

# E2E Network Slice Management Framework for 5G Multi-tenant Networks

Enrique Chirivella-Perez, Pablo Salva-Garcia, Ignacio Sanchez-Navarro, Jose M. Alcaraz-Calero, and Qi Wang

**Abstract**—Network slicing is one of the cornerstone features of the fifth generation (5G) networks to enable the multiplexing of virtualised logical networks over the same physical network infrastructure for various vertical business services. Architectures based on network slicing enable network operators to offer end-to-end (E2E) vertical services whilst assuring fulfilling tailored service level agreement (SLA) requirements. Despite the numerous benefits that this concept offers, the need for efficient, scalable and holistic E2E management of network slicing significantly complicates network operators' network management and imposes significant challenges in the quality of service (QoS). This paper proposes a novel intent-based slice manager framework over a service base 5G architecture that allows flexible definitions to customise network slices, automates all essential management tasks for 5G network slice providers, manages the life cycles of all sorts of network slices, and guarantees their QoS in a unified network slice management framework. The proposed framework is empirically validated in a realistic large-scale 5G multi-tenant infrastructure. The high scalability of the framework is based on the empirical results supporting more than 512 physical machines, 534288 virtual machines and more than two million network interfaces.

**Index Terms**—Automation, end-to-end network management, network slicing, 5G networks.

## I. INTRODUCTION

NETWORK slicing has emerged as a fundamental feature for the 5G success [1]. The ability to decouple resources into logical/virtual networks to address use cases with different needs realises the 5G promise of flexibility and optimal resource efficiency. In this sense, network slicing enables accommodating diverging quality of service (QoS) requirements, which can be categorised into massive machine-type communication (mMTC), ultra-reliable low-latency communication (URLLC) and enhanced mobile broadband (eMBB) content delivery [2], over a single physical network infrastructure. Moreover, multi-tenancy allows several operators or service providers to leverage the same physical infrastructure to reduce both capital

Manuscript received February 8, 2022 revised November 10, 2022; approved for publication by Chan-Byoung Chae, Division 3 Editor, April 17, 2023.

This work is funded by the European Commission Horizon 2020 5G-PPP Program under Grant Agreement Number H2020-ICT-2020-2/101017226 “6G BRAINS: Bringing Reinforcement learning Into Radio Light Network for Massive Connections” and under Grant Agreement Number H2020-ICT-2020-2/101016941 “5G INDUCE: Open Cooperative 5G Experimentation Platforms for The Industrial Sector NetApps.”

The authors are with the University of the West of Scotland, United Kingdom, email: echirivella@gmail.com, {Pablo.Salva-Garcia, Ignacio.Sanchez, Jose.Alcaraz-Calero, qi.wang}@uws.ac.uk

Enrique Chirivella-Perez Echirivella is the corresponding author.  
Digital Object Identifier: 10.23919/JCN.2023.000019

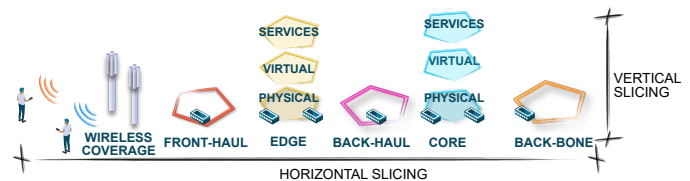


Fig. 1. Service base 5G architecture segments overview.

and operational expenditures [3] (CAPEX and OPEX, respectively).

Fig. 1 outlines a service-based 5G architecture, which contains the essential 5G services, components and network segments that may be involved in an end-to-end (E2E) network slice. It is noted that, from this point on, this research work will refer to the concept of horizontal and vertical slicing to differentiate between two sections of the same slice. We define the horizontal slice as a section of the slice that physically connects different network segments (e.g., fronthaul, edge, midhaul, core, and backhaul) and the vertical slice as an area involving the virtual and service layer of each network segment, respectively.

Although promising, 5G network slicing imposes several challenges. Firstly, available network resources have to be monitored to ensure that each slice provides the agreed QoS and E2E network performance. Secondly, network functions have to be defined and exposed to enable network slice creation. Thirdly, virtualisation and containerisation technologies must be in place to create logical/virtual machines and networks to enable a scalable slice deployment. Fourthly, dynamic orchestration of the different components of the network that are involved in the life-cycle of each slice is required so that these components can be optimally created on-demand and maintained at runtime [4]. Furthermore, to guarantee the agreed QoS at any point of the 5G architecture of an E2E slice (regardless of whether it is horizontal or vertical), it is paramount to enable access to all the different programmable network elements in the data plane including the wireless network segments. As opposed to its predecessors (3G/4G), 5G allows the partitioning of a physical network into independent virtual mobile networks which enables the multi-tenancy paradigm. However, this implies a new challenge in network traffic processing. Network packets now require a deeper level of encapsulation to allow differentiation in the transmissions between tenants and services. Although nowadays the data plane can be programmed using software-define networking (SDN) techniques [5],

Creative Commons Attribution-NonCommercial (CC BY-NC).

This is an Open Access article distributed under the terms of Creative Commons Attribution Non-Commercial License (<http://creativecommons.org/licenses/by-nc/3.0>) which permits unrestricted non-commercial use, distribution, and reproduction in any medium, provided that the original work is properly cited.

the final networking devices usually lack support for nested encapsulated 5G traffic packets processing [6]. Moreover, they are not interconnected and do not allow their real-time coordination during the life-cycle of a network slice including provisioning, instantiation/configuration, association, modification, migration, decommissioning and so on.

Our major contribution is the design and implementation of a novel E2E network slice management framework capable of managing the complete life-cycle of network slices throughout all network segments of a 5G infrastructure. We consider both horizontal and vertical slicing (depicted in Fig. 1) to successfully guarantee the QoS performance of an E2E slice at any point of the data plane. Our solution covers not only a static approach to network management but also a dynamic approach where E2E network slices instances (NSI) may migrate in real time over the dynamic environment of an operational network. The framework has been designed, prototyped and validated over a multi-tenant 5G network infrastructure, and optimised to enable coexisting vertical businesses to share the same physical resources, whilst fulfilling their own respective network slice requirements.

The proposed E2E network slice management framework provides the following contributions to knowledge beyond the state of the art:

- 1) The proposed framework allows the creation of tailored network slice templates (NSTs) that are defined in terms of network parameters such as data speed, quality, latency and reliability. It also accepts real-time modifications (add, remove or modify values) of those parameters, adding flexibility to management operations.
- 2) It enables the association of different services, users and applications with a specific NST, automatically calculating all the modifications through the data plane of the E2E slice that must be carried out during the NSI's deployment time. Thus, it reduces the complexity of network administration tasks.
- 3) It implements a novel network interface discovery technique that allows the framework to be aware of the involved interfaces that will be part of the E2E slice data path even before the virtual instances are up and running. This allows for a significant reduction in the NSI deployment time since the data plane can be programmed during the startup time of the virtual architecture.
- 4) It manages the complete life-cycle of network slice instances, including provisioning, association/disassociation, instantiation, and decommissioning (termination). Thus, it provides a unified monitoring point to determine the current state of each NSI.
- 5) It allows the explicit definition of geographic areas where the network slice should be available, empowering business models based on a flexible definition of service coverage.
- 6) It automates all the virtual functions for a coherent E2E NSI deployment in the data plane. It programs all the required network devices depending on both their programmable capabilities and their locations

(horizontal/vertical areas) through the E2E data path of the NSI. This includes programmable networking devices spread across the wired segments (e.g., physical/virtual interfaces, or software switches) but also those that are on wireless network segments. This fills the current gap in enabling the interconnection and coordination of the heterogeneous programmable network devices that may coexist in the data plane of any network segment.

- 7) It supports user device mobility enabling continuous availability of network slices whilst the user moves between different radio access networks. If necessary, existing network segment instances will be automatically migrated so that end users do not notice a degradation in service performance, thereby avoiding breaching service agreements.

The remainder of this paper is structured as follows. Section II presents related work, Section III provides detailed information about the slice manager in a large scale 5G multi-tenant architecture, Section IV provides a detailed architectural domain in the proposed framework, Section V provides detailed information about life-cycle and sequence diagrams, Section VI provides all the implementation details, Section VII provides a set of experiments validating the slice manager framework, describes the functional validation of the prototype testbed implementation, Section VIII provides the conclusions.

## II. RELATED WORK

There has been a growing interest to rethink development strategies around the challenge of dividing physical networks into multiple logical networks inside the same infrastructure. The network sharing paradigm was predicted years ago to be one of the main enablers for new business opportunities while decreasing capital and operational expenditure (CAPEX/COPEX) [7]–[9]. Passive network sharing was introduced in the early 2000s in Release 99 for the universal mobile telecommunications system (UMTS), later followed by 3GPP Rel-6 (UMTS), Rel-8 (LTE), and Rel-10 (LTE-A), which impose new requirements on the potential of network sharing. As network sharing requirements grew, the standardization bodies such as 3GPP, European telecommunications standards institute (ETSI) and others, had to accommodate their architectures and management extensions to pave the way towards more flexible, scalable and dynamic implementations with virtualization technologies [10]. The evolution from the network sharing paradigm to the first architectures demanding network slicing capabilities was presented in [11] where the authors provided an introduction to the concept of the network slice broker based on the 3GPP network sharing management architectures. Network sharing based on long-term contractual agreements was depicted in [12] where the 3GPP telecom management working group extended their architecture to include mobile virtual operators (MVO) into the network sharing paradigm, thereby paving the way to advanced mobile network slicing with multi-tenancy implications.

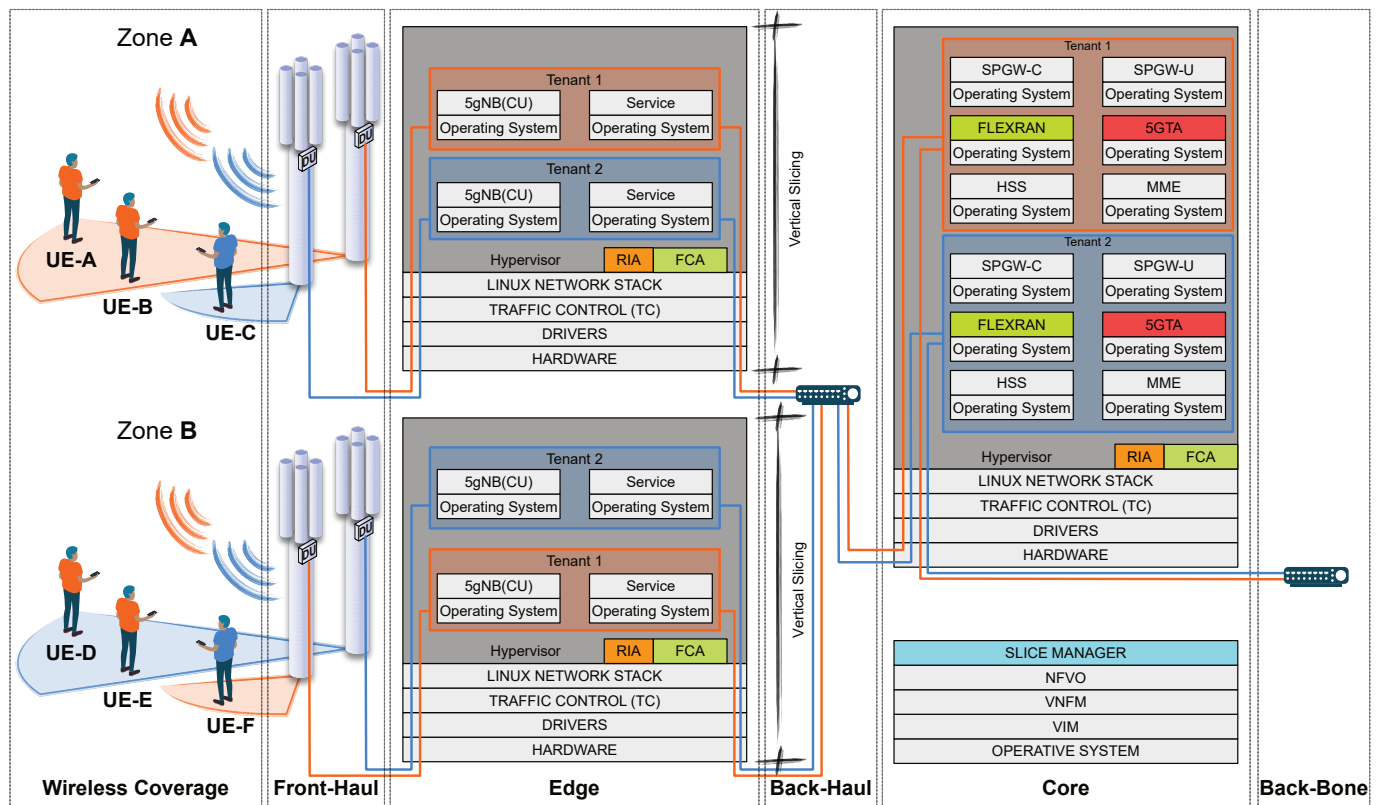


Fig. 2. 5G multi-tenant architecture extended with the proposed slice manager framework.

Furthermore, standardisation on network slicing has been gaining momentum. In particular, 3GPP TS 28.530 introduces the network slice concepts, use cases and requirements [13] and TS 28.531 defines the provisioning of network slicing for 5G networks and services [14], and TR 28.801 [15] makes the overall recommendations on management and orchestration on network slicing. It is noted that the proposed design in this paper is compliant with the relevant recommendations.

With the advent of 5G networks and looking toward future pre-6G expectations with the requirements of all evolutions [16], many projects and studies have dived into this field to tackle the ever-increasing traffic demand, which is pushing network operators to find new cost-efficient solutions while guaranteeing an agreed performance of some specific services. Network slicing has now become a powerful approach to ensuring certain levels of well-performing services. Nowadays, there is a lot of research and developments in the state of the art that leverage network slicing to guarantee QoS levels among heterogeneous use cases such as the Internet of things (IoT) [17]–[19], smart grids [20], [21], smart cities [22], eHealth (e.g., telemedicine, critical services) [23], [24], or intelligent transport, education and media and entertainment [25], among others.

Despite the considerable amount of research activities on network slicing to maintain or optimise the quality of the various services offered by vertical businesses, the management, orchestration and overall automation of the slice life-cycle needs to be further addressed. In Europe, there are several related projects from the

Horizon 2020 5G public-private partnership (5G-PPP). For instance, 5GTango [26] proposed an architectural design with network slicing capabilities [27], and SliceNet [28] developed an E2E cognitive network slicing and slice management framework [29], leading to considerable progress in this area. Moreover, global 5G initiatives such as 5G Americas [30] and IMT-2020 (5G) in China [31] have also presented their network slicing management and orchestration frameworks [32] based on the definitions of the main standardization bodies and/or industrial alliances such as the MGMN in [33], 3GPP in Rel-13 or the ETSI-OSM in [34], [35]. While projects are defining their own slicing framework architectures, there is still a lack of empirical results concerning the cost and possible side effects that the whole system may experience, especially in challenging operational situations with high service demands.

This related work also provides a comparative table with projects/research works related to E2E network slicing solutions. The table is divided into horizontal and vertical slicing with the segments involved in each of them. The horizontal slicing section contains all the different segments that are involved in such a slicing process i.e when a cross appears in the RAN means that the reference does not have that capability. The vertical slicing section involved all the different aspects regarding virtualisation and containers. This table also contains the capabilities of mixing different technologies across the different aspects of the infrastructure. Overall this table summarises as a glance at horizontal and vertical slicing making into account all the abstraction layers.

TABLE I  
COMPARISON EXISTING E2E NETWORK SLICE SOLUTIONS.

	Horizontal slicing						Vertical slicing					
	RAN	Front-haul	Edge	Back-haul	Core	Back-bone	Containers	Virtual machine	PNF	Container + service	Virtual machine + service	5G User
[36]	X	X	✓	X	✓	X	✓	✓	X	X	X	X
[37]	✓	X	✓	✓	✓	X	✓	✓	X	X	X	X
[38]	✓	✓	✓	✓	✓	✓	✓	✓	X	X	X	X
[39]	✓	X	✓	✓	✓	X	✓	✓	X	X	X	X
[40]	✓	X	✓	✓	✓	X	✓	✓	✓	X	X	X
[41]	X	✓	X	✓	X	✓	✓	✓	X	X	X	X
Ours	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓

The rows in Table I provide a cross or check in each of the columns where the E2E slice is addressed.

Similarly, has not been yet measured the recovery time when numerous mobile users change their locations and thus the involved NSIs (and/or the user attachment to those slices) have to be updated accordingly. This paper instruments the whole network slice life-cycle, taking into account different stressful scenarios and states of the 5G network with the aim to provide relevant empirical results. To realise this highly ambitious vision, an innovative slice management framework has been integrated to manage and orchestrate network slicing tasks automatically.

III. 5G MULTI-TENANT ARCHITECTURE

This section outlines the large-scale 5G multi-tenant architecture that we have set up on our premises and showcases how our proposed slice manager framework fits in. It also introduces its main components and sets them in place. Fig. 2 focuses on clarifying the different network segments of this service based architecture (SBA), its physical/virtual components and the different domains per tenant. In addition, it depicts an example of the data-plane path for an E2E network slicing use case.

The architecture presented here can be divided into two different parts, namely, management and managed parts respectively. Firstly, the management part consists of the already standardised management and orchestration (MANO) architecture, a key element of the ETSI network function virtualisation (NFV) [34], [42]. MANO is mainly composed of the virtual infrastructure manager (VIM), in charge of the life-cycle management of virtual infrastructures; the virtual network function manager (VNFM), in charge of the life-cycle management of the virtual network functions (VNFs); and the NFV orchestrator (NFVO), in charge of orchestrating the deployment of services and resources in a fully automated way. More details on the description of these components can be found in [43]. In this work, we have extended this architecture with the addition of our purposed slice manager. It is noted that a recent report from ETSI includes a new definition of the interfaces to support network slicing, ETSI MANO GR NFV-EVE 012 v3.1.1 [34], and our proposed framework has been designed to follow this standard. This contribution can be considered as a prototype implementation

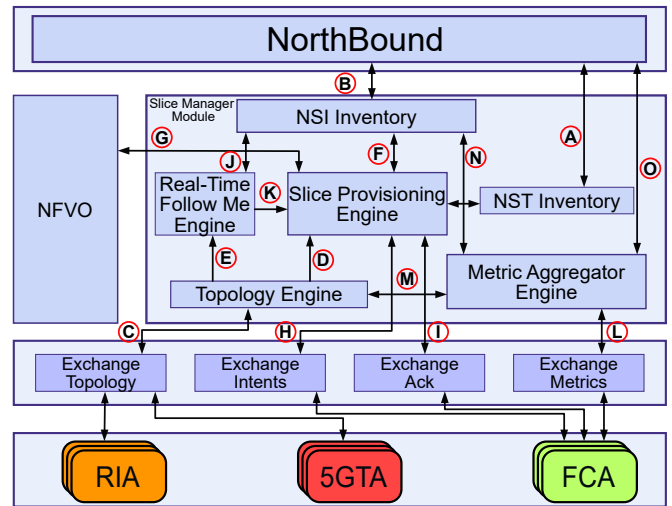


Fig. 3. Slice manager architecture design.

of that standard, which advances significantly beyond the state of the art in terms of the network slicing control functionalities. Secondly, the managed part refers to the rest of the components, which can be accessed and managed by the upper layers of the architecture. The different network segments shown here are referred to as the front-haul, edge, mid-haul, core and back-haul segments respectively. The radio access network (RAN) is usually deployed in the front-haul to provide wireless coverage. The edge network segment is either co-located with the RAN or located in its proximity to allow the deployment of services in the last mile close to final users. The mid-haul network interconnects the edge with the core network, which is usually a large-scale data centre of the telecommunication operator. The back-haul is the telecommunication network infrastructure that is typically managed by infrastructure operators for connecting different systems. In addition, each of the edges are providing coverage to different end users shown as orange and blue users. Each colour represents a different tenant at the infrastructure level.

The coloured components that are shown in Fig. 2 represent the main building blocks of the proposed slice manager framework and consequently our main extensions to the 5G multi-tenant architecture. Section IV will present more detailed information on their roles, responsibilities and tasks.

Finally, this figure also overviews an example of an E2E network slice. Fig. 2 depicts four users that are associated with different radio access networks, which in turn are connected to their respective edges. It is important to highlight that those users could also belong to different tenants. In this example, an E2E network slice has been created to ensure performance and the overall QoS committed. The data path depicted in this figure shows the main points of the infrastructure through which the network traffic passes. It is noted that this is just a representative example of a network slice and various other scenarios are also available.

#### IV. THE PROPOSED SLICE MANAGER FRAMEWORK.

The overall architecture of the proposed framework is mainly composed of four architectural domains as shown in Fig. 3. From a bottom-up perspective, the first layer (L1 in Fig. 3) is composed of monitoring, discovering and actuating agents distributed throughout the entire 5G infrastructure. The middleware layer (L2) comprises the communication channels through which all our components are connected. The third layer (L3 in Fig. 3) is the management layer, which consists of two main blocks: the NFVO coordinating network resources for VNFs and the life-cycle management of VNFs and network services, and our extension to manage network slices and their life-cycle. Finally, on top of the architecture (L4 in Fig. 3) is the north bound interface (NBI), from which any external authorised agent can access the exposed functions there to create, modify or monitor either NSTs or NSIs.

This section first presents the NST and NSI approaches in this work and then explains each of the components of the architecture represented in Fig. 3.

##### A. Slice Manager Network Slice Template and Network Slice Instance

The slice manager module is the main architectural component of the proposed framework, dealing with the complete life-cycle management of both NSTs and NSIs. In this context, it is essential to concisely define these two managed entities:

- **NST** describes an NSI in terms of instance-specific information represented by a set of different QoS requirements and characteristics (see Contribution 1 in Section I):
  - Minimum bandwidth warranted for the slice (bps).
  - Maximum bandwidth available for the slice (bps).
  - Priority of the traffic that belongs to the slice (integer value used to control latency, jitter and packet loss. The higher the value, the lowest the latency, jitter and probability of packet loss).
  - Traffic direction (ongoing or outgoing traffic).
  - Added value to VNF services such as VPN encryption or load balancing.
  - Coverage area indicated by one or a combination of the following (see Contribution 5 in Section I):
    - 1) A set of wired and wireless network interfaces where the network slice should be provided.
    - 2) A set of physical and/or virtual machines implying coverage over all their interfaces.
    - 3) A set of physical zones (A set of physical machines logically assigned to a specific zone).
  - Data plane programmability technology, specified in case the network administrator needs to have control over what technology (e.g., OpenAirInterface FlexRAN, Open vSwitch (OVS), and NetFPGA-SUME, subject to availability) will be employed to program the network devices through the E2E slice datapath.

The slice manager best effort delivery mode will be applied if any parameter is not specifically selected. Our

design does not impose an NST catalogue by default; instead, the user can define and store their own NSTs in the NST catalogue and maintain them in the NST Inventory (see “A” in Fig. 3).

- **NSI** represents the instantiation and deployment of an NST. When an NST is implemented, the whole E2E network datapath of the slice is automatically programmed/adapted to offer the specified service-level agreements, thereby becoming an NSI capable of being monitored and controlled throughout its life cycle. To this respect, an NSI in this work can leverage the most advanced data-plane programmability technologies available to allow complex packet processing to support heterogeneous traffic in multi-tenant 5G networks. This includes not only the traditional support for communications between physical machines and services running on them, but also for a wide spectrum of communications at other network levels such as virtual machines and their underlying services, 5G user equipment (UE), and multi-tenant communications. To ensure that the slice requirements are met, any single point of the datapath of the E2E slice has to tackle not just the traditional 5-tuple (source IP, destination IP, source port, destination port and transport protocol), which only allows network slices per physical machine or per service running in it, but also be able to process overlay networks such as VLAN, VXLAN, GRE, TRILL, NVGRE, NVO3, and GTP, introduced for user mobility and multi-tenancy support purposes (see Contribution 2 in Section I). These NSI exposed functions are reachable from the northbound application programme interface (API) of the slice manager and maintained by the NSI Inventory (see “B” in Fig. 3).

##### B. Monitoring, Discovering, and Actuating Agents

- **Resource inventory agent (RIA)** is in charge of discovering and keeping updating the complete inventory of physical and virtual machines together with their inter-networking and topological information (see Contribution 3 in Section I). This component is deployed in every applicable physical host and can discover the topology using network discovery protocols such as CDP, LLDP and SNMP, hypervisor APIs such as libvirt and libpci), and running daemons such as OpenStack nova-compute. More details of the architecture of the RIA are presented in [44].
- **5G topology agent (5GTA)** is in charge of discovering and keeping updating the complete inventory of 5G users connected to the 5G network together with their locations. This component is deployed as an extension to the 5G access and mobility management function (AMF) component that exposes an API to inform user connection, disconnection and mobility. More details of the architecture of the 5GTA can be found in [44].
- **Flow control agent (FCA)** abstracts a set of heterogeneous programmable data plane technologies (wireless and wired) and provides a unified control

interface for network slicing. The FCA also provides key performance metrics about the status of the network slices at a specific datapath point (e.g., a network interface). This component is deployed in each of the applicable physical machines in the architecture to communicate with their networking devices in a technology-dependent manner (e.g., OVS). For more details on the architecture of the FCA, the mechanisms that it abstracts from different technologies and the control functions that it provides for network slicing, refer to [45].

### C. Slice Manager Management Modules

The slice manager component is composed of the following modules, required to provide the management functionality expected.

**Topology engine:** This module is responsible to maintain topological information of the infrastructure up to date, including physical and virtual machines, their associated interfaces, and the logical and physical connections between such interfaces. It receives periodic messages from the message bus where all the topological agents of the framework report (see “C” in Fig. 3). This information is later on used to allow slice manager to understand which are the points of the network where control functions need to be configured throughout the datapath of the E2E slice (see “D” in Fig. 3). This module also generates a notification when a UE is involved in the handover across different radio access points (see Contribution 7 in Section I) (see “E” in Fig. 3).

**Slice provisioning engine:** This module is responsible for interacting with the control plane to enforce the NSI along all the different network interfaces involved along the E2E path. To do so, this set of interfaces is determined according to the definition of the NST (see “X” in Fig. 3) and it invokes the creation of the VNFs associated with the value-added services that are running in the context of the NSI (see Contribution 6 in Section I) (see “F” in Fig. 3). After that, this module employs the topological information to discover the network interfaces involved in the E2E path to complete an independent NSI per network interface.

This step only happens when an added-value VNF service such as VPN encryption or loadBalancing is needed (see letter “G” in Fig. 3)

Once each NSI is created automatically, it produces a set of intents for the FCA where the network interface belongs. These intents define the intention to enforce a network slice on such interfaces (see “H” in Fig. 3). The metadata reported together with the topological information allows this module to determine which FCA instance should receive a particular intent from a concrete network interface. After the FCA has programmed/updated the associated network interfaces as required from the Intents, it will return an acknowledgement with the status of the execution of such intent to allow controlling the life-cycle of this NSI (see “I” in Fig. 3).

**Real-time follow-me engine:** This module is responsible to detect mobility changes of users in the network and reconfigure the NSIs associated with those users to achieve the data plane programmability of the network slice across

different antennas when the user is moving. To do so, this engine gathers the topological information (see “E” in Fig. 3) and uses the activated NSIs in the system (see “J” in Fig. 3). It determines if the existing NSIs need to be removed from a given gNB/eNB and created in another one through a migration of the NSIs associated with the handover of the mobile user. If so, it informs the Slice Provisioning Engine of this (see “K” in Fig. 3) to send the associated intents to the control plane.

**Metric aggregator engine:** receives metrics from the infrastructure (see “L” in Fig. 3) and keeps them up to date. Such metrics can be aggregated in the temporal and spatial domains. To allow the spatial aggregation, the topological information is required by the module (see “M” in Fig. 3). This module consults the NSIs (see “N” in Fig. 3) to determine how to aggregate raw metrics to provide metrics associated to the NSIs. Such metrics are exposed in real-time using the northbound API (see “O” in Fig. 3).

## V. SLICE MANAGER SEQUENCE DIAGRAM AND LIFE-CYCLE

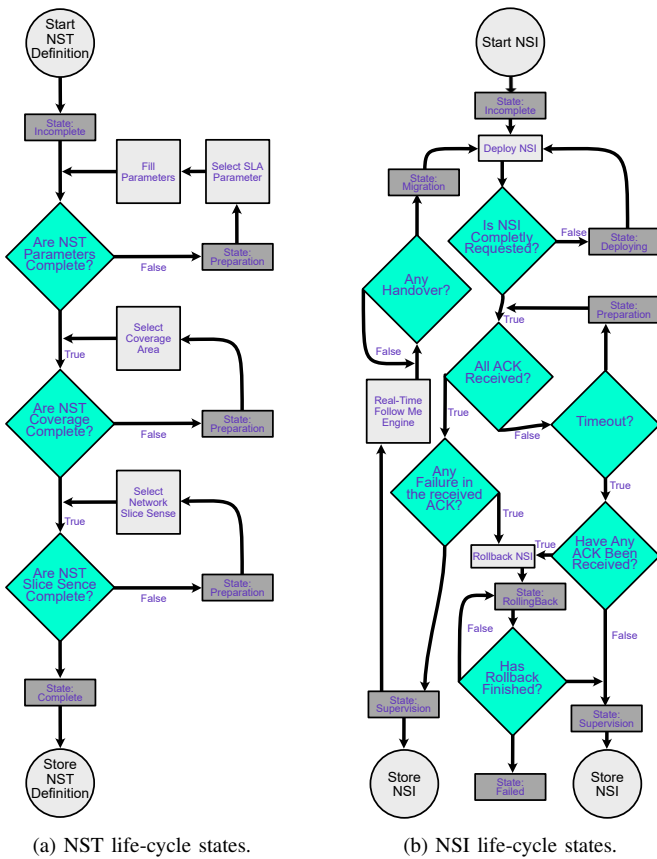
### A. Network Slice Management Life-cycle

Fig. 4 shows both NST and NSI life cycles in the proposed slice manager framework.

- **The NST life-cycle** (Fig. 4(a)) is initially in an *incomplete* state moving to the *preparation* state until the three main parameters including SLA Definition, Coverage and Slice Network Sense are configured. The SLA Definition and the Coverage are iterative processes, and thus the user can add different parameters to create their own NST. The Network Sense can only have one parameter: ingress, egress, or both. Once the three main parameters are configured, the state of the NST is *complete* and the NST definition is saved in the storage.
- **The NSI life-cycle** (Fig. 4(b)) is initially in an *incomplete* state when the NSI is created. After that, the NSI starts creating all the intents that match the definition of the NST and start sending all the intents to the FCA to deploy the E2E slice; this state is called *deploying*. Once the deployment is correctly enforced, the status changes to the *preparation* state; this state is made to check that all the intents have been enforced successfully. In case that an intent produces an error, the slice manager tries the same action five times. After five failed attempts, the NSI state moves to *rolling back*, which means that all the NSIs that have been deployed successfully need to roll back. When the rollback finishes, the NSI status is *failed*. After all the NSIs have been successfully deployed, the state moves to *supervision*, and it is stored. The *supervision* state is defined in the slice manager to detect in real-time any handovers, and it re-deploys the affected NSI to move it to the *migration* state whilst updating the topological information (see Contribution 4 in Section I).

### B. Network Slice Management Sequence Diagram

Fig. 5 depicts a sequence diagram of the four most common operations among the different architectural components



(a) NST life-cycle states.

(b) NSI life-cycle states.

Fig. 4. Life-cycle states of network slice instance and template.

related to network slice management, defined in the following subsections. The previous sections have included some partial information whilst this section contains a complete overview of all elements. Notice that each of the interactions in Fig. 5 contains a number in brackets matching the same number in the text within brackets.

### C. Bootstrapping

(1) Resource infrastructure agent (*RIA*), (2) 5G infrastructure agent (*5GTA*), (3) slice manager and (4) flow control agent (*FCA*) already described in Section IV-B start loading their configuration and conducting their main tasks. slice manager starts separated threads where each module will be working independently (Inventories (5), API (6), provisioning engine (7), metric aggregator engine (8), topology engine (9), and real-time follow-me engine (10)) already described in Section IV-C.

*RIA* starts collecting all the topological information in each of the applicable physicals computes where they are located such as the name of the interfaces and parameters associated. *RIA* also gathers information on the different bridges such as Linux switches (BRCTL) and OpenVSwitches (OVS) switches (11). Moreover, *RIA* discovers all the connections among networking elements to construct a centralised networking overview in the slice manager. Furthermore, *RIA* detects the hypervisor in each physical machine and inspects all the virtual machines and containers that are instantiated on

the physical machine. This feature monitors the life cycle of the virtual machines and containers to know whether they are running or not. *5GTA* starts collecting all the information from the 5G infrastructure in each gNB/eNB (12) by detecting the users connected across the 5G Infrastructure and indicating to which gNB they are connected (13).

slice manager inventories start with the subscriptions to different exchanges in the communication middleware to start receiving actions, metrics and intents messages (14). *RIA* and *5GTA* send periodic topology messages (15) (16) to slice manager. All messages from the middleware are directly stored (17) in the inventory, shared with topology engine (18) and indexed (19). *FCA* sends also periodic metrics (20). All such metrics are directly stored in the inventory (21), shared with metric aggregator engine (22) and they are then indexed (23). Moreover, *FCA* makes the subscription to actions and intents (24); all intents messages received are directly stored in the inventory (25), shared with the provisioning engine (26) and then indexed (27). At this stage, slice manager is bootstrapped and fully functional.

### D. Network Slice Template Definition

A user would like (28) to define a network slice template (29), the user automatically gets the previously configured and supported action definition elements (30) that can be part of the NST from the slice manager API. (31) The user selects the action definitions for the desired slice among all the options available (32) and sends them back to the slice manager API (33). At this point, slice manager has one network slice template (34) with all the parameters fulfilled except the area of coverage. Thus, the user retrieves from slice manager, the up-to-date topological information (35) to determine the coverage of the network slice (36). The user decides the coverage (37) and sends it to the slice manager (38). At this point, the network slice template is complete and stored inside the NST catalogue slice manager (39). It is noted that this is a management step and it does not have any interaction with the control plane.

### E. Network Slice Instance

The deployment of a new network slice instance is initiated by the end user, who provides the associated information for the NST and the scope of the network flows that belong to the network slice, and submits an NSI instantiation request to the slice manager API (40). The steps between (41) and (44) are optional and they will only happen in the case that the NST contains the added-value parameters such as encryption or load balancing use cases using SFC. At this moment, slice manager interacts with the VIM to perform the creation and starts the VNFs associated with the provisioning of the value-added services associated with this network slice if any, e.g., an encryption service (41). The service function chaining configuration associated with such network services is also configured using the VIM interface (41) described as one of the main contributions in the proposed innovations of the slice manager. *RIA* will report the new topological changes produced by the creation of such VNF (42) so that the latest topological changes are up to date (43) and transferred to the

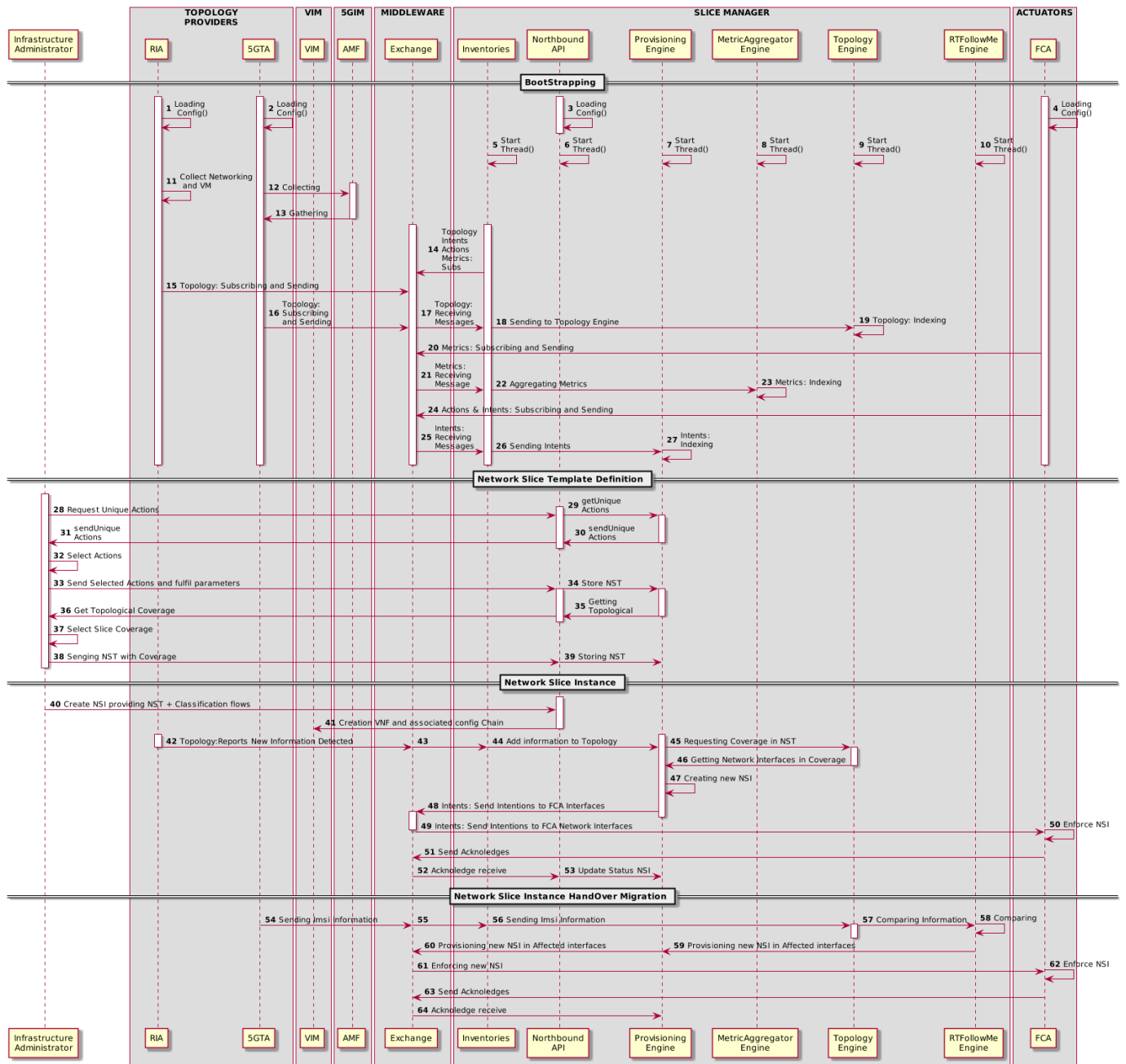


Fig. 5. Sequence diagram slice manager.

topology engine (44).

The provisioning engine will take the coverage information and the technology information available in the NSI (45) and calculate the list of the network interfaces that need to be used for enforcing the NSI to match the indicated coverage and technology using the latest topological information available (46). This set of network interfaces will be used to create a set of network Intents that will compose the plan to be orchestrated to enforce the E2E network slice (47). All these network Intents are sent (48) to the different FCA components of the system (49). Each FCA will then enforce the intent (50) and return an Acknowledgement (51) with the result of the execution of the enforcement (52). If not every

ACK is received or if all the ACKs are received but any of them is negative thereby incurring a rollback (see Fig. 4(b)), it will be considered as a failure. Otherwise, the NSI will be successfully enforced and at this moment the network slice will be at the operational stage (53).

#### F. Network Slice Instance Handover Migration

When 5GTA detects user mobility across gNBs/eNBs (54), it notifies the topology exchange (55) of such change. Consequently, it is received by the topology engine and the topological information is updated (56). The real-time follow-me engine constantly checks changes in such topological information (57) against the coverage information



of the network slices already instantiated (“deployed” state). When this change is detected (58), the real-time follow-me engine will calculate the affected set of interfaces that need to install the network slice and the affected set of interfaces where the network slice needs to be removed (59). This logic has been designed using a path-finding algorithm [46] that iterates the topological information to determine what kind of handover is taking place (60) [47]. It is noted that a handover can happen between gNBs allocated in the same physical machine or in different machines of the name zone or even between different zones, and thus the number of interfaces involved in these handovers is incrementally increased. This newly generated plan composed of a set of Intents required to re-deploy the NSI (61) with the new location of the user is sent to the Provisioning Engine, which sends such Intents to the FCA (62). The acknowledgement process is similar to the provisioning as already explained (63), (64). This allows a dynamic migration of network slices to follow moving users across gNBs and eNBs.

## VI. IMPLEMENTATION

The complete network slice management framework has been designed and prototyped. RIA and slice manager were implemented in Java 8. FCA was implemented in Python 3.7 and 5GTA was implemented in C, mainly due to facilitating the integration with the underlying control plane functions. OpenStack Rocky has been used as VIM implementation. The softwarised 5G network employed was the RAN and Core of the OpenAirInterface software components. In terms of network slicing, OpenAirInterface FlexRAN [48] was utilised for 5G RAN slicing. The NetFPGA-based 5G network slicing core [49] was adopted for network slicing over the physical infrastructure [50], and OVS network slicing extensions [51] were employed for network slicing over virtual infrastructures, respectively. The prototype implementation of the slice manager matches exactly the design described in previous sections and the infrastructure used for the testing and empirical validation was also similar to the one described in previous sections.

## VII. EXPERIMENTAL RESULTS

This section explains a set of experiments to validate the proposed E2E slice manager framework. This section does not provide any comparison with other frameworks since the currently available frameworks do not provide the same capabilities as shown in Table I as far as we know.

- Functional validation of the network slice management framework.
- Empirical analysis of topological scalability in the slice manager.
- Empirical analysis of NSI scalability in the slice manager.
- Functional validation of E2E slicing management.

### A. Execution Testbed

The testbed used for experimentation is composed of 10 machines Dell T5810 with Intel Xeon CPU Xeon E5-2665 v4, 8 Cores with hyper-threading, 20 MB Intel Smart Cache, 256GB of RAM, 2TB solid-state drive (SSD) and 2 x Intel X540-AT2 10 GbE. Of the 10 machines, one machine contains the slice manager, the other one hosts OpenStack infrastructure with two virtual machines (VMs) to act as a network controller and cloud controller respectively [10], and the remaining eight machines are OpenStack computes. There are 4 different physical zones, and each of them includes two machines registered per geographical zone called availability zones in OpenStack to allow edge and core logical separation in the management plane. The VIM contains two tenants sharing the cores and edges among them. This configuration creates a star network topology among all machines creating an overlay physical infrastructure. The VIM is running the OAI with one virtual machine in the edge connected directly to a x310 [52] and 5 virtual machines in the core, the 5gNB (CU) are interface machines connecting one antenna per tenant and per edge, 8 in total. Traffic traversing this testbed follows a nested network protocol structure with different levels of encapsulation: (1) MAC/IP/UDP, related to the communication between physical machines including medium access control (MAC), IP and the L4 transport protocol (TCP or UDP); (2) VXLAN/MAC/IP/UDP, used to isolate tenant traffic, especially for mobile network operators sharing the same physical 5G infrastructure as tenants, notice that VXLAN encapsulation protocol is used to achieve tenant isolation, however, other alternative protocols like GRE or GENEVE can be employed to fit the same purpose; (3) GTP/IP/UDP/SERVICE, introduced to allow the management of the user mobility across different gNB transparently without losing connectivity, GTP tunnel per UE between 5gNB (CU) and SPGW-U. Finally, the application header contains the data being transmitted by the end users, the SERVICE. Normal IP network packets use a very limited subset of these headers, for instance, MAC/IP/UDP/SERVICE, nonetheless, compared to this simple case, several additional headers have been added to achieve both multi-tenancy and mobility throughout the virtualised 5G network.

### B. Functional Validation of the Network Slice Management Framework

This section is to validate the expected functional behaviour of the slice manager to demonstrate that an E2E network slice is deployed across the data plane as a result of a distributed orchestration of the different FCAs to enforce the network intents in each network interface. Fig. 6 shows the throughput traffic in different points of the network in the infrastructure to show an example of what happens with a critical network service (top) and background competing traffic (bottom) when an E2E network slice is enforced along the different horizontal and vertical network segments that include gNB/eNB (wireless interface) together with physical and virtual infrastructures. Both figures share the same legend and are aligned in time so that the user can see the relationship between the competing

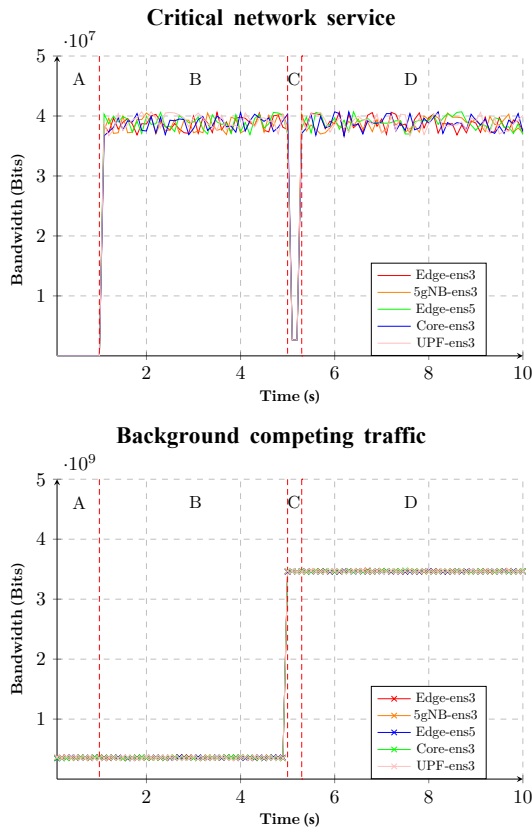


Fig. 6. E2E network slicing demonstration.

traffic and the traffic belonging to the mission-critical slice. There is no human intervention in this experiment.

Fig. 6 is divided into different steps identified by letters, where each of them is a different state with different behaviours of the slice manager.

- A Shows there is no critical service whilst the background traffic is flowing as normal.
- B Shows the critical network service starts the transmission with good quality as there is only low background traffic in the 5G infrastructure.
- C Shows high background traffic is injected into the infrastructure to saturate the links and decrease drastically the quality of the critical network service during 0.2 s.
- D Shows E2E network slicing is enforced in the infrastructure and thus the critical network service is guaranteed without interruptions while the high background traffic is not interfering with the critical service.

**C. Empirical Analysis of Topological Scalability in Slice Manager**

In order to evaluate the topological scalability of the slice manager, these experiments create different scenarios that represent in the slice manager java instances of zones, physical machines per zone, virtual machines per physical machine and network interfaces per physical and virtual machine. These different scenarios allow us to stress the slice manager in heavy conditions and also to analyse the time required in large-scale

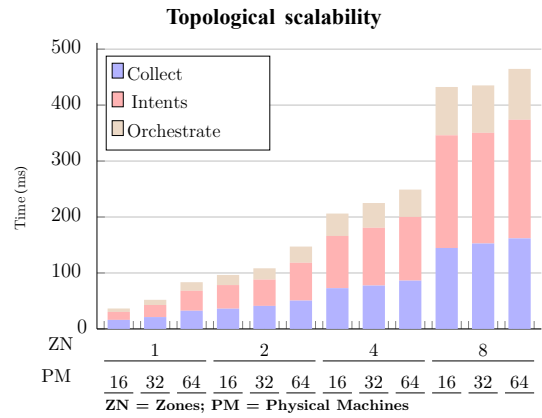


Fig. 7. NSI provisioning times.

scenarios. In Fig. 7, there are 2 X-axis groupings for various scenarios. The first X-axis, called *zones (ZN)*, is the number of physical zones available in the infrastructure associated with each scenario, exponentially ranging from 1 to 8. The second X-axis, called *physical machines (PM)*, ranges from 16 to 64 PMs associated with each of the zones available in that scenario. The number of *virtual machines (VM)* is always 1024 for a shake of simplicity. The Y-axis records the time required for the slice manager to collect all the network interfaces, generate the Intents with an associate NST, and orchestrate those intents in milliseconds. Each of the stacked bars plotted in Fig. 7 shows the three different steps measured along the execution of the scenarios. These times start when a new E2E network slice is deployed by the slice manager until all NSIs are completely sent to the distributed FCA in the infrastructure. Notice this experiment is large enough to be tested in a real environment as mentioned before. Thus, this experiment measures the scalability of the slice manager. The first time stacked in blue called *Collect* is the time the slice manager spends collecting all the topological information associated with the infrastructure used for the scenario. The second time stacked in red called *Intents* is the time the slice manager needs to create the set of Intents to enforce the NSI. The third time stacked in orange called *Orchestrate* is the time that the slice manager needs to submit all the Intents to the different FCA instances.

The proposed slice manager system yields high scalability from the fact that it performs impressively well of no more than half a second even in the most stressed scenario with eight ZN, 512 PMs (64 \* 8) and 524,288 VMs (1024 \* 512). It requires the deployment of the network slice in 2,101,248 network interfaces. It validates support for large-scale scenarios.

**D. Empirical Analysis of Network Slice Instance Scalability in Slice Manager**

This subsection provides an empirical validation of the scalability in the Slice Manager regarding the network slice instances. Fig. 8 shows four heatmaps where the X-axis are ranging the number of PMs from 1 to 32, the Y-axis is composed of the number of VMs per Zones. The number

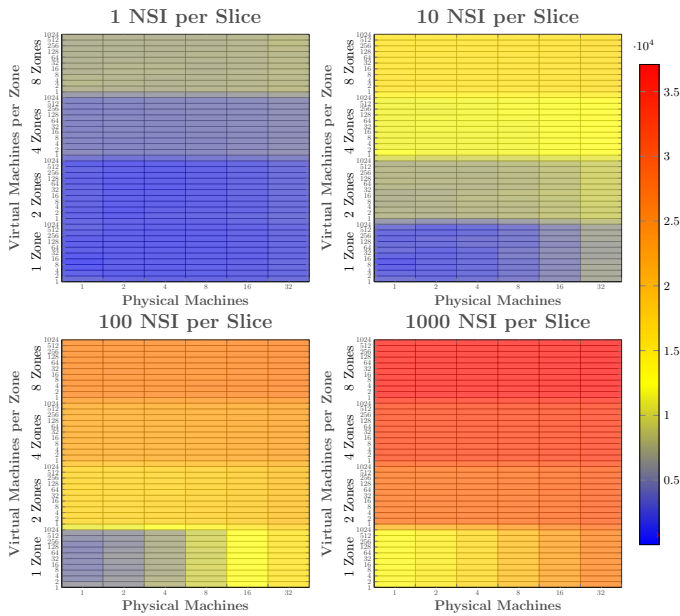


Fig. 8. Slice manager NSI scalability.

of VMs are ranging from 1 to 1024 in each zone within each PM. Each cell is an interpolation between blue and red, where blue is the minimum time (11 ms) and red is the maximum time (32 s) that the slice manager spends in creating the number of NSIs in milliseconds. There are four different experiments to measure the time with different numbers of NSIs (1, 10, 100, and 1000) created per network interface. In the first three scenarios (1, 10, and 100 NSIs per interface), only up to 2 s are needed. In the fourth scenario (1000 NSIs per interface), the worst case takes 32 s for eight zones with 32 PMs per zone. Notice that this is to stress the slice manager and in a normal scenario, it will never happen (to enforce 1000 NSI per slice).

### E. Empirical Validation of E2ESlicing

This subsection provides an empirical validation measuring the time of deploying an E2E Slice between two different UEs. The testbed used for this validation was described in Section VII-A. The scenario with 4 different zones and 2 tenants contains 8 antennas (model x310). Each antenna belongs to a different tenant and there is one tenant per zone. The experiment was done randomly by choosing 2 UE among all the available in the scenario (2048 UE). Once both UEs were chosen, the framework calculated the shortest path between both UEs gathering the information regarding all the network interfaces where the framework slice manager needed to enforce each of the NSI [47]. Once that information was collected, the slice manager created all the NSIs and sent them to each FCA. There exists four different combinations for the evaluated scenarios:

- Both UE belong to the same tenant and are in the same zone.
- Both UE belong to the same tenant and are not in the same zone.

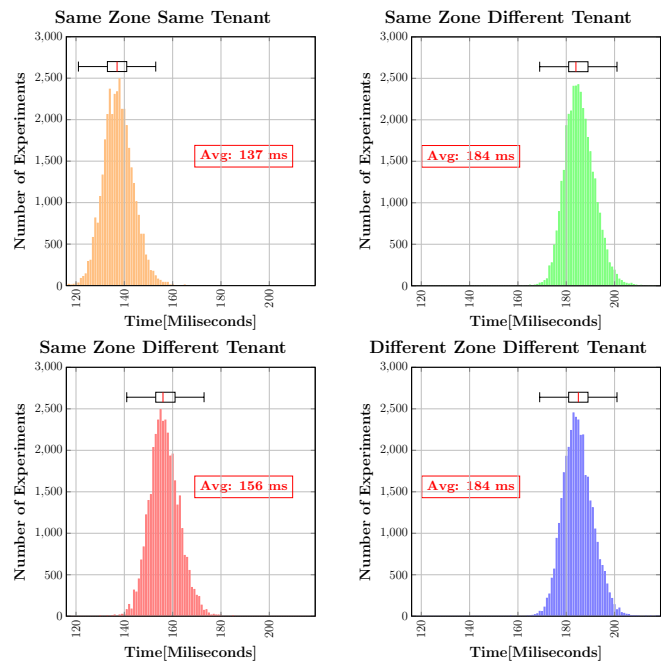


Fig. 9. Slice manager E2E network slicing scalability.

- Both UE do not belong to the same tenant and are in the same zone.
- Both UE do not belong to the same tenant and are not in the same zone.

## VIII. CONCLUSION

This paper presents a highly automated e2e network slice management framework, and the detailed design includes the overall architecture and components, network slice life-cycle management, the deployment of the proposed network slice management framework in a 5G multi-tenant infrastructure, and sequence diagrams for the slice management operations. The proposed framework is capable of flexibly defining network slices from different perspectives of various business models for network slice providers. For all kinds of such network slices defined, the framework can manage the slices' whole life-cycle E2E and achieves slice mobility across radio access networks for mobile slice users. All the operations after slice definition are automated for high efficiency and to reduce operating expenditure. The framework has been prototyped and empirically validated regarding the main functionality. Experimental results have demonstrated the high efficiency and scalability of the implemented system in terms of rapid network management operations over an emulated realistic large-scale 5G infrastructure with up to eight geographical zones, 512 physical machines and 524,288 virtual machines. In the worst-case scenario, creating a network slice instance only takes half a second, and managing slice handovers at around 1.4 s.

## REFERENCES

- [1] Network slicing key for industry-specific 5G services. [Online]. Available: <https://www.gsma.com/futurenetworks/digest/network-slicing-key-industry-specific-5g-services/>

- [2] P. Popovski, K. F. Trillingsgaard, O. Simeone, and G. Durisi, "5G wireless network slicing for eMBB, URLLC, and mMTC: A communication-theoretic view," *IEEE Access*, vol. 6, pp. 55 765–55 779, 2018.
- [3] R. Marco Alaez, J. M. Alcaraz Calero, F. Belqasmi, M. El-Barachi, M. Badra, and O. Alfandi, "Towards an open source architecture for multi-operator LTE core networks," *J. Netw. Comput. Appl.*, vol. 75, pp. 101–109, 2016.
- [4] E. Chirivella-Perez, J. M. A. Calero, Q. Wang, and J. Gutiérrez-Aguado, "Orchestration architecture for automatic deployment of 5G services from bare metal in mobile edge computing infrastructure," *Wireless Commun. Mobile Comput.*, vol. 2018, 2018.
- [5] E. Chirivella-Perez, J. Gutiérrez-Aguado, J. M. A. Calero, and J. M. Claver, "Towards a SDN-based architecture for analyzing network traffic in cloud computing infrastructures," in *Proc. IEEE SoftCOM*, 2015.
- [6] A. Shalimov, D. Zuikov, D. Zimarina, V. Pashkov, and R. Smeliansky, "Advanced study of SDN/openflow controllers," in *Proc. ACM CEE-SECR*, 2013.
- [7] A. Rao, "Network automation is key to delivering significant opex reduction and increasing agility in the 5G era," 2020. [Online]. Available: <https://www.analysismason.com/research/content/white-papers/mobile-network-automation-rma07-rma01-rma02/>
- [8] GSMA, "Mobile infrastructure sharing," pp. 1–23, 2012. [Online]. Available: <http://www.gsma.com/publicpolicy/wp-content/uploads/2012/09/Mobile-Infrastructure-sharing.pdf>
- [9] 5GPPP, "Supported by the 5G empowering vertical industries," Tech. Rep., 2020. [Online]. Available: [https://5g-ppp.eu/wp-content/uploads/2016/02/BROCHURE\\_5PPP\\_BAT2\\_PL.pdf](https://5g-ppp.eu/wp-content/uploads/2016/02/BROCHURE_5PPP_BAT2_PL.pdf)
- [10] E. Chirivella-Perez, J. Gutiérrez-Aguado, J. M. Claver, and J. M. A. Calero, "Hybrid and extensible architecture for cloud infrastructure deployment," in *Proc. IEEE CIT*, 2015.
- [11] K. Samdanis, X. Costa-Perez, and V. Sciancalepore, "From network sharing to multi-tenancy: The 5G network slice broker," *IEEE Commun. Mag.*, vol. 54, no. 7, pp. 32–39, 2016.
- [12] 3GPP, "3GPP TS 32.130 - Universal Mobile Telecommunications System; Telecommunication management; Network sharing; Concepts and requirements, Release 14," Tech. Rep., Jan. 2017. [Online]. Available: <https://portal.etsi.org/TB/ETSIDeliverableStatus.aspxhttp://www.etsi.org/standards-search>
- [13] 3GPP TS 28.530 version 15.0.0 - 5G; management and orchestration; Concepts, use cases and requirements, Release 15, Tech. Rep., Oct. 2018.
- [14] 3GPP TS 28.530 version 15.0.0 - 5G; management and orchestration; Provisioning, Release 15, Tech. Rep., Oct. 2018.
- [15] 3GPP TR 28.801 version 15.1.0 - 3rd generation partnership project; technical specification group services and system aspects; telecommunication management; study on management and orchestration of network slicing for next generation network, Release 15, Tech. Rep., Jan. 2018.
- [16] R. Dangi, P. Lalwani, G. Choudhary, I. You, and G. Pau, "Study and investigation on 5G technology: A systematic review," *Sensors*, vol. 22, no. 1, p. 26, 2021.
- [17] J. Ni, X. Lin, and X. S. Shen, "Efficient and secure service-oriented authentication supporting network slicing for 5G-enabled IoT," *IEEE J. Sel. Areas Commun.*, vol. 36, no. 3, pp. 644–657, 2018.
- [18] P. Salva-García, J. M. Alcaraz-Calero, Q. Wang, J. B. Bernabe, and A. Skarmeta, "5G NB-IoT: Efficient network traffic filtering for multi-tenant IoT cellular networks," *Secur. Commun. Netw.*, vol. 2018, pp. 1–21, Dec. 2018.
- [19] R. Trivisonno, M. Condoluci, X. An, and T. Mahmoodi, "mIoT slice for 5G systems: Design and performance evaluation," *Sensors*, vol. 18, no. 2, p. 635, Feb. 2018.
- [20] X. Xia *et al.*, "A survey on 5G network slicing enabling the smart grid," in *Proc. IEEE ICPADS*, 2019.
- [21] P. Hovila *et al.*, "5G networks enabling new smart grid protection solutions," no. June, pp. 3–6, Jun. 2019.
- [22] I. M. Elena-Mădălina Oproiu, Catalin Costea, Mihai Idu, Catalin Brezeanu, and Marius Iordache, "5G-Connected virtualized enterprise infrastructure for smart city," *Int. J. Internet Things Web Services*, vol. 4, pp. 27–34, 2019.
- [23] M. Roddy *et al.*, "5G network slicing for mission-critical use cases," in *Proc. IEEE 5GWF*, 2019.
- [24] Q. Wang *et al.*, "Enable advanced QoS-aware network slicing in 5G networks for slice-based media use cases," *IEEE Trans. Broadcast.*, vol. 65, no. 2, pp. 444–453, Mar. 2019.
- [25] P. Salva-García *et al.*, "Scalable virtual network video-optimizer for adaptive real-time video transmission in 5G networks," *IEEE Trans. Netw. Service Manage.*, 2020.
- [26] 5G Tango Consortium, "5GTANGO — 5GTANGO." [Online]. Available: <https://www.5gtango.eu/>
- [27] M. Peuster and S. Schneider, "D2.2 architecture design 5G development and validation platform for global industry-specific network services and apps," 5GTango Consortium, Tech. Rep., 2017. [Online]. Available: [https://www.5gtango.eu/documents/D22\\_v1.pdf](https://www.5gtango.eu/documents/D22_v1.pdf)
- [28] 2020 SliceNet Consortium, "SLICENET – SLICENET." [Online]. Available: <https://slicenet.eu/>
- [29] Q. Wang *et al.*, "SliceNet: End-to-end cognitive network slicing and slice management framework in virtualised multi-domain, multi-tenant 5G networks," in *Proc. IEEE BMSB*, 2018.
- [30] 5G Americas, "The voice of 5G and LTE for the Americas." [Online]. Available: <https://www.5gamericas.org/>
- [31] IMT-2020, "IMT-2020 (5G) promotion group." [Online]. Available: <https://5g-ppp.eu/imt-2020-5g-promotion-group-and-5g-ppp-announce-memorandum-of-understanding-for-5g/>
- [32] 5G Americas consortium, "Network slicing for 5G networks & services, white paper," Tech. Rep., 2016.
- [33] K. Martiny *et al.*, "5G network and service management including orchestration by NGMN alliance," pp. 1–32, 2017.
- [34] ETSI, "ETSI GR NFV-EVE 012 V3.1.1 (2017-12): Network functions virtualisation (NFV) Release 3; evolution and ecosystem; report on network slicing support with ETSI NFV architecture framework," vol. 1, pp. 1–35, 2017.
- [35] OSM, "OSM release FIVE: Technical overview," Tech. Rep., 2019.
- [36] I. Afolabi, J. Prados-Garzon, M. Bagaa, T. Taleb, and P. Ameigeiras, "Dynamic resource provisioning of a scalable E2E network slicing orchestration system," *IEEE Trans. Mobile Comput.*, vol. 19, no. 11, pp. 2594–2608, 2019.
- [37] V. Q. Rodriguez, F. Guillemin, and A. Boubendir, "5G E2E network slicing management with ONAP," in *Proc. IEEE ICIN*, 2020.
- [38] I. Afolabi, T. Taleb, P. A. Frangoudis, M. Bagaa, and A. Ksentini, "Network slicing-based customization of 5G mobile services," *IEEE Netw.*, vol. 33, no. 5, pp. 134–141, 2019.
- [39] A. De la Oliva *et al.*, "5G-transformer: Slicing and orchestrating transport networks for industry verticals," *IEEE Commun. Mag.*, vol. 56, no. 8, pp. 78–84, 2018.
- [40] P. Rost *et al.*, "Customized industrial networks: Network slicing trial at Hamburg seaport," *IEEE Wireless Commun.*, vol. 25, no. 5, pp. 48–55, 2018.
- [41] S. Huang, B. Guo, and Y. Liu, "5G-oriented optical underlay network slicing technology and challenges," *IEEE Commun. Mag.*, vol. 58, no. 2, pp. 13–19, 2020.
- [42] A. Fahmin, Y.-C. Lai, M. S. Hossain, and Y.-D. Lin, "Performance modeling and comparison of NFV integrated with SDN: Under or aside?" *J. Netw. Comput. Appl.*, vol. 113, pp. 119–129, 2018.
- [43] E. Chirivella-Perez *et al.*, "Selfnet 5G mobile edge computing infrastructure: Design and prototyping," *Software: Practice and Experience*.
- [44] I. Sanchez-Navarro, A. Serrano Mamolar, Q. Wang, and J. M. Alcaraz Calero, "Sgtoponet: Real-time topology discovery and management on 5G multi-tenant networks," *Future Gener. Comput. Syst.*, vol. 114, pp. 435–447, Jan. 2021.
- [45] A. Matencio-Escolar, J. M. Alcaraz Calero, P. Salvá-García, J. Bernal Bernabe, and Q. Wang, "Adaptive network slicing in multi-tenant 5G IoT networks," *IEEE Access*, pp. 14048–14069, 2021.
- [46] D. B. Johnson, "A note on Dijkstra's shortest path algorithm," *J. ACM*, vol. 20, no. 3, pp. 385–388, 1973.
- [47] I. Sanchez-Navarro, J. B. Bernabe, J. M. Alcaraz-Calero, and Q. Wang, "Advanced spatial network metrics for cognitive management of 5G networks," *Soft Comput.*, pp. 1–18, 2020.
- [48] X. Fokas, N. Nikaiein, M. M. Kassem, M. K. Marina, and K. Kontovasilis, "FlexRAN: A flexible and programmable platform for software-defined radio access networks," in *Proc. IEEE CoNEXT*, 2016.
- [49] R. Ricart-Sanchez, P. Malagon, A. Matencio-Escolar, J. M. Alcaraz Calero, and Q. Wang, "Toward hardware-accelerated QoS-aware 5G network slicing based on data plane programmability," *Trans. Emerging Telecommun. Technol.*, vol. 31, no. 1, Jan. 2020.
- [50] R. Ricart-Sanchez *et al.*, "Towards an FPGA-accelerated programmable data path for edge-to-core communications in 5G networks," *J. Netw. Comput. Appl.*, vol. 124, pp. 80–93, 2018.
- [51] A. Matencio-Escolar, Q. Wang, and J. M. Alcaraz-Calero, "Slicenetswitch: Definition, design and implementation of 5G multi-tenant network slicing in software data paths," *IEEE Trans. Netw. Service Manage.*, p. 14, 2020.
- [52] Ettus. (2022) USRP X310. [Online]. Available: <https://www.ettus.com/all-products/x310-kit/>



**Enrique Chirivella-Perez** is a Postdoctoral Researcher at the University of the West of Scotland, United Kingdom. He has been technically involved in the H2020 5G-PPP Phase 2 SLICENET and 5G-PPP Phase1 SELFNET project. Nowadays is involve in 6G-Brains, 5G-Induce, Arcadian-IoT His main interests include E2E network slicing, monitoring, network control and management, infrastructure zero-touch deployment, 5G NFV Services and software-defined networks in cloud computing and 5G networks in cloud computing. On the academic side, he has 20 publications in SCI-indexed international journals, conferences and European white paper. He has been involved as a Technical Programme Committee Member in more than 30 international conferences. In the industrial side, he has experience in more than 15 projects.



**Qi Wang** is a Professor at the University of the West of Scotland. He is the Technical Co-coordinator of EU H2020 5G-PPP SELFNET and SliceNet projects, and Co-principal Investigator of EU H2020 5G IN- DUCE and 6G BRAINS projects. He is a Board Member of the Technology Board of EU 5G-PPP. His research primarily focuses on 5G mobile networks, video networking and artificial intelligence.



**Pablo Salva-Garcia** is a Lecturer at the University of the West of Scotland, UK, where he received his PhD. Salva-Garcia joined such institution in 2018 as Post-Doctoral Researcher, where during 5 years, he was involved in several EU Horizon projects such as SelfNet, SliceNet or 6GBrains, among others. Salva-Garcia is an active researcher member of the Beyond5GHub and has produced top-tier research publications including international journals, conferences and project technical reports. His main interests include project management, network management, cognitive control plane, software-defined networks, cloud computing, video delivery in mobile edge computing and 5G networks.



**Ignacio Sanchez Navarro**, is a Post-Doctoral Researcher at the University of the West of Scotland. He is working with the beyond 5G hub group, pioneer in 5G and beyond network architectures, artificial intelligence (AI), computer vision, unmanned aerial vehicle (UAV) and Internet of things (IoT). His background is focused on the monitoring and management of multi-tenant 5G and Beyond networks. During his research carried out in the completion of the PhD, he has thoroughly investigated the multi-tenant 5G network topologies.



**Jose Alcaraz Calero** is Full Professor in Network at University of the West of Scotland (UWS). He has significant experience in large R&D projects, being involved of H2020, HE, and EPSRC, totalling more than 70m GBP. He has published more than 200 contributions, including 15 patents and IPRs, in international conferences and renowned journals in the fields of infrastructure management, network control, network security, network monitoring and cognitive close loops. Prof. Alcaraz-Calero is active in chairman and editorship activities where he has served in more than 20 international conferences and 20 editorial activities for journals like IEE Communication Magazine.