Emu5GNet: an Open-Source Emulator for 5G Software-Defined Networks

Tidiane Sylla
Univ Gustave Eiffel,
COSYS-ERENA,
F-77454 Marne-la-Vallée, France,
tidiane.sylla@univ-eiffel.fr

Leo Mendiboure
Univ Gustave Eiffel,
COSYS-ERENA,
F-77454 Marne-la-Vallée, France,
leo.mendiboure@univ-eiffel.fr

Marion Berbineau
Univ Gustave Eiffel,
COSYS,
F-59650 Villeneuve d'Ascq, France,
marion.berbineau@univ-eiffel.fr

Radheshyam Singh
DTU Electro,
Ørsteds Plads 343, Lyngby 2800,
Denmark,
radsi@fotonik.dtu.dk

José Soler
DTU Electro,
Ørsteds Plads 343, Lyngby 2800,
Denmark,
joss@fotonik.dtu.dk

Michael Stübert Berger
DTU Electro,
Ørsteds Plads 343, Lyngby 2800,
Denmark,
msbe@fotonik.dtu.dk

***Abstract*—Network emulators are a key component for the implementation and evaluation of new applications. A network emulator designed for 5G networks should support the deployment of a complete 5G architecture (core and access network) as well as the deployment of edge data centers (edge computing) and the implementation of flexible SDN (Software Defined Networks)/NFV (Network Function Virtualization) architecture. To date, there is no environment enabling this combination of features. In this paper, we introduce a new network emulator named Emu5GNet, allowing the deployment of applications in complex 5G networks enabling both emulation of the network and data processing. Then, we present the architecture set up to achieve this objective and identify some of the applications that could be deployed in this environment: network performance, mobility management, *etc*. We also propose a quick evaluation to show how our emulator could potentially be used.**

***Index Terms*—5G, SDN, NFV, Edge Computing, Emulator**

I. Introduction

Although their deployment has been initiated in many countries [1], 5G networks are still in the development stage. Many research and industrial projects focus on network issues [2] (e.g. Network Slicing) and data processing issues [3] (e.g. Edge Servers placement). Rapid prototyping and evaluation of such proposals usually rely on the use of emulation environments. These emulators provide a more realistic environment than simulators as they are based on the deployment of virtual machines and communication networks allowing the design of full-stack systems hosting real applications.

With 5G, different elements must be integrated into these emulators to provide a complete environment: 1) they should allow the emulation of multiple radio access technologies and the associated network cores (e.g. 5G core), 2) they should permit the deployment of different mobile nodes (e.g. pedestrians, cars) with variable applications requirements, 3) they should allow the deployment of Edge servers and 4) they should allow the implementation of a Software Defined Networking/Network Function Virtualization (SDN/NFV) architecture. In the literature, many network emulators have already been proposed focusing on a) SDN architectures deployment for wired networks [4]–[6], b) software defined wireless networks [7], [8], c) implementation of NFV solutions [9], [10] and d) deployment of 5G RAN [11].

Nevertheless, none of these emulators enable the implementation of complete 5G networks. That is why, in this paper, we introduce a new emulation platform, Emu5GNet, that could be used by researchers/industrial to propose and evaluate new solutions for future 5G networks. Such a platform could be used for different use cases: Internet of Things, vehicular communication networks, railway communication networks, *etc*. The main contributions of this paper are:

- A review of existing emulators for 5G networks;
- The design of a new emulator (Emu5GNet) for the rapid prototyping of network and data-centric 5G solutions;
- The identification of research directions that could be implemented and evaluated through Emu5GNet;
- The implementation of a simple Emu5GNet use case.

In this paper, Section 2 compares existing emulators for 5G networks. Then, Section 3 presents Emu5GNet. Finally, Section 4 introduces a use case of the proposed emulator.

II. Related works

In this section, existing emulators for 5G networks are presented and their limitations identified.

A. State-of-the-art solutions

Different types of emulators can be used for 5G networks:

- Emulator for 5G Radio Access Networks and 5G Core Networks: This type of emulator [11] or simulator [12] implements the whole 5G communication architecture. Such approaches aim to provide realistic results for emulated wireless communications. In addition, these platforms can be simply connected to open source 5G Core implementations [13]. However, these platforms

do not include the ideas of SDN control plane, data processing (data centers) or network functions;
- Emulator for wired SDN: These emulators, like EstiNet [14], Mininet [4] and Mininet-based emulators [6], [15], can be used to implement SDN-based core network solutions: path selection, load balancing, *etc*. Nevertheless, the potential applications of such environments for the integration of 5G applications remain limited;
- Emulator for wireless SDN: These emulators, including OpenNet [8] and Mininet-WiFi [7], allow the implementation of mobile nodes connected to access points (OpenNet: 4G, Mininet-WiFi: Wi-Fi) managed by SDN controllers. They can be used to implement solutions combining SDN and wireless communications (handover management for example);
- Emulator for SDN/NFV architectures: These emulators (e.g. Sonata [10]) complete SDN emulators with NFV components. This increases the number of applications that can be considered: network function placement, edge data center implementation, *etc*. However, these platforms do not consider wireless communications.

B. Positioning

Many network emulators can be considered for 5G networks (cf. section 2.A). Each of these emulators can be used for specific applications (e.g. wired/wireless SDN networks, data centers management). However, none of the existing environments support these different elements together. For example, an SDN-managed edge data center designed for wireless users using a 5G access network is not a scenario that could be considered with these emulators. That is why, in this paper, to overcome these limitations, we introduce a new emulation platform for 5G networks, integrating both Edge Computing, SDN/NFV, 5G access and core network ideas.

III. Emu5GNet: A New 5G Networks Emulator

In this section, we present a new emulator for 5G networks (Emu5GNet). We highlight A) the integration work that has been carried out, B) the resulting 5G architecture and C) the applications that could be imagined with Emu5GNet.

A. Integration of existing emulators/tools

To implement a complete 5G network, while taking into account the limitations of the existing emulation platforms (cf. Section 2.B), we considered the integration of different tools:
- Mininet-WiFi - Containernet: Containernet [9] is a fork of Mininet-WiFi using Docker containers as hosts. It allows the emulation of Wi-Fi networks (e.g. 802.11ac, 802.11p) managed by an SDN controller in a flexible environment (containers). This platform can be linked to the SUMO simulator [16], an open source software for microscopic traffic simulation (vehicles, trains, pedestrians, *etc*.) to implement complex scenarios;
- VIM-EMU: This platform [17] (SONATA project [10]) enables to locally prototype, deploy and evaluate network services. VIM-EMU, as Containernet, is based on Docker containers to enable quick and efficient deployment of services. This platform represents an efficient way to deploy and manage NFV functions and Edge servers using and deploying real orchestrators and services;
- UERANSIM: This tool [12] implements 5G User Equipment (UE) and 5G RAN (gNodeB) for both SA and NSA architectures. It represents an interesting building block for the implementation of a 5G communication architecture including the wireless segment. It is designed to support a large number of simultaneous communications and to evolve with the advances of 5G;
- Open5GS: This tool provides a C-language implementation for 5G Core [13]. It implements all 5G Core functions and can be used to deploy a complete 5G communication architecture. It is designed to be interconnected with 5G RAN platforms including UERANSIM.

An integration process was required to enable the implementation of Emu5GNet:
- VIM-EMU and Containernet - Mininet-WiFi compatibility: The VIM-EMU data centers are switches that have been modified to behave like edge servers (including CPU and storage models). Containernet was not designed to handle such nodes and was unable to recognize them. It was, therefore, necessary to integrate these two environments to enable the VIM-EMU data centers to be used in the Containernet - Mininet-WiFi platform;
- Open5GS and UERANSIM integration in Mininet-WiFi: These tools have not been designed to be integrated into a larger emulator. An integration and dockerization work of the 5G Core (Open5GS) and the 5G RAN (UERANSIM) was required to deploy end-to-end 5G communications in Mininet-WiFi - Containernet;
- VIM-EMU improvement: VIM-EMU is designed to deploy network functions and edge services in wired networks. The extension of VIM-EMU was necessary to allow the placement and migration of edge services for wireless 5G networks. This work involved adding new Application Programming Interfaces (APIs) to VIM-EMU and specifying new information about deployed services/functions: CPU, memory, *etc*.

B. Resulting architecture for Emu5GNet

Emu5GNet architecture (cf. Figure 1) is designed to deploy different types of nodes/elements:
- 5G UE and gNodeB: 5G UEs and 5G gNodeB can be instantiated in Emu5GNet using UERANSIM. These 5G UEs and gNB can be both mobile and fixed. UEs are implemented as specific Mininet-WiFi hosts and gNB as specific docker containers;
- 5G Core: A core 5G network can be deployed in Emu5GNEt using Open5GS. This 5G Core, implemented as a docker container, manages all the Emu5GNet UEs and gNodeB. In more complex scenarios the deployment of different network cores could easily be addressed;
- Wi-Fi Access Points and hosts: With Mininet-WiFi, Wi-Fi access points and Wi-Fi hosts can easily be deployed in

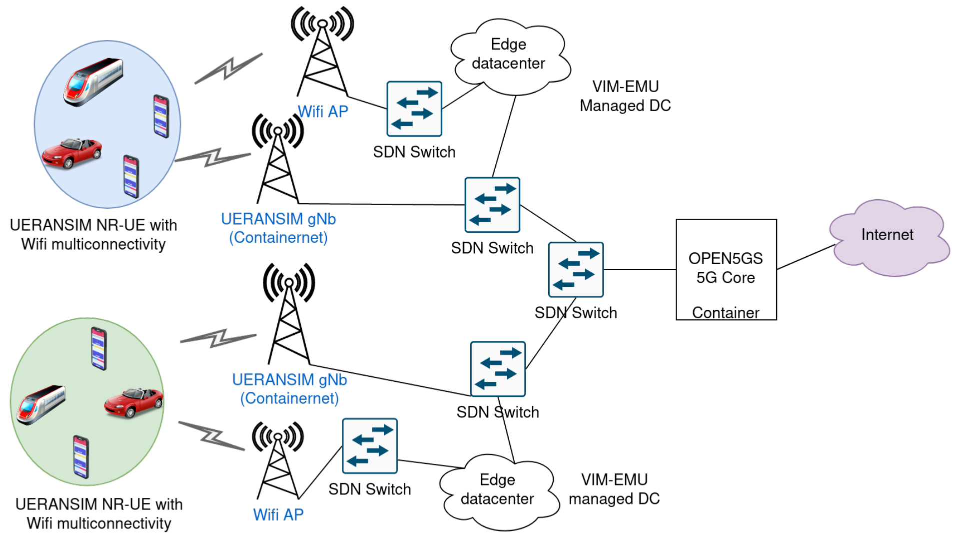

Fig. 1. Emu5GNet architecture

Emu5GNet. These nodes correspond to existing Mininet-WiFi - Containernet nodes and it was not necessary to modify them to enable their implementation;

- Edge Data Centers and Orchestrators: The deployment of edge data centers and orchestrators, as Docker containers, is possible in Emu5GNet using VIM-EMU. The edge orchestrator can manage the migration of edge services between the available servers (cf. section 3.A). All nodes (Wi-Fi, 5G NR) can connect to these servers.

The Emu5GNet architecture is managed by a central SDN controller. Wi-Fi hosts and 5G UEs can be integrated into a same Mininet host (multi-RATs device). Wi-Fi Access Points and gNBs can be connected to the same edge server. Moreover, all the nodes are connected to the Internet, increasing the number of deployable applications on these devices (e.g. streaming). Finally all devices can be mobile (SUMO) to implement enhanced scenarios.

C. *Potential applications using Emu5GNet*

Emu5GNET proposes a complete 5G architecture and can be used in multiple research projects:

- Research work on 5G communications performance: Emu5GNet includes both a 5G Core and 5GRAN. It could be used both for performance evaluation and to upgrade UERANSIM to include/emulate new features such as 5G handovers (currently being developed);
- Research work on multi-RATs architectures: Emu5GNet includes two main RATs, 5G and Wi-Fi. New RATs (e.g. 4G) could also be easily integrated. Multi-RATs connectivity is promoted in many contexts, in particular, for high bandwidth (flow aggregation) and high reliability (information duplication) scenarios. Emu5GNet could be used to propose new SDN-based solutions;
- Research work on SDN-based architectures: Emu5GNet enables the use of SDN for both wired and wireless network management. Numerous projects (e.g. load balancing, fault management) could be imagined using variable SDN controllers (e.g. OpenDaylight, Ryu) and mobile nodes (SUMO). Emu5GNet could therefore be useful for the implementation of new solutions for SDN-based 5G networks (e.g. architectures, protocols);
- Research work on edge data centers management: Emu5GNet allows the deployment of edge servers and the migration of VNFs/edge services between these servers. This realistic implementation is based on Docker containers and clear constraints (bandwidth, CPU, memory, *etc.*). Emu5GNet could be used to quickly implement and evaluate new proposals in this area.

These are just a few examples of potential applications of the platform defined in this paper. Studies related to Network Slicing, NFV or cybersecurity could also be imagined.

IV. Simple Use Case Using Emu5GNet

Emu5GNet was used to implement a simple scenario, based on SUMO. In this scenario, on a 3km2 map, we considered a variable number of cars (random trip) and a dozen of trains generating constant volumes of data (*iperf* command) and connected to 17 access points (5G + Wi-Fi 802.11n) uniformly distributed on the map. Three edge data centers are distributed on the map. Data generated by vehicles is automatically transmitted to the nearest data center. This scenario is presented in more details on the Emu5GNet Github Page [18].

This scenario corresponds to a basic use case of Emu5GNet: different RATs can be used by end devices to communicate with edge servers, with the Internet or with other devices. We compared, for these end devices, the level of performance of the two available RATs (802.11n and 5G NR). To do so, we assessed the maximum throughput allowed by each of these technologies and the associated latency (Round Trip

Time) in the context of a data transmission from the terminal equipments (trains, cars) to the nearest edge servers.

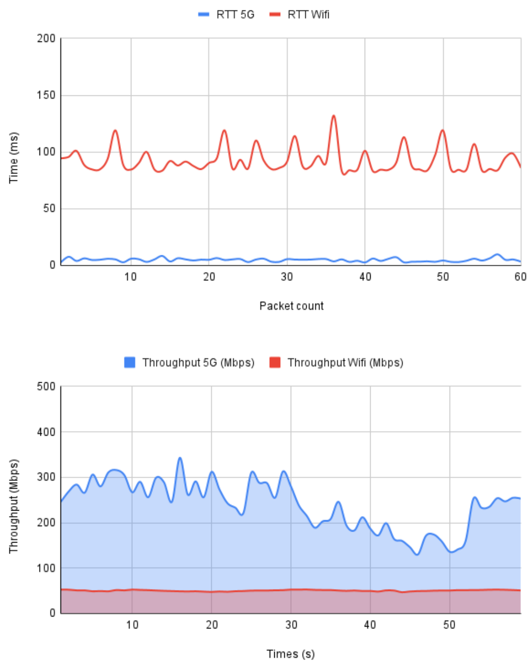

Fig. 2. Basic results obtained using Emu5GNet

The obtained results (cf. Figure 2) show that, both in terms of latency (RTT 10x lower on average) and maximum throughput (6 x higher on average), 5G NR technology performs better than 802.11n. This use case demonstrates that the proposed environment easily enables to simultaneously evaluate the performance of different RATs with simple commands (iperf). More complex applications could also be implemented: edges servers, NFV, *etc*. Using SUMO many use cases could also be designed and implemented. The simple deployment of edge servers and the possibility to create new APIs for the orchestrator could also provide an easy way to compare the performance of different edge service placement solutions. The potential for reuse of Emu5GNet could therefore be significant.

V. Conclusions

We introduced a new emulator for 5G networks: Emu5GNet. Emu5GNet implements the main elements of a 5G architecture: multi-access technologies (5G NR, Wi-Fi), flexible 5G core, SDN/NFV technologies, Edge Computing servers. It can already be used for the design, implementation and evaluation of new solutions related to 5G network (e.g. resource placement, wireless communication management). It has been shown in a simple use case, that results can quickly be obtained in this environment. As future work, we plan to further develop this platform by implementing more complex communication scenarios and providing new features.

VI. Acknowledgment

The work described has been carried out within the project "5G for future RAILway mobile communication system" (5GRAIL). This project has received funding from the European Union's Horizon 2020 research and innovation program, under grant agreement No 951725.e

References

[1] C. Blackman and S. Forge, "5g deployment: State of play in europe, usa and asia," 2019.
[2] L. Nadeem, M. A. Azam, Y. Amin, M. A. Al-Ghamdi, K. K. Chai, M. F. N. Khan, and M. A. Khan, "Integration of d2d, network slicing, and mec in 5g cellular networks: Survey and challenges," *IEEE Access*, vol. 9, pp. 37 590–37 612, 2021.
[3] A. Al-Ansi, A. M. Al-Ansi, A. Muthanna, I. A. Elgendy, and A. Koucheryavy, "Survey on intelligence edge computing in 6g: Characteristics, challenges, potential use cases, and market drivers," *Future Internet*, vol. 13, no. 5, p. 118, 2021.
[4] K. Kaur, J. Singh, and N. S. Ghumman, "Mininet as software defined networking testing platform," in *International conference on communication, computing & systems (ICCCS)*, 2014, pp. 139–42.
[5] S.-Y. Wang, C.-L. Chou, and C.-M. Yang, "Estinet openflow network simulator and emulator," *IEEE Communications Magazine*, vol. 51, no. 9, pp. 110–117, 2013.
[6] J. Yan and D. Jin, "Vt-mininet: Virtual-time-enabled mininet for scalable and accurate software-define network emulation," in *Proceedings of the 1st ACM SIGCOMM Symposium on Software Defined Networking Research*, 2015, pp. 1–7.
[7] R. R. Fontes, S. Afzal, S. H. Brito, M. A. Santos, and C. E. Rothenberg, "Mininet-wifi: Emulating software-defined wireless networks," in *2015 11th International Conference on Network and Service Management (CNSM)*. IEEE, 2015, pp. 384–389.
[8] M.-C. Chan, C. Chen, J.-X. Huang, T. Kuo, L.-H. Yen, and C.-C. Tseng, "Opennet: A simulator for software-defined wireless local area network," in *2014 IEEE Wireless Communications and Networking Conference (WCNC)*. IEEE, 2014, pp. 3332–3336.
[9] M. Peuster, J. Kampmeyer, and H. Karl, "Containernet 2.0: A rapid prototyping platform for hybrid service function chains," in *2018 4th IEEE Conference on Network Softwarization and Workshops (NetSoft)*. IEEE, 2018, pp. 335–337.
[10] S. Dräxler, H. Karl, M. Peuster, H. R. Kouchaksaraei, M. Bredel, J. Lessmann, T. Soenen, W. Tavernier, S. Mendel-Brin, and G. Xilouris, "Sonata: Service programming and orchestration for virtualized software networks," in *2017 IEEE international conference on communications workshops (ICC workshops)*. IEEE, 2017, pp. 973–978.
[11] N. Nikaein, M. K. Marina, S. Manickam, A. Dawson, R. Knopp, and C. Bonnet, "Openairinterface: A flexible platform for 5g research," *ACM SIGCOMM Computer Communication Review*, vol. 44, no. 5, pp. 33–38, 2014.
[12] K. Gökarslan, Y. S. Sandal, and T. Tugcu, "Towards a urllc-aware programmable data path with p4 for industrial 5g networks," in *2021 IEEE International Conference on Communications Workshops (ICC Workshops)*. IEEE, 2021, pp. 1–6.
[13] F. J. D. S. Neto, E. Amatucci, N. A. Nassif, and P. A. M. Farias, "Analysis for comparison of framework for 5g core implementation," in *2021 International Conference on Information Science and Communications Technologies (ICISCT)*. IEEE, 2021, pp. 1–5.
[14] S.-Y. Wang, "Comparison of sdn openflow network simulator and emulators: Estinet vs. mininet," in *2014 IEEE Symposium on Computers and Communications (ISCC)*. IEEE, 2014, pp. 1–6.
[15] G. Di Lena, A. Tomassilli, D. Saucez, F. Giroire, T. Turletti, and C. Lac, "Mininet on steroids: exploiting the cloud for mininet performance," in *2019 IEEE 8th International Conference on Cloud Networking (CloudNet)*. IEEE, 2019, pp. 1–3.
[16] D. Krajzewicz, "Traffic simulation with sumo–simulation of urban mobility," in *Fundamentals of traffic simulation*. Springer, 2010, pp. 269–293.
[17] U. Acar, R. F. Ustok, S. Keskin, D. Breitgand, and A. Weit, "Programming tools for rapid nfv-based media application development in 5g networks," in *2018 IEEE Conference on Network Function Virtualization and Software Defined Networks (NFV-SDN)*. IEEE, 2018, pp. 1–5.
[18] T. Sylla, "Emu5gnet," https://github.com/tidiosky/5grail-emu5gnet, 2022.