various RAN functional split options and possible transport options in 5G RAN.

The 5G cloud-RAN architecture, unlike 3GPP, consists of three RU, DU and CU blocks [11]. The fronthaul network splits into two interfaces which are referred to as the high layer split point connecting the DU to the CU and the low layer split point connecting the RU to the DU. The aim of this architecture is to provide a variety of options on how to split functions and where to place them. The CU component holds functions above high layer split interface, the DU holds functions between high layer split interface and low layer split interface, and the RU includes all functions below the low layer split interface.

NGMN architecture allows for flexible distribution of protocol stacks between the three components, each of which can be located in different physical locations offering different radio network deployment scenarios, each with different requirements. This provides new possibilities for modularity and flexibility. Figure 2.5 shows an example of placement scenarios at different sites namely: cell site which is close to the user, aggregation site that is an intermediate site, typically used for transport aggregation and edge site that is the most central site in RAN. For example, the left side of the Figure 2.5 shows the DU can be co-located with the RU at the cell site. The advantage of such an architecture is that the functions are close to the user making this architecture suitable for services requiring low latency. The benefit in centralisation, however, is low. On the other hand, the scenario in the middle of the Figure 2.5 gives an example of where the CU and DU are at the edge site, allowing more RAN functions to be centralised and thereby benefiting from centralisation advantages such as pooling gains.

2.2.3 RAN Split in O-RAN

Cloud-RAN is evolving towards the concept of Open-RAN (O-RAN). O-RAN aims to create an architecture that is open, scalable and intelligent with interoperable interfaces and off-the-shelf equipment leading to more competitive solution.

The overall O-RAN architecture consists mainly of four functional software components: DU, CU, RAN Intelligent Controller and Orchestration and Network Management Systems [12, 13] that are deployed as VNFs or containers and communicate with RU hardware to make it run more efficiently.

The key element of O-RAN architecture is RAN Intelligent Controller whose primary objective is to optimise RAN functionality by optimising RAN elements and resources. This is achieved by leveraging emerging technology such as artificial intelligent and machine learning to enhance resource management capabilities.

O-RAN primarily supports split 7-b (according to 3GPP terminology). For the fronthaul, operators are trying to push openness into the fronthaul that will connect between the various disaggregated components aiming to step away from the CPRI interface.

That said, O-RAN is envisioned as an architecture to support diverse requirements of the envisioned 5G vertical industrial applications.

2.3 Fronthaul Technologies

The Fronthaul network has an important role to play in the success of Cloud-RAN towards realising 5G. Relevant fronthaul solutions must fulfil the requirements of the supported service while assuring the correct performance of all procedures. Selecting the optimal option for fronthaul technologies depends on a number of factors, such as the requirements on the fronthaul, deployment scenario, i.e. the location and distance to the cell site, and the characteristics of the area where the fronthaul network is to be deployed. The following communication technologies and protocols are possible solutions for the deployment of fronthaul links.

2.3.1 Wireless Fronthaul

- Dedicated microwave links: microwave fronthaul links operating at carrier frequencies between 6 and 60 GHz can offer rates from $10 - 100$ Mbps up to 1 Gbps, depending on the range and weather conditions. Multi-path fading is a key factor that degrades the efficiency of microwave links operating below 10 GHz, whereas for microwave links operating above 10 GHz weather conditions are the primary cause of communication disturbance. The range limitation and sensitivity to weather events limit the scalability of the technology and its capability to sustain the traffic growth of 5G and beyond 5G [14].

- Millimetre-wave: millimetre-wave is another promising technology to deliver high bandwidth wireless in the 60 GHz and 70-80 GH ranges. Millimetre-wave relies on the high availability of wide-band RF channels to deliver such a high throughput using simple single-channel configurations [15]. This simplicity in design makes millimetre-wave beneficial in terms of cost to the fronthaul of high capacity. Millimetre-wave can achieve low latency of less than 200 μs of round trip delay per single hops. This is because the capacity at 60 GHz is delivered with simple single RF channel, so there is no need for signal processing that causes extra latency [15]. While microwave can achieve a latency of less than 1ms of round trip delay per hop. There are, however, drawbacks, such as high absorption and restricted coverage. The environment, such as rain and moisture, makes the millimetre-wave signal attenuation very high. Works in [16] offered a detailed tutorial on the use of millimetre-wave frequencies, in particular the 60 GHz band and the 70-80 GHz band, for connectivity as a key milestone for potential 5G networks.

2.3.2 Optical Fibre for Fronthaul

Optical fibre is the most advanced fronthaul technology.

- Dedicated fibre links: this is the most commonly regarded transport choice for fronthaul. They are attractive for their low latency and high capacity, as they can support up to 40 Gbps per channel. The key downside of dedicated fibre is the large number of fibre links required to connect RRHs to BBUs where each link carries a separate flow, thus the cost of deploying the fibre optic system increases linearly with the length and number of fibre links deployed. As a result, the deployment cost usually makes it prohibitive to connect several RRHs to the BBU using this technology. Another downside is that the fibres are not available at all sites.

- Passive Wavelength Division Multiplexing (WDM): when fibre resources are limited, it is possible to multiplex fronthaul channel on a link using a passive WDM multiplexer then the wavelengths are separated using passive demultiplexers. Hence, WDM provides a cost-effective solution for shorter distances of up to 70 km. Other WDM choices are Dense WDM (DWDM) and Coarse DWDM (CWDM). DWDM links can be amplified and can thus be used to transmit data at much longer distances. CWDM can accommodate up to 10 Gigabit. However, DWDM can provide an overall throughput as high as 100 Gbps. As no additional delay is added by passive WDM filtering, WDM latency can be as low as 5 μs. However, they are too expensive to deploy as dedicated fronthaul infrastructures.

- Passive Optical Networks (PON): PON is currently used as a low-cost solution to deploy a fibre optics-based fronthaul multi-hop network. PON is a fibre optic network that uses a point-to-multipoint topology and optical splitters to transmit data from a single transmission point to different endpoints for users. In comparison to the active optical network, PON is inherently efficient at operating costs, because electrical power is only needed at the point of transmission and reception.

  Among all forms of PONs, Time Division Multiplexing PON is seen as a cost-effective candidate as it is able to share optical fibres and transmission equipment across multiple fronthaul connections [17]. Packets of each Optical Network Unit are multiplexed using time division multiple access. Specific examples include Gigabit-PON (G-PON), which provides 2.5 Gbps downstream and 1.25 Gbps upstream, and Gigabit Ethernet PON (G-EPON), which is being upgraded to 10 G-EPON by IEEE 802.3a, offering data rate in the order of 10 Gbps downstream and upstream. G-PON and G-EPON are deployed as fibre to the home in fact G-PON is the most incorporated in an emerging protocol. However, it does not satisfy the latency requirement of 5G, especially on the upstream this is due to the dynamic bandwidth allocation based algorithm that is used to request PON resources for transmission in uplink. Another form of PON is WDM-PON that allows several optical network units to share fibre link. WDM-PON can provide dynamic resource allocation at low latency.

2.3.3 Ethernet Multi-Hop For Fronthaul

Ethernet multi-hop fronthaul: Ethernet fronthaul multi-hop networks can help reduce cost and simplify network deployment and management by sharing the network infrastructure among multiple DUs and Cloud-RAN systems through its packet-switched operation. Another major benefit of Ethernet is its capability of flexibly scaling with the dynamic nature of data traffic and can be extended to support different topology options. However, its use for fronthauling imposes many challenges, such as lack of synchronisation, high latency and high jitter (more details are covered in Section 2.4).

It is expected that the fronthaul network will evolve to a multi-hop topology. Indeed, recent packet-based fronthaul architecture is of considerable interest to industry and the research community to minimise costs and improve flexibility and scalability. .

- enhanced CPRI (eCPRI), which was introduced in 2017, defines packetised interface employing Ethernet or Internet protocol, as a possible interface for the fronthaul.
- O-RAN considered Ethernet as a selected transport.
- NGMN supports functional splits, which enables variable bit rate on fronthaul and loosens both latency and bandwidth requirements, which can be optimally transported in packet-based transport.

2.4 Towards Functional Split over Ethernet-Based Fronthaul for 5G

2.4.1 Functional Split over Ethernet Fronthaul: Considerations

Functional split in Cloud-RAN enables the CU to inter-work with the DU over a non-ideal transport network such as Ethernet. Among other options including optical fibre or wireless networks, Ethernet is a promising solution for the transport

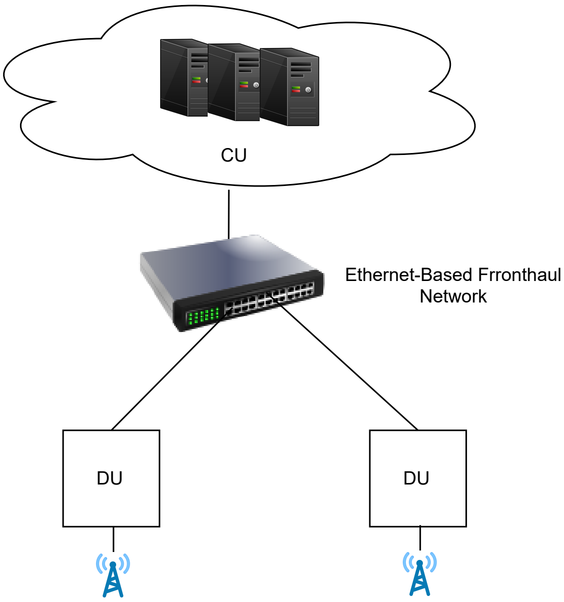
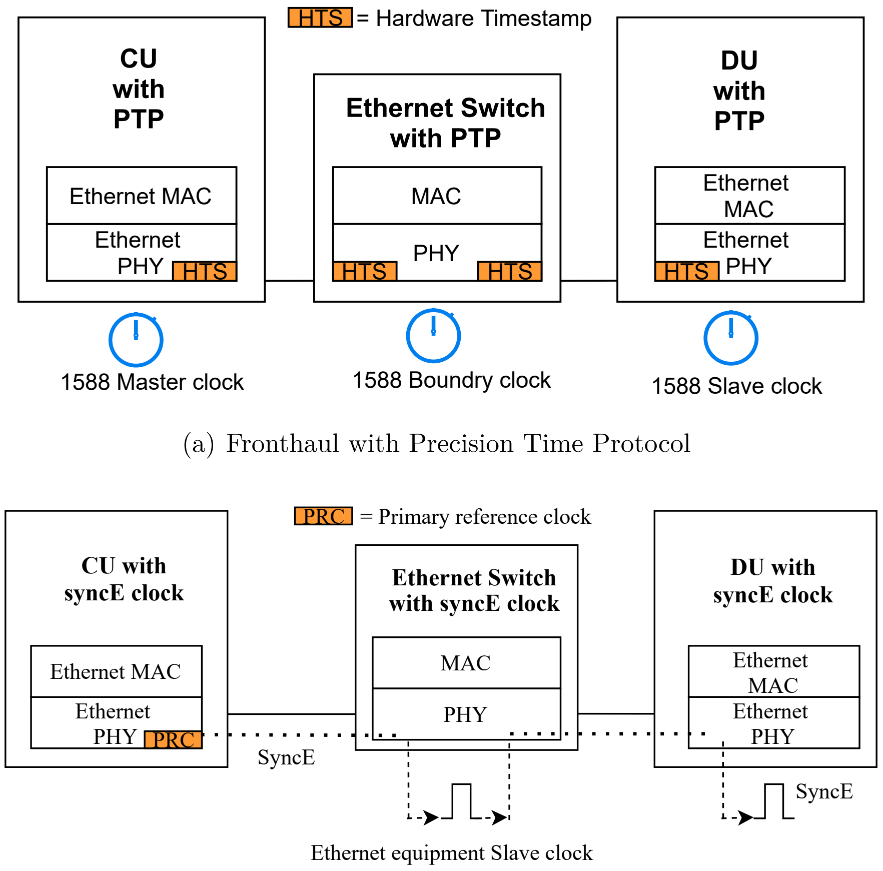


Fig. 2.6 Cloud-RAN architecture with Ethernet-based fronthaul.

network for 5G given its wide availability and cost-efficiency. In fact, this is in line with the main objective of Cloud-RAN, which is to reduce the costs of deployment. Using Ethernet links allows to:

- Use lower cost-industry off-the-shelf standard equipment.
- Sharing and convergence with Ethernet-based networks: Ethernet-based fronthaul is promoted by operators due to the existence of the Ethernet in the transport network of the operator. As a consequence, several standardisation bodies defined Ethernet as a possible solution. In particular, eCPRI supports flexibility in the splitting PHY functionalities. This is enabling its traffic to be carried on the Ethernet [18]. Consequently, eCPRI defines packetised interface employing Ethernet protocol, as a possible interface for the fronthaul.
- Enable statistical multiplexing gain: Ethernet-based fronthaul allows statistical multiplexing when traffic is variable to efficiently utilise the network capacity. As mentioned in section 2.2.1, functional split allows fronthaul traffic to scale with the actual traffic. The variable fronthaul traffics of different DUs on the same network can be multiplexed resulting in statistical multiplexing (see Figure 2.6). The work in [19] showed that multiplexing gain on fronthaul links can be achieved when traffic starts to be variable bit rate.



(a) Fronthaul with Precision Time Protocol

(b) Fronthaul with synchronous Ethernet

Fig. 2.7 Synchronisation in Cloud-RAN with Ethernet-based fronthaul.

- Monitor and orchestrate the network with the use of virtualisation and software defined network (SDN): with the use of these approaches in the Ethernet the performance of the system can be improved. The work in [20] provided scheduling of traffic in an Ethernet taking into account SDN. The results showed that the scheduling algorithm helped to overcome contention effects and remove frame jitter. Similarly, the work in [21] used an SDN based scheduling algorithm that removed jitter to satisfy the CPRI requirement for CPRI over Ethernet transmission.

- Enable network slicing; allows several operators to share the same transport network infrastructure [22];

However, Ethernet-based fronthaul requires packetised communication, which imposes challenges in delivering the KPIs expected by 5G as Ethernet-based fronthaul faces the following challenges:

- Lack of synchronisation: synchronisation, in terms of frequency, phase and time, is essential to the proper functioning of the cellular infrastructure.

  In Cloud-RAN, DU needs to be synchronised with CU in order for radio components in the network to maintain accurate transmission. Frequency offset between radio components can result in overlapping signals, beamforming distortions, adjacent channel interference and so on. 5G new radio requires synchronisation across frequency, time and phase.

  Although the CU can be equipped with a GPS receiver, it is expensive to equip each DU with a GPS receiver. The synchronisation problem in the Ethernet-based network can therefore be solved by **i)** adding the Precision Time Protocol (PTP) which is IEEE 1588v2 standard. PTP delivers frequency, phase and time for providing accurate synchronisation. This is achieved by a device called a Grandmaster, which distributes master timing to slaves across the network through packets carrying timestamps. Time stamping can be achieved either with software in application layer or in MAC layer or with hardware in MAC or PHY layer. Figure 2.7(a) is an example of time stamping in hardware. PTP with software can precisely synchronise clocks to sub-microsecond precision. Authors in [23] showed based on simulation the feasibility of providing accurate phase synchronisation using Precision Time Protocol. Whereas hardware-based PTP can even provide nanosecond time precision [24]. If the transport network includes switches which don't support PTP, this results in decreased synchronisation accuracy within a range of milliseconds due to variable network latency. It is also recommended that switches supporting PTP be used in order to achieve accurate precision. Figure 2.7(a) shows a switch with a boundary clock that acts as a slave to the master clock; **ii)** using synchronous Ethernet (syncE) which passes timing from node to node within PHY layer using a high quality clock reference called primary reference clock to deliver frequency synchronisation with high accuracy (Figure 2.7(b)). Each device between the source clock and the end device needs to be upgraded with a synchronous Ethernet equipment clock; **iii)** combining PTP for time and phase synchronisation with syncE for frequency synchronisation which can lead to the best result according to [25] this combination can reduce time error from 119.25 nanoseconds to 700 picosecond.

- High latency: in the 5G context, 3GPP specifies requirements for this metric for 5G services in which end-to-end latency ranges from 0.5 milliseconds to

50 milliseconds [26]. For example, latency in the tactile internet requires end-to-end latency of less than 1 millisecond. Consequently, in order for the fronthaul to support tactile internet applications it needs to ensure that the latency requirement is met. The latency in Ethernet-based fronthaul can be reduced by using low latency switches and applying path management. In this respect, authors in [27] proposes latency aware path computation and packet forwarding schemes that enhance the performance of the Ethernet-based fronthaul enabling eCPRI traffic to be transported at tolerable latencies.

- High jitter: the queues in the switches can be expected to cause variance in the delay. The jitter issue in the Ethernet-based network can be reduced or even removed by using different techniques such as buffers and various scheduling techniques [28]. In [29] the authors proposed to use a gap-filling aggregator in their Cloud-RAN architecture with MAC-PHY split (option 6 in Figure 2.2) over Ethernet fronthaul. They obtained a jitter less than 100 nanoseconds by applying a gap-filling aggregator.
- Low reliability: for 5G reliability, the criterion can be relaxed or critical with a reliability of 99.9% or 99.9999%, respectively. It is necessary to satisfy these requirements because, if they are not met, they could have a negative effect on the quality of experience. The work in [30] proposed a multiple description coding approach to improve the quality of the signal received at the cloud in the uplink assuming multiple paths packet-based fronthaul. They validated the effectiveness of the approach in terms of increasing the sum-rate of the system through numerical results. However, this work is based on numerical results and does not offer any indication of the level of reliability achieved.

,

2.4.2 Functional Split over Ethernet Fronthaul: Related Works

To this end, the interesting research questions arises to evaluate the feasibility of the different levels of centralisation, also called functional split over Ethernet. This question was addressed in the literature from both theoretical and implementation perspectives.

In [31] the authors investigated the effect of various packetisation methods on the maximum number of supported DU for different Cloud-RAN splits. The

Matlab simulation results, where different functional splits were simulated, showed that there is a relation between packetisation overhead and latency, which affects fronthaul efficiency.

Using a real testbed, the work in [32] presented a Cloud-RAN architecture with functionality split option 8 using Ethernet fronthauling. The authors evaluated the latency and the throughput at receiver by continuously sending text messages in uplink. The authors analysed the impact of different system parameters on the latency and throughput performance for UL transmission and compared the performance of the Cloud-RAN with that of the legacy distributed RAN. Both TCP and UDP are used as possible fronthaul protocols for transporting IQ signals. Their results showed that TCP can provide the reliability required for fronthaul to ensure that Cloud-RAN achieves the same throughput as distributed RAN; however with the cost of increased latency.

Furthermore, work in [33] analysed the jitter produced in the switching nodes of the fronthaul network in a Cloud-RAN architecture with split option 7-b (see Figure 2.2). Then proposed rules on the dimensioning of the link size based on a number of assumptions in relation to the traffic structure.

Using a hardware testbed, authors in [34] implemented split option 7-a over Ethernet fronthaul using OAI. They showed that higher functional splitting leads to fewer functions in radio unit, thus a lower percentage of CPU usage but requires higher fronthaul throughput and vice versa.

To elaborate further in this direction, three functional splits, PDCP-RLC, MAC-PHY and intra-PHY splits, are implemented in a hardware testbed over an Ethernet-based fronthaul. The thesis goes forward by implementing 3GPP-based traffic models for 5G services, i.e. URLLC, mMTC and eMBB and evaluate the effect of the traffic on the choice of a split option.

In this thesis, 3GPP Cloud-RAN architecture; consisting of two entities, CU and DU, each of which can host any of the RAN functions; is adopted.

2.5 Function Split in Support of Beyond 5G Technologies

2.5.1 Dynamic Flexible Functional Split

Flexible functional split has brought flexibility to RAN deployment by flexibly split baseband functions between CU and DU. For simplicity, functional splitting can be configured statically by deploying a particular split. However, this architecture might not satisfy the various requirements of the applications to be served or may lead to a waste of RAN resources. Thus, one intuitive concept is dynamically configured/re-configured functional splitting to customise RAN resources in order to adapt to service requirements and current traffic conditions to improve flexibility. This contributes to the so-called **dynamic flexible functional split**.

Dynamic functional split can leverage softwarization and virtualisation to seamlessly migrate baseband functions between CU and DU without triggering downtime.

Work in [35] proposed an orchestration framework to dynamically and jointly select the appropriate user radio load and functional split that improves the throughput and reduces the cost of Cloud-RAN system. BBU functions are encapsulated in a containerised BBU to allow on-demand resource allocation over multiple sites. The authors focused on functional split options at user level.

Work in [36] proposed a flexible decision which selects the best functional level that minimises the sum bandwidth and total power under delay constraint. They considered the eight functional splits defined in Figure 2.2 in DL. Their simulation shows that the sum of bandwidth and total power were reduced by up to 40%. Work in [37] proposed a flexible 5G RAN scheme to optimise the usage of resources while lowering the overall cost of BBU pool in terms of storage and computing resources.

Another approach used in the testbed to assess dynamic functional split is the *replicate-based* approach in which certain baseband functions are replicated in the RRH. Authors in [38] provided a flexible platform that is based on replication-based approach where MAC and RLC functions are replicated in the DU to avoid migrating functions. They showed the feasibility of switching between two functional splits; PDCP-RLC and MAC-PHY (split options 2-6 respectively using 3GPP terminology) at run-time at the expense of packet loss or extra delay.

The work in [39] implemented two flexible functional split (option 8 and 7-a, see Figure 2.2 using hardware testbed. Their implementation support the full flexible functional split, i.e. using *replicate-based*, in order to be able to switch between the two function split options.

In this regard, in Chapter 4 of this thesis, three different functional splits are implemented in a hardware testbed to support the three 5G classes each with different requirements from latency and fronthaul throughput perspectives. The aim is to provide a platform for *service-delivery*.

2.5.2 Network Slicing

Cloud-RAN is envisaged as a promising solution for the wireless infrastructure to support the flexibility and high scalability of the network in order to keep up with traffic growth. Besides, RAN Slicing has emerged as an effective way to deploy at the same time diverse specifications for heterogeneous 5G networks. The purpose of RAN slicing is to guarantee service experience requirements and to sustain efficiency. This could happen by slicing RAN and transport network resources.

For example, work in [40] demonstrated how different use case can impact the selection of RAN functional split. The authors proposed a model for the efficient placement of Virtualised Network Functions based on slice requirements of different traffic classes. Work in [41] proposed a model to optimise the centralisation level and throughput by jointly considering RAN slicing and functional split. Authors in [42] incorporated URLLC and eMBB by slicing Cloud-RAN architecture aiming at minimising the total power consumption,

Work in [43] showed how the use of packet-based fronthaul allows the RAN to be sliced. It also showed that the use of packet-based fronthaul for network slicing can offer great advantages in terms of resource use.

In view of the possible benefits of RAN slicing, Chapter 5 of this thesis proposes two types of fronthaul resource sharing, namely non-orthogonal sharing and orthogonal sharing. In the former, all fronthaul resources are shared between the services. In the case of orthogonal sharing, a dedicated amount of resources is allocated to services. For instance, a percentage of bandwidth or a percentage of available path is allocated to a service to guaranty the required QoS. The goal is to determine how this form of orthogonality makes it possible to provide the necessary service according to the particular service. More to be followed in Chapter 5.

2.6 Fronthaul Key Performance Indicators and Measurement Methodology

2.6.1 Fronthaul Key Performance Indicators

Vertical industrial applications in 5G are classified into three categories: URLLC, eMBB and mMTC. While eMBB and mMTC can be seen as an extension of services already supported in 4G networks with high data rate and massive connectivity as main requirements, respectively, URLLC represent novel services with

Table 2.3 KPI on the fronthaul considered for the analysis.

KPI	Note
Latency	The maximum allowable latency as defined in the 3GPP for a given functional split is considered to be a threshold above which the solution is not appropriate..
Jitter	Analysing the jitter makes it possible to assess how consistent the latency results are and how stable the solution is. This is crucial for application with critical latency and jitter requirements.
Reliability	Reliability is defined as the probability of being able to transmit a packet over the fronthaul within a latency deadline. The reliability-latency trade-off is evaluated when a solution to improve reliability has been used.
overhead	The percentage overhead for different packet sizes is quantified and analysed to determine bandwidth efficiency.

unprecedented requirements. URLLC supports vertical industrial applications with ultra-low latency and high reliability, the data rate in URLLC is not expected to be very high; examples include industrial networking autonomous and assisted driving services, and tactile internet services. eMBB requires a high data rate and reliable broadband access across a wide region. Possible applications may include cloud gaming, virtual augmented reality among other applications. mMTC aims at providing access to a large number of devices with low reliability.

To verify whether the cloud-RAN system and the used transport technology could fulfil the 5G service requirements, different KPIs are evaluated. The kPIs considered in this thesis are summarised in Table 2.3.

2.6.2 Measurement Methodology for Fronthaul KPIs

The measurement methodology in this thesis is based on standards 3GPP and RFC. Measurements are carried out during the execution of the testbed. Once the measurements are taken, the KPIs can be determined as follows.

A) Latency

Latency can be measured as either one-way latency or round-trip time (RTT) as specified by IEEE RFC2544. To measure the latency on the fronthaul, packets transported on the fronthaul are continuously monitored for long-running connections to gather as many samples as possible.

- Round trip time (RTT) latency is measured to evaluate the effect of functional split and the effect of splitting tight interaction between different functionalities in RAN. RTT is considered as an accurate way of measuring the destination's processing time including switching, queueing, scheduling and transporting data [44].

  In this thesis, therefore, RTT is the time it takes for a request packet that is successfully confirmed by reception of data packet over the fronthaul. The procedure of measuring RTT latency is as follows [45]:

  - Record a timestamp and sequence number for each outgoing request packet,
  - Take the current timestamp, as soon as possible, upon the receipt of data packet,
  - Read sequence number of the received data packet,
  - Match the received data packet with its associated request packet using sequence number,
  - Calculate a time difference between the request and data packets.

  Focusing on the downlink, Figure 2.8 shows a simplified representation of events included in the measurement of RTT. Mathematically, RTT of one sample can be expressed as:

$$\text{latency(RTT)}_{\text{sample}} = \sum_{i=1}^{11} \text{latency(i)}, \tag{2.2}$$

  latency(1), latency(7) is the time that the fronthaul interface takes to packetise data, write packet descriptor onto network driver transmit queue.

  latency(2), latency(8) is the time taken for network driver to fetch the packet and launch it onto the fronthaul network. Since packets are fetched one by one from the transmit queue, this latency is subject to a queueing delay.

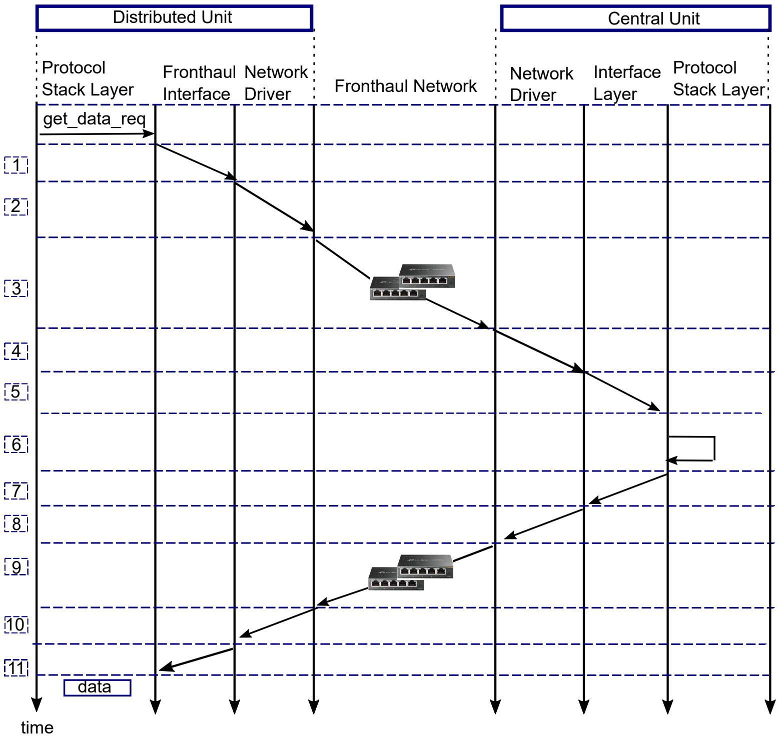


Fig. 2.8 Representation of the round trip time measurement.

latency(3), latency(9) is the time to transport data on the fronthaul network and is given by:

$$\text{latency}(3{,}9) = \sum_{i=1}^{\mathrm{N_{link}}} (\mathrm{t_{tran(i)}} + \mathrm{t_{prop(i)}}) + \sum_{i=1}^{\mathrm{N_{switch}}} (\mathrm{t_{proc(i)}} + \mathrm{t_{que(i)}}), \tag{2.3}$$

where $\mathrm{t_{tran}}$ is the transmission time that is the time taken to transmit a packet over a fronthaul link and defined as packet size over bit rate of the fronthaul link, $\mathrm{t_{prop}}$ is the propagation time that is the time taken for a packet to propagate between two nodes. The propagation time is calculated by the distance over propagation speed in the fronthaul. $\mathrm{N_{link}}$ is number of fronthaul segments between CU and DU, $\mathrm{t_{proc}}$ is the amount of time the switch requires to process a packet for routing (encapsulation/decapsulation and table lookup) and $\mathrm{t_{que}}$ is the queueing delay in a switch when there is a contention at its ports.

latency(4), latency(10) time taken for network driver to write packet descriptor in the queue. This latency is subject to a queueing delay when multiple data is sent over Ethernet sockets.

latency(5), latency(11) is the time that the fronthaul interface takes to read data from network driver queue and depacketise data by removing the Ethernet header from the packet.

latency(6) is the protocol stack processing time of the request at the DU to prepare data. This latency depends on the processing capability of central processing unit (CPU).

The average latency of RTT is computed as:

$$\mathrm{RTT_{aver}} = \frac{\sum_{i=1}^{N} \mathrm{latency(RTT)_i}}{N}, \tag{2.4}$$

where N is the number of RTT samples collected during execution of the experimental test.

- One-Trip Time (OTT) latency is the time it takes for a data packet to be successfully received by destination over the fronthaul. OTT latency of one sample can be expressed mathematically as:

$$\mathrm{latency(OTT)_{sample}} = \sum_{i=7}^{11} \mathrm{latency(i)}, \tag{2.5}$$

The average latency of OTT is computed as:

$$\mathrm{OTT_{aver}} = \frac{\sum_{i=1}^{N} \mathrm{latency(OTT)_i}}{N}, \tag{2.6}$$

where N is the number of OTT samples collected during execution of the experimental test.

B) Jitter

Jitter is the variation in the time between successive packets arriving. It is important for services with a strict latency budget. Jitter can be expressed as follows:

$$\mathrm{jitter_i} = \mathrm{latency(T)_{i+1}} - \mathrm{latency(T)_i} \ \forall i \in [1, N-1] \tag{2.7}$$

where $T \in \{\text{RTT, OTT}\}$

The average jitter is computed as:

$$\text{jitter}_{\text{aver}} = \frac{\sum_{i=1}^{N-1} \text{jitter}_{\text{i}}}{N-1} \tag{2.8}$$

C) Latency-Reliability

Data arriving at the Ethernet-based transmitter is stored in a transmit buffer, and the Ethernet driver serves the buffer in a first-in-first-out (FIFO) fashion. When data in the buffer cannot be served immediately it leads to a queueing delay, therefore the queueing model is designed to evaluate the performance of the Ethernet-based fronthaul.

Assuming that arrivals at fronthaul network follow a Poisson distribution, with average λ and that service time of each fronthaul path has an exponential distribution with mean $1/\mu$ seconds. Therefore, each fronthaul path can be modelled as M/M/1 queue following Kendall's Notation.

The average latency on each M/M/1 queue can be computed using Little's law as:

$$T = \frac{1}{\mu - \lambda} \tag{2.9}$$

In this thesis, latency-reliability of fronthaul path i is the probability that correct fronthaul transport of packet of B bytes occurs by a given deadline t. Accordingly, the probability of transporting a data packet of B bytes within a latency deadline t on fronthaul path i is the function $F(t, B)$ defined as:

$$F(t, B) = P(X \leq t) \tag{2.10}$$

The reliability can be found from the cumulative distribution function (CDF) of the latency and Equation 2.10 can be expressed as:

$$F(t, B) = 1 - P(X > t) = 1 - e^{-(\mu - \lambda)t} \tag{2.11}$$

Probability of error is defined as the complement of the latency-reliability. It is the probability that the time taken to transmit a data packet of size B from a source to a destination exceeds t (i.e. arriving late or being lost). Probability of error can be therefore expressed as follows:

$$\text{Probability of error}(t, B) = 1 - F(t, B) \tag{2.12}$$

2.7 Fountain Coding & Multiple Paths for Enhancing Reliability

Fountain coding is one of erasure coding methods widely used to improve reliability by adding redundancy to the data. Fountain coding is able to recover from a number of packet losses equal to the amount of redundancy introduced by coding.

Fountain codes are adopted by standards such as 3GPP Multimedia Broadcast Multicast Service [46] for broadcast file delivery and streaming services, and by the IETF RFCs [47]. They have been used in data storage applications [48], content distribution networks and collaborative downloading.

Fountain codes are rateless in the sense that they do not exhibit a fixed code rate but it is adapted to the channel conditions and adjusted in order to achieve the desired reliability even under heterogeneous channel conditions.

The original data is split into k equal blocks, and then coded into n using a (n, k) maximum distance separable code; this is referred to as (n, k) fountain code. One of the key advantages of fountain coding is the property that an (n, k) fountain code can be decoded if any k of the n transmitted symbols are received, i.e. the original k blocks can be recovered from receiving any k blocks out of n transmitted blocks.

Since the receiver needs to wait for the first k out of n blocks to be received to start decoding the blocks to retrieve the original data, we provide some background of finding the k$^{\text{th}}$ order statistic of n independent and identically distributed (i.i.d) random variables

Let X_1,X_2, ... ,X_n be n independent and identically distributed continuous random variables having a common density f and distribution function F. Let $X_{(k)}$ be the smallest k$^{\text{th}}$ of X_1,X_2, ... ,X_n.

Assuming X_i, $1 \leq i \leq n$ are exponential with mean $1/\mu$, then the expectation, the variance and the second moment of order statistic $X_{(k)}$ are defined respectively as:

$$E[X_{(k)}] = \frac{1}{\mu} \sum_{i=1}^{k} \frac{1}{n-k+i} = \frac{1}{\mu}(H_n - H_{n-k}) \tag{2.13}$$

and

$$V[X_{(k)}] = \frac{(H_{n^2} - H_{(n-k)^2})}{\mu^2} \tag{2.14}$$

and

$$E[{X_{(k)}}^2] = V[X_{(k)}] + E[X_{(k)}]^2 \tag{2.15}$$

with H_n being the generalised harmonic number defined as

$$H_n = \sum_{j=1}^{n} \frac{1}{j} \quad \text{and} \quad H_{n^2} = \sum_{j=1}^{n} \frac{1}{j^2} \tag{2.16}$$

Channel coding have been used for exploiting multiple interfaces such as Multipath Transmission Control Protocol [49]. The combination of channel coding and multi-path transmission are studied in number of experimental and simulation studies for improving system performance such as throughput and reliability.

In [50], a combination of diversity and network coding is used for improving throughput on the fronthaul link; and reference [51], which discusses how coded fronthaul transmission together with caching can reduce latency.

The work in [30] proposed a multiple description coding approach to improve the quality of the signal received at the cloud in the UL assuming multiple paths packet-based fronthaul. They validated the effectiveness of the approach through numerical results.

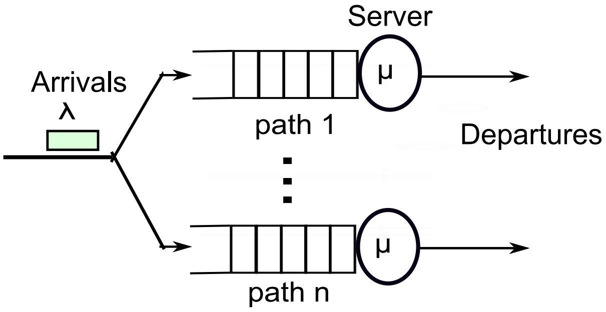


Fig. 2.9 n Parallel M/M/1 queues.

To this end, to achieve the stringent latency-reliability requirement of URLLC, it is proposed in this thesis to use a combination of channel coding and multi-path transmission. In this context, data is transmitted using fountain coding (n, k) over

n paths that can be assumed independent. Since each path is modelled as M/M/1 queue (see Section 2.6.2), the model, then, consists of n parallel homogeneous independent M/M/1 queues whereby incoming packets are encoded on arrival and serviced by n servers (see Figure 2.9). In this queueing model, the reliability is achieved as long as at least any k paths succeed in delivering the data within a latency deadline.

$$F(t) = \sum_{r=k}^{n} \binom{n}{r} F\left(t, \frac{B}{k}\right)^{r} \left(1 - F\left(t, \frac{B}{k}\right)\right)^{n-r}, \tag{2.17}$$

where F(t,B) is defined in Equation 2.10 and $k = 1...n$

Chapter 3

Evaluation of Packetised Fronthaul with MAC-PHY Split

3.1 Introduction

One of the architecture enablers of 5G is Cloud-RAN that is supported through multiple technological advances in the network including softwarization, virtualization and cloudification [39, 4]. Various advantages are enabled through centralisation of RAN functions including enhanced cooperative solutions, improved load balancing and RAN sharing, among others. On the other hand, it can introduce huge challenges in delivering low-latency services and requires high bandwidth availability to transport the baseband signals.

To address the extremely high data rate demand of CPRI, number of different techniques proposed in the literature. Compressed CPRI can be used to reduce capacity requirements in places where fronthaul faces bandwidth constraint. In this case, CPRI compression and decompression can enhance the utilisation of fronthaul link, up to three times [52, 53].

In addition to the throughput demands of CPRI, delay and jitter values must be kept to a minimum in order for a Cloud-RAN architecture to perform effectively. NGMN's guideline is to design a network such that the one-way latency is below 100 μs, and to maintain the frame delay variation effect.

Another approach to consider reducing bandwidth requirements is the use of alternative functional split options whereby some RAN functions may remain close to the DU, reducing the level of centralisation. However, such split of network functions depends largely on the availability of transport networks, and the final RAN configuration (i.e, distributed or centralised) will determine the level of cooperation and the deployment of some of the features being considered in the road to 5G. Linked to this, work in [54] surveys all transport network solutions available for fronthaul and discusses the impact of such technologies in the RAN context. In particular, how centralised RAN network functions impacts the service requirements but simultaneously impacts the level of cooperation among different transmitters.

On the other hand, there is a strong interest from telecom industry to leverage packet switched networks, such as Ethernet, in providing a cost-effective transport network solution for Cloud-RAN. This will allow the use of lower cost-industry standard equipment and sharing infrastructure already deployed for fixed networks. The most widely used transport protocol between the central entity and the remote unit is the CPRI, which has been specifically designed based on the requirements of digitised baseband signals. However, Ethernet is a best effort-based technology, and it is not designed to meet the low jitter and latency requirements for baseband signals transmission, i.e., CPRI. In this context, allowing for a higher layer split can allow the use of packet switched networks without degrading the overall RAN performance.

In this chapter, thus, we give some insights into the feasibility of MAC and PHY layer split, over the Ethernet. As latency and jitter are important requirements that play a key role in 5G, the foreseen latency and jitter for 5G services should also be provisioned over the fronthaul link. Therefore, in this Chapter we evaluate the two KPIs to show whether the Ethernet-based fronthaul can meet the requirements.

3.2 Experimental Testbed of Packetised Cloud-RAN

The experimental setup for the evaluation of the fronthaul for MAC-PHY split is depicted in Figure 3.1. The overall experimental testbed comprises an end to end LTE system from eNB to UE, and all functionalities of the protocol stack are implemented in the eNB as well as in the UE. The communication flow is shown in Figure 3.1, and a more detailed description is given in the following lines.

The experiment focuses on the DL direction, as the DL has strict latency requirement. IP packets are then injected in the eNB PDCP layer and are then handled by the whole protocol stack in eNB. In each sub-frame, the PHY layer in DU sends an indication to the scheduler function in the CU to prepare the DL data for transmission. The scheduler sends an indication to the PDCP layer which will

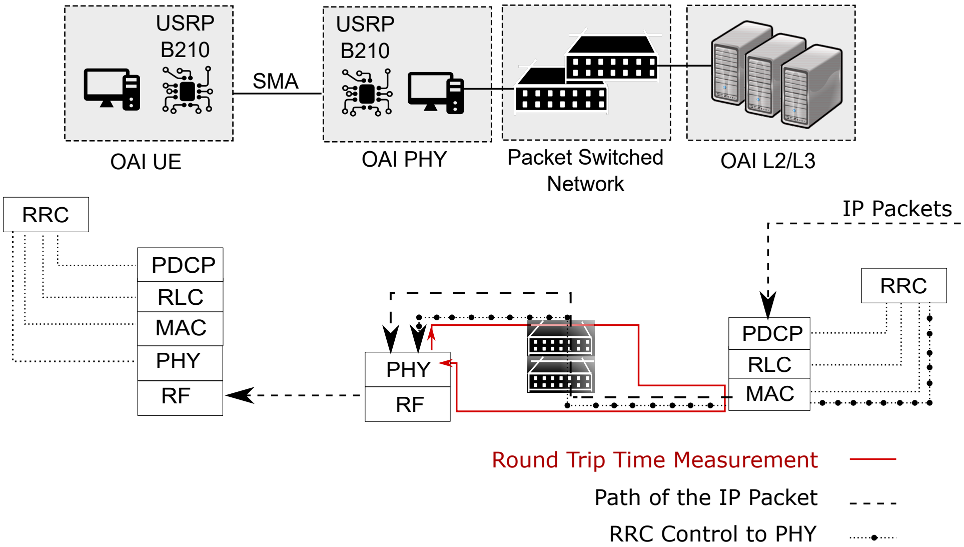


Fig. 3.1 End-to-end system experimental setup for MAC-PHY split over Ethernet fronthaul

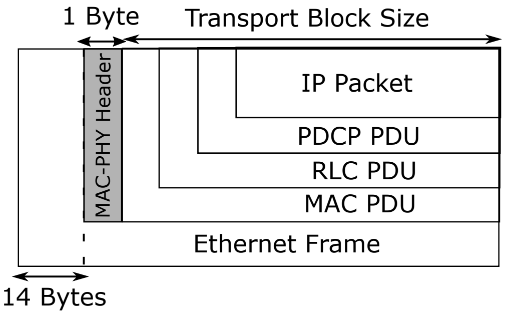


Fig. 3.2 IP packet encapsulation into Ethernet frame in CU protocol stack

fetch the IP packets and prepare the PDCP PDUs that are sent to the RLC. The RLC then informs the MAC layer of its buffer occupancy, the scheduler function decides how many bytes to get from the RLC buffer based on the Channel Quality Indicator stored for the UE. Once the MAC layer gets the specific number and size of RLC PDUs it composes the MAC PDUs. Afterwards, the Ethernet frame is composed by adding the MAC-PHY control header representing the message type (one byte) and layer 2 Ethernet MAC header fields (14 bytes consisting of source MAC address, destination MAC address, and packet type). The Ethernet frame is sent to network driver (see Figure 3.3) which attaches the preamble (7 bytes), start frame delimiter (1 byte) and frame check sequence (4 bytes) to compose Ethernet packet. The Ethernet packet is then transmitted to the DU via Ethernet.

Upon arrival, the DU de-packetises the Ethernet packet by removing the Ethernet and MAC-PHY control headers, and the PHY layer performs the Cyclic Redundancy Check and attaches the Cyclic Redundancy Check bits at the end of the transport block. After that PHY performs encoding, scrambling, modulation

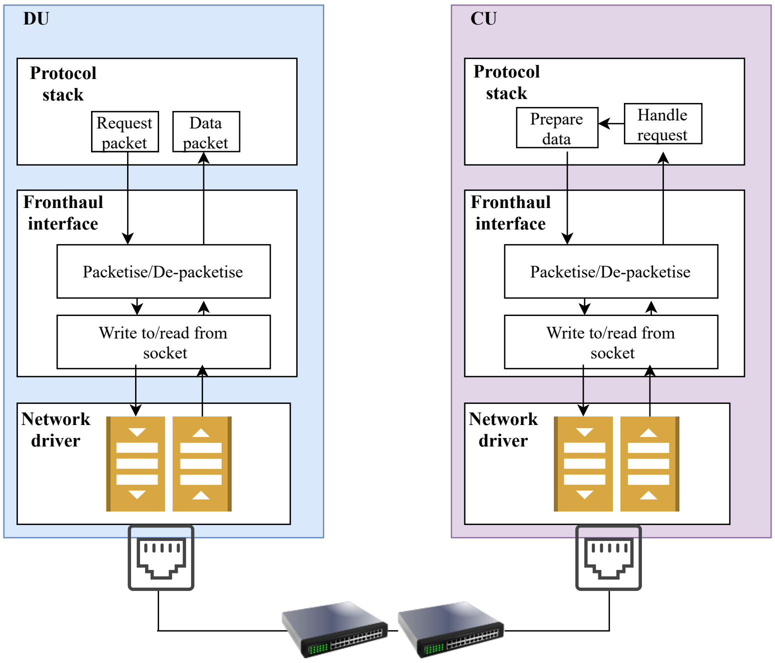


Fig. 3.3 Functional blocks involve in the round trip time measurement in the experimental. The orange arrows represent the latency.

and FFT functionalities. Then transmits the RF signal to UE, who handles the received DL data from PHY to PDCP in order to extract IP packets. Figure 3.2 shows a representation of the overall packetisation process.

The OAI UE is a fully compliant LTE UE based on the open source software implementation developed by the OAI community [55]. The software runs on an 8 gigabytes (GB) RAM with a Xeon 1220, 4 cores server and its connected via USB 3 (Universal Serial Bus 3) to a USRP (Universal Software Radio Peripheral) used to transmit and receive data. The OAI UE is attached to an OAI eNB (fully compliant LTE eNB) where the Cloud-RAN functional split takes place. As depicted in the figure, the OAI eNB is divided into two blocks, the DU corresponds to the OAI PHY block, where all the RF and PHY related functions take place. It runs on a PC with 4 GB RAM with an Intel core i5 with 4 cores. The second block corresponds to the CU, and it contains all higher layer functionalities (MAC, RLC and PDCP and layer 3 RRC), it runs on an 8 GB RAM with a Xeon 1220, 4 cores server. Both OAI eNB blocks are connected with Ethernet links with a capacity of 1 Gbps.

For the sake of completeness, Algorithm 1 shows the flow of the experiment running in OAI and the main configuration parameters are listed in Table 3.1. The experimentation is executed in sequential order, as shown in Algorithm 1. Once

Table 3.1 Open Air Interface (OAI) parameters

Parameters	Values
Carrier Frequency	2.68 GHz
System Bandwidth	5 MHz
Frame Type	FDD
Uplin Tx/Rx Antennas	1 Tx antenna / 1 Rx antenna
Tx Gain	100
Rx Gain	80
MCS	Adaptive Quadrature Amplitude Modulation (QAM): QAM, 16 QAM and 64 QAM)
Fronthaul Capacity	1 Gbps

UE is connected to the RAN, the DL IP packets are injected into PDCP in order to evaluate the fronthaul.

3.3 Analysis of MAC-PHY split

Based on the experimental setup described previously, we run a series of tests to study the feasibility of splitting the MAC and PHY layers with an Ethernet fronthaul network. The main focus of the experimental setup is to study the bandwidth, latency and jitter across the fronthaul network, and assess the suitability of Ethernet for such purpose. We also measure the impact of different packet sizes in throughput, latency and jitter.

Algorithm 1 Information Flow Between CU and DU

```
Inputs:
    CU and DU Initial Configuration
Run DU and CU
if Connection between CU and DU is established then
    eNB PHY → SS,MIB,SI
    Run UE
    procedure Cell Selection
        Band scanning
        UE Synchronization:
        UE ← SS,MIB,SI
    end procedure
    procedure Random Access
        UE → preamble
        UE ← Random Access Response (RAR)
        if Contention resolution is resolved then
            UE moves to Connected_mode
            procedure RRC Connection Reconfiguration
                Establish RAB
            end procedure
            Download IP Data to UE
            procedure KPI Measurements(Data)
                Measure latency of the fronnthaul
            end procedure
        end if
    end procedure
end if
```

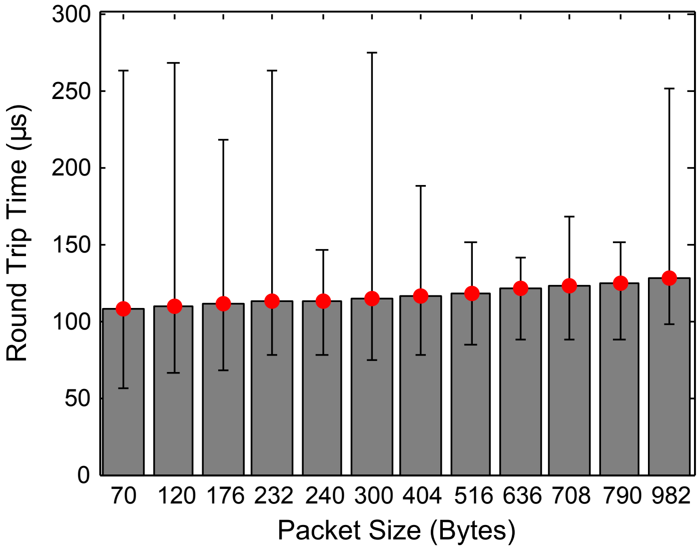


Fig. 3.4 Round trip time latency on the MAC-PHY split over Ethernet vs size of Ethernet packets.

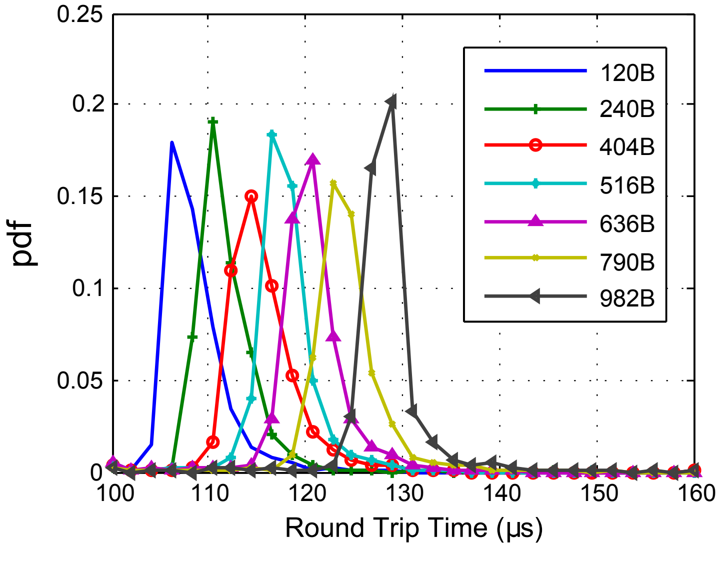


Fig. 3.5 Distribution latency on the MAC-PHY split for different size of Ethernet packets.

3.3.1 Analysis of Latency

The latency in this experiment is measured as the RTT of one packet from the MAC to the PHY layer, a graphical representation of the round trip measurement is given in Figure 3.1. The reason to measure RTT rather than a one-way latency is that the former can be measured more accurately in our setup; since the send and receive times are measured at the same physical location, the synchronisation between the two machines does not become a precision barrier.

RTT of one packet is measured according to Equation 2.2. A more detailed representation of the sequence diagram 2.8 is shown in Figure 3.3 which shows different functional blocks involved in the procedure of measuring the RTT.

The events involved in RTT measurement can be summarised as follows:

- Packetise/depacketise data and write/read the packet in/from the buffer of raw socket.
- Read/Write the packet from/to buffer and launch/get it to/from fronthaul network
- Network latency to transmit the packet.
- Software processing time of the request at the CU MAC to prepare data for DU PHY.

Here, the latency (2), (4), (8) and (10) in Figure 3.3 involves the delay caused by the kernel stack and the network driver which are subject to queueing delay. For example, when the network interface card receives Ethernet packets it stores them in a buffer waiting for further process. Then Dynamic Memory Allocation transfers the packet from the buffer to kernel space after which network interface card notifies kernel on the availability of packets in the buffer. Afterwards, the packets are copied from socket buffer to a user buffer.

The latency (3) and (9) in Figure 3.3 is the sum of the delay of transmission, the delay of propagation and the delay of waiting and processing in the two switches.

It is important to highlight that in this experiment in each TTI, only one packet is transmitted and therefore there is no queueing in the system. Therefore $t_{que}=0$ in Equation 2.3.

It is foreseen that transport block sizes will increase in 5G due to the inclusion of higher modulation order (up to 8 [56]). However, to support ultra-low latency applications in 5G, such as Tactile Internet [26], small packet sizes are expected. Figure 3.4 shows the RTT results obtained for different packet sizes. There is a slight, almost linear, increase of latency with the packet size, which is due to the fact that the MAC layer takes more time to prepare the data when higher packet size is used. Overall, the increase in latency is close to 20% from lower to higher packet size, 107.32 μs, for 70 bytes and increases to 128.18 μs for 982 bytes. Moreover, Figure 3.5 depicts the probability distribution function (pdf) of the experimental RTT, for the sake of clarity, some of the packet sizes have not been included in this figure. Despite the clear difference in average values, there is consistency as all distributions are within 1-2 μs deviation from the average value.

Based on the average result, Ethernet can meet the delay requirements of 250 μs which has been agreed so far in the community [2] and has been standardised in 3GPP (see Table 2.2)

3.3.2 Jitter Analysis

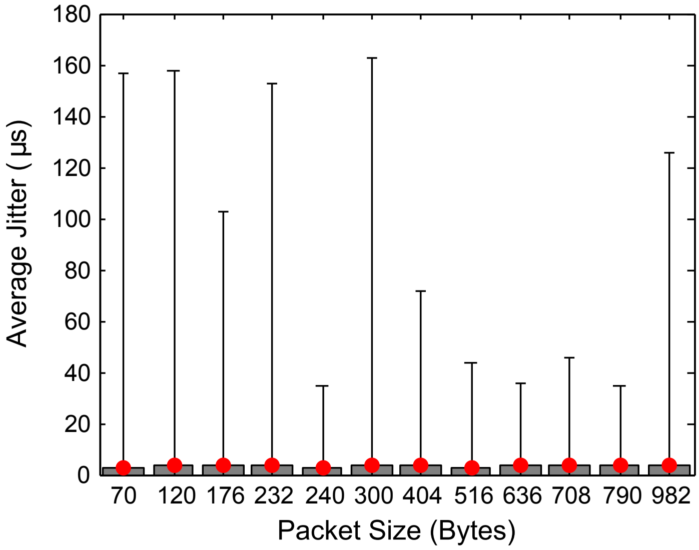


Fig. 3.6 Jitter on the MAC-PHY split over Ethernet vs size of Ethernet packets.

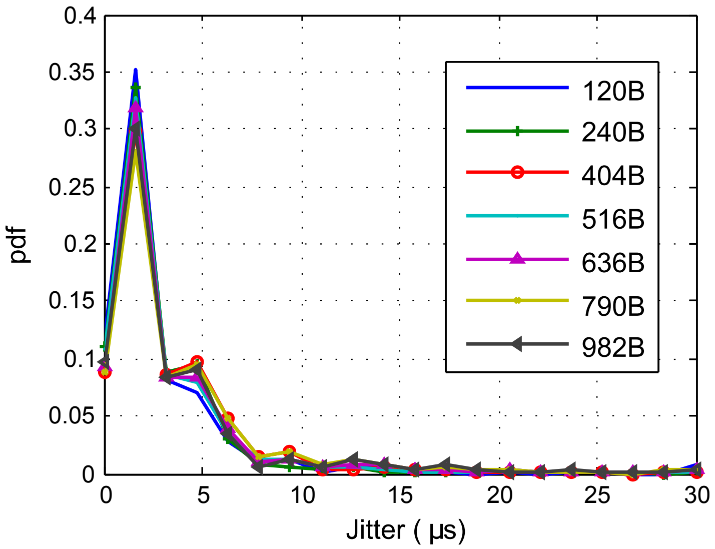


Fig. 3.7 Distribution jitter on the MAC-PHY split for different size of Ethernet packets.

Apart from the latency performance, jitter is another important limiting factor in any functional split. The jitter is introduced by computation, mainly due to the Operating System scheduler, which does not always respond in the same manner, even when considering low latency kernels. The transport network as well plays an important role in jitter. In our experimentation setup shown in Figure 3.1, the data goes through two switches, which will also introduce substantial latency variability.

The experimental jitter results are computed as defined in Equation 2.8 and shown in Figure 3.6; average results are almost equal for all packet sizes and close to a 3% of the average latency, however, maximum and minimum values span from 0 to 163 μs, which in principle gives the idea of high variability of values. Figure 3.7 shows the pdf of the experimental jitter samples, and it is shown that the probability of such extreme values is very low (close to zero) due to rare events or interrupts. Figure 3.7 also shows that the distribution is quite consistent despite the packet size. In particular, all cases analysed show similar average and distribution values. These results satisfy the jitter constraints of URLLC verticals as the results are within the range defined in NGMN, i.e. they remain below 160 μs [57].

3.3.3 Fronthaul Throughput Calculation

Finally, for each packet size, we experimentally calculate the fronthaul throughput. Figure 3.8 shows an example of an IP packet flowing down through the LTE protocol stack where each protocol layer adds its own header to the data units.

The size of the protocol header added can vary with each transmission time. It depends, for example, on the type and number of control data added by the protocol layers. Therefore, to calculate the percentage overhead, we consider the size of the transport block and not the size of the IP packet. In this case, the overhead is the MAC-PHY control header (1 byte) and the Ethernet header, which is set to 26 bytes as illustrated in Figure 3.9.

Table 3.2 shows throughput and percentage overhead in the fronthaul links, considering the packet description given in figure 3.2. The percentage of overhead decreases with the Ethernet packet size, as the 27 bytes overhead is fixed regardless of the Ethernet packet size.

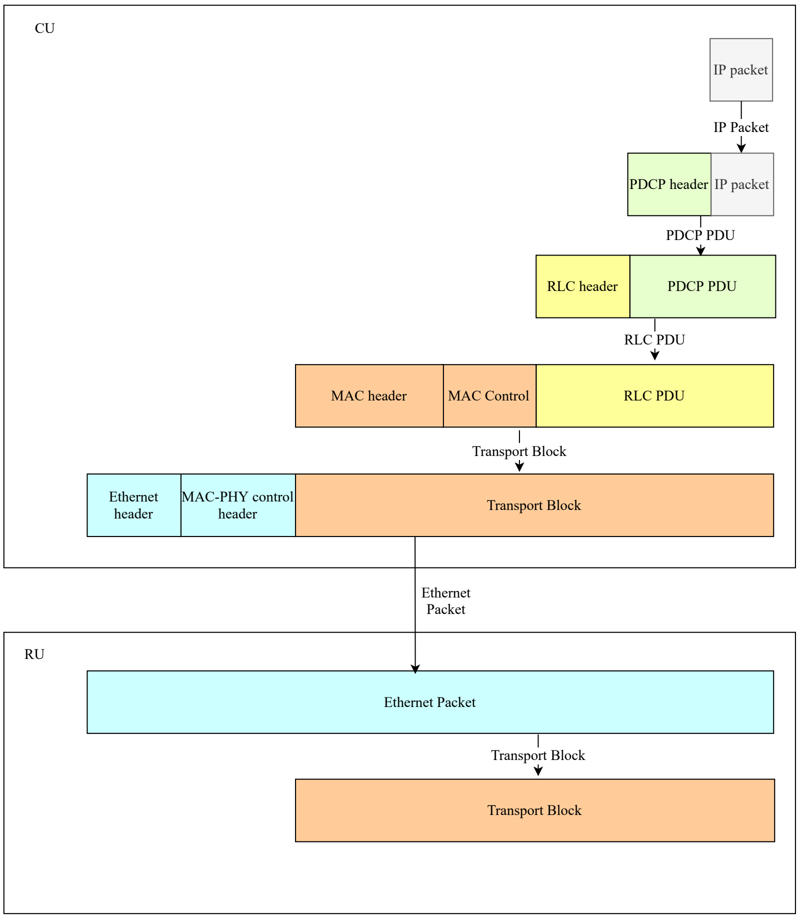


Fig. 3.8 Flow of IP packet through LTE protocol stack in Downlink

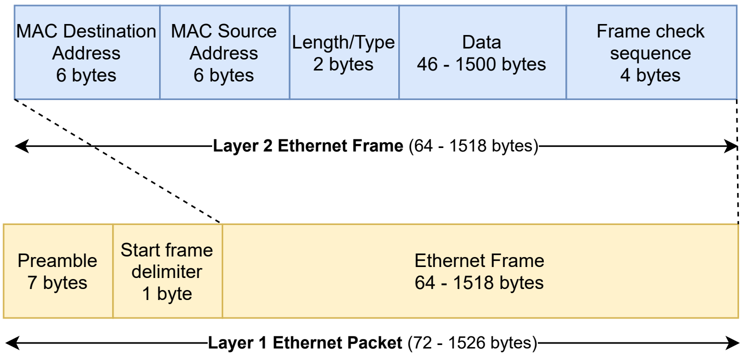

MAC Destination Address 6 bytes
MAC Source Address 6 bytes
Length/Type 2 bytes
Data 46 - 1500 bytes
Frame check sequence 4 bytes
Layer 2 Ethernet Frame (64 - 1518 bytes)
Preamble 7 bytes
Start frame delimiter 1 byte
Ethernet Frame 64 - 1518 bytes
Layer 1 Ethernet Packet (72 - 1526 bytes)

Fig. 3.9 Ethernet packet structure.

Table 3.2 Throughput and Percentage Overhead Calculation on Ethernet Fronthaul

Transport Block Size (bits/bytes)	Fronthaul Packet Size (bytes)	Percentage Overhead %	Fronthaul throughput (kbps)
440 / 55	82	32.927	656
840 / 105	132	20.455	1,056
1288 / 161	188	14.362	1,504
1736 / 217	244	11.065	1,952
1800 / 225	252	10.714	2,016
2280 / 285	312	8.654	2,496
3112 / 389	416	6.490	3,328
4008 / 501	528	5.114	4,224
4968 / 621	648	4.167	5,184
5544 / 693	720	3.75	5,760
6200 / 775	802	3.367	6,416
7224 / 903	930	2.903	7,440
7736 / 967	994	2.716	7,952

3.4 Impact of Experimental Parameters on Fronthaul Latency Budget

This chapter focuses on latency and jitter, which are among the main KPIs in 5G. Applicable to this experimental setup, from the perspective of KPI validation, the two metrics are used to test the results obtained against the target value and to assess the deterministic of the Cloud-RAN system being deployed. However, from the performance point of view, latency is the bottom line performance metric for the Cloud-RAN system. As latency has a direct impact on the stability of the system and it is most critical for success in the execution of the experiment. During the experiment, it was found that when the RTT latency reaches 300μs, the UE loses synchronisation with the network causing the RRC connection to be released, which requires the user to initiate a new connection.

Since latency has the largest effect on Cloud-RAN performance, the aim of this section will be to analyse the effects and contributions of the different parameters of the system on latency.

Various parameters can have an effect on latency performance, such as average packet size, arrival rate, fronthaul capability, switch speed and fronthaul network topology, i.e. fronthaul length, number of switches and number of DUs connected to a CU as well as the environment in which the experimental is executed.

In this section, we report some additional results on latency and jitter, highlighting in particular how packet size and traffic load affect fronthaul efficiency and light shedding in a test environment that can reduce latency and jitter.

3.4.1 Impact of Packet Size

Analysis in Section 3.3.1 revealed the effect of average packet size on average latency. It is shown that the average latency increases with the size of the packet, as there are more bits to be processed.

Further analysis of latency is carried out in this section, taking into consideration not just the average RTT latency but also the 25th and 95th percentiles of latency. Latency is evaluated for scenarios with packet sizes of 32 bytes, 300 bytes, 500 bytes and 1500 bytes to cover URLLC, mMTC and eMBB classes.

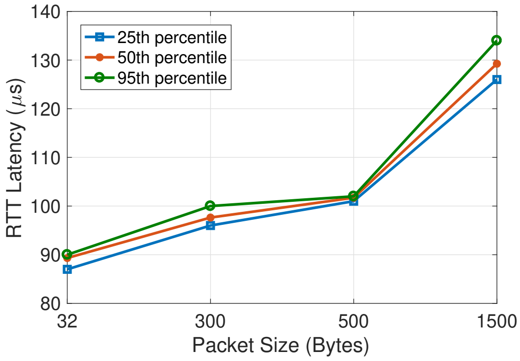


Fig. 3.10 Percentile of latency for different packet sizes on fronthaul.

Figure 3.10 shows the 25th, 50th and 95th percentiles of the latency and jitter for different packet sizes. It can be observed from Figure 3.10, the percentile latency increases with the increase in packet size as it requires more time to process it. The average and 95th percentile values do not vary significantly for packets of 32 bytes, 300 bytes and 500 bytes (the gap is < 3 μs). Whereas for packet size of 1500 bytes the difference between the average and 95th percentile values becomes slightly wider (the gap is 5 μs). Therefore, packet size of 1500 bytes has a longer tail of latency values. It is also noted that the 95th percentile latency remains below the acceptable latency on the fronthaul for MAC-PHY split for different packet sizes.

The size of the TB transmitted in MAC-PHY split over the fronthaul is determined by the available data to be transmitted, the channel quality indicator and the number of available resource blocks. The scheduler chooses the most appropriate MCS based on the reported channel quality indicator, and then calculates the transport block size that is a function of the MCS and the number of resource blocks which depends on the transmission bandwidth. As a result, increasing transmission bandwidth leads to a higher number of resource blocks that will allow experimenting with larger packet sizes.

3.4.2 Impact of Traffic Load

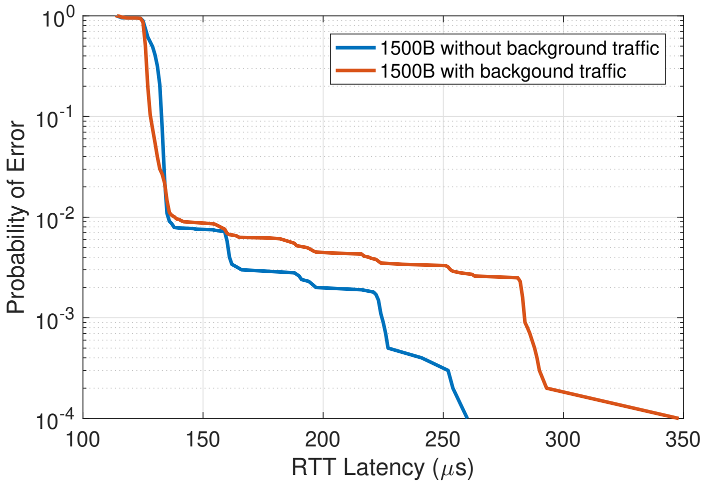


Fig. 3.11 Distribution latency on MAC-PHY split for packet of size 1500 bytes with and without background traffic.

Different fronthaul traffic load conditions my have different impact on the performance of the fronthaul in terms of latency. Traffic load is determined by a number of factors, such as packet size and arrival rates of packets, number of aggregated flows, and so on. In the case of a transport network with a higher traffic load, a queuing will be more likely than a transport network with a lower traffic load. Therefore, packets may experience different latencies depending on traffic load.

In this section, we evaluate the effect of mixing fronthaul and background traffics through experimentation. For this experiment, a payload of 1500 bytes is generated every 10 ms and added to the testbed described in Section 3.2 as background traffic. The worst-case scenario occurs when the two fronthaul flows, which are directed to the same destination, arrive at the Ethernet switch at the same time causing queueing delays.

Figure 3.11 shows the probability of error as defined in Equation 2.12 vs RTT latency with and without background traffic to show the effect of increasing traffic loads. RTT latency includes processing delays, queuing delays that are considered as random component for latency [58]. Hence, the probability of error shows the impact of the random components on the latency distribution and, more precisely, on its tail.

The Figure 3.11 shows that fronthaul can achieve a lower probability of error at lower latency when there is no background traffic. For example, when background traffic is introduced, fronthaul latency can experience an increase of 34.6% relative to when background traffic is not added at the error probability of 10^{-4}. This increase in latency occurs because the addition of background traffic increases the waiting time at the Ethernet switch.

3.4.3 Impact of Fronthaul Topology

The analysis provided in Section 3.3 is based on a transport network topology consisting of two switches operating at 1 Gbps and two Ethernet segments each with a length of 3 metres and a capacity of 1 Gbps.

The latency would be affected by the number of switches in the fronthaul network, switching capacity, Ethernet capacity and Ethernet length as shown in Equation 2.3. More importantly, if multiple DUs are connected to a CU through an Ethernet switch, there would be queueing due to statistical multiplexing that would cause non-deterministic latency. It would be beneficial to use network management in such a network topology.

3.4.4 Impact of Testbed Environment

In this thesis, the experimental is running over general purpose processors. The OAI software instance of the DU and CU run in Linux-based operating on Ubuntu 14 with Linux 3.9 kernel in two separate machines. The two machines consist of 4 cores with Intel i5 for DU and Intel i7 for CU and processor speed of 3.6 GHz.

The environment in which the experimental is being performed can have a significant impact on latency and jitter performance. Performance can, for example, be affected by the type of hardware and its specifications, such as the number of CPU cores available, the storage capacity, the frequency of the processor.

The performance can also be influenced by modes of operation such as the way in which memory and external interfaces are accessed, whether function blocks are sequentially or in-parallel processed.

For services with extreme latency constraints, it may be useful to use the following:

- Data Plane Development Kit for fast packet processing leading to networking latency optimisation.
- More powerful processor that will improve latency by reducing processing time.
- Larger number of cores and dedicate as many cores as necessary for processing.
- Dedicated hardware accelerator to which RAN functions that are computational intensive are offloaded for acceleration.
- Newest version of the Linux kernel which might provide improvements to the kernel's design.
- Parallel processing when possible is a fundamental principle to benefit from the available computing resources, which can reduce the processing time.

3.5 Concluding Remarks

In this chapter, we examine how using Ethernet as fronthaul can work in Cloud-RAN, with focusing on the MAC and PHY split. In this context, we setup a hardware-based experimentation platform and analyse latency and jitter experience with packetising data and fronthauling over the Ethernet. We can show that such split is feasible over the Ethernet as the latency results for different packet sizes are compliant with the requirements of 250 μs set by 3GPP (see Table 2.2). More specifically, the results obtained depend on the test scenario, which relates to the network configuration parameters, the environmental characteristics and the system traffic load.

In addition, the analysis shows that the throughput of the fronthaul scales with the user data, therefore MAC-PHY split has the advantage of not being directly affected by some of the 5G technologies such as massive number of antennas, i.e. massive MIMO.

Chapter 4

Flexible Functional Split for 5G Services over Ethernet Fronthaul

4.1 Introduction

Having demonstrated the feasibility of MAC-PHY split (option 6 according to the terminology in Figure 2.2) in Chapter3, this chapter upgrades the testbed developed in Chapter 3 to support more functional splits and also to model 3GPP-based traffic for 5G services, i.e. URLLC, eMBB and mMTC.

To simultaneously support URLLC, eMBB and mMTC, with their heterogeneous requirements, in a dynamic and on-demand manner, flexible deployment solutions are needed where functionalities can be moved across the network according to service requirements. Such flexibilities could be dynamic placement of network functions across the network through Network Function Virtualisation or slicing network to end-to-end separate instances each addressing different requirements [59, 22]. In this direction, flexibility in RAN has been discussed in the context of Cloud-RAN by splitting RAN functionalities between central, cloud-based unit and distributed unit, to achieve advantages such as cooperative solutions, improved load balancing and RAN sharing. The split could be pre-defined for different network slice, or it can be dynamically changed for different types of traffic or depending on the network conditions, which could be configured via top-level network controller and offered as-a-service. Studies on different functionality and how it can be offered as-a-service, and controlled by top-level network controller is presented in [60].

The choice of how to split RAN functions depends on a variety of factors, such as availability of RAN resources, fronthaul types, network deployment scenarios and the QoS requirements of the service. This chapter, then, analyses how requirements of the 5G traffic classes are met by providing different split between CU and DU. The analysis are performed through an experimental testbed using software defined radio (SDR) and the OAI [55]. The three implemented splits are options $-a7$, intra-PHY, option 6, MAC-PHY, and option 2, PDCP-RLC. The pros and cons of each split in terms of load on the fronthaul for each service under consideration, are then thoroughly studied in this platform.

4.2 Cloud-RAN and Various Layer Split

Functional split between CU and DU are adopted to address various challenges of radio access networks. As mentioned earlier, the split point for legacy Cloud-RAN is very close to the radio using CPRI. However, eight different options for such split are defined. Among all available splits, we will focus our attention on PDCP-RLC, MAC-PHY and intra-PHY splits (respectively, options 2, 6, and 7 when referring to Figure 2.2), which are mostly considered according to [61]. The pros and cons of the three splits under consideration in this Chapter are summarised in Table 4.1.

4.2.1 PDCP-RLC Split

In PDCP-RLC split (option 2 when referring to Figure 2.2), RRC and PDCP are executed in the CU while RLC, MAC, PHY and RF are executed in the DU. Having a MAC layer in the DU, fast HARQ can be achieved.

PDCP is responsible for header compression, ciphering, integrity protection and delivering DL processed control and user data to RLC in the form of PDCP PDU as shown in Figure 4.1. PDCP delivers PDCP PDUs to RLC once it processes the RRC or IP packets. Since there is no concatenation function in PDCP, if PDCP receives multiple packets from GPRS Tunneling Protocol, it sends more than one PDCP PDU to RLC via the fronthaul; as a consequence, the fronthaul traffic load increases with control- and user-plane (CP and UP, respectively) traffic. As there is one PDCP entity for each radio bearer, PDCP-RLC split is thus *bearer-based*.

In this approach, it is possible to distinguish between CP and UP traffic. Therefore, the former may be prioritised over the latter in the case of high traffic volume and limited fronthaul capacity. Furthermore, PDCP may use QoS applied to each Radio Access Bearer to ensure priority-based treatment of packets on the fronthaul.

Table 4.1 Cloud-RAN Functionality Splits: Pros and Cons

Split	Pros	Cons
PDCP-RLC	• HARQ is in DU enabling fast retransmission • fronthaul network can handle traffic from different bearers with different priorities • fronthaul data rate scales with the user data • Low overhead on the fronthaul	• Only RRC and PDCP are centralised • fronthaul traffic grows with UP/CP traffic load for each bearer
MAC-PHY	• L3 and L2 are centralised allowing coordinated scheduling • Multiplex data from different bearers into one TB • fronthaul data rate scales with the user data	• HARQ in CU may be challenging to meet HARQ time requirement • overhead on the fronthaul depends on PDCP, RLC, and MAC headers and the RB size
intra-PHY	• Architecture is closed to full centralisation • Coordinated scheduling and joint processing are possible	• fronthaul load increases with the bandwidth, number of sectors and antennas • Higher latency as the DU has to receive all packets related to resource elements before starting the inverse FFT

4.2.2 MAC-PHY Split

In the case of MAC-PHY split (option 6 when referring to Figure 2.2), PHY is located in the DU while layer 2 and 3 are centralised. MAC multiplexes MAC Service Data Units from one or different logical channels onto TBs then delivers the TBs to PHY. The TB size depends on scheduling decision which considers RLC buffer occupancy, the available bandwidth and selected modulation scheme. As a consequence of MAC multiplexing, the data delivery to PHY via fronthaul is taken per UE and not per radio bearer as in PDCP-RLC split case.

Unlike in PDCP-RLC split where HARQ is located in DU, in MAC-PHY split the HARQ is centralised. Hence this split option is more latency constrained, compared to PDCP-RLC split. In fact, there is a strict requirement of 4 ms HARQ response time set by 3GPP [62] for LTE. This requirement will further be restricted in 5G. This aspect may become challenging when considering high-latency or high-loaded fronthaul networks.

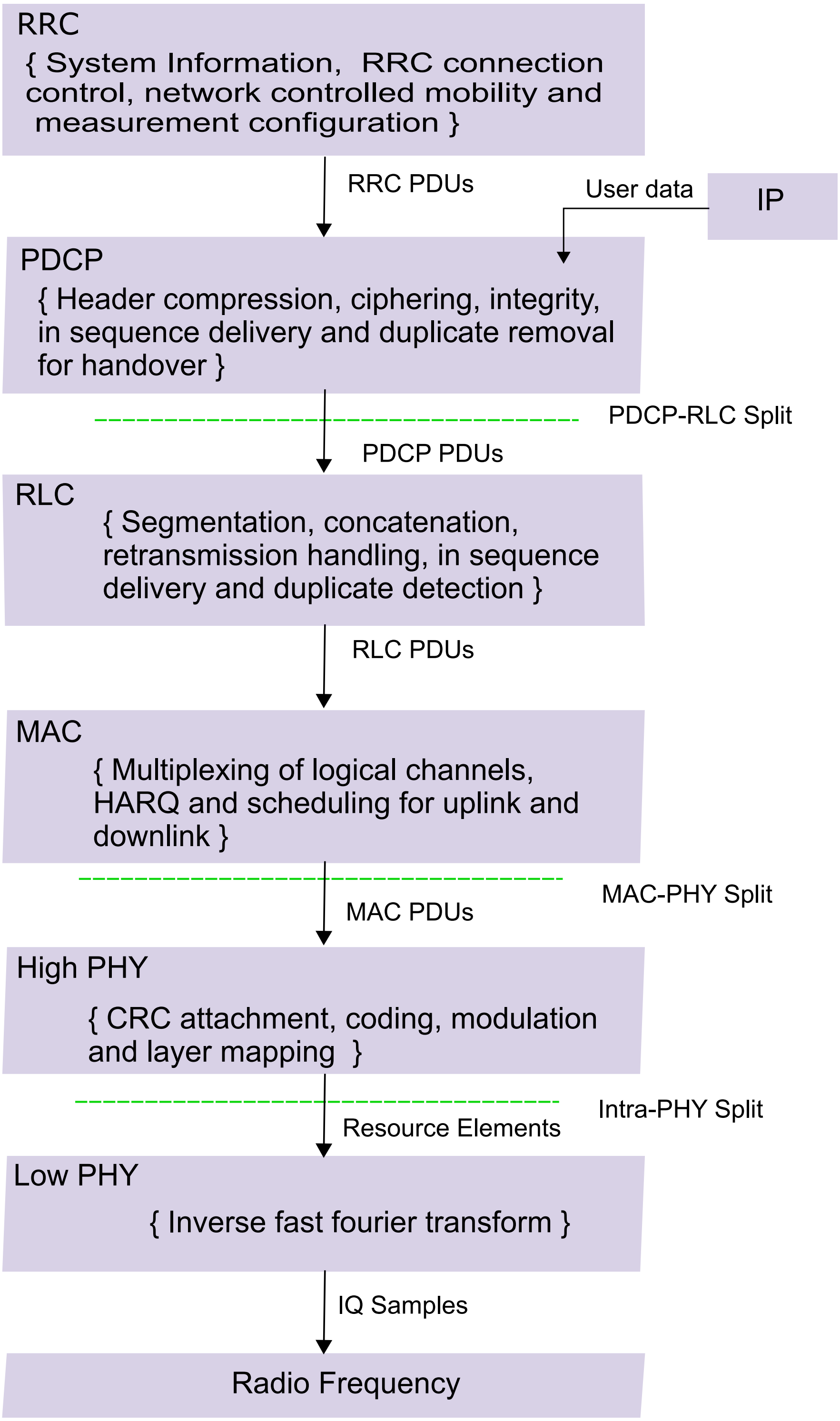

RRC
{ System Information, RRC connection control, network controlled mobility and measurement configuration }
RRC PDUs
User data
IP
PDCP
{ Header compression, ciphering, integrity, in sequence delivery and duplicate removal for handover }
PDCP-RLC Split
PDCP PDUs
RLC
{ Segmentation, concatenation, retransmission handling, in sequence delivery and duplicate detection }
RLC PDUs
MAC
{ Multiplexing of logical channels, HARQ and scheduling for uplink and downlink }
MAC-PHY Split
MAC PDUs
High PHY
{ CRC attachment, coding, modulation and layer mapping }
Intra-PHY Split
Resource Elements
Low PHY
{ Inverse fast fourier transform }
IQ Samples
Radio Frequency

Fig. 4.1 LTE protocol stack with functions for each protocol layer.

4.2.3 Intra-PHY Split

In intra-PHY split (option 7-a when referring to Figure 2.3), PHY functionality is split between CU and DU. The inverse FFT is performed in DU while other functionalities are performed in CU. Therefore, more functionalities are centralised compared to PDCP-RLC and MAC-PHY splits.

The fronthaul transports resource elements in the usable bandwidth, in which case the capacity of the fronthaul is independent of the actual user traffic. The load on the fronthaul has constant data rate and hence there is no multiplexing gain to be achieved. Since the fronthaul transports resource elements across all the configured orthogonal frequency division multiplexing (OFDM) symbols, the fronthaul transmission time granularity can either be symbol or subframe.

In our experimentation, the transmission per symbol is considered to avoid sending too large packet. In this approach, 14 packets are transported in total in each subframe, as there are 14 OFDM symbols in one subframe. In our experimentation, the size of each packet is ($12\ \times 2 \times N_{\mathrm{RB}}$) bytes, coming from the fact that 1 byte used to code one sub-carrier (thus, 12 bytes as there are 12 sub-carriers), 2 is because of the I/Q modulation (1 byte for I and 1 bytes for Q), and N_{RB} represents the number of resource blocks (RBs) the channel bandwidth is composed of.

In this experiment, the transmission bandwidth is set to 5 MHz corresponding to $\mathrm{N}_{\mathrm{RB}} = 25$ but increasing the bandwidth to 20 MHz increases N_{RB} four times and thus increases the fronthaul bandwidth demand for intra-PHY split four times. In addition, our experiment is based on 1 transmit antenna and 1 spatial layer, and increasing any of these increases the fronthaul bandwidth demand for intra-PHY split with the same factor. Further analysis of the fronthaul data rate in 5G is included in Section 4.5

4.3 Experimental Setup

Figure 4.2 shows the experimental setup to evaluate the performance of Cloud-RAN functionality splits using OAI [55].

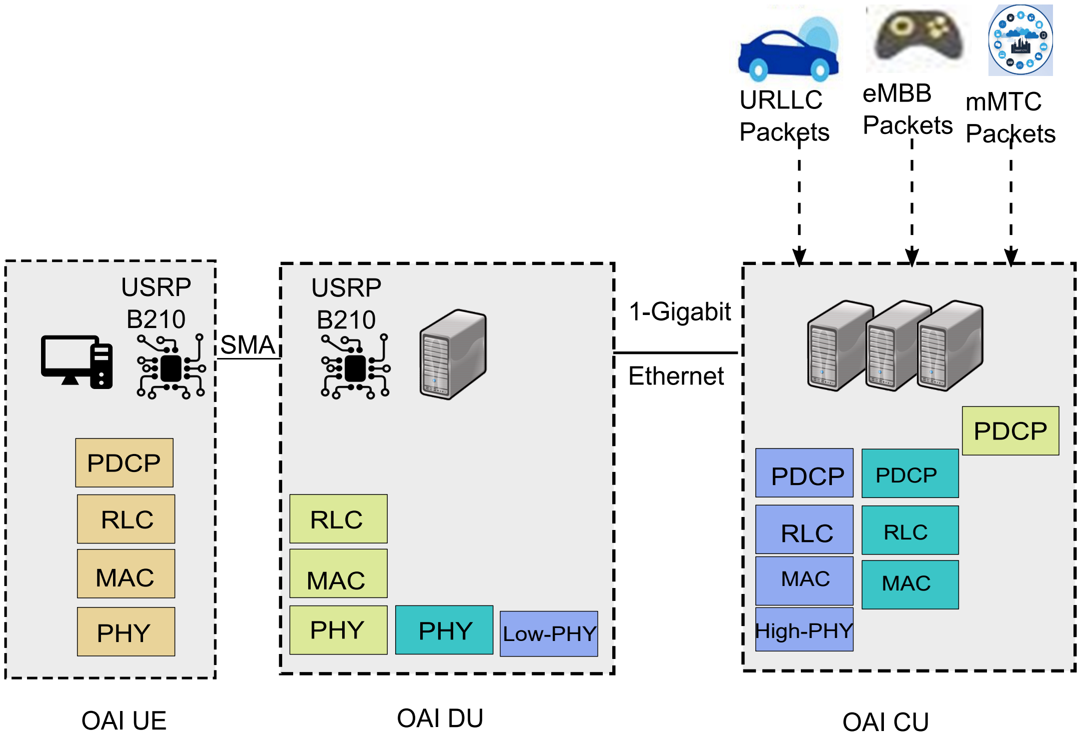


Fig. 4.2 Setup of the testbed platform for functional split over Ethernet.

The experimental testbed consists of an OAI UE based on the LTE OAI implementation and CU and DU whereby the Evolved NodeB (eNB) functionalities are implemented. The UE runs on a PC (4 GB RAM with an Intel core i5). The CU and DU run on two separate servers (8 GB RAM with a Xeon 1220, 4 cores), connected through an Ethernet link with capacity 1 Gbps[1]. Two USRPs are used to transmit/receive data between UE and DU. The USRPs are connected to their relevant machines via USB 3.

Radio parameters are listed in Table 3.1. The focus of our experimentation is on DL direction. We conducted various experiments to study the impact of the three splits discussed in Section 4.2 while running the fronthaul over Ethernet. The functions are shifted between CU and DU according to the functionality split to be evaluated. Once the UE is connected to the RAN and the Radio Access Bearer (RAB) is established successfully, the IP packets are injected in the CU on top of PDCP.

Traffic pattern for URLLC considers an industry-based closed-loop application [63] and it is modelled with 1000 UEs receiving packets (e.g., commands for

[1] In our experimentation, PDCP-RLC and MAC-PHY splits work also when connected through a switch, while the intra-PHY split does not support this setup due to time constraints. For the sake of uniformity, we thus used a direct Ethernet link to connect the CU and the DU for all the different splits.

actuators) from the network with a beta distribution (i.e., higher number of UEs to be served at the same time compared to mMTC) within a 10 s time interval with packets of 500 bytes [64]. We consider also two additional services as discussed in [63]: *(i)* eMBB, modelled by considering 10 simultaneous active UEs per cell with full buffer bursty traffic FTP model 3 [65] and IP packets of 1500 bytes; *(ii)* mMTC, modelled with 24000 UEs per cell uniformly accessing the network within a time interval of 60 s [66] with a packet size of 300 bytes [64] in order to model the reception of an acknowledgement following a report transmission in the UL.

Table 4.2 Traffic parameters for 5G services

Type of traffic	eMBB	URLLC	mMTC
Packet Size (Bytes)	1500	500	300
λ (packet/ms)	4	1	4

We used the experimental setup as shown in Figure 4.2 to test the different splits discussed in Section 4.2 for URLLC as well as eMBB and mMTC. In order to evaluate the performance in terms of introduced latency as well as jitter for each split. We performed other series of tests where CU and DU are located at the same machine. This was necessary in order to avoid synchronisation issues between two separate machines. In this case, network latency between the functions of the split was emulated by considering the latency figures achieved in the first set of tests, where latency due to physical transmission over Ethernet cable was taken into account.

4.4 Evaluation of Functionality Splits for Different 5G Traffic Classes

In this Section, we analyse the performance of the different splits discussed in Section 4.2 for URLLC, eMBB and mMTC.

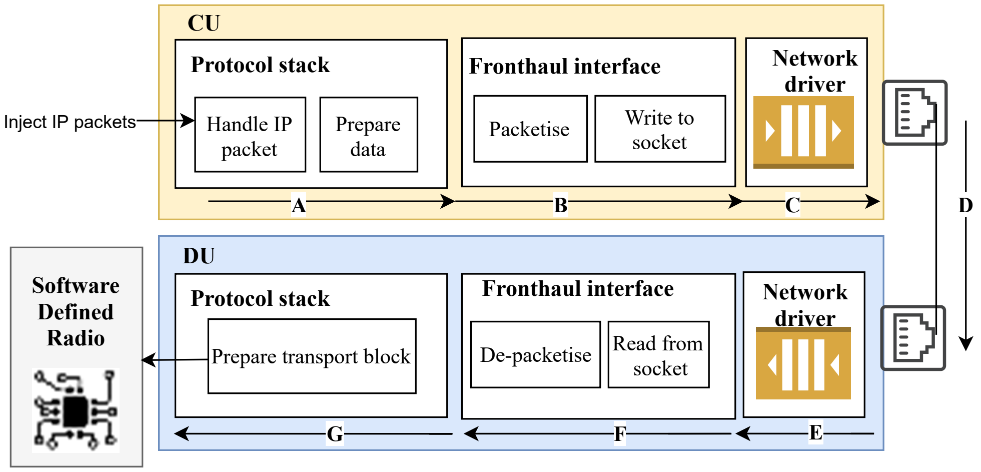


Fig. 4.3 Functional building blocks of Cloud-RAN with functional split over Ethernet and representation of latency for an IP Packet.

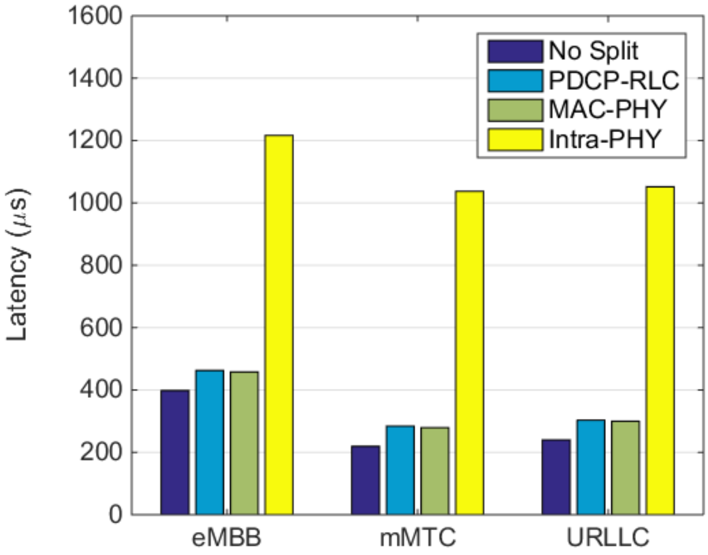


Fig. 4.4 Latency for an IP packet from when it is injected to PDCP layer to when it is transmitted to the UE.

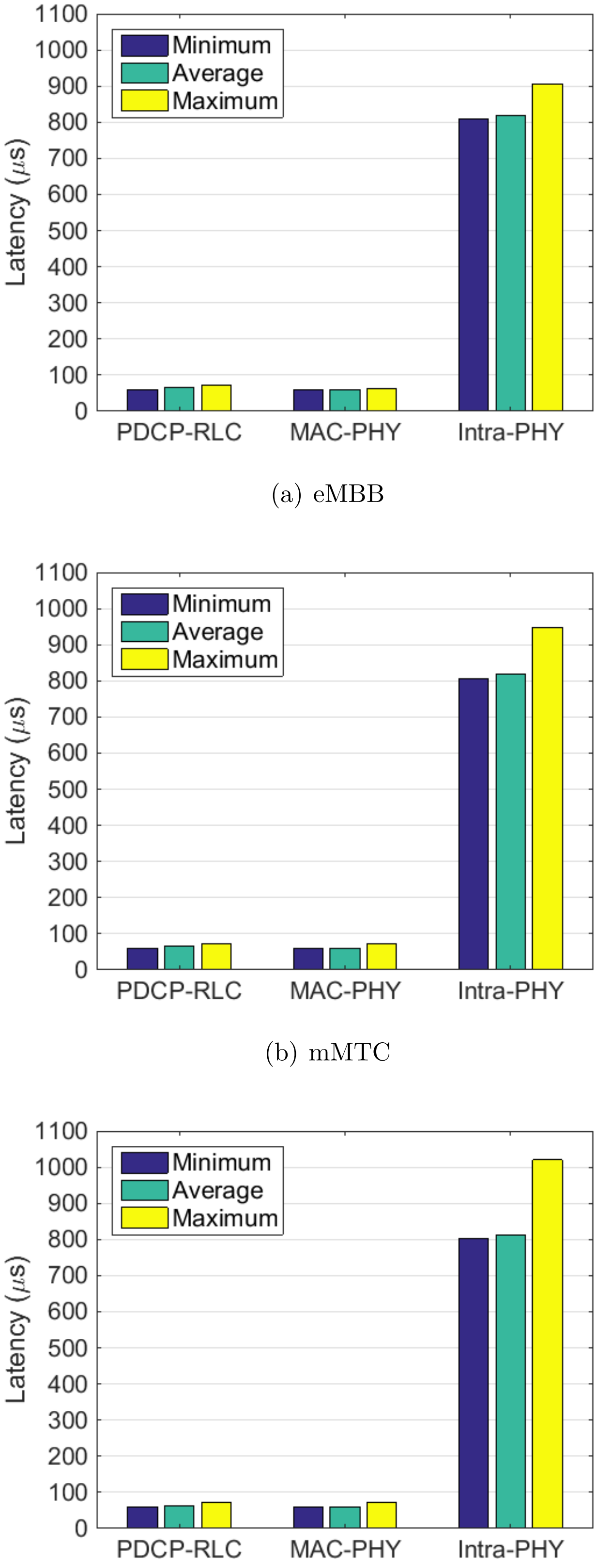
Figure 4.3 shows latency budget for an IP Packet from when it is injected to PDCP layer to when it is transmitted to the UE. The budget latency consists of latency(A) which is CU processing time, latency(B), latency(C), latency(D), latency(E) and latency(F) corresponding to latency(7), latency(8), latency(9), latency(10) and latency(11) respectively in Figure 2.8, and latency(G) which is DU processing time. Therefore the budget latency of IP packet can be expressed as $\sum_{i=A}^{G}$ latency(i)=latency(OTT) + latency(A) + latency(G), where latency(OTT) is defined in Equation 2.6.

Figure 4.4 analyses the total latency from PDCP to PHY for UP packets. The aim is to compare the latency introduced by the different functionality splits in addition to the legacy latency introduced by the protocol stack. From this analysis, we can note that the most affected service is eMBB, due to the huge packet size which introduces higher computation load (thus delay). The mMTC and URLLC services are affected by the splits in almost a similar way, with the difference that URLLC has a latency higher of $10\mu s$ than mMTC due to the higher packet size. In this analysis we can see that PDCP-RLC and MAC-PHY splits add a smaller latency than the intra-PHY split. This is due to the fact that, for the intra-PHY split, all symbols (i.e., 14 packets) need to be received by the low PHY layer for each transmission time and thus the effective delay introduced has to be considered from when the first packet (related to the first symbol) is transmitted from the CU to when the last packet (related to the last symbol) is received at the DU[1].

In Figure 4.5 the focus is on the latency introduced by each split (i.e., the time interval from when a packet transmission is triggered by the upper layer of the split to when the packet is successfully received by the lower layer of the split). This corresponds to summation of periods B to F in Figure 4.3 where latency(B), latency(C), latency(D), latency(E) and latency(F) corresponding to latency(7), latency(8), latency(9), latency(10) and latency(11) respectively in Figure 2.8. Therefore, latency in Figure 4.5 is computed as defined in Equation 2.6. It should be noted that in this experiment, when more than one packet is transmitted in one ms (Table 4.2) latency(C) and latency(E) (Figure 4.3) are subject to a queueing delay as packets are stored in a buffer and processed one by one by the network driver. Moreover, there are no switches in the experimental setup, hence $\mathrm{N_{switch}}$=0 in Equation 2.3. .

From this analysis, we can note that the PDCP-RLC and MAC-PHY splits work in a more stable way compared to intra-PHY in terms of added latency. In details, the average latency is almost constant for all the splits and equal to $\sim 65\mu s$ for PDCP-RLC and $\sim 60\mu s$ for MAC-PHY. It is thus interesting to note that the packet size of the different services affect the overall latency (as depicted in Figure 4.5) but not the one-way latency from upper to lower layers of the split, meaning that the highest source of delay comes from the processing of the packet through the protocol stack. On average, the latency of the intra-PHY split is equal to $\sim 810\mu s$. A further analysis can be found in Figure 4.6, which depicts the jitter of

[1] This behaviour is due to current OAI implementation, further improvements could be achieved when the DU manages the packets in parallel without waiting for the reception of all 14 packets. The implementation of this feature is out of the scope of this work.



(a) eMBB

(b) mMTC

(c) URLLC

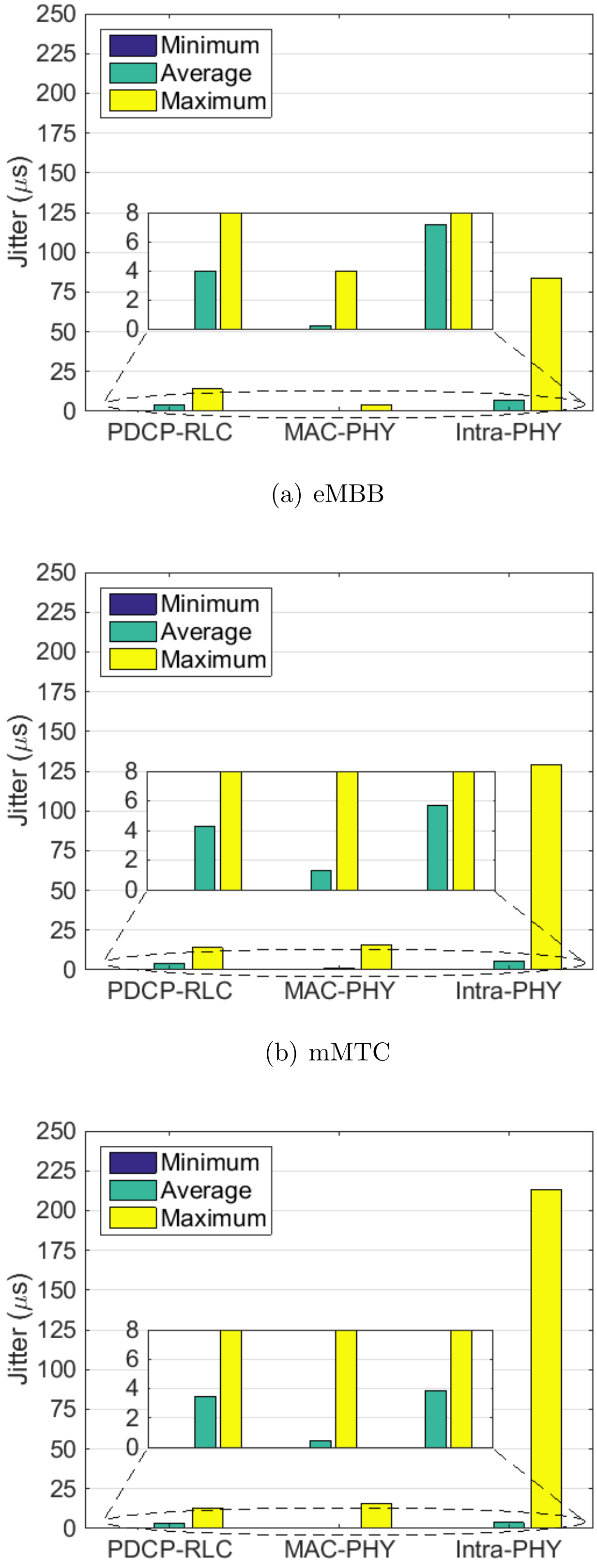
Fig. 4.5 Latency from the upper to the lower layer for different splits for 5G services.

the different splits. The jitter is computed as defined in Equation 2.8. On average, the lowest jitter is guaranteed by the MAC-PHY split, as the MAC and PHY layers work in a synchronous way thus reducing the delay variation. Higher jitter is obtained for the PDCP-RLC split, as in this case the PDCP sends a packet to RLC whenever it receives a packets from upper layers with thus higher latency variation. Finally, the highest jitter is obtained with the intra-PHY split due to the high number of packets (i.e., 14) transmitted every ms.

After analysing the performance in terms of latency and jitter, we now focus our attention on the overall pros and cons of the different splits for considered services. For URLLC, MAC-PHY split looks the most adequate solution as analysed above as it guarantees the lowest and more stable latency. A further analysis is shown in Figure 4.7, showing the performance when increasing the number of packets per ms injected at PDCP layer (as an evaluation of cases with heavy load or other use cases like for instance a mobile gateway simultaneously receives packets to be delivered to non-mobile equipped actuators). Figure 4.7 shows that the MAC-PHY split for URLLC has a stable performance when increasing the number of packets per ms, thus demonstrating the feasibility of this split for URLLC. The MAC-PHY split has in addition the advantage that PDCP is centralised, this being beneficial for multi-RAT convergence to increase reliability.

For delay-tolerant eMBB, all splits may be suitable from a latency point of view, but it is worth reminding that the demand of this service is mainly in terms of data rate. From this point of view, the intra-PHY split does not look to be a suitable solution as the load on the fronthaul directly depends on the available channel bandwidth (in our testbed with 5MHz bandwidth, the fronthaul rate for intra-PHY split is 67.2 Mbps for one antenna of one sector as in [2]). The MAC-PHY split is able to aggregate the packets on a UE-basis and the only added overhead is in terms of MAC header. This may thus help to reduce the load on the fronthaul in terms of pkt/s, and it thus makes the MAC-PHY a suitable split for delay-tolerant eMBB services. In addition, having a centralisation of PDCP and RLC layers would be beneficial for solutions such as multi-RAT convergence, dual-connectivity and better mobility management.

For mMTC, applications dealing with sensing (i.e., delay-tolerant) may be associated to any split, but the intra-PHY split may be a candidate solution for the following reason: all the traffic received by the CU (i.e., both user- and control-plane traffic) will be translated on the fronthaul with a fixed data rate as it depends only on the channel bandwidth. This is beneficial especially for new technologies expected to be used for mMTC such as Narrow-Band Internet of

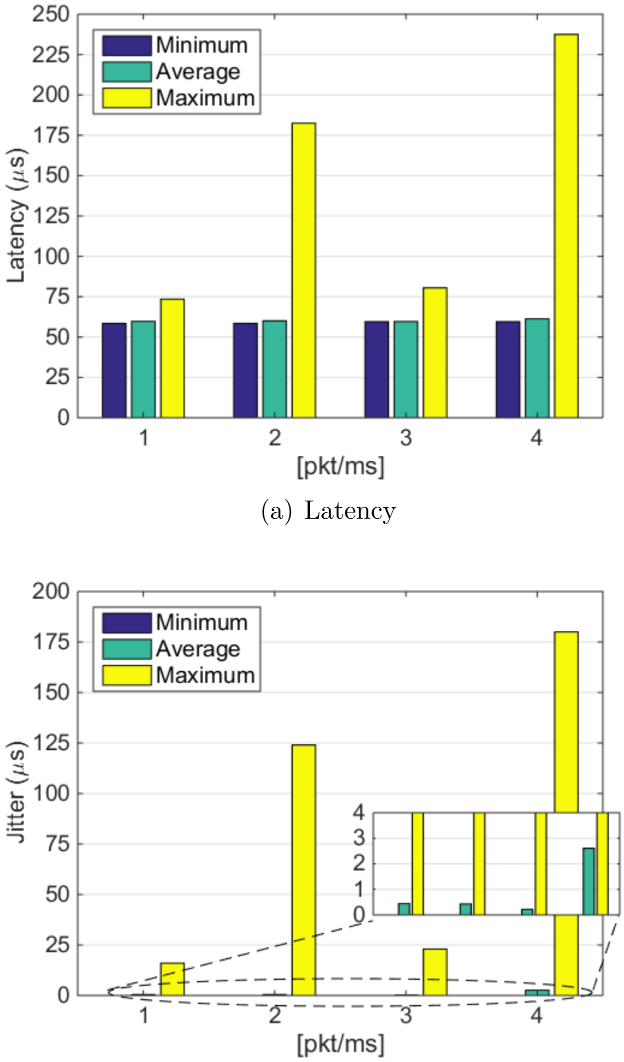


(a) eMBB

(b) mMTC

(c) URLLC

Fig. 4.6 Jitter for different splits for 5G services (the minimum jitter is equal to 0).



(a) Latency

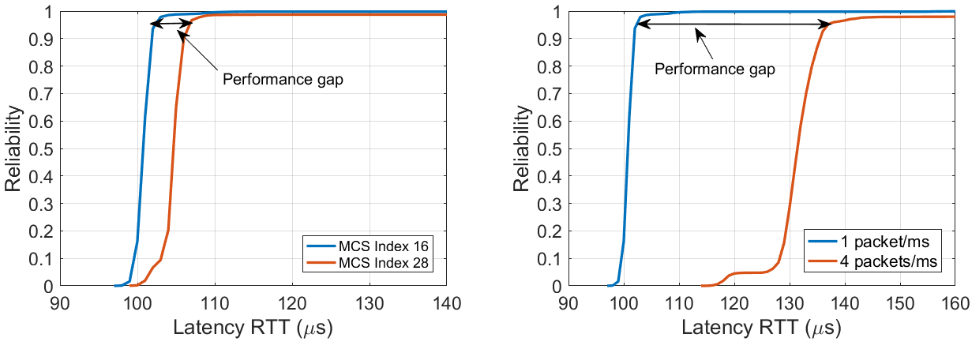
(b) Jitter

Fig. 4.7 Latency (a) and jitter (b) for URLLC with MAC-PHY split when varying the number of injected packets.

things (NB-IoT) [67]. By using the same implementation used in our testbed, the overall data rate on the fronthaul for intra-PHY split (where N_{RB} is equal to 1) for one antenna of one sector of NB-IoT would be 2.7 Mbps regardless the cell load.

Next, the focus is on measuring RTT latency as defined in Equation 2.3 which includes transport delay as well as processing and computing delay.

Figure 4.8 shows the latency distribution, i.e., cumulative distribution function (CDF) of the measured RTT latency when varying MCS index and packet arrival rate. The CDF captures the reliability defined as the probability that the measured RTT latency is lower than a predefined latency deadline (see Equation 2.10).



(a) Different MCS

(b) Different packet arrival rate

Fig. 4.8 Latency RTT for URLLC with MAC-PHY split when varying MCS (a) and packet arrival rate (b).

Figure 4.8(a) shows the impact of different MCS indices on reliability (MCS index 16 corresponds to MCS QAM while MCS index 28 corresponds to MCS 16 QAM).

Figure 4.8(a) indicates that the reliability degraded with the increase of MCS index. At 97 μs, the MCS index 16 starts providing some reliability, while the MCS index 28 starts providing some reliability at 99 μs. To achieve the same reliability, there is a performance difference of 5 μs between the MCS index 16 and the MCS index 28.

Scheduler selects an MCS index based on a channel quality indicator that the EU periodically reports. In this scenario, the same packet size is transported on the fronthaul when switching between the two MCS indices. The latency result is therefore primarily influenced by latency(6), i.e. protocol stack processing time (see Figure 2.8) and is a random component depending on the availability of resources at the time of data processing.

Figure 4.8(b) shows RTT latency distribution when increasing the URLLC packet arrival rates from 1 packet/ms to 4 packets/ms. Figure 4.8(b) shows that the reliability is lower when arrival rate is 4 packets/ms compared to when arrival rate is 1 packet/ms. There is a performance difference of 30 μs between the two arrival rates.

When the arrival rate is set to 4 packets/ms, there are two important factors to consider in this case:

- Multiplexing as many packets belonging to the same UE into one packet as possible to reduce overhead. This leads to a larger packet size that takes more time to process and transmit.
- Transmitting one packet after another, which causes a queueing delay.

To sum up, URLLC needs to have a maximum end-to-end delay of no more than 1 ms. So it is crucial to be aware of the various components of latency especially non-deterministic ones to avoid them.

4.5 Fronthaul Data Rate in the Context of 5G

5G technologies such as massive MIMO and wide channel bandwidth will drive new requirements for 5G transport networks in Cloud-RAN. Thus, in this section, the fronthaul bit rate is evaluated based on certain assumptions supported by 5G, such as number of antenna ports, number of spacial layers and bandwidth transmission.

For 5G, the bandwidth increases to 400 MHz, the number of antennas is 32 or more, and number of layers is increased to 8 for UL and DL. Larger bandwidth is associated with a higher number of subcarrier and higher number of layers is associated to higher number of streams. Consequently 5G configuration would lead to higher bit rate on the fronthaul as shown in Table 4.3. These fronthaul bit rates are calculated on the basis of [38.801] using the parameters set out in Table 4.3.

It is presumed that the backhaul data rate corresponds to the peak DL rate that can be supported by the relevant radio access technologies.

For PDCP-RLC split (option 2), the bandwidth includes RRC signalling for PDCP configuration, UP data and DL signalling. The DL signalling is related to the number of UEs who receives DL signalling. Based on the assumption that 0.4% of backhaul bit rate is DL signalling, the fronthaul bit rate for split option 2 is calculated as shown in Table 4.3.

For MAC-PHY split (option 6), the fronthaul bit rate takes into account of additional overhead for scheduling associated to PHY that needs to be transmitted over fronthaul. Fronthaul bit rate for MAC-PHY split is calculated assuming 3.4% of backhaul bit rate is scheduling and control signalling to be sent to PHY in DU.

For intra-phy split (option 7-1) where precoding is employed using only digital beamforming, each antenna element, thus, is addressed with radio signal's frequency domain symbols. In this case, the fronthaul data rate is proportional to number of subcarriers and number of antenna ports. In each transmission over fronthaul, MAC needs to send MAC control information to PHY that includes information on how to assign resource blocks, configure antenna etc. It is assumed that this overhead might amount to about 3.2% of frothaul data.

For intra-phy split (option 7-1) with precoding performed using digital beamforming in CU and analogue beamforming in DU and assuming that hybrid beamforming of dimension the same as number of layers is employed, then the fronthaul bit rate is proportional to number of layers.

The fronthaul bit rate will increase to the order of terabits for 5G deployments with a bandwidth of 1 GHz if the hybrid beamforming is not carried out for precoding. However, if hybrid beamforming is used, the fronthaul is reduced by a factor of 13.

Based on these results, multiple lanes of 100 Gbps may be used to handle the fronthaul bit rate of the order of thousands of Gbps. Compression fronthaul data can be used as one solution to minimise fronthaul capacity requirements. However, compression can only help to reduce the fronthaul data rate to some degree. In [68] compression decreases the data rate by a factor of three while achieving good efficiency.

Another approach, is to consider other functional splits that can reduce the bit rate of the fronthaul. Appropriate functional splitting should therefore be chosen based of the use case, the requirements of the applications to be served and the capacity of the available fronthaul.

4.6 Concluding Remarks

In this chapter, we study three functionality split options, i.e., PDCP-RLC, MAC-PHY, and intra-PHY, in a cloud-RAN environment, and their impact on delivering 5G traffics. These three splits are selected out of eight possible options presented by the standardisation community such that both lower layer and higher layer splits are examined. Above splits have been implemented in an SDR testbed in the OAI platform with an Ethernet-based fronthaul. We stated advantages and

Table 4.3 Expected fronthaul bit rate considering different functional splits.

Radio Access Technology	LTE 20 MHz	5G 100 MHz	5G 1 GHz
Modulation order	64 QAM	256 QAM	256 QAM
Peak data rate on backhaul	150 Mbps	4000 Mbps	40000 Mbps
Number of antenna ports	2	64	128
Number of spatial layers	2	8	8
Number of subcarriers	1200	6000	60000
Bitwidth per I or Q	16 bits	16 bits	16 bits
Number of symbols/ms	14	14	14
DL fronthaul bit rate split option 2	150.6 Mbps	4016 Mbps	40.16 Gbps
DL fronthaul bit rate split option 6	155.1 Mbps	4136 Mbps	41.36 Gbps
DL fronthaul bit rate split option 7-1	1.1 Gbps	177.5 Gbps	222 Gbps
DL fronthaul bit rate split option 7-1 with Hybrid Beamforming	1.1 Gbps	22.2 Gbps	3551 Gbps

disadvantages of each split in terms of load on the fronthaul for each service under consideration. Then, for each split, we evaluated latency and jitter. The evaluations show that the MAC-PHY split is the most suitable split option for URLLC as being able to guarantee the lowest delay and jitter due to the synchronisation needed between MAC and PHY layers.However, these results largely depend on the platform employed, fronthaul network topology, and the traffic load.

The functional splits provides the potential capability for RAN to support network slicing. This is because the latency and jitter results are different for each of the functional splits. In this context, each functional split may belong to a slice. Accordingly, the requirements provided by functional split can be guaranteed by the slice. As a consequence, a particular type of service is dedicated to an appropriate slice according to the need for an application. For example, a slice of MAC-PHY split would be suitable for applications requiring low latency and jitter.

Chapter 5

Reliable Ethernet-Based Fronthaul using multi-path in Support of Ultra-Reliable Low Latency Communication

5.1 Introduction and Contributions

After experimenting with different functional splits and evaluating their effect on the delivery of 5G traffic from latency and jitter perspectives in Chapter 4, the focus of this Chapter is on fronthaul reliability, that is another important performance metric, particularly for URLLC applications.

One of the major paradigm shifts in the mobile and wireless networking is the transition from a static network configuration to a flexible and reconfigurable network setup. Among the solutions that made their way into the architectural choices for the 5G of wireless networks, is the cloudification of the RAN through the Cloud-RAN architecture [69]. Cloud-RAN offers the flexibility to move the baseband functionalities to a CU in support of multiple DUs. The DUs are connected to the CU via the fronthaul. While enabling centralised processing and control, the introduction of the fronthaul introduces an additional segment in the network, over which the main KPIs should be delivered [54]. In particular, one of the key challenges in 5G is achieving high reliability and low latency KPIs at the same time in order to enable innovative applications and use cases, such as the Tactile Internet and industrial automation and control.

The 3GPP standardisation body has provided guidelines for delivering URLLC [70]. The new specification defines the New Radio interface, which introduces structural change in the PHY in terms of numerology in order to meet stringent constraints on latency and reliability. However, there is limited research on how fronthaul technologies can support the levels of latency and reliability needed for 5G URLLC services. This Chapter aims at addressing this knowledge gap.

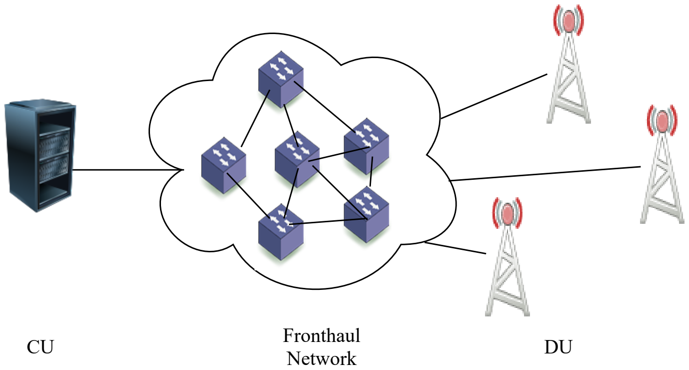


Fig. 5.1 Cloud-RAN architecture with multi-hop packet-based fronthaul network.

The conventional fronthaul topology consists of dedicated lines from the CU to each DU, i.e., of point-to-point links, that transports baseband radio samples in a serial manner. A more economic solution has recently emerged, whereby dedicated lines are replaced by a multi-path packet-based fronthaul network that can leverage the wide deployment of the Ethernet infrastructure [71] (see Figure 5.1). Multi-hop packet-based networks make it ever more challenging to ensure the high reliability and low-latency KPIs expected by 5G systems.

Increasing reliability is typically accomplished through retransmissions based on feedback or through redundancy. Depending on the split point between CU and DU, the latency requirements on the fronthaul link differ but the figures are in the range between 55 μs and 1 ms as shown in Section 4.4. Given this strict delay requirement, retransmission based on feedback is not a viable choice. In contrast, redundancy through transmission over multiple fronthaul links offers a feasible solution. In a multi-hop packet-based network, this can be realised by transmitting over multiple paths, or routes, between the CU and an DU.

While different options of split point between CU and DU have been introduced in the 3GPP standard [3] as shown in Figure 2.2, the focus in this thesis is on the split between PDCP and RLC. Accordingly, the fronthaul transports PDCP packet.

In this context, as a first option to leverage multi-path transmission, one could replicate the same PDCP data stream across multiple fronthaul paths, yielding *Multi-Path Transport with Duplication* (MPD). MPD ensures correct reception as long as any path succeeds in delivering the fronthaul data. This increases reliability, while also increasing fronthaul network congestion and hence potentially affecting the latency KPI. As a dual solution, one could split each PDCP PDU in disjoint blocks transmitted on different paths. Since each fronthaul path would need to carry less information, this approach would generally reduce congestion and transport latency on each path, but reliable transport would rely on the correct reception on all paths.

As a means to bridge the gap between the two extreme solutions highlighted above, in this chapter it is proposed to use coding techniques on the PDCP PDU transported by the fronthaul network. Coding can reduce the fronthaul transport overhead as compared to MPD, while still providing resilience to the potential unreliability on some of the fronthaul paths. Specifically in this Chapter, erasure coding methods, such as rateless or fountain, coding is considered. The considered approach is referred as *Multi Path Transport with Coding (MPC)*.

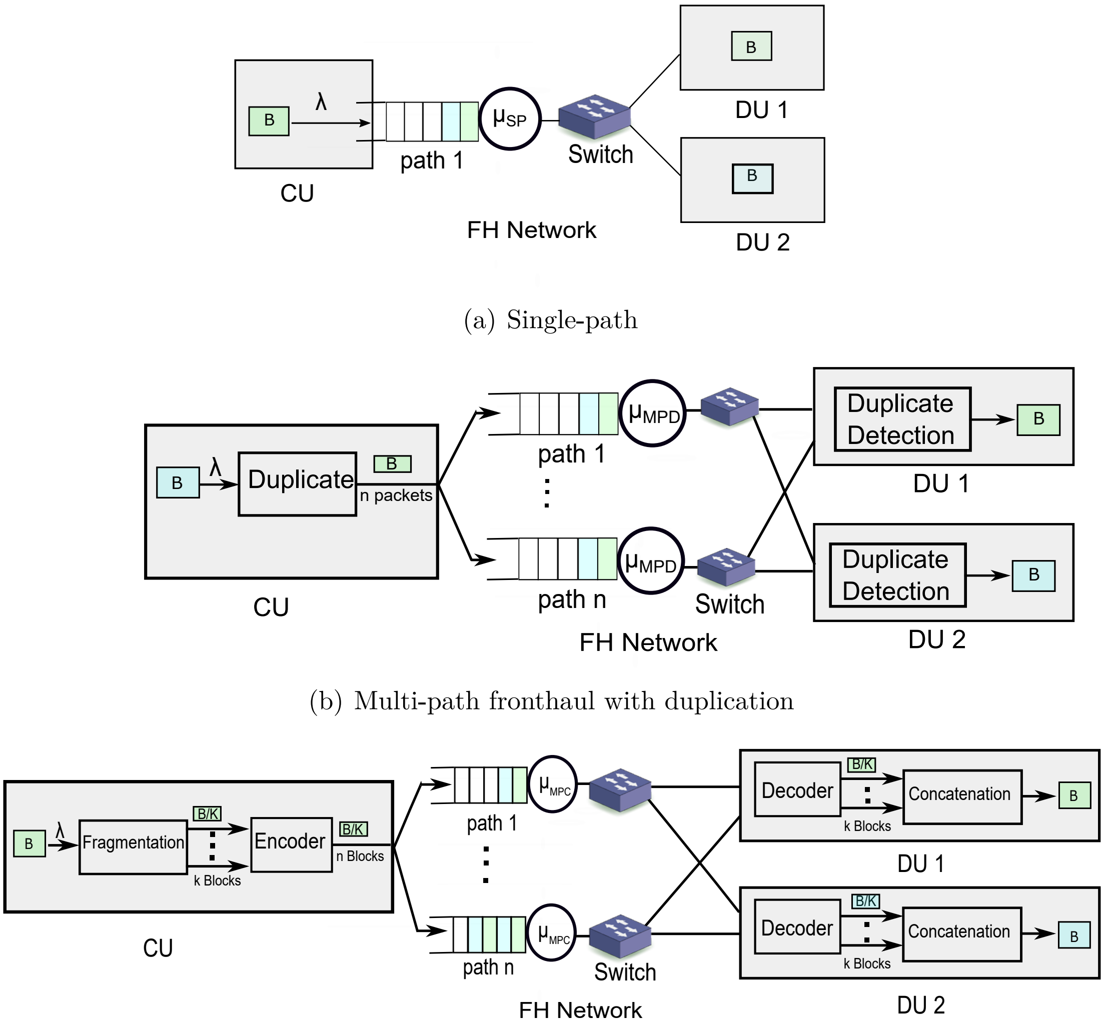
To elaborate, in this Chapter, a Cloud-RAN model with multiple packet-based fronthaul paths between a CU and DUs is considered. To this end, the contributions can be summarised as follows:

1. Performance analysis of baseline single-path (SP) transmission, MPD and MPC for DL communication. The performance is evaluated in terms of average latency for reliable delivery and of the reliability-latency trade-off;

2. Based on the insights from the analysis, we then present extensive experimental results concerning the performance of Cloud-RAN with multi-path fronthaul under MPD and MPC in the presence of both eMBB and URLLC services as described by the 3GPP standard;

3. From experimental results find optimal parameter settings to achieve the desired reliability and latency results.

4. Compare two strategies of sharing of fronthaul resources, namely orthogonal and non-orthogonal resource allocation modes.



(a) Single-path

(b) Multi-path fronthaul with duplication

(c) Multi-path fronthaul with erasure coding

Fig. 5.2 Fronthaul solutions for downlink communication.

5.2 Fronthaul Solutions and System Model

In 5G, it is recommended to avoid having a single point of failure that could affect a high number of radio sites. Where appropriate, it is important to enforce redundancy schemes. This is because, in a macro site where a single transport facility may aggregate multiple radios, failure or repair of such transport equipment may have an impact on a large operator serving area. The key requirement of the 5G RAN transport network is to allow meshed connectivity to enable reliability and resilience [72]. In this context, the scenario in which the 5G transport network offering multiple connectivity is considered in this chapter.

In this section, first the fronthaul transport strategies for SP, MPD, and MPC are detailed and then the system model for the Cloud-RAN system under study is presented.

5.2.1 Fronthaul Solutions

As discussed, in order to improve reliability of fronthaul transfer, one can in principle apply retransmission methods based on feedback, multi-path transmission with duplication, or MPD, and multi-path transmission with coding, or MPC. The improvement in reliability generally comes at the expense of latency. In the case of retransmissions, the need for feedback and for additional protocol control messages can increase the transmission time in a highly nonlinear manner [73], making it a non-viable solution for fronthauling. Not requiring any form of feedback, MPD can offer better latency performance, but the congestion caused by the transmission of duplicated packets over multiple interfaces can still entail unacceptable latency levels [74]. More generally, MPC can add controlled redundancy in order to obtain a desired trade-off between reliability and diversity. In the context of packet-based fronthaul, the most relevant coding schemes are erasure codes, which enable the recovery of a data stream, despite missing packets, as long as a sufficiently large number of encoded packets are received. The details of fronthaul solutions SP, MPD, and MPC are as follows:

1. Single Path (SP): As illustrated in Figure 5.2(a), with SP, the CU uses a single path to communicate to both DU1 and DU2, with the two data streams sharing the same path.

2. Multi Path with Duplication (MPD): With MPD, as shown in Figure 5.2(b), the CU replicates each received PDCP packet, intended for either DU, on

each of the n fronthaul paths. At each DU, the *duplicate detection* block detects the first successfully received frame and drops the rest.

3. Multi Path with Coding (MPC): With MPC, an (n, k) erasure code is used at the CU, with $k \leq n$. To this end, the CU carries out the following steps for each PDCP packet, which may be intended for either DU:

    - Fragment each PDCP packet into $k \leq n$ blocks;
    - Encode the k blocks into n encoded blocks of the same size using an erasure code;
    - Send each block into one of the n paths.

    By the properties of optimal, or Maximum Distance Separable, erasure codes, at each DU, the original PDCP packet can be retrieved if any k out of n encoded blocks are received successfully. Note that, with $k = 1$, MPD is obtained as a special case. Furthermore, setting $k = n$ yields a strategy where frames are split into disjoint segments, each sent on a different path. Overall, increasing k from 1 to n gives strategies that range from MPD to frame splitting, with the former having the largest block size and the latter the smallest block size.

5.2.2 System Model

A system model consisting of the downlink of a Cloud-RAN system with a single CU and a number of DUs, which are connected by a multi-path fronthaul network with n pre-defined and distinct paths is considered as shown in Figures 5.2(b) and 5.2(c). Each path ends at a switch that can deliver a PDCP packet to any of the DUs with negligible delay. To simplify the discussion, the case of two DUs is considered henceforth, but the treatment applies more generally. Each path is modelled by a queue with a single server and an exponential service time with rate c bits per second. More specifically, in order to transfer a PDCP packet of B bits, each queue takes an amount of time that is exponentially distributed with mean $1/\mu = B/c$ seconds. A schematic diagram of the queue model is shown in Figure 2.9. It consists of n parallel M/M/1 queues. A mathematical model of the distribution of the latency (waiting and transmission time) for each fronthaul solution is provided in Section 5.3.1 while their mathematical model of the distribution of the reliability is given in Equation 2.17 and described in detail in Section 5.3.2. Note that the queueing model abstracts the details of the multi-hop paths. Downlink traffic arrives at the CU with arrival rates and frame sizes that depend on the service type, as it will be further discussed in the next sections. As indicated in its metadata, each frame needs to be delivered to either of the DUs.

5.3 Analysis with Single Service

In this section, the performance of the Cloud-RAN system at hand is analysed assuming a single service with random arrivals of PDCP packets of size B bits with exponential inter-arrival periods with average $1/\lambda$ s. Let's assume that each frame may be tagged independently and with equal probability as being destined to either DU. The average latency required to obtain reliable fronthaul transport is studied first then the reliability-latency trade-off is studied.

5.3.1 Average Latency for Reliable fronthaul Transport

The analysis of the average latency is presented separately for SP, MPD, and MPC. It is emphasised that the latency is defined as the time elapsed from the time packet is transmitted over the fronthaul until the packet is successfully received. For average latency and Reliability-Latency Trade-Off analysis, MPD corresponds to MPC with $k = 1$.

Under the given assumptions, the average latency T_{SP} for SP can be expressed by using the standard average delay formula for M/M/1 queues. Hence, T_{SP} can be expressed as Equation 2.9, where $\mu = c/B$ is the average departure rate in frames per second.

For MPD, a simplified assumption is made to analyse the performance, that is, as soon as a frame is correctly received by the intended DU, all other $n-1$ copies of the same frame are deleted from the other paths. Note that this is not the case in the actual system given that it is practically difficult to remove all copies of a frame along all other paths. Therefore, the expression here provides a lower bound on the average latency. The bound is expected to be tight if the load of each path is sufficiently small. The next section provides some numerical evidence.

Under the given assumption, the end-to-end system can be studied as an M/G/1 queue in which the service time is the first-order statistic ($X_{(k)}$, k=1) of n exponential variables [48], each with mean $1/\mu_{\text{MPD}} = B/c$. The mathematical model of this system is shown in Figure 5.3. Therefore, the average latency T_{MPD} in the original setup can be lower bounded by using the Pollaczek-Khinchin formula

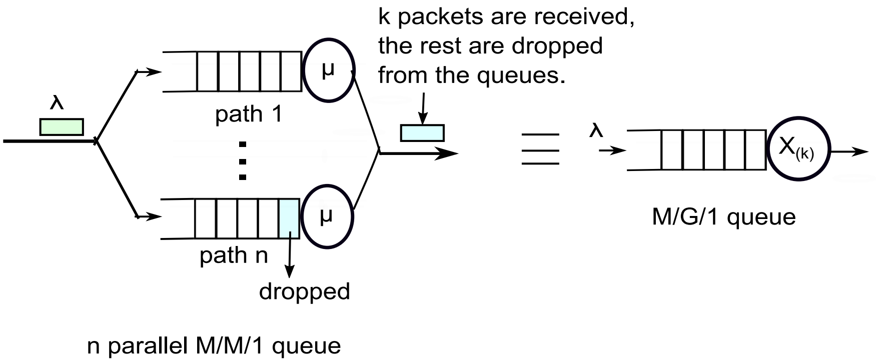


Fig. 5.3 n independent, parallel M/M/1 queues with identical arrival rate λ and service rate μ the system is equivalent to an M/G/1 queue with service time $X_{(k)}$

$T_{\text{MPD}} = E[X_{(k)}] + \frac{\lambda E[X_{(k)}{}^2]}{2(1-\lambda E[X_{(k)}])}$, where $k = 1$, $E[X_{(k)}]$ is defined in Equation 2.13 and $E[X_{(k)}{}^2]$ is defined in Equation 2.15.

With MPC, each frame is correctly detected as soon as the first k encoded blocks are received. In a manner similar to MPD and following [48], a lower bound on the average delay is obtained by considering a system in which, as soon as k encoded packets are received, the rest are dropped from the remaining queues.

As a result, the end-to-end system can be studied as an M/G/1 queue, in which the service time is a random variable distributed according to the k^{th} order statistic $(X_{(k)})$ of the exponential distribution with mean $1/\mu_{\text{MPC}} = B/(kc)$. Note that the service time for each queue has mean $1/\mu_{\text{MPC}}$ that decreases with k due to the smaller encoded blocks.

The mathematical model of this system is shown in Figure 5.3 in which the average latency can be again lower bounded by using the Pollaczek-Khinchin formula $T_{\text{MPC}} = E[X_{(k)}] + \frac{\lambda E[X_{(k)}{}^2]}{2(1-\lambda E[X_{(k)}])}$, where $1 < k \leq n$, $E[X_{(k)}]$ is defined in Equation 2.13 and $E[X_{(k)}{}^2]$ is defined in Equation 2.15.

5.3.2 Reliability-Latency Trade-Off

Within the context of 5G, and for the last mile communication, there is a large body of research on improving reliability while maintaining low latency. Generally speaking, the increase in reliability comes at the cost of latency. On the one hand, short block-length codes are less reliable; on the other hand, longer block-length codes are more reliable but require increasing latency, which may not be ideal for time-sensitive applications. It is, therefore, important to evaluate the relationship between the acceptable latency and the achieved robustness. This relationship is referred to in this thesis as reliability-latency trade-off.

In this section, an approximate analysis of the reliability-latency trade-off is provided by studying the probability that correct fronthaul transport occurs by a given threshold latency. The aim is to find optimal parameter settings to achieve the desired reliability and latency results.

This is the probability that the service time on each queue does not exceed t when the transported block is of size B. Now, the analysis of the reliability-latency trade-off is carried out in a manner similar to [75], as detailed next.

For SP, the reliability-latency curve is directly given as Equation 2.11. For MPD, a frame is correctly decoded as long as one path is successful in delivering it, which yields the reliability-latency function as defined in Equation 2.17, with $k = 1$.

With MPC, delivery is correct when any k of the n paths deliver an encoded packet correctly, which gives the reliability-latency to be defined as in Equation 2.17.

5.4 Experiments with eMBB-URLLC Services

In this section, a simulation model is developed to validate the analysis and to account for the more realistic scenario in the context of 5G, in which both eMBB and URLLC traffics coexist on the same Cloud-RAN system. It is noted that, while Tactile Internet applications are generally considered within the class of URLLC, it is expected that a combination of different traffic classes may be needed for the delivery of a particular Tactile Internet use case. Take remote medical intervention as an example. The application may require the transmission of a high definition real-time video stream, multiple sensors' data, as well as kinaesthesia data using a haptic device [76]. The video stream requiring high data rate can be classified as eMBB; sensors' data as massive machine-type communications; and kinaesthesia data that requires an end-to-end latency of 1 ms as URLLC traffic. Here, the focus is on the coexistence of eMBB and URLLC services. The traffic models are based on reference [64], where URLLC and eMBB are modelled with full buffer bursty traffic FTP model 3 [65] and IP packet size of 500 and 1500 bytes, respectively.

To elaborate, the assumption is made that eMBB and URLLC traffics are independent and characterised by arrival rates and packet sizes as shown in Table 5.1. For each traffic, frames are tagged independently and with equal probability as intended for either DU. There are $n = 10$ paths, with each path having a capacity of $c = 100$ Mbps.

Three different coexistence strategies for the eMBB and URLLC services are compared:

- *Fronthaul Bandwidth Orthogonal Allocation*: Each service is exclusively given a fraction of the capacity of each path;
- *Fronthaul Path Orthogonal Allocation*: Each path is allocated exclusively to either one or the other service;
- *Shared fronthaul*: Both traffic types share all fronthaul paths.

For all coexistence strategies, SP, MPD, or MPC are implemented in order to control the reliability-latency trade-off. Furthermore, for the orthogonal schemes based only bandwidth or path splitting, the analysis presented in the previous section applies separately to both services, whereas shared fronthaul requires a more complex analysis that is considered to be outside the scope of this contribution.

Table 5.1 System model parameters for 5G services

Type of traffic	eMBB	URLLC
Packet Size (Bytes)	1500	500
λ (packet/ms)	8	24

5.4.1 Average Latency for Reliable Fronthaul Transport

In this section, the average latency required for the successful delivery on the fronthaul of a PDCP packet is studied as investigated in Section 5.3.1.

Figure 5.4 shows the average latency as a function of the frame splitting factor k under MPC for both eMBB and URLLC services using shared fronthaul transmission. Figure 5.4 also shows the performance of SP for the reference. Note that MPD corresponds to MPC with $k = 1$. It is observed that MPC and MPD can drastically decrease the average latency as compared to SP for both eMBB and URLLC services. It is also seen that, under shared fronthaul, the delay of both services is quite close, given that the overall latency tends to be limited by queueing delays, i.e., by the time needed to traverse each shared path. Furthermore, for MPC there is an optimal value of k, which is around $k = 2$ for eMBB and $k = 3$ for URLLC. The lower optimal value of k generally depends on both arrival rate and packet size of both services. This plot provides insight on how to choose k.

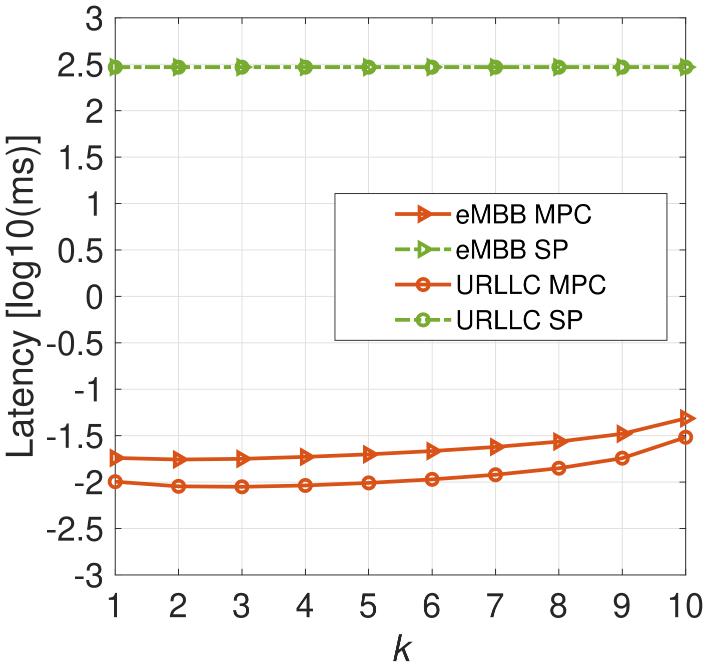


Fig. 5.4 Average latency as a function of the frame splitting factor k for SP and MPC for both eMBB and URLLC using shared fronthaul transmission. Note that MPD corresponds to MPC with $k = 1$.

For example, if the average latency for URLLC should not exceed 0.01 ms, then the choice $k \leq 5$ would satisfy the requirement.

Let's consider orthogonal fronthaul transmission schemes that can allocate a different amount of resources to eMBB and URLLC. For this analysis, MPC is adopted with $k = 2$. Figure 5.5 shows the average latency for eMBB and URLLC using orthogonal bandwidth allocation on the fronthaul as a function of the fraction of the available path capacity, c, that is allocated to eMBB. It is observed from the plot that, as compared to shared fronthaul transport, orthogonal bandwidth allocation allows one to obtain a lower average latency for URLLC. Note that this is not necessarily the case for eMBB service, which is characterised by larger PDCP packets. The figure also compares simulation and analysis, showing that the lower bounds derived in the previous section are tight when the load is not too high for each service.

To complement these results, Figure 5.6 shows the average latency under fronthaul path orthogonal allocation as a function of the number of paths, n_e from total $n = 10$, that are allocated to eMBB; the remainder n_u=$n - n_e$ are allocated to URLLC. The qualitative trend is the same as for bandwidth allocation. In particular, the average latency of URLLC can be reduced as compared to the shared

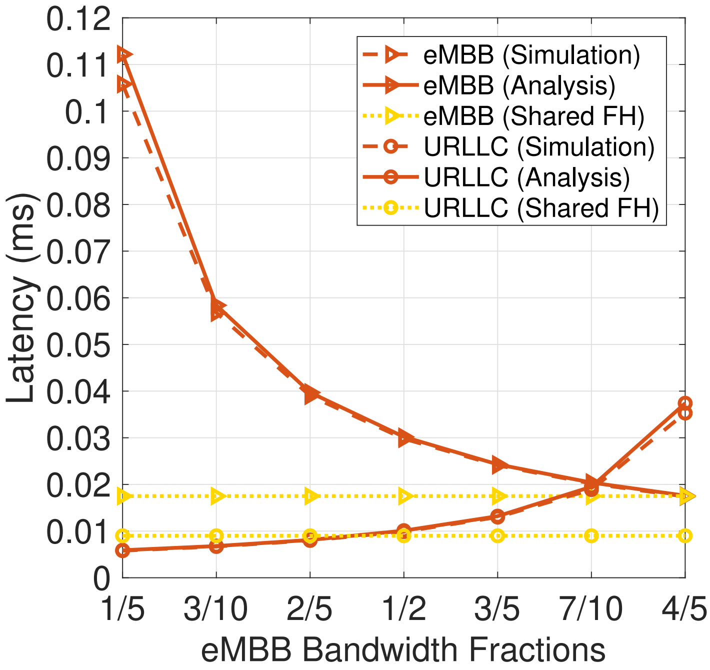


Fig. 5.5 Average latency as a function of the eMBB bandwidth fraction for MPC with $k = 2$ for both eMBB and URLLC using fronthaul bandwidth orthogonal allocation.

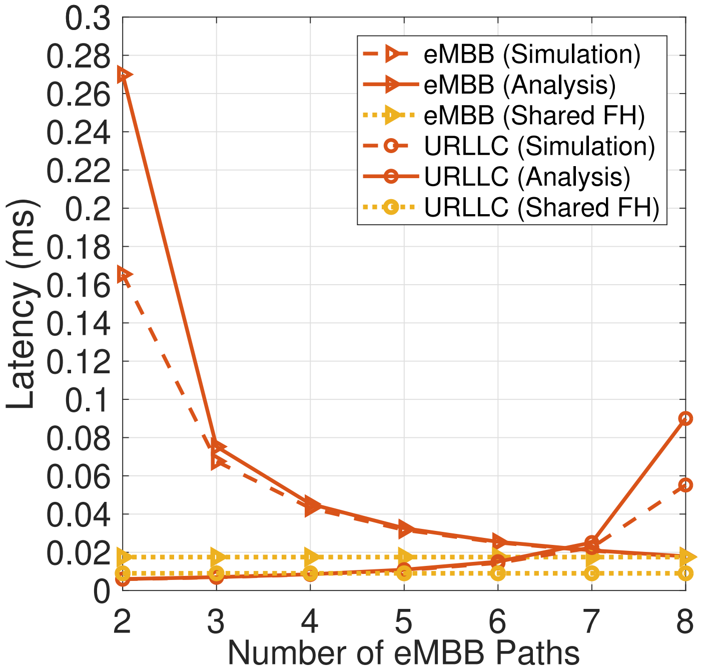
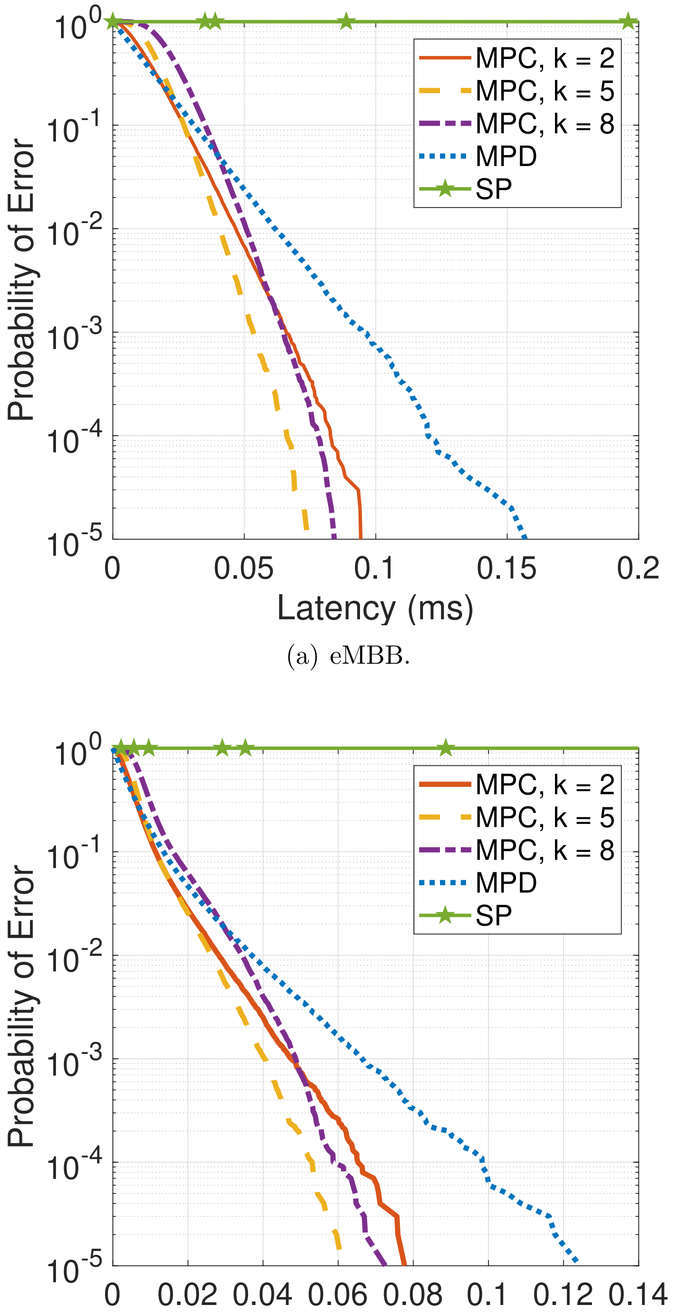


Fig. 5.6 Average latency as a function of the number of eMBB paths for MPC with $k = 2$ for both eMBB and URLLC using fronthaul path orthogonal allocation.

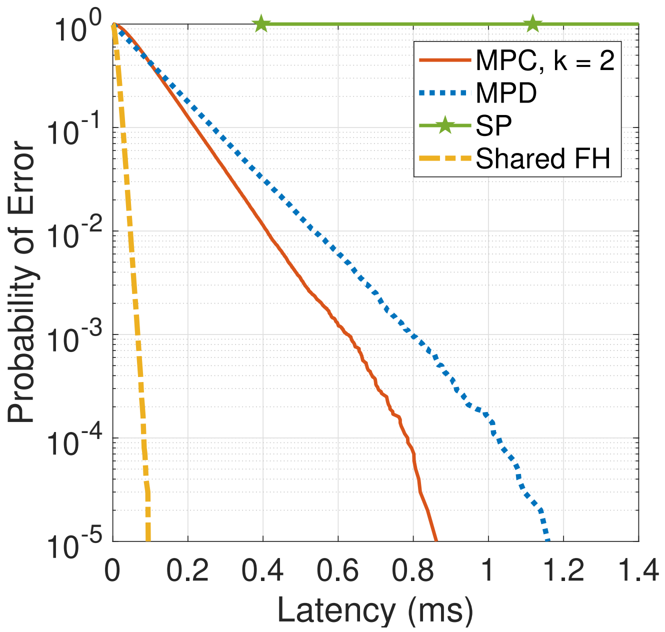
fronthaul case by means of orthogonal allocation. Furthermore, comparing path and bandwidth allocation schemes, it can be seen that bandwidth allocation can generally yield a more desirable trade-off between eMBB and URLLC performance.



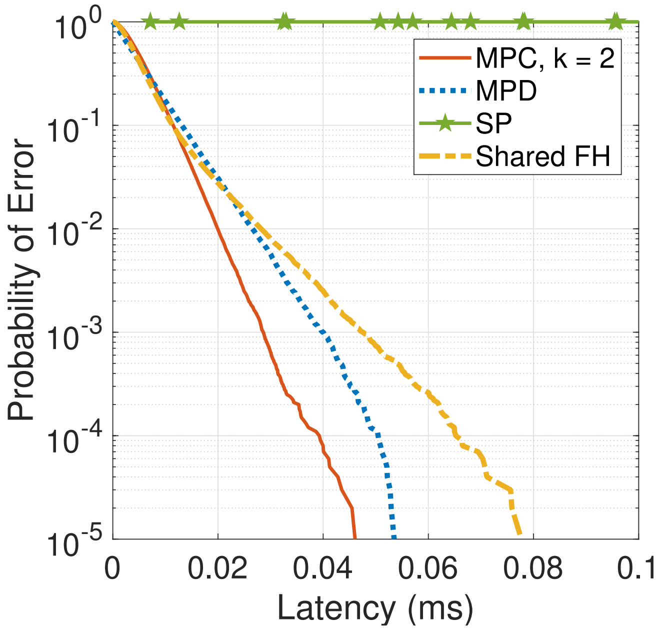
(a) eMBB.

(b) URLLC.

Fig. 5.7 Probability of error vs latency functions for SP, MPD, and MPC under shared fronthaul transport: (a) eMBB; (b) URLLC.



(a) eMBB.

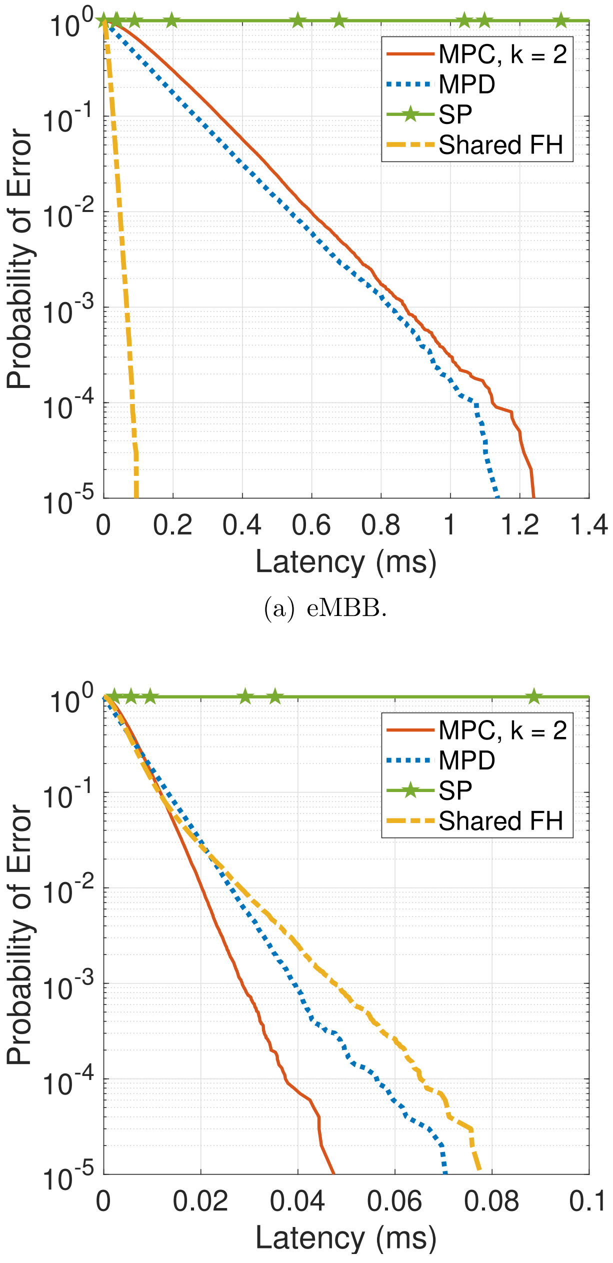


(b) URLLC.

Fig. 5.8 Probability of error vs latency functions for SP, MPD, and MPC under orthogonal fronthaul bandwidth split with eMBB bandwidth fraction 1/5: (a) eMBB; (b) URLLC.

5.4.2 Reliability-Latency Trade-Off

In this section, the trade-off between latency and reliability is considered by plotting curves of reliability versus latency as defined in the previous section. In order to focus on the main regimes of interest, the error probability function is plotted, which is defined as the complement of the reliability, i.e., $1 - F_{\text{SP}}(t)$ for SP and similarly for MPD and MPC. It is noted that the maximum requirement for error in the Tactile Internet applications is considered to correspond to the value of 10^{-5} as probability of error [26].



(a) eMBB.

(b) URLLC.

Fig. 5.9 Probability of error vs latency functions for SP, MPD, and MPC under orthogonal fronthaul bandwidth split with eMBB path fraction 1/5 ($n_e = 2$ and $n_u = 8$): (a) eMBB; (b) URLLC.

Figure 5.7 compares the reliability-latency of SP, MPD (or MPC with $k = 1$), and MPC with $k = 2$, $k = 5$, and $k = 8$ for eMBB and URLLC traffics using shared fronthaul transmission. First, as for the average latency, it can be noted the dramatic gains obtained by means of multi-path transport as compared to SP. Moreover, for both eMBB and URLLC, it is observed that MPC is instrumental in achieving high levels of reliability at moderate latency levels. For example in eMBB, in order to achieve a probability of error of 10^{-5} MPC with $k = 5$ requires 0.074 ms, while MPD entails a latency of 0.157 ms. Furthermore, thanks to the smaller packet sizes, URLLC traffic generally attains the same level of reliability at a lower latency in the presence of shared fronthaul.

Now, the performance under orthogonal resource allocation is considered. Figure 5.8 shows the probability of error as a function of the latency under bandwidth allocation with one fifth of the bandwidth allocated to eMBB. MPC with $k = 2$ can achieve a latency reduction of 0.3 ms and 0.0075 ms in eMBB and URLLC, respectively, with respect to MPD at the error probability of 10^{-5}. Furthermore, a larger bandwidth allocation to URLLC can significantly enhance the reliability of URLLC traffic at the cost of a larger latency for the eMBB service.

Finally, Figure 5.9 considers orthogonal path allocation with one fifth of the paths allocated to eMBB, i.e. $n_e = 2$ and $n_u = 8$. For URLLC, it can be seen that MPC can reduce latency by 0.023 ms as compared to MPD at the error probability of 10^{-5}. Moreover, the probability of error obtained with path split is improved by approximately 60% as compared to that obtained by non-orthogonal sharing of fronthaul resources. Nevertheless, bandwidth allocation is seen to provide a better trade-off between URLLC and eMBB performance.

5.5 Concluding Remarks

In this paper, we have studied the problem of ensuring low-latency and high-reliability in a Cloud-RAN system with multi-path Ethernet-based fronthaul network. The proposed solution is based on erasure coding and multi-path transmission on the fronthaul network. With this approach, the CU splits the original MAC frame into smaller blocks, encodes them into a larger number of encoded blocks, and then transmits them over the multiple paths. The solution is analysed and compared with conventional single-path fronthaul transport and multi-path methods based on duplication. The performance is evaluated in terms of average latency for reliable delivery and of the reliability-latency trade-off. The results consider the coexistence of URLLC and eMBB traffic on the fronthaul under both orthogonal and non-orthogonal fronthaul resource allocation modes. As a general conclusion, MPC can achieve a low error probability of 10^{-5} at lower latency than MPD by means of orthogonal as well as non-orthogonal shared fronthaul. Furthermore, in the presence of eMBB-URLLC coexistence, we showed that, by adequately managing the fronthaul resources via orthogonal allocation transmission, the average latency and the error probability can be effectively reduced as compared to shared fronthaul transport. These results can serve as a valuable guidance on how to effectively deploy multi-path fronthaul resources for the implementation of Tactile Internet Applications.

Chapter 6

Experimental Evaluation of Reliable Ethernet-Based Fronthaul

Within the context of 5G, there is a growing interest in the need to provide high reliability while maintaining low latency to support delivery of URLLC. To this end, and with an emphasis on Cloud-RAN, it is important for the fronthaul to maintain the strict KPIs of the URLLC.

In Chapter 5, a Cloud-RAN model with Ethernet-based fronthaul network for exploiting multi-path diversity using fountain code was proposed. The aim was to investigate reliability-latency trade-off on the fronthaul. It was shown analytically and through simulation that the solution is promising reliability enhancing one. Following up on Chapter 5, in this Chapter, an experimental evaluation of the proposed solution is provided to demonstrate the feasibility of its implementation in an industry-grade hardware testbed.

The remainder of this paper is organised as follows. In Section 6.1, detailed setup of the hardware experimental testbed is provided to examine the performance of the proposed system. Section 6.2 analyses experimental results. Finally, the conclusion is in section 6.3.

6.1 Experimental Testbed

For our experimentation, a long-term evolution system based on OAI Cloud-RAN platform with multiple packet-based fronthaul is considered to exploit the concept of multi-path diversity. Our choice of the long-term evolution module is based

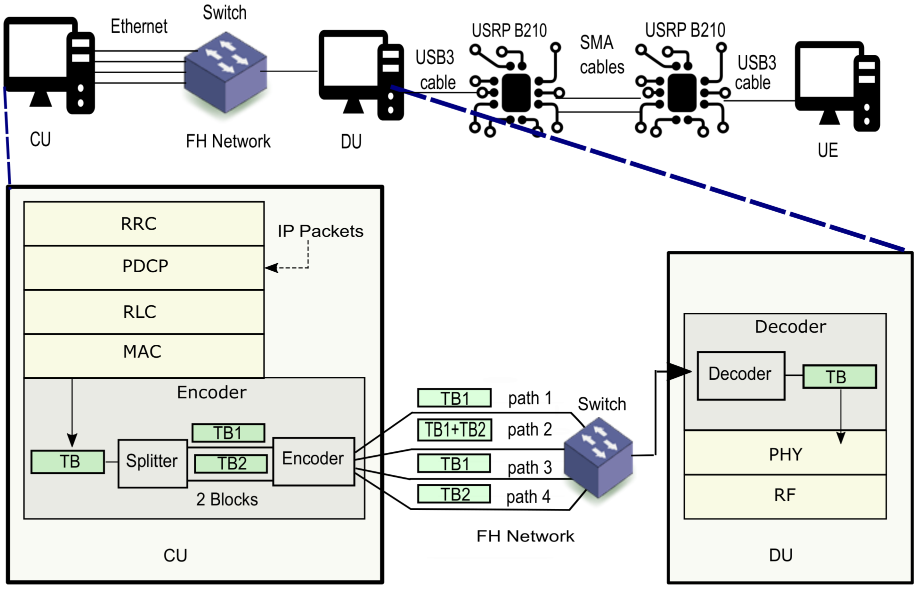


Fig. 6.1 End-to-End System Experimental Setup for Multi-path fronthaul with erasure coding (MPC) for DL communication with MAC-PHY split, below illustration of transport block (TB) encoding into encoded blocks in the CU and decoding in the DU.

on the availability of reliable open source platform. Hence, the user equipment, the CU and the DU are based on OAI implementation comprising full long-term evolution functionalities.

The experimental setup and data flow are shown in Figure 6.1. The CU and the DU run on two separate servers each with 8 GB RAM with a Xeon 1220, and 4 CPU cores. While the user equipment runs on a PC with 4 GB RAM with an Intel core i5. All hosts have a network interface card of a speed of 1 Gbps and operate on Ubuntu 14 with low latency kernel. The system operates on frequency division duplex on band 7 with 5 MHz bandwidth.

The CU entity accommodates MAC, RLC, PDCP, and RRC functionalities, while PHY and Radio Frequency functionalities are located in the DU. The CU entity is connected to 4 Ethernet fronthaul links each 3 meters long and with capacity 1 Gbps. The four links are connected to the 1 Gigabit switch. The output port of the switch is connected via 3 meters long Ethernet fronthaul, with capacity 1 Gigabit, to the DU. This latter is connected to the user equipment via USB 3 cables and two USRPs which are responsible for transmitting and receiving radio frequency signals. In this experimentation, the analysis is applied only on the DL.

To evaluate the performance of the system, once the radio access bearers for data flow are established, the IP packets are injected as user data at every frame into PDCP at CU entity. The PDCP, then, adds PDCP header to the IP packet then delivers it to RLC. RLC informs its buffer occupancy to MAC to determine the transport block (TB) size. MAC then requests data from RLC and composes the TB. Thereafter, the TB is processed by the coding block. The coding block has two main blocks (Figure 6.1):

- Splitter which split each TB into $k = 2$ equal blocks;
- Encoder that encodes the k blocks into $n = 4$ blocks of the same size using erasure code and sends each block into one of the n fronthaul paths;

The encoded blocks are then packetised to the Ethernet packet and transmitted to the DU via raw Ethernet socket over the fronthaul. One of the properties of fountain coding is the receiver can retrieve the original TB from any subset $\geq k$ of encoded blocks. Thus, the decoder starts decoding the message once the first k encoded blocks are received. If these blocks don't have enough information to retrieve the original message then the decoder has to wait for more blocks. In our analysis, the RTT latency in MPC is measured by considering the following events:

1. Network latency to transmit data request message from the DU PHY to MAC at CU entity.
2. Processing time of the request at the CU protocol stack.
3. Encoding time is time to encode the TB to 4 encoded blocks.
4. Packetisation of the encoded blocks at the MAC in CU entity.
5. Network latency to transmit the first $k + \epsilon$ blocks from the CU MAC to the DU PHY. Where $0 \leq \epsilon \leq n - k$.
6. Decoding time to reconstruct the original data.

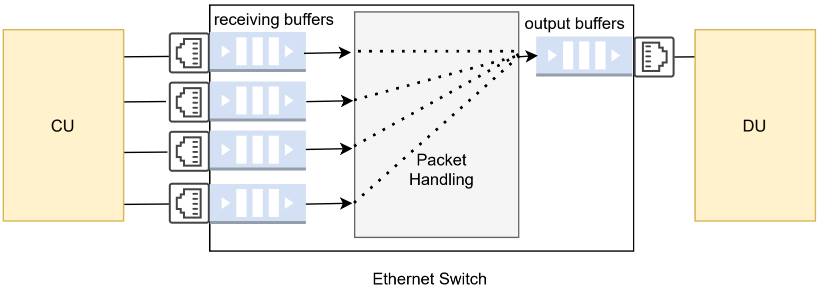
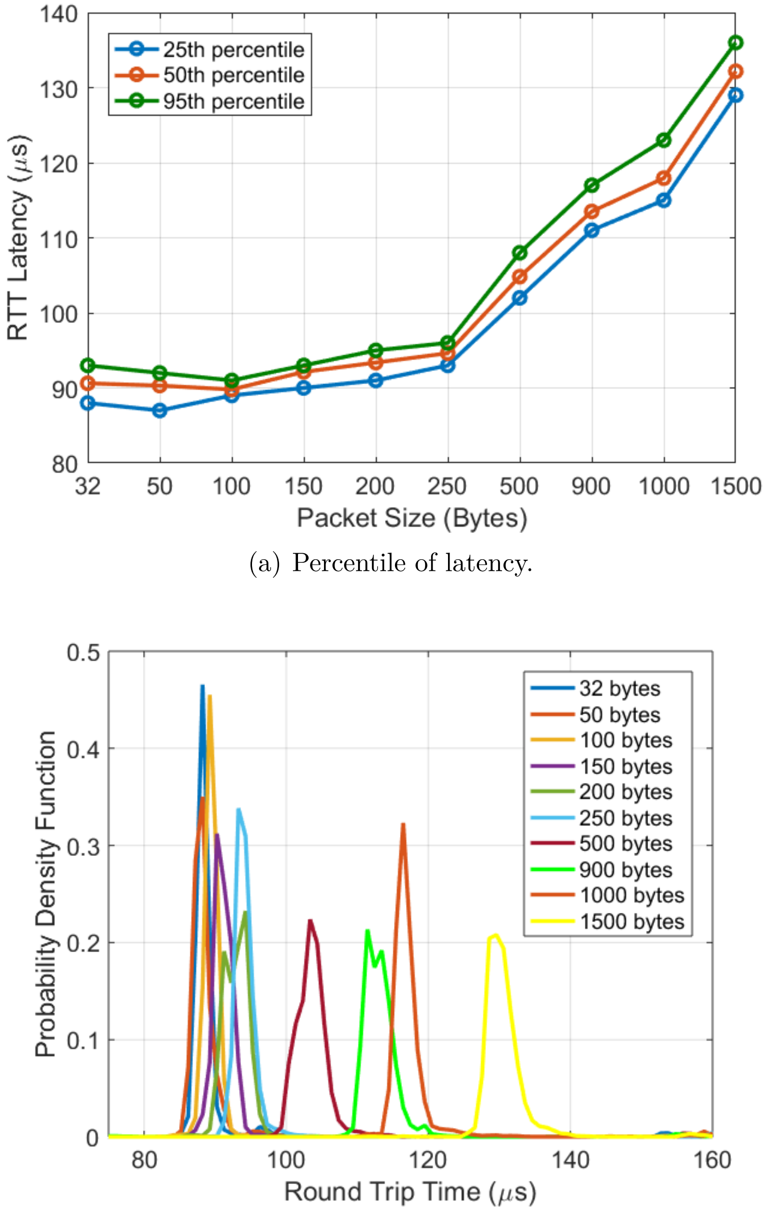


Fig. 6.2 Switch internal architecture.



(a) Percentile of latency.

(b) Distribution of latency.

Fig. 6.3 Latency on the Ethernet-based multi path fronthaul as a function of different packet sizes. (a) Percentile of latency as a function of the packet size. (b) Distribution of latency for different packet sizes.

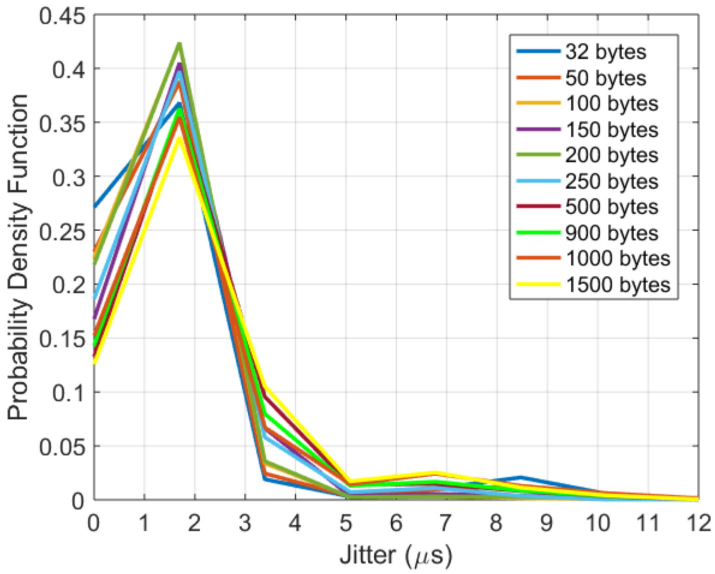


Fig. 6.4 Distribution of jitter on the fronthaul for different packet sizes.

6.2 Analysis of the Experimental Results

In this section, the performance of the system is evaluated by evaluating the results of the fronthaul latency, jitter, and reliability. How these quality metrics meet the requirements of the 3GPP are also evaluated. The 3GPP [3] specifies that the maximum allowed one-way latency for the MAC-PHY split should be 250 μs. The maximum allowed average RTT latency, therefore, is 500 μs (250×2). Since, as stated in Section 6.1, the RTT in this experiments is measured, the analysis considers 500 μs as the maximum acceptable latency on the fronthaul.

6.2.1 Analysis of latency and jitter for MPC

In the first experiment, the performance of MPC is evaluated in terms of latency and jitter. The performance is evaluated for scenarios with packet sizes of 32 bytes up to 1500 bytes to cover both URLLC and enhanced mobile broadband classes to understand how the packet size impacts the fronthaul latency. IP packets, thus, with different sizes, are injected in the PDCP layer in CU entity every frame. The protocol stack processes the packet then the MAC encodes it to 4 blocks and transmits them on the fronthaul. The latency is measured as the RTT as explained in Section 6.1. Figure 6.2 shows that 4 fronthaul paths are input to the switch, all of which must be directed to the same destination, i.e. DU. Therefore, packets may encounter queueing delay at the switch. Therefore $t_{\text{que}} \geq 0$ and there is 1

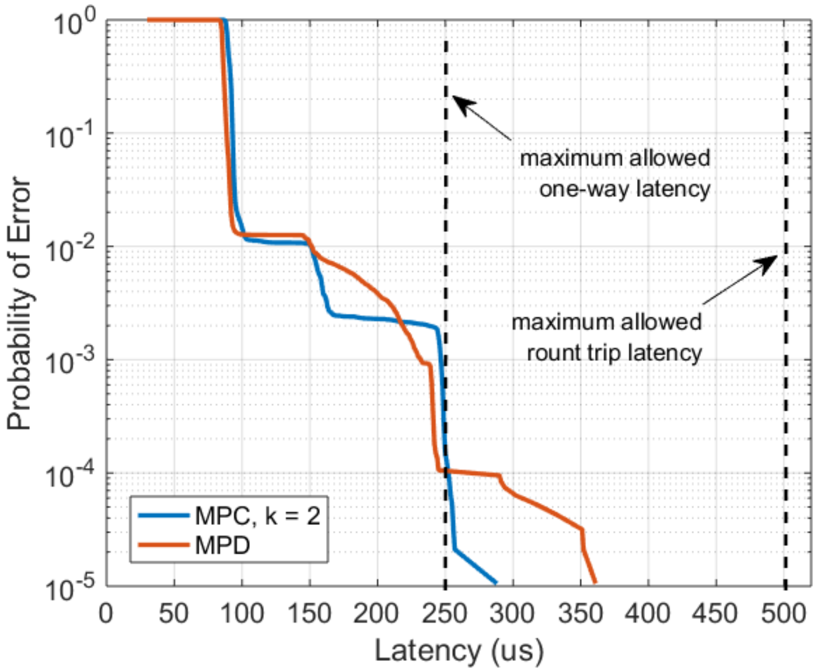


Fig. 6.5 Probability of error vs latency function for MPC with $k = 2$ and MPD. *The dashed line represents the maximum latency to be supported by the MAC-PHY split.*

switch in the experimental setup, hence $\mathrm{N}_{\mathrm{switch}}$=1 in Equation 2.3. The percentiles and pdf results of the RTT are shown in Figure 6.3.

Figure 6.3(a) shows results for different percentile values of the latency as a function of IP packet size. In general, as the IP packet size increases, percentile values of latency increases too; this is because larger packets take longer to be processed. The RTT for packet size 100 bytes is, however, less than for the 32 bytes and 50 bytes packet sizes. The reasoning behind this behaviour is that the packets 32 bytes and 50 bytes are split by the coding block in CU entity to 16 bytes and 25 bytes respectively. Even when the overhead header is added, the two packet sizes are still less than the minimum 64 bytes of Ethernet frames. Therefore, prior to transmitting these packets, padding is added by the Ethernet network card which adds extra delay to their RTT.

Figure 6.3(a) shows that MPC provides promising results as 95th percentile values remain below the maximum acceptable fronthaul latency for MAC-PHY split. Figure also shows that the 25th percentile, 50th percentile and 95th percentile values are of low variability as they are close to each other. The difference between 95th and 25th values is 8μs.

Focusing on 50th percentile result, if the packet size is less than 500 bytes, the increase in latency is low and it is around 4.43% from the packet size of 32

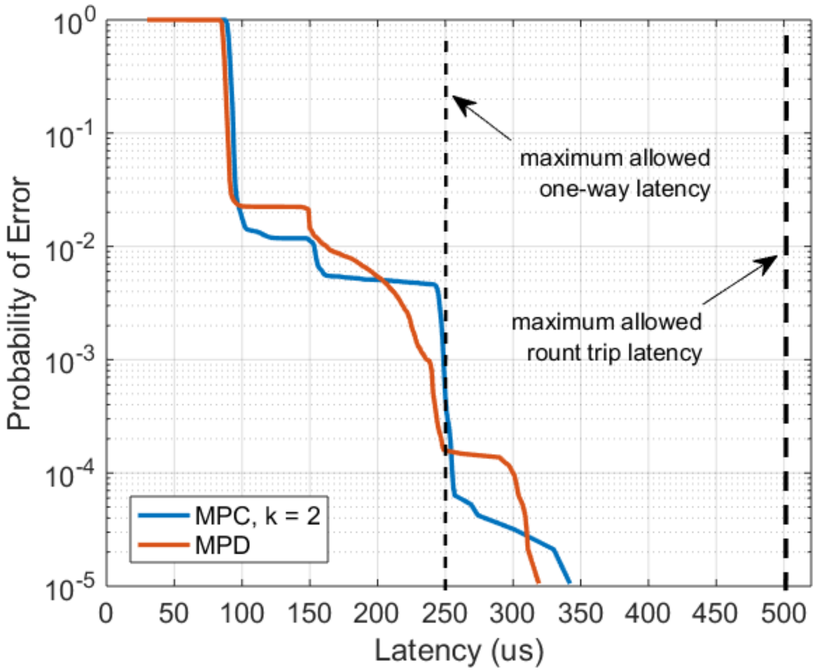


Fig. 6.6 Probability of Error vs latency functions for MPC with $k = 2$ and MPD when adding delay to fronthaul path 2. *The dashed line represents the maximum latency to be supported by the MAC-PHY split.*

bytes to the packet size of 250 bytes. On the contrary, with packet sizes larger than 500 bytes, the increase in latency is 26% for a packet 3 times larger than 500 bytes. It can be observed such kind of behaviour from Figure 6.3(b) whereby the probability density distributions of packet sizes from 32 bytes to 250 bytes are very close to each other. Whereas, for packet sizes ranging from 500 bytes to 1500 bytes, the probability density distributions are apart. Another observation from the Figure 6.3(b) is that the standard deviation increases with the size of the packet. For example, for packets of size 1500 bytes, the deviation is 10 μs which is approximately two times as large as for packets of size 32 bytes. This indicates that the larger the packet size, the higher the variations in the processing time of protocol stack, encoder/decoder and transport network are. This plot provides insight into how to choose the packet size to be transmitted on the fronthaul. For example, if the probability density distribution of the latency for a service should have a deviation of less than 8 μs, then the choice of packet size $\leq$ 250 would satisfy the requirement. It can be concluded with the Figures 6.3(a) and 6.3(b) that the average latency complies with the requirements of 3GPP, i.e. staying below 250 μs [3].

Further analysis of jitter can be found in Figure 6.4. Two factors primarily introduce jitter. First the packet switch network, as the switch can introduce variation in the packet delay when processing the packets. Second, is the operating system whereby the variation of system daemons and interrupts results in jitter. Observing from Figure 6.4, it can be seen that the distribution of jitter remains the same for different packet sizes and is just below 2 μs. The tail latency of Figure6.4 shows that latency values greater than 10 μs have a very low probability that is close to zero.

6.2.2 Comparison of MPC and MPD

In the second experiment, the analysis looks into the two diversity schemes, MPC and MPD and compare their performances. In particular, the analysis evaluates how the probability of error is improved by either of these two schemes and what the impact on the latency is. In this experiment, the focus is on the URLLC service, particularly on the tactile interaction in which reliability with low latency are key aspects. The payload sizes of the tactile interaction are typically ≤ 256 bytes and the error probability requirement is 10^{-5} [26].

Fixed IP packets of size 120 bytes are therefore injected in the PDCP every frame. Figure 6.5 presents the probability of error Vs latency. The probability of error defined as the probability that the measured RTT latency exceeds a predefined latency deadline (see Equation 2.12). It can be seen that for latency values less than 250 μs, the performance of MPC and MPD is generally the same. However, as latency increases to higher than 250 μs, MPC delivers lower error probability. MPC can achieve a latency reduction of 73 μs, with respect to MPD at the error probability of 10^{-5}. It is interesting to look specifically at the performance at the latency of 500 μs, which is the maximum acceptable RTT latency on the fronthaul for MAC-PHY split [3]. At this point, shown by the vertical black line on Figure 6.5, MPC clearly outperforms MPD and can achieve the error probability of 10^{-5} at 288 μs latency, leaving a margin of 212 μs to the maximum allowed RTT. Therefore, MPC is set to benefit from the diversity as well as the splitting of the packet into smaller blocks.

Now, the performance of multi-path fronthaul is evaluated by emulating delay in the fronthaul transport. In our next experimentation, a delay is added to the fronthaul path 2. Additional delay on path 2 will clearly affect the delivery of important information since link 2 carries information required for decoding data. To investigate whether the delay would affect the performance of MPC due to the waiting effect described in Section 6.1. The network delay is emulated by using *NetEM* to add a normal distribution delay of 100 μs ± 10 μs. Figure 6.6 shows the probability of error as a function of the latency. It can be seen that MPD and MPC start with the same probability of error, but after latency exceeds 100 μs, the performance of the two schemes alternates. However, MPD outperforms MPC at the error probability of 10^{-5}. Thus, in this case, where the extra delay is simulated in path 2, MPC is more affected than MPD because MPC needs to wait for more blocks to be received to successfully decode encoded blocks.

Here again, taking the maximum acceptable RTT latency of 500 μs on fronthaul for MAC-PHY split [3], which is shown by the vertical black line on Figure 6.6, MPC can achieve the error probability of 10^{-5} at latency 158 μs earlier than the maximum allowed RTT.

From the latency constraint perspective, the findings in this chapter demonstrate how the size of the packet has a significant effect on latency. To lower the latency, it is very important to choose the right size of the packet. Other than that, there are other aspects that need to be considered in the experiment deployed in Section 6.1 due to the use of the encoder that encodes every packet to multiple blocks. As such, the budget for latency is determined primarily by:

- CU internal process delay: time needed to encode and packetise data. The time taken depends essentially on capabilities of the hardware used. Therefore to control the latency, hardware features needs to be updated and optimised as well as deployment of specific devices, such as accelerator, should be considered.

- Switch queueing delay: time spent in the queuing at the switch aggregator (see Figure 6.2). The worst case scenario happens when all the four blocks arrive at the switch aggregator at the same time. Such a scenario is very likely to occur when the four blocks are serialised at the same time on four separate fronthaul paths with the same capacity. However, provided that there are 24 Mbps (1500 bytes $\times 8 \times 1000 \times n/k$) data incoming to the switch, the 1 Gbps link does not represent a bottleneck in the system. Nonetheless, to avoid switch port being loaded to an extent that there would be queuing, it could be more efficient to try adding another receiving port at the DU where two of the four paths are destined for port 1 and the other paths are destined for port 2.

6.3 Concluding Remarks

This paper validates Cloud-RAN model with the split between MAC and PHY over Ethernet with additional reliability using either repetition or channel coding over multiple paths. First, the performance of multi-path with coding (MPC) is analysed in terms of latency and jitter using different packet sizes. The results were satisfying in terms of average RTT latency being compliant with the requirements of 3GPP, i.e. staying below 500 μs. Then, the performance of MPC with multi-path with repetition (MPD) is compared from reliability perspective at the error probability of 10^{-5}. The experimental results show that MPC achieved an error probability of 10^{-5} at a latency lower than MPD, i.e. a latency reduction of 73 μs. However, when one of the paths is simulated with extra delay, MPC achieved the error probability of 10^{-5} at latency 20 μs later than MPD. Nevertheless, the two schemes can deliver the error probability of 10^{-5} while staying below the acceptable latency on the fronthaul for MAC-PHY split.

Chapter 7

Conclusions and Future Perspectives

In this Chapter, we summarise our contributions for Cloud-RAN with functional split over Ethernet-based fronthaul and propose some future works.

7.1 Conclusion

This study explores the flexible functional split in Cloud-RAN over Ethernet-based fronthaul using the end-to-end hardware testbed. Various functional splits have been made over the Ethernet fronthaul. The experimental models the traffic of the three 5G classes based on the 3GPP to represent real traffic. The thesis contains the findings of the analytic analyses as well as the Matlab simulations and the hardware experimental results.

The important findings are as follows:

- MAC and PHY functional split is feasible over the Ethernet as the latency results for different packet sizes are compliant with the requirements of 250 μs set by 3GPP. The throughput of the fronthaul split scales with the user data, therefore MAC-PHY split has the advantage of not being directly affected by some of the 5G technologies such as massive number of antennas, i.e. massive MIMO.

- Different functional split have different impact on delivering of 5G traffics from latency and jitter perspectives. Different applications need different functional splits to satisfy their requirements. Having different functional split enable to provide a platform supporting service delivery.

- Reliability of Ethernet-based fronthaul can be improved by by means of multi-path diversity and erasure coding while the latency remains below a strict latency bound required by this function split.
- The average latency and the error probability can be effectively reduced by adequately managing the multi-path fronthaul resources via orthogonal allocation transmission. These results can serve as a valuable guidance on how to effectively deploy multi-path fronthaul resources

7.2 Future Works

The experimental analysis using a hardware testbed provided in this thesis, was a big step towards the understanding of the impacts of packetisation and flexible functional split on supporting the envisioned 5G services. The followings are some future works that can be researched in this direction:

- 5G is expected to support a variety of services with heterogeneous requirements. As a result, the Cloud-RAN needs to be more flexible, and eventually more intelligence to support 5G and beyond 5G. Therefore, Cloud-RAN can take advantages of the machine learning and Artificial Intelligence (AI) capabilities to optimise the radio and system performance. These two approaches will help to continually collect data and KPIs from all the components of RAN. It is interesting to research how this intelligence can be used to select the best functional split for each scenario taking into account service requirements, traffic loads, RAN resources and the fronthaul network.
- Another important research is the virtualisation of the DU and the fronthaul switches together with the CU, which would make the whole RAN programmable and thus lead to an automated RAN. With this architecture, it would be interesting to evaluate various topology options, e.g. to place virtual CU and virtual DU at the same data centre or to place them at different data centres. It would also be interesting to exploit the slicing of the RAN and the fronthaul by leveraging virtualisation.
- Recently, the use of unmanned aerial vehicles (UAV) in the cellular networks has received an increasing interest. UAVs support Cloud-RAN architecture and, as such, they can be deployed as drone-based DU. An important direction is to implement Cloud-RAN in real drones, i.e. mount DU in drones, and

perform real-life experiments using various types of wireless fronthaul. Such experiments can provide insights into how drones can benefit from the advantages of Cloud-RAN, such as pooling, coordination scheduling and, more importantly, simpler and lighter base station drones, because only DUs are installed on drones.

- As the fronthaul can have a huge effect on the success of the fronthaul. It would also be important to explore different fronthaul types with new frequency bands. For example, experiment with millimetre-wave which is planned to be used for 5G communications. In addition, it would be useful to take a step forward with the approach proposed and validated in Chapters 5 and 6 respectively, by researching multi-path fronthaul links each with different types or capacities. In addition, different scheduling policies can also be implemented.

List of Abbreviations

1G	First Generation
3GPP	3rd Generation Partnership Project
BBU	Baseband Unit
CDF	Cumulative Distribution Function
CPRI	Common Public Radio Interface
CPU	Central Processing Unit
CU	Central Unit
DL	Downlink
eMBB	Enhanced Mobile Broadband
eNodeB	Evolved Node B
HARQ	Hybrid Automatic Repeat Request
FIFO	First-In-First-Out
FFT	Fast Fourier Transform
Gbps	Gigabit per second
IQ	In-Phase and Quadrature
KPI	Key Performance Indicator
LTE	Long Term Evolution
MAC	Medium Access Control
Mbps	Megabit per second

MCS	Modulation and coding schemes
MIMO	Multiple Input Multiple Output
mMTC	Massive Machine-Type Communications
MPD	Multi-Path Transport with Duplication
MPC	Multi Path Transport with Coding
NGMN	Next Generation Mobile Networks
OAI	Open Air Interface
O-RAN	Open-RAN
PDCP	Packet Data Convergence Protocol
PDF	Probability Density Function
PDU	Packet Data Unit
PHY	Physical
QAM	Quadrature amplitude modulation
QoS	Quality of Service
RAN	Radio Access Network
RLC	Radio Link Control Control
RRC	Radio Resource Control
RF	Radio Frequency
RRH	Remote Radio Head
RTT	Round Trip Time
RU	Radio Unit
SDR	Software Defined Radio
SP	Single-Path
USB	Universal Serial Bus

USRP	Universal Software Radio Peripheral
UE	User Equipment
UL	Uplink
URLLC	Ultra-Reliable Low Latency Communications

List of Symbols

GB	Gigabytes
Gbps	Gigabits per second
MHz	Megahertz
ms	Millisecond
μ**s**	Microsecond

Bibliography

[1] Massimo Condoluci and Toktam Mahmoodi, "Softwarization and virtualization in 5G mobile networks: Benefits, trends and challenges," *Computer Networks*, vol. 146, pp. 65 – 84, 2018. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1389128618302500

[2] NGMN, "Further Studies on Critical Cloud RAN Technologies," White Paper, NGMN, March 2015.

[3] 3GPP, "Study on New Radio Access Technology: Radio Access Architecture and Interface," 3GPP TR 38.801 V14.0.0 , Release 14, March 2017.

[4] L. Gavrilovska, V. Rakovic, A. Ichkov, D. Todorovski, and S. Marinova, "flexible c-ran: Radio technology for 5g," in *2017 13th International Conference on Advanced Technologies, Systems and Services in Telecommunications (TELSIKS).*

[5] China Mobile, "C-RAN: the Road Towards Green RAN," *White Paper*, vol. 2, 2011.

[6] "Common Public Radio Interface (CPRI): Interface Specification," V7.0, 2015.

[7] S. shew, "Transport network support of IMT-2020/5G," ITU Telecommunication Standardization Sector OF ITU, Tech. Rep., February 2018.

[8] P. Uthansakul and A. A. Khan, "Enhancing the Energy Efficiency of mmWave Massive MIMO by Modifying the RF Circuit Configuration," *Energies*, vol. 12, no. 22, p. 4356, Nov 2019. [Online]. Available: http://dx.doi.org/10.3390/en12224356

[9] R. Zi, X. Ge, J. Thompson, C.-X. Wang, H. Wang, and T. Han, "Energy Efficiency Optimization of 5G Radio Frequency Chain Systems," *IEEE Journal on Selected Areas in Communications*, vol. 34, no. 4, pp. 758–771, Apr. 2016.

[10] 3GPP, "Study on Small Cell Enhancements for E-UTRA and E-UTRAN – Higher layer aspects ," 3GPP TS 36.842 V0.2.0, Release 12, May 2013.

[11] NGMN, "5G RAN CU - DU Network Architecture, Transport Options and Dimensioning," White Paper, NGMN, April 2019, version 1.0.

[12] O-RAN, "Control, User and Synchronization Plane Specification," Technical Specification, O-RAN, 3 2019, version 01.00.

[13] O-RAN, "Management Plane Specification," Technical Specification, O-RAN, 3 2019, version 01.00.

[14] Emeka Obiodu, "THE 5G GUIDE," *GSMA*, April 2019.

[15] Small Cell Forum, "Backhaul technologies for small cells: Use cases, requirements and solutions," Small Cell Forum Document 049.07.02, February 2013.

[16] C. Dehos and J. L. González and A. D. Domenico and D. Kténas and L. Dussopt, "Millimeter-wave access and backhauling: the solution to the exponential data traffic increase in 5G mobile communications systems?" *IEEE Communications Magazine*, vol. 52, no. 9, pp. 88–95, 2014.

[17] Nomura, Hiroko and Ou, Hiroshi and Shimada, Tatsuya and Kobayashi, Takayuki and Hisano, Daisuke and Uzawa, Hiroyuki and Terada, Jun and Otaka, Akihiro, "First demonstration of optical-mobile cooperation interface for Mobile fronthaul with TDM-PON," *IEICE Communications Express*, vol. 6, 03 2017.

[18] CPRI, "eCPRI Interface Specification," Common Public Radio Interface, Interface Specification, 05 2019, v2.0.

[19] A. Checko, A. Popovska, M. Berger, and H. Christiansen, "Evaluating C-RAN fronthaul functional splits in terms of network level energy and cost savings," vol. 18, pp. 162–172, 04 2016.

[20] M. K. Al-Hares and P. Assimakopoulos and D. Muench and N. J. Gomes, "Modeling Time Aware Shaping in an Ethernet Fronthaul," in *GLOBECOM 2017 - 2017 IEEE Global Communications Conference*, 2017, pp. 1–6.

[21] T. Wan and B. McCormick and Y. Wang and P. Ashwood-Smith, "ZeroJitter: An SDN Based Scheduling for CPRI over Ethernet," in *2016 IEEE Global Communications Conference (GLOBECOM)*, 2016, pp. 1–7.

[22] S. Costanzo and I. Fajjari and N. Aitsaadi and R. Langar, "DEMO: SDN-based network slicing in C-RAN," in *2018 15th IEEE Annual Consumer Communications Networking Conference (CCNC)*, 2018, pp. 1–2.

[23] A. Checko and A. C. Juul and H. L. Christiansen and M. S. Berger, "Synchronization challenges in packet-based Cloud-RAN fronthaul for mobile networks," in *IEEE International Conference on Communication (ICC) Workshop*, June 2015.

[24] M. Dong, Z. Qiu, W. Pan, C. Chen, J. Zhang, and D. Zhang, "The design and implementation of ieee 1588v2 clock synchronization system by generating hardware timestamps in mac layer," in *2018 International Conference on Computer, Information and Telecommunication Systems (CITS)*, 2018, pp. 1–5.

[25] E. Lundqvist, "Timing and Synchronization over Ethernet," Ph.D. dissertation, Linkoping University, February 2015.

[26] 3GPP, "Technical Specification Group Services and System Aspects; Service requirements for the 5G system," 3GPP TS 22.261 V16.4.0 , Release 16, Jun 2018.

[27] Waqar and Kim, “Performance Improvement of Ethernet-Based Fronthaul Bridged Networks in 5G Cloud Radio Access Networks,” *Applied Sciences*, vol. 9, no. 14, p. 2823, Jul 2019. [Online]. Available: http://dx.doi.org/10.3390/app9142823

[28] M. Waqar, A. Kim, and P. K. Cho, “A Transport Scheme for Reducing Delays and Jitter in Ethernet-Based 5G Fronthaul Networks,” *IEEE Access*, vol. 6, pp. 46 110–46 121, 2018.

[29] P. Assimakopoulos and J. Zou and K. Habel and J. Elbers and V. Jungnickel and N. J. Gomes, “A Converged Evolved Ethernet Fronthaul for the 5G Era,” *IEEE Journal on Selected Areas in Communications*, vol. 36, no. 11, pp. 2528–2537, 2018.

[30] Seok-Hwan Park and Osvaldo Simeone and Shlomo Shamai, “Robust Baseband Compression Against Congestion in Packet-Based Fronthaul Networks Using Multiple Description Coding,” *Entropy*, vol. 21, p. 433, 2019.

[31] C. Y. Chang and R. Schiavi and N. Nikaein and T. Spyropoulos and C. Bonnet, “Impact of Packetization and Functional Split on C-RAN Fronthaul Performance,” in *2016 IEEE International Conference on Communications (ICC)*, May 2016, pp. 1–7.

[32] V. B. Vinnakota and N. Manne and A. Mondal and D. Sen and S. Chakraborty, “An Experimental Study of C-RAN Fronthaul Workload Characteristics: Protocol Choice and Impact on Network Performance,” in *2019 IEEE 89th Vehicular Technology Conference (VTC2019-Spring)*, 2019, pp. 1–7.

[33] P. Sehier and A. Bouillard and F. Mathieu and T. Deiss, “Transport Network Design for FrontHaul,” in *2017 IEEE 86th Vehicular Technology Conference (VTC-Fall)*, 2017, pp. 1–5.

[34] A. I. Salama and M. M. Elmesalawy, “Experimental OAI-based Testbed for Evaluating the Impact of Different Functional Splits on C-RAN Performance,” in *2019 Novel Intelligent and Leading Emerging Sciences Conference (NILES)*, vol. 1, 2019, pp. 170–173.

[35] S. Matoussi and I. Fajjari and N. Aitsaadi and R. Langar and S. Costanzo, “Joint Functional Split and Resource Allocation in 5G Cloud-RAN,” in *ICC 2019 - 2019 IEEE International Conference on Communications (ICC)*, 2019, pp. 1–7.

[36] M. Tohidi and H. Bakhshi and S. Parsaeefard, “Flexible Function Splitting and Resource Allocation in C-RAN for Delay Critical Applications,” *IEEE Access*, vol. 8, pp. 26 150–26 161, 2020.

[37] Haoran Mei and Limei Peng, “Flexible functional split for cost-efficient C-RAN,” *Computer Communications*, vol. 161, pp. 368 – 374, 2020. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0140366420318417

[38] A. M. Alba and J. H. G. Velásquez and W. Kellerer, “An adaptive functional split in 5G networks,” in *IEEE INFOCOM 2019 - IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*, 2019, pp. 410–416.

[39] C. Chang and N. Nikaein and R. Knopp and T. Spyropoulos and S. S. Kumar, “FlexCRAN: A flexible functional split framework over ethernet fronthaul in Cloud-RAN,” in *2017 IEEE International Conference on Communications (ICC)*, 2017, pp. 1–7.

[40] J. Yusupov and A. Ksentini and G. Marchetto and R. Sisto, “Multi-Objective Function Splitting and Placement of Network Slices in 5G Mobile Networks,” in *2018 IEEE Conference on Standards for Communications and Networking (CSCN)*, Oct 2018, pp. 1–6.

[41] B. Ojaghi and F. Adelantado and E. Kartsakli and A. Antonopoulos and C. Verikoukis, “Sliced-RAN: Joint Slicing and Functional Split in Future 5G Radio Access Networks,” in *ICC 2019 - 2019 IEEE International Conference on Communications (ICC)*, 2019, pp. 1–6.

[42] J. Tang and B. Shim and T. Chang and T. Q. S. Quek, “Incorporating URLLC and Multicast eMBB in Sliced Cloud Radio Access Network,” in *ICC 2019 - 2019 IEEE International Conference on Communications (ICC)*, 2019, pp. 1–7.

[43] Line M. P. Larsen and M. Berger and H. Christiansen, “Fronthaul for Cloud-RAN Enabling Network Slicing in 5G Mobile Networks,” *Wirel. Commun. Mob. Comput.*, vol. 2018, pp. 4 860 212:1–4 860 212:8, 2018.

[44] RFC 2681, “A Round-trip Delay Metric for IPPM,” Tech. Rep., Sep. 1999.

[45] RFC 2679, “A One-way Delay Metric for IPPM,” Tech. Rep., Sep. 1999.

[46] 3GPP, “Multimedia Broadcast/Multicast Service (MBMS),” 3GPP TS 26.347 V14.0.0 , Release 14, Tech. Rep. 26.347, April 2017.

[47] RFC 5053, “Raptor Forward Error Correction Scheme for Object Delivery,” Tech. Rep., Oct. 2007. [Online]. Available: https://tools.ietf.org/html/rfc5053

[48] G. Joshi and Y. Liu and E. Soljanin, “Coding for fast content download,” in *50th Annual Allerton Conference on Communication, Control, and Computing (Allerton)*, October 2012, pp. 326–333.

[49] Y. Cui and L. Wang and X. Wang and H. Wang and Y. Wang, “FMTCP: A Fountain Code-Based Multipath Transmission Control Protocol,” *IEEE/ACM Trans. on Networking*, vol. 23, no. 2, pp. 465–478, 2015.

[50] N. I. Sulieman and E. Balevi and K. Davaslioglu and R. D. Gitlin, “Diversity and network coded 5G fronthaul wireless networks for ultra reliable and low latency communications,” in *IEEE PIMRC*, October 2017.

[51] T. Ding and X. Yuan and S. C. Liew, “Network-coded fronthaul transmission for cache-aided C-RAN,” in *IEEE International Symposium on Information Theory (ISIT)*, June 2017.

[52] Chih-Lin, I and Huang, Jinri and Duan, Ran and Cui, Chunfeng and Jiang, Jesse Xiaogen and Li, Lei, "Recent Progress on C-RAN Centralization and Cloudification," *IEEE Access*, vol. 2, pp. 1030–1039, 2014.

[53] U. Dötsch and M. Doll and H. P. Mayer and F. Schaich and J. Segel and P. Sehier, "Quantitative Analysis of Split Base Station Processing and Determination of Advantageous Architectures for LTE," *Bell Labs Technical Journal*, vol. 18, no. 1, pp. 105–128, Jun 2013.

[54] M. Jaber and M. A. Imran and R. Tafazolli and A. Tukmanov, "5G Backhaul Challenges and Emerging Research Directions: A Survey," *IEEE Access*, vol. 4, pp. 1743–1766, 2016.

[55] Navid Nikaien, "OpenAirInterface Simulator Emulator," White Paper, OpenAir5GLab-EURECOM, July 2015.

[56] 3GPP, "5G;NR;Physical layer procedures for data ," 3GPP TS 38.214, V15.3.0, Release 15, 10 2018.

[57] NGMN, "Further Studies on Critical Cloud RAN Technologies," White Paper, NGMN, Feb. 2020.

[58] M. Bennis, M. Debbah, and H. V. Poor, "Ultrareliable and Low-Latency Wireless Communication: Tail, Risk, and Scale," *Proceedings of the IEEE*, vol. 106, no. 10, pp. 1834–1853, 2018.

[59] R. Ferrus and O. Sallent and J. Perez-Romero and R. Agusti, "On 5G Radio Access Network Slicing: Radio Interface Protocol Features and Configuration," *IEEE Communications Magazine*, vol. 56, no. 5, pp. 184–192, 2018.

[60] X. Foukas, M. K. Marina, and K. Kontovasilis, "Orion: RAN Slicing for a Flexible and Cost-Effective Multi-Service Mobile Network Architecture," in *Proceedings of the 23rd Annual International Conference on Mobile Computing and Networking*, ser. MobiCom '17. New York, NY, USA: Association for Computing Machinery, 2017, p. 127–140. [Online]. Available: https://doi.org/10.1145/3117811.3117831

[61] Nomor Research Newsletter, "3GPP 5G Adhoc: RAN Internal Functional Split," 3GPP Newsletter, Nomor Research Newsletter, Jan. 2017.

[62] 3GPP, "Evolved Universal Terrestrial Radio Access (E-UTRA); Physical layer procedures," 3GPP TS 36.213, version 8.8.0, Release 8, Oct. 2009.

[63] 5G-PPP, "5G PPP use cases and performance evaluation models," Apr. 2016. [Online]. Available: https://5g-ppp.eu/wp-content/uploads/2014/02/5G-PPP-use-cases-and-performance-evaluation-modeling_v1.0.pdf

[64] 3GPP, "Traffic Model for legacy GPRS MTC," GP 160060, 3GPP GERAN meeting 69, Feb. 2016.

[65] 3GPP, "Evolved Universal Terrestrial Radio Access (E-UTRA); Further advancements for E-UTRA physical layer aspects," 3GPP TR 38.814, version 9.2.0, Release 9, Mar. 2017.

[66] 3GPP, "RAN Improvements for Machine-type Communications," 3GPP TR 37.868, version 11.0.0, Release 10, Oct. 2011.

[67] Y. P. E. Wang and X. Lin and A. Adhikary and A. Grovlen and Y. Sui and Y. Blankenship and J. Bergman and H. S. Razaghi, "A Primer on 3GPP Narrowband Internet of Things," *IEEE Commun. Magazine*, vol. 55, no. 3, pp. 117–123, Mar. 2017.

[68] B. Guo, W. Cao, A. Tao, and D. Samardzija, "LTE/LTE-A signal compression on the CPRI interface," *Bell Labs Technical Journal*, vol. 18, no. 2, pp. 117–133, 2013.

[69] Dimitrios Pliatsios and P. Sarigiannidis and S. Goudos and G. Karagiannidis, "Realizing 5G vision through Cloud RAN: technologies, challenges, and trends," *EURASIP Journal on Wireless Communications and Networking*, vol. 2018, pp. 1–15, 2018.

[70] 3GPP, "NR; NR and NG-RAN Overall Description; Stage 2," 3GPP TS 38.300, V15.0.0, Release 15, December 2017.

[71] N. J. Gomes, P. Sehier, H. Thomas, P. Chanclou, B. Li, D. Munch, P. Assimakopoulos, S. Dixit, and V. Jungnickel, "Boosting 5g through ethernet: How evolved fronthaul can take next-generation mobile to the next level," *IEEE Vehicular Technology Magazine*, vol. 13, no. 1, pp. 74–84, 2018.

[72] GABRIAL BROWN, "New Transport Network Architectures for 5G RAN," White Paper, Fujitsu.

[73] J. Tan and B. T. Swapna and N. B. Shroff, "Retransmission Delays With Bounded Packets: Power-Law Body and Exponential Tail," *IEEE/ACM Trans. on Networking*, vol. 22, no. 1, pp. 27–38, 2014.

[74] J. J. Nielsen, R. Liu, and P. Popovski, "Ultra-Reliable Low Latency Communication Using Interface Diversity," *IEEE Trans. on Communications*, vol. 66, no. 3, pp. 1322–1334, 2018.

[75] J. J. Nielsen and R. Liu and P. Popovski, "Optimized Interface Diversity for Ultra-Reliable Low Latency Communication (URLLC)," in *IEEE Globecom*, December 2017, pp. 1–6.

[76] M. Dohler and et. al., "Internet of skills, where robotics meets AI, 5G and the Tactile Internet," in *European Conference on Networks and Communications (EuCNC)*, June 2017.