End-to-End Network Slicing Enabled Through Network Function Virtualization

Ibrahim Afolabi*, Miloud Bagaa*, Tarik Taleb* and Hannu Flinck§
* Aalto University, Email: firstname.lastname@aalto.fi and
§ Nokia Bell Labs, Espoo, Finland. Email: firstname.lastname@nokia-bell-labs.com

***Abstract*—Wireless networks have gone through several years of evolution until now and will continue to do so in order to cater for the varying needs of users. These demands are expected to grow in the future, both in size and variability. Hence, the 5G technology considers these variabilities in service demands and potential data explosion which could accompany users' demands at the core of its architecture. To enable 5G mobile handle these foreseen challenges, network slicing [24] is seen as a way forward as its standardization is progressing. In light of the proposed 5G network architecture base on network slicing, it is essential to be able to determine the correct virtual machine (VM) flavours in which to host the right type of network function based on the slice service requirements. In order to determine this, we carried out series of experiments involving the deployment of different VM flavours which may be suitable for different slices.**

I. INTRODUCTION

Several studies have revealed the numerous benefits of network resource sharing both on the capital and operational expenditures levels [7], [8], [13], [14], [17]. In particular, Radio Access Network (RAN) slicing which is an active form of network sharing is rapidly gaining momentum within telecommunications standardization bodies, so much so that, it is being touted that it would be an intrinsic part of the next fifth generation (5G) system architecture. For this to be possible, there are a number of essential conditions and requirements to be carefully considered and catered for. These requirements as mentioned in [2], [8] are Slice Isolation, Customization and Resources Utilization. To fulfill these requirements, wireless virtualization concepts such as Wireless Network Virtualization (WNV), Software Defined Networking (SDN) and Network Function Virtualization (NFV) are seen as possible candidates and key enablers to actualize this standard [3]–[5]. Network slicing is a mechanism that allows the sharing of a single network infrastructure between multiple network operators, whereby each operator provides its own unique functionalities and services to fulfill the needs of its users [6]. Indeed, a complete Evolved Packet System (EPS) architecture on a high level consists of three major parts, namely the User Equipment (UE), Radio Access Network (RAN) consisting of the Evolved Node B(s) (eNodeBs), and the core network (CN) part also called the Evolved Packet Core (EPC). In this context, we imagine an end-to-end (E2E) network slicing as the slicing of all these network parts.

Network slicing is the creation, instantiation and deployment of virtual instances of a complete mobile network from the access to the CN part, tailored towards a specific network service requirement or a set of similar business demands. In a nutshell, it involves the logical partitioning of an E2E mobile network, stitching together the necessary chain of virtual network functions from the access network to the CN in order to deliver a complete functional E2E mobile network deployable on virtual platforms. The virtual platforms could be VMs running on commodity servers locally or far away on cloud datacenters.

In order to actualize network slicing as a solution for the 5G system, a number of architectural challenges have to be considered, especially, when considering the fact that a mobile network is usually identified by its CN. This means that for multiple mobile network operators to share a commodity network infrastructure in the form of multiple network slices, each of the sharing tenants has to run on different virtual platforms hosted on the same hardware infrastructure while sharing the access network infrastructure. These virtual platforms will basically house the respective EPCs of the sharing partners. Bearing in mind that the virtual EPCs (vEPCs) do not necessarily have to be orchestrated from the same VM flavour, their respective service demands may determine their respective VM flavours as well as the size of the physical resource blocks to be assigned to them at the RAN. In particular, this paper focuses on determining the VM flavours and the right amount of resource blocks which is most suitable to host a vEPC considering the business requirements it has to accomplish, and develop an architecture for the orchestration of the resources.

Technologies such as WNV, SDN and NFV, would be leveraged for managing the E2E network slicing for reducing the Operational Expenditures (OPEX) while ensuring the Service Level Agreement (SLA) [17]. Indeed, the different VNFs would be placed in appropriate locations (cloud network or edge cloud) and the appropriate resources (CPU and memory) would be used to ensure the proper functionality of each E2E CN slice while reducing the overall cost. For enabling RAN slicing, we have updated the OpenAirInterface (OAI) RAN software to allow a single eNodeB to connect to multiple CNs in parallel. Two types of software-based CNs were deployed in our experimental setup: $i)$ OAI's CN and $ii)$ Aalto's CN. Slicing the RAN entails sharing the radio resources, ensuring that the traffic of the tenant EPCs are isolated from each other both on the control and data planes and manipulating the MAC

978-1-5386-3070-9/17/$31.00 ©2017 IEEE

scheduler of the RAN. The RAN scheduler is manipulated so that it allocates the right amount of physical resource blocks to the end users of the connected CNs.

The proposed framework, herein, mainly consists of EPCaaS orchestrator, SDN controllers (Opendaylight and/or ONOS) and NFV orchestrators. For each EPCaaS slice creation request, the EPCaaS orchestrator instructs the NFV orchestrators to create different VNF instances in different network clouds, and also instructs the SDN controllers to interconnect the created VNF instances. Besides the aforementioned contributions, we have also performed a set of benchmarking for matching between the features of E2E mobile network slices and their required resources in terms of CPU, memory, bandwidth and latency.

The rest of this paper is arranged in the following format. Network slicing background and related works are discussed in Section II. Section III introduces the proposed E2E mobile network slicing architecture. In Section IV, we present the used techniques for enabling E2E mobile network slicing platform. Finally, the paper concludes in Section V.

II. BACKGROUND AND RELATED WORKS

A. Network Slicing Enablers

In [5], the key enablers of network slicing are introduced and discussed in details. They are Wireless Network Virtualization, Software Defined Networking and Network Function Virtualization.

Wireless Network Virtualization uses five different means namely: Radio spectrum sharing, infrastructure sharing, Network slicing by service, user or application, abstraction layer definition which simplifies wireless access from heterogeneous network, and programmability and management of wireless networks to achieve network sharing and RAN slicing especially. Virtualization which happens to be a principal technology behind network slicing uses virtual networks to achieve wireless network virtualization.

Software Defined Networking is also an enabler as it separates the control plane from the data plane [9], [10]. Thereby dedicating a set of network modules to act as the controller which is known as the SDN controller. The separation between the control plane functionalities from the data forwarding functionalities brings the flexibility needed to achieve an almost perfect RAN slicing implementation. If SDN is carefully deployed to manage wireless network slices, it could turn out to be the necessary tool needed to ease the complexity that could accompany the management and programmability of wireless network slices.

Network Function Virtualization promotes the idea of removing network functions from dedicated physical network hardware equipments to run on any virtualization platform environment deployed in any location on the network. This could make it possible to decouple network functions running on proprietary network devices to run on decentralized and virtualized network servers which could be deployed at anytime in the network with respect to the network needs and service requirements [11], [12].

B. Related Work on Network Slicing

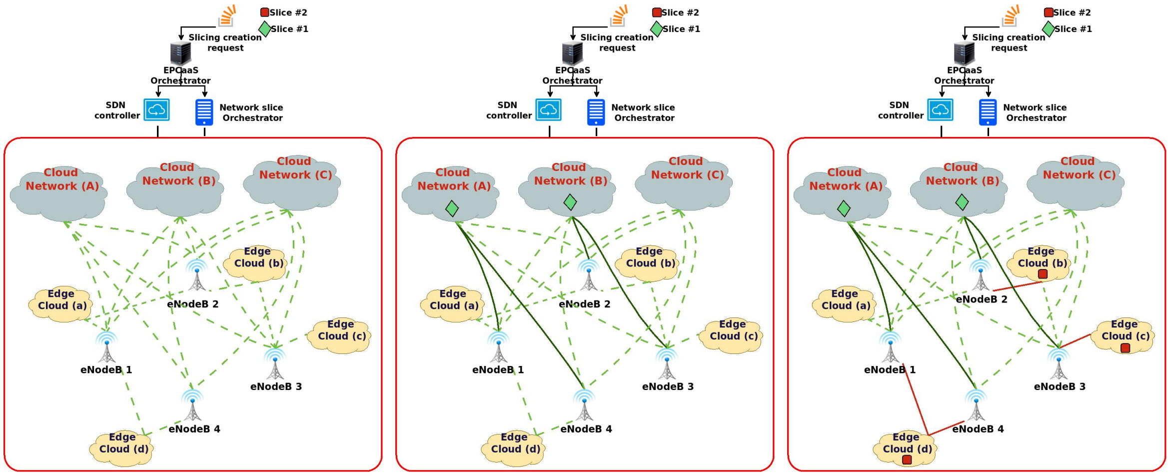
Already, there are a number of works that have proven the potential derivable advantages of RAN slicing amongst multiple tenants sharing network resources. Using the so called Multi-Operator Resource Allocation (MORA) criterion to mathematically prove the potential gains and cost saving benefits of utilizing a dynamic resource allocation amongst different mobile operators sharing a common RAN [13]. Another project [14] presented the idea of RAN Multi-tenant cell Slicing Controller (RMSC) which addresses these requirements using two different design approaches, a fully distributed and a fully centralized RMSC. Similarly, slicing solution based on spectrum sharing was described in [18] focusing more on sharing the RAN resources. It implemented a slice scheduler which could provide both Resource-based provisioning and Bandwidth-based provisioning slicing solutions.

A heuristic-based admission control mechanism was also developed in another work focusing on maximizing slice user's satisfaction based on RAN slice prioritization [19]. Also, there is the concept of RadioVisor [20] with a focus on the isolation of radio resources. The RadioVisor architecture includes three main components, the device and application to slice mapping, the radio resource allocation and finally the isolation function and slice manager. In [21], the Programmable RAN (PRAN) approach is proposed whereby the L1/L2 processing functionalities and scheduling tasks are deployed on commodity servers. In [22], the Virtual Prioritized Slice (VPS) examined the classification of slices into two major categories based on their delay tolerance levels. The slices are classified into Realtime (RT) and Non-Realtime traffic slices which is done for all the service providers at the network scheduler before allocating network resources to the slices with a focus on scheduling the physical resource blocks of the access network using a proportional fairness algorithm.

Despite the fact that these various related works have proven the potential gains of deploying EPS network components in virtual environments, they have done so with a focus on individual network parts only, and not considered a complete slicing of the entire network end to end. In contrast to these works, we are proposing a system which considers an end-to-end network slicing taking into consideration the entire EPS network components, called E2E network slicing.

III. MAIN OVERVIEW OF PROPOSED ARCHITECTURE

In this section, we present our suggested framework that will enable the management of EPC as a service (EPCaaS) platform across multi-domains clouds. This will be enabled through E2E network slicing which aims at virtualizing and slicing major network components which includes the RAN and CN resources. As mentioned earlier, enabling technologies such as NFV and SDN, will play a crucial role in enabling the E2E mobile network slicing. While NFV [1] technology will enable the elasticity and flexibility for creating different E2E mobile network slices across multiple domains, the SDN technology



(a) Network topology (b) Deployment of the first slice (c) Deployment of the second slice

Fig. 1. Proposed architecture overview

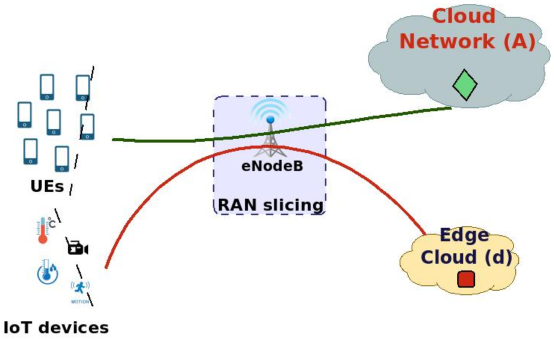


Fig. 2. E2E Slicing Scenarios

will enable the programmability of OpenFlow switches for ensuring the connectivity between different VNFs (eNodeB and CNs) of the same network slice.

Fig. 1 shows the main architecture overview of the proposed framework. For enabling E2E mobile network slicing in the suggested framework, a number of challenges are identified and discussed below. These challenges pertain to the two main parts involved in this slicing approach, the RAN and the Core part of the network.

1) Resource Allocation: In order for mobile network operators to maximize their profit, they tend to always keep the wireless frequency channels busy as much as possible. For this practice to continue and for network slice owners to make profit, the $5G$ RAN architecture needs to consider a seamless dynamic allocation of wireless resource blocks across slices [29] at the heart of the RAN MAC scheduler. As for the EPCaaS orchestrator which will be managing and controlling slice resource allocation in the CN, it is important to design one which will ensure fairness in resource allocation to the requesting network users [15], [16].

2) Isolation: Isolation is a process of ensuring that resources allocated to a particular network slice do not affect another or that the quality of resources allocated to a slice remains the same over an agreed duration of time [5]. This is particularly difficult to achieve due to the variability of radio frequency channels over a duration of time. In addition, slice isolation in wireless networks is especially challenging due to the mobility of end users getting services from different slices [2]. Moreover, the EPCaaS orchestrator should have updating mechanism which always keeps it up to date regarding records of allocated resources and the available allocatable network resources, so that, the quality of the already orchestrated resources will not be jeopardized in a bid to allocate new network resources to incoming request(s). It must ensure that slices do not experience both inter and intra slice interferences [19].

3) Customization: Flows belonging to different slices should be customizable for example, depending on the quality of service offered. Slices should be able to determine the flow quality of services independent of another slice [2]. There should be a programmable interface provided by the network virtualization solution to which will enable customization of flows attributes. The level of customization which could be offered is solely dependent on the allowable flexibility available through the deployed virtualization solutions and the service level agreement existing between the infrastructure provider and the network operators sharing the network resources [18].

As depicted in Fig. 1, the envisioned EPCaaS orchestrator should offer a RESTful API that allows the slice administrator to specify E2E mobile network slices and their features, such as network latency and link bandwidth. Moreover, the user can specify different management rules and policies for the instantiation and auto-scaling of different VNF instances created in variant cloud networks. According to the received requests, the EPCaaS orchestrator enforces the rules for a specified slice by communicating them to the virtual infrastructure manager (VIM). In this figure, dashed arrows between EnodeBs, edge clouds and CNs, indicate the network connectivity between them. The length of a dashed arrow is proportional to the distance between the network entities (i.e., eNodeB, Edge cloud and CN) it is connecting. The longer the distance between an eNodeB to a CN is, the higher the latency and the lower the bandwidth becomes. This figure shows the creation of two E2E mobile network slices. While Fig. $1(b)$ shows the deployment of the first E2E mobile network slice that will be used for connecting UE variants, Fig. $1(c)$ shows the deployment of the second slice used for the purpose of connecting IoT devices. As the first slice does not have any special requirements, the different E2E CN VNFs are instantiated in different CNs without any restrictions. Meanwhile, the second slice requires low latency and high bandwidth, for this reason the variant CN VNFs are instantiated close to eNodeBs at the edge clouds. Fig. 2 shows the RAN slicing for connecting the two E2E CN slices dedicated to UEs and IoT devices, respectively.

IV. TOWARDS ENABLING E2E NETWORK SLICING PLATFORM

As a means to enable our proposed architecture, we use the open source project of the OpenAirInterface Software Alliance (OSA) popularly called the OpenAirInterface (OAI). This project is developed and sponsored with the aim of softwarizing mobile network functions from the access network to the evolved packet core of the mobile network. It is a project which is aggressively supported and developed by an agile community of professional software developers both in academia and telecommunication industry. The OAI project is especially centered around virtualizing the virtualizable part of the access network and the entire EPC of the mobile network, so that these entities can respectively be deployed on virtual platforms. The OAI's holistic virtual mobile network solution is implemented in order to support the role out of the 5G technology enabled through network slicing and further reduce the cost of deploying mobile networks. The OAI's project is broadly divided into two parts, namely, the Openairinterface5g and the Openair-cn.

The Openairinterface5g is the project which is developed to carry out the Radio Access Network (RAN) functionalities of a mobile network and also includes the implementation of a software-based user equipment, while the Openair-cn is responsible for the set of functions running in all the major components of a complete mobile CN i.e. the Home Subscriber Server (HSS), Mobility Management Entity (MME) and the combination of both the Serving and Packet Data Network Gateways (SP-GW) following the 3GPP cellular network standards [25]. Similarly and in preparation towards the 5G technology support for network slicing, Aalto University also developed its own CN which in addition to following the 3GPP standards, also incorporates and integrates the concept of the SDN technology in its development. This is in a bid to have a better support for virtual deployment of the CNs and for easier backhauling of user packets from the eNodeB. Both the OAI's and Aalto university's mobile network solutions have been tested and are working, though the former has a larger community of developers than the latter, mainly due to the fact it is offered as an open source.

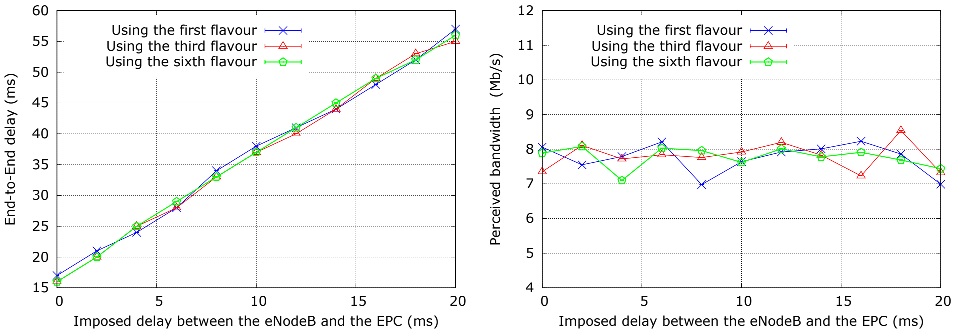
A. Enabling RAN Resources Slicing

Collectively, both the software-based mobile network solutions of OAI and Aalto University together provide a rich source of experimental platform for the deployment of virtual mobile networks as a means to enable network slicing. We successfully deployed and tested both the cloud RAN (C-RAN) network and the virtual EPC of the OAI solutions using a Commercial of The Shelf (COTS) UE to connect to the Internet. Similarly, using the OAI's C-RAN, we also successfully tested the Aalto's developed virtual EPC solution using a COTS UE. However, in a bid to test both virtual EPCs side by side while running in parallel and implement a RAN network slicing solution over the OAI's C-RAN, we developed the S1-flex interface on top of the C-RAN solution as specified by 3GPP [26]. The S1-flex configuration allows an eNodeB to connect to multiple EPCs in a pool area of EPCs [27]. The S1-flex configuration allows a complete separation of the connected pool of EPCs on both the control and data planes. This functionality was successfully implemented on top of the OAI's C-RAN and tests were conducted using both Aalto's and OAI's virtual EPCs by connecting different COTS UEs to different Packet Date Network (PDN) networks through the respective vEPCs. By so doing, we successfully sliced the resources of the RAN between two vEPCs.

Bearing in mind that the main functions of an access network is broadly divided into two, namely, the baseband functions and the radio frequency functionalities [28]. In our deployment, while the OAI RAN solution is deployable on virtual platforms in order to carryout the baseband functions, the radio frequency functionalities are accomplished using the software defined radio solution offered by National Instrument's USRP B210 embedded board. However, for better performance of the OAI's virtual RAN solution especially on resource constrained VMs, it is recommended to deploy the RAN solution directly on a bare metal PC due to real-time processing requirements of the RAN solution.

B. Benchmarking Tests for Enabling Smart Algorithms Deployment

Generally, network slicing involves the deployment of the entire mobile network functions and components to run on virtual platforms i.e. VMs running on commodity servers. However, in order to determine the most suitable flavour of VM on which to enable the deployment of different variants of network slices, there is need to carryout some tests involving both the EPCs, eNodeB and COTS UEs. In our tests which was conducted in an office environment, we varied different parameters especially those responsible for the quality of network links between the eNodeB and EPC VMs respectively. These parameters are the E2E latency and available bandwidth between the eNodeB's and the EPC's machines. By varying these link properties for different flavours of VMs which is determined by the number of allocated CPUs and available RAMs, for a single connected UE, the perceived bandwidths and latencies by the UE were recorded.



(a) End-to-End delay

(b) Perceived bandwidth

Fig. 3. Performance evaluation of EPS as a function of the imposed bandwidth between the eNodeB and the EPC.

We deployed a total of six virtual EPC flavours ranging from flavour F1, which has 2 CPUs and 2GB RAM to F6, with 12 CPUs and 12GB RAM all running on 20GB storage space. The intermediate flavours have an increment of 2 units each for the number of CPUs and RAM memories. Similarly, the link's bandwidth was varied from the highest bandwidth of approximately 950Mb/s to a lowest of about 100Mb/s between the eNodeB and vEPC nodes. In addition to that, we emulated the amount of E2E delay that the link between the eNodeB and EPC could possibly experience when the location of different flavours of the EPC VM is varied and recorded the E2E link delay perceived by the UE. Both parameters (i.e. delay and bandwidth) were varied using the Linux Traffic Control (tc) command. The resultant E2E bandwidths and latency was then measured from the UE using the Iperf network bandwidth measurement tool and by pinging a remote server on the Internet respectively.

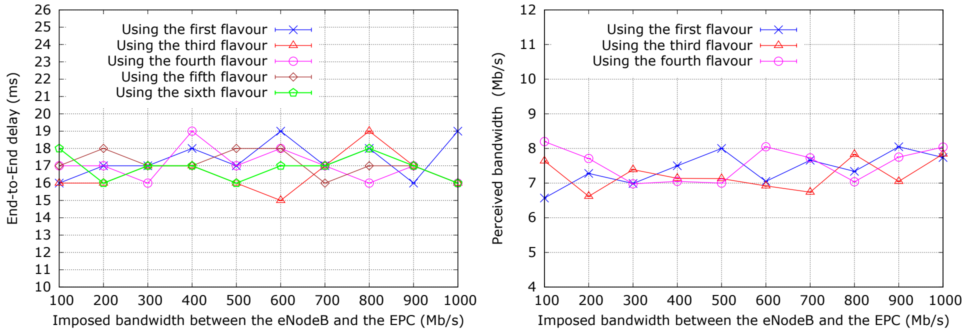
As mentioned above our experimentations are targeted towards evaluating broadly two major factors affecting a network link, the network link's throughput or bandwidth and the link's latency or delay. These evaluations are made based on the perceived throughputs and latencies by the connected COTS UE, which in our case is an LTE dongle. Our system setup consists of two laptops, a USRP B210 kit with a duplexer and 6db antenna. One of the laptops together with the LTE dongle is used as our UE, while the other laptop with 4 CPUs and 12GB RAM hosts the eNodeB. The EPC flavours were deployed on VMs running on remote servers. The USRP B210 functions as the Remote Radio Head (RRH) for the eNodeB which is running on our second laptop and connected to it via a USB3 port for fast I/O processing of control signals and user data respectively. The LTE capable UE attaches to the EPC through the eNodeB using user profile information available on programmable SIM cards.

Based on the E2E delays and bandwidths perceived by the UE, our evaluation results are divided into two major parts depicted by Figures 3 and 4 respectively. In Figure 3, three variants of the EPC VMs were deployed one after the other. During each deployment, additional system delay was imposed on the network link between them and the eNodeB. The imposed system delay varies between 1 to 20 milliseconds. As a result of these additional system delays, the UE perceived E2E delays between 16 to approximately 57 milliseconds as shown in Figure 3(a). Likewise in Figure 3(b), the E2E bandwidth perceived by the UE while connected to the Internet through the EPC VMs varies between approximately 7 to about 8.5Mb/s for all the deployed EPC flavours. Similarly in Figure 4, we constrained the link bandwidth between the eNodeB and EPC machines in order to understand its resultant effect on the available bandwidth perceived by the UE as well as the system delay, if there would be any variations. Figure 4(a) and (b) depict the results of the E2E delay and bandwidth perceived by the UE respectively. It is observed that the delays and bandwidth experienced by the UE is an average of 17ms and 7.4Mb/s respectively for all the tested EPC VM flavours.

V. CONCLUSION

This paper discussed the concept of E2E network slicing as an important vision of 5G mobile systems, highlighting its key enabling technologies. RAN slicing is an integral part of this concept. This paper provided a high level description of the relevant architecture and showcased how E2E mobile network slicing can be achieved leveraging the OAI source code and the AALTO CN solution. The E2E mobile network slicing depends principally on the S1-Flex concept, as per 3GPP standards, and the dynamic sharing of RAN resources.

For an efficient E2E network slicing architecture, a number of challenges are yet to be tackled. This includes determining the optimal amount of physical resource blocks to be assigned to a slice type in the RAN and the total resources (CPU and memory) to be allocated for the orchestration of the EPC VM using an EPCaaS orchestrator as proposed in our architecture. Whilst the introduced framework represents a simplistic, yet important step towards a practical implementation of the E2E network slicing concept, it is the authors' hope that the presented work would stimulate further research activities from the relevant community of researchers in developing efficient algorithms and mechanisms that would support such concept in line with the envisioned 5G system architecture.



(a) End-to-End delay

(b) Perceived bandwidth

Fig. 4. Performance evaluation of EPS as a function of the imposed delay between the eNodeB and the EPC.

ACKNOWLEDGMENT

This work was supported by the European Union's Horizon 2020 research and innovation programme under the 5G!Pagoda project with grant agreement No. 723172.

REFERENCES

[1] ETSI GS NFV 002, Network Functions Virtualization (NFV); Architectural Framework v1.1.1, 2013

[2] T. Taleb, B. Mada, M. Corici, A. Nakao, and H. Flinck, "PERMIT: Network Slicing for Personalized 5G Mobile Telecommunications," in IEEE Communications Magazine, Vol. 55, No. 5, May 2017, pp. 88 93

[3] T. Taleb, M. Corici, C. Parada, A. Jamakovic, S. Ruffino, G. Karagiannis, and T. Magedanz, "EASE: EPC as a Service to Ease Mobile Core Network," in IEEE Network Magazine, Vol. 29, No. 2, Mar. 2015. pp.78 88.

[4] T. Taleb, "Towards Carrier Cloud: Potential, Challenges, & Solutions," in IEEE Wireless Communications Magazine, Vol. 21, No. 3, Jun. 2014. pp. 80-91.

[5] M. Richart, J. Baliosian, J. Serrat and J. Gorricho, "Resource Slicing in Virtual Wireless Networks: A Survey", IEEE Transactions on Network and Service Management, pp. 1-1, 2016.

[6] P. Rost, A. Banchs, I. Berberana, M. Breitbach, M. Doll, H. Droste, C. Mannweiler, M. Puente, K. Samdanis and B. Sayadi, "Mobile network architecture evolution toward 5G", IEEE Communications Magazine, vol. 54, no. 5, pp. 84-91, 2016.

[7] T. Taleb, K. Samdanis, B. Mada, H. Flinck, S. Dutta, D. Sabella, "On Multi-Access Edge Computing: A Survey of the Emerging 5G Network Edge Architecture & Orchestration," IEEE Communications Surveys & Tutorials, (in press).

[8] I. Afolabi, A. Ksentini, M. Bagaa, T. Taleb, M. Corici and A. Nakao, "Towards 5G Network Slicing over Multiple-Domains", IEICE Transactions on Communications, 2017.

[9] A. Aissioui, A. Ksentini, A. Gueroui, and T. Taleb, "Toward Elastic Distributed SDN/NFV Controller for 5G Mobile Cloud Management Systems," in IEEE Access Magazine, DOI 10.1109/ACCESS.2015.2489930, Vol. 3, Nov. 2015.

[10] A. Ksentini, M. Bagaa, and T. Taleb, "On using SDN in 5G: the controller placement problem," in Proc. IEEE Globecom 2016, Washington, USA, Dec. 2016.

[11] T. Taleb, A. Ksentini, and R. Jantti, "Anything as a Service for 5G Mobile Systems", in IEEE Network Magazine, Vol. 30, No. 6, Dec. 2016.

[12] T. Taleb, K. Samdanis, and A. Ksentini, "Supporting Highly Mobile Users in Cost-Effective Decentralized Mobile Operator Networks," in IEEE Trans. on Vehicular Technology, Vol. 63, No. 7, Sep. 2014, pp. 3381-3396.

[13] P. Caballero, A. Banchs, G. de Veciana and X. Costa-Perez, "Multi-Tenant Radio Access Network Slicing: Statistical Multiplexing of Spatial Loads", arXiv Networking and Internet Architecture,07,2016.

[14] P. Garces, X. Perez, K. Samdanis and A. Banchs, "RMSC: A Cell Slicing Controller for Virtualized Multi-Tenant Mobile Networks", in proc. IEEE VTC'81, Glasgow, Scotland, May, 2015.

[15] F.Z. Yousaf and T. Taleb, "Fine Granular Resource-Aware Virtual Network Function Management for 5G Carrier Cloud," IEEE Network Magazine, Vol. 30, No. 2, Mar. 2016. pp. 110 115.

[16] F.Z. Yousaf, P. Loreiro, F. Zdarsky, T. Taleb, and M. Leibsch, "Cost Analysis of initial deployment strategies of a Virtual Network Infrastructure in a Datacenter", in IEEE Communications Magazine, Vol. 53, No. 12, Dec. 2015, pp. 60 - 66.

[17] A. Nakao, P. Du, Y. Kiriha, F. Granelli, A. A. Gebremariam, T. Taleb, and M. Bagaa, "End-to-End Network Slicing for 5G Mobile Networks", in J. Information Processing, Vol. 25, No. 1, Jan. 2017

[18] X. Costa-Perez, J. Swetina, T. Guo, R. Mahindra and S. Rangarajan, "Radio access network virtualization for future mobile carrier networks", IEEE Communications Magazine, vol. 51, no. 7, pp. 27-35, 2013.

[19] M. Jiang, M. Condolusi and T. Mahmoodi, "Network slicing management and prioritization in 5G mobile systems", European wireless 2016; 22 nd European wireless conference, 2016.

[20] A. Gudipati, L. Li and S. Katti, "RadioVisor: A Slicing Plane for Radio Access Networks", in ACM HotSdn'14, Proceedings of the Third Workshop on Hot Topics in Software Defined Networking, page 237-238. Chicago, IL, USA, August, 2014.

[21] W. Wu, L. Li, A. Panda and S. Shenker, "PRAN: Programmable Radio Access Networks", in ACM HotNets'13, Proceedings of the 13th ACM Workshop on Hot Topics in Networks, Los Angeles, CA, USA, October, 2014.

[22] A. Abdelhamid, P. Krishnamurthy and D. Tipper, "Resource Allocation for Heterogeneous Traffic in LTE Virtual Networks," 2015 16th IEEE International Conference on Mobile Data Management, Pittsburgh, PA, June, 2015, pp. 173-178.

[23] A. Bhamri, N. Nikaein, F. Kaltenberger, J. Hamalainen and R. Knopp, "Pre-processor for MAC-layer scheduler to efficiently manage buffer in modern wireless networks", in proc. IEEE WCNC, Istanbul, Turkey, April, 2014.

[24] "Study on Architecture for Next Generation System", the 3rd partnership project (3GPP), TR 23.799, version 14.0.0, Dec. 2016.

[25] "OpenAirInterface — 5G software alliance for democratising wireless innovation", Openairinterface.org, 2017. [Online]. Available: http://www.openairinterface.org/. [Accessed: 11- May- 2017].

[26] "LTE; Evolved Universal Terrestrial Radio Access Network (E-UTRAN); Architecture description", 3GPP TS 36.401 version 9.0.0 Release 9, Feb. 2010.

[27] "General Packet Radio Service (GPRS) enhancements for Evolved Universal Terrestrial Radio Access Network (E-UTRAN) access", 3GPP TS 23.401, Rel.13, Sep. 2015.

[28] A. Checko, H. Christiansen, Y. Yan, L. Scolari, G. Kardaras, M. Berger and L. Dittmann, "Cloud RAN for Mobile Networks; A Technology Overview", IEEE Communications Surveys & Tutorials, vol. 17, no. 1, pp. 405-426, First quarter, 2015.

[29] L. Qian, W. Geng, P. Apostolos, M. Udayan, "An end-to-end network slicing framework for 5G wireless communication systems", August, 2016.