Uniyal, N., Muqaddas, A. S., Gkounis, D., Bravalheri, A., Moazzeni, S., Sardis, F., Dohler, M., Nejabati, R., & Simeonidou, D. (2020). 5GUK Exchange: Towards Sustainable End-to-End Multi-Domain Orchestration of Softwarized 5G Networks. *Computer Networks*, *178*, [107297]. https://doi.org/10.1016/j.comnet.2020.107297

Peer reviewed version

License (if available):
CC BY-NC-ND

Link to published version (if available):
10.1016/j.comnet.2020.107297

Link to publication record in Explore Bristol Research
PDF-document

5GUK Exchange: Towards Sustainable End-to-End Multi-Domain Orchestration of Softwarized 5G Networks

Navdeep Uniyal[1], Abubakar Siddique Muqaddas[1], Dimitrios Gkounis[2],
Anderson Bravalheri[1], Shadi Moazzeni[1], Fragkiskos Sardis[3], Mischa
Dohler[3] *IEEE Fellow*, Reza Nejabati[1] and Dimitra Simeonidou[1] *IEEE Fellow*

University of Bristol, U.K.[1], Digital Catapult, U.K.[2], King's College London, U.K.[3]

Abstract

5G networks envisage to support a range of vertical industries, circumventing any potential barriers from converging various network technologies and administrative domains. Current solutions focus only on provisioning services within single administrative domains. There is also lack of standards for sustainable end-to-end multi-domain solutions that can use existing Network Function Virtualization (NFV) Management and Orchestration (MANO) systems. This is important to enable operators to collaborate and create innovative end-to-end services in a sustainable environment, where stakeholders can benefit without compromises. In this article, we present the 5GUK Exchange (5GUKEx), a novel hierarchical architecture to enable end-to-end orchestration with minimum overhead in complexity and performance while also allowing operators to maintain full control of their infrastructure. 5GUKEx allows operators to use their existing MANO systems for the single domain orchestration and build a multi-domain API based on standardized models exposed by service catalogues to coordinate the end-to-end service orchestration and interconnection. We built a prototype of the 5GUKEx and evaluated its performance through emulations showing that the 5GUKEx introduces minimum overhead. We also discuss the use-cases and trials using 5GUKEx in addition to the experiments focusing on the flexible nature of architecture, allowing us to use 5GUKEx to provide connectivity among multi domains over optical transport network.

Keywords: 5G Networks, Orchestration, MANO, Multi-Domain Network Service Orchestration

---

1. Introduction

As a major network evolution, 5G technologies are envisioned to deliver big performance improvements over the previous network generations. This is driven by the promise to empower multiple vertical industries and thus foster the flourishing of smart cities, IoT spaces, autonomous transportation and other complex and highly responsive systems. The 5G vertical applications pose stringent requirements in terms of high data rates, low latency and massive connectivity while realizing that only the reduction in costs and increase in deployment agility would result in profitable business models. This has led telecommunication companies and research communities to investigate technologies like Network Function Virtualization (NFV) and Software Defined Networks (SDN) as a promising technological foundation for the upcoming years.

Virtual Network Functions (VNFs) running inside datacenters are easy to instantiate, upgrade and scale while being more fault tolerant, which contributes to a decrease of operating cost and at the same time improves the performance and customer satisfaction. In turn, network equipment can be automatically and remotely configured using SDN technologies, making it easy to incorporate concepts of network slicing and hence allowing better management of available resources to the operators.

As a consequence, the need for specialized software to interact with this kind of infrastructure has acted as a driving force behind the development of Management and Orchestration (MANO) systems for NFV and SDN. Although organizations like IETF and ETSI play a major role in bringing the expertise of multiple communities together in order to create standards and guidelines for those systems [1], and initiatives such as Open Source MANO (OSM) [2], Open Baton [3] and ONAP [4] emerge from open source communities, network operators still rely on their own proprietary solutions and technologies for network orchestration.

This fragmentation endorses an isolating behavior, where existing MANO systems focus on orchestrating the compute and networking resources within a single network and infrastructure domain. As a result, the current efforts in 5G systems cannot support the development of applications that take advantage of the interconnection among multiple network and infrastructure providers, denying some of the very own design principles in the 5G vision [5]. The missing pervasive interconnection is even desirable from the point of view of the infrastructure providers. For example, consider a scenario where one provider does not possess any infrastructure in a remote location, but ensuring its presence is fundamental for its business. Upon an agreement this provider could make use of resources from another provider or even a local authority, quickly managing and orchestrating them in an automatic way.

Operators working closely together by combining various underlying 5G technologies and services offered by each of them will lead to a diverse feature-rich environment that can support innovative and profitable 5G services while minimising the time-to-market for new products and reducing the deployment overheads for inter-domain services. However, due to several concerns such as confidentiality and security, operators would prefer to hide any underlying infrastructure information, e.g., network configurations and specificity, that can expose their business. Abstraction of low-level infrastructure information is also a key principle of virtualised infrastructures and facilitates the high-level interoperability of heterogeneous platforms. Another challenge is the interoperability, meaning that just the existence of a common standardized API could create a "plug-n-play" inter-domain architecture that facilitates the introduction of more operator networks. Furthermore, current internet service stakeholders include the operators that own the infrastructure and content providers (like Amazon or Netflix), that are running Over-the-Top (OTT) services on the infrastructure. It is clear that these two actors also need to collaborate to provide the required user experience. Nevertheless, the advent of 5G and its service-oriented approach provides new business opportunities and introduces additional stakeholders with different backgrounds, e.g., from creative, media, health, education, entertainment industries, that could innovate on top of the available services and the shared infrastructure increasing the profit of both the operators and service providers. Tackling these challenges and meeting the requirements, creates a sustainable ecosystem that fully realizes the vision of 5G and beyond.

In this article, we consider a novel ETSI NFV MANO based architecture for virtualized network services which aims to enable orchestration of network services across multiple domains, while leveraging existing systems already in use or investigated by the industry. This architecture, known as the 5GUK Exchange (5GUKEx) focuses on providing a flexible and light weight plug-and-play solution, able to realize the 5G vision and create end-to-end connected networks. 5GUKEx can be seen as an exchange point responsible not only for interconnecting multiple networks and testbeds, but also for enabling multiple network operators and infrastructure providers to collaborate, in order to provide services to the end-user. The solution follows a hierarchical approach and relies heavily on standardized interfaces, therefore being complementary to the current standardization efforts.

The remainder of this article is described as follows. In Sec. 2, we describe the current work in multi-domain orchestration. In Sec. 3, we discuss in detail, the proposed architecture and its capabilities to fulfill the desired 5G vision. The detailed network service deployment using 5GUKEx will be showed in Sec. 4, followed by a performance evaluation of 5GUKEx in Sec. 5. Finally before the conclusion, we demonstrate that 5GUKEx can operate over multiple underlying technologies in Sec. 6

2. Related Work

The ETSI NFV standardization group has created the baseline architecture and related standards to enable the development of NFV MANO systems [1].

Most well supported open source MANO systems, such as OSM [2], OpenBaton [3] and SONATA [6], implement the ETSI NFV MANO models for describing the Virtual Network Functions (VNFs) and Network Services (NSes). However, they are designed to work in a single network domain environment since there is also a lack of standards that either model the multi-domain NSes or define the interfaces in a multi-MANO communication.

Recently, some efforts have been made to create a multi-domain orchestration solution for 5G networks, most notably in the 5GEx [7] and X-MANO [8] projects. 5GEx relies on a peer-to-peer interaction of multiple Multi-domain Orchestrators (MdOs), where each one is administered by an operator, to deploy services end-to-end. Each MdO further interacts with domain orchestrators which consist of SDN or NFV technologies that are responsible for the orchestration of a network segment within an operator domain. The MdOs can potentially add a performance overhead when orchestrating multiple dynamic network services across multiple testbeds. Unfortunately, the implementation and evaluation of 5GEx [7] is not available to be compared with the 5GUKEx.

X-MANO creates a cross-domain Management and Orchestration platform. The X-MANO architecture introduces Federation Agents (FAs) which provide resource availability in a domain to the Federation Managers (FMs). The FMs can in return work in a peer-to-peer manner with other FMs if needed to orchestrate the network services across multiple FMs. X-MANO does not describe the implementation details and experimental results, which makes it is difficult to compare the performance of the solutions. Furthermore, the authors [8] do not focus on the inter-domain connectivity solutions, which we believe is one of the most important aspect of the multi-domain orchestrators.

A recent survey in Network Services Orchestration [9] compare various single and Multi-domain Orchestration efforts with pros and cons of each architecture. Since the standards are yet to be finalized for MdOs, the various architectures bring a diverse spectrum of features. Defining an MdO is difficult, compared to the single domain orchestrators, as there is a cross-domain information exchange involved in the process [9]. In 5GUKEx, we address this issue by introducing a thin hierarchical multi-domain orchestration layer which builds on top of existing MANO systems and performs only service orchestration and interconnection, whereas resources are managed and controlled by the individual operators. We assume that the collaborating operators would like to hide their operator specific details and control of their network. Therefore, the research work discussed in [7, 8] differs from 5GUKEx in aspects of privacy and resource handling.

Deployment of NSes across datacenters using network slicing in a single administrative domain has already been extensively researched and efficiently implemented [10]. More recently, work [11] on a similar hierarchical architecture focusing on the end-to-end multi-domain network slicing and resource abstraction has been introduced. We believe 5GUKEx is an orthogonal solution to create multi-domain NS when compared to the 5GEx and X-MANO which can utilize the slicing solutions provided by [11] as well as the multi-site orchestration solutions [10]: more than creating a thin orchestration layer, one of the main objectives of 5GUKEx is to create a plug-n-play system which enables the deployment of use-cases over varied network technologies.

In addition to 5GEx and X-MANO, recent research projects [12, 13, 14, 15] have been funded under EU H2020 focused on creating an end-to-end network ecosystem supporting deployment and management of services across distributed testbeds. 5GVINNI [13] is working on creating an ecosystem for a multi-operator end-to-end 5G facility. It includes a testing and validation framework in addition to its multi-domain network orchestrator [16, 17] and employs Openslice [18] for the Network Service descriptor (NSD) and VNF descriptor (VNFD) onboarding, but does not describe the interconnection between multiple sites. 5GINFIRE [12] demonstrated a few multi-site orchestrated experiments where interconnection was done using VPN links between the participating sites [19]. Similar to 5GVINNI, 5GEVE [15] is creating an ecosystem to interconnect multiple sites across Europe with testing and validation framework along with intent based APIs. It follows an approach akin to 5GINFIRE and establishes VPN tunnels for setting up the datapath between multiple sites [20, 21]. 5GENESIS [14], which is another EU H2020 project, focuses on validation of the 5G KPIs for multiple 5G use-cases by creating an end-to-end 5G facility. For the Management and Orchestration of services across testbeds, 5GENESIS employs a MANO layer with a Slice Manager (SM) and a Network Management System (NMS). While SM provisions the interconnection, the NMS manages the resources of each testbed which differentiates it in principle from the 5GUKEx as explained above. For interconnection, 5GENESIS has proposed SD-WAN but the MdO architecture and implementation details are yet to be published.

The architectures discussed in [16] and [20] are extensive and provide a vast distributed 5G ecosystem for the verticals. However, the end-to-end interconnection is mostly performed by creating VPN tunnels. 5GUKEx, on the other hand, allows experimentation with multiple underlying network connectivity (including multi-technology interconnections), as further illustrated by the experiments in this paper (which focus on Layer-2 and Layer-0/optical interconnection). Secondly, these platforms have access to the resources of the individual testbeds, either by allowing the testbed monitoring or by allowing end users to upload their descriptors and VNF images through the portal. 5GUKEx avoids these by adopting the bottom-up approach, where the NSDs and VNFDs are encrypted and securely published to the catalog only by the testbeds preserving each testbed's independence and scope; and simplify the multi-level orchestration logic.

Table 1 compares a few features of the major MdOs, which reflects most of the efforts based on ETSI framework and takes in account the resource confidentiality of each testbed connecting to the MdO. However, as evident from Table 1, most of the MdOs do not support encryption of message and NS exchange with the local orchestrators and support only L3 overlay networks or other kinds of VPN technologies.

The proof-of-concepts [23, 24] show the feasibility of this approach and in further sections, we will discuss the complete architecture in greater detail. This approach facilitates the collaboration among the operators, since they can preserve corporate infrastructure information and use their SDN solutions and NFV-MANO implementations without disruption. For instance, CORD (Cen-

Multi-Domain Orchestrator	Resource Confidentiality	Network Technologies	Plug-n-Play	ETSI Compliant	Secured NS Transmission
5GUK Exchange	✓	L2, L0	✓	✓	✓
5GVINNI [13]	✗	L3 (Public IP)	✗	✓	✗
5GEVE [15]	✓	L2 VPN, L3 VPN	✗	✓	✗
5GINFIRE [12]	✓	L3 VPN	✗	✓	✗
5GEx [7]	✓	L3 (VxLAN)	✓	✓	✗
5GENESIS [22]	N/A	SD-WAN	N/A	✓	✗
X-MANO [8]	✓	No standard technology	✗	✗	✗

Table 1: Multi-Domain Orchestrator Comparison. (**N/A** *denotes "information not available" by the time this paper was written.*)

tral Office Re-architected as a Datacenter) re-architects the telco central offices (CO) by not only disaggregating the existing hardware devices but also enabling the collection of services, including access for residential, mobile, and enterprise customers [25]. It combines SDN, NFV, and elastic cloud services to build cost-effective, agile access networks. In CORD, computing instances, including VMs and containers, can be created and provisioned by OpenStack/Docker, SDN controller can provide control plane services, and multi-tenant services can be orchestrated by XOS framework [26]. Being built upon a suite of Open Networking Foundation (ONF) [27] projects that are part of the CORD project umbrella, the Converged Multi-Access and Core (COMAC) open source project brings convergence to operators' mobile and broadband access and core networks [28]. By leveraging 5GUKEx as the top hierarchical multi-domain orchestrator and utilising COMAC as a platform enabling user plane and control plane convergence for multiple access technologies at the edge, multi-domain NS can be created and orchestrated where access, edge, core or public clouds and multi-domains can be interconnected.

3. 5GUK Exchange Architecture

The architecture of the 5GUKEx is illustrated in Fig. 1. It assumes that the 5G networks, here referred to as *Islands*, are individually orchestrated by ETSI-based *Island Orchestrators*, e.g., OSM, OpenBaton, etc., and are connected to the 5GUKEx exposing their network service catalogues. The 5GUKEx is a lightweight hierarchical inter-domain orchestration platform that performs

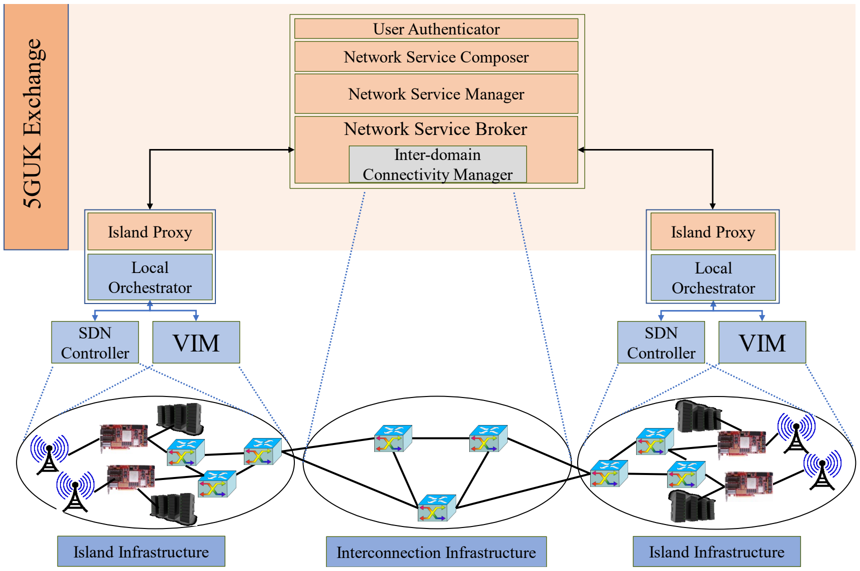

Figure 1: The 5GUK Exchange Architecture

mostly service orchestration. It delegates the heavyweight resource orchestration to the *Island Orchestrators* and interconnects the network services across the islands chaining together the running NSes in the individual islands. Along with the multi-domain NS deployment, 5GUKEx is designed to allow multi-network technologies to inter-operate using its plug-n-play design. As shown in Fig. 1, the Inter-Domain Connectivity Manager (IDCM) allows experimenters to plug-in varied underlying network technologies (e.g. packet, optical etc.) to create an end-to-end network. The 5GUKEx contains multiple components which are detailed as follows.

3.1. Island Proxy

The Island Proxy runs on top of the Island Orchestrators and serves as an intermediary between the 5GUKEx and the island orchestrator. The proxy serves as an isolation layer for security and policy purposes and is the main interface between islands and the 5GUKEx. As such, the proxy should expose a northbound interface compliant with the 5GUKEx API and a southbound interface compliant with ETSI MANO APIs. The requests from the 5GUKEx to the

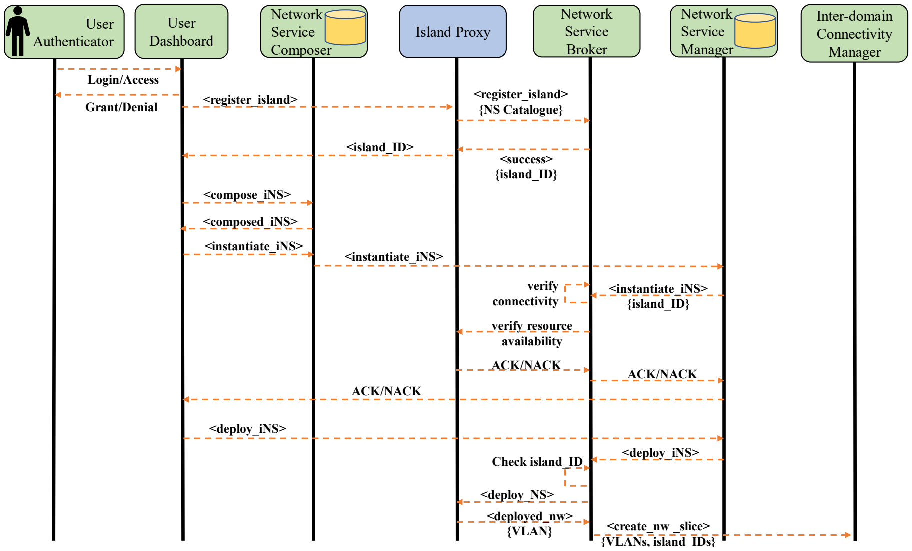

Figure 2: The 5GUK Exchange Control Flow

islands are received by the Island Proxy and forwarded to the Island Orchestrator, and the responses follow the opposite direction. The proxy handles the deployment and termination of running NSes on the local islands. Furthermore, it exposes Network Service Catalogues, i.e., the available network services at the local island in the form of ETSI MANO NSDs and optionally, the VNFDs, to the 5GUKEx during the registration of an island. During the deployment of an NS, the Island Proxy can operate in two ways to create the local island network using the local SDN controller. If the VLAN information and network endpoints are shared offline between the islands and 5UKEx, the deployed local network is called $StaticNetwork$; else if this information is shared while deploying the NS online, the deployed local network is called $DynamicNetwork$. Both approaches have some trade-offs which we will discuss in Sec. 5.

As shown in Fig. 2, the Island Proxy is the main piece of the architecture connecting islands to the 5GUKEx using the $\langle register\_island\rangle$ message received from the user. $\langle register\_island\rangle$ message forwarded from Island Proxy message contains the NS Catalogue with all NSes that the registering island can support. On successful registration, it receives a $\langle success\rangle$ message with a unique

$island\_ID$. During the NS instantiation and deployment phase, NS Broker communicates with Island Proxy to verify the available resources. On successful deployment, it shares the network VLAN information using $\langle deployed\_nw \rangle$ message with NS Broker to deploy the end-to-end network.

3.2. Network Service Broker (NSB)

The NSB interacts with the Island Proxy, implementing a common API among all the Island Orchestrators, based on the ETSI MANO NSDs/VNFDs and their elements. It acts as the intermediary between the Island Proxy and 5GUKEx. All messages to and from the 5GUKEx are passed over by the NSB. As shown in Fig. 2, during the island registration, Island Proxy sends the $\langle register\_island \rangle$ message to NSB which creates a unique ID for the registering island and sends it along the $\langle success \rangle$ message to the Island Proxy. During the inter-island Network Service (iNS) instantiation phase, it receives an $\langle instantiate\_iNS \rangle$ message from Network Service Composer. After extracting the *island_ID* from the iNS request, it checks the connectivity status of the island to 5GUKEx and also requests the Island Proxy to get the resource availability status. If the resources are available, it gets the acknowledgement ($\langle ACK \rangle$) message, implying that the requested island can run the requested NS. If the connectivity to the island is not established or if the resources are not available, the Island Proxy returns a negative acknowledgement ($\langle NACK \rangle$) message changing the status of the requested iNS to be unavailable. Once all the requested islands are capable to deploy and run the requested NSes, the user can deploy the iNS by invoking Network Service Manager to send $\langle deploy\_iNS \rangle$ message to the NSB, which in turn extracts the *island_ID* to send the $\langle deploy\_NS \rangle$ request to the respective islands. Once NS is deployed, it sends the network endpoints information as a set of VLAN ID, switch port and island ID using $\langle create\_nw\_slice \rangle$ request to Inter-Domain Connectivity Manager to create the network slice between the two connecting islands.

3.3. Inter-Domain Connectivity Manager (IDCM)

The IDCM module has two responsibilities: (a) serving as a bootstrapping point by setting up the control plane of the island and connecting it to the 5GUKEx and secondly, (b) being responsible for creating the datapath between the islands when an iNS is deployed. The IDCM employs an SDN controller to create the end-to-end sliced Layer-2 network and at the same time has a database to store the endpoint and connections to monitor and terminate the connections when the iNS is terminated. As shown in Fig. 2, as soon as the NS is deployed on all the islands and ⟨*create_nw_slice*⟩ request with all the endpoint VLANs with island_IDs are received from the NSB, the IDCM creates the end-to-end L2 network between the two endpoints to create the required datapath.

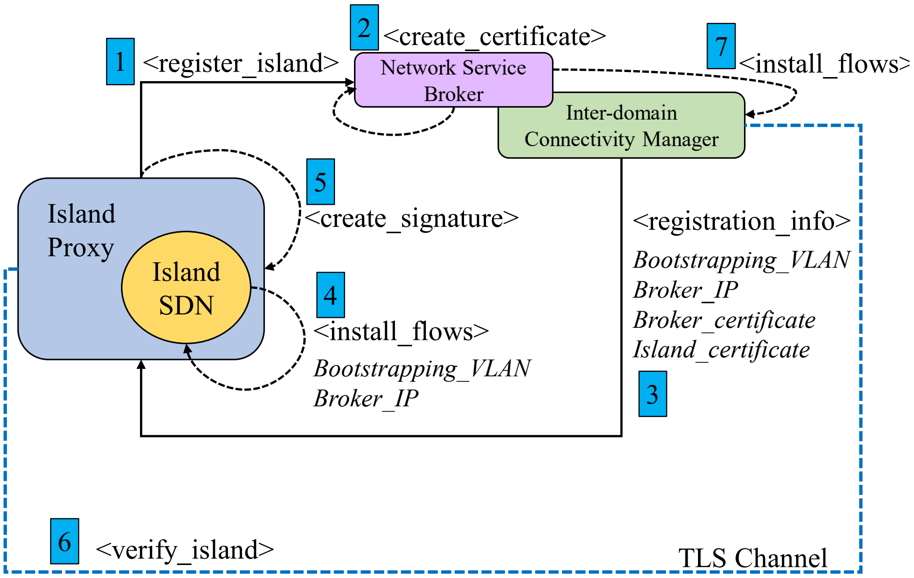

Figure 3: Island bootstrapping procedure

During the bootstrapping phase, upon receiving the ⟨*register_island*⟩ message, the NSB acts as the Certification Authority and generates the certificate and keys for the registering island. It requests the IDCM to do the island bootstrapping, which in turn allots the bootstrapping VLAN for the registering island. As shown in Fig. 3, along with the generated certificates and broker IP, it sends the bootstrapping VLAN to the registering island to create the bootstrapping control path. The SDN controller on the registering island installs the appropriate flows using the broker IP and the bootstrapping VLAN. Using the keys, it creates a client signature to be used during the verification process. The island then uses the generated certificate and the public key to verify the connection. Using Public Key Encryption (PKI) over the TLS channel, the island verification is completed before the flows are installed by IDCM to establish the control flow path.

The novelty of IDCM, as compared to other solutions discussed in [8, 7, 16, 19, 20, 29], is the two step process where the island registration is secured using the bootstrapping process and the NS deployment request is also encrypted using the public key of individual islands. Additionally, the datapath creation allows multiple options to create either Layer-2 VLANs or Layer-0 optical links. This also allows 5GUKEx to host innovative end-to-end experiments.

3.4. Network Service Manager (NSM)

The NSM is responsible for the life-cycle management of an inter-island Network Service (iNS). It stores the NS catalogues of the registered islands that the Network Service Composer can access them. It interacts with the NSB for requesting an iNS deployment in the islands and getting island responses about the deployment status and the network endpoints used by the running services which then the IDCM uses to dynamically interconnect the services. Using similar steps of interaction, the NSM can terminate a running inter-island NS. The NSM also provides information to users about the running services on the islands and relevant coarse-grained monitoring data. It can also be used to on-board or update NSes or VNFs given that such actions are allowed from an island policy. As shown in Fig. 2, NSM is first requested by the Network Service Composer to $\langle instantiate\_iNS \rangle$. The NSM first verifies the connectivity and available resources with the islands involved in the requested iNS by relaying the $\langle instantiate\_iNS \rangle$ request with the island_ID to NSB. On receiving an

$\langle ACK\rangle/\langle NACK\rangle$ message, it sends the appropriate acknowledgement message to the user. NSM receives the $\langle deploy\_iNS\rangle$ message to deploy the instantiated iNS which stores the requested iNS in its database before sending the request to the NSB.

3.5. Network Service Composer (NSC)

The NSC enables users of the 5GUKEx (developers, experimenters or service providers) to create inter-island NSes by combining and optionally modifying as necessary, the available NSes/VNFs of the islands. The composition results in templates of inter-island services that the user can choose to deploy. During the composition, the selected NSDs from each island are combined and appended with the island ID (obtained during island registration) to create the resulting iNS as a YAML file, similar to an ETSI standard NSD. As shown in Fig. 2, the NSC is the first point of interaction when a user tries to deploy an iNS. Using the Dashboard, experimenter can choose the NS from multiple islands and send the $\langle compose\_iNS\rangle$ request to NSC. NSC combines the requested NS descriptors and logical endpoints to create a iNS descriptor. On receiving the $\langle composed\_iNS\rangle$ message, user can instantiate the iNS by sending the $\langle instantiate\_iNS\rangle$ request to NSC. As discussed earlier, the request is relayed to NSM for further operations. Similar to NSM, NSC also maintains its own database to store the composed templates of iNS, which further allows new users to deploy the same iNS in future if needed as the information is available on the Dashboard. NSC also provides the REST APIs for automation and programmability to the users of 5GUKEx. Irrespective of the NS composition procedure (API or Dashboard), all the composed iNS are available to the users's dashboard.

4. Inter-island Network Service Deployment Procedure

As discussed before, the 5GUKEx provides means for its users to deploy an end-to-end network service encompassing different operator domains (in this context, we assume user to be the operator deploying the inter-island NS). As

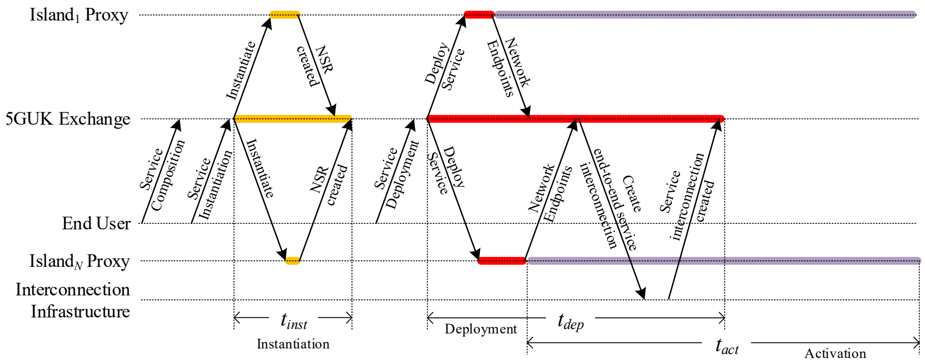

Figure 4: Network Service Deployment Procedure in the 5GUK Exchange

shown in Fig. 4, this procedure consists of the user first selecting different network services of different islands to compose an inter-island NS on the Network Service Composer module on the 5GUKEx. Then, the user requests the instantiation of the inter-island service, triggering the 5GUKEx to instantiate on the corresponding islands the individual network services which are part of the composed inter-island NS. The islands verify if they have enough local resources, e.g., compute, storage and memory, to deploy and start the individual network service. If there are available resources, each island creates a Network Service Record (NSR) and then sends the information to the 5GUKEx to signal that the request can be fulfilled. When the 5GUKEx receives the responses from all the relevant islands, it informs the user that the islands are ready to deploy the inter-island NS end-to-end. The instantiation time is referred to t_{inst}.

Once the user chooses to deploy the inter-island NS, the 5GUKEx contacts the islands to deploy the previously instantiated NSR. Upon receiving the deployment message by the 5GUKEx, the local island proceeds to deploy the network service using the Island Orchestrator. Once the service is being activated, in $DynamicNetwork$ mode, the local island provides information to the 5GUKEx about the network endpoints to be used at the island gateway. Else, if it is operating in the $StaticNetwork$ mode, it picks the endpoints from the database which are already agreed upon between the 5GUKEx and the island. Once the endpoints are received from all the islands, the IDCM module of the 5GUKEx creates the underlying data plane network service interconnection across the islands. After the inter-island service interconnection is provisioned, the deployment procedure is finished and the user is notified. The time for deploying an iNS is referred to t_{dep}. Meanwhile, the network service activation is carried out by the local islands and takes t_{act} time i.e., from the time that a service is deployed until it becomes active.

5. Implementation and Performance Evaluation

5.1. Implementation

The 5GUKEx architecture shown in Fig. 1, has been implemented as a series of distributed components written in the Python programming language. These components communicate mainly through REST APIs, although TCP sockets form the basis of an asynchronous notification system. On the south-bound, all the external components are interfaced through adapter modules to maximize flexibility while allowing the system to be extended on demand to support new technologies. The current implementation includes an ETSI-based adaptor (compatible with OSM) for connection via Island Proxy and an OpenDaylight (ODL) controller adaptor that enables the IDCM to dynamically control the interconnection infrastructure among the different islands.

5.2. Performance Evaluation Setup

To evaluate the performance of the 5GUKEx, we emulate the 5GUKEx and 4 islands. We utilize 5 Dell PowerEdge T360 servers, each equipped with Intel Xeon E5-2680 CPU having 56 cores and 64GB of RAM, running Ubuntu 16.04 as the host operating system. One server hosts the 5GUKEx and each of the remaining servers emulates a local island; each island consists of the ETSI NFV-compliant OSM as the local island orchestrator and OpenStack for management of the compute resources. Each server also hosts an instance of the ODL controller which controls the network resources of each island. We use 2 Corsa DP2100 OpenFlow switches, for the network data plane of all the islands and the 5GUKEx, which are interconnected. To emulate the network resources of the islands, a Corsa SDN switch is shared among the islands by reserving dedicated ports per island. For the interconnection infrastructure of the 5GUKEx, the second Corsa switch is used. All the islands have identical server, switch port and switch bridge configurations.

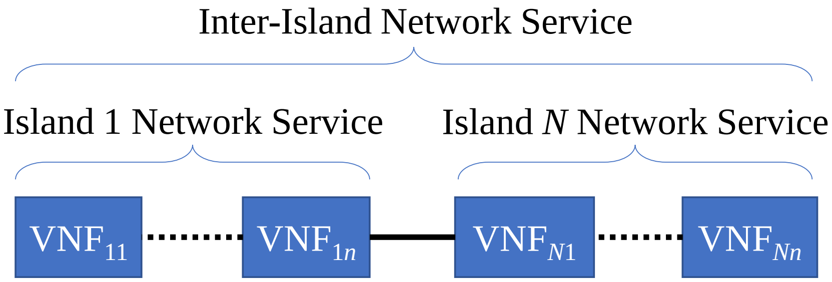

Figure 5: VNF chain spanning multiple islands

5.3. Experiment

We use the 5GUKEx to instantiate and deploy iNSs across multiple islands and measure the instantiation time (t_{inst}) and the deployment time (t_{dep}) for both $StaticNetwork$ and $DynamicNetwork$ modes, as well as the local island orchestration time, also referred as activation time (t_{act}). Each island exposes the same NSDs and VNFDs that the user composes to form an iNS by selecting a number of NSDs from multiple islands. An NS at each island consists of n connected VNFs and the NSs of the islands are stitched to each other by the inter-island network infrastructure using L2 connectivity to create the iNS. This results in an iNS that chains together multiple VNFs across multiple islands as shown in Fig. 5. Each VNF consists of a CirrOS which is a minimal Linux image used here as a baseline VNF [30].

To evaluate the performance of the 5GUKEx and its promise for lightweight inter-domain orchestration, since the 5GUKEx delegates the resource orchestration to the local islands, we compare t_{act} to t_{inst} and t_{dep}.

5.4. Results

We first deploy an NS consisting of n connected CirrOS VNFs at each local island without using the 5GUKEx to measure the local island network service activation time t_{act}. We run the tests for 20 times and the results are shown in Table 2 with 95% confidence. The time taken for a service to be active is at least 27.10 seconds for the case of a network service consisting of one VNF.

Table 2: Network Service Activation Time at each Island

Number of VNFs n **per Island**	**Time until activation [sec]**
1	27.10 ± 1.04
2	43.25 ± 1.40
3	64.66 ± 2.24
4	89.59 ± 1.56

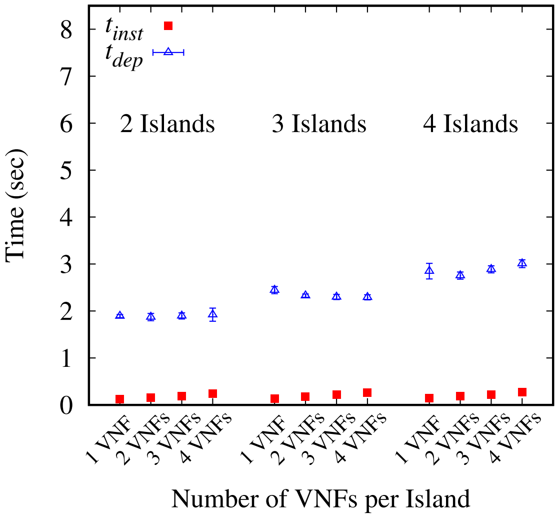

Figure 6: 5GUKEx Instantiation and Deployment Times - $StaticNetwork$

We then deploy an iNS from 5GUKEx that consists of the same network service as before, now deployed at multiple islands. We run sets of 20 tests,

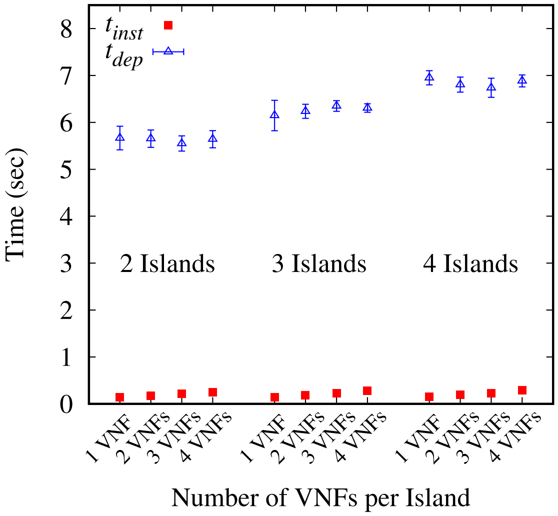

Figure 7: 5GUKEx Instantiation and Deployment Times - $DynamicNetwork$

modifying the number of VNFs per NS and the number of islands used on the inter-island service in each set; and we measure the instantiation t_{inst} and deployment t_{dep} times, as shown in Figs. 6 and 7 for the $StaticNetwork$ and $DynamicNetwork$ cases respectively. For the t_{dep}, 95% confidence intervals are presented; whereas for the t_{inst}, only the average is presented since the 95% confidence intervals are very small (in all cases within 5% accuracy). In contrast to the activation time t_{act}, the t_{inst} and t_{dep} times are minimal. The t_{act} for a network service containing four VNFs per island requires 89.59 seconds, whereas the deployment time t_{dep} at the 5GUKEx even across four islands takes only about 7 seconds for the $DynamicNetwork$ case. This is due to the fact that the 5GUKEx performs service orchestration and minimal network (re)configurations of the inter-island network infrastructure and because it delegates the performance-heavy orchestration of computational resources on the local islands. This shows that the 5GUKEx is a thin layer of orchestration with minimal overhead, consequently contributing towards the sustainability of the 5GUKEx platform.

Regarding the deployment time t_{dep}, we see that it increases when network services are deployed across more islands. This is due to the fact that more network flows are installed on the inter-island network to steer the relevant traffic accordingly. Furthermore, we see that t_{dep} remains the same with increasing number of VNFs per NS deployed at an island. This is due to the nature of the 5GUKEx being a thin orchestration layer as described earlier. In addition, the instantiation time t_{inst} is negligible compared to the deployment time t_{dep}. By adding the t_{dep} and t_{inst} times and comparing them to the t_{act} at each island, we can see that the total inter-domain orchestration overhead is minimal.

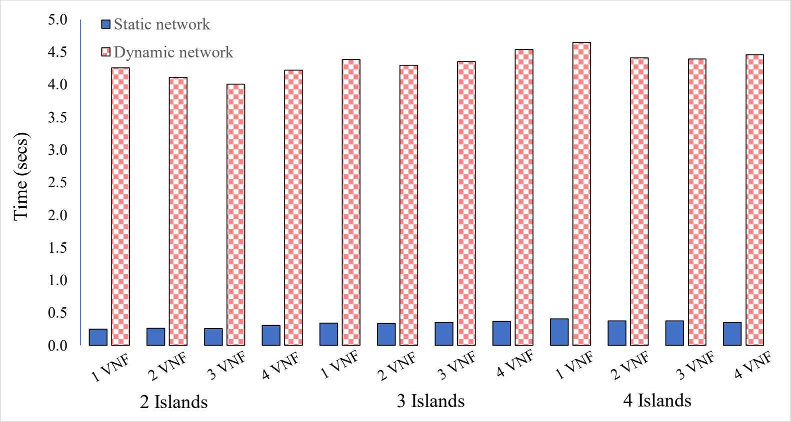

Figure 8: Network establishment time (Static vs Dynamic)

Comparing the results of Figs. 6 and 7, we observe that t_{inst} is the same in both $StaticNetwork$ and $DynamicNetwork$ cases since the process requires the local island to check if it has enough resources to deploy an NS; whereas, the t_{dep} in the $DynamicNetwork$ case increases as compared to the $StaticNetwork$. This is due to the fact that the Island Proxy deploys the local network at the island before sharing the VLAN information and endpoints, as compared to the $StaticNetwork$ where the local island network information is shared offline between the islands and 5GUKEx as stated in Sec. 3.1. Furthermore, we present the local island network establishment time for both cases in Fig. 8, which clearly shows that the $DynamicNetwork$ case requires more time due to the run-time selection of available VLAN for the L2 network at the local island. Results show that this time is the same for all combinations of number of VNFs and number of islands. It is due to the fact that in each case, one L2 network per island spanning across the inter-island infrastructure is deployed during the NS deployment phase.

6. Trials and Experiments using 5GUK Exchange

6.1. Large area musical orchestra trials

A major focus of 5G networks is to guarantee low latency and QoS for the applications and end-users. In order to realise it end-to-end, we conducted a unique trial involving musicians to conduct an orchestra across three locations in U.K. [24]. The three sites, University of Bristol (UoB), Digital Catapult (DC) (in London) and King's College London (KCL) hosted three Network Services using Open Source MANO [2] on each site as local orchestrator and 5GUK Exchange as the inter-domain orchestrator as shown in Fig. 10. As discussed in [24], the full-HD video and audio was transmitted across the three sites and was synchronized using Soundjack [31] and Ultragrid [32].

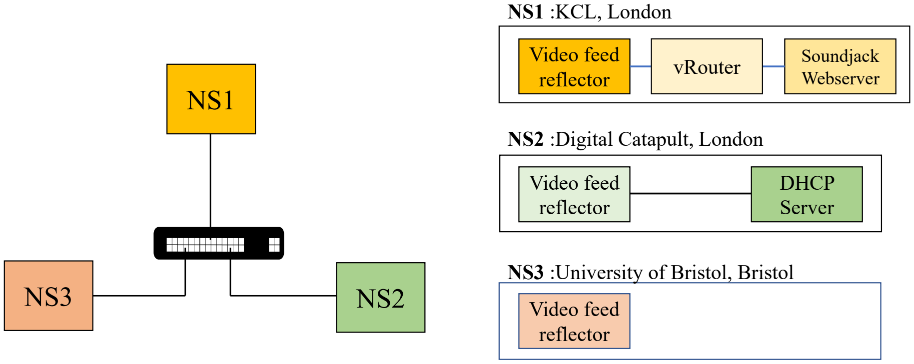

Figure 9: Musical Orchestra: VNF and NS setup

The setup of the experiment is shown in Fig. 10 with the network services shown in Fig. 9 distributed across the three sites. As shown in Fig. 9, the KCL island exposed the NS composed of Video feed reflector, virtual Router

Table 3: Network and Application latency during trials

Site 1	**Site 2**	**End-to-end Network Latency (RTT)**	**End-to-end Application Latency (RTT)**
University of Bristol	King's College London	5.6ms	21.0ms
University of Bristol	Digital Catapult	5.9ms	21.3ms
King's College London	Digital Catapult	4.6ms	20.7ms

(vRouter) and Soundjack Webserver as VNFs. The role of video reflector was to create a mirror of the video feed from each island to be distributed across all the connected sites using Ultragrid. vRouter was hosted in KCL as it was the gateway of the Layer 3 overlay network created on top of the L2 connectivity provided by the 5GUKEx and allowed services to access internet using the KCL node. Since KCL was the gateway, we hosted the Soundjack webserver at KCL as it follows a server-client model for participants to connect; once connected, the participants operate in a peer-to-peer manner. Similarly, all the devices connected in the network received their IP address lease from the DHCP server hosted at Digital Catapult which also deployed the Video feed reflector to broadcast its video feed from Ultragrid software. Lastly, the NS hosted at University of Bristol consisted only of a single VNF i.e. Video feed reflector to send its video feed across to other islands. The experiment involved musicians connected to the network at each site, where the audio and video feed was shared among the sites using Soundjack and Ultragrid respectively. Since we used two separate softwares for audio and video, the synchronization was more critical and hence, it was critical to keep the latency down to few milliseconds.

Table 3 shows the end-to-end network and application latency observed during the trial. Since we were able to keep the network latency around 5ms and hence the application latency below 30ms, consequently there was no noticeable lag observed during the experiment.

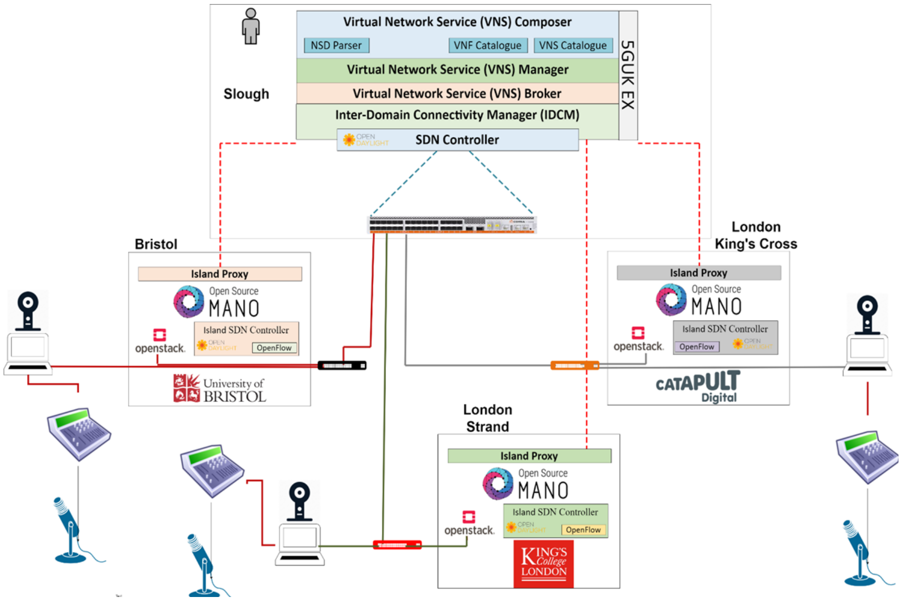

Figure 10: Orchestrating the Orchestra Trials [24]

6.2. *Using optical transport interconnection*

Optical networks are still one of the most preferred medium of transport where latency is an important parameter. Various efforts have been made and concepts have been introduced to slice the optical networks [10, 33]. Concepts like SLICE [33] show the need and flexibility of optical networks. As discussed in Sec. 3, 5GUKEx architecture allows us to plug-in such a system as the transport technology, further enhancing the experimentation and support to deploy cross-domain QoS intensive use-cases. Leveraging the modular capability of the 5GUKEx, we replaced the datapath establishment procedure by using an optical connection composed of multiple optical devices instead of an L2 link, as described in [34]. Detailed experiment setup and further details are included in [34].

While comparing the deployment of an iNS over 4 islands where each island hosted 1 VNF, we observe that the network establishment takes a longer time on an optical testbed compared to just creating the L2 data path. As shown

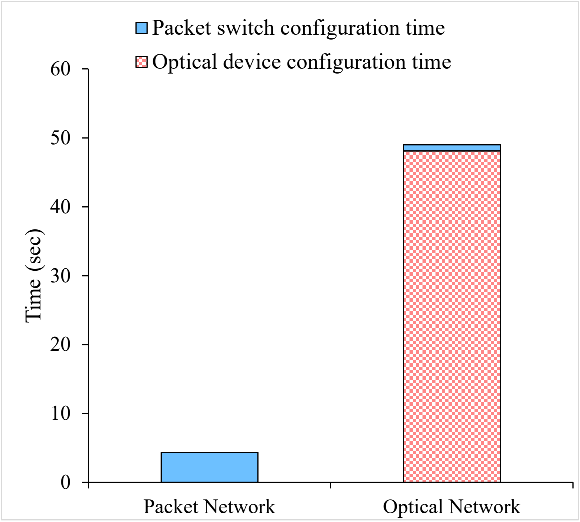

Figure 11: Network establishment time comparison (Optical vs Packet)

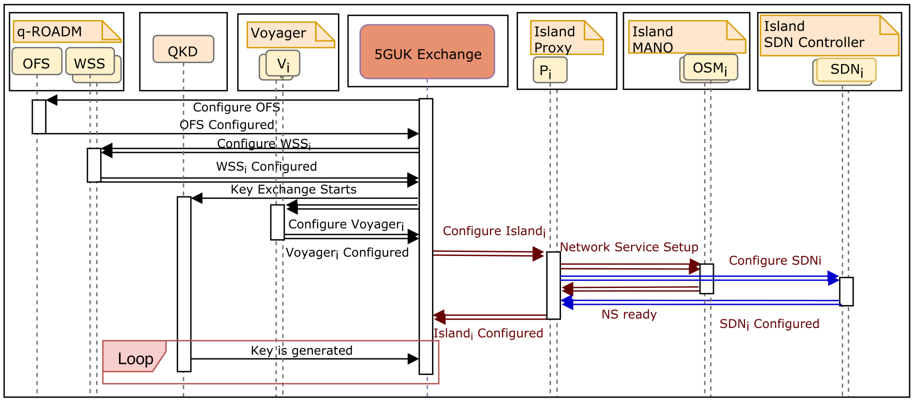

Figure 12: Quantum experiment sequence diagram [34]

in Fig. 11, configuring a datapath involving optical equipment takes approximately 9 times the time taken to configure just the packet network. Fig. 12 shows the sequence of control messages and the involved optical devices in the setup. Voyager is a combination of packet switch and the optical transponders, and 5GUKExchange also configured q-ROADM and the Quantum Key Distributor (QKD) in the process. The major difference in the observed network establishment time in Fig. 11 is due to the fact that the optical transponders used at each island in this experiment take a longer time to create the data path as it involves changing the modulation format as well as the wavelength of the corresponding optical port. The experiment proves the capability of the 5GUKEx to operate over multiple underlying technologies.

7. Conclusions and Future Work

In this article, we presented the 5GUK Exchange, a hierarchical multi-operator platform that aims to orchestrate end-to-end network services in a sustainable manner. The 5GUKEx builds a multi-operator API that is based on ETSI standards, allowing operators to integrate using their existing MANO systems, to hide any confidential infrastructure information and to provide flexibility in selecting any underlying SDN and NFV technologies. By brokering the orchestration of the individual network services to operators, the 5GUKEx becomes a lightweight solution in performance and complexity that performs multi-operator coordination and service interconnection.

We discussed how the 5GUKEx aims to build an open ecosystem that offers a range of diverse services without any operator boundaries and how it enables collaboration among operators and other 5G stakeholders to fulfil the 5G end-to-end vision. We presented the architecture of 5GUKEx; we implemented a prototype based on the service catalogues of ETSI NFV MANO-compliant solutions and evaluated its performance showing that the multi-domain orchestration layer of the 5GUKEx has minimum operational overhead. We discussed the flexibility of the architecture by two sets of experiments focusing on L2 interconnection network as well as the L0 optical network. As a future work, we plan to upgrade the design based on ETSI NFV Release 3 [35] and also integrate NFV MANO systems that are based on TOSCA service catalogues, such as ONAP, to the 5GUKEx. Intent based end-to-end service deployment would be another feature we plan to add on 5GUKEx. We also plan to experiment with layer-3 inter-connectivity in our 5GUKEx deployment. At last, we aim to integrate the 5GUKEx to a marketplace using Machine Learning and Artificial Intelligence frameworks to achieve our vision for MdO.

Acknowledgments

This work has received funding from the UK: DCMS UK 5G Testbeds & Trials Programme, EPSRC projects INITIATE (EP/P003974/1) and TOUCAN (EP/L020009/1) and from the EU: H2020 projects Metro-Haul, MATILDA, 5G-VICTORI (grant agreements 761727, 761898, 857201).

References

References

[1] ETSI, Network Functions Virtualisation (NFV); Management and Orchestration, ETSI GS NFV-MAN 001 (V1.1.1), Tech. rep. (2014).

[2] ETSI, OSM, `https://osm.etsi.org`, Accessed: 15-10-2019.

[3] Open Baton community, Open Baton: An extensible and customizable NFV MANO-compliant, `http://openbaton.github.io`, Accessed: 15-10-2019.

[4] ONAP Community, ONAP Architecture Overview, `http://www.onap.org`, Accessed: 15-10-2019.

[5] 5GPPP, 5G Vision, `https://5g-ppp.eu/wp-content/uploads/2015/02/5G-Vision-Brochure-v1.pdf`, Accessed: 15-10-2019.

[6] H. Karl, S. Dräxler, M. Peuster, A. Galis, M. Bredel, A. Ramos, J. Martrat, M. S. Siddiqui, S. van Rossem, W. Tavernier, G. Xilouris, DevOps for network function virtualisation: an architectural approach, Transactions on Emerging Telecommunications Technologies 27 (9) (2016) 1206–1215.

[7] C. J. Bernardos, B. P. Gerö, M. Di Girolamo, A. Kern, B. Martini, I. Vaishnavi, 5GEx: realising a Europe-wide multi-domain framework for

software-defined infrastructures, Transactions on Emerging Telecommunications Technologies 27 (9) (2016) 1271–1280.

[8] A. Francescon, G. Baggio, R. Fedrizzi, E. Orsini, R. Riggio, X-MANO: An open-source platform for cross-domain management and orchestration, in: IEEE Conference on Network Softwarization (NetSoft), 2017, pp. 1–6.

[9] N. F. S. de Sousa, D. A. L. Perez, R. V. Rosa, M. A. Santos, C. E. Rothenberg, Network Service Orchestration: A survey, Computer Communications 142-143 (2019) 69–94.

[10] A. Bravalheri, A. S. Muqaddas, N. Uniyal, R. Casellas, R. Nejabati, D. Simeonidou, VNF Chaining across Multi-PoPs in OSM using Transport API, in: Optical Fiber Communication Conference, 2019, p. W1G.4.

[11] T. Taleb, I. Afolabi, K. Samdanis, F. Z. Yousaf, On Multi-Domain Network Slicing Orchestration Architecture and Federated Resource Control, IEEE Network 33 (5) (2019) 242–252.

[12] 5GINFIRE, `https://5ginfire.eu`, Accessed: 27-02-2020.

[13] 5GVINNI (5G Verticals Innovation Infrastructure), `https://www.5g-vinni.eu/`, Accessed: 27-02-2020.

[14] 5GENESIS (5th Generation End-to-end Network, Experimentation, System Integration, and Showcasing), `http://www.5genesis.eu`, Accessed: 27-02-2020.

[15] 5GEVE (European Validation platform for Extensive trials), `https://www.5g-eve.eu`, Accessed: 27-02-2020.

[16] K. Mahmood, P. Grønsund, A. Gavras, M. B. Weiss, D. Warren, C. Tranoris, A. F. Cattoni, P. Muschamp, Design of 5G End-to-End Facility for Performance Evaluation and Use Case Trials, in: IEEE 2$^{\text{nd}}$ 5G World Forum (5GWF), 2019, pp. 341–346.

[17] 5GVINNI-Consortium, D1.1 Design of infrastructure architecture and subsystems (Dec. 2018).
URL `https://doi.org/10.5281/zenodo.2668754`

[18] Openslice, `https://openslice.readthedocs.io`, Accessed: 09-03-2019.

[19] A. P. Silva, C. Tranoris, S. Denazis, S. Sargento, J. Pereira, M. Luís, R. Moreira, F. Silva, I. Vidal, B. Nogales, R. Nejabati, D. Simeonidou, 5GinFIRE: An end-to-end open5G vertical network function ecosystem, Ad Hoc Networks 93 (2019) 101895.

[20] M. Gupta, R. Legouable, M. M. Rosello, M. Cecchi, J. R. Alonso, M. Lorenzo, E. Kosmatos, M. R. Boldi, G. Carrozzo, The 5G EVE End-to-End 5G Facility for Extensive Trials, in: IEEE International Conference on Communications Workshops (ICC Workshops), 2019, pp. 1–5.

[21] 5GEVE-Consortium, D1.3 5G EVE end to end facility reference architecture for vertical industries and core applications (Jan. 2020).
URL `https://doi.org/10.5281/zenodo.3628333`

[22] 5GENESIS Consortium, Deliverable D3.1, Management and Orchestration (Release A), Accessed: 27-02-2020.
URL `https://5genesis.eu/wp-content/uploads/2019/10/5GENESIS_D3.1_v1.0.pdf`

[23] D. Gkounis, N. Uniyal, A. S. Muqaddas, R. Nejabati, D. Simeonidou, Demonstration of the 5GUK Exchange: A Lightweight Platform for Dynamic End-to-End Orchestration of Softwarized 5G Networks, in: European Conference on Optical Communication (ECOC), 2018, p. TuDS.14.

[24] M. Shakir, F. Sardis, N. Uniyal, D. Gkounis, A. Carot, M. Dohler, D. Simeonidou, D. Kaleshi, 5GUK Trials: Distributing a Live Orchestra over 5G Architecture and Federated Resource Control, Global Communications, 2019.

URL https://www.comsoc.org/publications/ctn/5guk-trials-distributing-live-orchestra-over-5g

[25] L. Peterson, A. Al-Shabibi, T. Anshutz, S. Baker, A. Bavier, S. Das, J. Hart, G. Palukar, W. Snow, Central office re-architected as a data center, IEEE Communications Magazine 54 (10) (2016) 96–101.

[26] L. Peterson, S. Baker, M. De Leenheer, A. Bavier, S. Bhatia, M. Wawrzoniak, J. Nelson, J. Hartman, XoS: An extensible cloud operating system, in: Proceedings of the 2nd International Workshop on Software-Defined Ecosystems, ACM, 2015, pp. 23–30.

[27] ONF, Open Networking Foundation, https://www.opennetworking.org, Accessed: 04-12-2019.

[28] ONF Mobile Edge Cloud Ecosystem, Converged Multi-Access and Core (COMAC), https://www.opennetworking.org/comac, Accessed: 14-11-2019.

[29] B. Nogales, I. Vidal, D. R. Lopez, J. Rodriguez, J. Garcia-Reinoso, A. Azcorra, Design and Deployment of an Open Management and Orchestration Platform for Multi-Site NFV Experimentation, IEEE Communications Magazine 57 (1) (2019) 20–27.

[30] CirrOS, CirrOS in Launchpad, https://launchpad.net/cirros, Accessed: 15-10-2019.

[31] Soundjack, Soundjack, https://www.soundjack.eu/index.php, Accessed: 06-11-2019.

[32] Ultragrid, Ultragrid, http://www.ultragrid.cz/specification/, Accessed: 06-11-2019.

[33] M. Jinno, H. Takara, B. Kozicki, Y. Tsukishima, Y. Sone, S. Matsuoka, Spectrum-efficient and scalable elastic optical path network: architecture, benefits, and enabling technologies, IEEE Communications Magazine 47 (11) (2009) 66–73.

[34] R. Wang, R. S. Tessinari, E. Hugues-Salas, A. Bravalheri, N. Uniyal, A. S. Muqaddas, R. S. Guimaraes, T. Diallo, S. Moazzeni, Q. Wang, G. T. Kanellos, R. Nejabati, D. Simeonidou, End-to-End Quantum Secured Inter-Domain 5G Service Orchestration Over Dynamically Switched Flex-Grid Optical Networks Enabled by a q-ROADM, Journal of Lightwave Technology 38 (1) (2020) 139–149.

[35] ETSI, NFV Release 3 Description (v0.5.0), Tech. rep., Accessed: 06-04-2020 (2020).
URL `https://docbox.etsi.org/ISG/NFV/Open/Other/ReleaseDocumentation/NFV(20)0000016_NFV_Release_3_Description_v0_5_0.pdf`