DETERMINISTIC6G

Digest on Latency Measurement Framework

D4.2

The DETERMINISTIC6G project has received funding from the European Union’s Horizon Europe research and innovation programme under grant agreement no 1010965604.

Digest on Latency Measurement Framework

Grant agreement number: Project title: Project acronym: Project website: Programme:	101096504 Deterministic E2E communication with 6G DETERMINISTIC6G Deterministic6g.eu EU JU SNS Phase 1
Deliverable type: Deliverable reference number: Contributing workpackages: Dissemination level: Due date: Actual submission date:	Public Software Release D4.2 WP4 PUBLIC 31-03-2024 28-03-2024
Responsible organization: Editor(s): Version number: Status:	KTH Gourav Prateek Sharma and James Gross v1.0 Final
Short abstract:	This deliverable describes the data collection framework, which is used to collect measurement data in the 5G system. It provides an overview of the design and implementation of the data collection framework. The proposed framework is capable of conducting in-depth latency measurements across both commercial off-the-shelf (COTS) 5G setups and those leveraging OpenAirInterface (OAI).
Keywords:	5G, 6G, software, latency, packet delay, retransmissions, PTP, measurements, RAN

Contributor(s):	Samie Mostafavi (KTH) Gourav Prateek Sharma (KTH) James Gross (KTH)

Revision History

22/02/2024	Draft version for the first internal review
15/3/2024	First internal draft
26/03/2024	Final version

Disclaimer

This work has been performed in the framework of the Horizon Europe project DETERMINISTIC6G co-funded by the EU. This information reflects the consortium's view, but the consortium is not liable for any use that may be made of any of the information contained therein. This deliverable has been submitted to the EU commission, but it has not been reviewed and it has not been accepted by the EU commission yet.

Executive summary

In recent times, the packet delay and Packet Delay Variation (PDV) has become a key performance indicator (KPI) for networks, driven by the emergence of time-sensitive applications in areas such as adaptive manufacturing, exoskeletons, XR, and smart farming. As we transition from 5G to 6G, the necessity for highly reliable and low-latency communications intensifies to support these critical applications.

The DETERMINISTIC6G project aims to realize dependable time-critical communications in the future 6G networks through a set of enablers. A crucial aspect of enablers involves collecting extensive latency data from existing 5G networks. Existing network measurement frameworks do not fully capture the complexity of end-to-end packet delay or the impact of various 5G mechanisms that contribute to the total end-to-end packet delay.

To overcome these obstacles, we have proposed a measurement framework that facilitates detailed latency measurements across both Commercial-off-the-Shelf (COTS) 5G setups and those based on the OpenAirInterface (OAI) platform. This document details the framework's design and implementation. Initial results from sample measurements conducted on both COTS 5G and OAI 5G setups validate the framework's effectiveness and the depth of insights it provides. Moreover, the framework's usefulness in systematically optimizing end-to-end packet delays is demonstrated.

The data collected using this measurement framework will serve two purposes in the project: (i) developing data-driven simulation models of 6G DetCom nodes and (ii) building a dataset to train, validate and test data-driven latency prediction models.

Contents

1 Introduction

Traditionally, bandwidth utilization has been a Key Performance Indicator (KPI) for communication systems as most applications require that the network delivers the requirements in terms of average data rate. The improvements in the previous generation of mobile networks have been largely focused on improving the average KPI values and do not target the improvements of higher quantiles of KPIs, e.g., ensuring packet delay of 1 ms at 99.9999% reliability. With the advent of the 5G/5G-Advanced (5G-Adv) era, KPIs like packet delay[1] and Packet Delay Variation (PDV) have become very significant. This is true, especially in the context of time-critical applications that require guarantees on packet delay which are often combined with extreme reliability levels. These applications, ranging from Extended Reality (XR), exoskeletons, adaptive manufacturing and smart farming require stringent adherence to latency specifications in order to function safely and efficiently [DET23-D11]. 6G will need to support such applications so the significance of packet delay and PDV will only increase in the 6G era.

Depending on the network measurement KPIs of interest, a variety of network measurement tools have been developed for communication systems. Despite the importance of having accurate packet delay measurements, standard network measurement methods and tools have considerable limitations for measuring packet delay and its variations in 5G networks, as discussed later. In this report, we describe a latency measurement framework tailored for 5G/5G-Adv networks. This framework is designed to conduct comprehensive latency measurements and data collection across Commercial-Off-The-Shelf (COTS) 5G and OpenAirInterface (OAI) 5G systems. This report offers an overview of the challenges of the existing measurement tools and outlines our methodology for accurate latency measurements in 5G/5G-Adv networks. It is worth pointing out that this report is not meant as documentation of the measurement framework or user's guide nor does it provide instructions to set up the hardware and software components of the framework. The report accompanies the deliverable D4.2, which is a software framework aimed at performing latency measurements on COTS 5G and OAI 5G implementations. The software components constituting the latency measurement framework can be found in the project's public Github repository. We also provide links to the Zenodo repositories for the sample measurements collected using the developed framework on the two 5G setups. The links to the software and sample measurements are listed in Table 1.

Table 1 An overview of the software components and sample measurements relevant to the latency measurement framework.

Component name	License	Links
Network Latency Measurement Tool (NLMT)	GNU General Public License v2.0	Github Link Zenodo Link
Latency Measurement Framework	Apache License 2.0	Github Link Zenodo Link
Sample COTS 5G measurements	Creative Commons Attribution 4.0 International	Zenodo Link

[1] In this document, we refer to packet delay as the total time taken for a packet to travel from the communication unit on the sender's side to the communication unit on the receiver's side.

Sample OAI 5G measurements	Creative Commons Attribution 4.0 International	Zenodo Link

Next, we provide the context of the DETERMINISTIC6G project. After that, an overview of the existing tools relevant to latency measurements is presented along with their shortcomings in Section 1.2. The interaction between the latency measurement framework and other work packages of the project is described in Section 1.3. The outline of the rest of the document is presented in Section 1.4.

1.1 DETERMINISTIC6G Approach

Digital transformation of industries and society is resulting in the emergence of a larger family of time-critical services with needs for high availability which present unique requirements distinct from traditional Internet applications like video streaming or web browsing. Time-critical services are already known in industrial automation; for example, an industrial control application that might require an end-to-end “over the loop” (i.e., from the sensor to the controller back to the actuator) latency of 2 ms and with a communication service requirement of 99.9999% [3GPP16-22261]. In the same way, with the increasing digitalization similar requirements are appearing in a growing number of new application domains, such as extended reality, autonomous vehicles, and adaptive manufacturing [DET23-D11]. The general long-term trend of digitalization leads towards a Cyber-Physical Continuum where the monitoring, control and maintenance functionality is moved from physical objects (like a robot, a machine or a tablet device) to a compute platform at some other location, where a digital representation – or digital twin – of the object is operated [WPP+22]. Such Cyber Physical System (CPS) applications need a frequent and consistent information exchange between the digital and physical twins. Several technological developments in the Information and Communications Technology (ICT) sector drive this transition. The proliferation of (edge-) cloud compute paradigms provide new cost-efficient and scalable computing capabilities, that are often more efficient to maintain and evolve compared to embedded compute solutions integrated into the physical objects. It also enables the creation of digital twins as a tool for advanced monitoring, prediction, automation of system components, and improved coordination of systems of systems. New techniques based on Machine Learning (ML) can be applied in application design, that can operate over large data sets and profit from scalable compute infrastructure. Offloading compute functionality can also reduce spatial footprint, weight, cost, and energy consumption of physical objects, which is particularly important for mobile components, like vehicles, mobile robots, or wearable devices. This approach leads to an increasing need for communication between physical and digital objects, and this communication can span over multiple communication and computational domains. Communication in this cyber-physical world often includes closed-loop control interactions which can have stringent end-to-end KPI (e.g., maximum packet delay and PDV) requirements over the entire loop. In addition, many operations may have high criticality, such as business-critical tasks or even safety relevant operations. Therefore, it is necessary to provide dependable time-critical communications which provide service-assurance to achieve the agreed service requirements.

In the past, time-critical communication has mainly been prevalent in industrial automation scenarios with special compute hardware like Programmable Logic Controller (PLC), and is based on a wired communication system, such as EtherCat and Powerlink, which is limited to local and isolated network domains which is configured according to the specific purpose of the local applications [ECAT][PLNK]. With the standardization of Time-Sensitive Networking (TSN) and Deterministic Networking (DetNet), similar capabilities are being introduced into the Ethernet and IP networking technologies, which thereby provide a converged multi-service network allowing time critical applications in a managed network infrastructure aiming for consistent performance with zero packet loss and guaranteed low and bounded latency [TSN][DETNET]. The underlying principles are that the network elements (i.e., bridges or routers) and the PLCs can provide a consistent and known performance with negligible stochastic variation, which allows to manage the network configuration according to the needs of time-critical applications with known traffic characteristics and requirements.

It turns out that several elements in the digitalization journey introduce characteristics that deviate from the assumptions that are considered as baseline in the planning of deterministic networks. There is often an assumption for compute and communication elements, and applications, that any stochastic behavior can be minimized such that the time characteristics of the element can be clearly associated with tight minimum/maximum bounds. Cloud computing offers efficient and scalable computing resources, but introduces uncertainty in execution times. Wireless communications provide flexibility and simplicity, however they contain inherently stochastic components that lead to packet delay variations exceedingly significant compared to those found in wired counterpart. Additionally, emerging applications incorporate novel technologies (e.g. ML-based or machine-vision-based control) where the traffic characteristics deviate from the strictly deterministic behavior of old-school control [SPS+23]. In addition, it is expected that there will be an increase in dynamic behavior, where characteristics of applications, and network or compute elements may change over time in contrast to a static behavior that does not change during runtime. It turns out that these deviations of stochastic characteristics make traditional approaches to planning and configuration of end-to-end time-critical communication networks such as TSN or DetNet, fall short in their performance regarding service performance, scalability, and efficiency. Instead, a revolutionary approach to the design, planning and operation of time-critical networks is needed, which fully embraces the variability but also dynamic changes that come at the side of introducing wireless connectivity, cloud compute and application innovation. DETERMINISTIC6G has an objective to address these challenges, including the planning of communications and compute resource allocation for diverse time-critical services end-to-end over multiple domains, providing efficient resource usage and a scalable solution [SPS+23].

DETERMINISTIC6G takes a novel approach towards converged future infrastructures for scalable cyber-physical systems deployment. With respect to networked infrastructures, DETERMINISTIC6G advocates (I) the acceptance and integration of stochastic elements (like wireless links and computational elements) with respect to their stochastic behavior captured through either short-term or longer-term envelopes. Monitoring and prediction of KPIs, for instance latency or reliability, can be leveraged to make individual elements plannable despite a remaining stochastic variance. Nevertheless, system enhancements to mitigate stochastic variances in communication and compute elements are also developed. (II) Next, DETERMINISTIC6G attempts to manage the entire end-to-end interaction loop (e.g. the control loop from the sensor to the controller to the actuator) with the underlying stochastic characteristics, especially while embracing the integration of compute elements. (III) Finally, due to unavoidable stochastic degradations of individual elements, DETERMINISTIC6G advocates allowing for adaptation between applications running on top such converged and managed network infrastructures. The idea is to introduce flexibility in the application operation such that its requirements can be adjusted at runtime based on prevailing system conditions. This encompasses a larger set of application requirements that (a) can also accept stochastic end-to-end KPIs, and (b) that possibly can adapt end-to-end KPI requirements at run-time in harmonization with the networked infrastructure. DETERMINISTIC6G builds on a notion of time-awareness, by ensuring accurate and reliable time synchronicity while also ensuring security-by-design for such dependable time-critical communications. Generally, we extend a notion of deterministic communication, where all behavior of network and compute nodes and applications are pre-determined, towards dependable time-critical communication, where the focus is on ensuring that the communication (and compute) characteristics are managed in order to provide the KPIs and reliability levels that are required by the application. DETERMINISTIC6G facilitates architectures and algorithms for scalable and converged future network infrastructures that enable dependable time-critical communication end-to-end, across domains, including 6G.

1.2 Related Tools

The key purpose of the latency measurement framework developed within the DETERMINISTIC6G project is to provide a comprehensive dataset of latency measurements collected on COTS or OAI 5G systems. Due to several limitations, the existing network measurement tools cannot be used to fulfill our objectives with respect to data collection in DETERMINISTIC6G.

1.2.1 General Delay measurement tools

In Table 2, we list the widely used network measurement tools that are typically employed in communication networks to collect network KPIs including latency measurement. The operation mode for these tools is typically active, i.e., they generate and inject control data into the network in order to measure network parameters such as delay, delay variations and packet loss.

Table 2 An overview of general delay measurement tools.

Tool	Description	Reference
iperf3	- Mainly used for TCP/UDP throughput measurements - RTT delay, PDV and jitter measurements	[IPRF]
ping	- Actively measures round-trip time. - Uses Internet Control Message Protocol (ICMP) echo requests and reply	[PNG]
fping	- ping for multiple hosts - Uses ICMP	[FPNG]
IRTT	- Active round-trip measurements for delay, delay variation, packet losses - Client-server deployment - detailed statistics and traffic emulation	[IRTT]
Moongen	- High-speed packet generation using DPDK - Precise hardware timestamping for accurate delay and jitter	[EGR+15]

1.2.2 Delay Measurements tools for 5G

The above-mentioned tools have been widely used to perform typical KPI measurements (e.g., throughput, delay and packet loss) in networks. In addition to these general-purpose tools, there are tools developed to carry out specific measurements (including delay) for 5G networks. An overview of these tools can be found below in Table 3.

Table 3 An overview of delay measurement tools for 5G.

Tool	Features	Reference
5GGrowth	- Periodic one-way delay (OWD) evaluation combined with clock offset estimation (i.e., the clock difference between client and server) - Prometheus exporters for data collection	[5GR20-D42]
LatSeq	- LTE basestation internal packet delay measurement - Lightweight timestamping and logging	[RFH+21]

1.2.3 Purpose of the Latency Measurement Framework

The existing network measurement tools listed in Table 2 have been widely used for basic round-trip time (RTT) measurements. However, there are several limitations of these tools with respect to our requirement. Firstly, tools like ping and irtt primarily focus on measuring end-to-end RTT only. This limitation comes from the fact that they do not rely on time synchronization between the sender and receiver of messages. For basic network health and connectivity checks where detailed latency analysis is not critical these measurements are sufficient. However, in the context of time-critical communications it is crucial to accurately measure OWD and its variations [AAB+22]. Furthermore, as these tools are not designed specifically for 5G measurements, these tools fall short in capabilities such as capturing various delay components along the end-to-end path and correlating these with the relevant network conditions as discussed in Section 2.1.

There are several tools developed specifically for latency measurements in 5G networks as listed in Table 3. The end-to-end unidirectional link latency evaluator proposes an interesting solution to estimate the end-to-end OWD in the 5Growth monitoring platform. This solution does not require the same clock reference at the sender or receiver instead periodically performs clocks offset estimation and adjust the OWD measurements. This solution does not assume any time synchronization in the network though efficient time distribution solutions, e.g., PTP, are being widely deployed in 5G networks. Furthermore, measurements of only OWD delay are not sufficient. Capturing the accurate decomposition of the end-to-end OWD into its subsequent components (e.g., Radio Access Network (RAN) delay and core delay) is important for network analytics as well as to optimize the overall performance [MTS+24]. Latseq addresses this aspect to an extent by tracing packet traversal through the different layers inside the OpenAirInterface LTE stack. However, it still is not capable of decomposing *end-to-end* delay between the application endpoints spanning the UE, the RAN and the core. In addition to delay decomposition, recording network conditions (with timestamps) during measurements along with delays is essential for creating comprehensive datasets for data-driven latency prediction approaches [DET23-D21][MSG+23]. For the above-mentioned tools there are no inbuilt mechanisms to capture network conditions and systematically correlate them with the measured packet delay values.

Lastly, these tools (in their original form) lack capabilities for automation and are not well-suited for prolonged measurement sessions, which are often necessary for capturing latency variations over long periods of time. Prolonged measurement sessions are important to capture rare network or traffic events and their impact on OWD and its decomposition. For instance, a data segment experiencing more than two retransmission attempts might account for only 0.1% of all transmitted segments. Capturing these rare events with probabilities less than 10^{-3} is crucial to accurately characterize the tail of the 5G delay distribution and therefore requires running prolonged measurement sessions.

The purpose of our latency measurement framework is to overcome these limitations and provide a comprehensive and automated approach to latency measurement in 5G systems compared to the existing measurement frameworks. This framework is uniquely well-suited for collecting essential data to serve two requirements in the project. Firstly, it enables the gathering of detailed 5G system latency measurements, including various latency components and associated network conditions. This data is essential for training data-driven latency prediction models [DET23-D21]. Moreover, the latency measurements collected using the framework are to be used to develop wireless latency models of the simulator to implement a realistic simulation model of the 6G-DetCom bridge [DET23-D41].

1.3 Relation to other Work Packages

The developed latency measurement framework has various interlinkages with other tasks in the DETERMINISTIC6G project as shown in Figure 1.1. A comprehensive analysis of latency in 5G was presented in [DET23-D21]. The breakdown of 5G user plane latency serves as input to the design of the latency measurement framework by providing insights about major delay contributions. The measurements collected using the developed latency measurement framework are taken as input within two tasks in the project. First, they form the basis for the development of *data-driven latency prediction models*. The data collected using the latency measurement framework is used to train, test, and validate the developed latency predictors. Additionally, in WP4, they contribute to the development of *simulation models of 6GDetCom nodes*, representing the behavior of 5G/5G-Adv system using the latency measurement data collected on a 5G testbed.

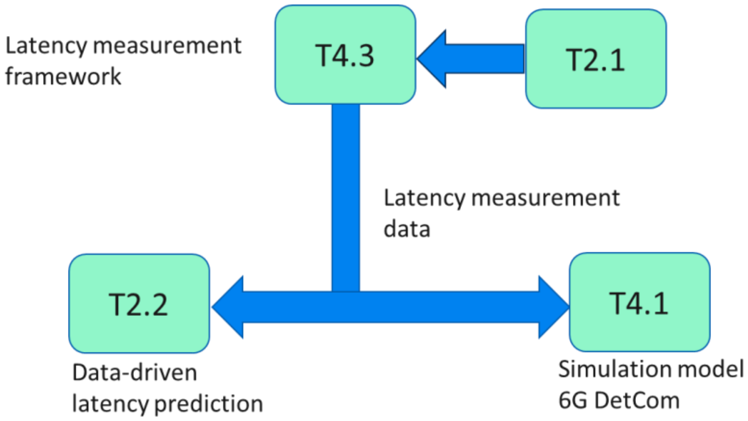

Latency measurement framework
T4.3
T2.1
Latency measurement data
T2.2
T4.1
Data-driven latency prediction
Simulation model 6G DetCom

Figure 1.1 Relationship of the latency measurement framework to other tasks in the project.

1.4 Structure of the Document

The rest of this document is organized as follows:

Section 2 provides a description of the latency measurement framework. First, an overview of the essential components of the end-to-end latency in 5G systems is presented. Next, we describe the design and implementation of the latency measurement framework and the software and hardware components of the framework.

Sections 3 and 4 detail the setup and procedures for data collection on COTS 5G and OAI 5G systems using the developed latency measurement framework, respectively. These sections cover the measurement setup and the high-level steps involved in the data collection process, along with showcasing a few samples of measurement outcomes.

The document concludes in Section 5, where it summarizes the document and discusses prospective future directions for the latency measurement framework.

2 Framework Description

In this section, we describe the design and implementation of the latency measurement framework. The latency measurement framework is mainly aimed at measuring 5G user plane packet delay and its variation. The user plane packet delay can either be measured as RTT or as OWD in the UL direction or the DL direction. Before discussing the design and implementation, we first provide an overview of the breakdown of the end-to-end 5G packet delay. Along with packet delay measurements, it is pertinent to also record network conditions that correspond to the measured delay. To this end, the framework allows the collection of a vast range of network conditions as discussed next.

2.1 5G Latency Breakdown

In contrast to the previous generations of mobile networks, packet delay and packet delay variation have been important KPIs in 5G. Therefore, it is important to study the end-to-end packet delay and its constituent components as well as the network mechanisms that impact these constituent components. A comprehensive analysis of 5G transmission latency has been presented earlier in D2.1. In this deliverable, we only present the overview of 5G latency analysis. This latency analysis will serve as input to the design rationale of the latency measurement framework.

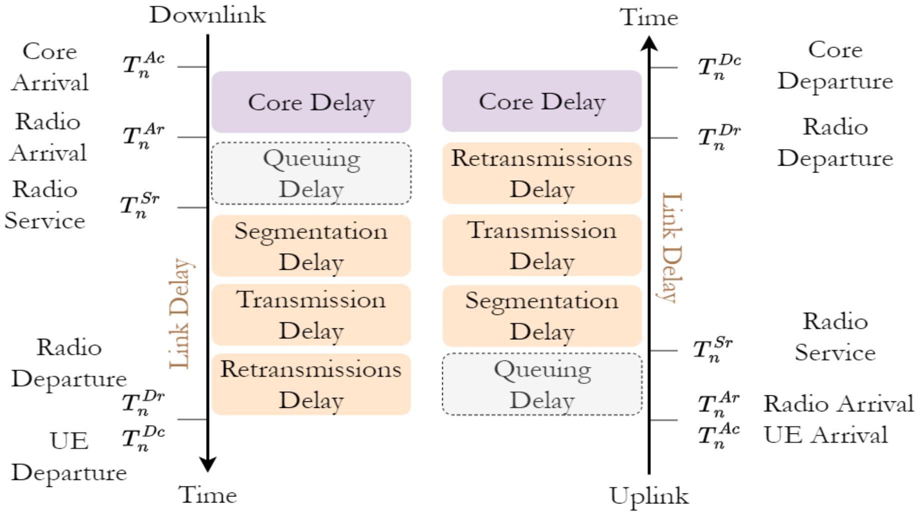


Figure 2.1 Components of end-to-end OWD in 5G.

The decomposition of packet delay in 5G can be done by distinguishing the two main domains in 5G networks, i.e., the RAN and the core. The delay experienced by a packet in the core network along the backhaul between RAN and the UPF is referred here as core delay. Core delay is significant in scenarios where the User Plane Function (UPF) / gateway of the network is several hops away from the RAN. Core delay can be measured by taking the difference between two timestamps corresponding: packet's departure from the UPF and packet's departure from the RAN as shown in Figure 2.1. For non-public 5G network deployments, core delay can be assumed to be fixed and small as compared to the RAN delay [RSI+21].

In contrast to the core, RAN contributes significantly to the packets traversing 5G network due to the stochastic nature of the wireless link. The packet delay in RAN can be further divided into queuing delay and link delay [3GPP18-94]. The queuing delay is the result of packets waiting to be transmitted on the wireless channel, where radio resources (e.g., Resource Blocks) might be occupied. A packet in an RLC queue has to wait for the packets in front of it to be serviced by the MAC layer. In addition to this delay, there can also be a waiting time for transmission grants, which also includes the delay in the control plane message exchange to allocate transmission grants. This delay component of the queueing delay is referred to as frame-alignment delay. Following queuing, link delay accounts for the time from when a packet is prepared for transmission to its successful reassembly at the receiver's end. This includes three components: segmentation delay, retransmission delay and processing delay.

Segmentation delay occurs when the Transport Block Size (TBS) is smaller than the packet size that results in the packet being segmented, and the segments being transmitted sequentially when the transmission slots are available. The selected TBS for a UE in both UL and DL is influenced by the Modulation and Coding Scheme (MCS) index, which is chosen based on channel quality to optimize spectral efficiency and minimize errors and Physical Resource Blocks (PRBs) allocated to a UE.

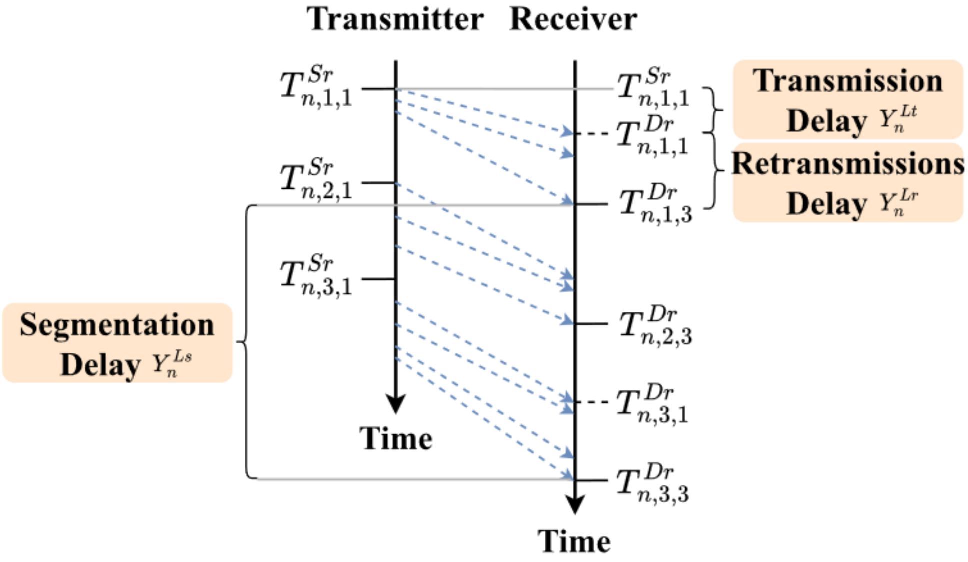


Figure 2.2 Link delay components of the 5G OWD. The dashed arrows indicate HARQ attempts.

Mobile networks utilize the Hybrid Automatic Repeat Request (HARQ) process to provide reliable transmission in unreliable wireless links. HARQ combines Forward Error Correction (FEC) and Automatic Repeat Request (ARQ) to correct errors by retransmitting lost or corrupted packets. The delay introduced by retransmissions is influenced by the MCS index, channel conditions, and the number of retransmission attempts, marking the time difference from the first to the last transmission attempt of a given packet segment.

Finally, there is a finite amount of time required for the Medium Access Control (MAC) layer to encode, modulate, and transmit a packet segment, and for the receiver to demodulate and decode it, regardless of the outcome of HARQ decoding. This delay encapsulates the time to transmit a segment across the radio link.

The process of decomposing link delay into specific components within a 5G network involves addressing the complexity introduced by the assignment of Protocol Data Unit (PDU) segments to multiple parallel HARQ processes. The challenge arises due to distinct transmission and retransmission delays for each segment, especially when they overlap as a result of HARQ pipelining. To tackle this, a systematic approach is proposed where the segment with the maximum service delay is identified. This serves as the basis for decomposing the service delay into transmission and retransmission delays for each packet's journey through the 5G network.

After identifying the segment with the maximum service delay, transmission delay and retransmission delay for the packet are identified. The remaining service delay is equal to the segmentation delay as shown in Figure 2.2. An illustrative example is given by $T_{n,m,l}^{Dr}$, representing the departure timestamp of packet *n* in the radio layer, specifically for the *m*-th segment and the *l*-th HARQ attempt.

Moreover, as each type of delay component is impacted by different parameters of the 5G system as well as traffic, it is imperative to gather and retain the parameters specified in Table 4 for every packet. This data is crucial to perform advanced delay analysis and for network optimization. Next, we will explore the complexities of the latency design and implementation in detail.

Table 4 Features to collect alongside the timestamps.

Delay Component	Information to Collect
Queueing Delay	Arrival Time in the RLC layer
	Queue Length (in bytes) in the RLC layer
	First Scheduled Slot in the TDD pattern
Segmentation Delay	Packet Size (in bytes)
	TBS (Transport Block Size in bytes)
	Segments Scheduled Slots
Retransmissions Delay	Number of Retransmissions
	Retransmission Slots
	MCS (Modulation and Coding Scheme) Index
	Channel Quality Indicators

2.2 Design and Implementation

The design of the latency measurement framework is based the following key principles:

Out-of-band Time Synchronization: To ensure accuracy in OWD measurements, the framework employs an out-of-band mechanism for time synchronization. This ensures that all components of the framework use the same time reference when timestamping packets / network conditions.

Microservice Architecture: By adopting a microservice architecture, the framework ensures that different software components and processes are decoupled and can operate independently. This architecture supports scalability, resilience, and ease of deployment, which are crucial for handling the prolonged measurements in 5G networks.

Out of Band Data Collection: Similar to time synchronization, the framework employs an out-of-band approach for data collection. In other words, the measurement data is transported from the measurement points to the data collection and aggregation module via channels distinct from the existing data plane traffic. This method helps in avoiding interference with the measurement process and ensures that the collected data is accurate and not affected by the measurement traffic itself.

The implementation of the latency measurement framework based on the above design involves several components and processes, each aimed at accomplishing a certain design goal.

Time Synchronization: The time synchronization of various components in the framework is accomplished using a Grandmaster Clock (GM) connected to a Global navigation satellite system (GNSS) antenna. An out-of-band Ethernet network is used to distribute time synchronization messages of PTP to all hosts involved in the measurement setup. All hosts were equipped with hardware timestamping-capable network interface cards (NICs) to PTP on an out-of-band network. The hosts run two types of ptp tools: ptp4l and phc2sys. The NIC clock is synchronized to the GM clock via ptp4l whereas the system clock is synchronized to the NIC clock via phc2sys [PTPL][PHCS]. Using this setup, the resulting time synchronization error was below 150 ns. The time synchronization for the COTS 5G setup is not purely based on PTP, which is explained in Section 3.1.

Traffic Generation and Timestamping: The framework includes a traffic generator that injects network traffic into the 5G system that can mimic real-world application scenarios. The traffic generator is deployed in a client-server model, i.e., a UDP client instance send traffic towards with a certain configurable period to the UDP server instance. In addition to traffic generation, the packets are timestamped at the client and the server. This approach helps in measuring the latency experienced by packets as they traverse through the network.

COTS 5G Measurement Points: The COTS 5G system currently does not have capabilities to collect metadata (e.g., features in Table 4) at the COTS base stations. Therefore, a measurement service was developed to be used at COTS UE devices to act as a measurement point. The proposed measurement service is capable of recording a few network conditions (e.g., RSRP and RSRQ) and exposing it on a http server. The collected data provides insights into the latency characteristics of standard network equipment and devices.

OAI 5G Measurement Points: The OAI 5G setup is based on the software 5G implementation of OAI. OAI offers a flexible and open-source platform for experimenting with 5G, allowing for a deeper analysis of packet delay and its variation. OAI 5G provides an opportunity to collect rich amounts of data by inserting measurement points in the different layers of 5G protocol stack at UE, RAN and/or core. Packet when traversing a layer (e.g., RLC) is timestamped along with the local identifier (e.g., RLC sequence number) as well as network metadata (e.g., number of PDUs waiting in the RLC queue). The identifiers are later used to track the end-to-end journey of each packet through the OAI 5G system.

Data Collection, Aggregation and Storage: Collecting data with minimal interference in the user plane traffic is essential, ensuring the process does not impact the measurement, i.e., packet delay should not be significantly changed due to the introduction of measurement points. Therefore, we selected the LatSeq project as our primary tool for data collection. LatSeq is tailored to extract timestamped information across different layers within OAI.

Real-time aggregation of data from different measurement points in the 5G system and processing the aggregated data is important for the latency measurement framework. To meet this requirement, we follow a microservices architecture (as mentioned in the design) for developing our framework by moving away from the file-based data exchange approach used in LatSeq. The data-aggregator is packaged as a Docker container, which is responsible for gathering data from all measurement points over via socket-based connections.

As probabilistic analysis of packet delay is important for certain applications, it is useful to optimize data storage with respect to that objective. To this end, the latency measurement framework incorporates InfluxDB, a database specialized in time-series data, ensuring the efficient storage of measurements and swift response to queries.

Analytics: The aggregated data can optionally undergo some analytical processing to derive meaningful insights into the network's latency performance. We have implemented simple analytics to demonstrate packet delay decomposition at different delay targets. The results for these analytics are discussed in Section 4.2.

It is worth mentioning that implementing and evaluating the latency measurement framework requires a flexible experimentation platform that can facilitate detailed end-to-end experiments. ExPECA serves as an ideal testbed for wireless communication and edge-computing studies, offering the ability to conduct experiments through the use of COTS 5G as well as Software-defined Radios (SDRs) and OAI 5G for enhanced reproducibility [MMR+23].

3 COTS 5G Data Collection

In this section, we provide a description of how the latency measurement framework can be used to collect measurements on the COTS 5G system. First, we describe the measurement setup and associated configurations. Next, we explain the workflow for the latency measurement in the COTS 5G system on the ExPECA testbed.

3.1 Measurement Setup

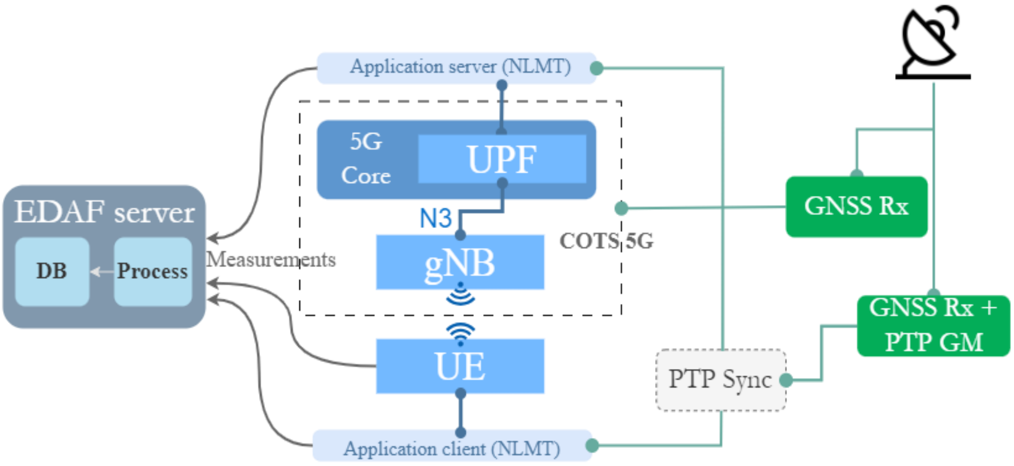


Figure 3.1 Measurement setup for data collection in COTS 5G setup.

The measurement setup for data collection on COTS 5G setup is shown in Figure 3.1. The COTS 5G network is an on-premises, non-public 5G network deployment located in an isolated location (KTH R1 hall) 25 meters below ground as shown in Figure 3.2. The COTS 5G radio units and the COTS 5G UE are installed on the roof and the walls, respectively, as described in the map of the testbed [EXPM]. The 5G system provides an accurate time reference through a GNSS-based synchronization. The same GNSS receiver is used to synchronize the PTP GM, which acts as a time source for the servers running the traffic generator client and server applications.

Figure 3.2 A picture of the underground site (at KTH R1) of the testbed.

A key component of the latency measurement framework is the Network Latency Measurement Tool (NLMT) [NLMT]. NLMT is based on an existing network measurement tool called IRTT (Isochronous Round-Trip Tester) [IRTT]. This tool accomplishes two tasks: (i) traffic generation and (ii) packet timestamping. The UDP packets of a given payload size are generated periodically at a fixed (configurable) interval by the client NLMT instance and are sent towards the server NLMT instance. The packet sequence numbers and timestamps are recorded at both client and server. In addition to this information, network information (RSRP and RSRQ) is sampled at the COTS UE at a fixed interval using a via measurement service continuously. This recorded information at client, server and the UE is transported to a remote server for storage and possible analytics.

The workflow for data collection on the COTS 5G system using the ExPECA testbed is shown in Figure 3.3. The procedure is documented in detail in the ExPECA user guide [EXPU]. Here, we only discuss the high-level steps. Two hosts (baremetal server nodes) and the COTS 5G system are reserved in the ExPECA testbed, and the relevant networking tasks are performed as explained in [EXPU]. Two containers are instantiated on the two reserved server hosts, to host the NLMT client and server, respectively. The NLMT server listens for packets sent by the NLMT client at UDP port 2112. The measurement session is initiated by running a script at the first host that triggers the NLMT client to run one or more measurement rounds. Using the object storage service, the collected data (e.g., JSON files) can be stored in *containers* to access them after the measurements are finished.

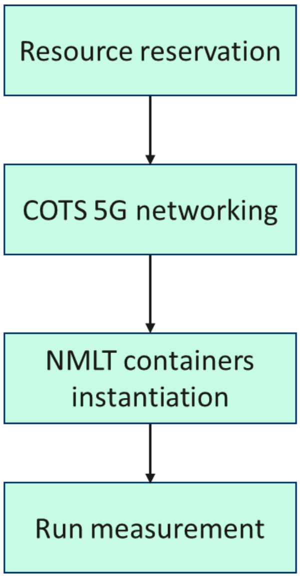


Figure 3.3 Procedure for latency measurements on COTS 5G setup.

3.2 Sample Measurement Results

Next, we present a sample latency measurements result collected on the COTS 5G system.

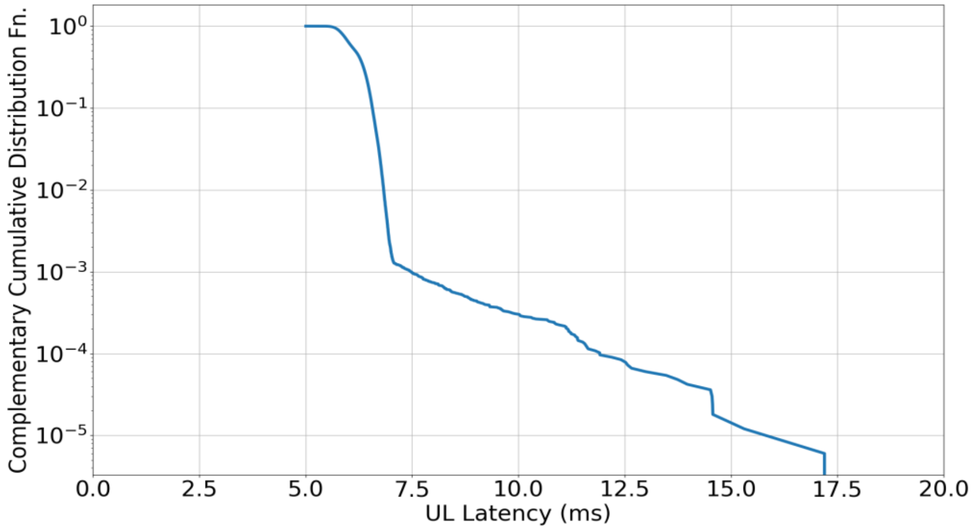


Figure 3.4 CCDF plotted from a sample measurement session of the uplink packet delay in the COTS 5G setup.

Figure 3.4 depicts the CCDF of uplink packet delay in the COTS 5G setup. The comprehensive latency measurements dataset on the COTS 5G system can be found in the repository [SSG+23].

4 OpenAirInterface 5G Data Collection

The data collected on the existing COTS 5G provides limited information about the 5G system. For example, the end-to-end latency measurements do not provide any information on the breakdown on latency in the end-to-end path due to limitation of the current setup. This is essential to perform comprehensive quantitative analysis of packet delay in 5G systems. Furthermore, it is important to capture the state of the 5G/5G-Adv system along with packet delay measurements to build efficient data-driven latency prediction models. As the COTS 5G does not offer much flexibility for such data collection, we have enhanced the framework for data collection in software-based 5G system implementation, e.g., OpenAirInterface 5G. OAI 5G is an opensource implementation of 5G that in conjunction with SDRs offers a flexible platform for 5G experimentation [OAI5G].

The latency measurement framework for OAI fulfills the objective of measuring the end-to-end delay along with its individual delay components for every packet. This detailed decomposition of the end-to-end delay provides an opportunity to perform advanced analytics.

Next, we describe the measurement setup used for data collection in OAI 5G. It is important to note that while the framework supports measurements with multiple 5G User Equipments (UEs), we will discuss the setup for a single UE only.

4.1 Measurement setup

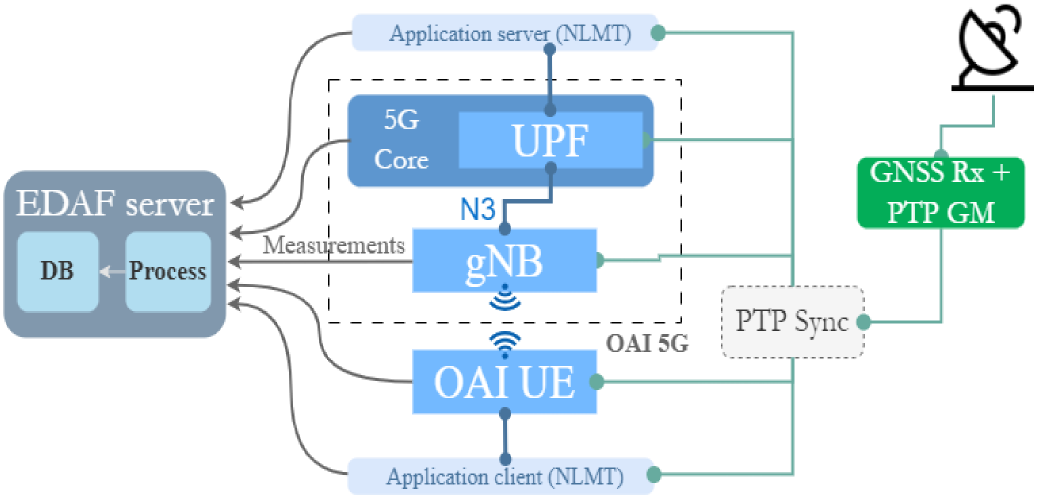


Figure 4.1 Measurement setup for data collection in OAI 5G setup.

The measurement setup for data collection on OAI 5G is shown in Figure 4.1.

Accurate time synchronization becomes more important for the latency measurement framework in the OAI 5G setup as compared to the COTS 5G, as the aim here is to accurately track the packet delay components from the client application endpoint to the server application endpoint. To this end, PTP-based time synchronization in an out-of-band wired network is used to provide time reference to different nodes in the setup as shown in Figure 4.1. The OAI 5G code was patched with measurement points as previously discussed. For the measurements in the OAI 5G setup, we also use the NLMT tools. However, in contrast to the COTS 5G, packet timestamps are collected in the NLMT client and server as well as in the 5G system. NLMT generated periodic UDP packets each 500 bytes in size every 10 milliseconds sent towards the NLMT server in the uplink direction. By running a measurement session for 20 minutes, samples corresponding to about 120,000 are collected. The experiments were conducted on an OAI 5G network that operated in the 5G NR n78 band using TDD mode, with 106 Physical Resource Blocks (PRBs) covering a 40 MHz bandwidth at 3.5 GHz and a Sub-carrier Spacing (SCS) of 30 kHz. A significant source of latency in these experiments was frame-alignment delay. The used TDD pattern is shown in Figure 4.2. With 5G frames lasting 10 ms, this setup resulted in the arrival of one packet per frame.

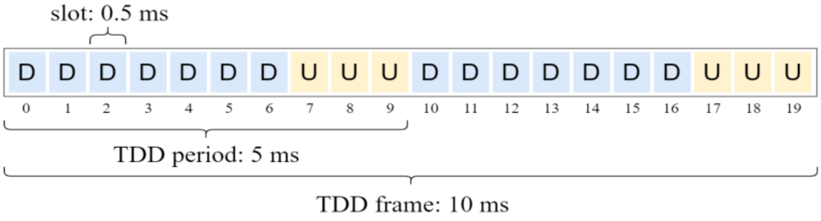


Figure 4.2 An illustration of the used TDD pattern used in the experiments. ”D” and “U” denote downlink and uplink slots.

In the current release of the framework, we have implemented data collection for 5G uplink measurements. This framework will be extended in future for data collection in the downlink direction as well. Figure 4.3 shows the high-level procedure to perform latency measurements on the OAI 5G setup on the ExPECA testbed. For further details about the OAI 5G hardware and software setup, please refer to the ExPECA documentation on OAI at [EXPO]. First, the required resources (e.g., three hosts, two USRP SDRs) are reserved. Next, the OAI 5G core network services are set up by instantiating corresponding docker containers on one host. These are MySQL, NRF, UDR, UDM, AUSF, AMF, SMF and SPGWU/UPF running as docker containers. The EDAF service which is responsible for data aggregation and processing is then instantiated on the core network host. Next, containers for UE and gNB are instantiated on the remaining two hosts. Finally, the measurement session(s) are initiated by instantiating the NLMT containers whose results are stored online through the EDAF service in a DB.

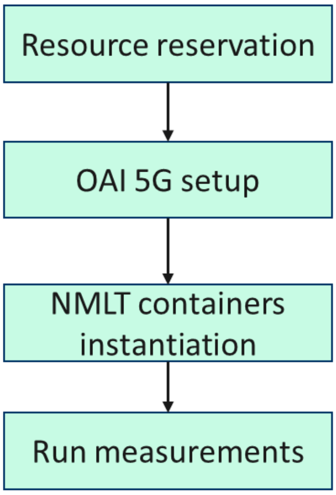
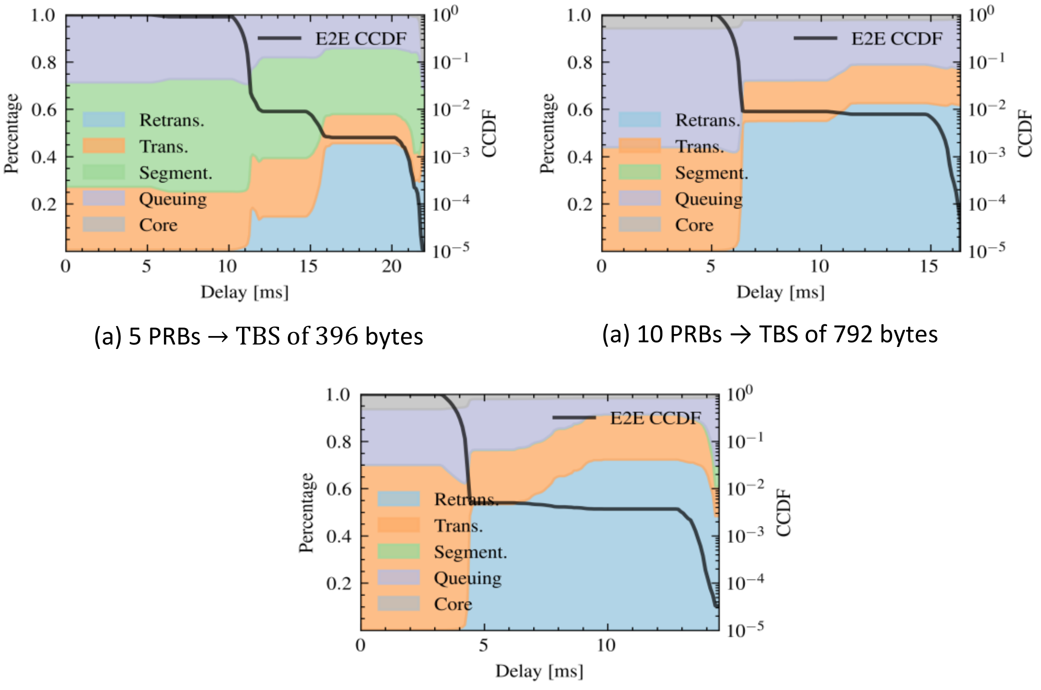


Figure 4.3 Procedure for latency measurements on OAI 5G setup.

Next, using the insights gathered from measurements collected through the latency measurement framework, we demonstrate how packet delay in 5G system can be optimized.

4.2 Sample Measurement Results



(a) 5 PRBs → TBS of 396 bytes

(a) 10 PRBs → TBS of 792 bytes

(c) 10 PRBs + optimized packet arrival times

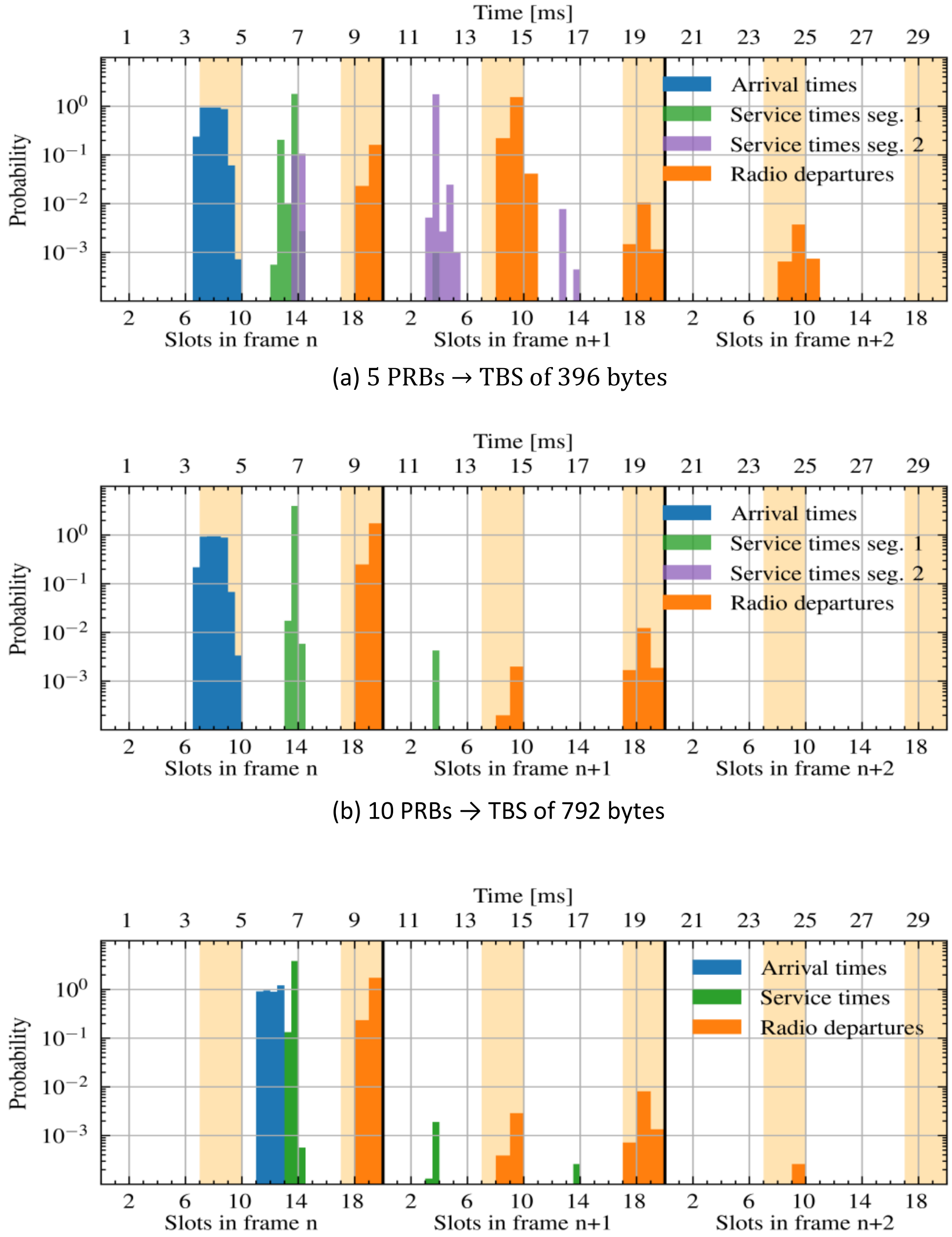
Figure 4.4 End-to-End CCDF and decomposition in experiments feature a fixed MCS index of 23 and 500-byte packets.

The measurement results and the decomposition analytics are presented in Figure 4.4 and Figure 4.5. Each subfigure (a, b and c) in Figure 4.4 and Figure 4.5 correspond to Experiments (a), (b) and (c). Experiment (a) serves as the baseline experimentation for the OAI 5G packet delay measurements. Figure 4.4 shows, on the right y-axis, the Complementary Cumulative Distribution Function (CCDF) for the measured end-to-end packet delays. The CCDF can be used to derive the Delay Violation Probability (DVP) for various end-to-end delay targets. Furthermore, the figure's left y-axis breaks down the proportional contributions of different components to the total end-to-end packet delay. Initially, we evaluate the DVP for a specific threshold, τ. For Experiment (a), achieving a 15 ms target results in a DVP of 10^{-2}, whereas for a τ = 5 ms target, the DVP nearly reaches 1. We then analyze the breakdown of delay contributions for each target. In both scenarios, the segmentation delay is the main contributor, accounting for 45% and 40%, respectively.

In Experiment (b), we attempt to eliminate the segmentation delay by increasing the uplink grant PRBs from 5 to 10, which allows for a TBS of 880 bytes. This size can accommodate a full 531-byte packet along with its headers, in contrast to the TBS of 396 bytes in Experiment (a). The smaller TBS in Experiment (a) resulted in a packet segmentation, thus adding a 5 ms increase in the end-to-end delay. Figure 4.5 depicts the distribution of radio arrival times, service times, and departure times, plotted over three consecutive frames to illustrate the packets' journey over time. Specifically, Figure 4.5 (a) highlights the delay caused by segmentation, with the second segment's service times lagging 10 or 20 slots behind the first, thereby delaying packet departure. This issue is addressed in Figure 4.5 (b) and Figure 4.5 (c), leading to significantly improved end-to-end delays. Nonetheless, experiment (b) maintains the same DVP for both τ=15 ms and τ=5 ms targets, requiring further optimization.

In Experiment (c), not only segmentation delay is eliminated the frame-alignment delay is minimized by reducing queuing delay by addressing the frame-alignment delay. The observed time gap in Figure 4.5, ranging from the packets' arrival in slots 7 to 10 to their first service in slots 12 to 14, results from packets arriving too early. By optimizing the arrival time offset, to minimize queuing delays), a reduction in end-to-end delays by 3 ms is expected.

The search for the optimal arrival, as shown in Figure 4.6, confirms our hypothesis, with frame arrival offsets between 1-2 and 6-7 achieving the lowest end-to-end delays. Here, we again measure the UL latency between a UE configured with 10 PRBs. As a result, this experiment limits DVPs for both τ=5 ms and τ=15 ms targets to 10^{-2} and 10^{-4}, respectively, fulfilling the application requirements.



(a) 5 PRBs → TBS of 396 bytes

(b) 10 PRBs → TBS of 792 bytes

(c) 10 PRBs + optimized packet arrival times

Figure 4.5 Histograms of service times and radio departure times.

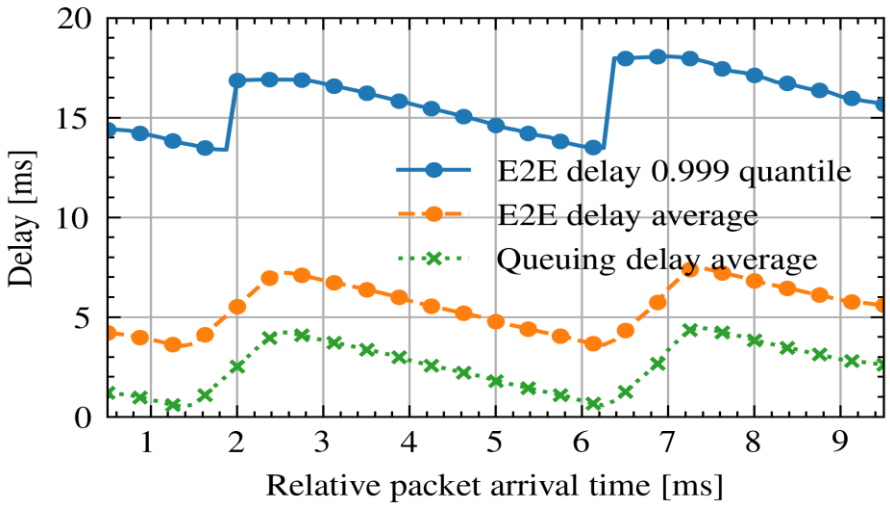


Figure 4.6 Minimizing end-to-end delay through optimizing arrival times offset relative to the 5G TDD period.

An additional insight from our analysis highlights the difference between lower and higher end-to-end delays, with retransmission delays becoming increasingly significant, constituting up to 50% of the total packet delay. This pattern is consistent across all experiments, emphasizing that infrequent but significant retransmissions are the primary factor behind outliers in end-to-end packet delays.

5 Conclusion

In recent years, there has been an increased focus on packet delay and its variation as a network KPI. This is a result of a variety of emerging time-critical applications from various domains such as flexible manufacturing, exoskeletons, XR and smart farming. Packet delay as a KPI is important as we transition from 5G towards 6G, where the demand for ultra-reliable, low-latency communications is surging for time critical applications.

The DETERMINISTIC6G project is at the forefront, crafting the enablers needed for time-critical communications in 6G networks. Within the project, it is important to gather comprehensive latency data from 5G networks. This data is crucial, serving dual purposes: training AI/ML models for latency prediction and creating simulation models of 6G DetCom node which is representative of 5G/5G-Adv networks. The widely used network measurement tools fall short, missing the nuanced dynamics of network latency in 5G as well as complex contribution of several 5G mechanisms to the overall end-to-end packet delay. The proposed framework is designed to address these challenges. The framework allows for conducting comprehensive latency measurements across both COTS 5G and OAI 5G setups. The framework's design and operational capabilities were outlined in this document, showcasing how it enables packet delay measurements. Sample measurements collected on COTS 5G and OAI 5G demonstrate its effectiveness and richness. Furthermore, the advantages of framework, especially in optimizing end-to-end packet delay is highlighted. The possibility of systematically optimizing end-to-end delay via framework's continuous measurement and decomposition analysis was demonstrated.

This tool holds significant potential for researchers interested in performing experimentation related KPIs like packet delay and PDV in 5G/5G-Adv. The framework provides a comprehensive setup for analyzing end-to-end delays within the OpenAirInterface 5G environment, which enables the evaluation of new features aimed at reducing latency. URLLC features in 5G as well as upcoming features being proposed for 6G aimed at optimizing latency are garnering interest. These features can be implemented on software-based 5G implementations like OpenAirInterface and tested using the framework to evaluate their performance under various conditions. Furthermore, the framework allows for the measurement of packet departure probabilities across different time slots, offering valuable input for creating optimized schedules that support the integration of 5G with TSN, a key ambition for advancing 5G/5G-Adv networks.

In future deliverable, we will report comprehensive measurements representing the relevant use cases using the proposed latency measurement framework and conduct an analysis of these collected measurements.

6 References

[3GPP16-22261]	3GPP TS 22.261, “Service requirements for the 5G system,” v19.4.0
[3GPP18-R94]	3GPP TSG-RAN WG1 Meeting 94 R1-1809277, “IMT-2020 self-evaluation: UP latency in NR”
[5GR20-D42]	5Growth Deliverable D4.2, “D4.2: Verification methodology and tool design”, Nov. 30, 2020.
[AAB+22]	J. Ansari, C. Andersson, P. de Bruin, J. Farkas, L. Grosjean, J. Sachs, J. Torsner et al., "Performance of 5G trials for industrial automation," Electronics, vol. 11, no. 3, 2022
[DET23-D11]	DETERMINISTIC6G, Deliverable 1.1, “DETERMINISTIC6G use cases and architecture principles,” Jun. 2023, https://deterministic6g.eu/index.php/library-m/deliverables
[DET23-D21]	DETERMINISTIC6G, Deliverable 2.1, “First report on 6G centric enablers,” Jun. 2023, https://deterministic6g.eu/index.php/library-m/deliverables
[DET23-D22]	DETERMINISTIC6G, Deliverable 2.2, “First Report on the time synchronization for E2E time awareness,” Dec. 2023, https://deterministic6g.eu/index.php/library-m/deliverables
[DET23-D31]	DETERMINISTIC6G, Deliverable 3.1, “Report on 6G convergence enablers towards deterministic communication standards,” Dec. 2023, https://deterministic6g.eu/index.php/library-m/deliverables
[DET23-D41]	DETERMINISTIC6G, Deliverable 4.1, “Digest on First DetCom Simulator Framework Release,” Dec. 2023, https://deterministic6g.eu/index.php/library-m/deliverables
[DETNET]	Deterministic Networking (DetNet) Working Group, [Online]. Available: https://datatracker.ietf.org/wg/detnet/about/
[ECAT]	EtherCAT technology group,. [Online]. Available: https://www.ethercat.org/en/technology.html
[EGR+15]	P. Emmerich, S. Gallenmüller, D. Raumer, F. Wohlfart, and G. Carle, “MoonGen: A Scriptable High-Speed Packet Generator,” In Proceedings of the 2015 Internet Measurement Conference (IMC '15). Association for Computing Machinery, New York, NY, USA, 275–287. https://doi.org/10.1145/2815675.2815692
[EXPM]	ExPECA Testbed Map, [Online]. Available: https://expeca.proj.kth.se/map/
[EXPO]	ExPECA OpenAirInterface Setup guide, [Online]. Available: https://github.com/KTH-EXPECA/openairinterface5g-docs/tree/main
[EXPU]	ExPECA User Guide, [Online]. Available: https://expeca.proj.kth.se/docs/

[FPNG]	D. Schweikert, [Online]. Available : https://fping.org/
[IPRF]	IPERF3, [Online]. Available: https://iperf.fr/iperf-download.php
[IRTT]	IRTT, [Online]. AvailableOnline: https://github.com/heistp/irtt
[MMR+23]	S. Mostafavi, V.N. Moothedath, S. Ronngren, N. Roy, G.P. Sharma, S. Seo, M.O. Muñoz and J. Gross, "ExPECA: An Experimental Platform for Trustworthy Edge Computing Applications," Nov. 2023, doi: 10.48550/arXiv.2311.01279
[MSG+23]	S. Mostafavi, G. P. Sharma, and J. Gross, "Data-Driven Latency Probability Prediction for Wireless Networks: Focusing on Tail Probabilities," arXiv preprint arXiv:2307.10648, 2023.
[MTS+24]	S. Mostafavi, M. Tillner, G. P. Sharma, and J. Gross, "EDAF: An End-to-End Delay Analytics Framework for 5G-and-Beyond Networks," arXiv preprint arXiv:2401.09856, 2024.
[NLMT]	NLMT, [Online]. Available: https://github.com/samiemostafavi/nlmt
[OAI5G]	OpenAirInterface 5G Implementation: https://gitlab.eurecom.fr/oai/openairinterface5g
[PHCS]	Phc2sys, [Online]. Available Online: https://linux.die.net/man/8/phc2sys
[PLNK]	Ethernet Powerlink standardization group,. [Online]. Available: http://www.ethernet-powerlink.org/
[PNG]	Ping, [Online]. Available: https://man7.org/linux/man-pages/man8/ping.8.html
[PTPL]	ptp4l, [Online]. Available: https://linux.die.net/man/8/ptp4l
[RFH+21]	F. Ronteix–Jacquet, A. Ferrieux, I. Hamchaoui, S. Tuffin and X. Lagrange, "LatSeq: A Low-Impact Internal Latency Measurement Tool for OpenAirInterface," 2021 IEEE Wireless Communications and Networking Conference (WCNC), Nanjing, China, 2021, pp. 1-6, doi: 10.1109/WCNC49053.2021.9417345.
[RSI+21]	J. Rischke, P. Sossalla, S. Itting, F. H. P. Fitzek and M. Reisslein, "5G Campus Networks: A First Measurement Study," in IEEE Access, vol. 9, pp. 121786-121803, 2021, doi: 10.1109/ACCESS.2021.3108423.
[SPS+23]	G. P. Sharma, D. Patel, J. Sachs, M. De Andrade, J. Farkas, J. Harmatos, B. Varga, H. -P., Bernhard, R. Muzaffar, M. Ahmed, F. Duerr, D. Bruckner, E.M. De Oca, D. Houatra, H. Zhang and J. Gross, "Toward Deterministic Communications in 6G Networks: State of the Art, Open Challenges and the Way Forward," in IEEE Access, vol. 11, pp. 106898-106923, 2023, doi: 10.1109/ACCESS.2023.3316605
[SSG+23]	S. S. Mostafavi, G. P. Sharma and J. Gross, "DETERMINISTIC6G COTS 5G latency measurements". Zenodo, Dec. 15, 2023. doi: 10.5281/zenodo.10390211.
[TSN]	Time-Sensitive Networking (TSN) Task Group, [Online]. Available: https://1.ieee802.org/tsn/
[WPP+22]	G. Wikström et al, "6G – Connecting a cyber-physical world", Ericsson white paper, GFTL-20:001402, February 2022, https://www.ericsson.com/4927de/assets/local/reports-papers/white-papers/6g--connecting-a-cyber-physical-world.pdf

7 List of abbreviations

3GPP	3rd Generation Partnership Project
5G	Fifth Generation
5G-Adv	5G Advanced
AI	Artificial Intelligence
CCDF	Complementary Cumulative Distribution Function
CDF	Cumulative Distribution Function
COTS	Commercial Off-The-Shelf
CPS	Cyber-Physical Systems
DB	Database
DetCom	Deterministic Communications
DetNet	Deterministic Networking
DVP	Delay Violation Probability
GM	Grandmaster
gNB	Next Generation NodeB
GNSS	Global Navigation Satellite System
INT	In-band Network Telemetry
IP	Internet Protocol
KPI	Key Performance Indicator
MAC	Media Access Control
MCS	Modulation and Coding Scheme
ML	Machine Learning
NIC	Network Interface Card
NLMT	Network Latency Measurement Tool
NR	New Radio
OAI	OpenAirInterface
OSI	Open Systems Interconnection
OWD	One-Way Delay
P4	Programming Protocol-Independent Packet Processors
PDCP	Packet Data Convergence Protocol
PDU	Protocol Data Unit
PHY	Physical Layer
PRB	Physical Resource Block
PTP	Precision Time Protocol
RLC	Radio Link Control
RSRP	Reference Signal Received Power
RSRQ	Reference Signal Received Quality
RTT	Round-Trip Time
SCS	Sub-Carrier Spacing
SDAP	Service Data Adaptation Protocol
SDR	Software-Defined Radio
TB	Transport Block

TBS	Transport Block Size
TCP	Transmission Control Protocol
TDD	Time Division Duplexing
TSN	Time-Sensitive Networking
UDP	User Datagram Protocol
UL	Uplink
UPF	User Plane Function
URLLC	Ultra-Reliable Low-Latency Communications
XR	Extended Reality