FACULDADE DE ENGENHARIA DA UNIVERSIDADE DO PORTO

Net Neutrality in the 5G Era

Rúben Daniel Nunes Santos

WORKING VERSION

Mestrado Integrado em Engenharia Eletrotécnica e de Computadores

Supervisor: Prof. Manuel Ricardo

Second Supervisor: Hermann Bergmann

February 27, 2023

Resumo

5G introduziu *network slicing*, permitindo a existência de várias redes lógicas independentes em cima de uma infraestrutura física. Estas redes lógicas são especialmente dedicadas a satisfazer os requisitos de qualidade de serviço de “use cases” heterogéneos do 5G. Isto contradiz o princípio da neutralidade da rede, que afirma que todo o tráfego deve ser tratado de forma igual, sem discriminação. Assim, coloca-se a questão: Como detetar a diferenciação do tráfego numa rede 5G? Para responder a esta pergunta, esta dissertação analisa a arquitetura da rede 5G e desenvolve uma nova abordagem da “Função de análise de dados da rede” (NWDAF). Para desenvolver e testar esta NWDAF, a rede central 5G e os seus componentes são emulados, e um novo caso de utilização é definido.

Palavras-Chave: *Neutralidade de Rede, NWDAF, Diferenciação de tráfego*

Abstract

5G introduced network slicing, enabling the existence of independent logical networks on top of a physical infrastructure. These logical networks are specially dedicated to meet the performance requirements of heterogeneous 5G use cases. This contradicts the premise of net neutrality, which dictates that all traffic should be treated equally, without discrimination. So, the question arises: How to detect traffic differentiation in a 5G network? To answer this question, this dissertation analyzes the architecture of the 5G network and develops a new approach to the "Network data analytics function" (NWDAF). To develop and test this NWDAF, the 5G core network and its components are emulated, and a new use case is defined.

Keywords: *Network Neutrality, NWDAF, Traffic Differentiation*

Contents

List of Figures

List of Tables

Abbreviations and Symbols

3GPP	3rd Generation Partnership Project
AMF	Acess and Mobility Management Function
AP	Application Function
AS	Autonomous System
CAP	Content and Application Provider
eMBB	enhanced Mobile Broadband
FCFS	First-Come First-Served
gNB	gNodeB
GPS	Generalized Processor Sharing
HTTP	HyperText Transfer Protocol
IP	Internet Protocol
ISP	Internet Service Provider
ML	Machine Learning
mMTC	massive Machine-Type Communication
NEF	Network Exposure Function
NF	Network Function
NN	Network Neutrality
NWDAF	Network Data Analytics Function
P2P	Peer-to-Peer
PFP	Packet Forwarding Prioritization
OAI	Open Air Interface
OAM	Operations Administration and Maintenance
QoS	Quality of Service
RAN	Radio Access Network
RSP	Rogue Super Peer
SDN	Software-Defined Networking
SDR	Software-Defined Radio
SMF	Session Management Function
TCP	Transmission Control Protocol
TD	Traffic Differentiation
UDP	User Datagram Protocol
UE	User Equipment
UPF	User Plane Function
URLLC	Ultra Reliable Low Latency Communication
VOIP	Voice over Internet Protocol
WFQ	Weighted fair queueing

Chapter 1

Introduction

5G uses a concept known as network slicing to virtualize the network with dedicated radio and computational resources. These slices are designed to customize certain Quality of Service (QoS) requirements such as throughput, latency, reliability, or connection density to account for the diversity of traffic. The customization of performance parameters in logical networks creates a set of virtual domains that give a different treatment to Internet traffic. This creates a potential conflict with the net neutrality principle, which states that all Internet traffic should be treated equally. The net neutrality concept is still a polemic subject. More conservative approaches indulge the idea that there should be no exceptions and traffic differentiation should not be allowed, while others support the segmentation of Internet traffic into classes, for example. Both points of view seem to agree that net neutrality imposes a non-discriminatory treatment of Internet traffic.

1.1 Objectives and contributions

The research question addressed in this dissertation is the following: How can we identify traffic differentiation in a 5G network in the context of net neutrality? Taking this into consideration, the following objectives were established:

- Define the concept of net neutrality.
- Characterize the 5G network architecture.
- Analyze the “Network data analytics function” (NWDAF).
- Simulate the 5G Core Network and its components.
- Define and implement a use case for testing.
- Test and validate the developed function.

In order to accomplish these objectives, a “Network Data Analytics Function” (NWDAF) specialized for the detection of network neutrality violation will be developed. This specialized

NWDAF will be aimed at monitoring traffic differentiation that could eventually lead to discriminatory practices and impact the net neutrality principle. As NWDAF is part of the network functions of the 5G network architecture, it will rely on data collected inside the network. As such, the data could be accessible to the telecom regulator based on its legal jurisdiction. The most critical metrics for detecting traffic differentiation (TD) are packet delay and packet loss. These two indicators allow us to determine whether the ISP has deliberately tampered with the packets in the UPF packet queue. The ISP can purposefully delay packets from a specific content and applications provider (CAP) by using Quality of Service (QoS) techniques such as traffic shaping. They can go even further and discard the packets entirely. The purpose is to detect these activities using the measurements provided by the Operations, Administration and Maintenance (OAM) 5G service and collected by the NWDAF.

As a use case, different flows from two separate CAPs will be monitored. A YouTube and a Netflix traffic flow will be created, with one of them facing throttling. Throttling is an intentional degradation in network performance for a given service, application, or class of applications. Everything will be done remotely, in an emulated environment, on top of a virtual 5G Core Network. Then, by comparing these flows, the NWDAF should be able to detect the TD and make the information available for the telecom regulatory entity.

This approach is an innovative way of using NWDAF and, most importantly, it will enable the detection of traffic differentiation (TD) practices inside the network. This is a game changer, as the information asymmetry between the regulator and the ISP will reduce, giving more transparency in traffic management practices.

1.2 Document structure

Chapter 2, Background and Fundamental Aspects, introduces fundamental concepts and technologies required to understand the problem at hand and the solution proposed in this dissertation. Chapter 3, State of the Art, further elaborates on the technical aspect by reviewing some of the prior work focused on solving a similar problem to that of this dissertation. Chapter 4, Preliminary work and workplan, frames the problem of this dissertation and presents the planned tasks to be developed to solve this problem. Chapter 5, conclusion, talks about what is being implemented and what is possibly going to be implemented in the future.

Chapter 2

Background and Fundamental Aspects

2.1 Network Neutrality

According to [1], net neutrality states that all Internet traffic, regardless of origin, destination, or content, should be treated equally. ISPs feel compelled to apply discriminatory traffic management practices to obtain a competitive advantage over the competitors, increase the number of customers, or charge higher fees. Discriminatory traffic management practices are often called Traffic Differentiation (TD). In some cases, TD can be considered reasonable and allowed, as long as it is transparent, such as i) Addressing illegal content (piracy, spam, or viruses); ii) Prioritizing DNS queries; and iii) Prioritizing special services, e.g., real-time health services. In general, TD can be seen as reasonable if it is beneficial to the network and its users as a whole. However, TD is detrimental to innovation, fair competition, and consumers' freedom of choice, three important factors deemed essential for the success of the Internet. [2]

Three major factors offer a problem to TD detection: 1) Detect TD as an external observer with no access to the network configurations; 2) Detect TD using only traffic (source address, destination address, port); 3) Identify which features trigger TD and where it is happening.

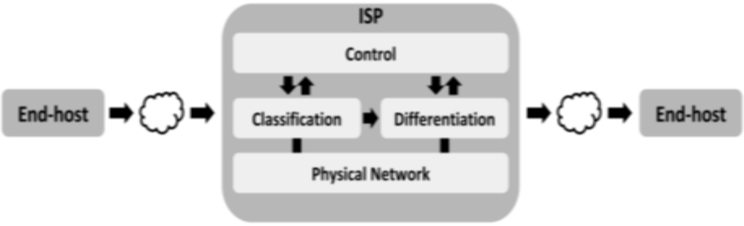


Figure 2.1: High-level description of TD on an ISP network [1]

Figure 2.1 exemplifies how an ISP can affect traffic between two end-hosts. Any traffic traversing through the ISP is classified and handled accordingly to the assigned class.The classification of the Internet traffic can be based on several criteria, such as: 1) TCP/UDP port, which may contain information about the traffic type; 2) Source address; 3) Destination address; 4) Application protocol.

Just like the Internet, the TD mechanisms used by ISPs are continuously evolving. Traffic characteristics, such as payload and port, can be used by commercial traffic shapers to determine many applications. Still, newer technologies, like software-defined networking (SDN), will increase flexibility and will enable various types of discriminatory behaviors to be easily implemented in the network. TD detection mechanisms will have to be able to adapt to the evolution of networks.

Since 2003, the number of individuals, businesses, and private and public institutions involved in the NN discussion has grown. Several CAPs, like Google and Netflix, support NN, while ISPs are generally against it. The scientific community has also chimed in, providing techniques for detecting NN infringements. Cases of network neutrality violations have been reported on all five continents, demonstrating that the topic is extremely complex and that monitoring and enforcing compliance with NN legislation has become a challenge. Overall, NN is a hotly debated topic as policies are adopted and withdrawn worldwide. [1]

2.2 5G

5G network slices are designed with the goal of optimizing the priorities of traffic. As such, they aim at providing low latency, high data rates, high reliability and a huge number of connected devices. The three main 5G network slices are: enhanced Mobile Broadband (eMBB), massive Machine-Type Communications (mMTC), and Ultra-Reliable and Low-Latency Communications (URLLC). eMBB corresponds to services that enable the transfer of large amounts of data for an improved user experience. mMTC corresponds to services designed to server a massive number of connected devices (IoT), which have low cost and consume very little energy. URLLC was projected for services that require ultra reliability and low latency. *Figure 2.2* presents some use cases for each of the three mentioned network slices.

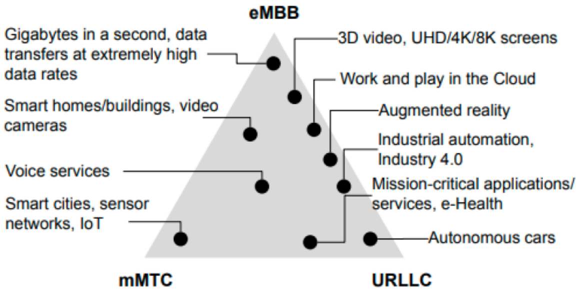

eMBB
Gigabytes in a second, data transfers at extremely high data rates
3D video, UHD/4K/8K screens
Work and play in the Cloud
Smart homes/buildings, video cameras
Augmented reality
Industrial automation, Industry 4.0
Voice services
Mission-critical applications/ services, e-Health
Smart cities, sensor networks, IoT
Autonomous cars
mMTC
URLLC

Figure 2.2: 5G eMBB, URLLC, and mMTC network slices use cases [3]

When it comes to its architecture, the 5G network can be decomposed into two planes: control plane and data plane. These two planes are independent, and they are connected by multiple interfaces. To provide scalability, evolution and flexible deployments, their functions are divided. Software-Defined Networking (SDN) makes this possible, by logically separating the two planes. The data plane, also known as user plane, has three main components: User Equipment (UE), Radio Access Network (RAN), and UPF. The UE connects to the UPF through the RAN. The

UPF acts as a router to the Internet. The data plane is the part of the 5G network responsible for the transportation of packets. The two planes, their main functions and the dissociation between the access network and Core network is shown in *Figure 2.3*.

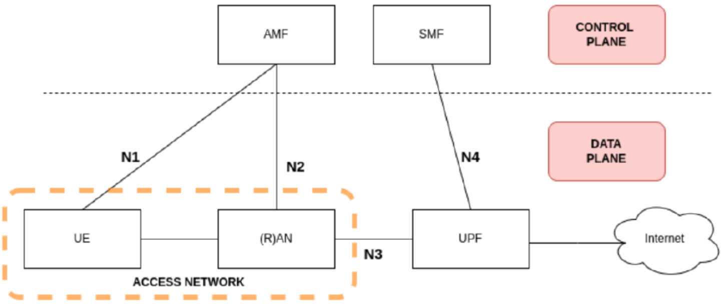


Figure 2.3: 5G network architecture [3]

The UPF acts as a gateway between the RAN and the Internet. As stated above, the UPF acts like a router, so it also has the functions of routing and forwarding packets. Quality of service (QoS) is also handled by the UPF. All the radio networking is ensured by the RAN. The RAN is primarily composed of a node, named gNB. The gNB is responsible for establishing connection between the UE and the core Network.

The main functions of the control plane are the Access and Mobility Management Function (AMF) and the Session Management Function (SMF). The AMF is responsible for access control and UE location management, while the SMF is in charge of managing sessions between the UE and the UPF. The control plane functions are virtualized, and they behave like web services. They are called Network Functions (NFs) and they are all independent of each other.

Essentially, the 5G Core network can be seen as a set of independent Web services that communicate with each other using HTTP, over the standard TCP/IP protocol stack. The HTTP protocol includes a set of methods that perform various tasks. The six main HTTP methods are: 1) GET, which is used to read a resource; 2) POST, which is used to create a resource; 3) PATCH, used to alter a portion of a resource; 4) DELETE, used to delete a resource; 5) PUT, used to update the full resource; 6) OPTIONS, used to obtain information about a resource.[3]

2.3 NWDAF

The Network Data Analytics function (NWDAF) is one of the Network Functions (NFs) of the 5G network and is able to provide analytics information to an NF consumer. Usually, an NF consumer is another NF of the 5G Network. The NWDAF is capable of getting analytics information about the network such as: 1) Slice load level analytics; 2) Observed Service Experience analytics; 3) NF load analytics; 4) Network Performance analytics; 5) UE related analytics; 6) User Data Congestion analytics; 7) QoS Sustainability analytics.

The NWDAF, just like other NFs of the Core Network, uses a subscribe/unsubscribe methodology. It can subscribe to specific events and receive notifications about them. As stated earlier, the NFs are basically web servers that interact with each other via HTTP (HTPP/2 more specifically). As such, the NWDAF uses an HTTP/2 based API to communicate with other NFs. It is important to note that the NWDAF can also subscribe to other NFs, and act as a consumer. That is how NWDAF is able to get analytics information from the network.

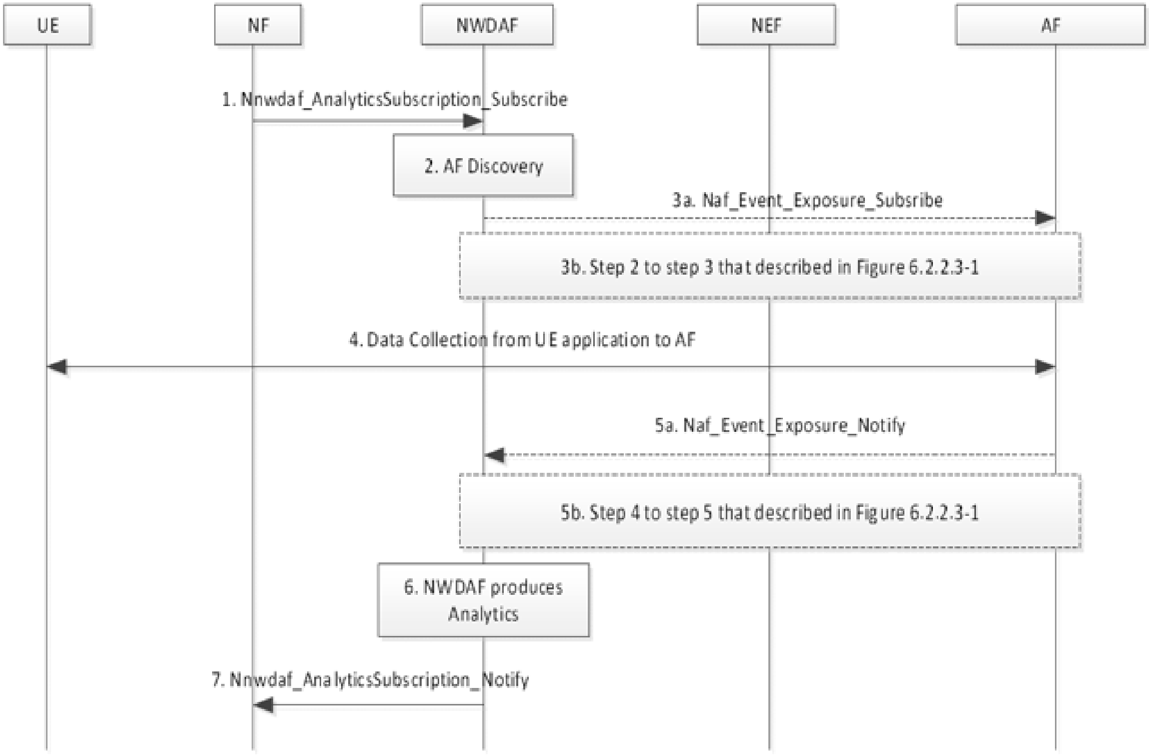

UE
NF
NWDAF
NEF
AF
1. Nnwdaf_AnalyticsSubscription_Subscribe
2. AF Discovery
3a. Naf_Event_Exposure_Subsribe
3b. Step 2 to step 3 that described in Figure 6.2.2.3-1
4. Data Collection from UE application to AF
5a. Naf_Event_Exposure_Notify
5b. Step 4 to step 5 that described in Figure 6.2.2.3-1
6. NWDAF produces Analytics
7. Nnwdaf_AnalyticsSubscription_Notify

Figure 2.4: Data collection procedure from UE [4]

Figure 2.4 shows how a NF can obtain analytic data about a User equipment (UE) from the NWDAF. It can be observed that the NWDAF can obtain information from other NFs (acting like a NF consumer), but it can also provide information for other NFs. The data collection function presented in Fig. 2.4, consists of the following steps:

1. An NF subscribes to analytics from the NWDAF;
2. NWDAF discovers the Application Function (AF) that provides data collection;
3. NWDAF subscribes to the AF for UE data collection;
4. The AF collects the UE data;
5. The AF receives the input data from the UE, and processes the data. The AF then notifies the NWDAF on the processed data;
6. The NWDAF produces analytics using the UE data received from the AF;
7. The NWDAF provides analytics to the consumer NF.

The NWDAF also has Machine Learning (ML) services. It follows the same subscribe/unsubscribe paradigm. A consumer can be notified when an ML model matching the subscription parameters becomes available, and it can also provide information about the ML model.[4]

2.4 QoS

This section provides an overview of Quality of Service (QoS) as interpreted by [5]. In theory, all packets on the Internet are treated equally, since the Internet is at its core a best-effort service. However, in practice, things are a bit different. Sometimes VOIP packets can be queued up behind other types of packets, and video stream packets can be throttled to favor other types of packets. It is well known that the performance of the Internet can be improved by treating packets differently.

Moreover, the network requirements for different types of applications are vastly different, e.g., some applications are very sensitive to delays, but can tolerate lower throughput, while others require high throughput and are resilient to delay. As a result, the packets of some applications can be given different treatment in order to improve their performance, without having noticeable effects on others. These ideas are implemented by some networks using QoS techniques.

2.4.1 Traffic Shaping

A router can get congested in two situations: 1) When the packet arrival rate is higher than the output rate of the router; 2) When a large burst of packets arrives at the router. Traffic shaping is a QoS technique that controls packet bursts and helps prevent the congestion of routers. There are two main traffic shaping methods: leaky bucket and token bucket. The leaky bucket approach uses a leaky bucket to control the rate at which traffic leaves a node. The leaking bucket maintains a steady outflow rate regardless of the rate of inflow. Any packets that exceed the bucket's capacity are discarded. The size of the bucket and the transmission rate are two characteristics that are unique to this technology and are usually user customizable [6].

In contrast, the token bucket method is less strict in terms of controlling the rate at which traffic leaves a node. Fig.2.5 represents a token bucket that shapes a packet flow. Aside from the Packet Buffer that stores the packets, there is also a Token Counter. The tokens arrive at the Token counter at a constant arrival rate of a tokens per second, and the token counter is limited to B tokens. If a packet with P bytes arrives at the buffer, it is only transmitted if there are at least P tokens in the counter. Then, the P tokens are removed from the counter, and the packet is sent.

2.4.2 Scheduling

Scheduling is a QoS technique that guarantees a minimum service rate. To control the sharing of a link, amongst packets of different classes, a scheduling strategy known as Weighted Fair Queuing (WFQ) may be used. Before analyzing and defining WFQ, it is necessary to consider an idealized version of WFQ, called Generalized Processor Sharing (GPS).

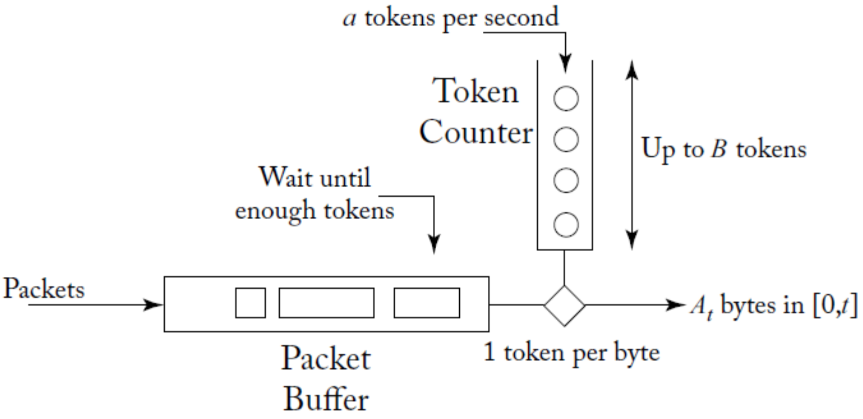


Figure 2.5: Token bucket shapes the traffic and limits its burst size [5]

The GPS system is represented in Fig.2.6. First, K classes are defined to classify each packet that arrives. Each packet is classified accordingly and waits in the corresponding queue to be transmitted. To each class is assigned a weight wk, where a higher weight represents a higher priority. This weight is used by the scheduler to determine the serving rate of the class. The rate at which a packet of each class is served is proportional to their weight:

$$\frac{wkC}{W} \tag{2.1}$$

where k represents the class of the packet, C is the line rate out of the router, and W is the sum of all the weights of the queues that are backlogged at time t. Since the scheduler cannot completely separate the packet, i.e, some bits of different packets are mixed, and the queues cannot be server simultaneously, this model is only theoretical.

WFQ is the model that best approximates GPS. The classification and queuing of packets follows the same procedure as GPS. However, just one packet at a time is transmitted by the scheduler, at the line rate. The scheduler calculates the finishing time of the packets and assigns the next packet to be transmitted as the packet that GPS would finish transmitting first among the remaining packets.

2.4.3 Implications on Net Neutrality

QoS, by definition, contradicts the Net Neutrality paradigm. Net Neutrality debates focus on whether or not QoS should be allowed. On the one hand, the defenders of an “Open Internet” reason that ISPs should not be allowed to provide better services for specific content providers as they please. The users who cannot afford better services could also be affected with the lack of neutrality.

On the other hand, opponents of neutrality argue that differentiation of services is clearly a positive asset for users. Also, ISPs might provide worse services overall for their customers because of the lack of incentive to upgrade their network.

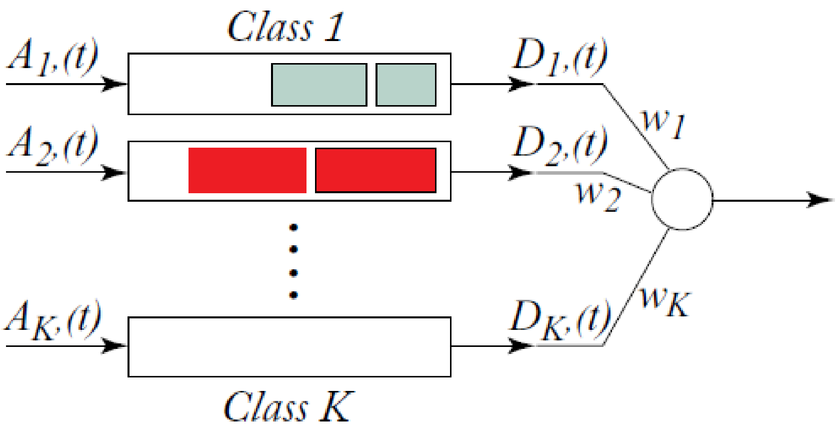


Figure 2.6: Generalized processor sharing (GPS) [5]

To try to satisfy both parties, regulations can be designed to allow service differentiation, while still maintaining part of the network under the neutrality paradigm.

2.5 Open-Source 5G software packages

2.5.1 Open-Source 5G Core Software Packages

To implement the 5G core network as a virtual network, there are a couple of software alternatives available. Only open-source software is included in this study.

Open Air Interface (OAI) [7] is one of the projects developed by the OpenAirInterface Software Alliance (OSA) and provides a 5G core network (5G CN) that currently supports all components to emulate the basic functionality of the Core Network. It has three deployment modes: 1) Minimalist 5GC; 2) Basic 5GC; 3) Slicing 5GC. NWDAF and Network Exposure Function (NEF) are said to be released in the future.

Free5GC [8] possesses a Web UI, but unlike OAI, there are no release dates announced for the NWDAF and NEF. Moreover, it is implemented using the GO language, which is not as widespread as the C language used by the other options available.

Open5GS [9] also has a Web UI, the documentation is detailed but not as good as OAI, plus no mention of NWDAF or NEF for future releases.

Magma [10] is a community project supported by many other entities, such as OAI and Advanced Micro Devices (AMD). It is developed using various programming languages, such as Python, C++ and Go.

Table 2.1 presents the four 5G software packages for the Core network and compares them.

	OAI	Free5GC	Open5GS	Magma
Programming Language	C, C++	Go	C	Various
Interface	CLI	CLI, WebUI	CLI, WebUI	CLI
Platform Support	Linux	Linux	Linux, MacOS	Linux, MacOS
NWDAF	Available in the future	No	No	No
NEF	Under-development	No	No	No
Documentation	Very Good	Good	Good	Good

Table 2.1: 5G Core open-source software packages comparison.

2.5.2 Open-Source 5G RAN software packages

To implement the 5G RAN as a virtual network, there are a couple of software alternatives available. Only open-source software is included.

OAI [7] offers two possibilities for RAN emulation: 1) OAI-RAN; 2) OAI-Emulator. Both are very complete, since they also implement the physical layer, unlike other similar software.

UERANISM [11] is a 5G UE and RAN (gNodeB) simulator. It can implement all the basic functions of UE and gNodeB, but some features are incomplete. Its simplicity makes it a suitable choice for testing. It emulates the RAN independently of the Core Network. It is no longer under active development.

Free5GRAN [12] is a 5G RAN stack. It is currently under active development, however it proposes a strategy centered on seamless integration with Software-Defined Radio (SDR).

srsRAN [13] was initially a 4G LTE project know has srsLTE. With the release of 5G new radio (NR), the developers started implementing new features to the code and renamed the project to srsRAN. It is under active development and new features are still being added every in release.

Table 2.2 presents the five 5G software packages for the RAN and compares them.

	OAI-RAN	UERANSIM	Free5GRAN	srsRAN	OAI-Emulator
Programming Language	C, C++	C++	C, C++	C, C++	C, C++
SDR Compatibility	Yes	No	Yes	Yes	No
Platform Support	Linux	Linux	Linux	Linux	Linux
Documentation	Very Good	Good	Good	Good	Very Good

Table 2.2: 5G Core open-source software packages comparison.

Chapter 3

State of the Art

In the literature, there are various studies regarding the detection of traffic differentiation by ISPs. These have drawn considerable attention for over ten years.

One of the first studies was carried out by [14], which measured the blocking of traffic belonging to a specific application or class of applications in their designated TCP or UDP port. It is claimed that their findings are among the first measurements of network neutrality.

The methodology used relies on P2P networks protocols, more specifically on the Gnutella protocol. By managing a Rogue Super Peer (RSP) and a measurement host, it is possible to redirect a client trying to join the Gnutella network to the measurement host's IP address and to a desired port. Fig. 3.1 depicts the base methodology used in this study.

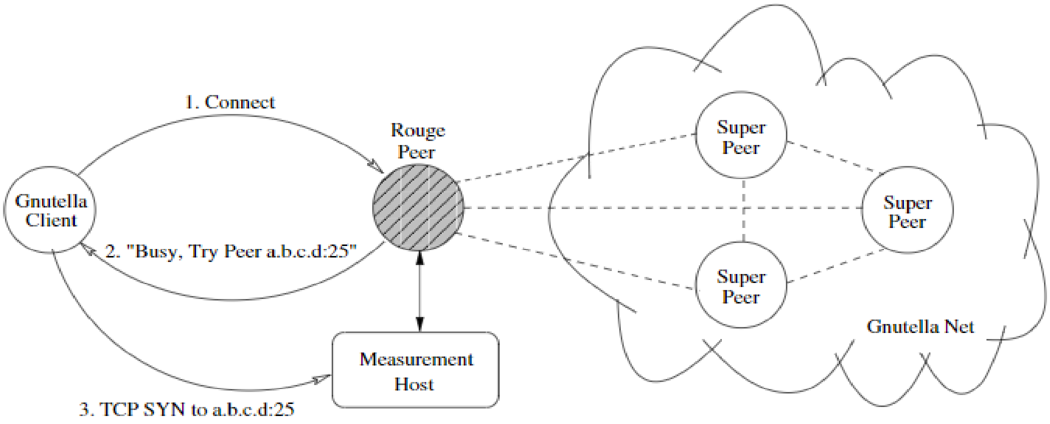


Figure 3.1: Methodology: The Rogue SuperPeer joins the Gnutella network. (1) Client attempts to connect; (2) RSP rejects, referring the client to a measurement host under our control; (3) By correlating connections, Internet port blocking is mapped. [14]

In 2009, [15] developed NetPolice as a system to verify traffic differentiation in backbone ISPs by taking loss measurements from end hosts. To accomplish this, numerous probes are launched in a distributed manner from a wide range of end systems. The study was conducted on 18 large ISPs across 3 continents over the span of 10 weeks.

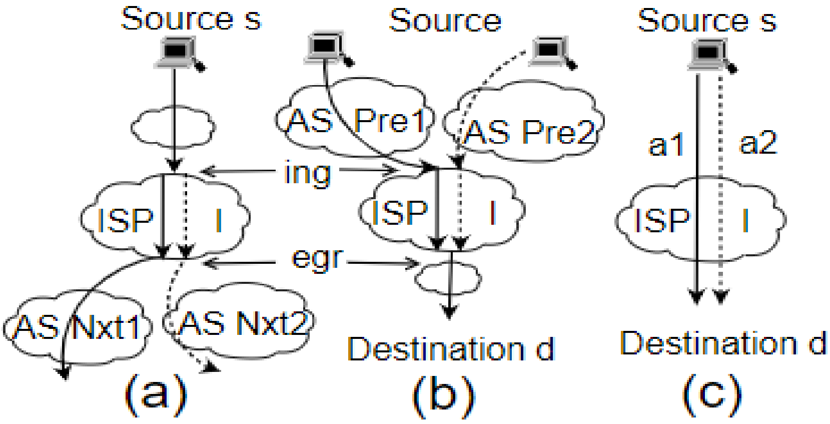


Figure 3.2: Detecting various types of differentiation with end-host based probing [15]

Fig.3.2 showcases the methodology used for detecting TD with NetPolice. In Fig.3.2(a) the end host (Source s) probes two different destination paths through the same Ingress and Egress of ISP I. This allows for the detection of differentiation based on the next-hop ASes. Fig.3.2(b) treats the opposite case, in which the previous-hop ASes are different. In Fig.3.2(c) the end host sends two probes through the same Ingress and Egress of ISP I with the same final destination but for different applications (a1 and a2). By comparing the performance of both paths, it can be detected eventual different treatment of the traffic due to its application type.

The authors of [15] state that this study is a crucial step towards the increase of transparency on the Internet. They also claim that, in contrast to previous researches, which focus on Broadband ISPs, NetPolice studies the detection of traffic differentiation in backbone ISPs. This requires the development of an intelligent path selection algorithm capable of measuring numerous ISPs internal paths.

In 2010, Glasnost was introduced by [16], with the purpose of identifying if an ISP was performing application-specific traffic shaping and reporting back that information to ordinary Internet users. The design of a system with this goal poses three main challenges:

- The system must be simple and generate quick results to gain popularity among users;
- To not have negative effects on the reputation and business of ISPs, the system should be resilient to measurement noise to avoid false accusations of differentiation;
- To keep up with the ever-changing differentiation policies of ISPs around the world, the system must be able to be upgraded;

Glasnost employs a client-server architecture. Clients connect to a Glasnost server and download and run a variety of tests. Each test creates a flow of application-level data that is utilized to measure the path between the client and the server. The data in these flows is specifically designed to identify traffic differentiation on the link between the client and the server.

Fig.3.3 provides an overview of how clients measure their Internet paths. A client first connects to a central website, which then redirects to a Glasnost measurement server. This dynamic redirection provides load balancing between the measurements servers and also simplifies the process of adding more serves to the redirection list.

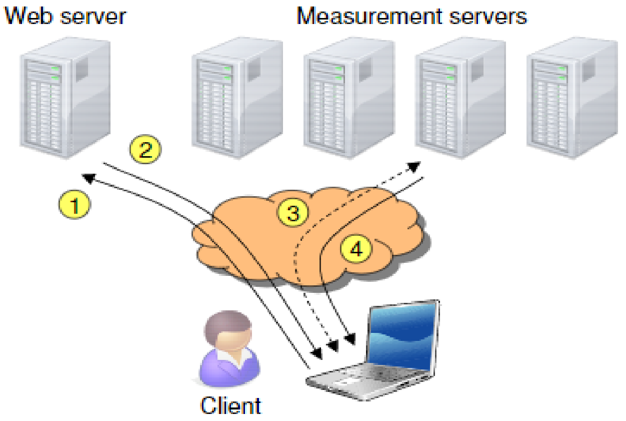
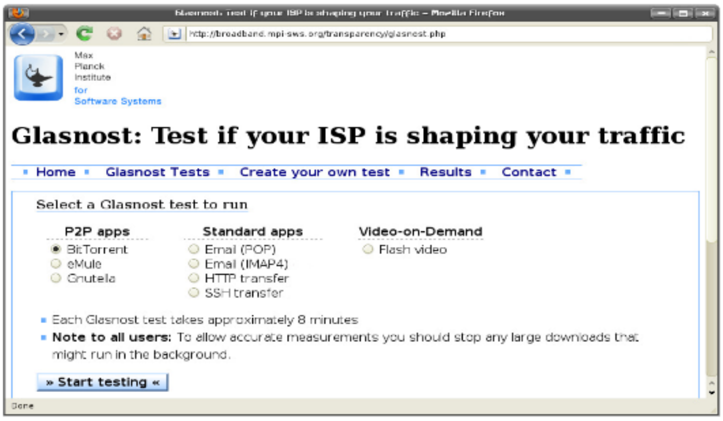


Figure 3.3: The Glasnost system.(1) The client contacts the Glasnost webpage. (2) The webpage returns the address of a measurement server. (3) The client connects to the measurement server and loads a Java applet. The applet then starts to emulate a sequence of flows. (4) After the test is done, the collected data is analyzed and a results page is displayed to the client. [16]

Fig.3.4 shows the user interface presented by the measurement server after the redirection of the client. It is a simple interface that lets the user select what type of traffic he/she wants to test. To start the test, the user just as to click on the "Start testing" button. Then, the Browser on the client's side downloads a java Applet that exchanges packets with the measurement server.


Glasnost: Test if your ISP is shaping your traffic – Mozilla Firefox
http://broadband.mpi-sws.org/transparency/glasnost.php
Max Planck Institute for Software Systems
Home
Glasnost Tests
Create your own test
Results
Contact
Done

Glasnost: Test if your ISP is shaping your traffic

Select a Glasnost test to run

P2P apps
● BitTorrent
○ eMule
○ Gnutela

Standard apps
○ Email (POP)
○ Email (IMAP4)
○ HTTP transfer
○ SSH transfer

Video-on-Demand
○ Flash video

- Each Glasnost test takes approximately 8 minutes
- **Note to all users:** To allow accurate measurements you should stop any large downloads that might run in the background.

» Start testing «

Figure 3.4: The Glasnost web interface [16]

Glasnost has been discontinued, as can be seen in [17]. The reason presented to abandon the project was that Java Applets no longer work on modern Web Browsers.

Also in 2010, [18] presented a user-level tool named POPI, which can detect packet forwarding prioritization (PFP) in routers through an end-to-end measurement. Three different metrics could be used to infer PFP: loss, delay and Out-Of-Order. The packet Loss based method has larger overhead, because differences only become observable when the associated link router queue becomes full and packets start to be dropped. Despite this, the loss metric was chosen as the best metric to infer PFP. It can detect various kinds of QoS techniques, and it is also less affected by parallelism in the forwarding paths.

The POPI tool was evaluated using statistical analysis, a simulation and a wide-area experiment in PlanetLab. The POPI tool was used to analyze 156 paths among 162 PlanetLab sites. Also, 15 paths were flagged as having multiple priorities, of which, 13 were validated by loss rate measurements. The authors of [18] concluded that the that POPI can infer the router's packet forwarding priority with high precision.

Still in 2010, [19] proposed Differential Probing or DiffProbe, with the objective of detecting if ISPs are emplying QoS to discriminate some traffic flows of their customers. It mainly focuses on delay discrimination and loss discrimination, by comparing a Probing flow P with an Application flow A.

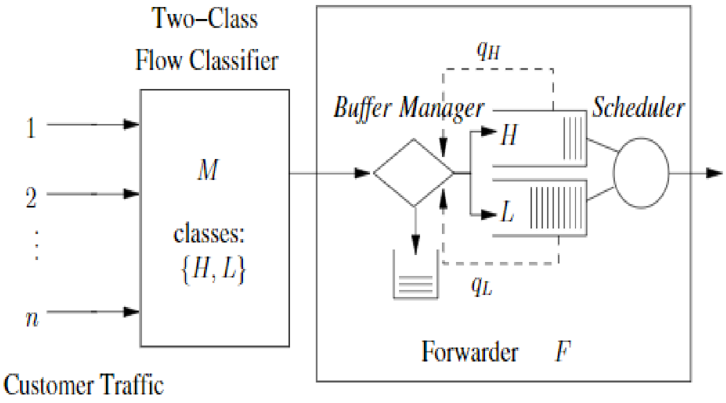


Figure 3.5: Model of access ISP discrimination [19]

Fig.3.5 describes the standard model used to implement DiffProbe. n users are connected to an ISP I and the user flows go through a classifier M. The flows are then marked as being High priority (H) or Low priority (L). If the Low priority flows are discriminated by the ISP, it is expected to see the traffic go to a different queue where a discriminatory packet schedule is applied. On the contrary, if the ISP does not discriminate traffic, the First-Come First-Served (FCFS) schedule should be applied. All the traffic should go to a single queue and packets should be forward as soon as they arrive, assuming the queue is empty.

The DiffProbe tool was tested on multiple ISPs, and the results show that the tool is accurate, as long as the discrimination of the traffic is perceivable by the user.

All the studies referenced up until this point, attempt to detect discrimination of traffic, however, they only focus on wired networks. This trend continued throughout the years, as evidenced by the statement of the 2018 study on neutrality in European Mobile Networks [20]: “Research about net neutrality mostly focused on the wired Internet, and little effort has been devoted to wireless scenarios”.

In 2019, [21] performed a large-scale study to assess the impact of content-based traffic differentiation methods applied in operational mobile networks, and fixed-rate bandwidth constraints were discovered to be the most frequently seen behavior.

This study points out two important differences between Internet traffic compared to previous years. Nowadays, the primary source of Internet traffic is video streaming platforms and not BitTorrent. Likewise, cellular network usage has skyrocketed in recent years, as more people than ever before access the Internet via mobile devices. With this in mind, the authors of [21] designed a mobile app named *Wehe* [22] that tests apps like YouTube, Netflix, Amazon Prime Video, NBC Sports, Vimeo, Spotify, Skype, and more. The findings indicate that ISPs utilize various throttling techniques and that the majority of throttling affects video streaming.

All the tools and techniques used in these works attempt to detect traffic differentiation (TD) and Network Neutrality violations without information about the ISP’s internal network. This lack of information makes it more complicated to detect such practices, as explained in the next Chapter.

Chapter 4

Preliminary work and workplan

4.1 Problem description

The proposed work in this dissertation focuses on: 1) Increasing the transparency of traffic management practices; 2) Demonstrate that the NWDAF can be used as a tool to assess TD; 3) Demonstrate how a Regulator can obtain performance information about the 5G network to evaluate potential discriminatory practices from the ISPs. The main objective of this dissertation is to develop a new use case for the Network Data Analysis Function (NWDAF) capable of identifying traffic differentiation from within the 5G Core Network. One of the main shortcomings of previous studies on the subject was that the mechanisms used were located outside the ISP network. As a result, traffic discrimination based on IP addresses, peering arrangements, interconnection congestion, traffic volume, or other characteristics other than IP payloads could not be detected [15]. Typically, an ISP could simply argue that the abnormal traffic behavior presented to them by network regulators was caused by external circumstances beyond the ISP's control.

Using the proposed NWDAF, network regulators will be able to acquire data from the 5G network, such as packet loss, delay, or throughput, in a timely manner.

4.2 Proposed solution

Fig.4.1 showcases the proposed architecture to implement and test the NWDAF. Two traffic streams of the same class (eMBB) from different Content and Application Providers (CAPs) will be compared. bCAP is the base CAP and will be used as a reference. tCAP is the target CAP and will be the one monitored to detect eventual discriminatory behavior from the part of the ISP. $t1$ represents the time in which a packet from tCAP arrives at the queue of the UPF. $t2$ represents the time that the same packet leaves the UPF to be routed to the corresponding gNB. The NWDAF can obtain this information from the Operations Administration and Maintenance (OAM) service and process the data. With this information, it is possible to determine the delay of the packet, infer packet loss and calculate the throughput of the whole stream of packets. An outside entity, in this case the telecom regulator, could have access to this data via the Network Exposure Function (NEF).

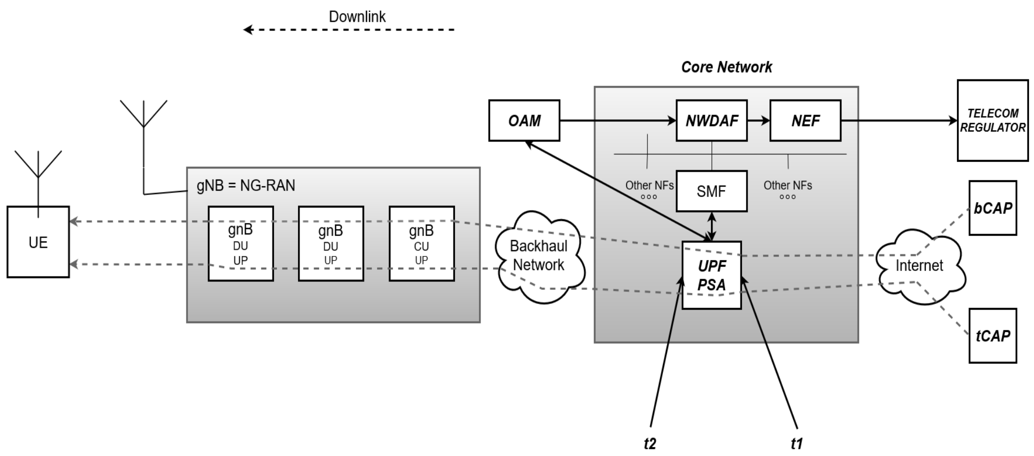


Figure 4.1: Diagram of the 5G network and the proposed solution

There are various types of traffic discrimination, such as: blocking, filtering, throttling, zero rating. The focus of this dissertation will be on detecting throttling, mostly because it is the most common practice. Throttling has a detrimental influence on network performance and should be detected using performance measurements. If the tCAP has a lower throughput than the bCAP, it is likely that some throttling could be happening. Throttling is also likely to be the cause if tCAP has a higher delay/latency or packet loss ratio than bCAP.

4.3 Work plan

In order to meet the dissertation objectives the following tasks will be carried out:

Task 1 - Install Open Air Interface (OAI) and all its components.

As can be seen in Table 2.1 and on Fig.4.2, OAI is the only open-source software package that is developing the NEF and the NWDAF. For this reason, and also because there is information available online, OAI was chosen to implement the 5G Core Network.

This should take about 19 days, and once completed, the system should look something like what is shown in Fig.4.3.

Task 2 - Specify, adapt or implement an NWDAF based on the current known norms that allows for the detection of discrimination practices.

This is the most complex task of the dissertation and should take around 61 days to complete. Upon completion of this task, there should be a basic NWDAF capable of collecting standardized metrics from the network.

Task 3 - Develop a test bench to test the prior installed 5G Core Network software package.

This should take about 25 days, and once completed, the system should look something like what is shown in Fig.4.4.

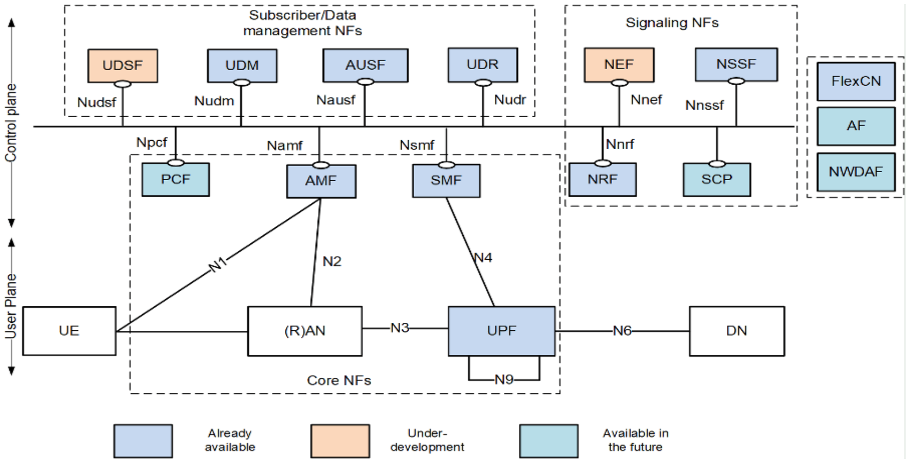
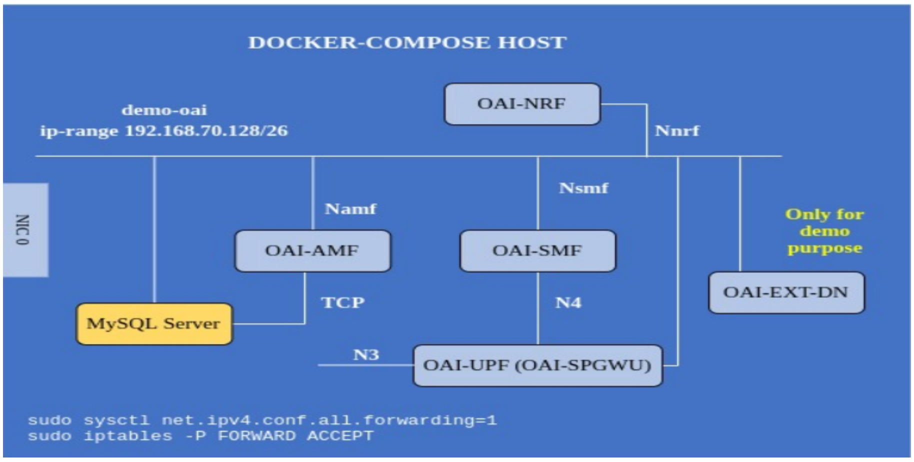


Figure 4.2: OAI Status [23]



```
sudo sysctl net.ipv4.conf.all.forwarding=1
sudo iptables -P FORWARD ACCEPT
```

Figure 4.3: OAI software 5G Core diagram [24]

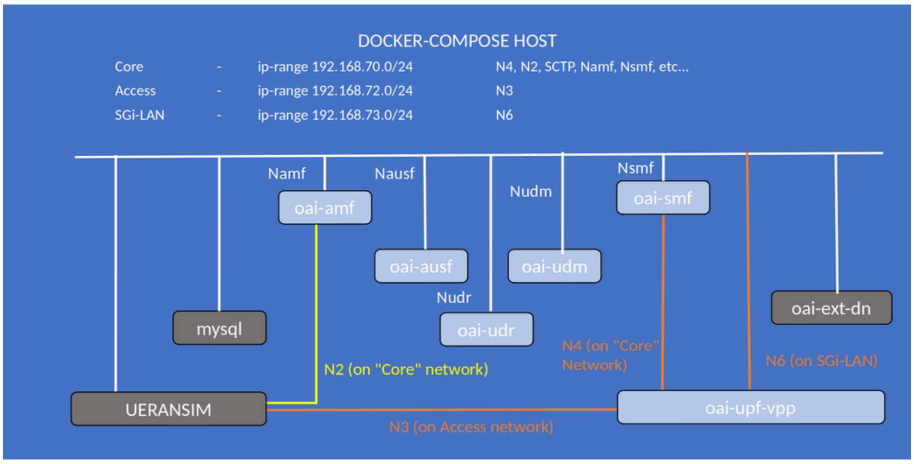


Figure 4.4: OAI software 5G Core diagram testing with UERANSIM [25]

Task 4 - Collect data from the NWDAF, including measurements and some graphics.

This should take about 24 days, and once concluded there some be evidence about the presence or not of discriminatory behavior from the part of ISP (one of the video streams will be given a different treatment to test if NWDAF works has expected).

Task 5 - Write the dissertation report

This is the final step of the dissertation, and all the time remaining should be spent on it. This task should take 32 days to complete based on the work plan.

Fig.4.5 shows the start and finish dates, as well as the duration of the tasks described above.

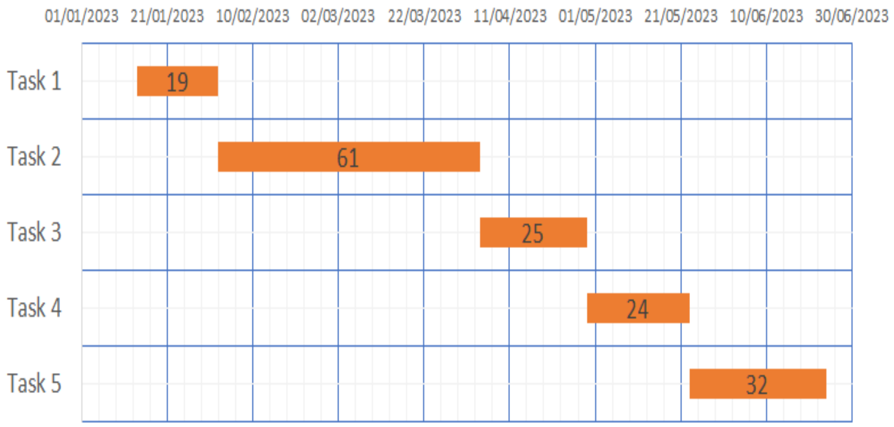


Figure 4.5: Gantt diagram of work plan

Chapter 5

Conclusions

The main objective of this dissertation is to adapt or develop an NWDAF capable of detecting traffic differentiation (TD) inside the 5G Core Network. The NWDAF should be able to obtain standardized metrics from the network and expose that information to the telecom regulator. It was defined a specific use case, in which the NWDAF will focus on detecting TD for eMBB traffic. A specific eMBB scenario is going to be tested, consisting of two video streams from two different CAPs. This is a significant breakthrough, given that NWDAF being used to check for net neutrality compliance is unprecedented.

References

[1] T. Garrett, L.E. Setenareski, L.M. Peres, L.C.E. Bona, and E.P. Duarte. Monitoring network neutrality: A survey on traffic differentiation detection. *IEEE Communications Surveys and Tutorials*, 20:2486–2517, 2018. doi:10.1109/COMST.2018.2812641.

[2] B. Van Schewick and D. Farber. Point/counterpoint: Network neutrality nuances. *Communications of the ACM*, 52:31–37, 2009. doi:10.1145/1461928.1461942.

[3] David Miguel de Almeida Coimbra Maia. *Control and Positioning of a 5G Radio Access Node Deployed in a Mobile Robotic Platform*. PhD thesis, 2022.

[4] 3GPP. 5G;Architecture enhancements for 5G System (5GS) to support network data analytics services. (23.288), 03 2022. Version 17.4.0. URL: https://portal.3gpp.org/desktopmodules/Specifications/SpecificationDetails.aspx?specificationId=3579.

[5] J. Walrand and S. Parekh. *Communication Networks: A Concise Introduction, Second Edition*, volume 10. 2017. doi:10.2200/S00804ED2V01Y201709CNT020.

[6] Itu-t an architectural framework for support of quality of service in packet networks, 2004.

[7] 5g core network – openairinterface. Accessed: 2022-12-24. URL: https://openairinterface.org/oai-5g-core-network-project/.

[8] Free5gc. Accessed: 2022-12-24. URL: https://www.free5gc.org/.

[9] Open5gs - open source project of 5gc and epc. Accessed: 2022-12-24. URL: https://open5gs.org/.

[10] Magma linux foundation project. Accessed: 2022-12-24. URL: https://magmacore.org/.

[11] aligungr/ueransim: Open source 5g ue and ran (gnodeb) implementation. Accessed: 2022-12-24. URL: https://github.com/aligungr/UERANSIM.

[12] free5g/free5gran: free5gran is an open-source 5g ran stack. the current version includes a receiver which decodes mib sib1 data. it also acts as a cell scanner. free5gran works in sa mode. Accessed: 2022-12-24. URL: https://github.com/free5G/free5GRAN.

[13] The srslte project is evolving. Accessed: 2022-12-24. URL: https://www.srslte.com/srslte-srsran.

[14] Robert Beverly, Steven Bauer, and Arthur Berger. The internet is not a big truck: Toward quantifying network neutrality. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 4427 LNCS:135–144, 2007. doi:10.1007/978-3-540-71617-4_14.

[15] Ying Zhang, Zhuoqing Morley Mao, and Ming Zhang. Detecting traffic differentiation in backbone isps with netpolice. *Proceedings of the ACM SIGCOMM Internet Measurement Conference, IMC*, pages 103–115, 2009. doi:10.1145/1644893.1644905.

[16] Marcel Dischinger, Krishna P. Gummadi, Massimiliano Marcon, Ratul Mahajan, Saikat Guha, and Stefan Saroiu. Glasnost: Enabling end users to detect traffic differentiation. *Proceedings of NSDI 2010: 7th USENIX Symposium on Networked Systems Design and Implementation*, pages 405–418, 2010.

[17] Glasnost: Test if your isp is shaping your traffic. Accessed: 2022-12-21. URL: https://broadband.mpi-sws.org/transparency/bttest.php.

[18] G. Lu, Y. Chen, S. Birrer, F.E. Bustamante, and X. Li. Popi: A user-level tool for inferring router packet forwarding priority. *IEEE/ACM Transactions on Networking*, 18:1–14, 2010. doi:10.1109/TNET.2009.2020799.

[19] Partha Kanuparthy and Constantine Dovrolis. Diffprobe: Detecting isp service discrimination. *Proceedings - IEEE INFOCOM*, 2010. doi:10.1109/INFCOM.2010.5461983.

[20] Enrico Gregori, Valerio Luconi, and Alessio Vecchio. Neutmon: Studying neutrality in european mobile networks. pages 523–528. Institute of Electrical and Electronics Engineers Inc., 7 2018. doi:10.1109/INFCOMW.2018.8407022.

[21] Fangfan Li, Arian Akhavan Niaki, David Choffnes, Phillipa Gill, and Alan Mislove. A large-scale analysis of deployed traffic differentiation practices. *SIGCOMM 2019 - Proceedings of the 2019 Conference of the ACM Special Interest Group on Data Communication*, pages 130–144, 8 2019. doi:10.1145/3341302.3342092.

[22] Wehe: Check your isp for net neutrality violations. Accessed: 2022-12-22. URL: https://wehe.meddle.mobi/index.html.

[23] Tien Thinh Nguyen, Lionel Gauthier, Sagar Arora, Rohan Kharade, Raphaël Defosseux, and Stefan Spettel. Openairinterface 5g core network: Status and roadmap. Accessed: 2023-01-05.

[24] Openairinterface 5g core network deployment using docker-compose and testing with dstest. Accessed: 2022-12-23. URL: https://gitlab.eurecom.fr/oai/cn5g/oai-cn5g-fed/-/blob/v1.1.0/docs/DEPLOY_SA5G_WITH_DS_TESTER.md.

[25] Openairinterface 5g core network deployment and testing with ueransim. Accessed: 2022-12-24. URL: https://gitlab.eurecom.fr/oai/cn5g/oai-cn5g-fed/-/blob/master/docs/DEPLOY_SA5G_WITH_UERANSIM.md.