5G Network Slicing Enabling Edge Services

Michail-Alexandros Kourtis[1], Themis Anagnostopoulos[2], Sławomir Kukliński[3], Michał Wierzbicki[3],
Andreas Oikonomakis[2], George Xilouris[1], Ioannis P. Chochliouros[4], Na Yi[5], Alexandros Kostopoulos[4],
Lechosław Tomaszewski[3], Thanos Sarlas[2], Harilaos Koumaras[2]
[1]ORION Innovations Private Company, Athens, Greece
[2]National Center of Scientific Research "Demokritos", Athens, Greece
[3]Orange Polska, Warsaw, Poland
[4]Hellenic Telecommunications Organization S.A. (OTE), Research Programs Section, Athens, Greece
[5]University of Surrey, Guildford, United Kingdom

***Abstract*—Network slicing already plays an important role as a critical enabler in the current 5G technology domain. 5G aims to disrupt and accelerate innovation in various vertical fields, among those is the vehicular industry. In the detailed scope of the 5G-DRIVE research project promoting cooperation between the EU and China, a set of trials is to be undertaken towards promoting 5G growth. In this paper, initially, we identify a variety of challenges arising from the 5G convergence to the automotive industry. Then we describe the specific innovative framework of the 5G-DRIVE research, together with a novel network slicing mechanism deployed at the edge. Then an analysis on the corresponding architectures is presented, and how they operate in a set of trials for new 5G services. The services described are a virtualized caching network function (vCache), and a deep packet inspection one (vDPI), which are deployed at the edge facilitating an edge 5G service. For each case, the services are deployed and evaluated in the 5G Drive platform using the Katana slicing framework. Additional analysis of the OSM slicing platform is presented. The results demonstrate the performance of a network slicing mechanism for 5G service deployments in an edge enabled platform.**

***Keywords*—Network Function Virtualization, Edge Cloud Computing, 5G, Network Slicing, OSM, V2X.**

I. INTRODUCTION

The rapid proliferation of mobile technologies and communications has driven their inclusion in our daily life, across various sectors, "as a whole" [1]. It is expected to increase their essential role in various upcoming modern services and facilities [2]. Similarly, mobile devices have become an indispensable accessory in our daily activities. Future IMT (International Mobile Telecommunications) systems and more specifically the 5G initiative plan to support emerging new vertical use cases, including applications that require very high data rate communications, a large number of connected devices, and ultra-low latency or high-reliability applications.

Within the forthcoming years, it is also expected that the underlying (usually heterogeneous) network infrastructure will be able of "connecting everything" according to an extended multiplicity of application-specific requirements (thus including users, things, goods, computing centres, content, knowledge, information and processes), in a purely flexible, mobile, and quite powerful way [3].

In this scope, 5G has identified three-pillar requirements for future networks: namely enhanced mobile broadband (eMBB), massive machine-type communications (mMTCs), and ultra-reliable low-latency communications (URLLCs). These requirements are vital for the support of upcoming user services that will demand significantly higher data rates and simultaneous multi-connectivity. However, the management and allocation of resources in a network level is still an open issue. The transition beyond 5G telecommunication networks brings more challenges concerning the provision, placement, migration, spectrum management and dynamic resource allocation [4].

The paradigm of network slicing has been introduced in order to address differentiated resource allocation requirements and bring service provision a step closer, especially for high demand vertical industries [5]. Network slicing will play a pivotal role in addressing varied vertical applications by enabling dedicated virtualized network slices for each vertical. This is due to the envisioned novel types of interactions and applications that the future smart connectivity ecosystem will bring [6]-[8].

In this paper, the 5G-DRIVE [9] approach is introduced, as a 5G enabled platform for vertical vehicular technologies. 5G-DRIVE, among other topics, focuses on the integration of virtualized network services running at the edge, that facilitate vehicular end-users and their respective systems. Additionally, considering critical slicing requirements of such an infrastructure, this paper presents a novel slicing framework, Katana, developed in the scope of the 5GENESIS project [10]. In order to evaluate the overall platform and the Katana slicing integration a set of prototypes VNFs (vCache, vDPI), developed in the frame of 5G-DRIVE, were deployed and their performance was evaluated in terms of deployment time and relative performance.

Consequently, and driven by the open challenges addressed, Section II presents in detail the scope of the ongoing 5G-DRIVE project, which aims to establish a collaborative framework for trials the EU and China and promote 5G penetration. In Section III a series of experimental results concerning OSM slicing and the Katana slice manager is presented, in order to assess current slicing solutions. Section IV presents the developed VNFs, and results of their evaluation in terms of performance in a 5G environment. Finally. Section V summarizes the presented work and outlines future research directions.

II. 5G DRIVE ARCHITECTURE

A. Context and Aims

Following by the previously described challenges towards a fast and reliable 5G evolution, the 5G-DRIVE project [9] is part of the H2020 ICT-22-2018 Call ("EU China 5G Collaboration"), which aims to perform a close collaboration between the European Union (EU) and China to synchronize 5G technologies and spectrum issues before the final roll-out of 5G. The project's overall concept is illustrated below, as shown in Fig. 1, which shows the three “core” streams and also depicts the flow from research through adaptation into existing testbeds and commercial testbed deployments, to the real-world trials of the 5G radio access network (RAN) and of the more extensive 5G network. The 5G-DRIVE project “brings together” solid research competence, commercial-grade testbeds, and some of the stakeholders who will eventually become significant customers of 5G systems.

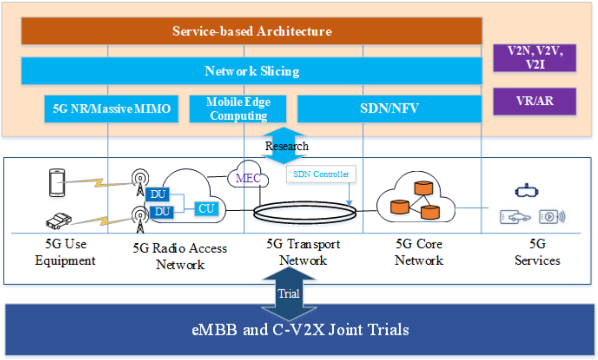

Fig. 1. Overall architecture of 5G-DRIVE.

For the needs of the 5G-DRIVE experimentation and evaluation activities, a joint trial testbed has been defined with a 5G testbed installation, co-located in OTE and ORION Innovations premises. The testbed is defined in an evolutionary approach and allows the gradual introduction and testing of new equipment, as well as new mechanisms, algorithms and protocols. These characteristics are exploited in the entire 5G-DRIVE’s context. In the research stream, the project investigates network and RAN slicing, mobile edge computing (MEC), massive multiple-input multiple-output (MIMO) for the 5G NR (New Radio), as well as SDN (Software-Defined Network) & Network Functions Virtualization (NFV) techniques applied to different traffic and load scenarios. Research-centric methods and mechanisms related to NFV, SDN and network slicing of the project are under development and implementation into the testbed. An overview of the joint trial testbed can be seen in Fig. 2.

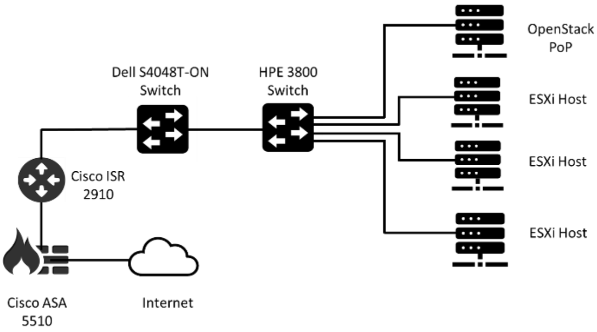

Fig. 2. Overview of the joint OTE – ORION testbed.

The core objective of the 5G-DRIVE project is to extensively trial eMBB and Vehicular to Everything V2X service delivery in real-world conditions. In the current work, a part of the eMBB related tasks is being implemented similar to the research and innovation in the virtualization and network slicing fields. A significant goal of the project framework is to build engagement between current 5G developments in the EU and China through joint trials and research activities so that to ease technology convergence, spectrum harmonization and business innovation in advance of potential large-scale market 5G deployment. The stringent requirements for the delivery of such services will be defined jointly with the mobile operators in the consortium as well as stakeholders form the automotive and intelligent transports markets. These are involved in the use case and trial requirements definition, as well as in the subsequent implementation/analysis.

The 5G-DRIVE effort is based upon the existing, currently, under design 5G standards, namely the 3GPP releases 15-16, and any relevant findings will be fed back into the appropriate standardization organization and working groups. The project will exploit the 5G testbed, which has been set up with commercial-grade and experimental equipment and are used to test new research outcomes and new services. To achieve this goal, 5G-DRIVE develops “key” 5G technologies and a pre-commercial testbed for 5G new services using mechanisms of slicing in collaboration with the “twinned” Chinese project led by China Mobile.

III. 5G NETWORK SLICING DESIGN AND EVALUATION

Current slicing solutions focus on different domains or a combination of thereof. However, at the edge, the research results are still in their infancy, especially regarding resource allocation across different layers of the network environment. Moreover, integration of MEC with network slicing is still an open issue [11]. Therefore, this paper presents two network slicing solutions, along with their performance evaluation for a set of test network services. Firstly, the OSM [12] slicing platform is presented and measured in terms of scalability. Secondly, the Katana slice manager [13] is introduced, a holistic approach for E2E service slicing for 5G networks. Katana slice manager is also evaluated in terms of end-to-end deployment time for two 5G-DRIVE prototype VNFs.

A. OSM Slicing Evaluation

The performance of slice orchestration is an important and so far this problem did not receive enough attention from researchers. In this section, we will describe some experiments related to the performance evaluation of the OSM orchestrator. To make the assessment, we have used the methodology proposed in [14]. The proposed in the paper KPIs are slice agnostic. The list of the measured KPIs includes:

(1) Slice Deployment Time (SDT) – a parameter that describes the interval between the slice deployment request and the moment in which slice is ready for operation. Unfortunately, this parameter depends on the slice template (blueprint) complexity, the performance of the orchestrator, and the time needed for the allocation of virtualized resources by the infrastructure. The slice complexity may deal with the footprint size of VNFs, their interconnection topology, amount of configuration parameters. Therefore, in a generic case, it is impossible to define the required value of SDT. It can be noted that SDT may be critical for some network slices, e.g. on-demand or short-lived ones, but much less significant for long-lived slices.

(2) Slice Deployment Time Scalability (SDTS) – a measure of scalability of slice deployment operations. To evaluate the scalability, it is possible to send N slice deployment requests of the same slice template and calculate SDTS in the following way:

$$SDTS = \frac{GSDT}{N \times SDT}$$

where GSDT is the overall time for the deployment of N identical slices and SDT is the deployment time of a single slice (as defined above). It is hard to determine the N value *a priori*. If the N value is too big, then there can be a problem with the availability of the requested resources. If it is too small, then the obtained result may not express the scalability of the orchestration well. We have used the SDTS parameter $N = 10$, and the SDTS is expected to be higher than 1.

(3) Slice Termination Time (STT) – a parameter that describes the interval between the slice termination request and the moment in which all slice allocated resources are released. If the time is long, it decreases the efficiency of the infrastructure resources usage.

The lifecycle KPIs have been obtained by observing interactions between the OSS/BSS and NFVO. Hence, the OSS/BSS can determine both the beginning and the end of the procedure. It is worth noting that in the case of disturbances of intra-MANO communication (e.g. OSS/BSS is notified about the delay of procedure execution due to the need of retrying). The main goal of the experiments was to evaluate the scalability of lifecycle orchestration of OSM. The experiments have been carried out using two identical computers, each of them had Intel® Core™ i7-8700 CPU @ 3.2 GHz and 16 GB memory. On the first machine, OSM MANO was installed, whereas on the second machine the OpenStack used as VIM has been installed. The tests were performed through deployment and termination from 1 to 50 identical instances using *hackfest-basic-vnf* template on CirrosOS image. This minimal Linux distribution was designed for use as a test image on clouds such as OpenStack. To average the results, each test was conducted ten times. We have measured the slice deployment time by deploying 1, 3, 5, 10, 15, 20, 30, 40 and 50 instances simultaneously and one by one using the same template. We have measured the SDT and the GSDT parameters. The normalized slice deployment time has also been calculated. The obtained GSDT and SDTS results are presented in Fig. 3, and Fig. 4, respectively.

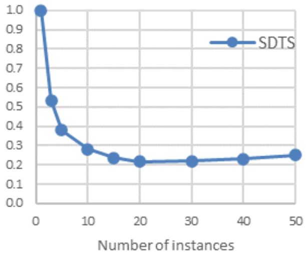

Fig. 3. SDTS results for a various number of instances.

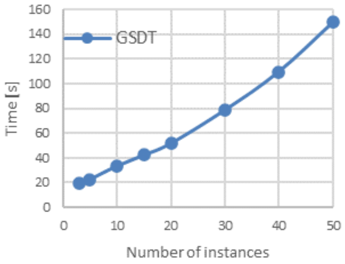

Fig. 4. GSDT results for a various number of instances.

In Fig. 3, we have presented the normalized value of SDTS – a deployment time of a single slice when N slice instances are deployed in comparison to a single slice deployment time. In Fig. 4, the obtained measurements for N = 3-50 are presented. As it can be found the GSDT increases nearly linearly in the tested range. For N = {1,3,5}, we have received significantly higher values than for the larger value of N. It can be justified by the need of the initialization of some libraries and system processes. Linear SDT increase shows good scalability of the OSM orchestrator.

The comparison of deployment (GSDT) and termination (STT) times of N slice instances is shown in Fig. 5.

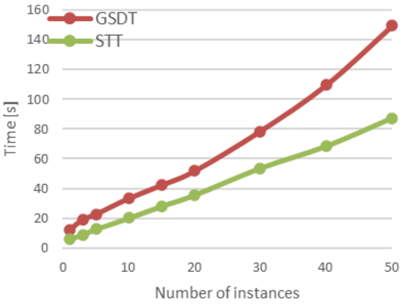

Fig. 5. Comparison of GSDT & STT for a various number of instances.

As it can be seen, the termination time of a single slice is nearly constant for N = 10-50 and much higher for the termination of a single slice. The obtained results show the excellent scalability of the OSM orchestrator again. A longer time for a single instance can be explained by a time needed to delete some processes that are common for all slice instances. The time of slices termination is only two times shorter than the time required for their deployment.

B. *Katana Slice Manager*

Katana slice manager is a novel network slicing solution, developed within the frame of 5GENESIS, aimed towards 5G deployments, with support of end-to-end slicing for multiple layers of the infrastructure. The design and implementation of Katana is based on the 3GPP Technical Report TR 28.801 “Telecommunication management; Study on management and orchestration of network slicing for next generation network” [15], following the concepts of the 3GPP specification the following definitions have been implemented:

- Network Slice Instance (NSI): An NSI includes all functionalities and resources necessary to support certain set of end-to-end communication services.
- Network Slice Template (NST): The NST describes the Network Slice to be created.
- Components of an NSI: The NSI is comprised of Virtual or Physical Network Functions (NFs). These NFs can be dedicated to an NSI or shared among multiple NSIs. If the NFs are interconnected, the Slice Manager contains information relevant to connections between these NFs, such as topology of connections, network graph, link requirements, etc.
- Network Slice Subnet Instance (NSSI): An NSI may be composed of NSSIs. The NSSIs may represent different domains of the physical infrastructure, such as the NFVI, Transport Network, RAN, etc. An NSSI may include other NSSI(s).

The Katana Slice Manager is based on a highly modular architecture, built as a mess of microservices, each of whom is running on a docker container. The key advantages of this architectural approach are that it offers simplicity in building and maintaining applications, flexibility, and scalability, while the containerized approach makes the applications independent of the underlying system. Fig. 6 shows an overview of the building blocks that comprise the Katana Slice Manager.

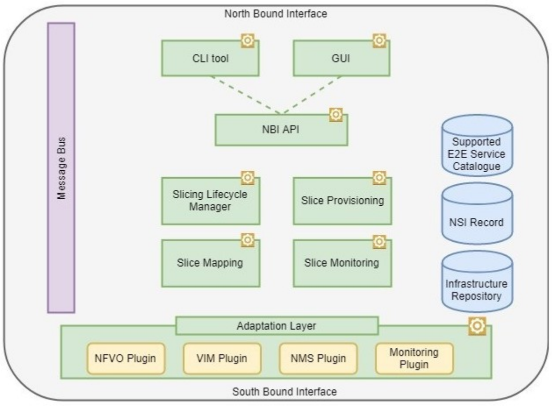

Fig. 6. Overview of the Katana Slice Manager.

Based on the diagram in Fig. 6, Katana is split in a multiple components-specific functions. The core components and backbone of Katana are the following:

a. North Bound Interface API

The North Bound Interface API module implements RESTful APIs, that can be consumed by a component of upper layers, e.g. the Experiment Lifecycle Manager, a user/experimenter or the Slice Manager administrator. This module is the brain of the 5GENESIS Slice Manager, implementing the Create, Read, Update and Delete functions. The role of this component is twofold. On the one hand, it receives requests from the NBI API module and takes any necessary actions for the activation, modification or deactivation of a network slice. On the other hand, it receives messages from the Slice Monitoring module, regarding the status change of a deployed slice. It interacts with the other components of the Slice Manager, in order to trigger the process that needs to start, depending on the received messages.

b. Slice Mapping

This module hosts a very important process that runs during the slice creation phase, the placement process. This process is responsible for optimally selecting the infrastructure resources to be used for a new slice, based on the slice requirements, as they are described in the NST, and the available resources of the infrastructure layer.

c. Slice Provisioning

The Slice Provisioning module receives requests from the Slicing Lifecycle Manager service in order to set up, configure or delete the Wide Area Network paths, the isolated NFVI tenant spaces and all the required Network Services, to configure the radio component parameters and register the newly created slice to the Monitoring system. It does so by using the Virtual Infrastructure Manager (VIM), NFV Orchestrator (NFVO), Network Management System (NMS) and Monitoring Plugins of the Adaptation Layer.

d. Slice Monitoring

The Slice Monitoring module is responsible for monitoring the health and the status of every deployed slice. It uses the Adaptation Layer plugins to send status check messages to the MANO components below the Slice Manager and reports any slice status change to the Slicing Lifecycle Manager service.

e. Adaptation Layer

The Adaptation Layer module provides a level of abstraction regarding the underlying domain technology, making it feasible for the Slice Manager to operate over any MANO layer component without any modifications to its core functionality, as long as the proper plugin has been loaded. This module is comprised of VIM, NFVO, NMS and Monitoring plugins, one for each of the MANO layer components supported by the Slice Manager. The responsibility for each plugin is to receive request messages from the Slicing Lifecycle Manager and translate them to the proper API call of the supported component. After that, it is in charge of properly handling the responses from the underlying components to the API calls.

C. Network Slicing Deployment for 5G new services

After the detailed description of the Katana Slice Manager, the experimental process proceeded with a set of trials for a group of 5G-DRIVE VNFs, the vCache and the vDPI, in terms of slice deployment time. In Fig. 7 the detailed results of the slice deployment times, total and per task/step can be seen.

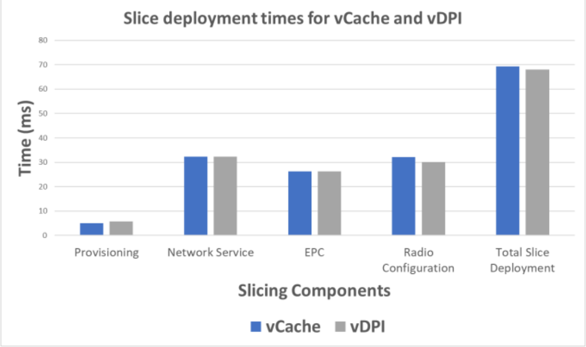

Fig. 7. Slice deployment times for vCache and vDPI.

As it can be deduced from Fig. 7, the total time for a network service deployment does not vary much in relation to the type of the service and is calculated to approximately 68 s. It should be noted that the total time includes the entire end-to-end deployment process, from the VNF placement to the RAN configuration. The tests were performed in the joint OTE-ORION testbed, with an OpenStack Queens VIM, and a commercial Amarisoft 5G system. Additionally, to the presented times, the total calculated time includes the Placement time (~0.0030 s) and the WAN deployment/setup time (0.12 s), which were omitted from the figure as they do offer minor insight on the total duration of the process.

IV. EVALUATION OF 5G NEW SERVICES

In this section, an experimental set of tests is performed in the 5G-DRIVE testbed in order to evaluate and assess not only the edge services capabilities but also the integration of the adopted slicing mechanism – the experimental test set comprised of 2 prototype VNFs, a vCache and a vDPI. The vCache VNF aims to improve the access of HTTP content at the edge of the infrastructure, in our case the edge server. The vCache stores recently accessed content, and updates in an intelligent manner, providing the user with a significant improvement in the time required to access specific content. The vCache is based on the widely used tool Squid [16] and was modified in order to serve HTTPS content also. Moreover, modifications were performed on the initial setup of the VNF in order to comply with the Katana slice manager's API and endpoints. The vCache was deployed in the form of an OpenStack VM, with 2 CPUs and 8 GB of RAM, over an OpenStack Queens.

In the first set of tests, the vCache was used to cache native HTTP content. As 5G-DRIVE is focused on the synergies between EU and China partners, among the websites tested, were the china.org.cn and baidu.com. As it can be clearly observed in Fig. 8, the vCache provided noteworthy improvement on the access time, reducing the latency up to 9 times less, in the case of baidu.com and 6 times less in the case of china.org.cn. These conclusions can be justified by the usually poor access to Chinese-based web content from European CDNs.

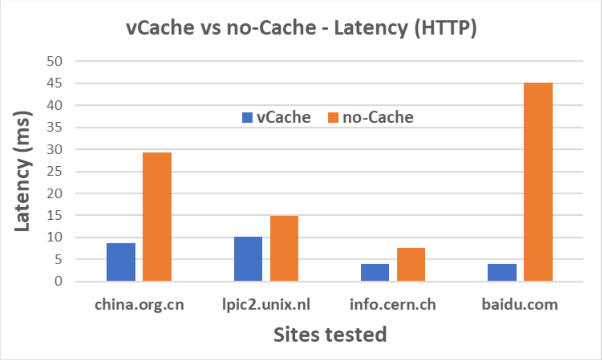

Fig. 8. Latency results for HTTP websites using vCache.

Based on the first set of outcomes, the vCache showed promising results, but in a limited group of website content, as the trend for web services forces modern websites to operate over HTTPS. However, the native Squid implementation does not support caching of encrypted content, thus disabling the normal operation of the vCache. Therefore, an HTTPS module was integrated in the vCache package so as to support a wider variety of web services. In Fig. 9, the results for access times of web services over HTTPS are presented.

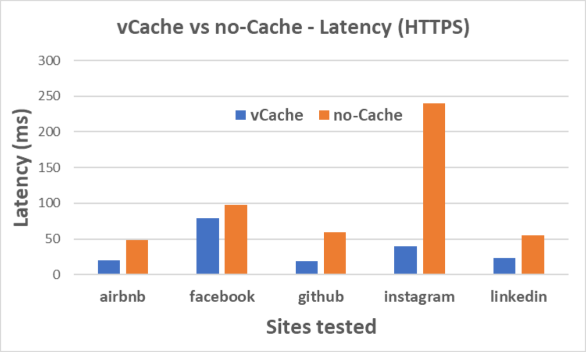

Fig. 9. Latency results for HTTPS websites using vCache.

As it can be deducted by the figure most HTTPS website access times are slightly improved, with the exception of Instagram, where the latency using the vCache is improved up to 4 times. Additionally, it can be concluded that, after comparing the 2 sets of results, HTTPS web services have a significantly increased load time (~50-200 ms), whereas HTTP services' latency varies in the range of 5-45 ms. This can be explained by the encrypted handshake communication required by the server and the client, in order to attain a secure connection.

Overall, the results show that the deployment of a vCache VNF at the edge proves to be a pivotal step towards an acceleration in vertical industries that want to exploit the benefits of 5G. In respect to 5G-DRIVE, the vCache can be deployed in a 5G-enabled platform, where the user will experience significantly lower loading times for various web services.

Another aspect investigated in these experiments was the correlation between the performance of the vCache service and various hardware metrics, e.g. CPU, RAM, Hard Disk writes. In the frame of 5G-DRIVE, the VNFs are planned to be run in a limited edge infrastructure where resources are to be planned carefully, so as to achieve maximum efficiency. For the case of vCache a Prometheus agent was integrated in the virtualized service and collected the noted metrics across the duration of the experiments. The collected data were processed by an ML mechanism based on TensorFlow, which outputted the results in Fig. 10 in the form of a correlogram.

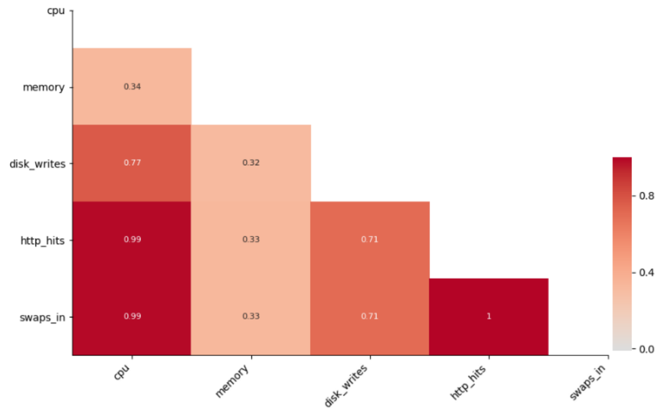

Fig. 10. Correlation graph for HTTP_hits and CPU, RAM, Hard Disk writes metrics.

The results presented in Fig. 10 provide an interesting insight in the inner computational demands of the vCache. Initially, it can be deducted that the CPU is the core of processing for the virtualized caching mechanism, and secondly the Hard Disk. Contrary to the general conception, RAM plays a secondary role in terms of content caching. It must be noted that the experiments were performed for 2 cache state scenarios, in the first scenario the cache memory was erased, and in the second one, it was left intact, thus measuring the actual relation between caching and hardware metrics.

In the second set of experiments, the vDPI VNF was used as an edge network analysis and security service. The vDPI was measured in terms of packet processing performance in various platforms, e.g. x86 and ARM. The main purpose of this evaluation was to measure the performance capabilities of each platform in order to assess their integration feasibility in an edge environment. In order to compress the computational demands of the vDPI, the service was virtualized using Docker, as it is significantly lighter than an OpenStack VM. In this context, it should be noted that ARM platforms, although less performant, are more energy efficient. The experiments were performed using Intel's DPDK-2220 x86 board, an ST Microelectronics B2239B ARM board, and a Raspberry Pi 3 B+ board.

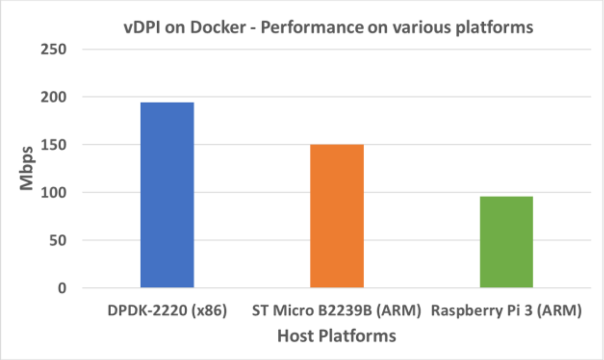

Fig. 11. vDPI packet processing performance results for various platforms.

From Fig. 11 it can be concluded that the x86 DPDK board clearly outperforms the respective ARM boards, by a margin of ~50-100 Mbps. It should be noted that all platforms are quite portable and energy-efficient and provide a viable candidate for an edge processing unit. Of course, the performance cannot be compared to a typical NFV setup [17]-[19], as the platform specifications are quite limited.

V. CONCLUSIONS

The paper presented the 5G enabled prototype architecture for edge services of the 5G-DRIVE project, which aims to evolve and improve collaboration between EU and China, in the field of 5G and novel vehicular services. The manuscript not only presented current open challenges in the field of 5G network slicing but also presented the preliminary integration with the Katana slice manager, a slicing mechanism developed in the frame of 5GENESIS and a detailed slicing evaluation of the OSM platform.

Additionally, the paper presented the two novel VNFs developed in the frame of 5G-DRIVE and evaluated their performance when deployed at the edge, through a slicing mechanism. The key outcome of the manuscript is the actual integration and performance analysis of prototype edge services through a slicing mechanism on a real testbed.

As 5G-DRIVE progresses, additional use cases will be investigated, along with trials closer to the vehicular type of communications. Furthermore, the presented Katana slicing mechanism will also be evolved in order to support more dynamic slice resource allocation in the RAN, as was investigated in [20]-[21].

ACKNOWLEDGMENTS

The research leading to these results has been supported by the H2020 5G-DRIVE ("HarmoniseD Research and trIals for serVice Evolution between EU and China") project, (no. 814956) and the 5GENESIS H2020 5G-PPP (no. 815178).

REFERENCES

[1] International Telecommunication Union – Radiocommunications Sector (ITU-R), "Recommendation M.1645-0 (06/2003): Framework and overall objectives of the future development of IMT-2000 and of systems beyond IMT-2000".

[2] International Telecommunication Union – Radiocommunications Sector (ITU-R), "Report ITU-R M.2243 (11/2011): Assessment of the global mobile broadband deployments and forecasts for International Mobile Telecommunications".

[3] J.G. Andrews, S. Buzzi,, W. Choi, *et al*., "What Will 5G Be?," IEEE JSAC, Special issue on 5G Wireless Communications Systems, vol. 32(6), pp. 1065-1082, 2014.

[4] European Commission, "5G: Challenges, Research Priorities, and Recommendations – Joint White Paper". European Commission, Strategic Research and Innovation Agenda.

[5] S. E. Elayoubi, S. Ben Jemaa, Z. Altman, and A. Galindo-Serrano, "5G RAN Slicing for Verticals: Enablers and Challenges, IEEE Communications Magazine, Institute of Electrical and Electronics Engineers (IEEE), 2019, 57, 28-34.

[6] A. Ghosh, Maeder, M. Baker, and D. Chandramouli, "5G Evolution: A View on 5G Cellular Technology Beyond 3GPP Release 15", IEEE Access, Institute of Electrical and Electronics Engineers (IEEE), 2019, 7, 127639-127651.

[7] Kourtis, M.-A.; Blanco, B.; et al., "A Cloud-Enabled Small Cell Architecture in 5G Networks for Broadcast/Multicast Services", IEEE Transactions on Broadcasting, Institute of Electrical and Electronics Engineers (IEEE), 2019, 1-11.

[8] R.-A. Stoica, and G. T. F. de Abreu, "Massively Concurrent NOMA: A FrameTheoretic Design for Non-Orthogonal Multiple Access", in 2018 IEEE 52nd Annual Asilomar Conference on Signals, Systems, and Computers, Oct. 2018.

[9] 5G-DRIVE H2020 5G-PPP Project [https://5g-drive.eu].

[10] G.Gardikis, A. J. Morgado, S. Mumtaz, J. S. de Puga, C. E. Palau, C. Skiadasn, H. Koumaras, M.-A. Kourtis, *et al.*, "The 5GENESIS testing facility as an enabler for integrated satellite/terrestrial 5G experimentation", 2019 IEEE Wireless Communications and Networking Conference Workshop (WCNCW), IEEE, 2019.

[11] L. Tomaszewski, S. Kukliński, R. Kołakowski, "A New Approach to 5G and MEC Integration", In: I. Maglogiannis, L. Iliadis, E. Pimenidis (eds.) "Artificial Intelligence Applications and Innovations. AIAI 2020 IFIP WG 12.5 International Workshops", AIAI 2020. IFIP Advances in Information and Communication Technology, vol. 585. Springer, Cham.

[12] OSM website: https://osm.etsi.org/

[13] Katana website: https://github.com/medianetlab/katana-slice_manager/

[14] S Kukliński, and L. Tomaszewski, "Key Performance Indicators for 5G network slicing", IEEE Conference on Network Softwarization (NetSoft 2019), 08/2019, Paris, France.

[15] 3GPP, " Technical Specification TS 28.500 - Telecommunication management; Management concept, architecture and requirements for mobile networks that include virtualized network functions," July 2020. [Online]. Available: https://portal.3gpp.org/desktopmodules/Specifications/SpecificationDetails.aspx?speci ficationId=2935.

[16] Squid website: http://www.squid-cache.org/

[17] T. Soenen, *et al.*, "Empowering Network Service Developers: Enhanced NFV DevOps and Programmable MANO", IEEE Communications Magazine, Institute of Electrical and Electronics Engineers (IEEE), 2019, 57, 89-95.

[18] M.-A. Kourtis, M. J. McGrath, *et al.*, "T-NOVA: An Open-Source MANO Stack for NFV Infrastructures", IEEE Transactions on Network and Service Management, Institute of Electrical and Electronics Engineers (IEEE), 2017, 14, 586-602.

[19] M.-A. Kourtis, G. Xilouris, V. Riccobene, M. J. McGrath, G. Petralia, H. Koumaras, G. Gardikis, and F. Liberal, "Enhancing VNF performance by exploiting SR-IOV and DPDK packet processing acceleration", 2015 IEEE Conference on Network Function Virtualization and Software Defined Network (NFV-SDN), IEEE, 2015.

[20] P. S. Khodashenas, B. Blanco, M.-A Kourtis, *et al.* "Service Mapping and Orchestration Over Multi-Tenant Cloud-Enabled RAN" IEEE Transactions on Network and Service Management, IEEE, 2017, 14, 904-919.

[21] M.-A. Kourtis, B. Blanco, J. Perez-Romero, *et al.*, "A Cloud-Enabled Small Cell Architecture in 5G Networks for Broadcast/Multicast Services IEEE Transactions on Broadcasting", Institute of Electrical and Electronics Engineers (IEEE), 2019, 65, 414-424.