

# Enabling Edge Computing Deployment in 4G and Beyond

Roberto Bruschi  
DITEN, University of Genoa, Italy  
roberto.bruschi@unige.it

Chiara Lombardo  
S2N National Lab., CNIT, Italy  
chiara.lombardo@cnit.it

Franco Davoli  
DITEN, University of Genoa, Italy  
S2N National Lab., CNIT, Italy  
franco.davoli@unige.it

Sergio Mangialardi  
S2N National Lab., CNIT, Italy  
sergio@tnt-lab.unige.it

Guerino Lamanna  
Infocom S.r.l, Genoa, Italy  
guerino.lamanna@infocomgenova.it

Jane Frances Pajo  
DITEN, University of Genoa, Italy  
S2N National Lab., CNIT, Italy  
jane.pajo@tnt-lab.unige.it

**Abstract**—While the integration of fourth-generation (4G) networks with Edge Computing technologies would anticipate the improvements foreseen by the coming of 5G, as well as smoothen the transition to the new technology, 4G does not natively support Edge Computing. Therefore, specific functionalities for user-plane integration and isolation of tenant spaces are required for effectively deploying Edge Computing in 4G networks. This paper describes the design of the end-point between the mobile and edge environments that has been integrated in the telecom layer platform of the MATILDA Project. Such end-point, designed in a Virtual Network Function (VNF), allows intercepting and forwarding data and control traffic towards external Data Networks. Instances of this VNF can be horizontally scaled according to a decision policy, which determines the minimum number of instances required for the current load. Results show that the latency ascribable to the VNF processing is sufficiently low to satisfy the delay budget for all 5G use cases up to 10 ms and that the decision policy based on the QoS Class Identifiers (QCIs) allows scaling with the traffic load, while still fulfilling the performance requirements of each application.

**Keywords**— Edge Computing, Virtual Network Functions, 5G, Quality of Service

## I. INTRODUCTION

With the upcoming fifth-generation (5G) mobile networks gaining momentum, the dramatic performance improvement promoted by this technology will be the driver for new vertical business models involving all the stakeholders, from vertical industries to Over-The-Top (OTT) providers and software developers. To this end, Edge Computing, initially defined by European Telecommunications Standards Institute (ETSI) as Multi-access Edge Computing (MEC) [1], has been widely accepted as a key technology [2] to bring application-oriented capabilities onto computing and storage facilities within telecom operators' infrastructures, much closer to end users. By exploiting softwarized infrastructures powered by Network Functions Virtualization (NFV) [3] and Software-Defined Networking (SDN) [4], Edge Computing allows to support a wide range of new use cases with low latency requirements and a high degree of personalization of networking, billing and features [5] enabled by the knowledge of user location and the network data available within the telecom premises.

A number of functionalities will be natively available in 5G networks for Edge Computing integration, such as the User Plane Function (UPF) and the Session Management Function (SMF) [6]. Since 4G networks will still exist for several years, their integration with Edge Computing would not only anticipate the technological (and economic)

improvements foreseen by the coming of 5G, but also allow for a smoother transition to the new technology. However, 4G was not conceived to support edge computing and a number of issues, mainly related to user-plane integration and isolation of tenant spaces, need to be overcome for a seamless and effective deployment.

In this paper, we address the deployment of Edge Computing in a 4G network and also provide some insights on the potential applications to 5G as well. In more details, the paper describes the end-point realized between the mobile and edge environments, by intercepting data and control traffic and managing forwarding towards locally-attached external Data Networks (DNs). Such end-point, designed in a Virtual Network Function (VNF), has been integrated in the telecom layer platform of the MATILDA Project [7] and is subject to the orchestration mechanisms in place for the fulfilment of the Quality of Service (QoS) requirements: the orchestrator performs horizontal scaling on the VNF instances according to a decision policy, which determines the minimum number of instances required for the current load on the basis of the User Equipment (UE) bearers and all associated QoS Class Identifiers (QCIs). This capability, along with the design based on the RESTful API technology, can be crucial for the application of these mechanisms to 5G as well, where latency requirements will be heterogeneous and even more stringent and the 5G Service-based Architecture (SBA) will modularize the design of the core functionalities making them fully pluggable.

Results compare the impact on the system occupation and on the latency obtained with two steering mechanisms and highlight the agility of the proposed deployment by showing how horizontal scaling performed according to the QCI of each incoming bearer can scale better with the traffic load, while still fulfilling the performance requirements.

The paper is organized as follows. Section II outlines the reference network environment and issues of Edge Computing integration. Section III describes the proposed deployment, while Section IV reports its evaluation. Finally, conclusions are drawn in Section V.

## II. EDGE COMPUTING DEPLOYMENT AND ISSUES

Edge Computing has been widely accepted as a crucial technology for achieving low latency targets. As such, while this paradigm is seen as a key pillar for 5G, the original ETSI MEC reference architecture [8] was actually defined to suit any mobile networks. This heterogeneity allows all the stakeholders involved in the mobile ecosystem to benefit from the evolution of the telecommunications business brought forth by Edge Computing, while still relying on 4G networks.

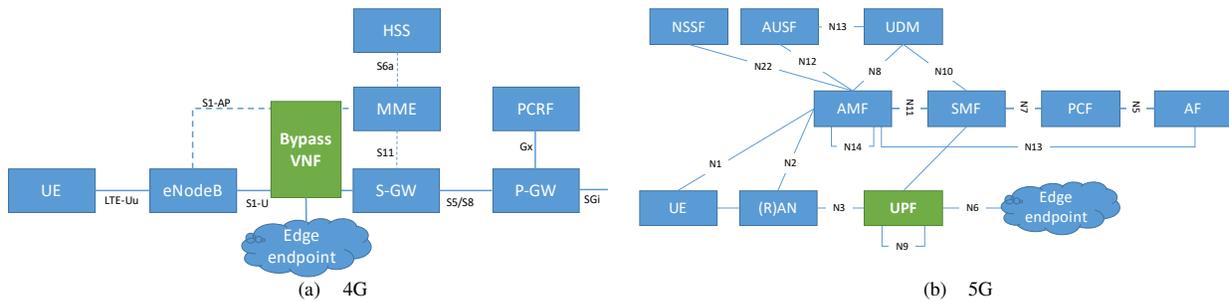


Fig. 1. Realization of the Edge Computing attach point (a) in 4G and (b) in 5G networks.

However, while Edge Computing deployment in 4G is seen as an opportunity to support applications with locality and/or low latency requirements, as well as a “gateway” to network infrastructure/service upgrades towards 5G, Edge Computing solutions have to be designed as an add-on feature to the pre-existing 4G networks in order to offer services in the edge, and as such they present a number of issues, especially related to user-plane integration and isolation of tenant spaces.

In fact, in order for Edge Computing technology to foster a dramatic reduction of latency times throughout high service continuity levels [9], application components running in different datacenters should be connected to UEs and among themselves. However, as identified again by the ETSI MEC Working Group (WG) in [9], the user-planes of applications and NFV services might be hosted in different isolated tenant spaces of VIMs. User-plane traffic cannot be exchanged easily between two isolated tenant spaces requiring the presence of so-called “attach points”. Attach points correspond to the virtual networks interconnecting application components and VNFs hosted in the VIM. As such, they are fundamental for the proper exchange of user-plane traffic between the application and network service providers. However, since independent stakeholders operate in isolated tenant spaces with potentially heterogeneous access/privacy requirements, the realization of attach points between applications and NFV services is a nontrivial task [10].

The 3GPP 5G core specifications define a set of new functionalities for enabling integrated Edge Computing deployments in 5G networks. Among these functionalities, the UPF realizes all the user-plane operations: its forwarding rules can be determined by application components themselves (by means of a SMF) to steer predetermined traffic flows towards a locally-attached external data network, which can be seen as the attach point between the application and the mobile network domains.

For 4G networks, the current 3GPP 4G architectural specification does not allow exposing reference points externally to realize these attach points. For this reason, additional functionalities are required to overcome the current specifications that do not allow exposing the S1-AP [11] and S1-U [12] protocol interfaces externally, but only to Mobility Management Entity (MME) and Serving Gateway (S-GW) nodes. As described in [13], Edge Computing requirements and performance are impacted by the location of the Edge Computing attach point. For example, installing the Edge Computing host at the SGI interface is considered suitable for 5G use cases in which the communication with the operator’s core site is optional, such as Mission Critical Push to Talk (MCPTT), and Machine-to-Machine (M2M) communications. On the other hand, a scenario in which the

attach point lies between the eNodeB and the Enhanced Packet Core (EPC) is very convenient in the presence of a C-RAN deployment.

The latter solution, which is called “bump-in-the-wire” and is shown in Fig. 1(a), has been developed in the scope of the MATILDA [7] and TRIANGLE [14] projects, by implementing a VNF that allows defining bearers on a per-bearer (more specifically, per-TEID, Tunnel Endpoint Identifier) basis, including VLAN tags, and to manage them by means of a RESTful interface. Nevertheless, although in the remaining of the paper the main focus will be on the evaluation of this implementation in the 4G context, the proposed solution can be ported to 5G, as well. In fact, the design principles, and even the code itself, of the Bypass VNF are suitable to be deployed as UPF (see Fig. 1(b) for an example): aside from providing the same basic functionalities, compatibility with the 5G SBA is ensured by the 3GPP decision of using RESTful APIs [15] for both the Core Network internal communication and North-/South-bound interfaces.

### III. PROPOSED EDGE COMPUTING DEPLOYMENT

The Bypass VNF realizes the Edge Computing attach point by intercepting data and control traffic before reaching the EPC, as shown in Fig. 1(a) and described in details in Section III.A. Depending on the traffic load, more than one instance of the Bypass VNF may be required to fulfil the QoS of each bearer. To this end, a decision policy has been designed to horizontally scale the Bypass VNF instances and to balance their load in a way that minimizes the resource utilization while respecting QoS constraints. The mathematical model driving the decision policy is described in detail in Section III.B.

#### A. Bypass VNF Functionalities

The main functionalities performed in the Bypass VNF are illustrated in Fig. 2. The Identify function intercepts S1-AP messages and parses their content against the information available at the OSS to univocally recognize the UE, to identify the eNodeB where the UE is attached and handover events, as well as to understand the configuration of its S1 bearers. If the intercepted packets do not belong to any of the deployed applications, they are directed again to the EPC without performing any further operations on them.

Then, an additional functionality, the Bypass, injects and retrieves packets from the S1-U protocol, which is formed by couples of unidirectional GPRS tunneling protocol - User (GTP-U) instances univocally identifying the source and destination IP addresses and the source and destination TEIDs. Furthermore, this functionality is in charge of realizing the end-point between the edge applications and the

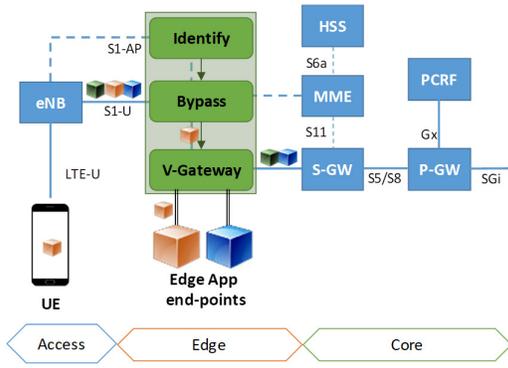


Fig. 2. Bypass VNF for introducing MEC in 4G networks.

RAN. To this end, when a packet belonging to the application of interest is identified, its GTP-U is removed and a VLAN tag is added to the packet that is then sent to the end-point. Finally, a Virtual Gateway is used to check the tag and forward the packet to the corresponding application end-point.

### B. TEID-Aware Decision Policy

We consider a datacenter  $i$  with CPU, RAM and disk capacity defined as  $C^{(i)}$ ,  $R^{(i)}$  and  $D^{(i)}$ , respectively.  $i$  allows deploying a maximum number  $V^{(i)}$  of bypass VNF instances. Since only one edge datacenter has been considered in this study, in the remaining of the paper the index ( $i$ ) will be neglected. Let  $U$  be the number of users whose traffic is sent to the edge datacenter. The traffic load of a user  $u$  entering the bypass instance  $v$  can be expressed as  $\lambda_v^u$ . We assume these traffic loads to be random variables independent among themselves. So, we can write the average value of the traffic load entering the bypass instance  $v$  as follows:

$$\lambda_v = \sum_{u=1}^U \lambda_v^u x_{uv} \quad (1)$$

with  $x_{uv}$  being a binary variable that equals 1 if the traffic of user  $u$  is handled by the bypass instance  $v$ .

The latency affecting  $u$  is a function of the traffic load and can be defined as

$$W_u(\sum_{v=1}^V \lambda_v^u x_{uv}) = W_u(\lambda_v) \quad (2)$$

We can associate the user to  $W_u^*$ , corresponding to the most stringent latency requirement among the QCI of the applications the user is subscribed to. In assigning the traffic of  $u$  to an instance  $v$ , it is required that the latency affecting the user stays below  $W_u^*$ . Hence, we can define the following constraint:

$$W_u(\lambda_v) \leq W_u^* \quad (3)$$

The objective of the orchestration criterion is to find the minimum number  $v^*$  of VNF instances required to process the incoming traffic. In more detail, by defining a binary variable  $y_v$  that equals 1 if the VNF instance  $v$  is active, to include only active instances in the optimization procedure, the problem can be stated as follows:

$$v^* = \min \sum_{v=1}^V y_v \quad (4)$$

$$\sum_{v=1}^V c_v y_v \leq C \quad (5)$$

$$\sum_{v=1}^V r_v y_v \leq R \quad (6)$$

$$\sum_{v=1}^V d_v y_v \leq D \quad (7)$$

$$W_u(\lambda_v) \leq W_u^*, \quad u = 1, \dots, U \quad (8)$$

$$y_v \in \{0, 1\} \quad (9)$$

Equations (5)-(7) constrain the VNFs' resource requirements to be below the available datacenter capacity. In order to satisfy Equation (8), it is required to find  $\lambda^*$ , which is the maximum incoming load allowing to fulfil the latency cap  $W_u^*$ . Since  $W_u$  is a function of the total load of the VNF as expressed in Equation (2), all flows  $u$ , once assigned to an instance  $v$ , will experience the same latency, regardless of their individual requirements. In fact, the presence of a classifier inside the VNF would cause an additional computational time which would even be useless in the case of saturation, where losses would appear before classification. Hence, the most stringent among all  $W_u^*$  must be taken into account for all flows, and Equation (8) becomes:

$$W_u(\lambda_v) \leq \min_{\forall u: x_{uv}=1} W_u^*, \quad u = 1, \dots, U \quad (10)$$

Since we can assume  $W_u$  to be monotonic, it can be inverted to provide a function  $\lambda_v^u(W_u)$ , and we can then determine its upper bound  $\lambda^*$  by interpolating the characterization of the Bypass VNF (provided in a datasheet, or empirically determined from performance evaluations), which gives the latency as a function of the load and thus can be used to determine  $\lambda^*$  for any given  $W_u^*$ , as will be shown in the next section.

## IV. EVALUATION

This section reports the results of a number of tests performed to assess the behavior of the proposed Edge Computing deployment. In details, Section IV.A provides a characterization of the Bypass VNF, and Section IV.B evaluates the performance of the proposed attach point in 4G. Considerations on 5G deployment are reported as well.

### A. Characterization of the Bypass VNF Performance

Tests have been performed to characterize the delay between the UE and the destination of its traffic ascribable to the presence of the Bypass VNF. Specifically, the testbed consists of a traffic generator and two servers, as shown in Fig. 3. Both servers are equipped with an Intel Xeon E5-2643 v3 processor (2 CPUs 3.40 GHz, with 6 cores and 128 GB of RAM), the operating system is Debian 9.0, kernel version 4.13.4 x86\_64. The transmitting port of the traffic generator behaves as a UE, and the receiving one represents the end-point of the traffic; latency is measured between these endpoints. The first server provides the GTP encapsulation required to emulate the behavior of the S1-U protocol, which is not available in the traffic generator, and is realized by using a Linux Virtual Machine (VM). The second server hosts a VM that contains the Bypass VNF, which inspects the incoming traffic and, when packets belonging to the service of interest are identified, removes their GTP-U, adds a VLAN tag and then sends the packets to the end-point (e.g., the receiving port of the traffic generator). Traffic has been transmitted at increasing rates from 10 to 200 kpkt/s, with packet sizes set to 1440 Bytes.

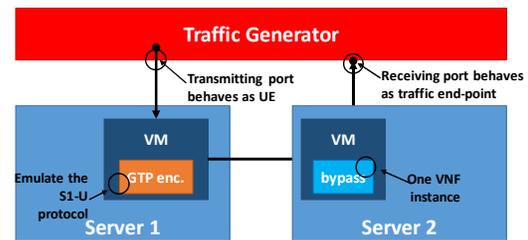


Fig. 3. Testbed configuration.

The obtained performance is shown in Fig. 4. The latency ascribable to the VNF processing is sufficiently low to satisfy the delay budget for all services as per 3GPP Standardized QCI characteristics [16], considering a radio round-trip time of 20 ms [17] and an LTE backhaul delay of 20 ms [16]. If we consider the application to the 5G scenario, we can reduce radio and backhaul delays to 4 [18] and 1 ms [6], respectively. Considering standardized 5QI values from [6], such performance is sufficient to handle almost all of the use cases under the URLLC umbrella; end-to-end latency requirements below 10 ms, such as for the Electricity Distribution- high voltage use case, will require specific infrastructure interventions to ensure radio delays below 2 ms.

### B. Numerical Results

As anticipated in Section III.B, the experimental measures in Fig. 4 can be interpolated to obtain a function characterizing the behavior of the Bypass VNF. With this function, for any desired threshold on the latency  $W_u^*$ , it is possible to determine the corresponding maximum traffic load  $\lambda^*$  allowed for a specific instance. In order to test the outcomes of the policy, we consider two simple orchestration mechanisms and compare their impact on the system occupation and the latency.

The first mechanism, named Case A in the results, provides the minimization of the number of VNF instances without discriminating on the specific QoS requirement of each bearer: in more details, traffic is shared on a per-TEID basis among a number of active Bypass VNF instances whose  $\lambda^*$  is the same for each instance, corresponding to the most stringent latency requirement among the hosted applications. This case corresponds to the application of the decision policy in Section III.B to the total incoming traffic load. The second mechanism, Case B, still minimizes the number of active instances, but the decision policy also takes into account the QCI of the bearers and applies the policy for each class. Accordingly, subgroups of VNF instances (one for each class identifier) are obtained, which have different  $\lambda^*$  and receive traffic from the corresponding bearers. It is worth noting that this case is not in contrast with the assumption made for Equation (10), because the classification is not performed by the VNF itself but by the orchestrator.

In order to test and compare the two cases, we consider a number of bearers varying from 100 to 1000, each one with a random traffic rate between 1000 and 2000 pkt/s. Such rates have been selected according to the Cisco Mobile Visual Networking Index (VNI) mobile speed forecasts [19]. Each bearer corresponds to one of the available applications, which have been selected among the 3GPP use cases to provide heterogeneous latency requirements. Namely, App1 belongs to the Discrete Automation use case, and App2 and App3 to Real-Time Gaming and Conversational Voice, respectively. Their packet delay budgets [16] correspond to 10 ms, 50 ms and 100 ms, respectively. The association between bearers (TEIDs in the following graphs) and applications has been randomly generated as well, and the offered load of each application, along with the total load, for a growing number of TEIDs, is shown in Fig. 5.

Fig. 6 reports the traffic load shares among the number of VNF instances selected by the orchestrator in Case A. In this case, the orchestrator adds a VNF instance when the available ones reach the value  $\lambda^*$  of incoming load, which for this test case, for each active instance, equals 200 kpkt/s to satisfy the

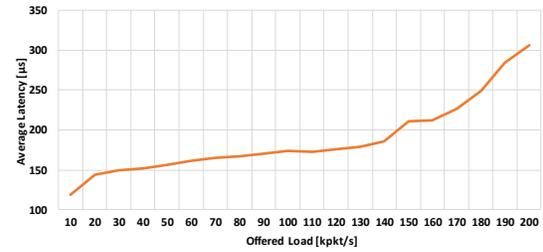


Fig. 4. Average latency of the Bypass VNF obtained for increasing offered load.

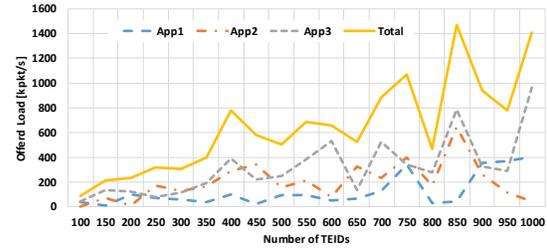


Fig. 5. Traffic generated to test the orchestrator.

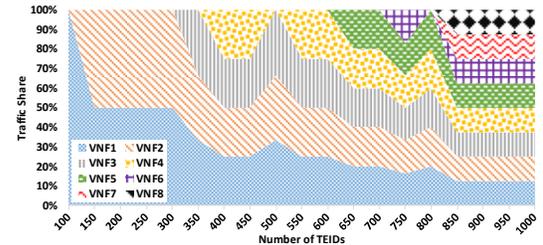


Fig. 6. Optimal number of Bypass instances and load shares according to increasing number of TEIDs for Test Case A.

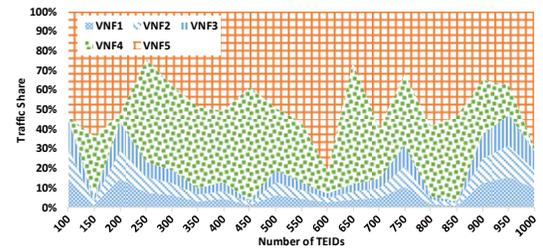


Fig. 7. Optimal number of Bypass instances and load shares according to increasing number of TEIDs for Test Case B.

latency constraint of App1. As the traffic grows with the number of TEIDs, the optimal allocation corresponds to an even share of the traffic among the available instances.

The traffic shares for Case B are shown in Fig. 7. In this case, the decisions are not based on the total traffic, but the number of VNF instances depends on the traffic ascribable to the individual Apps rather than the aggregate. As a result, three instances, namely VNF1, VNF2 and VNF3 share the traffic from the bearers associated with App1, leaving the remaining two VNFs to handle a higher traffic volume with lower latency requirements. The number of VNFs required in Case A, handling all traffic according to the most stringent latency threshold, is higher for higher loads with respect to Case B, but for lower rates there is a significant saving of resources, as the sharing in Fig. 7 is performed among five VNFs throughout the whole test.

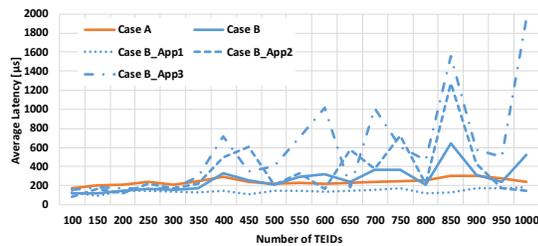


Fig. 8. Latency results for the two test cases.

Finally, the average latency ascribable to the Bypass for the two cases is reported in Fig. 8. Since in Case B bearers are associated to specific QCIs, the average latency is also reported on a per-App basis (dotted lines in Fig. 8) in addition to the average one. It can be noticed that Case A provides lower average latencies for a number of UEs above 400: in fact, for this load, as can be seen from Fig. 6 and Fig. 7, the number of active instances in Case A overcomes the one in Case B. However, the latency obtained for the traffic associated with the App1 QCI requirement in Case B is always lower than the one in Case A.

Although the policy of Case A scales better with the number of bearers and, on average, it allocates a lower number of VNF instances, by sharing traffic according to different QCI levels it is possible to achieve better granularity and fulfil heterogeneous QCI requirements even in the presence of huge amounts of traffic: in fact, while above 800 TEIDs case A provides lower average latencies, it does so at the cost of instantiating three more VNFs, while Case B uses only five VNFs, still respects the desired QCI requirements and even provides better latencies for App1 and App2 with respect to Case A. This aspect will be particularly relevant considering the growth of mobile traffic, the heterogeneous requirements of the use cases and their low latency requirements that will be fostered by 5G.

## V. CONCLUSIONS

This paper has proposed a solution for the deployment of Edge Computing in 4G networks. While Edge Computing is widely recognized as a fundamental technology to fulfil mobile network requirements, and as enabler for 5G networks, its integration with current mobile networks is penalized by the 4G architectural specification that does not allow exposing reference points externally to realize the attach points between the radio and the edge environments. To this end, this paper has proposed a design of the end-point between the access and the edge networks. This solution, integrated in the telecom layer platform of the MATILDA Project, has been designed in a VNF and allows intercepting and forwarding data and control traffic towards applications allocated in the edge network. Moreover, the VNF instances can be horizontally scaled according to a decision policy, which determines the minimum number of instances required for the current load.

Results have assessed that the Bypass VNF can satisfy the delay budget for all 5G use cases up to 10 ms and can be horizontally scaled with the traffic load, while still fulfilling the performance requirements of each application.

## ACKNOWLEDGMENT

This work has been supported by the Horizon 2020 5G-PPP Innovation Action MATILDA (Grant Agreement no. 761898) and by the Horizon 2020 Innovation Action SPIDER (Grant Agreement no. 833685).

## REFERENCES

- [1] ETSI GS MEC 002 2016, "Mobile Edge Computing (MEC); Technical Requirements", URL: [http://www.etsi.org/deliver/etsi\\_gs/MEC/001\\_099/002/01.01.01\\_60/gs\\_MEC002v010101p.pdf](http://www.etsi.org/deliver/etsi_gs/MEC/001_099/002/01.01.01_60/gs_MEC002v010101p.pdf)
- [2] 5G PPP Architecture Working Group, "View on 5G Architecture (Version 2.0)," URL: [https://5g-ppp.eu/wp-content/uploads/2017/07/5G-PPP-5G-Architecture-White-Paper-2-Summer-2017\\_For-Public-Consultation.pdf](https://5g-ppp.eu/wp-content/uploads/2017/07/5G-PPP-5G-Architecture-White-Paper-2-Summer-2017_For-Public-Consultation.pdf).
- [3] M. Chiosi et al., "Network Functions Virtualization: An Introduction, Benefits, Enablers, Challenges and Call For Action," In Proceedings of the SDN and OpenFlow World Congress, Darmstadt, Germany. ETSI White Paper. URL: [https://portal.etsi.org/nfv/nfv\\_white\\_paper.pdf](https://portal.etsi.org/nfv/nfv_white_paper.pdf)
- [4] "Software-Defined Networking: The New Norm for Networks, Open Networking Foundation (ONF)," White Paper, Apr. 2012.
- [5] "The Tactile Internet", ITU-T Technology Watch Report, August 2014.
- [6] The 3GPP Association, "System Architecture for the 5G System," 3GPP Technical Specification (TS) 23.501, Stage 2, Release 16, version 16.0.2, Apr. 2019.
- [7] MATILDA - A Holistic, Innovative Framework for Design, Development and Orchestration of 5G-ready Applications and Network Services over Sliced Programmable Infrastructure, <http://www.matilda-5g.eu/>.
- [8] Several authors, "Mobile-Edge Computing – Introductory Technical White Paper," Sept., 2014. <https://portal.etsi.org/portals/0/tbpages/mec/docs/mobile-edge-computing-introductory-technical-white-paper-v1%2018-09-14.pdf>
- [9] ETSI Group Report MEC 018, "End to End Mobility Aspects", version 1.1.1, October 2017.
- [10] "Mobile Edge Computing (MEC); Deployment of Mobile Edge Computing in an NFV Environment," ETSI Group Report (GR) MEC 017, version 1.1.1, Feb. 2018.
- [11] 3GPP, "Evolved Universal Terrestrial Radio Access Network (E-UTRAN); S1 Application Protocol (S1AP)," Technical specification no. 36.413, version 13.6.0, Release 13, ETSI TS 136 413, Aug. 2017. URL: [http://www.etsi.org/deliver/etsi\\_ts/136400\\_136499/136413/13.06.00/ts\\_136413v130600p.pdf](http://www.etsi.org/deliver/etsi_ts/136400_136499/136413/13.06.00/ts_136413v130600p.pdf).
- [12] 3GPP, "Evolved Universal Terrestrial Radio Access Network (E-UTRAN); S1 general aspects and principles", Technical specification no. 36.410, version 9.1.1, Release 9 ETSI TS 136 410, May 2011. URL: [http://www.etsi.org/deliver/etsi\\_ts/136400\\_136499/136410/09.01.01\\_60/ts\\_136410v090101p.pdf](http://www.etsi.org/deliver/etsi_ts/136400_136499/136410/09.01.01_60/ts_136410v090101p.pdf).
- [13] "MEC Deployments in 4G and Evolution Towards 5G", ETSI White Paper, February 2018.
- [14] TRIANGLE Project - 5G Applications and Devices Benchmarking, <https://www.triangle-project.eu/>.
- [15] The 3GPP Association, "Common API Framework for 3GPP Northbound APIs", 3GPP Technical Specification (TS) 23.222, Stage 2, Release 15, version 15.2.0, Jul. 2018.
- [16] The 3GPP Association, "System Architecture for the 5G System," 3GPP Technical Specification (TS) 23.501, Stage 2, Release 15, version 15.2.0, Jun. 2018.
- [17] P. Synnergren, T. Dudda, "LTE Latency Improvement Gains", Online: <https://www.ericsson.com/en/blog/2014/11/lte-latency-improvement-gains>.
- [18] O. Teyeb, G. Wilkström, M. Stattin, T. Cheng, S. Faxér, H. Do, "Evolving LTE to fit the 5G future", Online: <https://www.ericsson.com/en/ericsson-technology-review/archive/2017/evolving-lte-to-fit-the-5g-future>
- [19] Cisco Visual Networking Index: Forecast and Trends, 2017–2022 White Paper.