# Enabling Edge Computing Deployment in 4G and Beyond

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

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

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

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

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

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

***Abstract*— Edge Computing is widely recognized as one of the enabling technologies for the upcoming fifth-generation (5G) mobile networks. By bringing application-oriented capabilities within the telecom operator infrastructure, a wide range of new use cases will be supported, with low latency requirements and a high degree of personalization of networking, billing and features. While the integration of 4G networks with Edge Computing technologies would anticipate the technological 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 QCI 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.

5G-ready applications will be composed of independent, cloud-native "microservices" [1] running on individual execution environments and deployed across multiple facilities. An application orchestrator is in charge of managing the 5G-ready application lifecycle, as well as the interconnectivity among its microservices, in order to fulfill the application performance requirements even in geographically distributed, multi-domain datacenters. This design takes advantage of both the programmable network and computational infrastructure, allowing high scalability levels and effective agility.

To this end, Edge Computing, initially defined by European Telecommunications Standards Institute (ETSI) as Multi-access Edge Computing (MEC) [2], has been widely accepted as a key technology [3] 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) [4] and Software-Defined Networking (SDN) [5], 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 [6] 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) [7]. 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 [8] 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 proposes a brief state of the art on the integration of MEC in 4G networks and beyond. Section III outlines the reference network environment and issues of Edge Computing integration. Section IV describes the proposed deployment, while Section V reports its evaluation. Finally, conclusions are drawn in Section VI.

## II. Related Work

While the attention on the deployment of Edge Computing in 5G is expanding in the scientific research, especially concerning the UPF design, the integration over LTE networks is still a relevant topic, as the coexistence of these two mobile network generations is expected to last for several years to come. Moreover, as will be shown in the next section, practices and lessons learned in 4G will foster a more effective 5G core design; in fact, the papers considered in this brief state of the art often do not show a clear separation between 4G and 5G, confirming the non-binary nature of the two technologies.

As explained in the introduction, our work focuses on the integration of MEC in a 4G network as well as the orchestration of the virtual resources providing the attach points. The most similar contribution to the first topic can be found in [9]. The authors recognize the interest for deploying MEC solutions in current 4G infrastructures; in this respect, they propose a management architecture that combines elements from the standard NFV framework with MEC functionalities. Moreover, the integration of the proposed architecture into LTE network is implemented, with reference to [10], by adopting the "Distributed SGW with Local Breakout" approach, that is, co-locating MEC hosts and the serving gateway (SGW) at the edge in the same VNF. The paper applies this architecture to the use case of a mobile edge application for innovative video service provision during crowded events. Authors of [11] aimed to evolve the 4G core towards 5G by deploying most EPC control functions in the edge to be closer to the eNBs. In the paper, such proximity is exploited to develop a state sharing mechanism across different data centers to transport state information in a controlled manner over the network, without further details on the actual integration aspects between the access and the edge. [12] explores the possibilities of designing MEC over fiber-wireless (FiWi) networks for WLAN, 4G LTE, and LTE-A HetNets. In more details, MEC servers are integrated at the edge of FiWi networks, in co-location with Optical Network Units (ONUs). The main focus of the paper is a TDMA-based unified resource management scheme for MEC over Ethernet-based FiWi networks. [13] analyzes the benefits of deploying mission-critical push-to-talk (MCPTT) services in an ETSI MEC architecture. Their comparison of the conventional network architecture and the distribution of MEC-based service planes emphasizes the architectural limitations of MEC solutions on the current EPC when dealing with emergency situations.

The work of Costa-Requena et al. [14] can still fit in the 4G category, as the paper consists of the implementation of a UPF integrated in a SDN switch, but it also discusses the migration from legacy 4G user plane to 5G UPFs. In particular, the UPF realizes the interconnection with the services in the edge by terminating GTP tunnels and steering traffic into dedicated L1/L2 links using VLANs. For the integration with the LTE core, the authors propose and compare two scenarios: in the first one, the UPF resides in the same VM as the EPC, while in the second one only the control plane of the UPF is located in the core with the data plane being in the edge closer to the RAN. Multiple UPF modules can be assigned on a per slice/user basis, but while the authors acknowledge that having a large set of UPFs might cause orchestration issues, they do not propose any specific solutions in this work.

Regarding Edge Computing deployment in 5G networks, interest is currently pointed towards the design of the UPF at the data plane and of the SMF at the control plane, with most of the focus on the former. For example, [15] deals with the application of prediction techniques to optimize mobile networks. In this respect, the paper proposes a learning-based user plane management in which UPFs are placed by SMF according to user behavior predictions: exploiting the user's forecasted next point of attachment, the SMF can anticipate the most suitable user plane position in terms of local transport network topology, the current load of active UPFs and the utilization of related user plane paths, as well as the direction of user movement before handovers occur.

[16] proposes a service function chaining (SFC) framework for enabling third-party stakeholders deploying proprietary UPFs in the 5G core. The key point for a fruitful coexistence of multiple UPFs is sharing the context information among the mobile infrastructure and the service providers. By exploiting context sharing, it is possible to dynamically determine the most suitable chain of UPFs for each traffic flow regardless of the owner of the individual functions. Finally, [17] focuses on call flow and load balancing algorithms at SMF and UPF level. In particular, the authors propose an algorithm for load balancing between 5G and WiFi: if the traffic recipient is connected to both access networks, the UPF can decide to balance traffic between them according to the real time information on load and network capacity. Such information is provided by a signaling call flow mechanisms that allows propagating current load messages between gNBs/WiFi APs and the core network.

A key contribution of our work is represented by the mechanism for horizontally scaling the virtual resources providing the attach points. Although no other works pairing the design of Edge Computing functions for 4G integration with their orchestration are currently available in the literature, a number of relevant papers are still worth mentioning for their contribution to the orchestration solution realm.

Authors of [18] focus on designing an integrated NFV and MEC orchestration solution for the deployment of container-based network services at the network edge. Their architecture is based on the Open Baton Management and Orchestration (MANO) framework and aims to overcome the typical lack of resources in edge nodes by deploying network functions in lightweight execution environments and orchestrating them among multiple points-of-presence. In [19], the authors argue that the ETSI MEC architecture is access-agnostic, so they consider a deployment independent from the mobile network generation. They propose a QoS awareness enhancement by means of a host proximity zoning framework for latency-aware MEC instances placement. [20] proposes a reference architecture for the orchestration and management of the Edge Computing ecosystem. After providing an analysis of the most common virtualization/centralization trends, with notable attention to the usage of the channel state information (CSI)

generated by UEs in LTE networks, the authors compare the impact of different eNB virtualization techniques on the availability of CSI at the Edge Computing platform.

## III. Telecom Layer Platform for 5G-ready Applications

The scope of the MATILDA Project is to deliver a holistic and innovative 5G framework to undertake the design, development and orchestration of 5G-ready applications and 5G network services over programmable infrastructures. To this goal, a telecom layer platform has been designed to realize the autonomic management of the lifecycle of 5G network slices and edge computing resources. Since the project was funded in 2017, the platform prototype has been initially designed to work with the 4G access and core technologies available ~~in~~ that moment; at the time of writing, integration with recently acquired ~~SDR~~ devices that can be programmed as 5G gNB [21] is in progress.

In accordance with [22], the main stakeholders actively involved in this environment are three: the vertical industry owing the application, the telecom service provider delivering 5G services, and the telecom infrastructure provider offering computing and communication facilities.

Fig. 1 depicts the main functional blocks composing the telecom layer platform, highlighted with the red dotted line:

- the **Operations Support System** (OSS), in charge of managing all functions and operations required for the placement of a 5G-ready application over a network slice, as well as maintaining the information on all the data regarding the deployed applications, network services, available resources, and so on
- the **NFV Orchestrator** (NFVO), responsible for the lifecycle management of the network services, both those composing the base 4/5G services and the ones provided to slices
- the **Wide-area Infrastructure Manager** (WIM), devoted to manage and monitor the wide-area communication resources, to create overlay networks for vertical applications and base telecommunication services, as well as to provide information on the resources available in the distributed 5G infrastructure
- the **Virtual Infrastructure Manager** (VIM – one instance per each distributed computing facility), abstracting and exposing computing, storage, and networking capabilities of datacenters within the 5G infrastructures.

As shown in the figure, the OSS and the NFVO act in the network service provider domain, while the VIM and the WIM in the network infrastructure provider. Vertical industries can autonomically manage the lifecycle of their application graphs by means of Vertical Application Orchestrators (VAOs). It is worth noting that all of these building blocks and their reference points are fully compliant with the specifications of the ETSI NFV architectural framework [23].

Even though, for the sake of simplicity, the figure shows only one stakeholder per domain, in ~~the~~ reality the whole architecture and related control systems have been designed with "multi-tenancy" and "multi-domain" as foundational principles. In fact, 5G-ready applications will be deployed across multiple geographically-distributed VIMs, potentially owned by more than one infrastructure provider.

In order to maintain the connectivity among the chainable components deployed in different datacenters, and between the application front-end components and the UE, a number of NFV services are required to terminate the network slice assigned to the vertical application and abstract the underlying infrastructure. The realization of the "attach points", represented by the red bullets in Fig. 1, still represents an open question precluding Edge Computing deployment in 4G networks.

### A. *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 [24] 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.

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

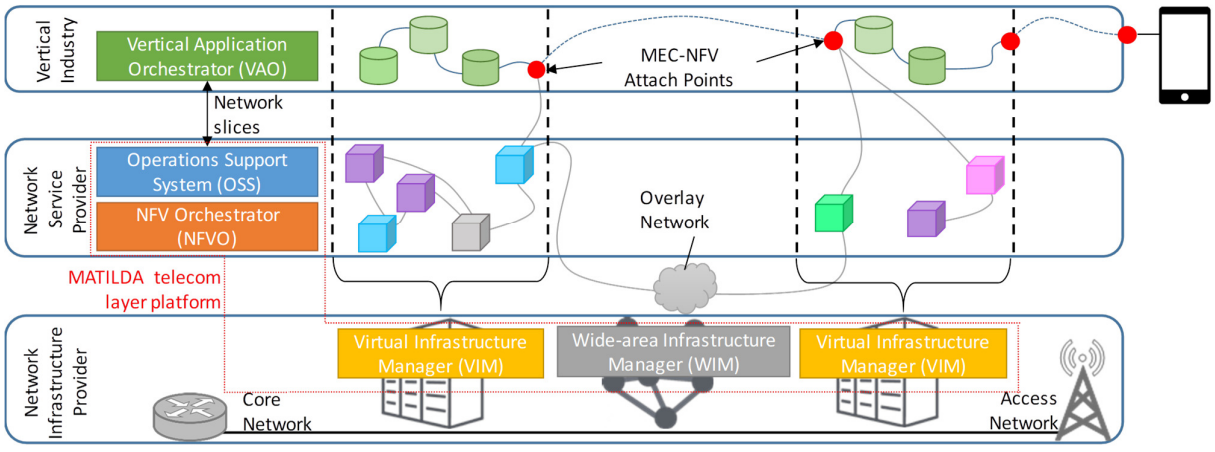
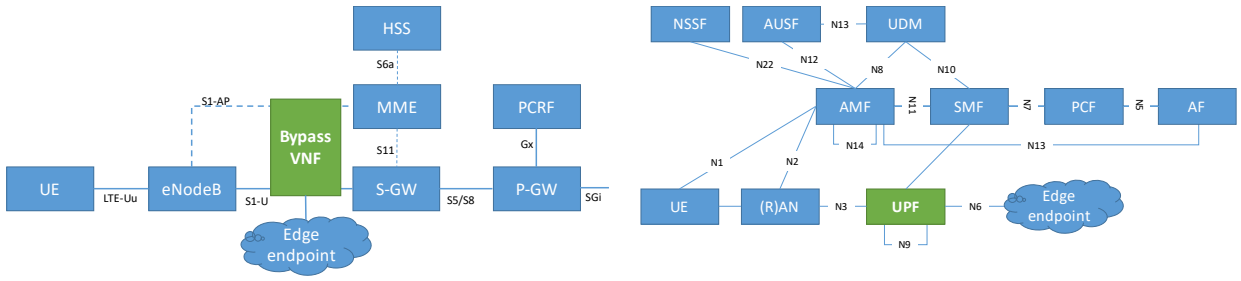



Fig. 1. Example of deployment of an application, driven by the MATILDA framework, into multiple VIMs over a 5G infrastructure.




(a) 4G (b) 5G

Fig. 2. Realization of the Edge Computing attach point (a) in 4G and (b) in 5G networks.

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 [25], 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 [25], 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, making the realization of "attach points" between applications and NFV services a non-trivial task [26]. In detail, such attach points correspond to the virtual networks interconnecting application components and VNFs hosted in the VIM.

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 [27] and S1-U [28] protocol interfaces externally, but only to Mobility Management Entity (MME) and Serving Gateway (S-GW) nodes. As described in [10], 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. 2(a), has been developed in the scope of the MATILDA [8] and TRIANGLE [29] 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. 2(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 [30] for both the Core Network internal communication and North-/South-bound interfaces.

## IV. 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. 2(a) and described in details in Section IV.A. Depending on the traffic load, the OSS may ask the NFVO to package more than one instance of the Bypass VNF in order to guarantee the fulfilment of the QoS requirements for each bearer. To this goal, 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 IV.B.

### A. *Bypass VNF Functionalities*

The main functionalities performed in the Bypass VNF are illustrated in Fig. 3. 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 RAN. To this goal, 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.

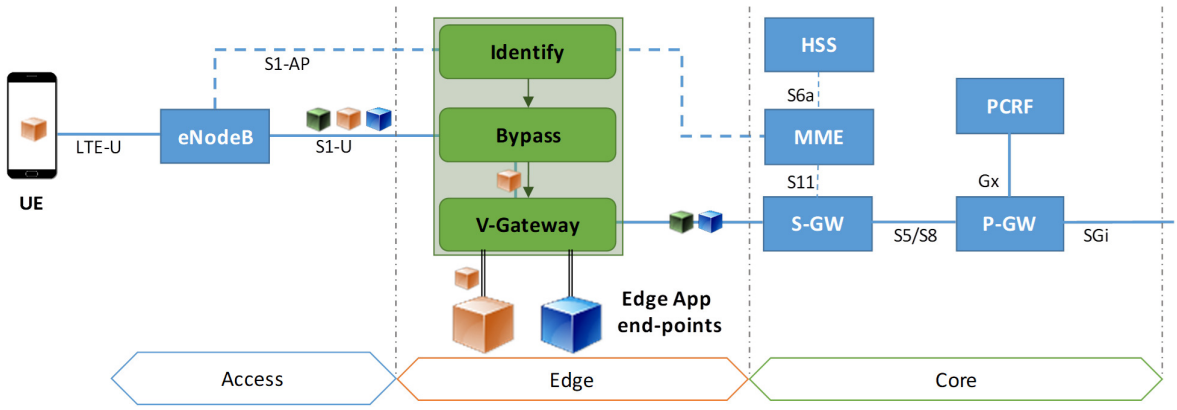



Fig. 3. Bypass VNF for introducing MEC in 4G networks.

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

$$W_u\left(\sum_{u=1}^{U} \lambda_v^u x_{uv}\right) = W_u(\lambda_v) \quad (2)$$

We can associate the user to $W_u^*$, corresponding to the most stringent latency requirement among the QCIs 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:

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

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:

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

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

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

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

$$W_u(t) \le W_u^*, \quad u = 1, \dots, U \quad (8)$$

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

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:

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

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.

## V. 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 V.A provides a characterization of the Bypass VNF, and Section V.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. 4. 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. 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

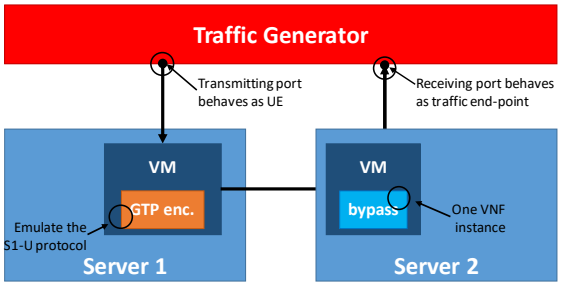



Fig. 4. Testbed configuration.

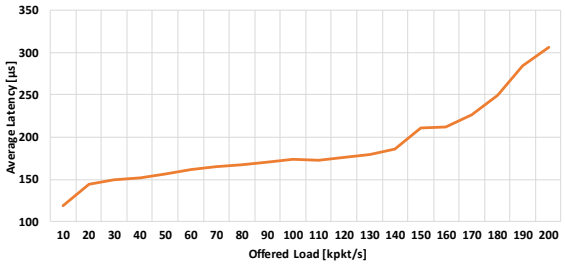



Fig. 5. Average latency of the Bypass VNF obtained for increasing offered load.

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.

The obtained performance is shown in Fig. 5. The latency ascribable to the VNF processing is sufficiently low to satisfy the delay budget for all services as per 3GPP Standardized QCI characteristics [31], considering a radio round-trip time of 20 ms [32] and an LTE backhaul delay of 20 ms [31]. If we consider the application to the 5G scenario, we can reduce radio and backhaul delays to 4 [33] and 1 ms [7], respectively. Considering standardized 5QI values from [7], 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 IV.B, the experimental measures in Fig. 5 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.

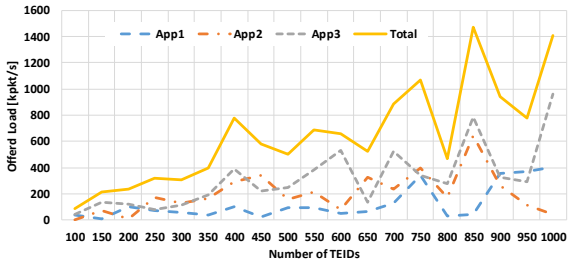



Fig. 6. Traffic generated to test the orchestrator.

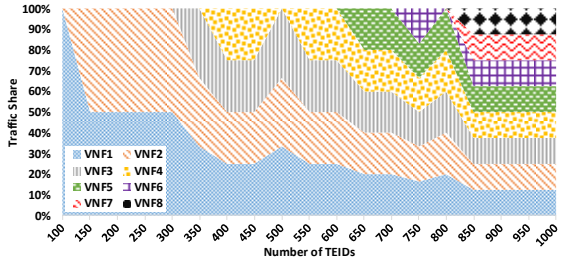



Fig. 7. Optimal number of Bypass instances and load shares according to increasing number of TEIDs for Test Case A.

TABLE I. REFERENCE APPLICATIONS USED IN THE FOLLOWING RESULTS

| Name | 3GPP Use Case | Packet Delay Budget [ms] |
|---|---|---|
| App1 | Discrete Automation | 10 |
| App2 | Real-Time Gaming | 50 |
| App3 | Conversational Voice | 100 |

This case corrsponds to the application of the decision policy in Section IV.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 [34]. Each bearer corresponds to one of the available applications, which have been selected among the 3GPP use cases to provide heterogeneous latency requirements as summarized in Table I. 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 [31] 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. 6.

Fig. 7 reports the traffic load shares among the number of VNF instances selected by the orchestrator in Case A. In this

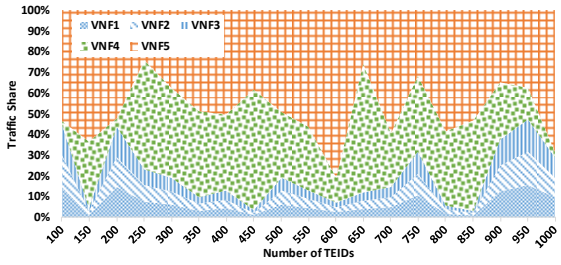



Fig. 8. Optimal number of Bypass instances and load shares according to increasing number of TEIDs for Test Case B.

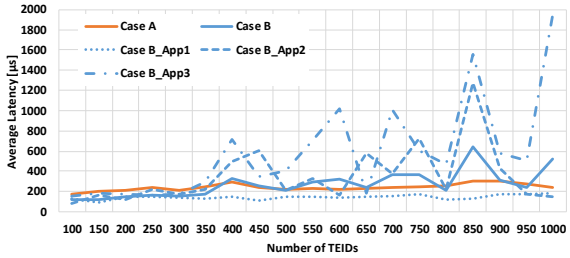



Fig. 9. Latency results for the two test cases.

case, the orchestrator adds a VNF instance when the available ones reach the value λ* of incoming load, which for this test case, for each active instance, equals 200 kpkt/s to satisfy the 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. 8. 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. 8 is performed among five VNFs throughout the whole test.

Finally, the average latency ascribable to the Bypass for the two cases is reported in Fig. 9. 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. 9) 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. 7 and Fig. 8, 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.

## VI. 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 goal, 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] P. Jamshidi, C. Pahl, N. C. Mendonça, J. Lewis, S. Tilkov, "Microservices: The Journey So Far and Challenges Ahead," IEEE Software, vol. 35, no. 3, 2018, pp. 24–35.

[2] 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

[3] 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.

[4] 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

[5] "Software-Defined Networking: The New Norm for Networks, Open Networking Foundation (ONF)," White Paper, Apr. 2012.

[6] "The Tactile Internet", ITU-T Technology Watch Report, August 2014.

[7] 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.

[8] 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/.

[9] G. Cattaneo, F. Giusti, C. Meani, D. Munaretto, and P. Paglierani, "Deploying cpu-intensive applications on mec in nfv systems: The immersive video use case," Computers, vol. 7, no. 4, p. 55, 2018.

[10] "MEC Deployments in 4G and Evolution Towards 5G", ETSI White Paper, February 2018.

[11] Cau, E., M. Corici, P. Bellavista, L. Foschini, G. Carella, A. Edmonds, and T.M. Bohnert, "Efficient exploitation of mobile edge computing for virtualized 5g in epc architectures", 4th IEEE International Conference on Mobile Cloud Computing, Services, and Engineering (MobileCloud), (March 2016), 100–109.

[12] B. P. Rimal, D. P. Van, M. Maier, "Mobile edge computing empowered fiber-wireless access networks in the 5G era", IEEE Commun. Mag., vol. 55, no. 2, pp. 192-200, Feb. 2017.

[13] R. Solozabal, A. Sanchoyerto, E. Atxutegi, B. Blanco, J.O. Fajardo, F. Liberal, "Exploitation of Mobile Edge Computing in 5G Distributed Mission-Critical Push-to-Talk Service Deployment", IEEE Access, vol. 6, pp. 37665-37675, 2018.

[14] J. Costa-Requena, A. Poutanen, S. Vural, G. Kamel, C. Clark, S. K. Roy, "SDN-Based UPF for Mobile Backhaul Network Slicing", European Conference on Networks and Communications (EuCNC), pp. 48-53, June 2018.

[15] S. Peters, M. A. Khan, "Anticipatory User Plane Management for 5G," IEEE 8th International Symposium on Cloud and Service Computing (SC2), Paris, France, 18-21 Nov. 2018.

[16] C. Ge, D. Lake, N. Wang, Y. Rahulan, R. Tafazolli, "Context-Aware Service Chaining Framework for Over-the-Top Applications", 5G Networks, First International Workshop on Intelligent Cloud Computing and Networking (ICCN 2019), Paris, France, May 2019.

[17] D. Das, S. Natarajan, N. Subburayalu, "Dynamic Load Balancing across Multi-radio Access Bearers in 5G," 11th International Conference on Communication Systems & Networks (COMSNETS), Bengaluru, India, jAN. 2019.

[18] S. Peng et al., "QoE-oriented mobile edge service management leveraging SDN and NFV", Mobile Inf. Syst., vol. 2017, Nov. 2017.

[19] M. C. Filippou, D. Sabella, and V. Riccobene, "FlexibleMEC service consumption through edge host zoning in 5G networks," arXiv preprint arXiv:1903.01794, 2019

[20] G. Carella, M. Pauls, T. Magedanz, M. Cilloni, P. Bellavista, L. Foschini, "Prototyping NFV-based Multi-access Edge Computing in 5G ready Networks with Open Baton", 3rd IEEE Conference on Network Softwarization (IEEE NetSofi 2017).

[21] "AMARI Callbox Series - The network on your desk", Online: https://www.amarisoft.com/products/test-measurements/amari-lte-callbox/.

[22] 3GPP, "Study on management and orchestration of network slicing for next generation network," TR 28.801, version 15.0.0, Sept. 2017.

[23] ETSI "Network Functions Virtualisation (NFV), Management and Orchestration," ETSI GS NFV-MAN 001 V1.1.1, URL: http://www.etsi.org/deliver/etsi_gs/NFV-MAN/001_099/001/01.01.01_60/gs_nfv-man001v010101p.pdf

[24] 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

[25] ETSI Group Report MEC 018, "End to End Mobility Aspects", version 1.1.1, October 2017.

[26] "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.

[27] 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_60/ts_136413v130600p.pdf.

[28] 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.

[29] TRIANGLE Project - 5G Applications and Devices Benchmarking, https://www.triangle-project.eu/.

[30] 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.

[31] 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.

[32] P. Synnergreen, T. Dudda, "LTE Latency Improvement Gains", Online: https://www.ericsson.com/en/blog/2014/11/lte-latency-improvement-gains.

[33] O. Teyeb, G. Wikström, 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

[34] Cisco Visual Networking Index: Forecast and Trends, 2017–2022 White Paper.