# On Evaluating Different Trends for Virtualized and SDN-ready Mobile Network

Imad ALAWE[1], Adlen KSENTINI[2], Yassine HADJADJ-AOUL[1], Philippe BERTIN[1], and Amélie KERBELLEC[1]

[1]name.surname@b-com.com, IRT b<>com, Rennes, France
[2]adlen.ksentini@eurecom.fr, EURECOM, Sophia-Antipolis, France

***Abstract*—Software Defined Networking (SDN) is one of the key enablers for evolving mobile network architecture towards 5G. SDN involves the separation of control and data plane functions, which leads in the context of 5G, to consider the separation of the control and data plane functions of the different gateways of the Evolved Packet Core (EPC), namely Serving and Packet data Gateways (S and P-GW). Indeed, the envisioned solutions propose to separate the S/P-GW into two entities: the S/P-GW-C, which integrates the control plane functions; and the S/P-GW-U that handles the User Equipment (UE) data plane traffic. There are two major approaches to create and update user plane forwarding rules for such a partition: (i) considering an SDN controller for the S/P-GW-C (SDNEPC) or (ii) using a direct specific interface to control the S/P-GW-U (enhancedEPC). In this paper, we evaluate, using a testbed, those two visions against the classical virtual EPC (vEPC), where all the elements of the EPC are virtualized. Besides evaluating the capacity of the vEPC to manage and scale to UE requests, we compare the performances of the solutions in terms of the time needed to create the user data plane. The obtained results allow drawing very interesting remarks, which may help to dimension the vEPC's components as well as to improve the S/P-GW-U management procedure.**

***Keywords*—SDN, vEPC, 5G, KPI, OpenAir, GTP, OpenFlow**

## I. Introduction

Upcoming 5G systems aim to not only increase the user throughput, but also bring into light a complete new mobile network architecture, which should be very flexible, dynamic, scalable and built on demand to fulfill the requirements of the new envisioned 5G services and applications. To clearly specify the expectations behind 5G systems, the 3GPP as well as the NGMN have produced some documents [1] [2] summarizing several envisioned services and use-cases. From these services and use-cases three categories emerge: (i) Enhanced Mobile Broadband; (ii) Massive Machine Type Communication (MTC); (iii) Ultra-low latency. Stemming from the fact that these three categories have different expectations in terms of network connectivity and Key Performance Indicators (KPI), the envisioned 5G architecture should be programmable and able to run these three categories by using (sharing) the same physical infrastructure.

978-1-5090-4026-1/17/$31.00©2017 IEEE

To fulfill heterogeneous services' requirements, while using the same infrastructure, network virtualization solutions, based on SDN, Network Function Virtualization (NFV) and Cloud Computing, are mandatory. In this context, different initiatives from industry and academia are proposing ways and methods to integrate NFV and SDN in the 4G mobile core. Most proposals converge toward a common commitment, which consists in separating control planes' and user planes' functions of the EPC elements (i.e., MME, and S/PGW). But, there is still a divergence in using or not an SDN controller in 5G. This divergence is explained in the following section, by indicating the standardization bodies and their respective strategies, and by clarifying the ins and outs of the different approaches.

The main contribution of this paper consists in evaluating the different visions, for the future 5G, using a testbed. The comparative analysis, reveal practical insights into the dimensioning of vEPC's components, and allow drawing very interesting remarks, which may help to improve the S/P-GW-U management procedure.

The remainder of this paper is organized as follows. Section II provides a classification of the existing architectures and lists the major differences between them. Section III is dedicated to introduce the deployed architectures for our benchmark. Section IV introduces the test scenarios and the test methodology. Section V is reserved to results analysis and discussions. Finally, Section VI concludes this paper.

## II. Related work

A high number of research work have been conducted to discuss and propose solutions to virtualize the EPC and enable the "softwarization" of the mobile core network, featuring SDN and NFV, as a first step

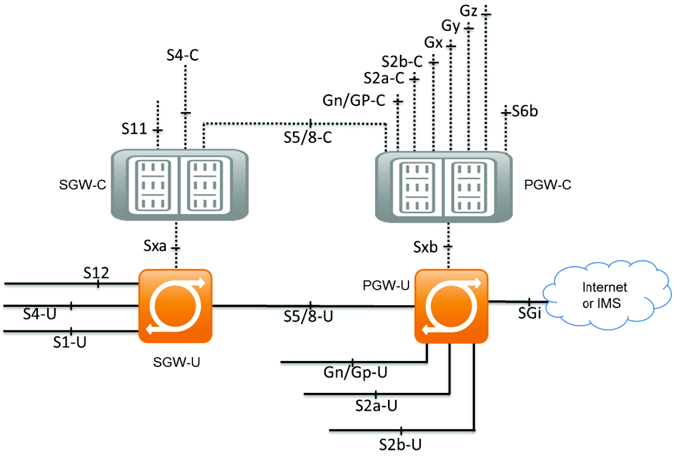


Fig. 1. 3GPP TR 23.714 Split Architecture.

towards 5G. Thus, three main approaches are found today: (i) vEPC; (ii) enhancedEPC; and (iii) SDNEPC.

The vEPC approach consists only on virtualizing the components of the 4G EPC (i.e. MME, SGW, PGW), and running them in virtual machines (VM) in the cloud i.e. [3].

The enhancedEPC is proposed by the 3GPP, via the CUPS group. They are developing a set of specifications and new interfaces in order to separate data plane and control plane functions. The new elements and interfaces that emerged from the function separation process are depicted in Figure 1. Besides splitting the SGW and the PGW, the CUPS architecture devises new interfaces: the Sxa interface connecting the SGW-C to the SGW-U and the Sxb interface connecting the PGW-C to the PGW-U. In addition, referring to Figure 1, we can notice that the interfaces S11 and S1U are kept to connect, respectively, the MME to the new SGW-C and the eNodeB to the SGW-U; while interface S5/8, which was used to connect the SGW to the PGW in the legacy EPC, is split into S5/8 C and S5/8 U to connect, respectively, the SGW-C to the P-GW-C and the S-GW-U to the P-GW-U. Last but not least, CUPS does not consider introducing an SDN controller in their architecture. Basically, the CUPS group is intending to use the GTP protocol with extensions or upgrades.

The SDNEPC approach is split into two major visions. The first one is called clean slate architecture. It unveils a complete novel and flat architecture composed by OF switches and SDN controllers. As an example, the solution presented in [4] withdraws GTP tunnels and proposes a full IP flat architecture. The latter is managed by SDN controller located close to the MME, while the S/P-GW are replaced by common OF switches. Although the proposed solution is novel, it contains some concerns in terms of scalability and QoS support. The second vision of SDNEPC relies on the current 4G architecture, keeping compatibility with the 3GPP specifications, to include SDN into the mobile core network. Most of the existing works, [5] [6] [7] [8] [9], belong to this category and mainly assume a separation of data and control planes' functions. The difference between these works consists in using or not virtual instances of the EPC entities (i.e. MME and S/P-GW), and in specifying which one should be split.

## III. Benchmark Architectures

Having reviewed the different initiatives, from industry and academia, to integrate SDN in mobile networks, we present in this section the deployed architectures for our benchmark.

### A. Virtual EPC (vEPC)

The first selected architecture (vEPC) as mentioned above represents the most classical one. The HSS, the MME, and the S/P-GW are each deployed on a separate VM and interconnected using the 3GPP reference interfaces (i.e. S1 and S11). The data plane is also following the 3GPP specifications using GTP bearers with the GTPv2 protocol. This architecture is based on OpenAir-CN project [3].

### B. SDN EPC

The second selected architecture (SDNEPC) is compatible with the 3GPP specifications, and aims to evolve the current 4G architecture towards integrating SDN. We omitted considering the "clean slate" solution (introduced in Section II), as the latter is unlikely to be adopted due to the weaknesses mentioned in section II. Thus, this architecture integrates an SDN controller based on OpenDayLight (ODL) [10] with a REST API that we developed to support 3GPP procedures. Figure 2 depicts the mobile network architecture featuring SDNEPC. Hence, it splits S/P-GW-C from S/P-GW-U. The S/P-GW-C is deployed on a VM and communicates with the ODL controller on northbound REST API through HTTP. Our programmable user plane relies on a modified version of Open Virtual Switch (OVS) [11] that we have patched (mainly to match GTP header) in order to perform as an S/P-GW-U. It provides an endpoint for the GTP tunnel created with the eNodeB. From there the traffic is routed using OF protocol, according to the flow rules pushed by ODL, following the request from the S/P-GW-C entity. In Figure 2, each SGW-U is connected to several eNBs using the S1-U interface. The interface between the SGW-C and the SDN controller is based on REST API. The MME is connected to the SGW-C via the S11 interface, and we assume that the SGW-C is connected to the PGW-C via S5/S8-C (specified in CUPS) interface. Note that one SGW-C (resp. PGW-C)

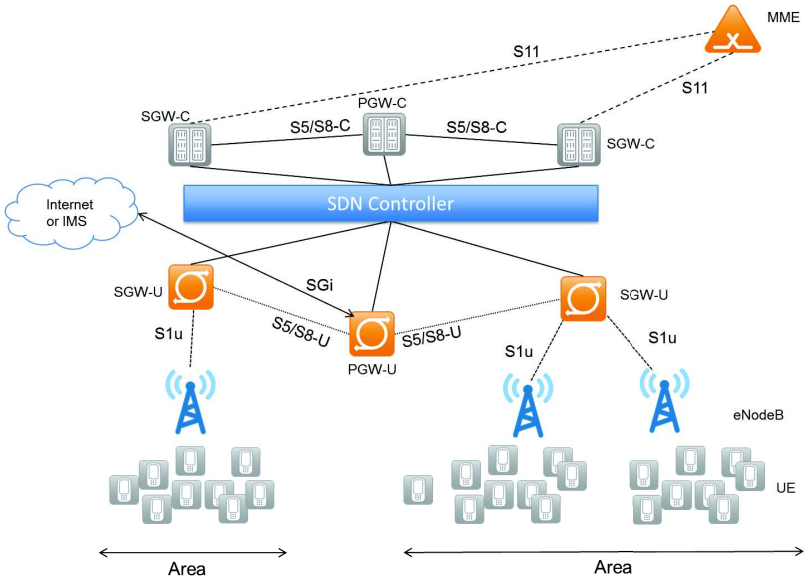


Fig. 2. EPC with SDN integration.

may manage one or more SGW-U (resp. PGW-U). For the sake of simplicity, our implementation merges the SGW and PGW in one element, and the latter is split into S/P-GW-C and S/P-GW-U to integrate SDN. Figure 3 illustrates a simplified example of creating a default bearer in case of using an SDN controller. All the messages in black are 3GPP standardized messages. The ones in blue are SDN specific messages. After the RACH procedure, the UE begins by requesting a Radio Resource Control (RRC) connection with the eNodeB. The latter triggers the creation of a NAS session for the connected UE. After checking the UE authorization, the MME triggers a request to the S/P-GW-C to create a session and to initiate the creation of a default bearer for the UE. The S/P-GW-C initializes the session context for the UE, and creates Bearer (i.e. tunnel ID), which will be communicated to the MME. The latter communicates the Tunnel ID and other information to the eNodeB, which in turn initializes a Tunnel ID and communicates the value to the S/P-GW-C via the MME. Once the eNodeB's tunnel ID is received, the S/P-GW-C sends an SDN UE Setup rule to the SDN controller in order to enforce the bearer at the concerned S/P-GW-U. The Setup rule is implemented using the REST API, which includes the tunnel IDs and the end-point tunnel IP addresses (i.e. the IP addresses of the eNodeB and the S/P-GW-U). The SDN controller translates the REST API message into OF rules to be sent to the S/P-GW-U. Two rules should be enforced, for the Uplink and the Downlink. The detach procedure and the dedicated bearer creation procedure share the same spirit, i.e. the data plane traffic is separately managed by the SDN elements and controlled by the S/P-GW-C via the SDN controller.

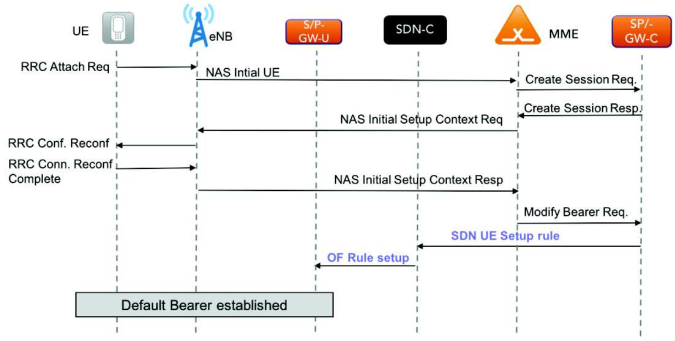


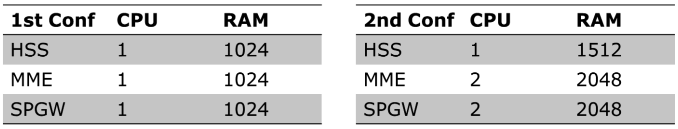
Fig. 3. UE Attach Call Flow with SDN integration.

### C. Enhanced EPC

Finally, the last selected architecture (enhancedeEPC) is merely similar to the previous one, except that no SDN controller is used. The S/P-GW-C directly pushes the flow rules using OVS commands via a direct interface. Through this architecture, we are able to mimic the 3GPP CUPS proposed architecture.

## IV. Test scenarios and methodology

As stated earlier, our test platform has been built with open source components so, all the obtained results are reproducible. To emulate the eNodeB and UEs, we used the emulation mode of OAI [12], known as OAISIM. Only physical channel is simulated. Protocol standard stacks from the Medium Access Control (MAC) to the Packet Data Convergence Protocol



| 1st Conf | CPU | RAM | 2nd Conf | CPU | RAM |
|---|---|---|---|---|---|
| HSS | 1 | 1024 | HSS | 1 | 1512 |
| MME | 1 | 1024 | MME | 2 | 2048 |
| SPGW | 1 | 1024 | SPGW | 2 | 2048 |

TABLE I
Configuration template of the EPC modules.

(PDCP) layers are following operations as real eNodeB. However, the UE part has been patched to enable the deployment of several UEs in parallel on the same machine (i.e. several instances of NAS layer). Therefore, we are able to generate (up to 900) simultaneous UE connection requests (NAS and RRC).

Since our first objective is to evaluate the performances of the virtualized instances of the EPC, we selected two different configurations of the VM; consisting in changing the amount of CPU and RAM dedicated to VMs running the MME and S/PGW. Table I details the amount of CPU and RAM allocated to a VM. The first configuration dedicates one CPU and 1 GB of RAM to the VMs hosting, separately, the HSS, MME and S/P-GW; while the second one doubles the resources dedicated to the VM, except for the HSS. Indeed, thanks to rehearsal tests, it was observed that the HSS element does not consume much CPU or RAM. Therefore, the VM running the HSS element will keep its 1 CPU for both configurations; but its RAM will be slightly increased in the second configuration. On the other hand, the CPU and the RAM of the VMs running the MME as well as the S/P-GW will be doubled, aiming at clearly demonstrating the effects of increasing/decreasing the CPU or the RAM on the performances. The S/P-GW-C, for the second and third architectures, uses the same configuration as the S/P-GW.

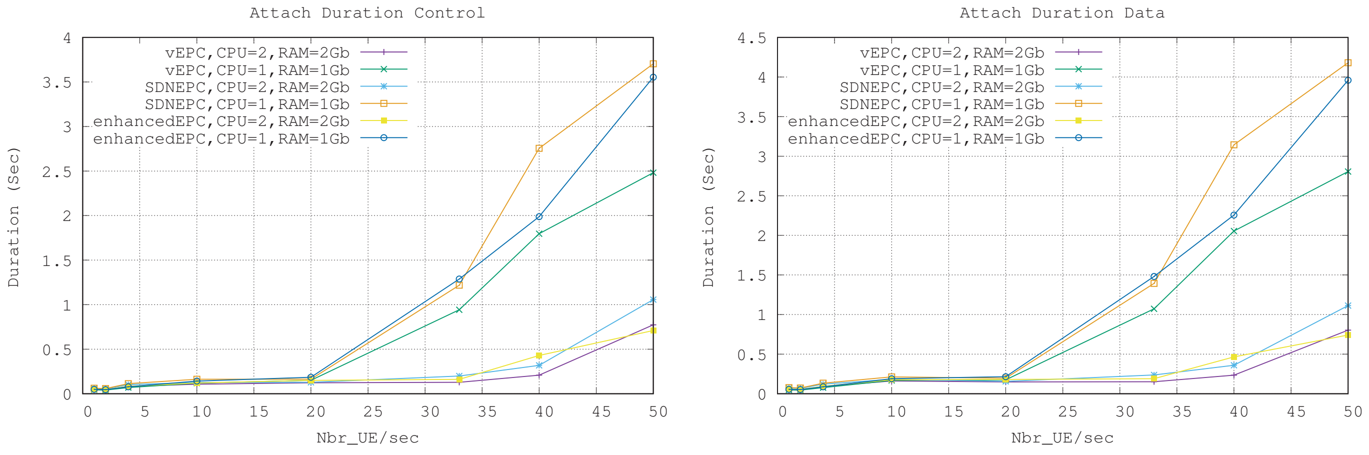
All the emulated UEs connect to the mobile network via the same emulated eNodeB. With this number of UEs (900 UEs), combined with the technical tests listed below, we will be able to clearly show the vEPC elements' performances and their capacity to manage UE requests. Each test consists in launching a predefined number of UEs per second, until reaching the maximal number of emulated UEs. Launching a UE consists in sending the RRC messages, followed by the NAS attach initialization. The realized scenario increases the number of attach per second from 1 UE to 50 UEs per second. The tests are stopped when reaching the maximum number of UE, i.e. 900 UEs. For each generated point, i.e. a number of UE per second and a specific vEPC configuration, the test is repeated 30 times to obtain accurate results.

Since our objectives is to evaluate the performance of the three architectures: vEPC, enhancedEPC, and SDNEPC; we defined four KPIs. The first one is the control plane attach duration, which represents the elapsed time between an attach request and an attach accept referring to the 3GPP standard. In other words, it is the time taken by the MME (and HSS) to identify, authenticate and register a UE in the network. The second KPI is the complete control and data plane attach duration. It is the time elapsed between an attach request and the modify bearer response of the S/P-GW in the vEPC architecture. For both enhancedEPC and SDNEPC versions, it is the time elapsed between an attach request and the moment that OVS creates the two OF rules in regard to UE data plane traffic (uplink and downlink). The third KPI indicates the ratio of treated UE requests during one second; i.e. the ratio of the number of UEs successfully attached by report to the number of attach requests received during one second. The last KPI gives the CPU consumption of the vEPC entities (mainly the VM hosting the MME). The first, third and fourth KPI are used to investigate the capacity of the vEPC to scale with the number of requests, and the response time of the vEPC to completely attach a UE. Besides indicating the capacity of the vEPC to establish the UE data plane, the second KPI allows to compare the performance of the three architectures. Indeed, one important concern of splitting the S/P-GW and integrating an SDN controller is the added latency to establish the UE data plane.

## V. Results And Discussions

### A. Results

Figure 4a illustrates the average attach duration of 900 UEs, over the control plane, by report to the number of attach request per second. This figure includes the performances of the three architectures featuring the two configurations. Based on the VM configuration, the performance increases proportionally to the CPU and RAM amounts. For the first configuration, i.e. only 1 CPU and 1 Gb of RAM, the vEPC handles very well the attachment procedures until 20 attach requests per second. Beyond this number, the average of the attach duration starts to increase. In addition, we notice that the second configuration pushes the limits of the vEPC to 40 attach requests per second. By comparing both configurations' results, we are able to conclude that doubling the CPU and the RAM size allows doubling its capacity to process more users' requests. Also, we remark that in both configurations, the three selected architectures have nearly the same performances. We argue this by the fact that the three architectures involve the same control plane entities (i.e. MME and HSS) for the attachment procedure. The difference involves only the data path construction, while the



(a)

(b)

Fig. 4. Attach duration over (a) control plane, (b) control and data plane.

MME and the HSS modules remain the same for the three architectures; i.e. the functions related to the attachment procedure.

Figure 5 indicates the ratio of accepted UE by report to the number of requests over the control plane. As in the previous figure, we observe that the three architectures have a similar performance. For both configurations, we remark the same thresholds, as in Figure 4a, from where the vEPC is not able to follow the pace of UE attach requests. Indeed, when the number of requests increases up to 20 (resp. 40) for the first (resp. second) configuration, the average attach duration increases indicating that some requests were not satisfied during the considered time interval of $1s$.

Figure 4b illustrates the average duration of the attach request over both the user and control planes versus the number of request per second. The figure includes the three architectures using the two configurations. First, we observe that, for the three architectures, the attach duration over the data plane is almost the same as the attach duration over the control plane; which means that the datapath is built on demand with insignificant latency.

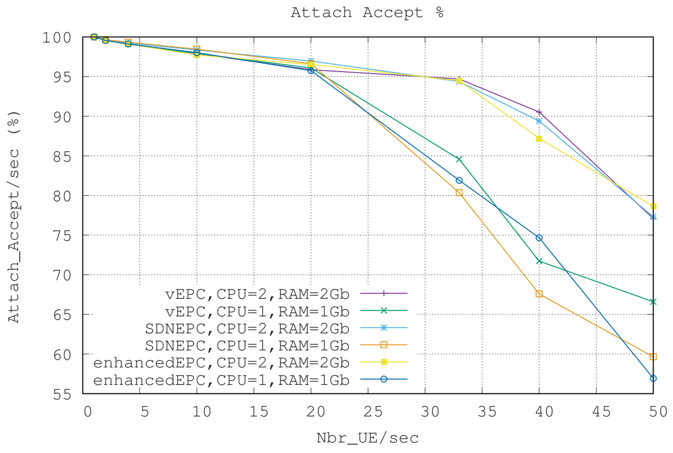
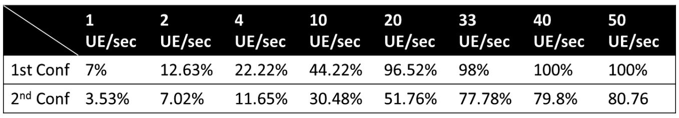


Fig. 5. UE accepted ratio over the C plane.

Second, we remark that the performance of the vEPC, the enhancedEPC and the SDNEPC version is slightly different. When the EPC is not congested; 20 UEs per second for the first configuration and 40 UEs per second for the second one; the difference between the vEPC and the enhancedEPC is about $1ms$. This is mainly due to the half of an RTT (one way communication) between S/P-GW-C and OVS (the RTT was around $2ms$). For the SDNEPC, the attach duration average is merely $2ms$ more than the vEPC. We argue this value by the RTTs of connections between the S/P-GW-C and the SDN controller, and between the SDN controller and S/P-GW-U.

In order to check the root reason of the performance degradation (as shown Figure 4a and 5), we draw in Table II the CPU usage of the VM (hosting the MME) for both configurations, focusing only on the case of vEPC. Indeed, as illustrated in Figures 4a and 5, the architectures merely achieve the same performances. From this table, we clearly observe that the points from



| | 1 UE/sec | 2 UE/sec | 4 UE/sec | 10 UE/sec | 20 UE/sec | 33 UE/sec | 40 UE/sec | 50 UE/sec |
|---|---|---|---|---|---|---|---|---|
| 1st Conf | 7% | 12.63% | 22.22% | 44.22% | 96.52% | 98% | 100% | 100% |
| 2nd Conf | 3.53% | 7.02% | 11.65% | 30.48% | 51.76% | 77.78% | 79.8% | 80.76 |

TABLE II
MME CPU usage over pattern tests.

where the attach duration are exponentially growing, i.e. 20 and 40 for configuration 1 and configuration 2, respectively, correspond to high CPU usage. Accordingly, we were able to saturate the VM capacity, leading to degrade the performance of the MME. This observation is very important for dimensioning the VM hosting the MME element; we have the choice to double the VM capacity or to deploy multiple instances of the MME.

### B. Discussion

From the results illustrated in Figure 4b, we clearly conclude that using an SDN controller doesn't introduce a consequent delay to establish the data plane for UEs. It worths mentioning that integrating SDN paradigm in mobile networks gives more flexibility for introducing new services especially with the programmability offered by NFV. Also, decoupling the control plane and the data plane ensures flexibility in system scalability. Thus, control plane can be re-dimensioned without touching the OVS and vice versa unlike classical S/P-GW which needs to be re-dimensioned completely even if one of the plane is not over-headed. Although this argues in favor of using SDN and NFV in 5G, still, there is a need for optimization techniques because we can clearly conclude that the OF protocol is not adapted for the mobile network use-case. Indeed, in traditional IP network, an OF rule may be used to forward the traffic for many services and many users at the same time. Unfortunately, in mobile networks, two flow rules are needed for each service on a UE. For example, to configure the data path of one service (i.e. one GTP bearer) used on the UE, it needs the installation of two flow rules: one flow rule for the uplink and one flow rule for the downlink. Each bearer needed by a user will obligatory lead to the installation of flow rules. This factor affects data plane by adding more latency in OVS, due to processing a huge table of flow rules, each time a packet needs to be routed. Finally, the obtained results open several possibilities to improve the deployment of virtualized EPC. Figure 4a allows to better dimension the vEPC elements, particularly the MME. Indeed, for both configurations, there is a point from where the attach duration grows exponentially, showing a clear degradation of the response time of the MME; which is mainly the results of the VM CPU saturation. Solutions could consist in scaling the system by either increasing the VM capacity or adding two new VMs (one contains the MME and another one a Load Balancer). Clearly, the choice between these two solutions has to further be investigated. Moreover, the results of Figure 4a could be used to dimension the vEPC before the deployment. To do that, we fit a polynomial function to our measurement results (Figures 4a). We found that the following expression reasonably describes our data (the coefficient of determination in our fit was $R2 = 0.9992$) and approximates the average attach duration by report to the number of UE requests per second as follows: $f(x) = 0.07 + 0.026x - 4.27x^4 + 3.96x^5$. Where $f(x)$ is the average attach duration and $x$ the number of requests per second. Thus, it is possible to predict the VM size according to a target average attach duration.

## VI. Conclusion

In this paper, we used a testbed platform (based on open source tools) to evaluate the performances of: (i) a virtual EPC (vEPC); (ii) an enhancedEPC where data and control plane are decoupled; (iii) an SDNEPC where an SDN controller is deployed. We were interested on metrics regarding the control plane and data plane performances. The obtained results allow us to draw very interesting conclusions towards dimensioning the virtualized version of EPC, in terms of CPU capacity, as well as the flexibility and the scalability brought by SDN and NFV in mobile networks.

## Acknowledgement

EURECOM contribution has been partially funded by the European Framework Program under H2020 grant agreement No. 723172 5G!Pagoda project

## References

[1] "3GPP." [Online]. Available: http://www.3gpp.org

[2] "NGMN, 5g white paper." [Online]. Available: https://www.ngmn.org/5g-white-paper.html

[3] "OpenAirCoreNetwork." [Online]. Available: https://gitlab.eurecom.fr/oai/openair-cn

[4] S. Shanmugalingam and P. Bertin, "Programmable mobile core network," in *ISCC*. IEEE, 2014, pp. 1–7.

[5] S. Ben Hadj Said, M. R. Sama, K. Guillouard, L. Suciu, G. Simon, X. Lagrange, and J.-M. Bonnin, "New control plane in 3gpp lte/epc architecture for on-demand connectivity service," in *CloudNet*. IEEE, 2013, pp. 205–209.

[6] X. An, W. Kiess, and D. Perez-Caparros, "Virtualization of cellular network epc gateways based on a scalable sdn architecture," in *GLOBECOM*. IEEE, 2014, pp. 2295–2301.

[7] J. Mueller, Y. Chen, B. Reichel, V. Vlad, and T. Magedanz, "Design and implementation of a carrier grade software defined telecommunication switch and controller," in *NOMS*. IEEE, 2014, pp. 1–7.

[8] A. Basta, W. Kellerer, M. Hoffmann, K. Hoffmann, and E.-D. Schmidt, "A virtual sdn-enabled lte epc architecture: a case study for s-/p-gateways functions," in *Future Networks and Services (SDN4FNS)*. IEEE, 2013, pp. 1–7.

[9] A. Ksentini, M. Bagaa, and T. Taleb, "On using sdn in 5g: the controller placement problem," in *Global Conference (Globecom)*. IEEE, 2016.

[10] "OpenDayLight." [Online]. Available: https://www.opendaylight.org/

[11] "OpenvSwitch." [Online]. Available: http://openvswitch.org/

[12] "OpenAirInterface5G." [Online]. Available: https://gitlab.eurecom.fr/oai/openairinterface5g