5G NR Simulation in NS-3

Assignment 13

MPA-KPM

10. 12. 2023

Authors: Bc. Adam Hora

Bc. Boris Hynšt

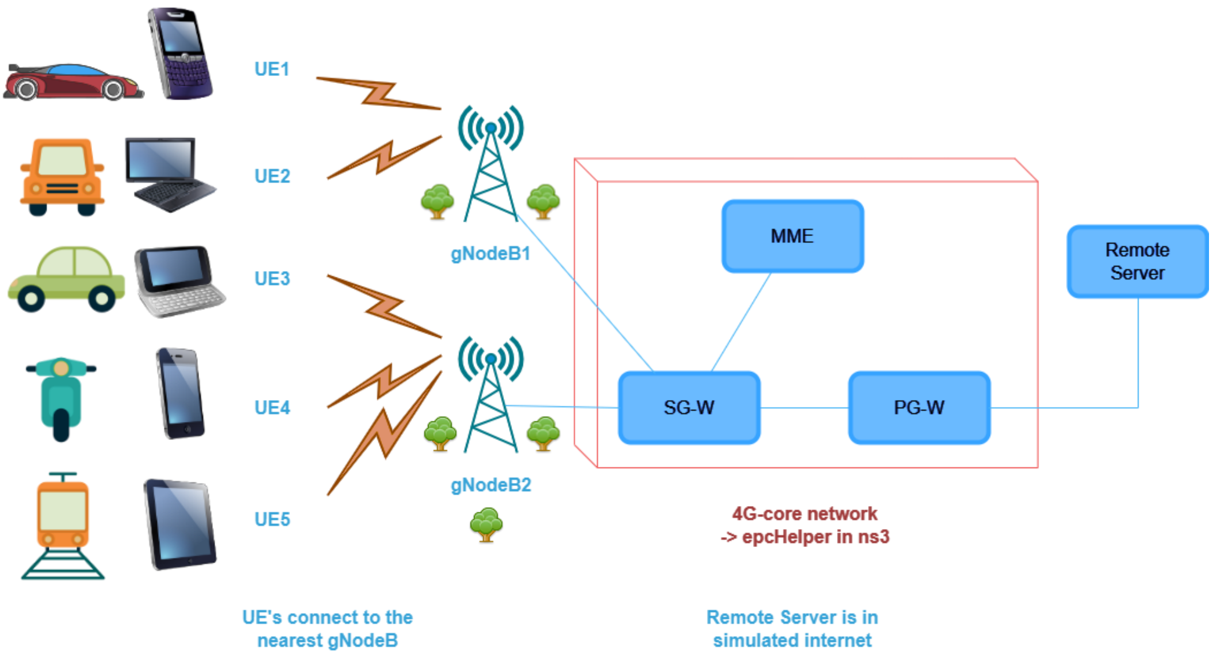
Bc. Jan Hrubý

Bc. Václav Pastušek

1. Introduction

The principal objective of this project is to offer participants practical exposure to simulating a 5G-NSA (Non-Standalone) network through the Network Simulator 3 (NS-3) framework. Participants will leverage their current knowledge to craft a detailed network scenario, fine-tune 5G New Radio (NR) parameters, and conduct an in-depth analysis of network performance.

2. Network design



3. Program launch

Before the program will be launched, it is important to place the source file 13th-group_5G-NSA.cc into folder .../lena5g/ns-3-dev/scratch, assuming Lena module is already installed and ready to use on the device.

The program can be run from the terminal, using command ./ns3 run "13th-group_5G-NSA --param=value". It is important that the terminal is opened from the .../lena5g/ns-3-dev folder. In order to print help with all the parameters into the terminal, command ./ns3 run "13th-group_5G-NSA --PrintHelp" can be issued.

Application parameters

parameter name	description	default value	unit
numerology	5G numerology value, must be 0 – 4.[1]	0	N/A
carrierFrequecy	Frequency of the carrier wave.	1 500 000 000	[Hz]
bandwidth	Bandwidth specification.	100 000 000	[Hz]
txPower	Transmitter power for the gNodeBs.	43	[dBm]
gNodeBNumber	Number of the gNodeBs in the simulation.	2	N/A
baseStationDistance	Distance between individual gNodeBs.	300	[m]
endUserDevicesNumber	Number of the UEs in the simulation.	5	N/A
packetSize	Payload of the packet.	1400	Bytes
packetsPerSecond	Packets send by devices generating traffic.	100	N/A
simTime	Duration of the simulation	2	[sec]
pcapFilesGeneration	Ability to enable / disable traffic capture.	false	N/A

Simulated scenarios

The assignement consisted of a few conditions, the simulation had to run without an error, it had to include at least two gNodeBs, five UEs and a remote server. UEs were supposed to stream video to the remote server while moving by a speed of a vehicle driving through the city. We have performed five different simulations, within two groups by two simulations and one group by one simulation.

First two groups of simulations were supposed to monitor simple network, which consisted of five UEs, those connected to two gNodeBs set up using Hexagonal grid scenario. They were then wired to 4G core network to simulate 5G NR Non-Standalone scenario using epcHelper. This helper is used for simplified and automated set up of individual 4G core nodes, namely SG-W, PG-W and MME. To the PG-W was then connected remote Server which was placed inside virtualised Internet. The final topology can be seen on the diagram below, this diagram has been generated by the script and visualised using NetAnim tool. Unfortunately we were not able to access MME node and move it to custom position.

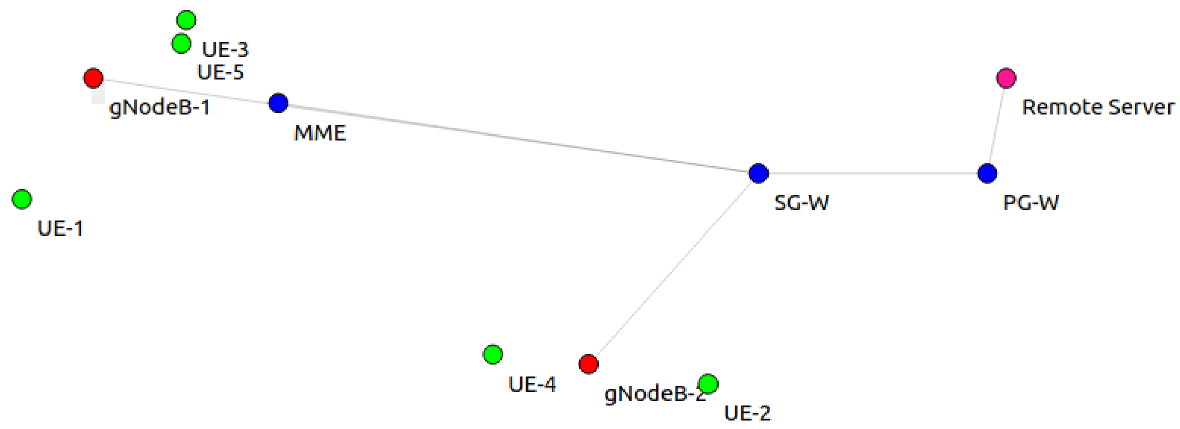


Figure 1: Green = End User Devices | Red = Base Stations | Blue = Core Network | Pink = Remote Server

[1] 1 – Numerology values can be found in 5G-Lena NR module overview from 2022, at www.cttc.es.

The script also automatically generates raw data which can be used to visualize the Hexagonal Grid Scenario. Our code includes function which automatically turns these raw data into .pdf file, using program gnuplot. The simple Hexagonal Grid Scenario for two gNodeBs is located on the picture below, more complicated example can be found within the last simulation.

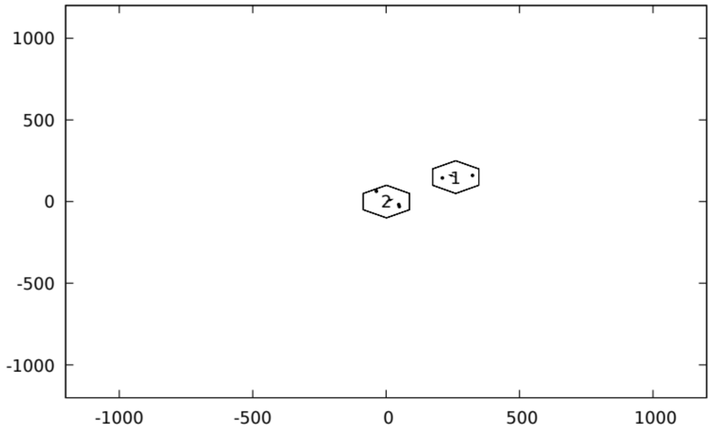


Figure 2: Hexagonal Grid Scenario diagram for 2 gNodeBs and 5 UEs

Comparison between simulations of the first group

This group was a short simulation, with duration of four seconds. The goal was to simulate and capture the difference between two exact scenarios of 5G NSA technology operating under normal conditions while using different numerologies. Most of the parameters were left default, but few key values were changed. Packet size was changed to 1430 Bytes, to better simulate video stream traffic. With this payload size, the final packet was about 1500 Bytes large after adding headers, that is usually the Maximum Transmission Unit allowed within most Ethernet networks. Packets per second were changed to 1300, this value better representes constant video streaming of HD quality. Simulation time was for puprose of the simulation lenghtened to 4 seconds. Comparison between the key metrics in the network can be seen in the tables below.

Metric/Node	**UE-1**	**UE-2**	**UE-3**	**UE-4**	**UE-5**
Tx Packets	5200	5200	5200	5200	5200
Rx Packets	5158	5163	5158	5156	5168
Packet Loss [%]	0.81	0.71	0.81	0.85	0.62
Throughput [Mbps]	16.712	16.728	16.712	16.705	16.744
Mean delay [ms]	8.836	7.497	10.374	11.197	8.321
Average sim. throughput [Mbps]	**16.720**				
Average sim. delay [ms]	**8.321**				

Figure 3: Metrics for simulation with numerology 0

Metric/Node	**UE-1**	**UE-2**	**UE-3**	**UE-4**	**UE-5**
Tx Packets	5200	5200	5200	5200	5200
Rx Packets	5177	5177	5178	5177	5178
Packet Loss [%]	0.44	0.44	0.42	0.44	0.42
Throughput [Mbps]	16.773	16.773	16.777	16.773	16.777
Mean delay [ms]	0.872	0.878	6.473	1.209	0.867
Average sim. throughput [Mbps]	**16.775**				
Average sim. delay [ms]	**2.060**				

Figure 4: Metrics for simulation with numerology 4

As the result of this simulation, we have found out the expected. Even tho the scenarios were set up exactly the same, choice of the numerology plays a huge role when it comes to the quality of the connection. The numerology four had four times better delay on avarage, which is most likely caused by multiple times larger subcarrier spacing and physical resource block, shorter OFDM symbol, ect.

Comparison between simulations of the second group

This group included longer simulation, with duration of ten seconds and mainly lowered transmittor power, which has been lowerd to 17 dBm. This was necessary for scenario where UEs are slowly leaving the range of the base stations, while keeping the duration of the simulation and build time of the simulation lower. The goal was again to compare captured key metrics between the exact same scenarios with different numerology in use. The parameters were pretty much the same as in the first simulation group, how ever aside the duration of the simulation and the lowering of transmittor power, amount of packets sent every second also had to be lowered to 300, due to NetAnims limitations. Comparison between the key metrics of this scenario can be found in tables bellow.

Metric/Node	**UE-1**	**UE-2**	**UE-3**	**UE-4**	**UE-5**
Tx Packets	3001	3001	3001	3001	3001
Rx Packets	2991	2988	2989	889	2792
Packet Loss [%]	0.33	0.43	0.40	70.38	6.96
Throughput [Mbps]	3.634	3.630	3.632	1.080	3.392
Mean delay [ms]	6.673	11.938	10.818	6.502	61.766
Average sim. throughput [Mbps]	**3.074**				
Average sim. delay [ms]	**19.539**				

Figure 5: Metrics for simulation with numerology 0

Metric/Node	**UE-1**	**UE-2**	**UE-3**	**UE-4**	**UE-5**
Tx Packets	3001	3001	3001	3001	3001
Rx Packets	2995	2821	2995	895	2496
Packet Loss [%]	0.20	6.00	0.20	70.18	16.83
Throughput [Mbps]	3.639	3.426	3.639	1.087	3.033
Mean delay [ms]	1.177	6.775	2.063	1.0910	36.282
Average sim. throughput [Mbps]	**2.965**				
Average sim. delay [ms]	**9.478**				

Figure 5: Metrics for simulation with numerology 4

As the result of this simulation, we have confirmed that the further the UE moves from the gNodeB, the higher the packet loss and ping gets. We have how ever found out an information that was new to us. In case of UE-4, it spawned within the parameters set at the beginning of the simulation. During the simulation however, it passed close to one of the base stations, that caused the UE to disconnect and because our script does not include handovers or reconnection scenario, the UE remains disconnected till the end of the simulation. On the pictures below can be seen the starting position of the UEs at the beginning of the simulation and the ending possition of the UEs at the end of the simulation

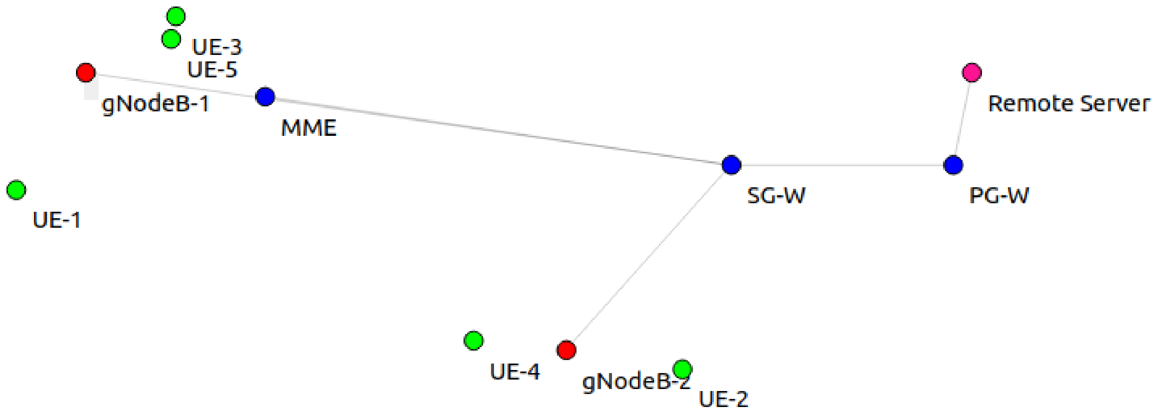


Figure 5: Simulation start

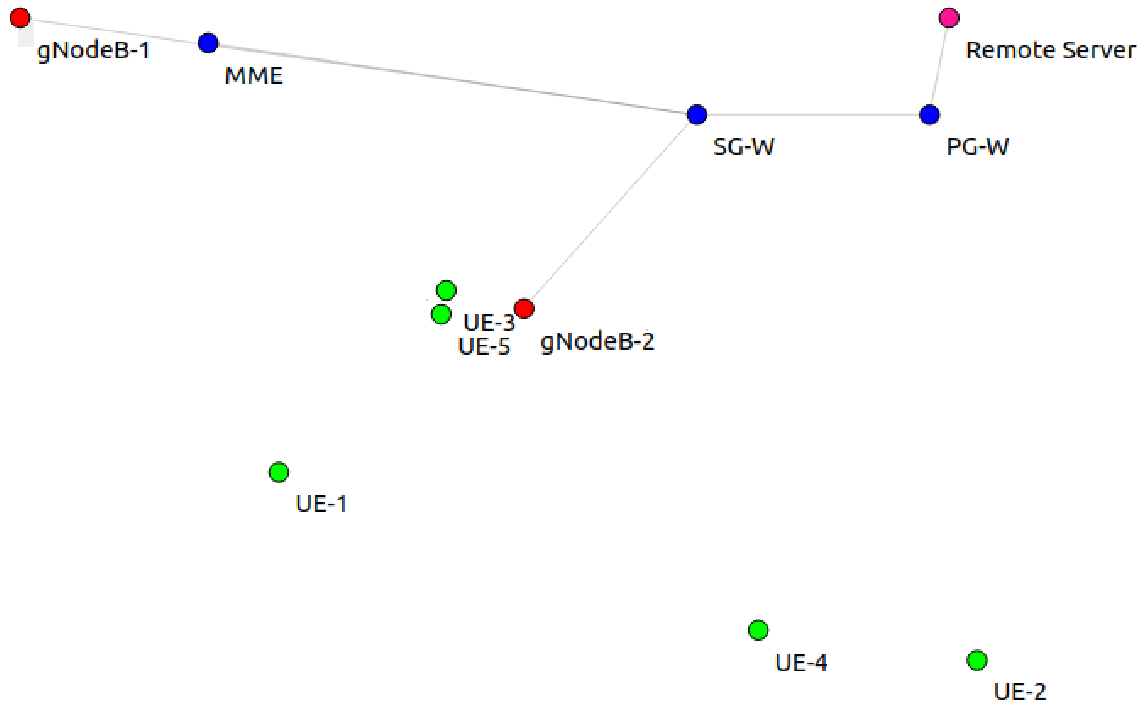


Figure 6: Simulation end

Third scenario group

This group was different from all the previous, because it includes only one simulation and it had more of an experimantal purpose. The target of this simulation was to push the Hexagonal Grid Scenario to it's limits, while still being able to execute the simulation without any errors. Because of this fact the network has different topology, it consists of the maximum number of gNodeBs supported by Hexagonal Grid Scenario, which is 37 and 72 UEs. For this to run properly, without running into any limitation issues with NetAnim and other tools, we had to tweak the parameters a little. The parameteres were same as for the first two simulations, with the obvious change of the number of UEs and gNodeBs, amount of packets sent every second had to be lowered to 50 as well as duration of the simulation to 1 second. The Hexagonal Grid Scenario diagram as well as the visual representation of the real network can be seen on the pictures below. Due to the amount of the output data, they can be found in an organised .txt file inside the .zip file along with this documentation.

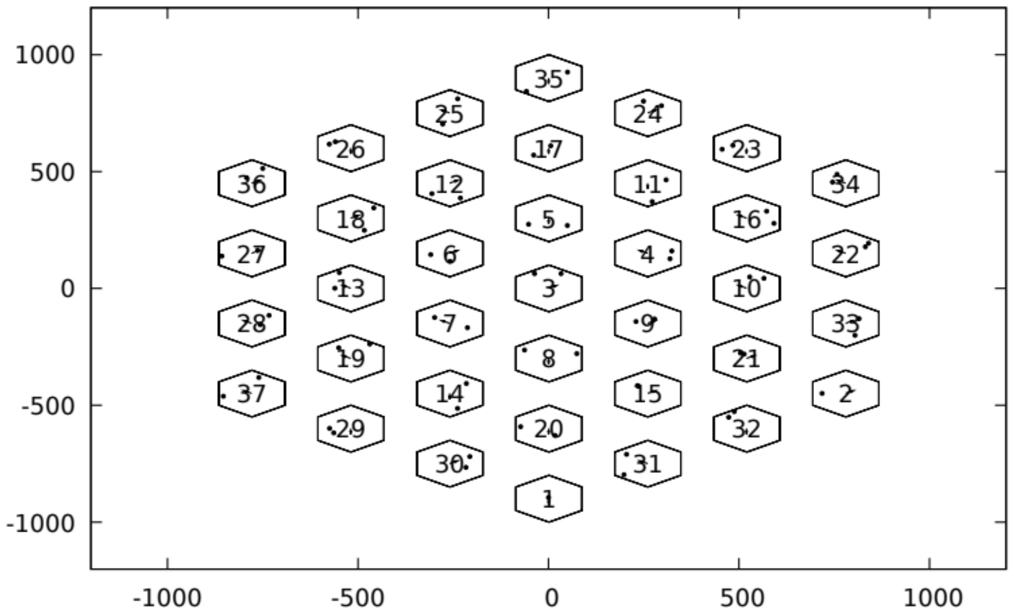


Figure 7: Hexagonal Grid Diagram

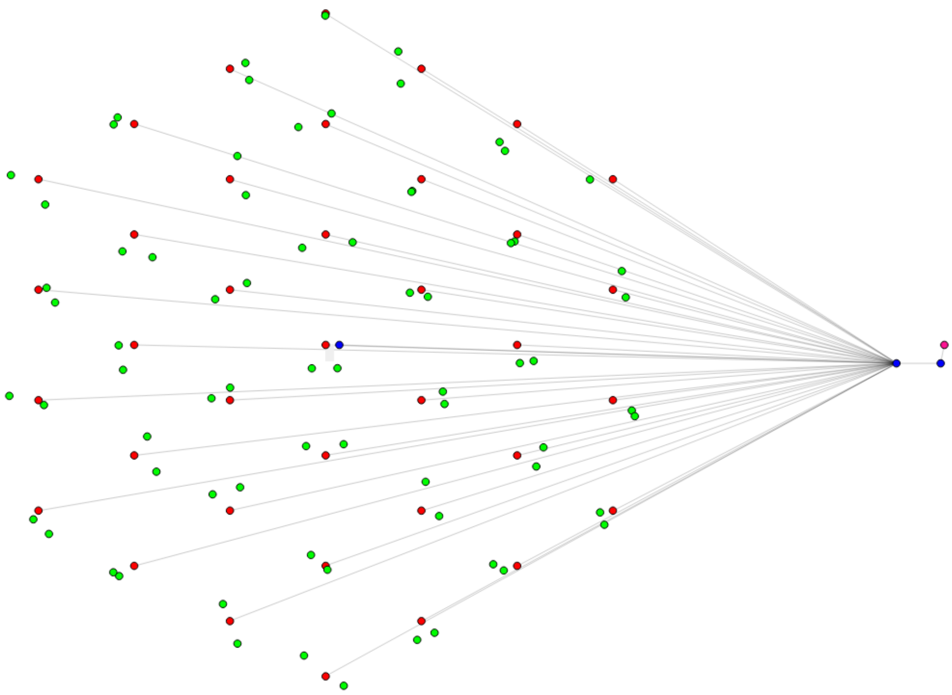


Figure 8: Visualization of a real network layout, using NetAnim tool.

Problems encountered

The main problem which we encountered during any part of the project, was lack of proper documentation, examples, or any other sources which could be used for better understanding of 5g lena module. Due to this problem we were forced to experiment and reverse engineer multiple functions, which took extreme amount of time and slowed down the project noticeably. Another problem was lack of function which would allow us to make a pointer for the MME node inside epcHelper, leading to inability to place the MME with the rest of the core network and giving it a proper mobility model, leaving it floating in the space.

Conclusions and future implementation

We believe that this project fulfills the requirements which have been set in the assignment. We have also put quite a lot of thought to the structure of the code as well as to its user friendliness. We have implemented simple funcions that ensure that all the files created by this code are located in the coresponding folder and are ready to be used. The code is also written in a form which should be understandable to any person with basic knowledge in the field and has proper comments to describe almost every function and its purpose within the simulation. This should allow for future expansions and improvements of the code without problems. As a future expansions or improvements could be seen for example use of changing speed and direction of the UEs, ability for the gNodeBs to handover individual UEs as they leave gNodeBs area to another. It could also be good idea to add a function, that allows for reconnect of the UE, if it comes too close to the gNodeB and is disconnected, or to prevent this from happening.