A Network Simulator-based Method of Dynamic Traffic Generation for 5G Network Slicing

Zhiyi Ding

A Thesis
in
Department
of
Computer Science and Software Engineering

Presented in Partial Fulfillment of the Requirements
For the Degree of
Master of Computer Science at
Concordia University
Montréal, Québec, Canada

November 2021

CONCORDIA UNIVERSITY
School of Graduate Studies

This is to certify that the thesis prepared

By: **Zhiyi Ding**
Entitled: **A Network Simulator-based Method of Dynamic Traffic Generation for 5G Network Slicing**

and submitted in partial fulfillment of the requirements for the degree of

Master of Computer Science

complies with the regulations of this University and meets the accepted standards with respect to originality and quality.

Signed by the final examining committee:

____________________________________ Chair
Dr. Denis Pankratov

____________________________________ Examiner
Dr. Denis Pankratov

____________________________________ Examiner
Dr. Dhrubajyoti Goswami

____________________________________ Thesis Supervisor
Dr. Brigitte Jaumard

Approved by ____________________________________
Dr. Dhrubajyoti Goswami, Graduate Program Director

December 3, 2021 ____________________________________
Dr. Mourad Debbabi, Dean
Gina Cody School of Engineering and Computer Science

Abstract

A Network Simulator-based Method of Dynamic Traffic Generation for 5G Network Slicing

Zhiyi Ding

While machine learning models and algorithms are now often use for various network design, planning, provisioning, etc, not much 5G data is available. A Dynamic traffic generator is a valuable tool for evaluating and testing network performances. In this thesis, we present a novel traffic generation framework that meets some of the most important 5G network requirements. One of the distinctive features in 5G is the use of network virtualization, which allows network operators to partition the network into multiple "independent slices", each of which can carry different types of traffic, services, or applications such as enhanced Mobile Broadband, Ultra-Reliable Low Latency Communications, massive Machine Type Communications, VoIP, etc . To meet the new 5G requirements a novel framework that addresses the key related problems is designed, i.e., modeling virtual network functions, allowing end-to-end measurement of the key performance indicators, with a realistic network traffic modeling. The results and evaluation show that our framework is a powerful traffic generator for the time being to test 5G networks.

Acknowledgments

First of all, I would like to express deepest gratitude to my supervisor Dr. Brigitte Jaumard to offer me the opportunity to join the group, and to give us continuous patience, guide and support for completing this thesis and so on. I would also like to express my warm appreciation to Dr. Oscar Delgado who acted as a co-supervisor to guide and push me all the way through. I cannot be more grateful with your help. I also want to thank you the help and kindness from my team mates: Trong Tuan Tran, Daniel Aragão Abreu Filho, Tran Nguyen Phuc, for their hard work, contribution and dedication to the project. I have acquired a lot of experience through the journey of Master study. Last but not least, special thanks to Fadi Bishay to provide us with professional advices on 5G slicing from Ciena, and to Vincent Bissonnette to offer us technical assistance from BesLogic, Montreal. I sincerely wish all the best and success to this group in the future.

Contents

List of Figures

List of Tables

Abbreviations

3GPP The 3rd Generation Partnership Project. 1, 5

5G Fifth Generation Cellular Wireless Network. 1, 2, 4–6, 8, 9, 13, 16, 21, 28

5GC 5G Core Network. 5, 9, 12, 13

AMF Access and Mobility Management Function. 13

API Application Programming Interface. 21

AR Augmented Reality. 3

CBR Constant Bit Rate. 18, 28

CN Core Network. 2

CPU Central Processing Unit. 13, 14, 17, 19, 21, 22, 28

DL Down Link. 13, 21, 22

eMBB Enhanced Mobile Broadband. 3, 5, 6, 28

FPS Frame Per Second. 19

gNB logical 5G radio node. 12, 16

HD High Definition. 1, 3, 19

IoT Internet of Things. 1

ITU International Telecommunication Union. 1, 5

KPI Key Performance Indicator. 15, 21, 29

LR Low Resolution. 18

LSR label Switching Router. 16

LTE(4G) Long-Term Evolution. 2, 5

MANO MANagement and Orchestration. 2

MLPS Multi-protocol Level Switching. 12, 13, 16

mMTC Massive Machine-Type Communication. 3, 5, 6, 28

NFV Network Function Virtualization. 2

OSI Open System Interconnection. 9

PPP The Point-to-Point Protocol. 13

PPS Packets Per Second. 19

RAM Random Access Memory. 13, 14, 17, 19, 21, 22, 28

RAN Radio Access Network. 2, 5, 12, 13, 16

SLA Service Level Agreement. 4

SMF Session Management Function. 13

Sumo Simulation of Urban MObility. 9, 34

TN Transport Network. 2, 5, 12, 16

UEs User Equipment's. 12, 16, 18

UL Up Link. 13, 21

UPF User Plane Function. 12, 13, 16, 17, 28

URLLC Ultra-reliable and low latency Communication. 3, 5, 6, 28

VANET Vehicular Ad hoc Network. 9

VNF Virtual Network Function. 4–6, 8, 13–16, 19, 21, 22, 28

VoIP Voice Over Internet Protocol. 1, 18, 28

VPN Virtual Private Network. 4

XML Extensible Markup Language. 11

Chapter 1

Introduction

1.1 General Background of 5G

Wireless broadband technologies, which address the ever-increasing demands for faster and more reliable communications, contain some of the most critical research areas now and in the following years. One of the leading network technologies is Fifth Generation Cellular Wireless Network (5G) [2], i.e., the next generation of cellular network that includes all the stakeholders and players: standards and normative organizations (The 3rd Generation Partnership Project (3GPP), International Telecommunication Union (ITU), etc.), telecommunication operators, manufacturers, content and service providers which constitute a massive collection of efforts to standardize, design and deploy the next-generation cellular network services. According to the latest Ericsson Mobility Report [3], 5G subscriptions will reach 580 million by the end of 2021.

Some of the goals of 5G technologies are to support high data rates, to have more reliable connections, and to be able to handle numerous users. Which in turn enable services such as High Definition (HD) TV, Voice Over Internet Protocol (VoIP), mobile video conference, massive use of Internet of Things (IoT) devices, etc. While researching and developing innovative 5G solutions, it is essential to model new network scenarios, test alternative solutions and measure the most important network metrics. This thesis introduces a novel dynamic traffic generator to generate 5G traffic data.

1.2 5G Network Slicing

History of mobile cellular network slicing derives from [4] with the Radio Access Network (RAN) sharing from the Long-Term Evolution (LTE(4G)) standards which allow the operators to share the network resources in common RAN. 5G adopts the concept and extends to the Transport Network (TN) and core networks Core Network (CN) to support the ongoing-growing and differentiated demands of communication services.

The concept of 5G network slicing can be described as several virtual and independent networks built on shared physical infrastructure. Each slice is a partitioning of a network, which is logically tailored to accommodate a particular business-driven use case, such as self-driving cars, remote surgery, smart cities etc. Here are some critical technologies raised to enable 5G network slicing, including Network Function Virtualization (NFV), Service Chaining, and MANagement and Orchestration (MANO).

- NFV is a technology to decouple the software implementation of network functions from dedicated hardware, which is a foundation to enable the creation of network slicing on shared physical infrastructure.

- Service Chaining is a network capability that enables network traffic to connect in a specific order of network functions. It allows specific users or applications to traverse only on the necessary network function to manage a network on the changing demand dynamically.

- MANO is a framework proposed by the European Telecommunications Standard

Institute (ETSI) [5] to automate the management of Network Slices' life-cycle from creation, activation, running, to decommission.

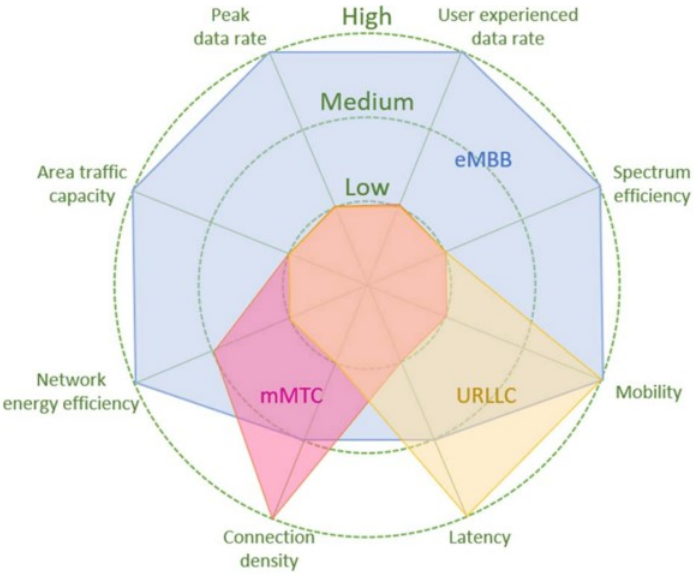


Figure 1: 5G Traffic Classes [1]

As Figure 1 shows, 5G traffic is described by three traffic classes, which can be described as follows:

- Enhanced Mobile Broadband (eMBB): This class of traffic addresses customer-centric with high data rates transmission. The objective is to increase the seamless user experience. The significant applications include HD Video Streaming and Augmented Reality (AR).

- Ultra-reliable and low latency Communication (URLLC): This class of traffic has stringent requirements in terms of throughput, latency, and availability, e.g., self-driving car, remote surgery, and industrial automation.

- Massive Machine-Type Communication (mMTC): This class of traffic focuses on a vast number of connected devices and relatively small light data frames, e.g., smart homes/cities.

1.3 Problem Statement

Due to high demands from both academia and the industry for 5G datasets, we found that access to a real-life 5G network to generate a proper and organized dataset is difficult due to confidentiality and competition among the customers (network services) of the network carriers. Using a network simulator became an alternate method to generate traffic datasets.

Network traffic generators, in general, are concerned with modelling and simulating the transmission and payloads of the packets produced by the devices connected to a network. It is often used to investigate a state of a network and evaluate its performance.

OMNET++ is among the top option of the public Network simulators, and we chose it as the base of our 5G traffic generator because of its 5G extension Simu5G [6] was commonly-used to simulate a 5G network and its ability to extend to support 5G network slices which are not available in the current version.

5G network virtualization uses network slicing [7] to support virtual networks over a single physical network infrastructure. Slicing allows a logical separation of a physical network to provide remote connectivity while running on the same shared infrastructure. 5G network virtualization enables network operators to, for example, assign a given network slice to certain kinds of devices or to fulfill the requirements specified on the Service Level Agreement (SLA), including delay, jitter, packet loss, etc.

Each network slice should be able to handle many types of network infrastructure (Virtual Private Network (VPN), cloud services) and Virtual Network Function (VNF). These new features enable network operators to dynamically create virtual 5G networks with unique capabilities, opening up the potential for new use cases.

In order to to have a tool that will allow researchers to test and measure 5G wireless networks and support features of 5G network slicing, we have designed a novel simulation framework that can generate traffic flows according to multiple traffic patterns. In addition, we modelled the three main kinds of computing resources for VNF, i.e., CPU, RAM and storage, which are needed for network slices in order to provide the network services, for example, compute resource required of running VNFs like routers, firewalls in a nearby data center.

1.4 Literature Review

Before introducing the design and implementation of the 5G Traffic Generator, it is essential to review the 5G traffic and 5G traffic generation in general.

In the survey of [1], authors summarized the 5G traffic characteristics of the three major 5G traffic classeseMBB, URLLC and mMTC defined in 3GPP, ITU, etc. Different user scenarios and network requirements were compared in these three kinds. On the other hand, [8] researched 5G traffic based on traffic flows' behaviours, identified and classified the traffic categories based on traffic flow traits such as packet counts, packet size distribution, flow size, etc. Moreover, measure the network's performance on a simple test-bed to validate the results.

As for 5G traffic generation, there are two general ways of generation. The first one is to set up a self-built 5G network environment using a network simulator or test-bed. [9, 10] utilized the network simulator called OMNET++ and its 5G extension Simu5G [6] to crate their own 5G network environment to generate and analyze traffic data. For other research groups, [11, 12] built a more complex 5G network test-bed on the 5G Core Network (5GC) and TN to simulate the main functions of the 5G networks and to produce and analyze the results. The second way to perform the traffic generation is to use the mathematical model to replicate the data from a small portion to a large scale. For example, [13] generated the time series data on the 5G RAN by modelling self-similarity function and compare the results with real LTE(4G) traffic.

1.5 Challenges

This section presents the major challenges we faced to complete this thesis.

- First, 5G is a broad topic including a lot of background knowledge, and it takes much effort to read and investigate the different aspects of the network.

- Secondly, to access a real 5G network and obtain the relevant 5G dataset had proven to be extremely difficult due to the low availability of the traffic data and also due to confidentiality concerns.

- Lastly, We designed a proper 5G simulator-based environment and fine-tuned the network dimensioning parameters to satisfy the needs of the user's applications.

1.6 Contributions

This thesis designs a Network Simulator to generate 5G dynamic slicing traffic using the open-source network simulator OMNET++.

The proposed traffic generator is realistic for the following two reasons:

1. Fairly realistic characteristics distribution of three 5G traffic classes(eMBB, URLLC, and mMTC) based on the survey [1].

2. The usage of the 5G network slices, which is derived from the car/pedestrian/truck traffic of the City of Montreal [14] and from the following IEEE papers [15, 16].

The main contributions are as follows:

- We describe the design and implementation of our dynamic traffic generator, which uses real urban traffic.

- We present the modelling and capabilities of network slicing, introducing the ability to handle multiple VNF and to slice the 5G network.

- We perform network dimensioning to adjust the traffic generation to a given network and input a separate service assurance research project (Closed-Loop management) with Ciena.
- We present simulation results that show the practical applicability of our framework.

Chapter 2

Implementation

2.1 Dynamic Traffic Generator

Gathering network traffic data plays a critical role in network planning, network orchestration, and traffic forecasting. However, collecting practical and meaningful 5G network traffic data is not an easy task because there are not so many networks available, and the complexity of having available data that captures all the new features 5G has. Researchers turned to Network Simulators to model the behaviours of 5G. In the work of [9], the authors implemented Motorway and Urban scenarios of 5G Cellular-Vehicles-to-Network(C-V2N) based on the SimuLTE [17] framework, and more recently, the Simu5G framework [6] was released. Nevertheless, not all the features needed for a more comprehensive simulation are available (network, slicing, VNF, etc.). To describe our traffic generator, we first briefly introduce the tools we have used.

2.2 Tools

OMNET++ is introduced as a modular, discrete event-driven, and C++-based network simulator that allows users to simulate the behaviours of a communication network and extend its functionalities. INET is a library for OMNET++. It implements many components of communication network and multiple network protocol across the Open System Interconnection (OSI) model, including TCP, UDP in the transport layer, IPv4 and IPv6 in the network layer and Ethernet, PPP in the data link layer. Simu5G [6] is one of its most recent extensions that enables 5G simulation on the data plane through the Radio Access Network (RAN) and the 55GC. Moreover, Vehicular Ad hoc Network (VANET) system is also supported in OMNET++, the coupling of Veins [18] and Simulation of Urban MObility (Sumo) [19] enables users to integrate and simulate real-life traffic including different types of vehicles (cars, trucks, etc), speed, departure, and destination, etc. Additionally, city maps can be incorporated into Sumo by generated them using OpenStreetMap [20].

2.3 Traffic Generation Design

Fig. 2 shows the workflow of our 5G traffic generator. We have defined two procedures in order to obtain, filter and process the raw data from [14] and transform it into a clean and compatible format with OMNET++.

To construct a real-life 5G network scenario, we used the traffic data (information at traffic light intersections) obtained from the City of Montreal, publicly available at [14]. The original data contains the information of different categories of vehicles, i.e., cars, trucks, bikes, pedestrians, and the direction of the traffic (north, south, east, west). The data is presented as a time series (data sequence recorded every 15 minutes). There are two main functionalities in Procedure 1. The first one is to generate a map file by selecting a subset of the intersections. The selected intersections generate the city map in the simulation environment.

To choose these intersections, we selected the area that contained the traffic information of three types: cars, pedestrians, and trucks, also at least last one week. In this case, eighteen intersections out of hundreds are selected from Ahuntsic-Cartierville, Montreal.

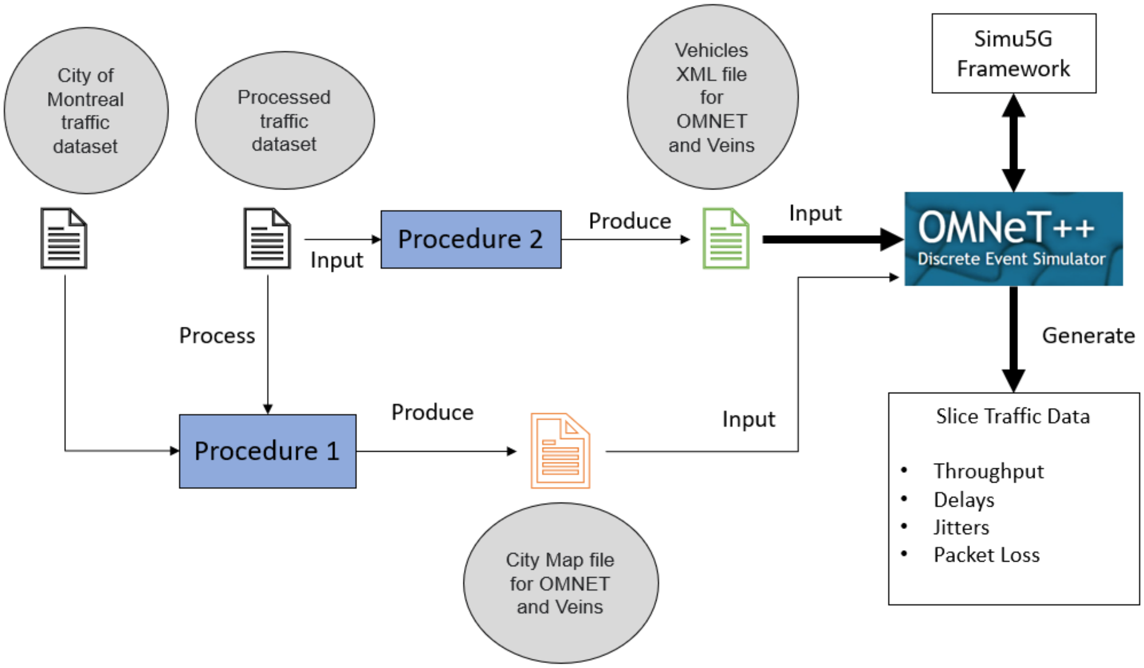


Figure 2: 5G traffic generator workflow

Then intersections are connected by the lanes to form a directed graph. Example figure is shown in Fig. 3

The second function of procedure 1 is to count the traffic of three kinds of vehicles: cars, pedestrians and trucks. The exact traffic numbers are shown in Fig. 4. Also, we summarized the traffic from the original time slot fifteen minutes to one hour to make it easier for the integration with OMNET++ in the next step.

In Procedure 2, the traffic number of vehicles through intersections are further processed to generate random trips. Therein, random trips for each vehicle are created: we first set one hour departure time for each vehicle so that our simulation is near to the city of Montreal traffic. Secondly, a pair of source and destination coordinates are chosen from map file based on the cumulative travelling distance. Starting from the source, the first possible connected lane will be selected. And, the selection algorithm is stopped by the condition of minimum and maximum distance. At last, random routes associated with each

Figure 3: Example of generated city map

vehicle connected to source and destination are created . We tuned the traffic parameters, including the min/max distance, the travelling speed and departure time of each type of vehicle, to guarantee the travelling time for each route is an approximately the same interval. The output of Procedure 2 is a Extensible Markup Language (XML) file containing the following information:

- vehicle type (car, truck, pedestrian)
- unique ID (per each vehicle)
- Departure time
- Route from source to destination (per vehicle)
- Travelling speed (per vehicle type)

In summary, we used the traffic dataset from the city of Montreal to generate the map and random routes for different types of vehicles. We first selected the traffic information of eighteen intersections and summarized the traffic number of each type of vehicle in 24 hours. The traffic pattern is shown in Fig. 4. A random route algorithm is implemented to generate the source to destination pair for each vehicle to represent and simulate this traffic pattern. In order to control the travelling time, we tuned the traffic parameters like min/max

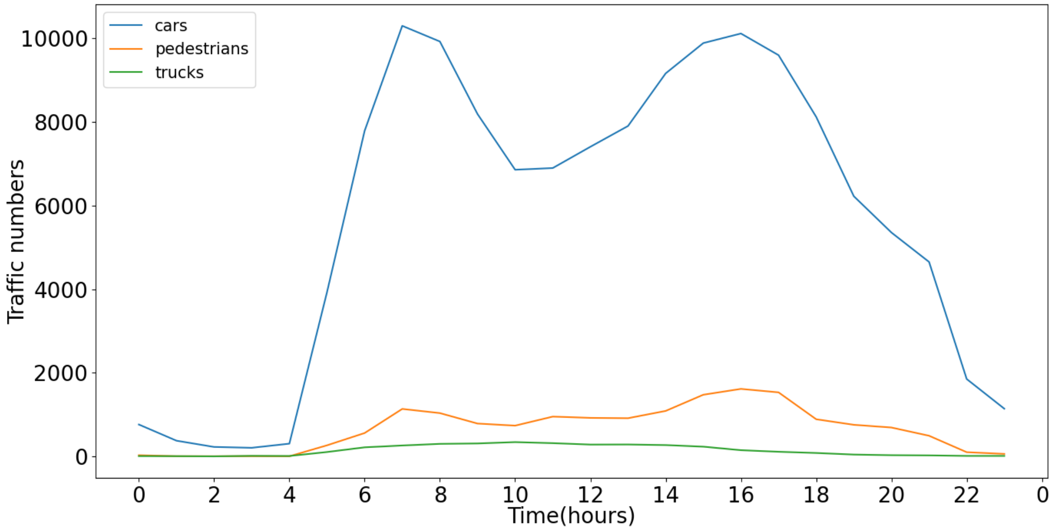


Figure 4: Traffic summary of number of cars, pedestrians, and trucks

distance and travelling speed. It is necessary to explain that the routes are not guaranteed to travel through certain intersections w so that the final simulation traffic is not exact traffic in these eighteen intersections. We simulated the traffic ad hoc to represent the traffic in a whole area instead of the specific intersection.

2.4 Network Slicing

The current version of Simu5G does not include network Slicing, so besides the functions already available, we have added network slicing across the RAN, the TN, and the 5GC. The RAN is sliced at each base station, called logical 5G radio node (gNB). Each slice creates a tunnel from the User Equipment's (UEs) that is assigned to a particular slice to the base station, and then the traffic is routed to the TN. The Transport Network consists of routers, specifically Multi-protocol Level Switching (MLPS) routers; thus, the packet transmission over the TN is done by switching labels. The User Plane Function (UPF) connects the base station of RAN and the Internet. It is used to route and forward the packets between base stations and servers, buffer the traffic for idle mobile devices, account for the number of subscriber usage, and guarantee the Quality-of-Service. Also, UPF needs to be extensible to support the even-growing speeds and even-increasing number of subscribers [21]. We must highlight that although MLPS was already supported by OMNET++, its interface to the 5G network had to be added.

2.5 Virtual Network Functions

As part of the network slicing, we have also included VNF consumption model in the UPF. UPF is one of the key VNF components deployed in the 5GC. Other VNFs such as Access and Mobility Management Function (AMF) and Session Management Function (SMF) are not implemented in our simulator. The main function of UPF is to route and forward packets between the RAN and servers. To measure the resource consumption of the UPF, we model three kinds of computing resources, namely Central Processing Unit (CPU), Random Access Memory (RAM), and Storage. Each VNF entity can be mapped to a single slice or multiple slices.

Our VNF infrastructure model is shown in Fig. 5, each VNF entity is represented by VNF1, VNF2, etc. The Max Resource Provider function (MaxProvider) determines the max resource capacity of each VNF entity. The VNFManager is in charge of routing packets to the assigned VNF based on the configuration provided by the SliceVNFMapProvider; similarly to the MaxProvider, this configuration can be dynamically updated during simulation. Notice that the ability to update the VNF configuration while the simulation is running is a new feature we added (OMNET++ does not natively support to change parameters during run time), Allocating, de-allocating, and validating the resource consumption is done by each VNF, whenever a packet is served, the changes are notified to the VNF by the The Point-to-Point Protocol (PPP) module. Packets are scheduled in a First In First Out (FIFO) queue regime and are forwarded to the right PPP interface (Up Link (UL)/Down

Link (DL)). If the current VNF's utilization exceeds the defined max capacity, packets are put in the queue and are dropped if they are not served at a given amount of time.

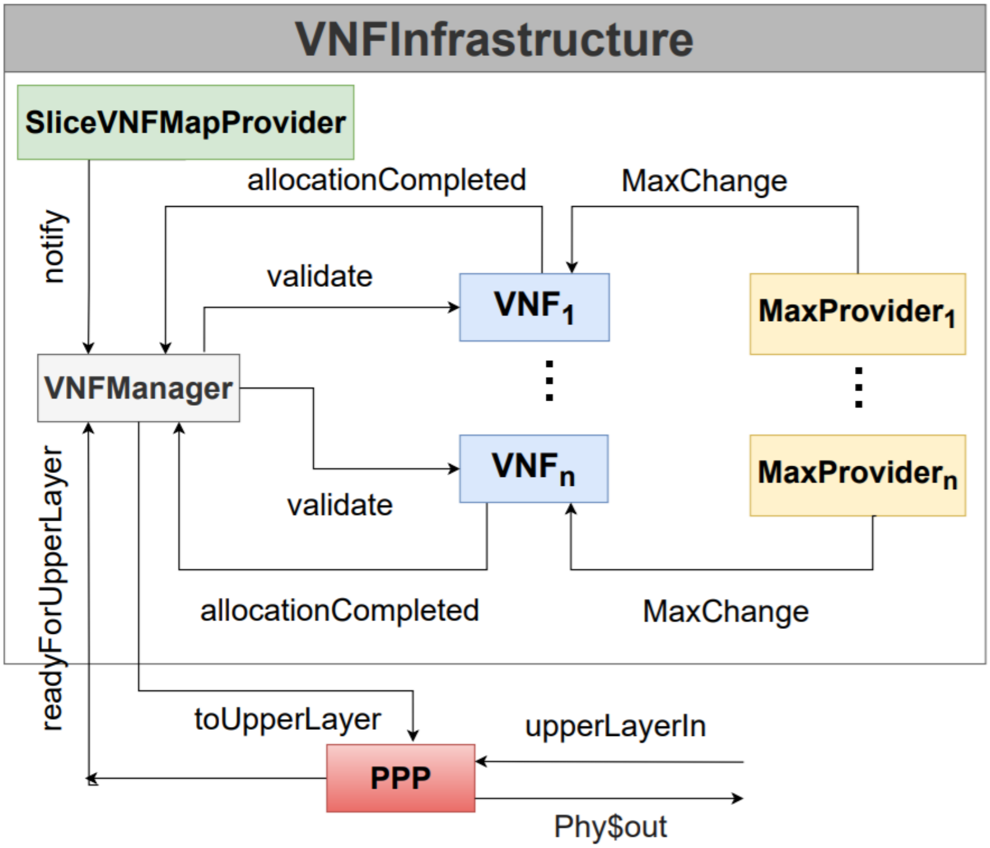


Figure 5: VNF resource allocation

To calculate the CPU utilization, we have used the assumption that CPU is the essential resource and its utilization is proportional to the traffic handled by the VNF [22, 23, 24].

There is no available model for RAM and Storage, and they may differ from one VNF to the next. We, therefore, devised a statistical model that draws a value from a distribution p and then multiplies it by the user's application requirement. As a consequence, each application has a specific requirement in terms of CPU, RAM, and Storage. To simulate the delay inside the VNF, we assume that the delay increases with the CPU, RAM, and Storage utilization. We then model the delay by using an exponential distribution as described in the following equation:

$$Delay_{\text{VNF}} = e^{kc} + e^{gm} + e^{hs} - 3, \tag{1}$$

where k, g, h are parameters that control the maximum delay on each resource, and c, g, s are the current VNF's utilization on each resource.

All the described components of our 5G traffic generator are implemented and integrated into OMNET++ to be able to observe and record the Key Performance Indicator (KPI)s for each slice.

Chapter 3

Simulation Setup

3.1 Simulation Setup

This chapter describes the network setups established in the environment to test the simulation. We also describe the traffic information associated with each slice and the description of applications used by the different types of UEs. At last, all the essential VNF related parameters are listed (specifications of the hardware and OS).

3.2 End-to-End Network Slicing

Fig. 6 shows our 5G network setup, eight gNB are connected via MLPS label Switching Router (LSR), this corresponds to the construction of the RAN domain which receives and grants the requests from UEs, each gNB contains a list of IP addresses attached to each slice and serve to route the traffic of the Transport Network. Onto the TN, LSRcore routers (LSRcore2 and LSRcore4) receive the traffic granted from the RAN, these routers connected to LSRcore1 and LSRcore3, which connected to LSRupf. In turn, LSRupf is then connected to the 5G core element UPF which is processes (buffering UL and DL data) and forwards the User data to the rightful servers. At the UPF a single VNF for each slice is modelled to measure the three computing resource: CPU, RAM, and Storage from UPF to the application servers.

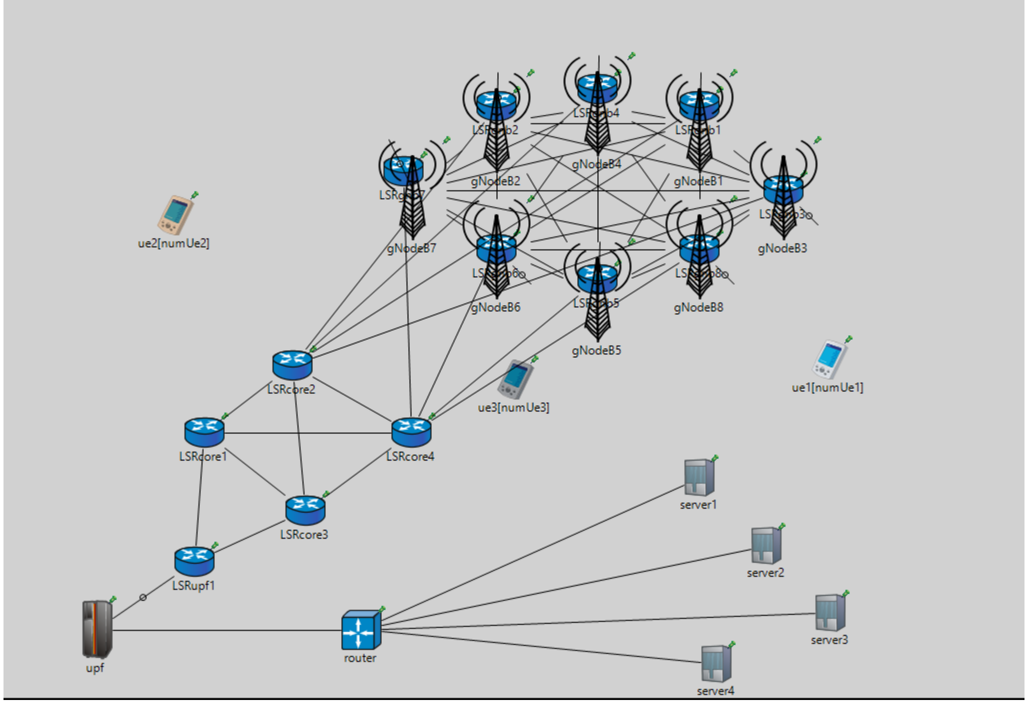


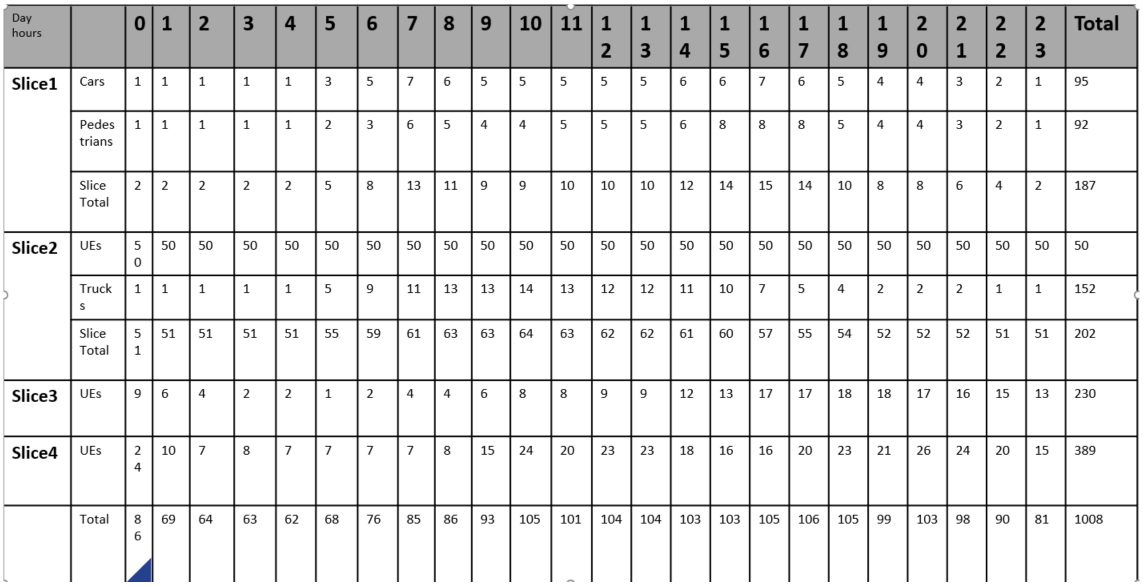
Figure 6: Illustration of our 5G network setup

3.3 Slices: Traffic Summary

Table 1 summarizes the traffic information configured in the simulation, each slice consist of a specific type of traffic and an application. Fig. 7 shows the detail traffic of four slices every hour.

Table 1: Type of traffic

Slice Number	**Slice Category**	**User Device Type**	**Application**
Slice 1	eMBB	Car & Pedestrians	HD Video
Slice 2	mMTC	Truck & Static Users 1	CBR
Slice 3	URLLC	Static Users 2	LR Video
Slice 4	VoIP	Static Users 3	VoIP



Day hours		0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	Total
Slice1	Cars	1	1	1	1	1	3	5	7	6	5	5	5	5	5	6	6	7	6	5	4	4	3	2	1	95
	Pedestrians	1	1	1	1	1	2	3	6	5	4	4	5	5	5	6	8	8	8	5	4	4	3	2	1	92
	Slice Total	2	2	2	2	2	5	8	13	11	9	9	10	10	10	12	14	15	14	10	8	8	6	4	2	187
Slice2	UEs	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50
	Trucks	1	1	1	1	1	5	9	11	13	13	14	13	12	12	11	10	7	5	4	2	2	2	1	1	152
	Slice Total	51	51	51	51	51	55	59	61	63	63	64	63	62	62	61	60	57	55	54	52	52	52	51	51	202
Slice3	UEs	9	6	4	2	2	1	2	4	4	6	8	8	9	9	12	13	17	17	18	18	17	16	15	13	230
Slice4	UEs	24	10	7	8	7	7	7	7	8	15	24	20	23	23	18	16	16	20	23	21	26	24	20	15	389
	Total	86	69	64	63	62	68	76	85	86	93	105	101	104	104	103	103	105	106	105	99	103	98	90	81	1008

Figure 7: Slices traffic summary by hours

Traffic variation over time is another vital aspect of our traffic generator. Start and finish times of the different UEs among different slices are controlled according to our traffic sources of application usage. The presence of vehicles (cars, pedestrians, and trucks) are extracted from the data of the city of Montreal [14]. In addition, the number of users in Slice 3: Low Resolution (LR) video is converted from [15] to analyze the daily usage of youtube through a campus WiFi network and the users in Slice 4: VoIP, is extracted from the number of calls recorded by a real-life VoIP provider in [16].

It is also worth mentioning that scaling factors are applied for each slice because too many connected user devices would unexpectedly terminate the simulation. It is because the number of UEs handled by OMNET++ is limited(in a reasonable amount of time). Nevertheless, the traffic patterns were preserved.

3.4 User Applications

We have defined our application libraries (based on the OMNET++ in-built applications) like video, Constant Bit Rate (CBR), and VoIP to accommodate new features. For example, some functions added to control the application's start time and finish time, VNF trailers also included in calculating the resource consumption on each VNF. Traffic parameters are also set to simulate real-life applications as shown in Table 2, For example, we want to simulate HD video in slice 1. By definition of HD Video [25], the minimum requirement is 720×1280 which is also called 720p. It consumes around 0.9 MB in each video frame, then if we consider an average frame rate of 30 Frame Per Second (FPS) [26], and each packet has 1024 bytes [27] thus 900 packets per frame, and 27000 Packets Per Second (PPS), so the send interval 1/PPS = 0.037ms.

Table 2: Parameters of the slices' applications

Slice Number	**Sending Rate (ms)**	**Packet Size (B)**	**After Scaling (ms)**	**Scaling Factor**	**Data Rate (Mbps)**
Slice 1	0.037	DL: [464, 576]	3.7	100	1.072
Slice 2	0.3	UL: [184, 256]	30	100	0.056
Slice 3	0.56	DL&UL: [454, 576]	56	100	0.056
Slice 4	0.2	DL&UL:[128, 232]	20	100	0.069

3.5 VNFs Parameters

VNF related parameters also need to be configured at the beginning of the simulation. We specify the maximum capacity of the CPU, RAM, and Storage and record the utilization of each resource. The maximum capacity can be dynamically changed during simulation. In the experiments, the maximum values are static. The detailed configuration is presented in Table 3.

Table 3: VNF parameters

Slice number	**VNFs Computational Resources**					
	cpu	***ram***	***storage***	***max cpu***	***max ram***	***max storage***
Slice 1	0.35/0.25	64MB	128MB	1.5	928MB	1.82GB
Slice 2	0.1	32MB	64MB	1	528MB	1.20GB
Slice 3	0.2	58MB	108MB	1.2	548MB	1.36GB
Slice 4	0.15	56MB	104MB	0.8	256MB	512MB

Chapter 4

Results

4.1 Results and Analysis

In this Chapter, results generated by our 5G traffic generator are presented, including the throughput, End-to-End delays, Jitters, packet losses, and VNFs' consumption of CPU, RAM, and Storage, of the four slices. All the results are generated using the configuration parameters mentioned in Chapter 3.1 and exported to a self-developed python Application Programming Interface (API) to be processed, plotted, and analyzed.

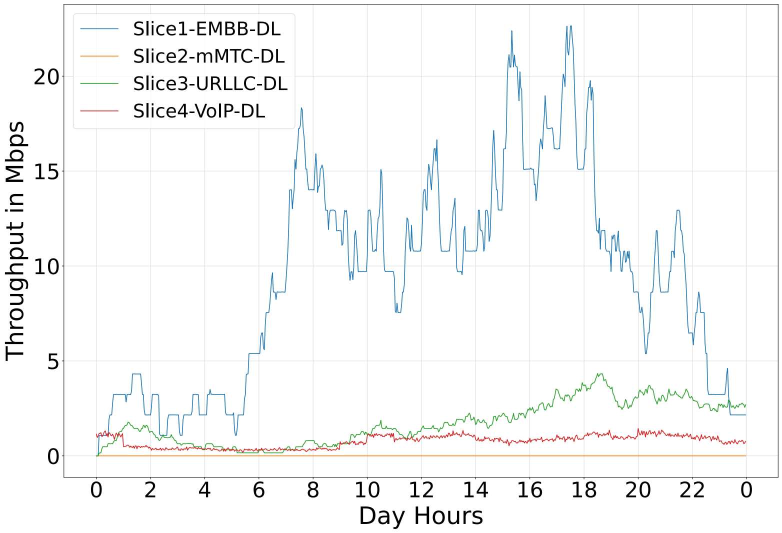
4.2 Experiments Results

Fig. 8 shows the DL and UL throughput in each of the four slices. It can be seen in the figures that the traffic varies over time, and the peaks and deeps vary based on the types of application.

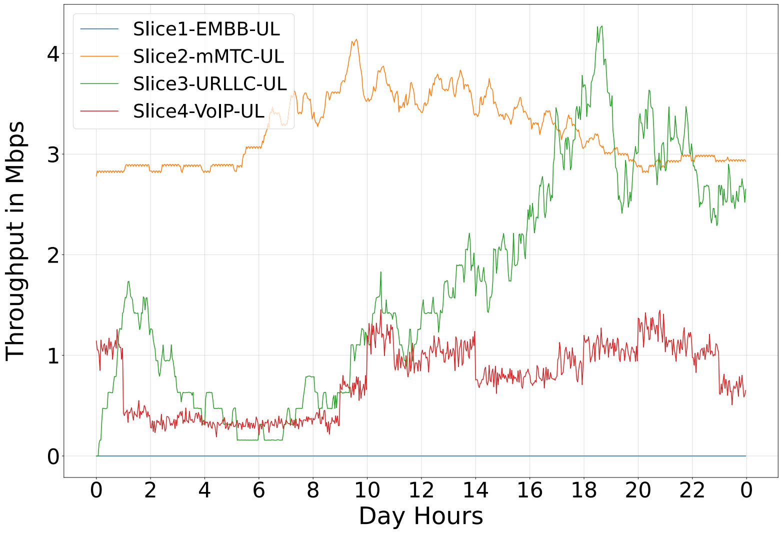
Delays, jitters, and packet losses are shown in Fig. 9, 10 and 11, which are three of the main KPIs to assess the performance of a network. Jitters are commonly used to measure the packet delay variation. It can be either positive or negative. Jitter has a huge impact on network performance. The result shows that the VoIP slice has the highest jitters range from

-6 ms to 6 ms. We also measure the packet losses11. The packet losses rate, in general, are less than 2%. Moreover, in the DL direction, Slice 1's packet losses are even less than 0.2%.

Finally VNFs' consumption is depicted in Fig. 12. Slice 1, 3, 4' resources consumption percentage (CPU, RAM, and Storage) are around 20%-30%. In contrast, Slice 2 is around 70% to show how the usage for VNF can reach the maximum. This ratio is determined by the values set in Table 3 and might be adjusted depending on the testing needs. Here, we can also observe that there is a correlation between resource consumption and throughput, as is generally expected. The more traffic there is the more use of the CPU, RAM, and Storage.

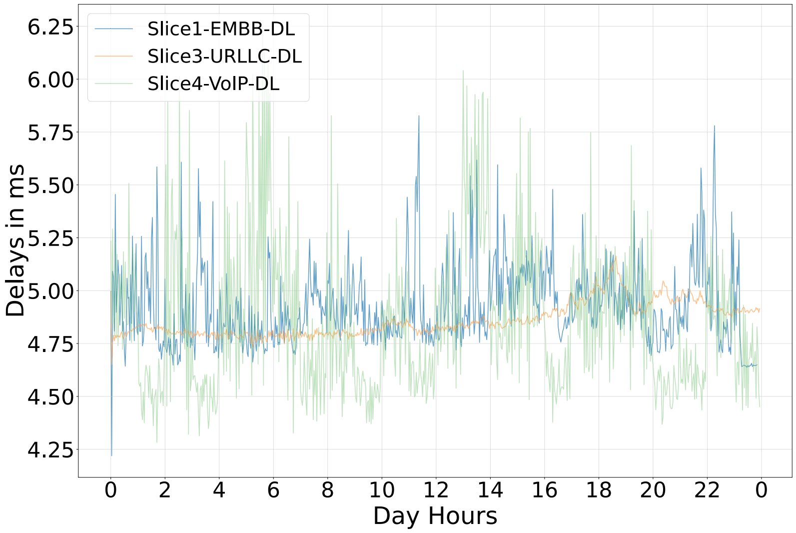


(a) Downlink throughput

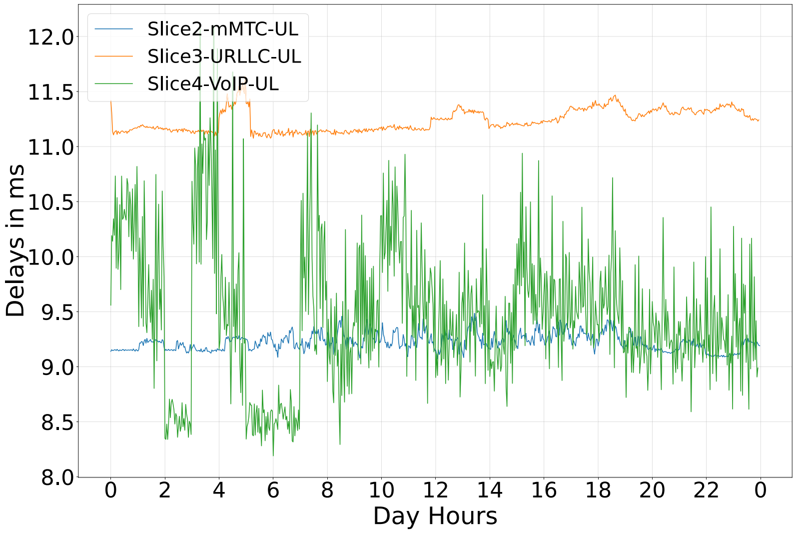


(b) Uplink throughput

Figure 8: Throughput

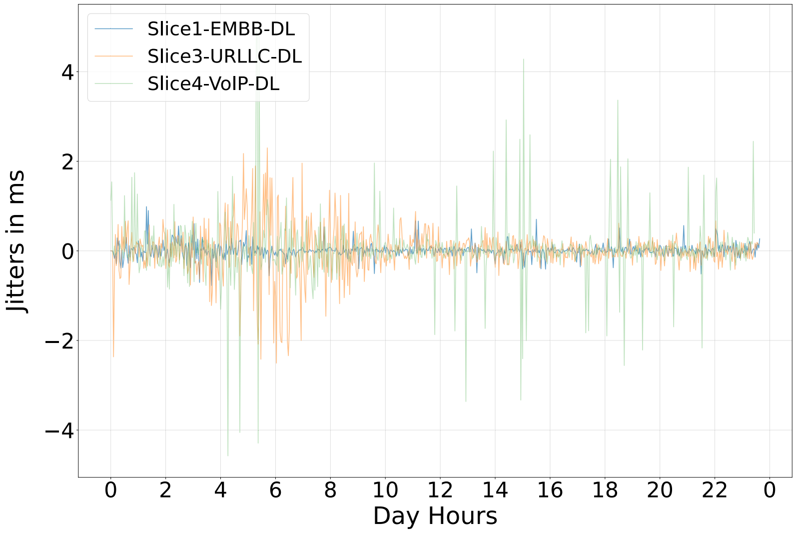


(a) Downlink

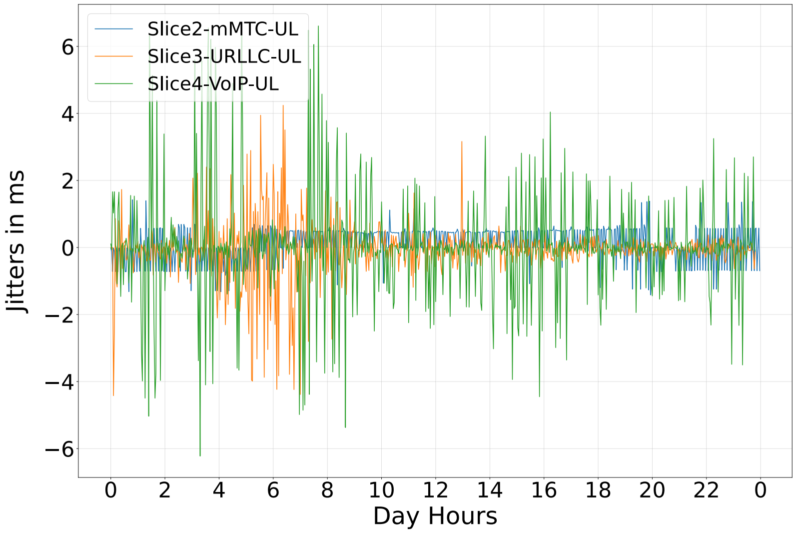


(b) Uplink

Figure 9: End-to-end delay

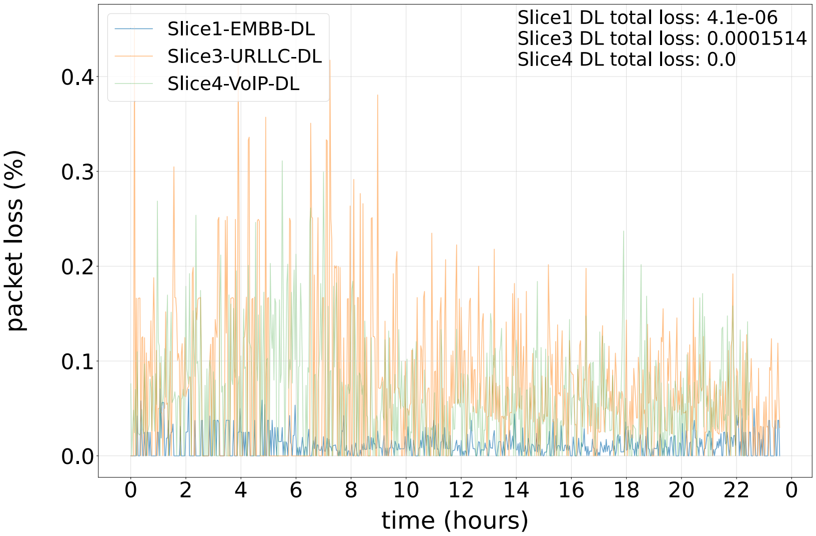


(a) Downlink

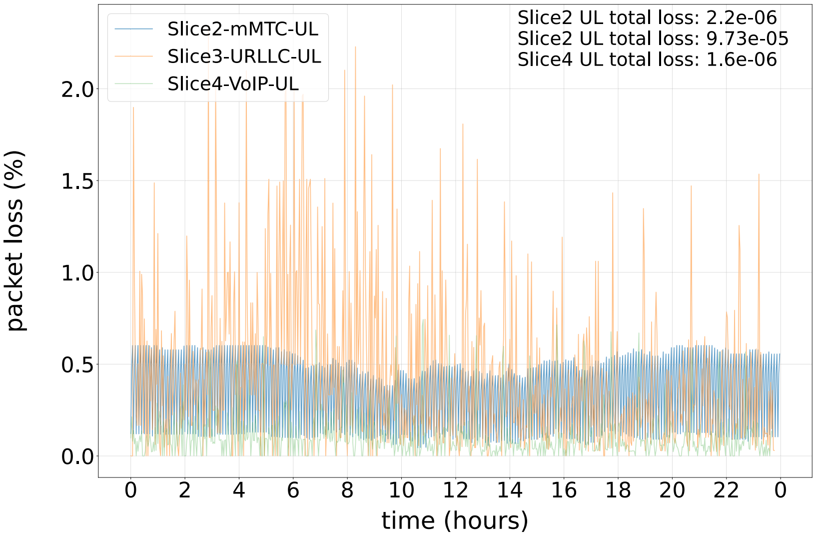


(b) Uplink

Figure 10: Jitter

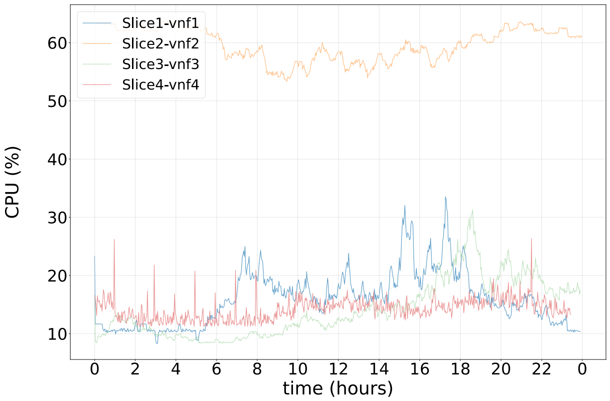


(a) Downlink

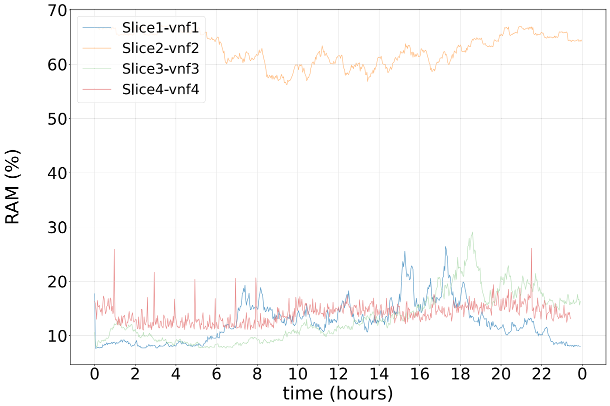


(b) Uplink

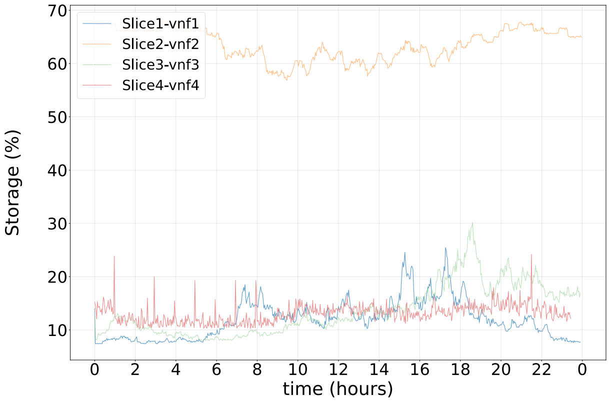
Figure 11: Packet loss



(a) CPU usage



(b) RAM usage



(c) Storage usage

Figure 12: VNF

Chapter 5

Conclusion and Future work

5.1 Conclusion

This thesis introduces a novel traffic generator that allows us to generate dynamic traffic using a network simulator: OMNET++. It showed and discussed that our generator is mainly designed and implemented for evaluating and testing 5G networks with slicing capabilities. The proposed approach generates traffic for each network slice according to the characteristics and requirements of a given service. Additionally, a VNF consumption model is integrated into our 5G traffic generator so that we can measure the usage of CPU, RAM and Storage for each slice in the UPF.To test and show the results, corresponding traffic and VNF parameters are also given in the context, and they can be tuned and adjusted in future work.

Different applications: Video, CBR and VoIP have been implemented to simulate the traffic variation for different 5G slices: namely eMBB, URLLC, mMTC, and VoIP. Finally, we show the key performance metrics of the slices that can be used to analyze the network behaviour.

5.2 Future work

This section discussed the potential future work that could be accomplished using our traffic generator, including two parts: extension and potential use.

- The first one is related to the extensions: The traffic volume can be increased so as the traffic of some slices are extracted from the data of the City of Montreal, two directions can achieve this extension: the length can be extended to days, weeks or even months to generate more data in the same paradigm to generate more relevant network data. the second direction is to adjust the scaling factor which adapts to the OMNET++, the limitation on the number of user devices could be solved in a future version of OMNET++.
- The second is the use of the generator: The output of the traffic generator can be applied in many fields. Ongoing research is in progress with a closed-loop algorithm, which sets a static threshold on different KPI for each slice and tests with various actions to test if the threshold is violated or not. Another application is the essential data generation for traffic prediction, which allows the researchers to perform the network actions proactively.

Bibliography

[1] J. Navarro-Ortiz, P. Romero-Diaz, S. Sendra, P. Ameigeiras, J. J. Ramos-Munoz, S. Lopez, and M. Juan. A survey on 5G usage scenarios and traffic models. *IEEE Communications Surveys Tutorials*, 22(2):905–929, 2020.

[2] M. Shafi, A.F. Molisch, P.J. Smith, T. Haustein, P. Zhu, and P. De Silva. 5G: A tutorial overview of standards, trials, challenges, deployment, and practice. *IEEE Journal on Selected Areas in Communications*, 35(6):1201 – 1221, June 2017.

[3] Ericsson. Mobility report. `https://www.ericsson.com/en/mobility-report`, June 2021.

[4] K. Flynn. RAN sharing. `https://www.3gpp.org/news-events/3gpp-news/1592-gush`, 2021.

[5] S. Dahmen-Lhuissier. Open source mano: Open source solutions: Mano nfv.

[6] G. Nardini, D. Sabella, G. Stea, P. Thakkar, and A. Virdis. Simu5G–an OMNeT++ library for end-to-end performance evaluation of 5G networks. *IEEE Access*, 8:181176–181191, 2020.

[7] P. Subedi, A. Alsadoon, P.W.C. Prasad, S. Rehman, N. Giweli, M. Imran, and S. Arif. Network slicing: a next generation 5G perspective. *EURASIP Journal on Wireless Communications and Networking*, 102:72–79, 2021.

[8] G. Soos, D. Ficzere, and P. Varga. Towards traffic identification and modeling for 5G application use-cases. *Electronics*, 9(4), 2020.

[9] T. Deinlein, R. German, and A. Djanatliev. 5G-sim-V2I/N: Towards a simulation framework for the evaluation of 5G V2I/V2N use cases. *European Conference on Networks and Communications (EuCNC)*, pages 353–357, 2020.

[10] M. Emara, M. C. Filippou, and D. Sabella. Mec-assisted end-to-end latency evaluations for C-V2X communications. *European Conference on Networks and Communications (EuCNC)*, pages 1–9, 2018.

[11] Q. Guo, R. Gu, Z. Wang, T. Zhao, Y. Ji, J. Kong, R. Gour, and J. P. Jue. Proactive dynamic network slicing with deep learning based short-term traffic prediction for 5G

transport network. In *2019 Optical Fiber Communications Conference and Exhibition (OFC)*, pages 1–3, 2019.

[12] G. Lee, J. Lee, Y. Kim, and J. G. Park. Network flow data re-collecting approach using 5G testbed for labeled dataset. *2021 23rd International Conference on Advanced Communication Technology (ICACT)*, pages 254–258, 2021.

[13] D. Corcoran, P. Kreuger, and C. Schulte. Efficient real-time traffic generation for 5G RAN. In *IEEE/IFIP Network Operations and Management Symposium (NOMS)*, pages 1–9, 2020.

[14] Ville de Montréal. Counts of vehicles, cyclists and pedestrians at intersections with traffic lights. *Urban Planning and Mobility Department*, 2020.

[15] J. Laulajainen, A. Arvidsson, T. Ojalaand J. Sepp/"anen, and M. Du. Study of youtube demand patterns in mixed public and campus wifi network. *International Wireless Communications and Mobile Computing Conference (IWCMC)*, pages 635–641, 2014.

[16] P. Galiotos, T. Dagiuklas, and S. Kotsopoulos. Call-level VoIP traffic modelling based on data from a real-life VoIP service provider. *IEEE Globecom Workshops (GC Wkshps)*, pages 1–7, 2015.

[17] A. Virdis, G. Stea, and G. Nardini. SimuLTE - a modular system-level simulator for LTE/LTE-a networks based on OMNeT++. *4th International Conference On Simulation And Modeling Methodologies, Technologies And Applications (SIMULTECH)*, pages 59–70, 2014.

[18] C. Sommer, R. German, and F. Dressler. Bidirectionally coupled network and road traffic simulation for improved IVC analysis. *IEEE Transactions on Mobile Computing*, 10(1):3–15, 2011.

[19] P. A. Lopez, M. Behrisch, L. Bieker-Walz, J. Erdmann, Y. Flötteröd, R. Hilbrich, L. Lücken, J. Rummel, P. Wagner, and E. Wiessner. Microscopic traffic simulation using SUMO. *21st International Conference on Intelligent Transportation Systems (ITSC)*, pages 2575–2582, 2018.

[20] OpenStreetmap Foundation (OSMF). web software, 2021.

[21] R. MacDavid, C. Cascone, P. P Lin, B. Padmanabhan, A. ThakuR, L. Peterson, J. Rexford, and O. Sunay. A P4-based 5G user plane function. In *Proceedings of the ACM SIGCOMM Symposium on SDN Research (SOSR)*, page 162–168, 2021.

[22] S. Van Rossem, W. Tavernier, D. Colle, M. Pickavet, and P. Demeester. Profile-based resource allocation for virtualized network functions. *IEEE Transactions on Network and Service Management*, 16(4):1374–1388, 2019.

[23] S. Demirci and S. Sagiroglu. Optimal placement of virtual network functions in software defined networks: A survey. *Journal of Network and Computer Applications*, 147:102424, 2019.

[24] F. X. Wang, R. L. Ling, J. Zhu, and D. Li. Bandwidth guaranteed virtual network function placement and scaling in datacenter networks. *IEEE 34th International Performance Computing and Communications Conference (IPCCC)*, pages 1–8, 2015.

[25] Bl blog: Online camera gear rental, news, tutorials, Jul 2021.

[26] Bl blog: Online camera gear rental, news, tutorials, Jul 2021.

[27] V. Joseph and B. Chapman. Class-of-service requirements for quad-play networks. In *Deploying QoS for Cisco IP and Next Generation Networks*, pages 71–114. Morgan Kaufmann, 12 2009.

Appendix A

Procedure 1

This is the pseudo code for procedure 1 in Chapter 2, Section 3. (2.3)

Algorithm 1 Pseudo code of procedure 1

```
Input ← Raw data from City of Montreal [14]
Output ← Overall hourly processed Montreal traffic data
procedure 1
    for each 5 minutes in the Montreal data set do
        for each Intersection do
            Get the number of pedestrians, cars and trucks from all direction
        end for
    end for
    Compute hourly numbers for pedestrians, cars and trucks for each intersection
    Sum over each type of traffic (pedestrians, cars, trucks) for all intersections
end procedure
```

Appendix B

Procedure 2

Command: Sumo -randomTrips **(vehicle_type, numbers, max_travel_lanes, vehicle_scaling_factor)**
This is the pseudo code for procedure 2 in Chapter 2, Section 3. (2.3)

Algorithm 2 Pseudo code of Procedure 2

```
Input ← (Montreal Map file; Hourly traffic number of each type)
Output ← Hourly set of random routes
procedure 2
    for each Type of Vehicle do
        Obtain the Hourly traffic number of each type (Car, Pedestrians, Trucks) generated in the procedure 1
        for $hours(time) = 1, 2, \ldots, 24$ do
            current_intersection ← start_intersection
            current_travel_lanes ← 0
            while current_travel_lanes < max_travel_lanes do
                Iterate all the connected lanes, add to random routes
                current_intersection ← next_intersection
            end while
            Generate random routes in ad hoc way across Map of Montreal
        end for
    end for
    Combine the files for all types of vehicles
end procedure
```

Appendix C

Additional procedure

In addition,outcome of procedure 2 is integrated into OMNET++ to assign the traffic of each slice as followings

```
procedure
    Slice 1 ← Cars and Pedestrians produced in procedure 2
    Slice 2 ← Trucks produced in procedure 2 and constant number of Static Users
    Slice 3 ← Static Users obtained from usage of video [15]
    Slice 4 ← Static Users obtained from usage of VoIP [16]
end procedure
```

Appendix D

Summary of Traffic per Slice

Figure D.1 shows the traffic summary of all four slices after running Sumo and be integrated into OMNET++.

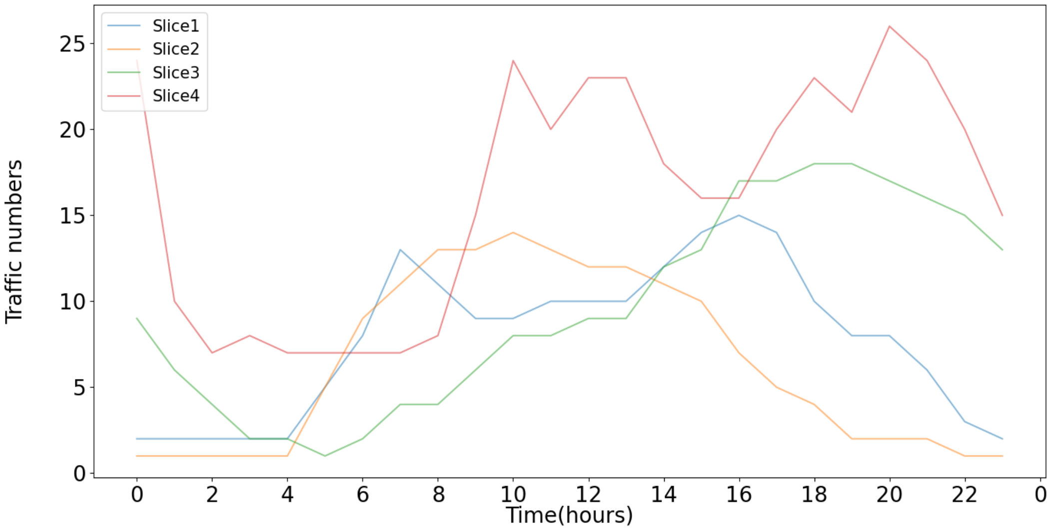


Figure D.1: Slice traffic summary