NFV-enabled Experimental Platform for 5G Tactile Internet Support in Industrial Environments

Prodromos-Vasileios Mekikis, Kostas Ramantas, Angelos Antonopoulos, Elli Kartsakli, Luis Sanabria-Russo, Jordi Serra, David Pubill, and Christos Verikoukis

***Abstract*—As industries are under pressure for shorter business and product lifecycles, there is an extensive effort from the research community for novel and profitable automation processes. This effort has given rise to the 5G Tactile Internet, which is characterized by extremely low latency communication in combination with high availability, reliability and security. In this paper, we discuss the key technologies to support the Tactile Internet characteristics in industrial environments and, then, we showcase the implementation of a novel 5G NFV-enabled experimental platform. Given that ultra-reliable low-latency communications is crucial for the manufacturing process, we demonstrate that, in our setup, sub-millisecond end-to-end communication is attainable, proving the suitability of our platform for tactile Internet industrial applications.**

***Index Terms*—Network function virtualization, 5G networks, Tactile Internet, Software-defined networking, IIoT, Industrial automation.**

I. Introduction

AROUND two centuries ago, the first industrial revolution brought significant changes on the manufacturing processes and influenced every aspect of the daily life. Since then, the unprecedented technological innovations and the increasing need for massive, reliable and rapid production have recently driven the rise of the Tactile Internet and the Industrial Internet of Things (IIoT) [1] [2]. Under these labels, various crucial technological advances have emerged in order to reach the key performance indicators (KPIs) set by the current trends of Industry 4.0 and the fifth generation (5G) networks ecosystem. For instance, the sensitivity of control circuits when controlling devices moving rapidly (such as industrial robots) requires an end-to-end latency significantly below 1 millisecond per sensor [3].

Employing technologies from 5G in industrial automation environments enables them to enhance their connectivity, latency, and bandwidth, while it is also possible to reduce the cost of their Information and communications technology (ICT) systems through a set of technologies under the umbrella of virtualization. To be more specific, network function virtualization (NFV) is a technique that can significantly benefit industries by optimizing their network services. It allows a hardware-free implementation of networks as it decouples several network functions from previously required network devices, such as firewalls, and runs them as software, i.e., virtual network functions (VNFs), at a data center. In this way, the NFV infrastructure does not only drop the deployment cost, as less equipment and installation personnel are needed, but it also reduces the service creation time from hours to minutes resulting in an extensively more efficient procedure [4].

To automate even further the networking procedures in the IIoT, software-defined networking (SDN) can be employed, which is a complementary approach to NFV that separates the control and forwarding planes to offer a centralized view of the network. Moreover, for the handling of the physical and virtual resources that support the network virtualization, an NFV management and orchestration (MANO) is responsible for the lifecycle management of the VNFs and it focuses on all virtualization-specific management tasks necessary in the NFV framework. To that end, a service chain of connected VNFs, i.e., a service function chain (SFC), can be created to automatically run a requested application based on the current traffic demand. This capability can be employed by industries to set up sets of connected VNFs that allow the use of a single network connection for many services that have different characteristics.

Although the set of aforementioned technologies can substantially improve the efficiency of the network in IIoT, there is still the obstacle of the proximity to the cloud. Since ultra-reliable low-latency communications (URLLC) are paramount for industrial environments, the network congestion might hinder the connection with the cloud. Therefore, multi-access edge computing (MEC) has been proposed to address this issue by establishing a cloud-based ICT service environment at the network edge [5]. Thus, real-time, high-bandwidth, low-latency access to radio network information becomes reality and improves application performance by achieving related task processing closer to the user.

During the last years, various NFV/SDN implementations have been demonstrated to prove the efficacy of the aforementioned technologies. In [6], the authors validate the simplification of the deployment process for future 5G networks. Furthermore, the authors in [7] present their NFV-enabled testbed that introduces intelligence, self-organizing, and autonomic capacities to 5G networks, providing an open environment to foster innovation and decrease the capital expenditure (CAPEX) and operational expenditure (OPEX) of new applications. Moreover, the authors in [8] present their experimental setup of a convergent 5G service scenario

This work has been funded by…

P.-V. Mekikis, K. Ramantas, and E. Kartsakli are with Iquadrat Informatica S.L., Barcelona, Spain, (email: {vmekikis,kramantas,ellik}@iquadrat.com)

A. Antonopoulos, L. Sanabria-Russo, J. Serra, D. Pubill, and Christos Verikoukis are with the Telecommunications Technological Centre of Catalonia (CTTC/CERCA), Spain, (e-mail: {aantonopoulos, luis.sanabria, jordi.serra, david.pubill, cveri}@cttc.es).

involving a MEC node to show the reconfigurability of their network in real-time based on the load. Although all these works present significant results for the future 5G networks, they do not focus on the demanding industrial requirements.

In this paper, we investigate the introduction and adaption of the tactile Internet in industrial environments using a set of SDN/NFV technologies. To evaluate the 5G capabilities under such scenarios, we employ the well-defined SEMIoTICS architecture[1] to build a 5G NFV-enabled experimental platform that consists of open-source software and extends the capabilities of current industrial KPIs by leveraging NFV, SDN and MEC technologies. To that end, the contribution of our work is threefold:

i) We provide details on the adopted SEMIoTICS architecture under industrial use cases and discuss how concepts, like NFV and SDN, affect IIoT networks in terms of reliability, latency, and cost.
ii) We adapt the aforementioned 5G architecture in our platform to put the industrial KPIs for URLLC and mMTC in the control loop.
iii) We evaluate the performance of our NFV-enabled platform to prove its suitability for tactile Internet industrial applications, while we provide useful insights for the deployment and operation of such platform under industrial environments.

The rest of the paper is organized as follows. Section II presents the 5G architecture that has to be adopted in industrial use cases and discusses in detail the NFV and SDN concepts and how they affect our architecture. The presentation of our open-source 5G testbed is given in Section III. The results regarding the quality of service for critical slices and the latencies from the allocation of the computing resources is presented in Section V. Finally, the paper concludes with Section VI.

II. SDN & NFV BASED IIOT NETWORKS – SEMIOTICS ARCHITECTURE

Within the Tactile Internet context, IIoT networks have the critical role of providing ultra-reliable and ultra-responsive network connectivity between IoT devices and the end-user applications. However, as traffic from different vertical applications traverses the network infrastructure, the availability of compute and network resources may vary, often producing negative effects on resource-demanding or delay-sensitive applications, such as haptic communications [9]. The 5G vision proposes a flexible network design, such that a single physical infrastructure is shared among different applications, while service requirements are guaranteed leveraging technologies such as SDN and NFV [10].

NFV and SDN are complementary technologies that achieve the level of abstraction and flexibility required to satisfy stringent applications' requirements while maximizing network infrastructure reutilization. Specifically, NFV decouples physical network functions (PNFs) (e.g.: firewalls, routers, load-balancers, etc.), from dedicated hardware by implementing the same functionality in software, coined virtual network functions (VNFs) [11]. VNFs may then be instantiated in data centers at backend clouds, or on top of devices equipped with compute and storage resources at the edge [12]. Its specifications could be modified according to requirements or load, and then decommissioned when no longer needed; freeing compute, network and storage resources for other VNFs. Furthermore, VNFs are not limited to network functionality, e.g. data processing in the form of one or many VNFs could be created closer to the sensor/actuator, yielding important latency reductions.

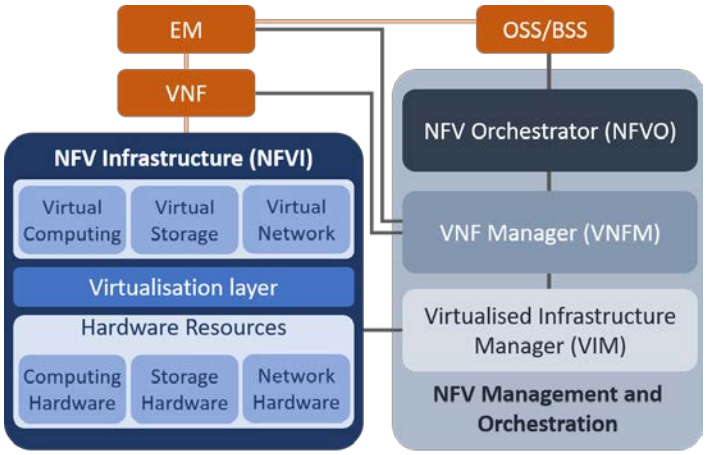

Figure 1 NFV Architecture [15]

SDN facilitates network management through a softwarization approach. Namely, it decouples the data plane from the control plane, centralizing network management in a so-called SDN controller. With a global view of the network resources, SDN controller applications can take advantage of the numerous southbound interfaces (e.g.: OpenFlow [13], NETCONF [14], among others) to gather network state information and act upon each forwarding device (i.e. PNF or VNF) configuration accordingly, e.g. by establishing data flow paths that guarantee certain quality of service (QoS) requirements. Together, SDN and NFV enable dynamic compute and network resources allocation for heterogeneous QoS requirements, which helps to circumvent the undesired effects of a changing network environment on sensible applications, such as haptic communications, Industry 4.0, autonomous driving, robotics, manufacturing, among others.

ETSI's efforts towards the standardization of the 5G vision has yielded the Network Functions Virtualization Architecture (NFVA) [11], which leverages the dynamism, flexibility, and reusability provided by SDN and NFV primitives. NFVA's several components handle the lifecycle and interconnection of VNFs in order to expose virtual Network Services (NS) to applications. ETSI's NVFA [15] is shown in Figure 1, including its main components: NFV Infrastructure (NFVI), Virtual Infrastructure Manager (VIM), VNF Manager, and the NFV Orchestrator (NFVO).

A. Virtual Infrastructure Manager (VIM)

Inside the NFVA, the VIM is responsible for the control and management of the interaction between VNFs and the NFVI

[1] SEMIoTICS webpage: https://www.semiotics-project.eu/

hardware resources, such as compute, storage and network, as well as their virtualization [11]. It takes care of exposing a pool of virtualized resources derived from the NFVI, as well as allocating such resources to VNFs.

VIMs also manage virtual network overlays to connect VNFs using SDN, but this task could also be left to an external SDN controller. For instance, state of the art VIMs such as OpenStack include a module that relays virtual network information to any compatible SDN Controller, such as OpenDaylight, through the so-called ML2 plugin [16] [17].

B. VNF Manager

It is responsible for VNF lifecycle management. That is, the instantiation, scaling, and termination of one or several VNFs. State of the art VIMs often include a service for VNF Management, like Tacker in OpenStack.

C. NFV Orchestrator (NFVO)

NS are often composed of several VNFs connected together in a predefined order, sometimes spanning more than a single VIM. This is called a Service Function Chain (SFC) or Virtual Network Function-Forwarding Graph (VNF-FG) within the NFVA framework (see Figure 2). Automated instantiation of VNF-FG's components (e.g.: constituent VNFs, virtual links, and allocation of storage) is carried out by the NFV Orchestrator, which is able to gather information about the NFVI from one or several VIMs through standardized reference points or APIs [15]. Moreover, information regarding the available VNFs (through a collection of catalogs on-boarded by the corresponding NFVI administrators), allocated resources, performance metrics about VNFs and virtual links, NFVI faults (outage) information, among others [18] could be used to monitor and update NS.

Thereby, NFVO works as an automation tool for instantiating and terminating NS from a centralized control position. Furthermore, it enables unprecedented infrastructure reutilization by allowing scaling out NS at runtime (e.g. for preserving KPIs), or freeing resources at low-demands periods for energy savings.

D. Satisfying stringent application requirements: SEMIoTICS Architecture

SDN/NFV allow sharing the physical network infrastructure among heterogeneous network services, i.e., with different QoS requirements, e.g. low-latency vs delay-tolerant, by isolating applications/resources using network slicing and Virtual Tenant Networks (VTN) [19]. Network Slicing was originally proposed by the Next Generation Mobile Network Alliance (NGMN) to ensure service isolation and offer performance guarantees to the tenants. These techniques greatly reduce the associated CAPEX/OPEX of creating a separate network deployment. Furthermore, coupled with NFVO's ability to monitor the entire NFVI, it is possible to satisfy stringent application requirements in a dynamic and flexible manner.

The SEMIoTICS architecture leverages NFV/SDN in a three-layer framework, namely Field, Network, and Backend/Cloud (see Figure 3 [20]). Each layer is composed of devices with different compute, storage, and network characteristics. Nodes with greater compute and storage resources are located at the Backend/Cloud layer, where computation-heavy NS, storage, VIM, NFVO, and other backend IIoT controllers are deployed.

Data transport from and to the Backend/Cloud must traverse the Network Layer. Such segment of the SEMIoTICS architecture usually hosts network VNFs (e.g. virtual switches, routers, firewalls, load-balancers, or the interconnection of these in the form of a NS), which could be tuned for satisfying each application requirement via the VIM/SDN Controller.

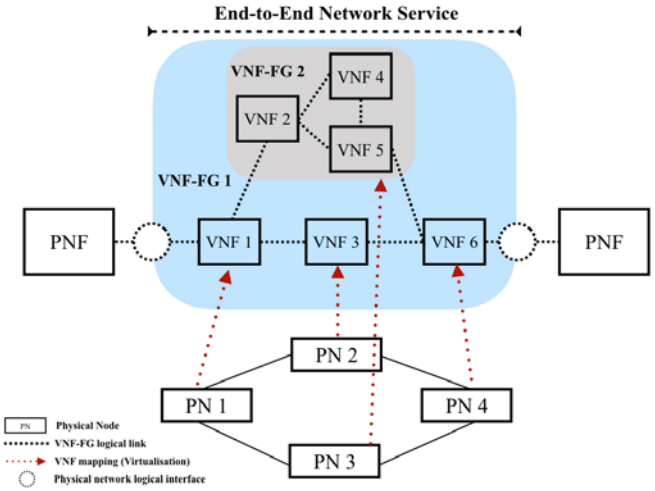

Figure 2 Example of an end-to-end Network Service with VNFs and nested VNF-FG [11]

Lastly, IoT/IIoT gateways occupy the Field layer. Such devices count with enough resources to provide embedded intelligence, data processing, or predictive capabilities to applications via VNFs; usually yielding important delay reductions due to their proximity to sensor/actuators.

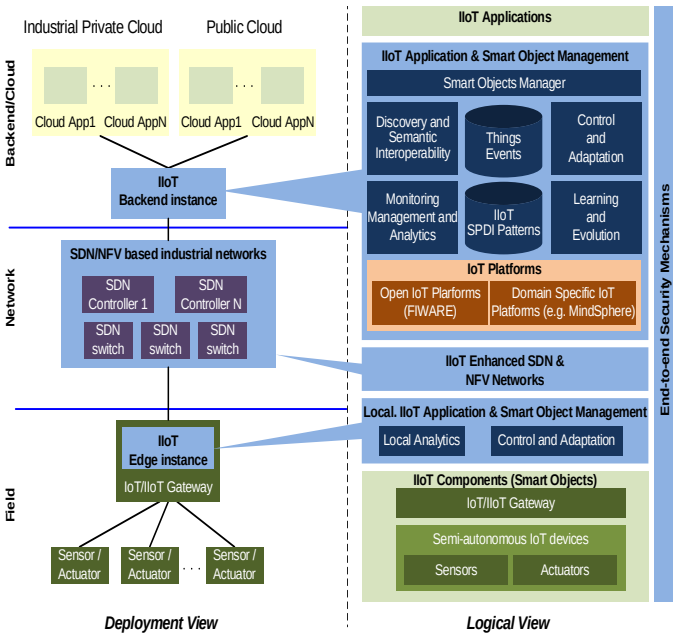

Figure 3 Envisaged architecture of SEMIoTICS framework [20]

Haptic communications are an example of extreme latency constraints, requiring a maximum roundtrip delay below 1ms. To leverage such delay limitations within the SEMIoTICS

architecture, NFV/SDN could be used to prioritize/isolate sensible traffic by creating network slices/VTN. Furthermore, predictive or other kind of data processing engines in the form of VNFs could be created closer to the sensor/actuators (Field layer in Figure 3), anticipating the user's actions and reducing the delay of the sensory feedback from the sensor/actuators, as suggested in [10].

All in all, SDN/NFV provide the much-needed flexibility to support heterogeneous application requirements. The softwarization of PNF into VNF, and the ability to create, monitor, update, and destroy such functions according to changing network conditions allows a single hardware infrastructure to be shared among different applications. Furthermore, the three-layer architecture proposed by SEMIoTICS not only provides the aforementioned abilities, but also allows the development of embedded intelligence at all layers, opening the way for different kinds of optimizations that should enable the next generation of applications.

III. Testbed Description

We have implemented an end-to-end SDN/NFV testbed, whose main focus is to enable secure and dependable smart sensing and actuation in IoT and IIoT application scenarios. Our testbed implements an end-to-end SDN/NFV architecture, complete with the local cloud, SDN networking and Field layers that demonstrate smart actuation, monitoring and analytics functionalities. Our testbed includes the following hardware, as shown in Figure 4.

- One 4-core 64-bit server with 16 GB RAM acts as the Controller, and hosts all services related to Management, Orchestration and SDN control.
- Two 6-core 64-bit servers with 32 GB RAM act as the Compute Nodes, or hypervisors, that hosts all IIoT services in dedicated VMs.
- Two Odroid C2 Single-Board Computers (SBCs) act as the Field layer Virtualized IoT gateway. An 802.15.4 radio module is employed to interconnect Field devices (smart sensors) with the gateway.
- Field layer smart sensors transmit temperature, humidity, and light intensity values wirelessly over 802.15.4. In a future work, the testbed will be extended with a 5G RAN from the OpenAirInterface project, so that NB-IoT support is also added.
- SDN access switches are employed in the Network layer, to interconnect the Compute Nodes and IIoT gateways.

IIoT services related to smart monitoring and actuation are implemented in the form of VNFs that can be automatically deployed and orchestrated by the cloud controller. Currently, we have implemented and deployed VNFs for smart monitoring and actuation (see Figure 5), each in a dedicated Tenant Network, that compete for resources. In what follows, the 3 individual layers of the IIoT testbed, i.e., Backend Cloud, Network and Field are presented in detail.

Figure 4 IIoT Testbed infrastructure, showing the Controller node, IoT gateways, smart sensor and actuators (smart lights)

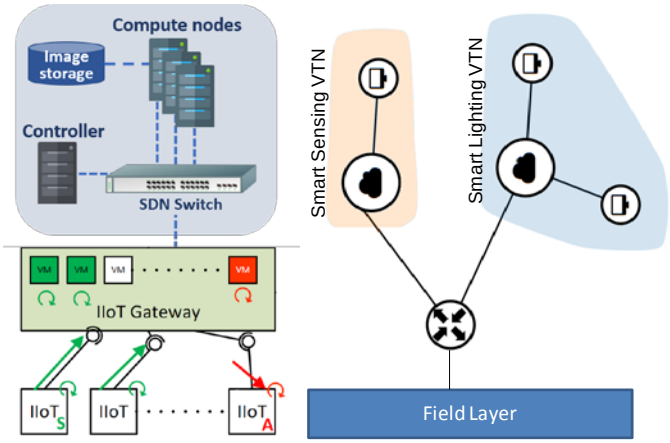

Figure 5 Testbed architecture and Virtual Tenant Networks (VTNs)

A. Backend Layer

The local cloud of our testbed, as seen in Figure 6, is based on the OpenStack ecosystem, which is responsible for deploying VMs and managing their lifecycle. Openstack is a complex software framework with multiple components that handle security and authentication, VM image storage, VM instantiation and termination, etc. In our testbed, a Controller node hosts all OpenStack services in Linux Containers. Linux Containers (LXD) is an emerging virtualization solution which allows services to run almost to the "bare metal" with minimal performance penalties, but with the requirement that they share the same kernel with the host (in this case the Controller node). The following OpenStack services are deployed in our Controller:

- **Glance** stores the VNF (or VM) images in its local filesystem
- **Keystone** acts as the identity service, keeping track of Openstack users and their respective permissions (e.g., admin, user, etc.)
- **MySQL** stores configuration options in a master database
- **Neutron** is the OpenStack networking layer, which handles connectivity among VMs and applications. It is responsible for deploying end-to-end slices and virtual networks among VNFs that can physically reside in different physical servers

- **Openstack-dashboard** implements the OpenStack Horizon GUI which allows us to manage our network and VMs with an easy to use GUI.
- **Tacker** serves as the VNF Manager, which handles the delivery of end-to-end network services. It supports the lifecycle management of network services, catalogue management and on-boarding/configuration of network services and VNFs.
- **Nova** is the OpenStack hypervisor service. Nova employs KVM (i.e., Kernel-based Virtual Machine) technology to natively execute multiple VMs at a host operating system.

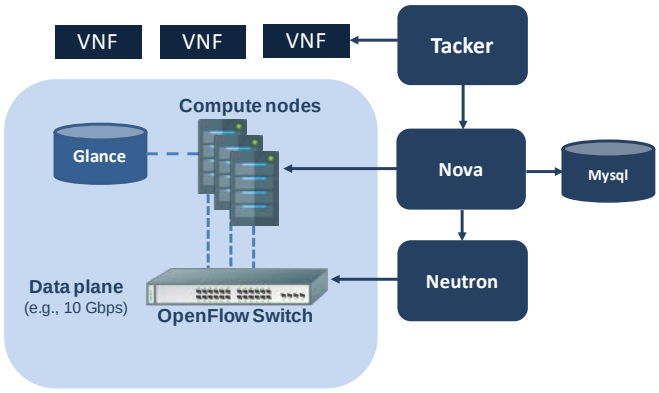

Figure 6 Backend Cloud

All services in our NFV enabled Testbed are packaged in VNFs that are hosted in dedicated VMs that are placed in Compute Nodes (or hypervisors) by OpenStack Tacker, i.e., the VNF Manager. We have currently deployed two VNFs, one for Smart Monitoring, denoted as VNF1, and one for Actuation, denoted as VNF2. Compute nodes are inter-connected by the data plane, which is implemented with SDN switches. VNF metadata are described by VNF Descriptors (VNFDs). VNFDs define service behavioral and deployment information in a template file which is based on TOSCA standards and is written in YAML. This allows deployment and orchestration of services to be performed automatically by OpenStack Tacker, which serves as the platform VNF Manager. OpenStack Tacker implements a Resource Orchestrator which coordinates the allocation and setup of the computing, storage and network resources that are necessary for the instantiation and interconnection of VNFs. Moreover, it performs Resource Checks to ensure that the VNF requirements are met. This allows the automatic deployment and lifecycle management of services, without user interaction. Moreover, VNFs can be individually scaled, i.e., multiple instances can be deployed to meet user demand. Moreover, the VNF Manager can migrate VMs to a different hypervisor for optimization purposes. For example, to meet service KPIs a VNF may have to be moved to a hypervisor with a lower load. VNF migration is a relatively complex procedure and care should be taken not to cause downtime. Specifically, there are two modes of operation for VNF migration:

- Legacy mode involves shutting down and then restarting the VM that hosts the VNF in a different hypervisor.
- Live migration mode involves running both instances (in the old and new hypervisor) in parallel while the migration is performed, and only migrating RAM contents as a final step. This mode causes minimal service disruption.

B. Networking Layer with Slicing support

As mentioned in the previous section, the testbed networking layer is based on Neutron. Neutron, Openstack ecosystem's SDN controller, is responsible for centrally controlling the virtualized network, as well as for deploying tenant networks to interconnect VMs and VNFs. Tenant networks represent isolated Layer 2 domains, and communication among them, as well as with external networks, is only possible via Layer 3 routing. Neutron consists of the following agents:

- **Neutron-server** accepts API requests from other OpenStack components and enforces the network model and IP addressing of each port.
- **Neutron-openvswitch-agent** provides Layer-2 connectivity to VMs that run in Compute Nodes. Moreover, it deploys virtual networks (or network slices).
- **Neutron-dhcp-agent** allocates DHCP IP addressing for tenant private networks.
- **Neutron-l3-agent** Implements a virtual Router (vRouter) which handles Layer3 routing among tenant networks.

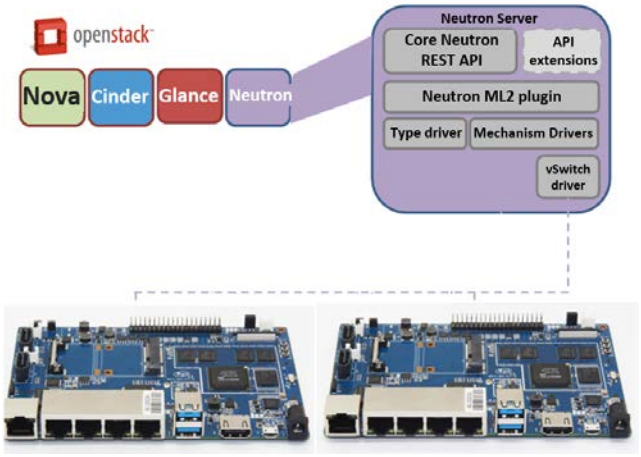

Figure 7 Testbed Networking Layer

Our testbed also employs virtual SDN switches for the Data Plane and the Access network, that interconnect Compute Nodes with the Field Layer via virtualized IIoT gateways. SDN switches are implemented with Open vSwitch (OvS), a production quality, multilayer virtual switch licensed under the open source Apache 2.0 license. A 4-port ARM-based platform is employed for the switch hardware, with Gigabit Ethernet ports. The access switches are also controlled by the Neutron controller via the OpenStack ML2 API (see Figure 7) which supports Open vSwitch out of the box. ML2 (Modular Layer 2) module bundled with OpenStack supports a wide variety of Layer 2 technologies. ML2 introduces the concept of drivers,

which are divided in type drivers and mechanism drivers as shown in Figure 7. The Neutron controller leverages the ML2 API to communicate QoS policies to the SDN switch. QoS rules are stored at the OvS database and applied to the OvS switch ports, forming the basis to implement slicing. The QoS model supported by Neutron and Open vSwitch, shown in Figure 7 includes three QoS rules that appropriately manage the network ports' priority queues:

- Differentiated Services Code Point (DSCP) marking of packets allows traffic prioritization
- Bandwidth limit prevents individual VNFs from saturating the network
- Minimum bandwidth guarantee reserves bandwidth

The aforementioned QoS policies can be applied to Tenant Networks via the centralized Neutron controller APIs. From the 3 QoS policies supported, bandwidth guarantee is the most critical for Industrial IoT networks that often need strict delay and throughput assurances (e.g., for infrastructure monitoring and smart actuation use cases). End-to-end slicing is implemented in our testbed by reserving bandwidth in all switch ports that lie across the path from an IIoT gateway to the VNF. Bandwidth reservation is performed via the Neutron QoS API. However, it must be noted that the OpenStack ecosystem is not able, in its current iteration, to offer strict end-to-end guarantees to applications. Specifically, the underlying infrastructure can't guarantee that hypervisor network interfaces will never be over-subscribed when scheduling new VMs. Hence, an additional verification and Live Migration support step was implemented in our testbed. Overall, service deployment involves the following steps:

1. The VNF image file is uploaded to Glance image storage
2. A VNFD file is supplied to the VNF Manager with service metadata and requirements.
3. The VNF Manager instantiates the VNF, which is automatically placed at a Data Centre hypervisor or at the IIoT gateway.
4. An end-to-end slice is deployed based on service requirements, using Neutron QoS APIs.
5. A verification step checks if the hypervisor interface was over-subscribed
6. If the verification fails, perform Live Migration of the VNF to a hypervisor with available resources and go to step 4.

Regarding Step 6, in a real-world scenario, multiple hypervisors may be available with enough resources to host the VNF. One of many algorithms proposed in the literature, e.g., First Fit, Best Fit, or Worst Fit, could be applied.

C. *Field Layer*

Our testbed Field layer includes a virtualized IIoT gateway, shown in Figure 8, that interconnects a set of smart sensors and smart light actuators with the backend cloud. Our IoT gateway supports KVM virtualization, enabling us to push VNFs down to the gateway tier. As a result, we can have a MEC node that allows services with ultra-low latency requirements to be pushed to the edge, hence minimizing latency. The relatively modest resources available at the gateway, which is implemented with a 64-bit ARM-based Single-Board Computer, means that it must be used for a minimum number of VNFs with low processing needs.

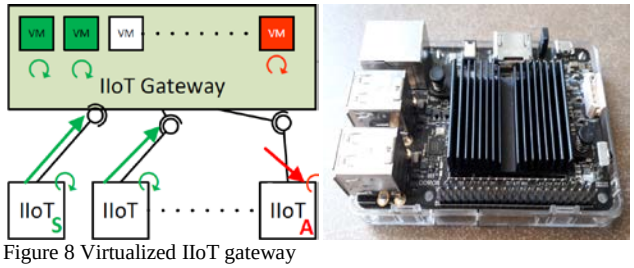

Figure 8 Virtualized IIoT gateway

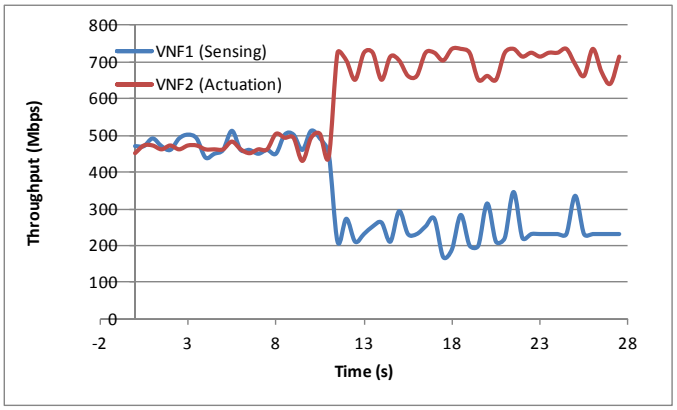

Figure 9 Throughput measurement vs. time for VNF1, VNF2

For the field-layer smart sensors, we employ custom-designed battery operated 802.15.4 and BLE devices that perform periodic measurement of CO2, Temperature and Light (Lux) values. Sensor values are encapsulated in IPv6 packets and transmitted to the IIoT gateway via MQTT. The actuators are commercial Philips Hue Smart Lights that are connected to the IIoT gateway via a Hue bridge. The Sensors and Actuators are communicating with the respective VNFs, that are hosted at the Cloud or IIoT gateway hypervisors.

IV. Experimental Results

In this section, the IIoT testbed is evaluated in terms of its ability to guarantee bandwidth reservations in Tenant Networks with slicing, as well as the effectiveness of Live Migration in optimizing VM placement. Finally, the suitability of a virtualized IIoT gateway, which is capable of hosting VNFs, for industrial and haptic applications is also evaluated. In all our experiments, the traffic was generated with the D-ITG traffic generator [21], which is able to generate TCP traffic with various profiles, e.g., Pareto, Exponential, etc., as well as write trace files. Moreover, a Smart Sensing and an Actuation VNF were deployed, each in a dedicated Tenant Network (see Figure 3), that compete for testbed resources.

A. *Tenant Network Slicing*

In this experiment, we measured the maximum throughput that could be sustained between the two VNFs, both hosted at the Backend Cloud, and a client device which was connected at the Field layer. At first, the link capacity, which is 1 Gbps, is

equally shared by the two VNFs, as shown in Figure 9. At time t=11s the Neutron API is employed to setup an end-to-end Network Slice for VNF2, with a dedicated throughput of 700 Mbps. Figure 9 shows that the measured throughput of both VNFs changes instantaneously to 700 Mbps for VNF2 and 300 Mbps for VNF1. This was achieved with successful bandwidth reservation at the hypervisor network interface, as well as at the SDN switch output port where the client device is connected.

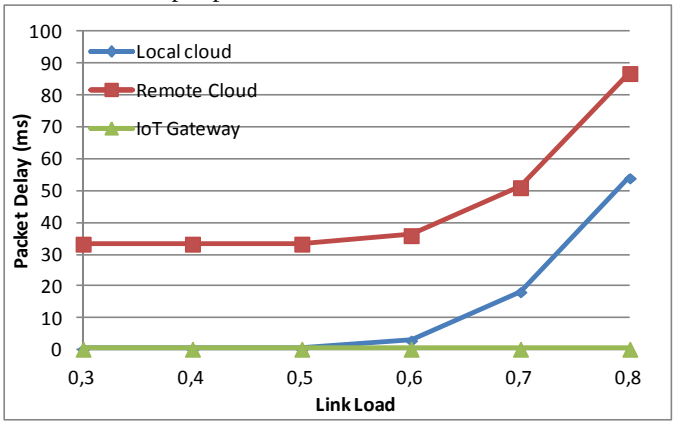

Figure 10 Packet delay vs. Load for different VNF placement options

B. *VNF packet delay*

In terms of resource usage, slicing is a relatively inefficient solution, as it reduces statistical multiplexing gains and therefore wastes capacity, hence it is often reserved only for the most critical services. An alternative solution to afford low latencies to delay-sensitive services is to place them directly at the IIoT gateway. This way, they bypass the Network Layer and its potential bottleneck, and can directly communicate with Field Layer devices. In the following experiment, the Round-Trip Time (RTT) of packets transmitted from the actuation VNF to the Hue bridge is measured, when it is placed at the backend cloud, or directly at the virtualized IIoT gateway. The RTT of the local cloud is also compared to the cloud service provided by the smart light vendor. In both cases background traffic with an Exponential traffic profile is also generated, with a Load that varies from 0 (no background traffic) to 0.8 (severe congestion). The link load, or link utilization, refers to the ratio of the link throughput versus the link capacity, i.e., the proportion of the link capacity used for packet transmission. The measured packet delay of the actuation VNF, when hosted at the Local or Remote cloud or at the Gateway is plotted in *Figure 10*. We conclude that sub-millisecond latencies are achievable for services hosted directly at the IIoT Gateway, which are unaffected by network congestion. Therefore, given that URLLC is crucial for the manufacturing process, we show that our platform can attain sub-millisecond end-to-end communication, proving the suitability of our platform for tactile internet industrial applications. This is also possible for local cloud services, as long as the link load is less than 0.5, which can be achieved with dedicated slices. However, as shown in *Figure 10*, even when slicing is employed, queueing delay of Exponential traffic increases noticeably when input load exceeds 50%. Hence, a dedicated slice typically uses up twice the bandwidth required on average and is therefore considered an expensive solution. Finally, Remote Cloud solutions should be avoided for delay sensitive services, as they are subject to significantly higher latencies.

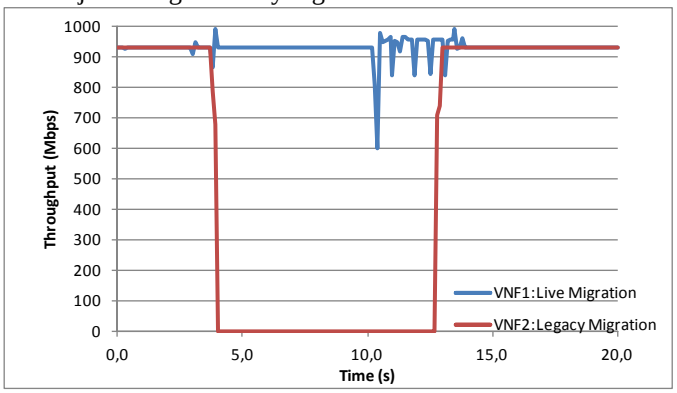

Figure 11 Throughput vs. time for Live and Legacy migration

C. *VM Migration*

In our last experiment, we explore whether VM migration is an efficient mechanism for the optimal placement of VNFs. Specifically, we test the service disruption caused when VMs are migrated to a different hypervisor at the backend cloud. Figure 11 shows how the throughput measurement of the two VNFs in 0.1 second intervals, when measured from a Field layer client device. The migration time was found comparable is both cases, as in our testbed it is dominated by the copying of Virtual Hard Disk of the VMs. However, in the case of Legacy migration a service disruption of around 8.5 seconds was measured, while services and TCP connections would terminate and need to be restarted. [2] On the other hand, Live Migration caused no service disruption and was only noticeable by a small drop in the measured throughput, which dropped by 40% for a duration of less than 0.5 seconds. Towards the end of the memory copy, the instance is paused for a short time (typically around 50 milliseconds) so that the remaining few memory pages can be copied to the destination VM without interference from the source instance memory writes. For zero-downtime migration, OpenStack offers the more advanced Auto-convergence and Post-copy options.

V. Conclusion

The need for shorter business and product lifecycles has urged manufacturing companies to explore novel production processes that leverage 5G technologies. Towards this direction, the Tactile Internet define the requirements for the much-needed reliability, low latency and availability in industrial automation. In this paper, we investigated the capabilities of NFV and SDN to satisfy these requirements through an NFV-enabled Experimental Platform that follows the SEMIoTICS framework. In our results, we experimentally proved that sub-millisecond latencies are achievable for services hosted directly at the IIoT Gateway, which are unaffected by network congestion. To that end, it is evident that our approach can satisfy the demanding industrial needs. In our future work, we plan to include an NFV orchestrator in our platform, as well as a high-density field layer. In this way, it will be possible to experiment with the scaling-in/out capabilities of our system that would enable a more robust operation in an industrial environment.

REFERENCES

[1] G. P. Fettweis, "The Tactile Internet," *IEEE Vehicular Technology Magazine*, vol. 9, no. 1, pp. 64-70, 2014.

[2] M. Aazam, S. Zeadally and K. A. Harras, "Deploying Fog Computing in Industrial Internet of Things and Industry 4.0," *IEEE Transactions on Industrial Informatics* , vol. 14, no. 10, pp. 4674 - 4682, 2018.

[3] I. T. U. ITU-T, "The Tactile Internet," *ITU-T Technology Watch Report*, 2014.

[4] E. Chirivella-Perez, R. M. Alaez, J. M. A. Calero, Q. Wang and J. Gutierrez-Aguado, "UWSIO: Towards automatic orchestration for the deployment of 5G monitoring services from bare metal," in *2018 IEEE WCNC*, Barcelona, 2018.

[5] ETSI, "ETSI.org: MEC in 5G Networks," June 2018. [Online]. Available: https://www.etsi.org/images/files/ETSIWhitePapers/etsi_wp28_mec_in_5G_FINAL.pdf. [Accessed October 2018].

[6] L. Cominardi, C. J. Bernardos, P. Serrano, A. Banchs and A. de la Oliva, "Experimental evaluation of SDN-based service provisioning in mobile networks," *Computer Stand. & Interfaces*, vol. 58, pp. 158-166, 2018.

[7] P. Neves et. al., "The SELFNET approach for autonomic management in an NFV/SDN networking paradigm," *International Journal of Distributed Sensor Networks*, vol. 12, no. 2, p. 2897479, 2016.

[8] S. Fichera, M. Gharbaoui, P. Castoldi, B. Martini and A. Manzalini, "On experimenting 5G: Testbed set-up for SDN orchestration across network cloud and IoT domains," in *2017 IEEE Conference on Network Softwarization (NetSoft)*, Bologna, 2017.

[9] A. Aijaz, M. Dohler, A. H. Aghvami, V. Friderikos and M. Frodigh, "Realizing the Tactile Internet: Haptic Communications over Next Generation 5G Cellular Networks," *IEEE Wireless Communications*, vol. 24, no. 2, pp. 82-89, 2017.

[10] M. Simsek, A. Aijaz, M. Dohler, J. Sachs and G. Fettweis, "5G-Enabled Tactile Internet," *IEEE Journal on Selected Areas in Communications*, vol. 34, no. 3, pp. 460-473, 2016.

[11] ETSI, "ETSI.org: Network Functions Virtualisation (NFV); Architectural Framework," 10 2013. [Online]. Available: https://www.etsi.org/deliver/etsi_gs/nfv/001_099/002/01.01.01_60/gs_nfv002v010101p.pdf. [Accessed 23 October 2018].

[12] T. Taleb, K. Samdanis, B. Mada, H. Flinck, S. Dutta and D. Sabella, "On Multi-Access Edge Computing: A Survey of the Emerging 5G Network Edge Cloud Architecture and Orchestration," *IEEE Communications Surveys Tutorials*, vol. 19, no. 3, pp. 1657-1681, 2017.

[13] N. McKeown et. al, "OpenFlow: enabling innovation in campus networks," *ACM SIGCOMM Computer Communication Review*, vol. 38, no. 2, pp. 69-74, 2008.

[14] R. Enns, M. Bjorklund, J. Schoenwaelder and A. Bierman, "RFC 6241: Network Configuration Protocol (NETCONF)," June 2011. [Online]. Available: http://www.rfc-editor.org/info/rfc6241. [Accessed 23 October 2018].

[15] ETSI, "ETSI.org: Network Functions Virtualisation," 2018. [Online]. Available: https://www.etsi.org/technologies-clusters/technologies/nfv. [Accessed 2018 October 24].

[16] J. Denton, Learning OpenStack Networking (Neutron) Second Edition, Birmingham, UK: Packt Publishing Ltd., 2015.

[17] R. Toghraee, Learning OpenDaylight: The art of deploying successful networks, Birmingham: Packt Publishing Ltd., 2017.

[18] ETSI, "ETSI.org: Network Functions Virtualisation (NFV) Release 2; Management and Orchestration; Functional requirements specification," 1 February 2018. [Online]. Available: https://standards.globalspec.com/std/10274130/gs-nfv-ifa-010. [Accessed 2018 October 24].

[19] I. Afolabi, T. Taleb, K. Samdanis, A. Ksentini and H. Flinck, "Network Slicing and Softwarization: A Survey on Principles, Enabling Technologies, and Solutions," *IEEE Communications Surveys & Tutorials*, vol. 20, no. 3, pp. 2429-2453, 2018.

[20] SEMIoTICS, "SEMIoTICS: Smart End-to-end Massive IoT Interoperability, Connectivity and Security," [Online]. Available: https://www.semiotics-project.eu/.

[21] A. Botta, A. Dainotti and A. Pescapé, "A tool for the generation of realistic network workload for emerging networking scenarios," *Computer Networks*, vol. 56, no. 15, pp. 3531-3547, 2012.

Dr.Christos Verikoukis is currently a Fellow Researcher at CTTC (Head of the SMARTECH department) and an adjunct professor at Barcelona University (Electronics Department). He has published 122 journal papers (h-index 31) and over 180 conference papers. He is also co-author in 3 books, 16 chapters in different books and he filled 3 patents. He has supervised 15 Ph.D. students and 5 Post Docs researchers since 2004.

Dr Angelos Antonopoulos received his Ph.D. degree from the Technical University of Catalonia (UPC) in 2012. He is a Researcher with CTTC/CERCA. He has authored over 80 peer-reviewed publications (h-index: 19) on various topics, including 5G wireless communications, network virtualization, energy efficient network planning and network economics.

Dr Luis Sanabria-Russo received his M.Sc. and PhD degrees in Information and Communications Technologies from Universitat Pompeu Fabra, Barcelona, Spain in 2016. His research interests are in SDN/NFV strategies for IoT, specifically, leveraging the radio heterogeneity and cloud technologies that accompany the future 5th generation (5G) of wireless systems.

Dr Jordi Serra received an Electrical Engineering degree from Universitat Politècnica de Catalunya (UPC), Barcelona, Spain, in 2006. He has been a Research Engineer at CTTC since April 2007, where he has worked in several research projects dealing with MIMO applicability to satellite networks, multicarrier modulations for LTE systems.

David Pubill joined CTTC as Research Engineer in August 2006 in the area of Access Technologies, and worked in several national and international projects based on WiMAX His main research interests are in smart grid, energy efficiency, smart metering, smart building/home, Internet of Things and Wireless Sensor Networks (WSN).

Dr Elli Kartsakli received her Ph.D. in Wireless Telecommunications from the Technical University of Catalonia (UPC) in February 2012. Her primary research interests include cross-layer medium access control (MAC) layer optimization for multiuser and cooperative schemes, energy-efficient sensor networking and M2M communications, SDN and cloud-based architectures.

Dr Kostas Ramantas has received the Diploma of Computer Engineering, the MSc degree in Computer Science and Engineering and the PhD degree from the University of Patras, Greece. His research interests are in modelling and simulation of network protocols, and scheduling algorithms for QoS provisioning.

Dr Prodromos-Vasileios Mekikis has received his PhD degree from the Department of Signal theory and Communications of the Technical University of Catalonia (UPC), Spain, in 2017. His main research interests include Network Function Virtualization, Wireless Energy Harvesting and connectivity in massive IoT networks.